COUPLING INFORMATION TRANSMISSION WITH AN APPROXIMATE MESSAGE PASSING RECEIVER

by

David Duncan McNutt

Submitted in partial fulfillment of the requirements
for the degree of Master of Applied Science

at

Dalhousie University
Halifax, Nova Scotia
May 2016

Dedicated to all of the energetic photons

Table of Contents

List of Figures

Abstract

Spatial graph coupling has been immensely successful in the field of error correcting codes. Motivated by this success the coupling approach has been implemented for more general communication formats, including multiple-access communications. We explore the application of spatial coupling to modulation of data streams, and employ a recently proposed approximate message passing compressed sensing technique to develop an algorithm for the demodulation of datastreams transmitted over the multiple-access additive white Gaussian noise channel and demonstrate that near-optimal sum-rate performance can be achieved.

List of Abbreviations and Symbols Used

AIS : Automatic Identification System

ALOHA : Not an acronynm; communication protocol

AMP : Approximate Message Passing

ASM : Application Specific Messages

AWGN : Additive White Gaussian Noise

BEC : Binary Erasure Channel

BER : Bit Error Rate

BPDN : Basis Pursuit Denoising

CDMA : Code Division Multiple Access

CRDSA ALOHA : Contention Resolution Diversity Slotted ALOHA

CS-TDMA : Carrier Sense Time Division Multiple Access

dB : Decibels

ECRA ALOHA : Enhanced Contention Resolution ALOHA

FDMA : Frquency Division Multiple Access

IDM : Interleaved-division Multiplexing

IDMA : Interleaved-division multiple access

IMO : International Maritime Organization

IRSA ALOHA : Irregular Resolution Slotted ALOHA

ITU : International Telecommunications Union

LASSO : Least Absolute Shrinkage and Selection Operator

LDPC : Low Density Parity Check

LDPCC : Low Density Parity Check Codes

LP : Linear Programming

MAC : Multiple Acces Channel

MAP : Maximum a Posteriori Probability

MIMO : Multiple Input Multiple Output

MSE : Mean Squared Error

MMSE : Minimum Mean Squared Error

OMP : Orthogonal Matching Pursuit

PS-CDMA : Partitioned Spreading Code Division Multiple Access

S-AIS : Satellite Automatic Identification System

SO-TDMA : Self Organizing Time Division Multiple Access

sup : supremum

TDMA : Time Division Multiple Access

VDES : Very High Frequency Data Exchange System

VHF : Very High Frequency

$\boldsymbol{A}$: Sensing Matrix

α : Modulation Load or System Load

$\boldsymbol{b}_l$: Preamble Information Bits for l^{th} user

$\mathcal{C}$: Capacity

$\boldsymbol{C}$: Coupling Matrix

$\boldsymbol{d}$: A Pseudo-Random Binary Sequence

$\boldsymbol{H}_t$: Channel Matrix at Time t

K or L : Number of Users

M : Repetition Factor

N : Length of Encoded Binary Sequence v_l

$\boldsymbol{n}, \boldsymbol{w}$ or $\boldsymbol{\xi}$: Additive Gaussian Noise Term

P : Total Power of All Users

P_l : Power of l^{th} User

$\boldsymbol{P}_l$: Permutation Matrix for l^{th} User

R : Rate

$R_{i,t}$: Covariance Matrix for i^{th} Iteration at Time t

$\boldsymbol{S}_l$: Signature Sequence Matrix for l^{th} User

σ : Noise Variance

θ_l : Time Offset for l^{th} Group of Users

$\boldsymbol{v}_l$: Encoded Binary Sequence of l^{th} User

W : Coupling Window

$\boldsymbol{y}$: Multi-User Received Signal

Acknowledgements

First and foremost, I must thank my supervisor Dmitry Trukhachev for his patience and knowledge. For very similar reasons I would also like to thank my parents Vicki Steeves and Doug McNutt; as a rule, mathematicians are hard people to deal with.

While completing this degree, I know I could not have done so without my friends. In particular I'd like to thank Sam Baker, Alisha Dawn Cobham, Paul Dickson, Rachelle Gammon, Katherine Hurley, Sara Lawlor, Candace MacIntosh and Andy Sears for their friendship, support, and advice.

Chapter 1

Introduction

The focus of this thesis is on a low-complexity implementation of the coupling information transmission multi-user communication technique based on the spatial graph coupling principle [3]. Each user transmits one or more independently encoded data streams over the multiple access channel. Each data stream is encoded and modulated such that the relations between the data bits and modulation symbols within each transmitted stream are represented by a bipartite graph. The data streams are transmitted with time offsets enabling coupling of the transmitted graphs into a single continuous graph chain at the receiver. Modulation of each data stream is accomplished with repetition, permutation, and spreading of data.

The principle of spatial graph coupling [4, 5, 6] which originated in the area of low-density error-correction codes, has recently attracted significant research attention. Construction of coupled graph models on which sub-optimal iterative message-passing decoding can asymptotically achieve the optimal maximum a posteriori probability (MAP) decoding limit has now been applied in a number of fields including multiple-access communications. Regular (un-coupled) random code-division multiple access (CDMA) [7] as well as its sparse counterparts [8, 9, 10] can only achieve limited multi-user loads of equal power users. The application of spatial graph coupling removes this limitation [2, 11, 12], and in addition, can make the system capacity-achieving asymptotically [2, 11].

The receiver performs iterative message-passing interference cancellation and data estimation on the received graph chain, followed by individual error-correction decoding, performed for each data stream individually, providing near-capacity operation [2, 11]. Slightly different modulation graphs resulting from partition-spreading CDMA [9], generalized modulation [13], interleaved-division multiple access (IDMA) [14], sparse CDMA [8], and, interleaved-division multiplexing (IDM) [15] result in similar coupled equations characterizing the iterative interference cancellation behaviour at the receiver.

In practice, however, achieving high multi-user loads and spectral efficiency requires

large amount of data streams and large repetition factors. This presents a challenge for the existing message-passing algorithms in terms of both memory and computational complexity. To make the modulation format based on spatial graph coupling practical we must consider an appropriate receiver. We introduce a demodulation technique based on approximate message passing (AMP) proposed by Donoho et al. [16] for compressed sensing problems.

A version of the AMP is specially derived for our multi-user situation and allows for efficient implementation, a large savings in complexity, and simplified operation in the "near-capacity" regime. The demodulation complexity per bit is constant per iteration and does not scale with the processing gain, contrary to existing algorithms [2, 11, 12]. The algorithm is first derived for the theoretic case in which users transmit their data streams at specific times, and then extended to the case where the packets arrive at random times considering a particular application of the technique.

As an application we adapt our modulation/demodulation approach to address random channel access in maritime data exchange systems. These systems have received significant research attention in recent years due to the rapidly growing number of vessels and information services required for their operation including positioning, tracking, and navigation. While the automatic identification system (AIS) has proven to be a very successful means of tracking ships, the capacity of AIS will soon be exceeded; in some ocean areas, such as the Gulf of Mexico, the current AIS load is already above 50% of its capacity [17]. At the same time new applications such as e-navigation require additional capabilities to exchange data.

The new standard of VHF data exchange (VDES) which is currently under development by the International Telecommunications Union (ITU) will include satellite ship tracking, ship-to-shore, and ship-to-ship communication and extend the services provided by AIS [18]. Random access communications will be an important mechanism of accessing the communication channel resources in VDES. The current time-division based random access (TDMA) mechanism in AIS may not be suitable for the upcoming traffic demands for VDES. This suggests that a new random access mechanism must replace TDMA.

We focus on an uplink communication scenario in which multiple number of vessels transmit their messages to a satellite charing the same communication channel. We propose a way to apply coupling data transmission in a random access fashion, using the AMP

algorithm at the receiver. To do so we adapted our modulation scheme to include the transmission of a preamble prior to sending each datastream's payload. Similarly for demodulation at the receiver, preamble detection was inserted into the iterative loop for the AMP algorithm, its convergence is studied and we demonstrate that the achievable multi-user loads are superior to that of competing random-access systems.

The thesis is divided into five subsequent chapters and a conclusion. In Chapter 2 VHF maritime data exchange systems are introduced. Current TDMA media access schemes and the limitation of these for the future application in maritime data exchange are discussed.

In Chapter 3 the theory behind the spatial graph coupling method is introduced using low-density parity-check convolutional codes - the original coupled system. After that two examples describing applications of spatial graph coupling to multi-user communications are given. The first example discusses the coupling information transmission [2] for the multiple access AWGN channel with single antenna at the receiver, while the second example is an original contribution for the multiple input and multiple output (MIMO) case where the receiver has two antennas. The chapter discusses the original message-passing interference cancellation receiver for the coupling information transmission.

Chapter 4 discusses sparse estimation problems and the variety of approaches used for reconstruction of information. The basic theory behind message-passing algorithms on graphs is introduced and a particular example, the min-sum algorithm is discussed. The chapter concludes with a summary of the cost function and messages used for the message-passing algorithm used in the receiver for the coupling information transmission.

Modulation for communication over the AWGN channel where arrival of each datastream is known, using spatial graph coupling is elaborated on in Chapter 5. Specifically, the approximate message passing algorithm is presented and used to demodulate the received signals. Numerical results are provided to illustrate the efficiency of the AMP based demodulation. The chapter concludes with a mathematical derivation of the AMP algorithm from the message-passing rules introduced in Chapter 3 and using compressed sensing approach from Chapter 4.

In Chapter 6, we consider the modulation for communication over the AWGN channel where users access the channel randomly. This chapter is dedicated to the application to the maritime data exchange systems and in particular to the uplink to satellite scenario. The main feature of the development is the iterative reception in which detection of packets

and the multi-user demodulation is happening in parallel. Simulation results highlighting various system loads are presented.

Finally, Chapter 7 presents the conclusion and an outlook on the future work.

1.1 Summary of Contributions

The work presented in this thesis contributes to improving the efficiency of communication over the AWGN multiple access channel by exploiting spatial graph coupling and techniques from compressed sensing. As a result of this effort, two papers have been prepared for publication:

- D. Truhachev, and D. McNutt (2015)
  Coupling Information Transmission with Approximate Message Passing, IEEE Communications Letters, under revision.

- D. Truhachev, and D. McNutt (2015).
  Joint Packet Detection and Decoding for Maritime Data Exchange Systems. MTS/IEEE Oceans 2015.

The first paper is based on the work presented in Chapter 5 which focuses on the transmission of data from multiple users over the multiuser AWGN channel. The focus is on a theoretic setup in which arrival for each signal is assumed to be known, and all users transmit with equal power. The derivation of the AMP algorithm, and numerical simulations illustrating the achievable sum-rate of the algorithm in conjunction with coupling transmission are presented.

The second paper was published in the proceedings of the international conference MTS/IEEE Oceans 2015. This paper is based on the work presented in Chapter 6 where we consider the transmission of data from multiple users over the MAC AWGN channel where the packets arrival are now random. Each transmitting user forms a packet and includes a preamble to the message payload in order to ensure detection, demodulation and decoding at the receiver. In this paper we outline how the AMP algorithm derived in Chapter 5 can be adapted to allow for the detection of packets using pseudo-random binary sequences that are known to the receiver, and present the performance of this system in terms of total throughput of the system versus signal-to-noise ratio. The impact of the carrier frequency shift due to the satellite's motion is also taken into account.

Chapter 2

VHF Data Exchange System (VDES)

2.1 The Automatic Identification System

Due to the rapidly growing number of vessels and the increasing number of information services required for ship operation there has been a significant research effort directed at improving maritime data exchange systems. One important component of these exchange system has been the automatic identification system (AIS) used by ships and by vessel traffic services to identify and locate vessels by exchanging data with other nearby ships, shore base stations, and satellites.

The automatic identification system (AIS) supplements marine radar as a secondary method of collision avoidance for water transport. The International Maritime Organization (IMO) requires AIS to be fitted aboard any ship with gross tonnage equal to 300 or more, and all passenger ships. The information provided by this system consists of a unique identification, position, course, and speed for each ship. This allows for maritime authorities and each vessel's watchstanding officers to track and monitor vessel movements.

In order to record and transmit this information AIS integrates a standardized VHF transceiver with a positioning system receiver, electronic navigation sensors and a gyrocompass or rate of turn indicator. Any vessel fitted with AIS transceivers can be tracked by AIS base stations, or when out of range of terrestrial networks through satellites that have been equipped with special AIS receivers which can manage the large number of vessel signatures. Outside of collision avoidance, many applications have been developed for AIS including fishing fleet monitoring and control, vessel traffic services, marine security, aids to navigation, search and rescue, accident investigation, and fleet and cargo tracking.

Terrestrial AIS uses two globally allocated Marine Band Channels 87 and 88 with transmitting frequencies 161.975 MHz and 162.025 MHz respectively, the system transmits using Gassian minimum shift-keying (GMSK). AIS uses the frequency channels by employing frequency division duplexing, where the upper leg of these channels are used for shore-to-ship and ship-to-ship digital messaging, while the lower leg of the channels is used

for ship-to-shore messaging.

In contrast, ship-to-satellite messages (long range AIS) uses the Maritime Band channels 75 and 76 with frequencies 156.775 MHz and 156.825 MHz. Currently the ships only transmit to satellites, with no messages received from the satellites in order to provide navigational information for all ships. For this reason, there is no need for duplex channels when long range AIS is used [19]

While ship-to-ship and ship-to-shore have a range of ten to twenty miles, Satellite-AIS (S-AIS) has a much larger coverage. Shipboard AIS transceivers can reach much further vertically; up to the 400 km orbit of the International Space Station (ISS). Since satellite receive footprint is hundreds of miles in radius, such receiver needs to be able to handle a huge number of vessel signatures.

By its very nature, AIS must allow vessels to access the channel without interfering with each other, the AIS transceivers use Time Division Multiple Access (TDMA) to transmits and receive messages. The AIS standard comprises several sub-standards (or types) for individual products, but there are two major product types for vessel-mounted transceiver:

- **Type A :** Targeted at large commercial vessels, these transceivers use Self-Organized TDMA (SO-TDMA) as a method of random channel access.

- **Type B :** Targeted at smaller commercial and leisure vessels, there are two subtypes depending on whether SO-TDMA or Carrier Sense TDMA (CS-TDMA) is used as a method of random channel access by the transceiver.

2.2 Time Division Multiple Access

Time division Multiple access allows multiple users to share the same frequency channel by dividing each minute frame into 2250 slots, during which users transmits one after another. How this slot structure is accessed is determined by the chosen scheme.

- **Self Organized-TDMA :**

  This is the multiple access approach associated with the Class A transceivers. Each user maintains an updated slot map in its memory allowing for prior knowledge of available slots. This is achieved by sharing a common time of reference derived from

GPS time. Before a message is sent the user will pre-announce their transmission in a known free slot indicating their intention to use a particular slot for the next transmission. This allows other users to update the slot map, and avoid those slots that are known to be in use [19].

As users move in and out of communication regions, they will encounter new users with different slot allocations, causing the slot map to be recomputed. From these rules a dynamic and self organizing system over time and space can be determined.

- **Carrier Sense-TDMA :**

  This is the multiple access approach associated with the Class B transceivers. The access scheme is fully interoperable with SO-TDMA transmission, and also ensures priority is given to SO-TDMA transmissions. To do this, a user using CS-TDMA determines the slot timing by observing the timing of a user using SO-TDMA. Then, by continuously monitoring the channel background noise level, the user can use this as a reference for a received signal strength measurement at the start of each TDMA slot.

  When a transmission is required, a TDMA slot is picked at random and if the signal strength at the start of the slot is above the background level, the slot is assumed to be in use and the transmission is deferred, otherwise the slot is assumed to be empty and the slot used. There are limitations to this approach, this will only work on a slot by slot basis, unlike SO-TDMA which can use multiple concurrent slots.

In terrestrial AIS, SO-TDMA allows for autonomous management of capacity in busy areas, by implementing "slot-reuse rules" when all TDMA slots are occupied. The slots occupied by the most distant users are re-used for local transmission. This reduces the size of an AIS region and ensures position reports from closer users are not affected. This is important as these users are most relevant to navigation. This approach no longer works in satellite AIS when a satellite receives many messages from many vessels.

2.3 VDES

As AIS usage increased globally in terms of number of vessels and variety of applications related to AIS grow, the ITU Radiocommunication Assembly have recognized that

a new VHF data exchange system must be introduced to accommodate the growing number of users, and the diversification of data communication applications. This system will allow for harmonized collection, integration, exchange, presentation, and analysis of marine information onboard and ashore to enhance berth to berth navigation, and associated services for safety and security at sea.

While the proposed VHF data exchange system will incorporate other maritime safety related applications outside of the scope of AIS, it must improve AIS functionality in order to allow for increased usage of the channel. Running continuously, this implementation must ensure that the functions of digital selective calling, AIS and voice distress, safety, and calling communication are not impaired.

VDES will require multiple frequency channels in the VHF band for AIS, application specific messages (ASM), voice and other data communication needs such as e-navigation [17].

Current time-division based random access (TDMA) mechanism in AIS has a number of limitations such as the necessity to maintain slot maps and limitations imposed by the number of available slots. Specifically in satellite uplink communication with a large population of users accessing the channel that coordinating channel access will be difficult due to the finite number of slots per minute per frame, and the difficulty to organize slot maps due to large feedback delays. Hence, alternative, more flexible media access approaches would be desirable for VDES. In particular random media access and multi-packet reception would provide a very useful functionality in this scenario. Chapter 6 is this thesis describes a potential application of the coupling information transmission technique with an AMP receiver for VDES communication scenarios.

Chapter 3

Spatial Graph Coupling

3.1 Introduction

Error-correction codes on sparse graphs such as low-density parity-check codes (LDPC)s that have been a popular theoretic area of research codes are frequently used in practical communications by now. LDPCs are included in a numerous sttandards such as wireless local-area networks WiFi [20], WiMax, digital video broadcasting DVB-S2, 10GBase-T Ethernet and others. Due to the ubiquity of iterative belief-progagation-based decoding of sparse graphical codes, and their incorporation into many communications standards, it would be beneficial to construct new codes based on these algorithms that are formally capacity achieving.

While it is known that polar codes achieve channel capacity [21], these are not easily implemented for applications. With this in mind, it has been proven that a collection of sparse codes that are spatially coupled will have an improved decoding threshold which approaches the maximum-aposteriori decoding (MAP) of the uncoupled sparse code [6, 22]. This is known as *threshold saturation*. Originally the concept of spatial graph coupling arose from the construction of convolutional low-density parity-check codes (LDPCCs) by Felström and Zigangirov [4], spatial graph coupling has been readily adapted to other applications like source coding and multiple access communications.

Whatever the application of spatial graph coupling may be, the method involved is essentially equivalent to the case of coupling graph-based error control codes. For this reason we will introduce the concepts of spatial graph coupling in the guise of an exposition of LDPCCs on the Binary Erasure Channel (BEC), before generalizing into the framework of coupled vector recursions. As a way to sidestep the issues involved in proving threshold saturation for the spatially coupled codes, we conclude with an introduction to Lyapunov stability theory; which will allow one to prove that a coupled system may attain the capacity of the transmission channel.

3.2 LDPC Convolutional Codes Also Known as Spatially Coupled LDPC Codes

The low-density parity-check codes are linear block codes of large size, which are well-liked as their sparse graphical structure allows them to be efficiently decoded using message pasing algorithms. A convolutional LDPC code can be constructed by taking the parity-check matrix of an LDPC code and cutting diagonally as shown in Figure 3.1, which produces a lower-triangular matrix $\mathbf{H}_0$ and an upper-triangular matrix $\mathbf{H}_1$ of size $(N-K)\times N$:

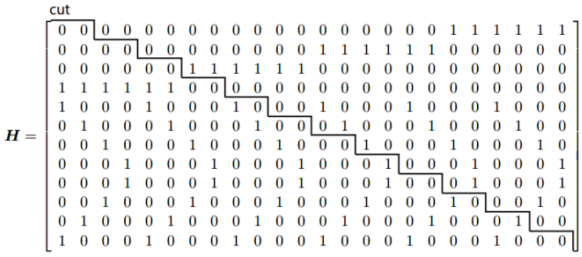

Figure 3.1: Unwrapping of Gallager-type (3,6) LDPC matrix $\mathbf{H}$. Taken from [1].

By pasting these matrices together, with $\mathbf{H}_0$ on top of $\mathbf{H}_1$ and repeating this diagonally as in Figure 3.2 ad infintum, the resulting parity-check matrix $\mathbf{H}_{conv}$ will have infinite extent. Since the original LDPC matrix was sparse, the matrix $\mathbf{H}_{conv}$ will be sparse as well, and the same message-passing algorithm may be used to decode this new code.

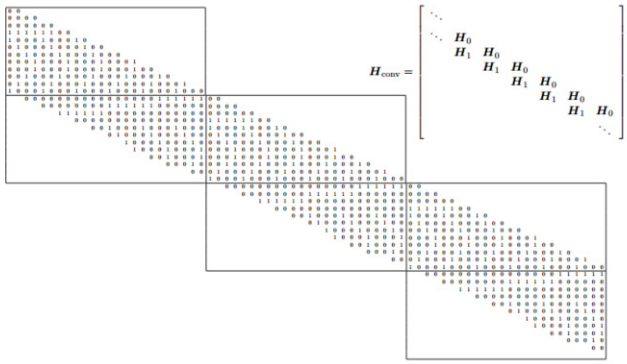

Figure 3.2: Construction of the parity-check matrix $\mathbf{H}_{conv}$. Taken from [1].

This new code may be treated as a convolutional code, with constraint length equal to the length of the original LDPC code, i.e., $\mu = N$. Realistically, a code cannot be transmitted for infintely long duration, and so one must typically have a starting point and (possibly) a termination point, which may occur after a large number, L, of sections. Thus

$\mathbf{H}_{conv}$ will be a band-diagonal matrix of size $L(N-k) \times LN$. The effect of terminating at either or both ends of the code is ultimately beneficial, as the ratio of information bits to coded bits is locally smaller around the termination points, and so a locally improved bit error rate is expected. It turns out that this improvement is carried out to the entire codeword via iterative decoding.

Just as in the original case of an LDPC block code, we may construct a Tanner graph of a convolutional LDPC code, which illustrates the band-diagonal structure of the parity-check matrix. Drawing the graph in sections as in Figure 3.3, it is easily seen that the first bits are checked by lower weighed check nodes than in the remaining sections. This may be treated as having the earlier bits before the starting point known in advance, and hence set to zero.

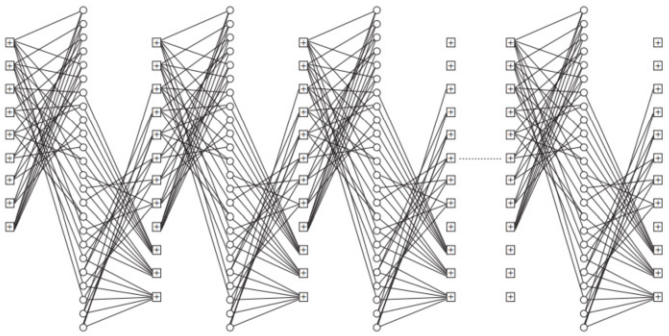

Figure 3.3: Tanner graph of the LDPCC code from Figure 3.2. Taken from [1].

Unlike the example given in Figure 3.3, the originating LDPC code must be much larger, and in practice $N \approx 10^3$ and larger are typical for practical applications. The Tanner graph for such a code would be too confusing to be used as a conceptual tool, instead we consider a proto-graph representation of the code, given in Figure 3.4, where each edge of the proto-graph represents $\frac{N}{2}$ edges, connecting $\frac{N}{2}$ variable nodes and check nodes respectively. Each of the $\frac{N}{2}$ pairs of nodes are connected by a permuted arrangement of the edges, with each of the proto-graph edges representing a different permutation.

Each section in Figure 3.4 represents a N-sized section of the convolutional LDPC code. From this perspective one can see that the endpoints of the structure have better protections; the parity nodes at either end have degree two, followed by nodes with degree four, and the remaining interior nodes have degree six, and all variable nodes have degree three While there is some rate loss associated with termination of the code, one may see

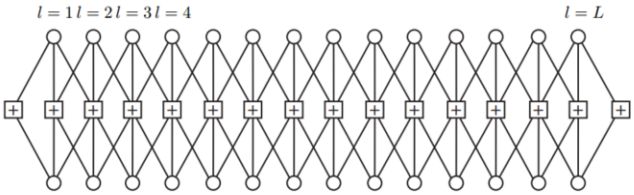

Figure 3.4: Proto-graph representation of the $(3, 6)$ LDPCC given in Figure 3.2. Taken from [1].

that as $L \to \infty$ the design rate of the LDPCC code is

$$R = \lim_{L\to\infty} 1 - \frac{(L+1)(N-K)}{LN} = \frac{K}{N}, \tag{3.1}$$

just like for the block LDPC, $R = \frac{1}{L}$ for the (3,6) code case.

The decoding threshold for convolutional LDPCs generated from regular $(J, 2J)$ LDPC codes was studied in [5, 23], and is summarized in Figure 3.5 for several lengths, L, of the structure. Notice that as $L \to \infty$, those regular codes with large J achieve a threshold that reaches the capacity limit of the binary-input AWGN channel at $E_b/N_0 = 0.19dB$ with rate $R = 0.5$, this is defined as

Definition 3.2.1. The iterative decoding threshold of a code family is the largest channel parameter ϵ^* (for BEC) or smallest SNR^* (for AWGN) such that for any $\epsilon < \epsilon^*$ ($SNR > SNR^*$) the decoding error probability of iterative message passing decoding goes to 0 as a function of the code length N with constraint length μ and the number of decoding iterations I. Typically exponential error probability decay is a function of $N(\mu)$ is required as well.

Alternatively for small L the rate loss experienced by the LDPCC code given by the formula (3.1) will be smaller than the rate of the block version of the same code, and hence codes with lengths significantly larger than ten are required to achieve the same threshold. Admittedly, for most applications the effort involved for the gain in threshold may not be necessary, as the code lengths in the block code case required to gain a fraction of a dB in terms of performance will be on the order of 100,000 compared to 1,000-10,000 of the base code. The interest in convolutional LDPCs lie in the connection to spatial graph coupling which allows for the construction of message-passing code ensembles that are proven to achieve the Shannon's capacity limit.

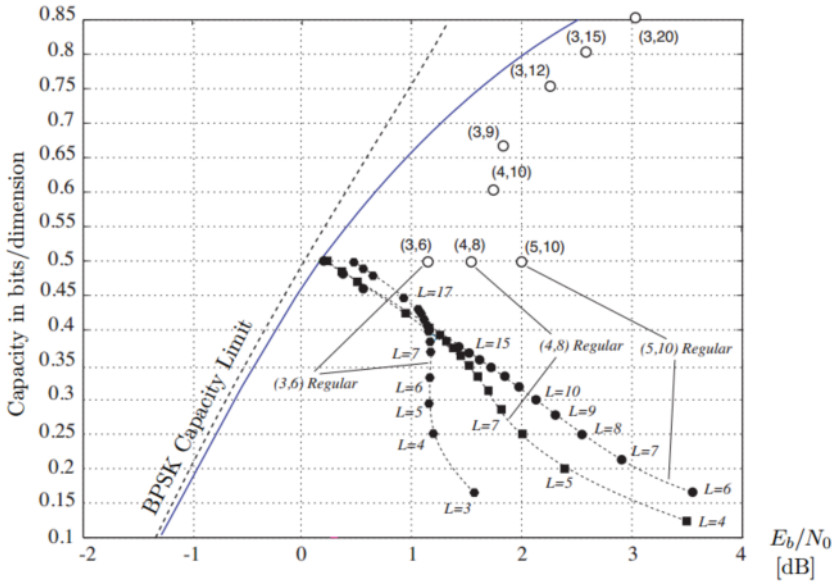

Figure 3.5: Decoding thresholds for LDPCC codes for varying lengths, L, and degree profiles under message-passing decoding. Hollow circles denote block codes, and black circles denote convolutional or coupled codes. Taken from [1].

3.3 Spatial Graph Coupling: From Graphs to Recursion Equations

As was noted in the introduction, the method behind the creation of the LDPCC's is only a particular incarnation of a much broader concept known as spatial graph coupling. Using the LDPCC codes as an example we will examine the density evolution equations and show how this may be generalized.

Figure 3.4 this can be visualized as a coupled ensemble of codes, where each edge denotes M edges of the Tanning graph of the LDPC codes. When the codes are coupled, a fixed number of the connections are attached to the neighboring codes. The coupling window W will be the number of neighbors involved on both sides of this bi-partite graph. While coupling may involve an arbitrary number of edges, to study this analytically we will require uniform edge spreading so that each line in the coupled graph represents $1/W$ of the edges of the original code. The edges are permuted randomly with the constraint that the original node degrees of the uncoupled codes are unchanged. The check nodes at start and end of the code imply that $1/W$ of the variable nodes in the first and last code are known and hence equal to zero.

To illustrate how a LDPC code would be decoded, we consider a simple channel,

namely the Binary Erasure Channel BEC given in Figure 3.6, where each bit transmitted has a probability $\epsilon \in (0, 1)$ of being erased.

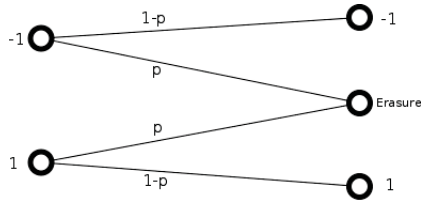

Figure 3.6: A diagram representing the Binary Erasure Channel

Consider a (d_v, d_c) LDPC block code, i.e., d_v edges come out of each variable node and d_c edges from a check node. We denote $\mathcal{A} = \{-1, 0, +1\}$ as the channel output, $r_i \in \mathcal{A}$ the received symbol at variable node i, and $d_i \in \mathcal{A}$ the decision at variable node i. A message from variable node i to check node j is represented by $u_{i\to j} \in \mathcal{A}$ and a message from check node j to variable node i is $\beta_{j\to i} \in \mathcal{A}$. Finally we write $V_{j\setminus i}$ as the set of variable nodes which connect to check node j, except for variable node i, and $C_{i\setminus j}$ to be the set of check nodes which connect to variable node i, excluding check node j. To decode an LDPC code on the BEC, we have the following steps:

1. Initialize $d_i = r_i$ for each variable node. If $r_i = 0$ then the received symbol i has been erased and variable i is said to be unknown.

2. Variable nodes send $\mu_{i\to j} = d_i$ to each check node $j \in C_i$

3. Check nodes connected to variable node i send $\beta_{j\to i} = \Pi_{l\in V_{j\setminus i}} u_{l\to j}$ to i. I.e., if all incoming messages are different from zero, the check node sends back to i the value that makes the check consistent, otherwise it sends back a zero for "unknown".

4. If the variable i is unknown and at least one $\beta_{j\to i} \neq 0$ set $d_i = \beta_{j\to i}$ and declare the variable i to be known. For the simple BEC, once a variable is known its status can not be changed.

5. When all variables are known, stop. Otherwise go back to Step 2.

To study the behaviour of this decoding scheme, density evolution analysis allows a recursive relation to be derived describing the statistical behaviour of a regular block LDPC code over the binary erasure channel (BEC) ([1], section 6.2) we consider the tree structure

given in Figure 3.7. Let $\epsilon_v^{(l)}$ denote the probability of erasures at the variable nodes at iteration l, and $\epsilon_u^{(l)}$ be the probability of erasures at the check nodes at iteration l, then for a regular code, in which the node degrees are all equal, the following recursive relation can be established

$$\epsilon_u^{(l-1)} = 1 - [1 - \epsilon_v^{(l-1)}]^{d_c - 1}, \;\; \epsilon_v^{(l)} = \epsilon[\epsilon_u^{(l-1)}]^{d_v - 1} \tag{3.2}$$

starting with $\epsilon_u^{(0)} = \epsilon$ with ϵ is the initial probability of erasure.

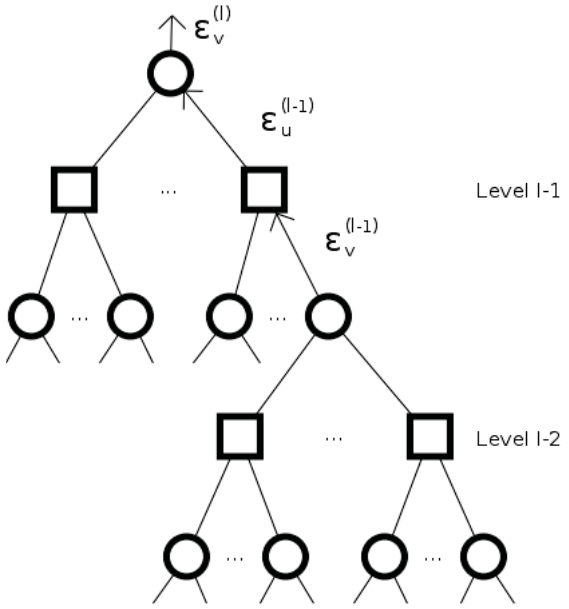

Figure 3.7: Local subtree structure of a large LDPC code

Combining these two equations produces a one-dimensional iterated map for the probability that any variable node remains uncorrected ,

$$x^{(l)} = f(x^{(l-1)}, \epsilon) = \epsilon(1 - [1 - x^{(l-1)}]^{d_c - 1})^{d_v - 1}, \tag{3.3}$$

where $x^{(l)}$ is the probability of bit erasure at iteration l in a (d_v, d_c) regular LDPC block code of infinitely large size $N \to \infty$. We need this to ensure that there is a tree-like neighborhood of decoding for each symbol. We define a check node function in (3.3) as

$$g(x) = 1 - (1 - x)^{d_c - 1}, \tag{3.4}$$

and the statistical variable node function as

$$f(y, \epsilon) = \epsilon y^{d_v - 1}, \tag{3.5}$$

the dynamic behaviour of a regular LDPC code may be characterized by the fixed points of the equation

$$x = f(g(x), \epsilon). \tag{3.6}$$

The above equation has one fixed point at $x = 0$, but can have two or three depending on ϵ, and the supremum of ϵ such that only one fixed point exists, we say ϵ^* is called the belief propagation threshold (BP threshold). For this system, convergence to the limit point $x^{(\infty)} = 0$ requires

$$x - \epsilon f(g(x), \epsilon) > 0, \;\; x \in [0, \epsilon]. \tag{3.7}$$

Alternatively, we may formulate the threshold value of the parameter ϵ as

$$\epsilon_s = \sup\{\epsilon \;\; s.t. \;\; x^{(\infty)} = 0\}. \tag{3.8}$$

Through numerical experimentation, it may be shown that the largest erasure probability that can be corrected by the code is $\epsilon = 0.4294.$ for the (3,6) code [1].

In the coupled case, we must consider the convergence behaviour of each system involved in the coupled ensemble, thus we must consider a vector convergence equation of the form

$$\boldsymbol{x} = \boldsymbol{F}(\boldsymbol{x}, \epsilon), \tag{3.9}$$

where $\boldsymbol{x} = [x_1, ..., x_L]$ is a vector of the statistical variable for each of the $L + 1$ coupled dynamical systems illustrated in figure 3.4, and $\boldsymbol{F}$ is a monotonic decreasing function. Returning to the coupled system of L copies of L graphs, with a uniform window of size

W, the variables are updated as

$$x_i^{(l+1)} = \frac{1}{W} \sum_{k=0}^{W-1} f\left(\frac{1}{W} \sum_{j=0}^{W-1} g\left(x_{i+j-k}^{(l)}, \epsilon \right) \right). \tag{3.10}$$

The boundary conditions may be imposed by requiring that $x_l = 0$ for $l \notin [0, L]$, which corresponds to variables that are already known, which may be pilot signals, frame markers, preambles or any other fixed signals we wish to couple to our code ensemble.

Writing this in vector form, we denote the vector functions $\boldsymbol{f}(\boldsymbol{x}, \epsilon)$ and $\boldsymbol{g}(x)$ which operate on each components of the vector $\boldsymbol{x}$ as a scalar with $f(\cdot)$ and $g(\cdot)$ respectively. Then equation (3.10) may be expressed as

$$\boldsymbol{x}^{(l+1)} = \boldsymbol{A}^T \boldsymbol{f}\left(\boldsymbol{A}\boldsymbol{g}\left(\boldsymbol{x}^{(l)}, \epsilon \right) \right) = \boldsymbol{F}(\boldsymbol{x}^{(l)}, \epsilon), \tag{3.11}$$

where $\boldsymbol{A}$ is a band-diagonal matrix with non-zero entries $1/W$ for all $j \geq i$ and $j - 1 \leq W$, and zeros otherwise.

The parameter ϵ in this instance represents the channel erasure probability. More generally it could be a signal to noise ratio, or another parameter characterizing the channel. Typically one is interested in the threshold of the coupled system which is the largest ϵ such that an arbitrary initial vector $\boldsymbol{x}^{(0)}$ converges to the zero vector.

Denoting the set of all possible values of the erasure probabilities for each system $\boldsymbol{x}$ as $\mathcal{X}^0$, these will be strictly non-negative and hence the form of (3.11) implies

$$\boldsymbol{F}(\boldsymbol{x}, \epsilon) \leq \boldsymbol{x} \text{ and } \boldsymbol{F}(\boldsymbol{x}, \epsilon) \in \mathcal{X}^0. \tag{3.12}$$

Thus for any $l > 0$ and $\boldsymbol{x}^{(l)} \in \mathcal{X}^0$ we have that $\boldsymbol{x}^{(l+1)} \leq \boldsymbol{x}^{(l)}$ and $\boldsymbol{x}^{(l+1)} \in \mathcal{X}^0$, and so we will have a monotonically non-increasing sequence of points if we continue to iterate the equation. If we can find some initial point $\boldsymbol{x}^{(0)}$ such that the resulting sequence of points is strictly monotonically decreasing, i.e., $\boldsymbol{x}^{(0)} > \boldsymbol{x}^{(1)} > \boldsymbol{x}^{(2)} > ...$ then this must converge to a limiting point $\boldsymbol{x}^{(\infty)} \geq 0$. Ideally this point should be a *stable fixed point*, it is possible for particular values of ϵ that there are *unstable fixed points* where the system passes through this fixed point and moves towards another fixed point.

The boundary conditions play an important role in the convergence of the coupled system, as these allow the dynamical variables x_l closer to the exterior of the ensemble to

converge faster than the interior variables. This allows for the boundary conditions to propagate to the interior of the system over several iterations, which would not be possible without the boundary conditions.

To study the convergence of the system (3.11), an approach was introduced by [24] and expanded upon in [25, 26] which was based upon the function $U(\boldsymbol{x}, \epsilon) : \boldsymbol{x} \to \mathbb{R}^1$ called the *potential function*

$$U(\boldsymbol{x}, \epsilon) = \int_0^{\boldsymbol{x}} \boldsymbol{g}'(\boldsymbol{z})[\boldsymbol{z} - \boldsymbol{f}(\boldsymbol{g}(\boldsymbol{z}), \epsilon)]d\boldsymbol{z}. \tag{3.13}$$

whose origin arose from the continuous-time approximation for the system (3.11) given by

$$\frac{d\boldsymbol{x}(t)}{dt} = \boldsymbol{F}(\boldsymbol{x}(t), \epsilon) - \boldsymbol{x}(t), \;\; t > 0, t \to \infty. \tag{3.14}$$

With this function we have a general result for any recursion equation of the above form:

Theorem 3.3.1. *For large window sizes $w \to \infty$, an iterative system of sparse graphs couples as in* (3.11) *converges to $\boldsymbol{x}^{(l)} \to \mathbf{0}$, if $U(x) > 0$. This implies that the coupled system threshold may be given by the following criteria:*

$$\epsilon_c^* = \textit{sup}\{\epsilon : U(x, \epsilon) > 0, \; \forall x \leq \epsilon\}. \tag{3.15}$$

Using this theorem, it may be shown that the coupled (3,6) LDPC code ensemble for the BEC channel will converge for erasure rates up to $\epsilon = 0.48815$ which is much closer to the capacity $C_{BEC} = 0.5$ than the uncoupled case.

In general, for large $w \geq 0$, $\epsilon_c^* \geq \epsilon^*$; coupling always improves performance. Exploiting theorem 3.3.1, it can be shown that certain convolutional LDPC codes based off the (d_v, d_c) LDPC codes achieve capacity over the binary erasure channel [1]:

Theorem 3.3.2. *For large window sizes, $w \to \infty$, an iterative system of sparse coupled regular (d_v, d_c) LDPC codes achieve the capacity of the BEC,*

$$\epsilon_c^* \to \alpha, \;\; for \;\; \alpha = \frac{d_v}{d_c}, \textit{ and } d_v \to \infty.$$

Equivalently, the rate of the code $R = 1 - d_v/d_c$ approaches the channel capacity, $R \to 1 - \epsilon = C_{BEC}$ for $\epsilon \to \epsilon_c^*$.

The proof follows by computing the potential function for the regular (d_v, d_c) LDPC codes:

$$U(x,\epsilon) = \frac{1}{d_c} - \frac{(1-x)^{d_c}}{d_c} - x(1-x)^{d_c-1} - \frac{\epsilon}{d_v}\left(1-(1-x)^{d_c-1}\right)^{d_v}. \tag{3.16}$$

and determining the largest ϵ^* for which $U(x,\epsilon^*) \geq 0$ for all $x \in [0,1]$.

The phenomenon of threshold saturation is in fact very general [6], and has been shown to be applicable to other communication scenarios such as the isotropic multiple access channel used with uniformly random signal sets [2]. For this reason it is important to show rigorously, whatever the application, that the coupled system converges to the zero vector. In the case of the LDPCC codes one is able to work with the potential function which exists as the original recursion equation is scalar-valued. In the case of vector recursions, the existence of a potential function is no longer assured. In the next section we will explore a related tool that will help determine convergence.

3.4 Convergence Analysis of Spatially Coupled Systems

From the previous section we have seen that the performance of spatially coupled ensembles of LDPC codes relied on the convergence, to the point $\boldsymbol{x}^{(\infty)} = \mathbf{0}$, of an iterative vector recursion equation:

$$\boldsymbol{x}^{(l+1)} = \boldsymbol{F}(\boldsymbol{x}^{(l)}, \epsilon), \quad \boldsymbol{F}(\boldsymbol{x}, \epsilon) \leq \boldsymbol{x} \text{ and } \boldsymbol{F}(\boldsymbol{x}, \epsilon) \in \mathcal{X}^0. \tag{3.17}$$

We will focus on the case where $\boldsymbol{x}$ is non-increasing, however it should be noted that there may be other stable points such that $\boldsymbol{x}^{(\infty)} \neq \mathbf{0}$ which might cause a small initial value of $\boldsymbol{x}^{(0)}$ to move towards a non-zero fixed point; given the current application these points should be avoided as they are not helpful for applications.

If we pick an initial point $\boldsymbol{x}^{(0)}$ such that the resulting sequence of points is strictly monotonically decreasing, i.e., $\boldsymbol{x}^{(0)} > \boldsymbol{x}^{(1)} > \boldsymbol{x}^{(2)} > ...$, then the sequence will converge to some limiting point $\boldsymbol{x}^{(\infty)}(\boldsymbol{x}^{(0)}, \epsilon) \geq \mathbf{0}$. We wish to determine when a given vector recursion will converge to the point $\boldsymbol{x}^{(\infty)}(\boldsymbol{x}^{(0)}, \epsilon) = \mathbf{0}$ for a particular ϵ. This question may be approached using the Lyapunov method. This will simplify the behaviour of individual trajectories in a dynamical system, like the one given above, to the study of the system's

properties in subspaces of $\mathcal{X}^0$ through the study of Lyapunov functions [27].

As before, $\mathcal{X}^0$ denotes the smallest set containing $\boldsymbol{x}^{(l)}$ for all $l \geq 0$ and we introduce an open subset $\mathcal{X}^1 \subset \mathcal{X}^0$ such that $\mathbf{0} \notin \mathcal{X}^1$. For the domain of ϵ, $\mathcal{E}$, we have two subsets $\mathcal{E}_0 = \mathcal{E} \setminus \{0\}$ and $\mathcal{E}_2 \subset \mathcal{E}_0$. The purpose of $\mathcal{E}_2$ will be explained momentarily, but this subset willl be related to the issue of global asymptotic stability.

Definition 3.4.1. The solution $\boldsymbol{x}^{(l)} = \mathbf{0}$ to (3.11) is globally asymptotically stable if $\lim_{l\to\infty} \boldsymbol{x}^{(l)} = \mathbf{0}$ for all $\boldsymbol{x}^{(0)} \in \mathcal{X}^0$.

Lyapunov introduced a special type of function in $\mathcal{X}^0$ defined by the following criteria:

Definition 3.4.2. A continuous function $V(\boldsymbol{x}) : \mathcal{X} \to \mathbb{R}^1$ is called a Lyapunov function for the system (3.11) with $\boldsymbol{x} \in \mathcal{X}^0$, $\epsilon \in \mathcal{E}_2$ if it satisfies the following conditions:

$$V(\mathbf{0}) = 0, \tag{3.18}$$

$$V(\boldsymbol{x}) > 0 \; for \; \boldsymbol{x} \in \mathcal{X}^1, \tag{3.19}$$

$$V(\boldsymbol{f}(\boldsymbol{g}(\boldsymbol{x}), \epsilon)) - V(\boldsymbol{x}) < 0 \; for \; \boldsymbol{x} \in \mathcal{X}^1, \; \epsilon \in \mathcal{E}_2. \tag{3.20}$$

We may provide sufficient conditions for global asymptotic stability of the system (3.11)

Theorem 3.4.3. *Given that $V(\boldsymbol{x})$ is a Lyapunov function for the system* (3.11) *for $\epsilon \in \mathcal{E}_2$, the solution $\boldsymbol{x}^{(l)} = \mathbf{0}$ is globablly asymptotically stable.*

Thus for any $\epsilon \in \mathcal{E}_2$, any initial choice of $\boldsymbol{x}^{(0)}$ will converge to the origin $\mathbf{0}$.

While this was formulated for a continuous system, this may be related to the iterative equation (3.11) by writing this as

$$\boldsymbol{x}^{(l)} - \boldsymbol{x}^{(l+1)} = \boldsymbol{q}(\boldsymbol{x}^{(l)}) = \boldsymbol{x}^{(l)} - \boldsymbol{F}(\boldsymbol{x}^{(l)}.\epsilon) \tag{3.21}$$

We may build a Lyapunov function for this system by considering its continuous-time analog,

$$\frac{d\boldsymbol{x}(t)}{dt} = -\boldsymbol{q}(\boldsymbol{x}(t)), \; t > 0, t \to \infty. \tag{3.22}$$

We use the *variable gradient method* to construct a Lyapunov function for this new system that satisfies (3.18) - (3.20). Let $V : \mathcal{X} \to \mathbb{R}^1$ be a continuously differentiable function

with

$$\boldsymbol{h}(\boldsymbol{x}) = \left(\frac{\partial V}{\partial \boldsymbol{x}}\right)^T = \left[\frac{\partial V}{\partial x_1}, ..., \frac{\partial V}{\partial x_n}\right]^T.$$

The derivative of $V(\boldsymbol{x})$ along the trajectories of (3.22) are now given by

$$\frac{dV(\boldsymbol{x})}{dt} = -\frac{\partial V}{\partial \boldsymbol{x}}\boldsymbol{q}(\boldsymbol{x}) = -\boldsymbol{h}^T(\boldsymbol{x})\boldsymbol{q}(\boldsymbol{x}) \tag{3.23}$$

To satisfy (3.19), $\boldsymbol{h}$ must be the gradient of a positive function, while condition (3.20) requires

$$\frac{dV(\boldsymbol{x})}{dt} = -\boldsymbol{h}^T(\boldsymbol{x})\boldsymbol{q}(\boldsymbol{x}) < 0,\ \boldsymbol{x} \in \mathcal{X}^1. \tag{3.24}$$

If this is the case, then $V(\boldsymbol{x})$ for both system (3.21) and (3.22) will be the line integral

$$V(\boldsymbol{x}) = \int_0^{\boldsymbol{x}} \boldsymbol{h}^T(\boldsymbol{x})ds, \tag{3.25}$$

which will be path-independent if and only if the Jacobian matrix of $\boldsymbol{h}(\boldsymbol{x})$ is symmetric

$$\frac{\partial h_i}{\partial x_j} = \frac{\partial h_j}{\partial x_i}\ \ i, j \in [1, n]. \tag{3.26}$$

since $\boldsymbol{h}(\boldsymbol{x})$ is constructed to be the gradient of a potential function.

Generally the vector recursion $\boldsymbol{q}$ will not be conservative, so that the Jacobian matrix of $\boldsymbol{q}(\boldsymbol{x})$ is not symmetric, and the hunt for $\boldsymbol{h}$ is not altogether a simple task. As a simplifcation, we may consider $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{B}(\boldsymbol{x})\boldsymbol{q}(\boldsymbol{x})$, where $\boldsymbol{B}$ is a positive-definite $n \times n$ matrix, then (3.24) becomes

$$\boldsymbol{h}^T(\boldsymbol{x})\boldsymbol{q}(\boldsymbol{x}) = \boldsymbol{q}^T(\boldsymbol{x})\boldsymbol{B}^T(\boldsymbol{x})\boldsymbol{q}(\boldsymbol{x}) > 0. \tag{3.27}$$

and (3.25) is now

$$V_{\boldsymbol{B}}(\boldsymbol{x}) = \int_0^{\boldsymbol{x}} [\boldsymbol{B}(\boldsymbol{s})(\boldsymbol{s} - \boldsymbol{f}(\boldsymbol{g}(\boldsymbol{s})))]^T d\boldsymbol{s}. \tag{3.28}$$

This will be a Lyapunov function for the system (3.23) if $V_{\boldsymbol{B}}(\boldsymbol{x}) > 0$, and condition (3.24)

is satisfied for $\boldsymbol{x} \in \mathcal{X}^1$. If this is the case then $\lim_{l\to\infty} \boldsymbol{x}^{(l)} = \mathbf{0}$.

If we have found a suitable Lyapunov function, we may obtain a region of the parameter space $\epsilon \in \mathcal{E}_2$ where our coupled system converges, although this region may be smaller than the actual convergence region in $\mathcal{E}$. The region $\mathcal{E}_2$ is determined by the boundary conditions. If the coupled system has infinite extent, or is coupled together in a loop, there will be no difference between the regions for this system and the uncoupled system respectively. However, if we allow for $x_i = 0$ for $i < 0$ and/or $i > L,\ \ L < \infty$ then the region $\mathcal{E}_2$ for the coupled system will be larger than that of the uncoupled system.

We state two more definitions in order to formulate this precisely.

Definition 3.4.4. The coupled-system threshold with $x_i = 0\ i \notin [1, L]$ is defined as

$$\epsilon_c^* = \sup\{\epsilon \in \mathcal{E}_2 | \boldsymbol{x}^{(\infty)}(\mathbf{1}, \epsilon) = \mathbf{0}\}. \tag{3.29}$$

The boundary conditions ensure that $\epsilon_c^* \geq \epsilon_s^*$ from (3.8), with equality occuring when $w = 0$. In some cases it may not be possible to determine the value ϵ_s^*; for example if $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{g}'(\boldsymbol{x})\boldsymbol{q}(\boldsymbol{x})$ does not produce a symmetric Jacobian, then the original potential function in (3.13) cannot be constructed. In this case we may consider the following

Definition 3.4.5. For a positive-definite matrix $\boldsymbol{B}$ the coupled-system threshold $\epsilon_c(\boldsymbol{B})$ is defined as

$$\epsilon_c(\boldsymbol{B}) = \sup\left\{\epsilon \in \mathcal{E}_2 | \min_{\boldsymbol{x}\in\mathcal{X}^0} V_{\boldsymbol{B}}(\boldsymbol{x}) \geq 0\right\}. \tag{3.30}$$

For any positive-definite matrix $\boldsymbol{B}$ we have that $\epsilon_c(\boldsymbol{B}) < \epsilon_c^*$ which shows that the Lyapunov function provides sufficient, but not necessary conditions for convergence. However, if one is able to show that $\epsilon_c(\boldsymbol{B})$ coincides with the capacity limit of the channel, then it automatically becomes a necessary condition.

There is one more issue that must be considered when determining the largest region $\mathcal{E}_2$, and this concerns condition (3.20) when considering the largest region $\mathcal{X}_0$ for which global convergence of the dynamical system is suitable for the application it models. It is possible that some initial values lead to fixed points $\boldsymbol{x}^{(\infty)} \neq \mathbf{0}$ such that $\boldsymbol{x}^{(\infty)} - \boldsymbol{f}(\boldsymbol{g}(\boldsymbol{x}^{(\infty)}), \epsilon) = \mathbf{0}$. At these points we must show that these points are unstable, so that any perturbation from this point will cause the system to converge to $\mathbf{0}$. If we wish to include these points, we

must relax (3.20) to be a non-strict inequality. With these points in mind, it is possible to choose an appropriately coupled ensemble of codes for which $\mathbf{0}$ is globally asymptotically stable. [27]

Theorem 3.4.6. *There exists a function $w_0(\boldsymbol{f}, \boldsymbol{g})$ such that for any positive-definite matrix $\boldsymbol{B}$, $W \geq w_0(\boldsymbol{f}, \boldsymbol{g})$, $L \geq 2w + 1$ and $\epsilon < \epsilon_c^*(\boldsymbol{B})$, the only stable fixed point of the system* (3.11) *is $\boldsymbol{x}^{(\infty)} = \mathbf{0}$.*

3.5 Coupling Information Transmission Modulation

3.5.1 Multiple-Access Channel: Single Antenna Receiver

To illustrate the flexibility of spatial graph coupling, we provide an example outside of coding theory, where a signalling format has been constructed which achieves capacity on the additive white Gaussian noise (AWGN) channel [2]. Here we introduce coupling information transmission which will be the focus of the contribution in Chapter 5 and 6[1]. The signals transmitted over the AWGN channel are formed through the superposition of L independent modulated data streams. These may arise from the same or several different terminals and are superimposed at the receiver along with AWGN noise.

We note that spatial graph coupling is closer to data transmission than digital transmission, as we have assumed some form of digital-to-analog conversion prior to each datastreams transmission using an accepted form of modulation[2]. Similarly analog-to-digital conversion is assumed at the receiver and demodulation is performed before data recovery for spatially coupled graphs begins. Noting this we will call the combination of the two processes at the transmitter and receiver modulation and demodulation respectively.

For each datastream, a binary information sequence, $\mathbf{u}_l = u_{1,l}u_{2,l}u_{3,l}, ..., u_{K,l}$, enters a binary forward error correction encoder of rate R=K/N. Each bit of the encoded binary sequence, $\mathbf{v}_l = v_{1,l}v_{2,l}v_{3,l}, ..., v_{N,l}$, with $v_{j,l} \in \{-1, 1\}, j = 1, 2, ..., N$ is replicated M times and permuted producing a sequence, $\tilde{\mathbf{v}}_l = \tilde{v}_{1,l}\tilde{v}_{2,l}\tilde{v}_{3,l}, ..., \tilde{v}_{MN,l}$. The L modulated datastreams are added together after certain chosen time offsets. The first group of datastreams starts at time $t = 1$, after τ_2 intervals the second group of datastreams initiates.

[1] In the modulation format presented in Chapter 5 and 6, signature sequences are used for modulation, however, in the following modulation format this is not included

[2] For example, GMSK is used as a digital modulation approach for AIS.

For $t \in [\tau_2 + 1, \tau_2 + \tau_3]$ the signal consists of the superposition of the first two groups of datastreams. After a delay of τ_3 symbols the third group of datastreams are added, and so on. This is called *datastream coupling*, and the modulated signal at time, t, is:

$$s_t = \sum_{l=1}^{L(t)} \tilde{v}_{t,l}, \;\; t = 1, 2, ...w, \text{ with } L(t) = \min_{l>0}\{\sum_{i=2}^{l+1} \tau_i \geq t\}.$$

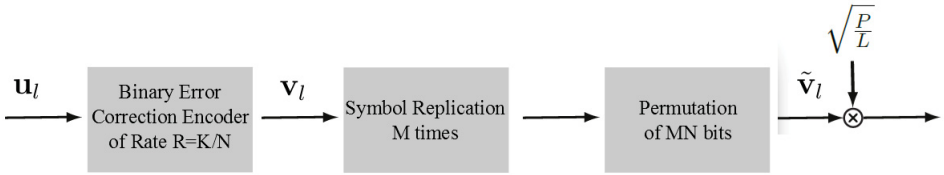

Figure 3.8: Modulation of a binary information sequence for one datastream.

Demodulation is achieved through an iterative approach. The received signal

$$\mathbf{y} = \mathbf{s} + \mathbf{n}, \;\; n \sim N(0, \sigma^2)$$

contains M replicas of each transmitted bit $v_{j,l}$, $j \geq 1$ and $l \in [1, L]$. Denoting $\mathcal{T}(j, l)$ as the set of indices t such that the signal s_t and hence y_t contains $v_{j,l}$, and $\mathcal{J}(t)$ to denote the set of all index pairs (j, l) such that $v_{j,l}$ is included in y_t, as in Figure 3.10, then for each bit $v_{j,l}$ we choose the set of received signals $\{y_t\}_{t \in \mathcal{T}(j,l)}$ to form a vector $\boldsymbol{y}_{j,l}$.

As each y_t, $t \in \mathcal{T}(j, l)$ contains $v_{j,l}$ then

$$y_{j,l} = \boldsymbol{h} v_{j,l} + \xi_{j,l} \tag{3.31}$$

where $\boldsymbol{h} = \sqrt{1/L}(1, 1, ..., 1)$, and the vector $\xi_{j,l} = (\xi_{j,l,t_1}, \xi_{j,l,t_2}, ..., \xi_{j,l,t_M})$ where $(t_1, t_2, ..., t_M) = \mathcal{T}(j, l)$ is the noise-and-interference vector with respect to the signal $v_{j,l}$. This may be broken into two pieces

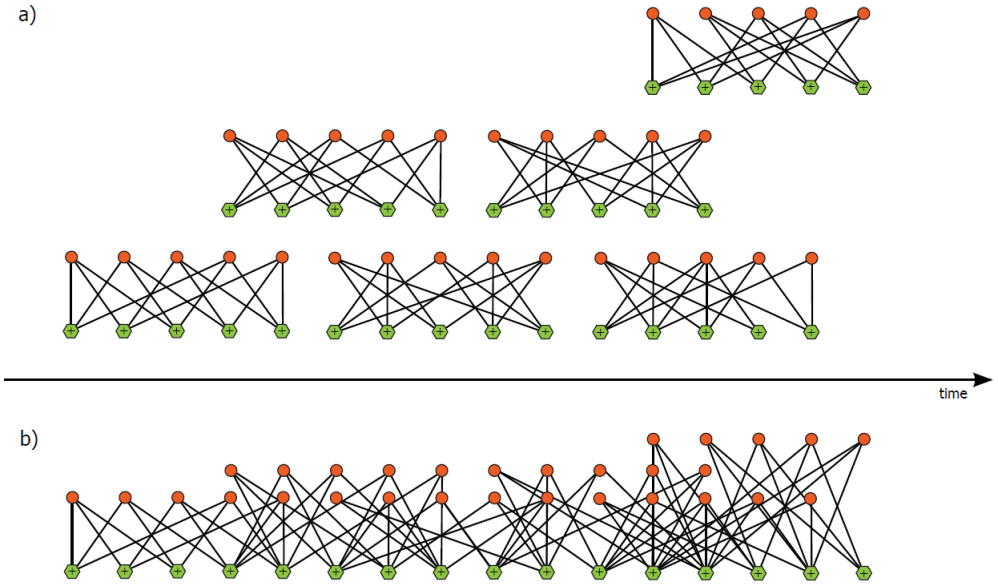

Figure 3.9: Encoded data bits $v_{j,l}$ are represented by "variable" nodes which are connected to "channel" nodes representing modulation symbols s_t. Graphs representing individual data streams are shown in a. Variable nodes are depicted as circles while channel nodes are hexagons. The example in b shows the graph obtained by stream coupling with three datastreams in which the information is modulated into five bit blocks. The transmission time offsets are $\tau_2 = 3$ and $\tau_3 = 8$. Taken from [2].

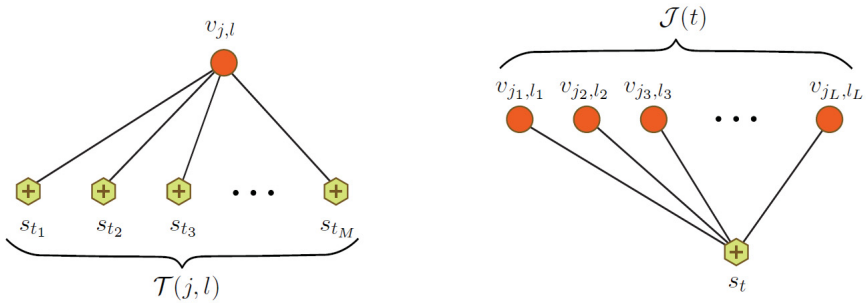

Figure 3.10: The relation between variable and channel nodes and the sets of indices involved. Taken from [2]

$$\xi_{j,l,t} = \sqrt{\frac{1}{L}} \sum_{(j',l') \in \mathcal{J}(t) \setminus (j,l)} v_{j',l'} + n_t, \quad t = t_1, t_2, ...t_M. \tag{3.32}$$

As L increases, we may invoke the Central Limit Theorem, implying that the vector $\xi_{j,l}$

converges to a Gaussian random vector with independent zero-mean components and covariance matrix $\boldsymbol{R}_{j,l} = diag(\sigma^2_{t_1}, \sigma^2_{t_2}, ...\sigma^2_{t_M})$ where σ^2_t denotes the variance of $\xi_{j,l,t}$. For $t > 2W+1$ the number of elements in the set $\mathcal{J}(t)$ is L and so the variance of $\sigma^2_t = (L-1)/L + \sigma^2$ according to (3.32) When $t \leq 2W+1$, the size of the set $\mathcal{J}(t)$ is less than L, and so the variances are influenced by the interference cancellation throughout the demodulation iterations, the expression based on σ^2_t is preferred.

With this vector $\boldsymbol{y}_{j,l}$ one may perform minimum mean-squared error (MMSE) filtering to form a SNR-optimal linear estimate of $v_{j,l}$ given by

$$z_{j,l} = \boldsymbol{w}^T_{j,l}\boldsymbol{y}_{j,l} = \boldsymbol{w}^T_{j,l}\boldsymbol{h}v_{j,l} + \boldsymbol{w}^T_{j,l}\xi_{j,l} \tag{3.33}$$

$$\boldsymbol{w}^T_{j,l} = (\boldsymbol{I} + \boldsymbol{h}^*\boldsymbol{R}^{-1}_{j,l}\boldsymbol{h})^{-1}\boldsymbol{h}^*\boldsymbol{R}^{-1}_{j,l}. \tag{3.34}$$

This minimizes $||\boldsymbol{w}_{j,l}\boldsymbol{y}_{j,l} - v_{j,l}||^2$. The resulting SNR of the signal $z_{j,l}$ is then

$$\gamma_{j,l} = \boldsymbol{h}^*\boldsymbol{R}^{-1}_{j,l}\boldsymbol{h} = \frac{1}{L}\sum_{t\in\mathcal{T}(j,l)}\frac{1}{\sigma^2_t} \tag{3.35}$$

As $v_{j,l} \in \{1, -1\}$, and takes each of the two values with equal probability $1/2$, one can form a conditional expectation estimate $\hat{v}_{j,l}$ of $v_{j,l}$ as

$$\hat{v}_{j,l} = \mathbb{E}(v_{j,l}|z_{j,l}) = \tanh(z_{j,l}\gamma_{j,l}) = \tanh\left[z_{j,l}\frac{1}{L}\sum_{t\in\mathcal{T}(j,l)}\frac{1}{\sigma^2_t}\right] \tag{3.36}$$

With all of the estimates $\hat{v}_{j,l}$ computed for all data bits $v_{j,l}$ $j = 1, 2, ...$ and $l \in [1, L]$ the next iteration starts with interference cancellation performed by computing $\boldsymbol{y}^{(1)}_{j,l}$ with components

$$y^{(1)}_{j,l,t} = y_t - \sqrt{\frac{1}{L}}\sum_{(j',l')\in\mathcal{J}(t)\setminus(j,l)}\hat{v}_{j',l'}, \;\; t \in \mathcal{T}(j,l). \tag{3.37}$$

From which we have the vector

$$\boldsymbol{y}_{j,l}^{(1)} = \boldsymbol{h} v_{j,l} + \xi_{j,l}^{(1)} \tag{3.38}$$

with components of $\xi_{j,l}^{(1)}$

$$\xi_{j,l,t}^{(1)} = \sqrt{\frac{1}{L}} \sum_{(j',l') \in \mathcal{J}(t) \setminus (j,l)} (v_{j',l'} - \hat{v}_{j',l'}) + n_t, \;\; t \in \mathcal{T}(j,l). \tag{3.39}$$

This process continues by computing new estimates $\hat{v}_{j,l}^{(1)}$ by repeating the same procedure to compute (3.34) and (3.36), and the estimation and interference cancellation steps are repeated for a number of iterations. In order to avoid reusing the same information at each iteration, one computes M extrinsic vectors $\boldsymbol{y}_{j,l,t}^{(i)}, \; t \in \mathcal{T}(j,l)$ for each bit $v_{j,l}$ at iteration i and the components of $\boldsymbol{y}_{j,l,t}^{(i)}$ are then

$$\begin{aligned} y_{j,l,t'}^{(i)} &= y_{t'} - \sqrt{\frac{1}{L}} \sum_{(j',l') \in \mathcal{J}(t) \setminus (j,l)} \hat{v}_{j',l',t'}^{(i-1)} \\ &= \sqrt{\frac{1}{L}} v_{j,l} + \sqrt{\frac{1}{L}} \sum_{(j',l') \in \mathcal{J}(t) \setminus (j,l)} (v_{j',l'} - \hat{v}_{j',l',t'}^{(t-1)}) + n_{t'} \end{aligned} \tag{3.40}$$

where $t' \neq t$ and $t' \in \mathcal{T}(j,l)$. As before the vectors $\boldsymbol{y}_{j,l,t}^{(i)}$ are used to form M values for $z_{j,l,t}^{(i)}$ according to (3.34) and corresponding estimates $\hat{v}_{j,l,t}^{(i)}$ using (3.36) for each replication of $v_{j,l}$.

After a fixed number of iterations, I, the estimated bits are passed to a decoder to decode the fowrward error correction codes used to encode the information into the datastreams.

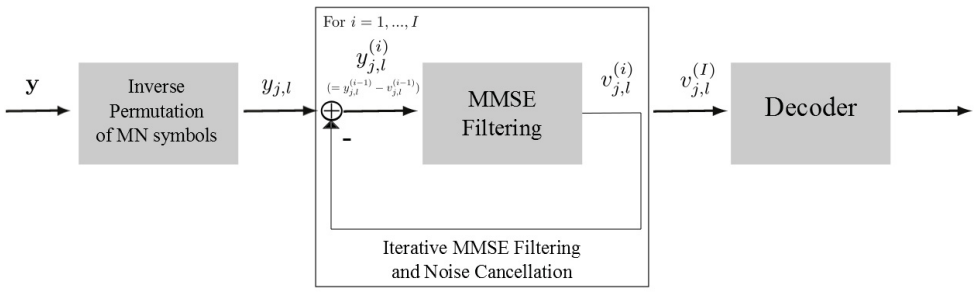

Figure 3.11: General Structure of iterative demodulation prior to decoding.

As in the case of the LDPC codes, one can produce a recursive relation describing the evolution of the noise-and-interference power. Assuming that the data block of MN symbols consists of $2W+1$ equal length subblocks, with time measured in subblocks, and that the L datastreams are split into $2W+1$ equal size groups. The analysis of the evolution of the noise-and-interference power at iteration i for symbols $v_{j,l}$ transmitted at time t is now:

$$x_i^t = \frac{1}{2W+1} \sum_{j=-W}^{W} g_m \left(\frac{1}{\alpha} \frac{1}{2W+1} \sum_{l=-W}^{W} \frac{1}{x_{i-1}^{t+j+l}} \right) + \sigma^2, \tag{3.41}$$

where the function g_m is given as

$$g_m(a) = \mathbb{E}[(1 - \tanh(a + \zeta\sqrt{a}))^2], \;\; \zeta \sim N(0,1)$$

and α is the modulation load.

Definition 3.5.1. The **modulation load** is the ratio of the number of users divided by the number of replicas made for each message

$$\alpha = \frac{L}{M}$$

Assuming that transmission starts at $t = 1$, and that at each instant, t, the modulation load is increased by $\alpha/(2W+1)$ for $t \in [1, 2W+1]$ the initial conditions are

$$\begin{gathered} x_0^t = 0, \; t \leq 0; \\ x_0^t = \tfrac{t}{2W+1} + \sigma^2, \; t \in [1, 2W+1]; \\ x_0^t = 1 + \sigma^2, \; t > 2W+1. \end{gathered} \tag{3.42}$$

The system converges if equation (3.41) approaches some small value. Notice that (3.41) resembles a SC system, where α acts as the ϵ parameter. However, to put this into the appropriate form where $x^{(\infty)} = 0$ will be a fixed point, we must examine the fixed points of the uncoupled case. Setting $W = 0$, we produce the recursion describing the evolution of the noise-and-interference power during the demodulation iterations:

$$x_i = g_m \left(\frac{1}{\alpha x_{i-1}} \right) + \sigma^2. \tag{3.43}$$

Depending on the choice of α this equation will have one, two, or three fixed points for $x \in [0, \infty)$. In the case where (3.43) has three fixed points such that $x^{(1)} < x^{(2)} < x^{(3)}$. The roots $x^{(1)}$ and $x^{(3)}$ will be stable, while $x^{(2)}$ will be an unstable fixed point.

Denoting the capacity of the AWGN channel with SNR $1/\sigma^2$ as

$$\mathcal{C}(\sigma^2) = \frac{1}{2}\log_2\left(1 + \frac{1}{\sigma^2}\right) \tag{3.44}$$

we may provide upper and lower bounds on the stable roots where the proof is given in [2]:

Lemma 3.5.2. *For $\sigma^2 \leq 1$, the following statements are satisfied:*

1. *for $\alpha \in [0, \mathcal{C}(\sigma^2)]$, $\sigma^2 \leq x^{(1)} \leq (1 + e^{-1/\sigma})\sigma^2 \leq 2\sigma^2$,*
2. *for $\alpha \in [4, \mathcal{C}(\sigma^2)]$, $1 + \sigma^2 - \frac{2}{\alpha} \leq x^{(3)} \leq 1 + \sigma^2$,*
3. *for $\alpha \in [5, \mathcal{C}(\sigma^2)]$, $x^{(3)} = 1 + \sigma^2 - \frac{1}{\alpha(1+\sigma^2)} + \rho(\alpha, \sigma^2)$ where ρ is bounded by a function $\tilde{\rho}(\alpha)$ such that $|\rho(\alpha, \sigma^2)| \leq \tilde{\rho}(\alpha) = o\left(\frac{1}{\alpha}\right)$ for $\sigma^2 \in [0, 1]$,*
4. *for $\sigma \leq 0.01$ and $\alpha \geq 2$, $0.3 \leq x^{(3)} \leq 1 + \sigma^2$*

With approximate neighbourhoods of each fixed point known, we introduce the following functions:

$$f(x;\alpha) = g_m\left(\tfrac{1}{\alpha}\left(\tfrac{1}{x^{(1)}} - x\right)\right) + \sigma^2 - x^{(1)},$$
$$g(x) = \tfrac{1}{x^{(1)}} - \tfrac{1}{x^{(1)}+x},$$

the recursion (3.41) may now be written as

$$x_i^t = \frac{1}{2W+1}\sum_{j=-W}^{W} f\left(\frac{1}{2W+1}\sum_{l=-W}^{W} g\left(x_{i-1}^{t+j+l}\right);\alpha\right).$$

The potential function for the uncoupled case (W=0) is then

$$U(x;\alpha) = ln\left(\frac{x + x^{(1)}}{x^{(1)}}\right) - \frac{\sigma^2 x}{x^{(1)}(x + x^{(}(1))} - \alpha\int_{\frac{1}{\alpha(x+x^{(1)})}}^{\frac{1}{\alpha x^{(1)}}} g_m(y)dy.$$

The convergence of this spatially coupled system has important implications for the achievable rate and capacity of the signaling format [2]. If the system (3.41) converges

to its smallest root $x^{(1)}$ of (3.43) after I iterations, the SNR of the demodulated data bits equals $M/(Lx^{(1)}) = 1/(\alpha x^{(1)})$. Each replica suffers noise-and-interference power $x^{(1)}$, and the replicas are added together. Thus, the total communication rate achievable by the system will be

$$\mathcal{R}(\alpha, \sigma^2) = \frac{L}{M}\mathcal{C}_{\text{BIAWGN}}\left(\frac{M}{Lx^{(1)}}\right) = \alpha\mathcal{C}_{\text{BIAWGN}}\left(\frac{1}{\alpha x^{(1)}}\right).$$

where $\mathcal{C}_{\text{BIAWGN}}(a)$ denotes the capacity of the binary-input AWGN channel with SNR, a. Letting $\bar{\alpha}(\sigma^2, W)$ denote the maximum load for which the coupled system converges to $x^{(1)}$ for a given σ^2 and W the following theorem holds [2]:

Theorem 3.5.3. *There exists a* $\bar{W} > 0$ *such that* $\forall\ W > \bar{W}$

$$\mathcal{R}(\bar{\alpha}, \sigma^2) = \mathcal{C}(\sigma^2) - \frac{\frac{3}{2ln2}}{\mathcal{C}(\sigma^2)} + o\left(\frac{1}{\mathcal{C}(\sigma^2)}\right).$$

as $\sigma^2 \to 0$ *and thus* $\lim_{\sigma^2 \to 0}(\mathcal{C}(\sigma^2) - \mathcal{R}(\bar{\alpha}, \sigma^2)) = 0$

3.5.2 MIMO Block Fading Channel

Consider a multi-user MIMO system, where K data streams are transmitted by L independent users to a MIMO receiver with two receiving antennas. User $l \in \{1, 2, \cdots, L\}$ transmits a signal vector $\tilde{\boldsymbol{v}}_l$ (single data stream). First a binary information sequence $\boldsymbol{u}_l = u_{1,l}, u_{2,l}, u_{3,l}, \cdots, u_{K,l}$ enters a binary forward error correction encoder of rate $R = K/N$. Then each bit of the encoded binary sequence $\boldsymbol{v}_l = v_{1,l}, v_{2,l}, v_{3,l}, \cdots, v_{N,l}$, where $v_{j,l} \in \{1, 1\}$, $j = 1, 2, \cdots, N$ is replicated M times and permuted producing a sequence $\tilde{\boldsymbol{v}}_l = \tilde{v}_{1,l}, \tilde{v}_{2,l}, \cdots, \tilde{v}_{MN,l}$. The power of each user is P/M. The signals $\mathbf{y}_1$ and $\mathbf{y}_2$ received at the two receive antennas at time t are then

$$y_{t,1} = \sqrt{\frac{P}{M}}\sum_{i=1}^{L} h_{1,t,i}\tilde{v}_{j,i} + n_{1,t}$$
$$y_{t,2} = \sqrt{\frac{P}{M}}\sum_{i=1}^{L} h_{2,t,i}\tilde{v}_{j,i} + n_{2,t}$$

or more compactly by combining the two vectors as

$$\mathbf{y}_t = \sqrt{\frac{P}{M}}\mathbf{H}_t\boldsymbol{\nu}_t + \boldsymbol{n}_t \tag{3.45}$$

where $\mathbf{y}_t = [y_{t,1}y_{t,2}]^T$, $\mathbf{H}_t = [\mathbf{h}_1, ..., \mathbf{h}_L]$ with $\mathbf{h}_l = [h_{1,t,l}, h_{2,t,l}]^T$ denotes the channel matrix, $\boldsymbol{\nu}_t = [\tilde{v}_{t,1}, \tilde{v}_{t,2}, \cdots, \tilde{v}_{t,L}]^T$, and $\mathbf{n}_t = [n_{t,1}n_{t,2}]^T$ is the noise vector. Let us assume that the channel us fixed, i.e., $\mathbf{H}_t = \mathbf{H}$, for all t.

To discuss the case of asynchronous transmission along with the possibility of data coupling for the modulated data-streams, we define two sets. Let $\mathfrak{T}(j,l)$ denote the set of indices t such that the the signal s_t representing $y_{\cdot t}$ contains $\mathbf{y}_{j,l}$, and $\mathfrak{I}(t)$ denote the set of all index pairs for which $v_{j,l}$ is included in $\mathbf{y}_t$. Additionally we add an index to the columns of H_t, $\mathbf{h}_{t,l}$ - as these may vary with time. Now, for each bit $v_{j,l}$ we must use the set of received signals $\{\mathbf{y}_t\}$, $t \in \mathfrak{T}(j,l)$ to form a $2M$ component vector $\mathbf{y}_{j,l}$, containing the bit $v_{j,l}$:

$$\bar{\mathbf{y}}_{j,l} = \bar{\mathbf{h}}v_{j,l} + \bar{\zeta}_{j,l}$$

where $\bar{\mathbf{h}}_l = \sqrt{\frac{P}{M}}[\mathbf{h}^T_{t_1,l}, ..., \mathbf{h}^T_{t_M,l}]$ and the noise-and-interference vector with respect to the signal $v_{j,l}$ given by, $\bar{\boldsymbol{\zeta}}_{j,l} = (\boldsymbol{\zeta}^T_{j,l,t_1}, ..., \boldsymbol{\zeta}^T_{j,l,t_M})$ with $(t_1, t_2, ..., t_M) = \mathfrak{T}(j,l)$. The components of this vector are now:

$$\boldsymbol{\zeta}_{j,l,t} = \sqrt{\frac{P}{M}} \sum_{(j',l') \in \mathfrak{I}(t) \setminus \{(j,l)\}} \bar{\mathbf{h}}_{l'} v_{j',l'} + n_t, \quad t = t_1, t_2, ..., t_M.$$

As L increases, the Central Limit Theorem implies that the vector $\zeta_{j,l}$ converges to a Gaussian random vector with zero-mean components and covariance matrix, $\boldsymbol{R}_{j,l,0}$. Assuming that each time t the fading is not corelated, this will be a $2M \times 2M$ block matrix with the following 2×2 matrices along the diagonal:

$$\boldsymbol{R}_{0,t} = \begin{bmatrix} \mathbb{E}\mathbf{y}^2_{1,t} & \mathbb{E}\mathbf{y}_{1,t}\mathbf{y}^*_{2,t} \\ \mathbb{E}\mathbf{y}_{2,t}\mathbf{y}^*_{1,t} & \mathbb{E}\mathbf{y}^2_{2,t} \end{bmatrix}$$

where $t = t_1, ..., t_M$ and $(\,)^*$ denotes the conjugate transpose, $\mathbb{E}$ denotes the expected value

and

$$\mathbb{E}\mathbf{y}_{1,t}^2 = \sum_{l'=1}^{L} |h_{1,t,l'}|^2 \frac{P}{M} + \sigma^2$$
$$\mathbb{E}\mathbf{y}_{2,t}^2 = \sum_{l'=1}^{L} |h_{2,t,l'}|^2 \frac{P}{M} + \sigma^2$$
$$\mathbb{E}\mathbf{y}_{1,t}\mathbf{y}_{2,t}^* = \sum_{l'=1}^{L} h_{1,t,l'} h_{2,t,l'}^* \frac{P}{M}$$
$$\mathbb{E}\mathbf{y}_{2,t}\mathbf{y}_{1,t}^* = \sum_{l'=1}^{L} h_{2,t,l'} h_{1,t,l'}^* \frac{P}{M} \ .$$

The vector $\bar{\boldsymbol{\zeta}}_{j,l}$ has zero mean and its covariance matrix is the $2M \times 2M$ diagonal block matrix $\bar{\boldsymbol{R}}_{j,l} = \text{diag}(\boldsymbol{R}_{0,t_1}, ..., \boldsymbol{R}_{0,t_M})$. With these quantities we may perform a linear filtering on $\bar{\mathbf{y}}_{j,l}$ to form an SNR-optimal linear estimate of $v_{j,l}$ as

$$z_{j,l}^{(0)} = \sqrt{\frac{P}{M}} \bar{\mathbf{h}}_l^* \bar{\boldsymbol{R}}_0^{-1} \bar{\mathbf{y}}_{j,l} \ ,$$

we may compute the resulting signal to noise ratio (SNR) of $z_{j,l}^{(0)}$ as

$$\gamma_{j,l,0} = \frac{P}{M} \bar{\mathbf{h}}_l^* \bar{\boldsymbol{R}}_{j,l}^{-1} \bar{\mathbf{h}}_l \ .$$

As $v_{i,l} \in \{-1, 1\}$ and takes each of the two values with probability $\frac{1}{2}$ we may form a conditional expectation estimate $\hat{\mathbf{v}}_{i,l}$ as

$$\hat{v}_{j,l}^{(0)} = \mathbb{E}(v_{j,l} | z_{i,l}^{(0)}) = \tanh(z_{j,l}^{(0)} \gamma_{j,l,0}).$$

Repeating this process for all data bits $v_{i,l}$, $i \in [1, N]$ and $l \in [1, k]$ we produce estimates $\hat{v}_{i,l}^{(0)}$ for all of the data bits, we now proceed to the next demodulation iteration by first performing an interference cancellation step by computing the vectors $\bar{\mathbf{y}}_t^{(1)}$ for all values of t and $l \in [1, k]$

$$\begin{aligned}
\mathbf{y}_{j,l,t}^{(1)} &= \mathbf{y}_t - \sum_{(j',l')\in\Im(t)\setminus\{(j,l)\}} \mathbf{h}_{l'}\hat{v}_{j',l'}^{(0)}\sqrt{\frac{P}{M}} \\
&= \sqrt{\frac{P}{M}}\mathbf{h}_l\mathbf{v}_{j,l} + \sqrt{\frac{P}{M}}\sum_{(j',l')\in\Im(t)\setminus\{(j,l)\}} \mathbf{h}_{l'}(\tilde{\mathbf{v}}_{j',l'} - \hat{\mathbf{v}}_{j,l'}^{(0)}) + \boldsymbol{n}_t \\
&= \sqrt{\frac{P}{M}}\bar{\mathbf{h}}_l\mathbf{v}_{j,l} + \boldsymbol{\zeta}_{j,l}^{(1)}
\end{aligned}$$

where $\boldsymbol{\zeta}_{j,l}^{(1)}$ denotes the two-dimensional noise and interference vector for the bit $v_{j,l}$ at time t for the first iteration.

In order to estimate a particular bit $v_{j,l}$ we collect M replicas of $v_{j,l}$ transmitted at times $t_1, t_2, \cdots, t_M$ to produce the $2M$-vector $\bar{\mathbf{y}}_{j,l}^{(1)} = [\boldsymbol{y}_{t_1}^{(1)\,T}, \boldsymbol{y}_{t_2}^{(1)\,T}, \cdots, \boldsymbol{y}_{t_M}^{(1)\,T}]^T$, and the $2M$-vector of the interference and noise $\bar{\boldsymbol{\zeta}}_{j,l}^{(1)} = [\boldsymbol{\zeta}_{t_1}^{(1)\,T}, \boldsymbol{\zeta}_{t_2}^{(1)\,T}, \cdots, \boldsymbol{\zeta}_{t_M}^{(1)\,T}]^T$. This implies

$$\bar{\mathbf{y}}_{j,l}^{(1)} = \sqrt{\frac{P}{M}}\bar{\mathbf{h}}_l v_{j,l} + \bar{\boldsymbol{\zeta}}_{j,l}^{(1)} \tag{3.46}$$

Again, for large values of L, the Central Limit Theorem implies that the vector $\bar{\boldsymbol{\zeta}}_{j,l}^{(1)}$ converges to a Gaussian random vector with zero-mean components and covariance matrix, $\boldsymbol{R}_{j,l,1}$, which is a block diagonal matrix with 2×2 block matrices:

$$\boldsymbol{R}_{1,t} = \begin{bmatrix} \mathbb{E}\mathbf{y}_{1,t}^{(1)\,2} & \mathbb{E}\mathbf{y}_{1,t}^{(1)}\mathbf{y}_{2,t}^{(1)*} \\ \mathbb{E}\mathbf{y}_{2,t}^{(1)}\mathbf{y}_{1,t}^{(1)*} & \mathbb{E}\mathbf{y}_{2,t}^{(1)\,2} \end{bmatrix}$$

Denoting $\sigma_{\text{mmse,l,0}}^2 = \mathbb{E}[1 - \tanh(z_{j,l}^{(0)}\gamma_{l,0})]$ the components of the new covariance matrix

are:

$$\mathbb{E}\mathbf{y}_{1,t}^{(1)\ 2} = \sum_{l=1}^{L} |h_{1,t,l}|^2 \frac{P}{M} \sigma^2_{\text{mmse,l,0}} + \sigma^2$$
$$\mathbb{E}\mathbf{y}_{2,t}^{(1)\ 2} = \sum_{l=1}^{L} |h_{2,t,l}|^2 \frac{P}{M} \sigma^2_{\text{mmse,l,0}} + \sigma^2$$
$$\mathbb{E}\mathbf{y}_{1,t}^{(1)} \mathbf{y}_{2}^{(1)*} = \sum_{l=1}^{L} h_{1,t,l} h^*_{2,t,l} \frac{P}{M} \sigma^2_{\text{mmse,l,0}}$$
$$\mathbb{E}\mathbf{y}_{2,t}^{(1)} \mathbf{y}_{1}^{(1)*} = \sum_{l=1}^{L} h_{2,t,l} h^*_{1,t,l} \frac{P}{M} \sigma^2_{\text{mmse,l,0}} \ .$$

With these quantities we may perform a linear filtering on $\bar{\mathbf{y}}_{j,l}^{(1)}$ to form an SNR-optimal linear estimate of $v_{j,l}$ as

$$z_{j,l}^{(1)} = \sqrt{\frac{P}{M}} \bar{\mathbf{h}}_l^* \bar{\boldsymbol{R}}_1^{-1} \bar{\mathbf{y}}_{j,l}^{(1)} \ .$$

we may compute the resulting signal to noise ratio (SNR)

$$\gamma_{i,l,1} = \frac{P}{M} \bar{\mathbf{h}}_l^* \bar{\boldsymbol{R}}_1^{-1} \bar{\mathbf{h}}_l \ .$$

As $v_{j,l} \in \{-1, 1\}$ and takes each of the two values with probability $\frac{1}{2}$ we may form a conditional expectation estimate $\hat{v}_{j,l}$ as

$$\hat{v}_{j,l}^{(1)} = \mathbb{E}(v_{j,l} | z_{j,l}^{(1)}) = \tanh(z_{j,l}^{(1)} \gamma_{j,l,1}).$$

Repeating this process for all data bits $v_{j,l}$, $i \in [1, N]$ and $l \in [1, k]$ we produce estimates $\hat{v}_{j,l}$ for all of the data bits, one repeat all demodulation iterations by computing may proceed to the next demodulation iteration $i > 1$ by first performing an interference

cancellation step by computing the vectors

$$\mathbf{y}_t^{(i)} = \mathbf{y}_t^{(i-1)} - \sqrt{\frac{P}{M}} \sum_{(j',l') \in \mathfrak{I}(t) \setminus \{(j,l)\}} h_{l'} \hat{v}_{j',l'}^{(i-1)}$$

for all t, combining these copies to produce $\bar{\mathbf{y}}_{j,l}^{(i)}$ and applying a linear filter on this vector to produce a new SNR optimal linear estimate

$$z_{j,l}^{(i)} = \sqrt{\frac{P}{M}} \bar{\mathbf{h}}_l^* \bar{\boldsymbol{R}}_i^{-1} \bar{\mathbf{y}}_{j,l}^{(i)}$$

with signal to noise ratio (SNR)

$$\gamma_{j,l,i} = \frac{P}{M} \bar{\mathbf{h}}_l^* \bar{\boldsymbol{R}}_i^{-1} \bar{\mathbf{h}}_l \ ,$$

using the new covariance matrix $\boldsymbol{R}_{j,l,i} = diag(\boldsymbol{R}_{t_1}, ..., \boldsymbol{R}_{t_M})$, where

$$\boldsymbol{R}_t = \begin{bmatrix} \mathbb{E}\mathbf{y}_{1,t}^{(i)\ 2} & \mathbb{E}\mathbf{y}_{1,t}^{(i)} \mathbf{y}_{2,t}^{(i)*} \\ \mathbb{E}\mathbf{y}_{2,t}^{(i)} \mathbf{y}_{1,t}^{(i)*} & \mathbb{E}\mathbf{y}_{2,t}^{(i)\ 2} \end{bmatrix}$$

with components:

$$\begin{aligned}
\mathbb{E}\mathbf{y}_{1,t}^{(i)\ 2} &= \sum_{l=1}^{L} |h_{1,t,l}|^2 \frac{P}{M} \sigma^2_{\text{mmse,l,i-1}} + \sigma^2 \\
\mathbb{E}\mathbf{y}_{2,t}^{(i)\ 2} &= \sum_{l=1}^{L} |h_{2,t,l}|^2 \frac{P}{M} \sigma^2_{\text{mmse,l,i-1}} + \sigma^2 \\
\mathbb{E}\mathbf{y}_{1,t}^{(i)} \mathbf{y}_2^{(i)*} &= \sum_{l=1}^{L} h_{1,t,l} h_{2,t,l}^* \frac{P}{M} \sigma^2_{\text{mmse,l,i-1}} \\
\mathbb{E}\mathbf{y}_{2,t}^{(i)} \mathbf{y}_1^{(i)*} &= \sum_{l=1}^{L} h_{2,t,l} h_{1,t,l}^* \frac{P}{M} \sigma^2_{\text{mmse,l,i-1}} \ .
\end{aligned}$$

After the final $I-th$ iteration we pass the bit estimates $\hat{v}_{j,l}^{(I)}$ to the error correction decoders. These operate with the bit estimates with $SNR's\ \gamma_{l,I}$.

This demodulation approach produces a similar recursion equation, however it is now a vector recursion. This vector recursion is not conservative and so a Lyapunov potential function cannot be determined to study its threshold saturation behaviour.

Chapter 4

Compressed Sensing Using Message Passing

4.1 Introduction

The recovery of an unknown high-dimensional sparse signal based on a collection of noisy measurements has been studied extensively in the past decade [28, 29, 30, 31, 32, 33]. The interest in estimation problems with sparsity constraints arises from the potential applications in signal processing, such as denoising, compression and sampling [34, 35, 36, 37, 38]. The original compressed sensing problem considered noisy measurements of a deterministic vector, and it is assumed that x_0 has a sparse representation $x_0 = D\alpha_0$ where D is a given dictionary and the majority of the entries of α_0 are zero.

This permits the representation of x_0 as a linear combination of a small number of "atoms" or columns of D. Thus the goal of any estimation technique is to recover either x_0 or its sparse representation given by α_0. There are several practical approaches to this problem, these include ℓ_1 relaxation methods such as the Dantzig selector [33], Basis pursuit denoising (BPDN) which is also known as the Lasso. [28, 29, 39]. These approaches rely on methods from linear programming (LP), and these approaches can be substantially more expensive in applications than the standard linear reconstruction schemes. As an alternative one can consider iterative algorithms that achieve reconstruction performance comparable to the LP-based approaches while running considerably faster. Examples of these are greedy algorithms such as thresholding and orthogonal matching pursuit (OMP) [40], and implementations of message passing.

Several authors have approached compressed sensing from the information-theoretic perspective [41, 42, 43, 44]. Notably Wu and Verdu presented information-theoretic lower bounds on the undersampling rate in terms of Rényi information dimensions of the estimated vector for noiseless [43] and noisy sensing [44] cases. This has allowed for compressed sensing to be considered for vectors with components drawn from a given set based on some probability distribution which allows one to drop the sparseness requirement on the input vector. The connection between achievable compressed sensing undersampling

rates and multi-user efficients were computed using methods from statistical mechanics [44, 45].

In [46] it is shown that a special construction of band-diagonal sensing matrices built from random Gaussian matrices and the AMP algorithm have been used to approach the ultimate reconstruction limits according to Rényi information dimension. In terms of expanding the achievable limits of the reconstruction, spatial graph coupling has made significant progress. It has been shown by Kudekar and Pfister [47], and Krzakala et. al. [48] that coupled sensing matrices can be used efficiently for compressed sensing.

4.2 Sparse Estimation Problems

In this section we describe two common estimation problems. Wherein the unknown parameter has a sparse representation with respect to a known dictionary. A measurement of the unknown deterministic signal $\mathbf{x}_0 \in \mathbb{R}^n$ is given by $\mathbf{y} \in \mathbb{R}^m$

$$\mathbf{y} = \mathbf{A}\mathbf{x_0} + \mathbf{w} \tag{4.1}$$

where $\mathbf{w}$ is an independent, identically distributed (IID) Gaussian noise with zero mean and variance σ^2, and $\mathbf{A}$ is a known $m \times n$ matrix. We require that A is non-adaptive, implying that measurements do not depend on measurements made previously.

It is assumed as prior knowledge that there exists a sparse representation of $\mathbf{x}_0$, i.e.,

$$\mathbf{x}_0 \in S = \{\mathbf{x} \in \mathbb{R}^n \ : \ \mathbf{x} = \mathbf{D}\alpha, \ ||\alpha||_0 \leq s\}. \tag{4.2}$$

The set S describes signals $\mathbf{x}$ which can be formed from linear combinations of s columns from $\mathbf{D}$ (commonly called atoms). The dictionary $\mathbf{D}$ is a known $n \times p$ matrix with $n \leq p$, and it is further assumed that $s < p$ to ensure only a subset of atoms in $\mathbf{D}$ can be used to represent any signal in S, and that this number s is known.

As an example, in [16] the authors consider three possible compressed sensing problems which are similar to (4.1) with potentially Gaussian random matrices $\boldsymbol{A}$ and information vectors of three types:

- Sparse vectors with non-negative entries.
- Sparse vectors with signed components.

- Vectors with components $x_i \in [-1, 1]$ and at most $k \ll n$ entries with values in the interior $(-1, 1)$.

Given such a measurement, the goal is to obtain an estimate $\hat{\mathbf{x}}$ whose average mean squared error (MSE),

$$MSE = E[||\hat{\mathbf{x}} - \mathbf{x}_0||_2^2] \tag{4.3}$$

is as small as possible. As $\mathbf{x}_0$ is deterministic, the expectation above is taken over the noise of $\mathbf{w}$, but not $\mathbf{x}_0$, and so we may treat this as a function of $\mathbf{x}_0$.

It may also be of interest to estimate the coefficient vector α_0 for which $\mathbf{x}_0 = \mathbf{D}\alpha_0$, say for the purpose of model selection [28, 33]. In such a scenario, one must construct an estimator $\hat{\alpha}$ whose MSE, $E[||\hat{\alpha} - \alpha_\mathbf{0}||_2^2]$ is as low as possible. Unless $\mathbf{D}$ is unitary, estimating α_0 is not equivalent to estimating $\mathbf{x}_0$. Of course when estimating for α_0 we may combine the two matrices to produce an equivalent problem

$$\mathbf{y} = \mathbf{H}\boldsymbol{\alpha_0} + \mathbf{w} \tag{4.4}$$

where $\mathbf{H} = \mathbf{AD}$ is an $m \times p$ matrix and

$$\boldsymbol{\alpha_0} \in T = \{\boldsymbol{\alpha} \in \mathbb{R}^p \ : \ ||\boldsymbol{\alpha_0}||_0 \leq s\}. \tag{4.5}$$

This is just a special case of (4.1) and (4.2) with the p-dimensional identity matrix acting as $\boldsymbol{D}$ in (4.2). Signal estimation problems differ in the properties of the dictionary $\mathbf{D}$ and measurement matrix $\mathbf{A}$. For example, distinctly different problems arise depending on whether the dictionary is a basis or an over-complete frame.

Alternatively, in accordance with the results [43, 44, 45] one can drop the requirement that $\boldsymbol{x}$ is sparse and instead consider vectors with components as random variables with some probability distribution. In this work we consider the sensing problem

$$\mathbf{y} = \mathbf{A}\mathbf{x_0} + \mathbf{w}$$

where $\mathbf{x}$ with entries arising from the binomial distribution, $\mathbf{w}$ is a noise vector as stated in (4.1), and $\mathbf{A}$ is a band-diagonal sparse $m \times n$ matrix.

4.3 Inference from Message Passing

We have considered linear measurement problems of the form [49]

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}$$

where $\boldsymbol{A}$ is a known measurement matrix and $\boldsymbol{w}$ a noise vector. The graphical model approach to reconstructing the original vector $\boldsymbol{x}$ is to postulate a joint probability distribution on the transmitted and received vectors which is of the form

$$p(d\boldsymbol{x}, d\boldsymbol{y}) = p(d\boldsymbol{y}|\boldsymbol{x})p(d\boldsymbol{x}) \tag{4.6}$$

The conditional distribution $p(d\boldsymbol{y}|\boldsymbol{x})$ models the noise process, while the prior $p(d\boldsymbol{x})$ encodes information on the vector. In compressed sensing this can describe its sparsity properties. Either of these distributions (or both) is factorized according to a specific graph structure, and the resulting posterior distribution $p(d\boldsymbol{y}|\boldsymbol{x})$ can be used to infer $\boldsymbol{x}$ given $\boldsymbol{y}$.

To illustrate this, we consider the conditional distribution of $\boldsymbol{y}$ given $\boldsymbol{x}$ which is equivalent to specifying the distribution of the noise vector $\boldsymbol{w}$. We will assume that $p(\boldsymbol{w})$ is Gaussian with zero mean and variance $\beta^{-1}\boldsymbol{I}$, and so

$$p_\beta(d\boldsymbol{y}|\boldsymbol{x}) = \left(\frac{\beta}{2\pi}\right)^{\frac{n}{2}} \exp\left(-\frac{\beta}{2}||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||_2^2\right) d\boldsymbol{y} \tag{4.7}$$

For the prior $p(d\boldsymbol{x})$ we use the product distribution with identical factors.

The posterior distribution of $\boldsymbol{x}$ given observations $\boldsymbol{y}$ admits an explicit expression derived from $p_\beta(d\boldsymbol{x}, d\boldsymbol{y})$

$$p_\beta(d\boldsymbol{y}|y) = \left(\frac{\beta}{2\pi}\right)^{\frac{n}{2}} \frac{1}{p(\boldsymbol{y})} \exp\left(-\frac{\beta}{2}||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||_2^2\right) d\boldsymbol{y}\Pi_{i=1}^n p(d\boldsymbol{x}_i). \tag{4.8}$$

Despite this explicit expression, computing marginals or expectations of this distribution is computationally hard.

Notice that the square residuals $||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||_2^2$ decompose into a sum of m terms, producing a simpler expression

$$p_\beta(d\boldsymbol{y}|y) = \left(\frac{\beta}{2\pi}\right)^{\frac{n}{2}} \frac{1}{p(\boldsymbol{y})} \Pi_{a=1}^m \exp\left(-\frac{\beta}{2}(y_a - (\boldsymbol{A}\boldsymbol{x})_a)^2\right) d\boldsymbol{y} \Pi_{i=1}^n p(d\boldsymbol{x}_i). \tag{4.9}$$

where $(Ax)_a$ is the $a-$th entry of the vector $\boldsymbol{A}\boldsymbol{x}$. This factorized structure can be expressed as bipartite graph including a "variable node" $i \in [1, n]$ for each variable x_i and a "factor node" $a \in [1, m]$ for each term $\Psi_a(\boldsymbol{x}) = \exp\left(-\frac{\beta}{2}(y_a - (Ax)_a)^2\right)$ Variable i and factor a are connected by an edge if and only if $\Psi_a(\boldsymbol{x})$ depends non-trivially on x_i (i.e., $A_{ai} \neq 0$). We will call such a bipartite graph, a *factor graph*.

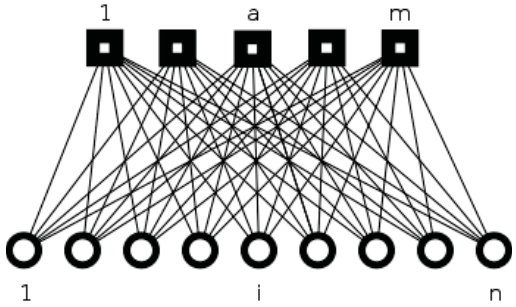

Figure 4.1: Factor graph associated to the probability distribution (4.9). Circles correspond to variables $x_i,\ i \in [1, n]$ and squares correspond to measurements $y_a,\ a \in [1, m]$.

An estimate of the signal can be extracted from the posterior distribution in several ways. One could use the conditional expectation

$$\hat{\boldsymbol{x}}_\beta(\boldsymbol{y}; p) = \int_{\mathcal{R}^n} \boldsymbol{x} p_\beta(d\boldsymbol{y}|\boldsymbol{x}), \tag{4.10}$$

this estimate achieves the minimal mean square error if $p_\beta(d\boldsymbol{x}, d\boldsymbol{y})$ is the **actual** joint distribution of the problem. In practical terms, this estimator is hard to compute. To produce something more tractable, we suppose that $p(dx_i) = p_{\beta,h}(dx_i) = c f_{\beta,h}(x_i) dx_i$ for $f_{\beta,h}(x_i) = e^{-\beta h(x_i)}$ an un-normalized probability density function. As β becomes large, the integral (4.10) becomes dominated by the vector $\boldsymbol{x}$ with the highest posterior probability p_β and one can replace the integral in $d\boldsymbol{x}$ with a maximization over $\boldsymbol{x}$ and define the following minimization over a cost function $C_{A,\boldsymbol{y}}(\boldsymbol{z}; h)$,

$$\hat{\boldsymbol{x}}(\boldsymbol{y}; h) = argmin_{\boldsymbol{z}\in\mathcal{R}^n} C_{A,\boldsymbol{y}}(\boldsymbol{z}; h)$$
$$C_{A,\boldsymbol{y}}(\boldsymbol{z}; h) = \tfrac{1}{2}||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{z}||_2^2 + \textstyle\sum_{i=1}^{n} h(z_i) \tag{4.11}$$

where h arose from the assumption of an un-normalized probability density function. Assuming the cost function has a unique minimum. This problem is amenable to iterative approaches like LASSO, BPDN and mesage passing.

4.4 Message Passing: The Min-sum Algorithm

Message passing is often too complex to implement, instead the solution to the optimization problem (4.11) can be determined using an appropriate *approximate message passing algorithm*. To illustrate this, we start with the min-sum algorithm, which is an iterative belief-propagation algorithm [49].

We introduce a general cost function over $\boldsymbol{x} = (x_1, \cdots, x_n)$ that decomposes according to a factor graph as in Figure 4.1:

$$C(\boldsymbol{x}) = \sum_{a\in F} C_a(x_{\partial a}) + \sum_{i\in V} C_i(x_{i'}). \tag{4.12}$$

Here F is the set of m factor nodes, and V is the set of n variable nodes. We denote the set of neighbors of factor node a (those variable nodes connected to the factor node a) as ∂a and $\boldsymbol{x}_{\partial x} = \{x_i : i \in \partial a\}$. The basic variables here are messages; a message is associated to each directed edge in the underlying factor graph. In our case the messages are functions on the optimization variables which denoted as $J^t_{i\to a}(x_i)$ and $\hat{J}^t_{a\to i}(x_i)$ for "variable to factor" and "factor to variable" messages respectively at iteration t. Modulo up to addition to additive constant, the messages are updated at t-th iteration as

$$J^{t+1}_{i\to a}(x_{i'}) = C_i(x_{i'}) + \textstyle\sum_{b\in\partial i\setminus a} \hat{J}^t_{b\to i}(x_{i'}),$$
$$\hat{J}^{t+1}_{a\to i}(x_{i'}) = \min_{\boldsymbol{x}_{\partial a\setminus i}} \left[C_a(\boldsymbol{x}_{\partial a}) + \textstyle\sum_{j\in\partial a\setminus i} J^t_{j\to a}(x_{i'})\right]. \tag{4.13}$$

Iterating over t, the optimum is approximated by

$$\begin{aligned} \hat{x}_i^{t+1} &= arg\min_{\boldsymbol{x}_i \in \mathcal{R}^n} J_i^{t+1}(x_{i'}), \\ J_i^{t+1}(x_{i'}) &= C_i(x_{i'}) + \textstyle\sum_{b \in \partial i} \hat{J}_{b \to i}^t(x_{i'}). \end{aligned} \tag{4.14}$$

There are many ways to modify the basic iteration to provide better properties. The approximate message passing algorithm presented in Chapter 5 and 6 is derived to simplify the message passing rules from section 3.5:

$$\begin{aligned} x_{j \to a}^{t+1} &= \eta_t \left(\textstyle\sum_{a' \neq a,\ a' \in \partial j} z_{a' \to j}^t \right) \\ z_{a \to j}^t &= y - \sqrt{P} \textstyle\sum_{j' \neq j,\ j' \in \partial a} x_{j' \to a}^t, \end{aligned} \tag{4.15}$$

where $\eta_t(\cdot) = \tanh(\cdot)$ denotes the threshold function applied component-wise to the vector input and $\sqrt{P}$ is a constant. Over t iterations the optimum is

$$\hat{x}_i^{t+1} = arg \min_{\boldsymbol{x}_i \in \mathcal{R}^n} \left[\sum_{b \in \partial i} z_{b \to i}^t \right]. \tag{4.16}$$

Chapter 5

Coupling Information Transmission with Approximate Message Passing

In this section we focus on the coupling information transmission multi-user communication technique based on the spatial graph coupling principle [3] where the packets transmitted from their respective datastreams have perfect timing. Each user transmits one or more independently encoded data streams over the multiple access channel.

Each data stream is encoded and modulated such that the relations between the data bits and modulation symbols within each transmitted stream are represented by a bipartite graph. The data streams are transmitted with time offsets enabling coupling of the transmitted graphs into a single continuous graph chain at the receiver. The receiver performs iterative message-passing interference cancellation and data estimation on the received graph chain, followed by individual error-correction decoding, performed for each data stream individually, providing near-capacity operation [2, 11].

Modulation of each data stream is accomplished with repetition, permutation, and spreading of data. We start with an introduction to the coupling information transmission system and present its compressed sensing representation in Section 5.1, we formulate the AMP-based demodulation algorithm in Section 5.3, and give the derivation of the algorithm in the Appendix. Simulation results and conclusions are given in Section 5.4.

In terms of expanding the achievable limits of the reconstruction, spatial graph coupling has made significant progress. It has been shown by Kudekar and Pfister [47], and Krzakala et. al. [48] that coupled sensing matrices can be used efficiently for compressed sensing. We consider a demodulation technique based on approximate message passing (AMP) proposed by Donoho et al. [16] for compressed sensing problems, and construct a version of the AMP which is specially derived for our multi-user situation.

We start with an introduction to the coupling information transmission system and present its compressed sensing representation in Section 5.1, we formulate the AMP-based

demodulation algorithm in Section 5.3, and give the derivation of the algorithm in the Appendix. Simulation results and conclusions are given in Section 5.4.

5.1 System Model

We consider a communication scenario where a number of distinct transmitters send independently encoded and modulated streams of data to the common receiver. The encoding format follows the coupling data transmission technique outlined in [11][2]. Each data stream $\ell = 1, 2, 3, \cdots$ is encoded and modulated as shown in Fig. 5.1. First, a binary information sequence $\boldsymbol{u}_\ell = u_{1,\ell}, u_{2,\ell}, \cdots, u_{K,\ell}$, where $u_{j,\ell} \in \{-1, 1\}$ is encoded with a binary error correction code of rate $R = K/N$ to produce an N bit codeword $\boldsymbol{v}_\ell$.

Each bit $v_{j,\ell} \in \{-1, 1\}$, $j = 1, 2, \cdots, N$ of $\boldsymbol{v}_\ell$, is then replicated M times. The resulting sequence is permuted using a binary permutation matrix $\boldsymbol{P}_\ell$ and bitwise multiplied with a pseudo-random binary signature sequence specific to data stream ℓ, $\boldsymbol{s}_\ell = s_{1,\ell}, s_{2,\ell}, \cdots, s_{MN,\ell}$, with $s_{j,\ell} \in \{-1, 1\}$, producing a sequence $\tilde{\boldsymbol{v}}_\ell = \tilde{v}_{1,\ell}, \cdots, \tilde{v}_{MN,\ell}$. The resulting binary phase shift keying (BPSK) signal $\tilde{\boldsymbol{v}}_\ell$ is multiplied by an amplitude $\sqrt{P_\ell/M}$ and transmitted over the Gaussian multiple access channel (see Fig. 5.1). The bit error rates (BER) are computed assuming error-correction codes which are capacity achieving on a binary-input Gaussian channel given post detection SNR. Thus the goal for demodulation is to reduce the SNR to an acceptable threshold in order to pass to the error-correction codes.

The encoding and modulation of the vector $\tilde{\boldsymbol{v}}_\ell$ can be represented using matrix notation as

$$\sqrt{\frac{P_\ell}{M}}\tilde{\boldsymbol{v}}_\ell = \sqrt{\frac{P_\ell}{M}}\boldsymbol{S}_\ell\boldsymbol{P}_\ell\boldsymbol{R}\boldsymbol{v}_\ell = \boldsymbol{B}_\ell\boldsymbol{v}_\ell \tag{5.1}$$

where $\boldsymbol{R}$ is the $MN \times N$ band-diagonal repetition matrix with M ones in each row, $\boldsymbol{P}_\ell$ is an $MN \times MN$ permutation matrix, and $\boldsymbol{S}_\ell = \text{diag}(\boldsymbol{s}_\ell)$ is an $MN \times MN$ diagonal matrix that has the signature sequence $\boldsymbol{s}_\ell$ on its diagonal.

We focus on a transmission scenario where k modulated data streams are transmitted over the channel with time offsets of $\theta_1 \leq \theta_2 \leq \theta_3 \leq \cdots \leq \theta_k$ symbol (chip) durations.[1]

[1] Without loss of generality we assume that the received data streams are chip synchronized. Processing of fractional offsets is discussed in Section 5.3.

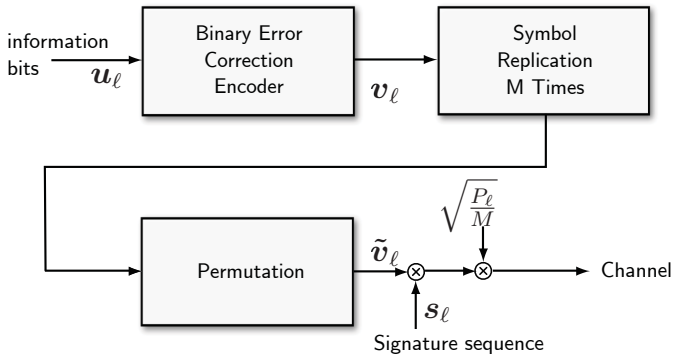

Figure 5.1: Encoding and modulation of the ℓth data stream.

The received vector $\boldsymbol{y}$ is given by

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{v} + \boldsymbol{\xi} \, . \tag{5.2}$$

where

$$\boldsymbol{A} = \boldsymbol{C} \left[\boldsymbol{B}_1^{\mathrm{T}}, \boldsymbol{B}_2^{\mathrm{T}}, \cdots, \boldsymbol{B}_k^{\mathrm{T}} \right]^{\mathrm{T}} \, , \tag{5.3}$$

$\boldsymbol{C}$ is a coupling matrix illustrated in Figure 5.2, and $\boldsymbol{v} = [\boldsymbol{v}_1^{\mathrm{T}}, \boldsymbol{v}_2^{\mathrm{T}}, \cdots, \boldsymbol{v}_k^{\mathrm{T}}]^{\mathrm{T}}$ is the composite encoded data vector of all data streams. Multiplication by the coupling matrix performs superposition of the data streams transmitted with time offsets. The noise vector $\boldsymbol{\xi}$ consists of iid real-valued Gaussian noise samples $\xi_t \sim \mathcal{N}(0, \sigma^2)$ with $t = 1, 2, 3, \cdots$.

It is convenient to represent each transmitted data stream using a bipartite graph in which each encoded data bit corresponds to a variable node and each modulated symbol by a constraint node. A variable node has M outgoing edges connecting it to the constraint nodes containing the M replicas of the variable node bit.

Data stream transmission with time offsets initiates coupling of the graphs corresponding to the individual data streams into a continuous graph chain at the receiver. Hence, we refer to such transmission as coupling transmission (a regularized coupling transmission with equispaced offsets is described in Section 5.4). A constraint node in the received graph corresponds to a particular symbol (chip) time instance and is connected to all variable nodes for which those respective bits are transmitted at that particular time. Let us denote the set of indices of the variable nodes connected to constraint node t by $\mathcal{J}(t)$ and the indices of constraint nodes connected to variable node j by $\mathcal{T}(j)$[2] (see Fig. 5.4).

[2]We omit index l and index all data bits with a single index j for simplicity.

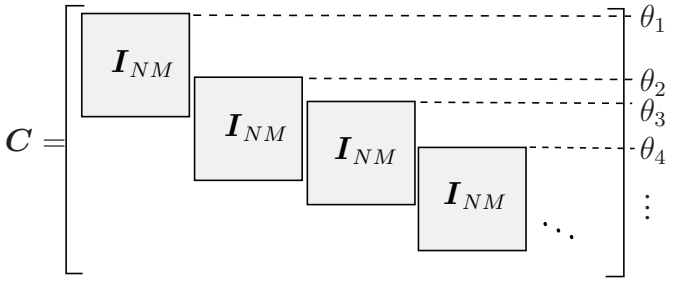

Figure 5.2: The $MN \times MN$ coupling matrix $\boldsymbol{C}$ consists of identity matrices surrounded by zeros. Each identity matrix $\boldsymbol{I}_{MN}$ corresponds to an incoming data stream and its row offset corresponds to the time offset (starting time) of that data stream.

The alternative "compressed sensing" representation of the system is based on the sparse $m \times n$ matrix $\boldsymbol{A}$ (see (5.2)). Note that the non-zero elements in row t, $t = 1, 2, \cdots, m$ of $\boldsymbol{A}$ are indexed by the set $\mathcal{J}(t)$ and represent the indices of the data bits v_j that are transmitted at time t. The non-zero elements in a column j, $j = 1, 2, \cdots, n$, of $\boldsymbol{A}$ represent times at which a particular bit v_j is transmitted over the channel and are indexed by the set $\mathcal{T}(j)$.

5.2 Demodulation and Decoding

The receiver starts with joint iterative demodulation (multi-user detection) of the data streams. At each iteration of the iterative demodulation process each bit v_j is estimated based on the M received symbols that contain the M transmitted replicas of v_j (bit estimation stage). The estimated bits are then utilized to remodulate the data streams and perform interference cancellation (interference cancellation stage). After the interference cancellation stage the data streams are again passed to the bit estimation stage and the process is repeated for a number of iterations until most of the inter-stream interference is removed.

The individual error correction decoding of the data streams is performed *after* the final demodulation iteration. The decoding is not included into the demodulation loop to simplify the system implementation and reduce the computational complexity. The system is still capacity achieving asymptotically [11][2].

The iterative demodulation process can also be seen as message-passing on the system's graph chain. The interference cancellation message $z^i_{t \to j}$ from constraint node t to variable node j at demodulation iteration i and the bit estimation message back from node t to node

j at demodulation iteration $i+1$ are given by

$$z^i_{t\to j} = y_t - \sum_{j'\neq j,\ \ j'\in\mathcal{J}(t)} A_{tj'} v^i_{j'\to t} \tag{5.4}$$

$$v^{i+1}_{j\to t} = \tanh\left(\sum_{t'\in\mathcal{T}(j),t'\neq t} A_{t'j}\frac{z^i_{t'\to j}}{\sigma^{2,i}_{t'}}\right), \tag{5.5}$$

where $\sigma^{2,i}_t$ is the noise and interference power experienced by the tth symbol of the received signal at demodulation iteration i (see Fig. 5.4). The bit estimation performs maximum ratio combining over all signals $z^i_{t\to j}$ containing bit v_j and then takes the conditional expectation estimate (see [11]). The interference cancellation step features subtraction of all data streams except stream j from the received signal.

The number of messages computed, stored, and transmitted at each iteration of the message-passing based iterative demodulation scales with the total number of the data bits as well as the repetition factor M. An implementation, besides the main iteration loop, requires a loop involving the exclusion of one message $t' = t$ or $j' = j$.

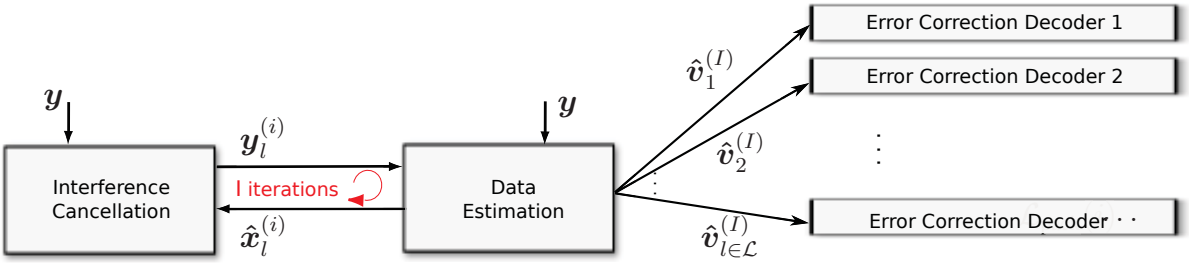

Figure 5.3: Diagram of the iterative receiver, with error correction decoding for each data stream.

5.3 The Proposed AMP-Based Demodulation Algorithm

Let $\hat{\boldsymbol{v}}^i$ denote an estimate of the entire data vector $\boldsymbol{v}$ computed at iteration i and $\boldsymbol{z}^i$ denote the residual noise and interference between the data streams at iteration i. The

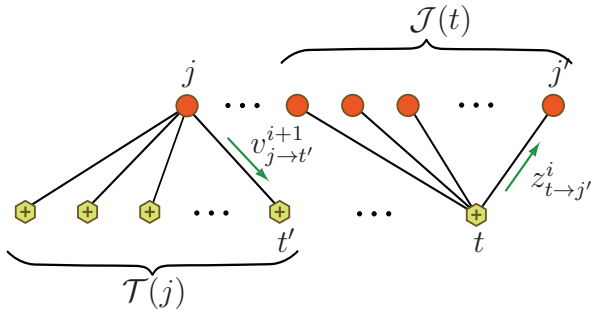

Figure 5.4: Graph representation of the modulated data.

proposed algorithms steps are as follows:

$$\boldsymbol{z}^i = \boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{v}}^i + \frac{\boldsymbol{z}^{i-1}}{\boldsymbol{\sigma}^{2,i-1}} \circ (\boldsymbol{A}_2 \operatorname{sech}^2(\boldsymbol{w}^{i-1})), \tag{5.6}$$

$$\boldsymbol{w}^i = \boldsymbol{A}^T \frac{\boldsymbol{z}^i}{\boldsymbol{\sigma}^{2,i}} + \boldsymbol{A}_2^T \frac{1}{\boldsymbol{\sigma}^{2,i}} \circ \hat{\boldsymbol{v}}^i \,. \tag{5.7}$$

$$\hat{\boldsymbol{v}}^{i+1} = \tanh(\boldsymbol{w}^i) \tag{5.8}$$

where $\circ$ denotes the Hadamard (component-wise) product, $\boldsymbol{A}_2 = A \circ A$ and 'sech' is the hyperbolic secant function. The derivation of the algorithm is given in Appendix. The initialization is done using

$$\boldsymbol{z}^0 = \boldsymbol{y}, \qquad \hat{\boldsymbol{v}}^0 = \boldsymbol{0} \,. \tag{5.9}$$

and $\boldsymbol{\sigma}^{2,i}$ is computed from the variance of the residual $\boldsymbol{z}^i$.

The algorithm description is almost as compact and simple as linear multiuser detection for CDMA such as a mean squared error (MSE) detector with just an addition of two correction terms but without matrix inversions. Moreover, since matrix $\boldsymbol{A}$ is sparse, the complexity of the actual matrix multiplication scale only with the number of non-zero matrix entries. The role of the coupling matrix $\boldsymbol{C}$ to add data streams with time offsets, and therefore, the operation $\boldsymbol{Av}$ can be performed as addition of the vectors $\boldsymbol{B}_\ell \hat{\boldsymbol{v}}_\ell$ with offsets θ_ℓ. The number of operations performed per iteration scales only with the number of data bits or received chips (which differ by the systems load factor). The implementation requires a single iterative loop.

Finally, in the case that the time offsets $\theta_1, \theta_2, \cdots$ are not precisely chip time synchronous the interference cancellation operation in (5.6) can be performed on sample basis.

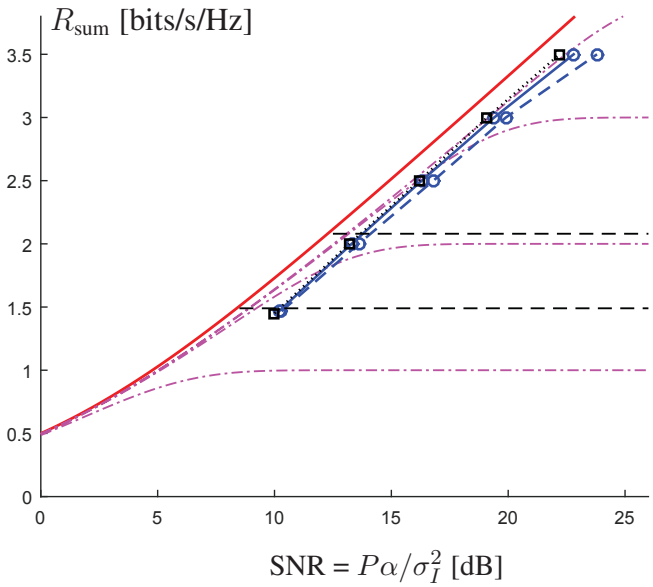

Figure 5.5: Simulated coupling data transmission with AMP demodulation for $M = 250$ (solid circles), $M = 100$ (dashed circles), full message-passing $M = 250$ (dotted squares), capacity of the GMAC channel (solid curve).

5.4 Numerical Results

The proposed algorithm can be applied in the case of arbitrary data stream transmission offsets $\theta_1, \theta_2, \cdots$. We present numerical results for a regularized offset scenario in which the system is asymptotically capacity achieving [2]. Let us define a coupling parameter W and assume that each packet consists of $2W + 1$ sections of length $\eta_W = MN/(2W + 1)$ bits, where MN is divisible by $2W + 1$. We then assume that data streams are transmitted in groups of L_{w} where L_{w} is a parameter determining the system load $\alpha = L_{\text{w}}(2W+1)/M$. The time offsets are multiples of η_W, i.e. $\theta_1 = \theta_2 = \cdots = \theta_{L_{\text{w}}} = 0$, $\theta_{L_{\text{w}}+1} = \theta_{L_{\text{w}}+2} = \cdots = \theta_{2L_{\text{w}}} = \eta_W$ and so on, $\theta_j = \eta_W \lfloor j/L_{\text{w}} \rfloor$. Small random deviations from this time offsets will not influence the results.

Fig. 5.5 demonstrates simulated achievable sum rate of the coupling transmission with the proposed AMP algorithm. We consider coupling window size $W = 12$, code length $N = 250$ and focus on the case of equal power streams, $P_1 = P_2 = \cdots = P$, known to be the hardest for multi-user detection at high system loads. AMP demodulation with repetitions factors $M = 250$ and $M = 100$ and given by the solid and dashed circle curves respectively. The dotted squares curve represents full message passing for $M = 250$. The rates are computed assuming error-correction codes which are capacity achieving on a binary-input Gaussian channel given post detection SNR. The solid curve represents the

channel capacity. The two horizontal dashed lines represent the maximum load $\alpha = 1.49$ achievable by the individually optimal detector of random CDMA [7] and the maximum load $\alpha = 2.08$ achievable by the un-coupled partitioned spreading DMA (PS-CDMA) or sparse CDMA [10] - the limits of the un-coupled systems. Dash-dotted curves show capacities of 2, 4, 8-PAM input Gaussian channel.

The coupled data transmission with AMP follows the capacity of the channel within 2dB while systems loads scale up to $\alpha = 3.5$. The results for smaller repetition factor $M = 100$ loses 0.5dB to $M = 250$ curve and starts to degrade slightly for load $\alpha = 3.5$. The proposed algorithm results in a negligible performance loss compared to complete message-passing while it presents a significant complexity and implementation advantage.

5.5 Derivation of AMP Algorithm

Let us define two quantities representing the elements of the estimated data vector at iteration $i+1$ and the vector of the interference cancellation residual at iteration i

$$\hat{v}_j^{i+1} \stackrel{\text{def}}{=} \tilde{\eta}\left(\sum_{t=1}^{m} A_{tj}\frac{z_{t\to j}^i}{\sigma_t^{2,i}}\right), \qquad z_t^i \stackrel{\text{def}}{=} z_{t\to j}^i - A_{tj}\hat{v}_j^i\,. \tag{5.10}$$

Expressing $z_{t\to j}^i$ from (5.10) we get $z_{t\to j}^i = z_t^i + A_{aj}\hat{v}_j^i$. Hence

$$\sum_{t=1}^{m}\frac{A_{tj}z_{t\to j}^i}{\sigma_t^{2,i}} = \sum_{t=1}^{m}\frac{A_{tj}z_t^i}{\sigma_t^{2,i}} + \sum_{t=1}^{m}\frac{A_{tj}A_{tj}}{\sigma_t^{2,i}}\hat{v}_j^i. \tag{5.11}$$

Substituting (5.11) into (5.5) we obtain

$$\begin{aligned} v_{j\to t}^i &= \tilde{\eta}\left(\sum_{t'\neq t}^{m}\frac{A_{t'j}z_{t'\to j}^{i-1}}{\sigma_{t'}^{2,i-1}}\right) \\ &= \tilde{\eta}\left(\sum_{t'\neq t}^{m}\frac{A_{t'j}}{\sigma_{t'}^{2,i-1}}z_{t'}^{i-1} + \left[\sum_{t'=1}^{m}\frac{A_{t'j}^2}{\sigma_{t'}^{2,i-1}} - \frac{A_{t'j}^2}{\sigma_{t'}^{2,i-1}}\right]\hat{v}_j^{i-1}\right). \\ &\approx \tilde{\eta}\left(\sum_{t'\neq t}^{m}\frac{A_{t'j}}{\sigma_{t'}^{2,i-1}}z_{t'}^{i-1} + \sum_{t'=1}^{m}\frac{A_{t'j}^2}{\sigma_{t'}^{2,i-1}}\hat{v}_j^{i-1}\right). \end{aligned} \tag{5.12}$$

By neglecting the small term A_{tj}^2 ($\approx \frac{P_\ell}{M}$ for some ℓ, or 0) we obtain (5.12). Re-writing the

first term in (5.12) above leads to

$$v_{j\to t}^{i} = \tilde{\eta}\left(\left(\boldsymbol{A}^T \frac{\boldsymbol{z}^{i-1}}{\boldsymbol{\sigma}^{2,i-1}}\right)_j - A_{tj}\frac{z_t^{i-1}}{\sigma_t^{2,i-1}} + \sum_{t'=1}^{m} \frac{A_{t'j}^2}{\sigma_{t'}^{2,i-1}}\hat{v}_j^{i-1}\right)$$

Taking a Taylor series expansion up to the linear term, we find

$$\begin{aligned} v_{j\to t}^{i} \approx\ & \tilde{\eta}\left(\left(\boldsymbol{A}^T \frac{\boldsymbol{z}^{i-1}}{\boldsymbol{\sigma}^{2,i-1}} + \boldsymbol{A}_2^T \frac{1}{\boldsymbol{\sigma}^{2,t-1}} \circ \hat{\boldsymbol{v}}^{i-1}\right)_j\right) \\ & - A_{tj}\frac{z_t^{i-1}}{\sigma_t^{2,i-1}}\tilde{\eta}'\left(\left(\boldsymbol{A}^T \frac{\boldsymbol{z}^{i-1}}{\boldsymbol{\sigma}^{2,i-1}} + \boldsymbol{A}_2^T \frac{1}{\boldsymbol{\sigma}^{2,i-1}} \circ \hat{\boldsymbol{v}}^{i-1}\right)_j\right) \end{aligned} \tag{5.13}$$

where $\circ$ denotes the Hadamard (component-wise) product, $\boldsymbol{A}_2 = A \circ A$, and $\boldsymbol{A}^T$ denotes the transpose of the matrix A

Let us define the vector

$$\boldsymbol{w}^{i-1} \stackrel{\text{def}}{=} \boldsymbol{A}^T \frac{\boldsymbol{z}^{i-1}}{\boldsymbol{\sigma}^{2,i-1}} + \boldsymbol{A}_2^T \frac{1}{\boldsymbol{\sigma}^{2,i-1}} \circ \hat{\boldsymbol{v}}^{i-1}\ . \tag{5.14}$$

We can now represent (5.13) as

$$v_{j\to t}^{i} \approx \hat{v}_j^i - \frac{A_{tj} z_t^{i-1}}{\sigma_t^{2,i-1}}\tilde{\eta}'\left(w_j^{i-1}\right) \tag{5.15}$$

and we also have

$$\hat{v}_j^i = \tilde{\eta}(w_j^{i-1})\ . \tag{5.16}$$

To compute z_t^i we apply (5.15) to (5.10) and obtain

$$\begin{aligned} z_t^i &= y_t - \sum_{j\neq j'}^{n} A_{tj'} v_{j'\to t}^i - A_{tj}\hat{v}_j^i \\ &= y_t - \sum_{j\neq j'}^{n} A_{tj'}\left(\hat{v}_j^i - \frac{A_{tj'} z_t^{i-1}}{\sigma_t^{2,i-1}}\tilde{\eta}'(w_{j'}^{i-1})\right) - A_{tj}\hat{v}_j^i \\ &= y_t - (\boldsymbol{A}\hat{\boldsymbol{v}}^i)_t + \sum_{j'=1}^{n} \frac{A_{tj'}^2 z_t^{i-1}}{\sigma_t^{2,i-1}}\tilde{\eta}'(w_{j'}^{t-1}) - \frac{A_{tj}^2}{\sigma_t^{2,i-1}}\tilde{\eta}(w_j^{t-1}). \end{aligned}$$

Again we will drop $A_{tj}^2 \approx \frac{P_\ell}{M}$ leaving the approximation

$$z_t^i \approx \left(\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{v}}^i + \frac{\boldsymbol{z}^{i-1}}{\boldsymbol{\sigma}^{2,i-1}} \circ (\boldsymbol{A}_2 \tilde{\eta}'(\boldsymbol{w}^{i-1})) \right)_t . \tag{5.17}$$

Finally (5.17), (5.14), and (5.16) give the AMP algorithm.

Chapter 6

Joint Packet Detection and Decoding for Maritime Data Exchange Systems

Random access communications is an important mechanism of accessing the communication channel resources in the upcoming VHF data exchange, VDES as was previously mentioned in Chapter 2. In satellite uplink communications a large varying user (vessel) population is accessing the channel and coordinated channel access will be difficult to implement. At the same time is it important to allow as many vessels as possible to communicate and share channel resources efficiently. Hence, the present time-division based random access mechanism may not be able to cater to the growing traffic demand. In addition to ship-to-satellite requests for downlink data transfer, tracking, and emergency messages, short message exchange between ships and ship-to-shore can also be efficiently handled with random access. Often, vessels do not need to establish a link but just need to send a single packet.

We consider an approach to random access at the physical layer that is based on techniques for simultaneously receiving several packets originating from distinct transmitters. In a random access setting such receiver needs to accomplish two tasks: detect the start of each incoming packet (timing acquisition) and decode the packets. The latter typically also involves cancellation of mutual interference between the packets. It has been observed already in the 90s for the case of code-division multiple access that the detection task limits the multi-packet reception (MPR) capabilities of a receiver [50][51]. Later it has been shown [52][53] that joint detection and decoding can improve the MPR capability. It has also been proved in [54] that for packets modulated using generalized modulation format and for specific packet power distribution the detection results in a loss that amounts to a small fraction of the channel capacity asymptotically.

We focus on a spread spectrum transmission system that uses a constant envelop modulation and a spread preamble as in VDES uplink option [18], suitable for a simple implementation in a vessel transmitter. We propose an algorithm for packet encoding at the

transmitter and joint packet detection and decoding at the receiver. Due to the specific encoding format, from a communications theoretic perspective we can represent each message as a bi-partite graph relating the data bits and the transmitted modulated symbols. At the receiver consecutively received interfering packet graphs couple into a chain of graphs. Our proposed joint iterative preamble detection, interference cancellation and decoding algorithm is based on the AMP technique [16] and operates on the received graph chain. Numerical results demonstrate that incorporation of the preamble detection into the iterative decoding loop leads to a relatively minor performance loss compared to the throughput of the system with an a priori timing knowledge assumption. The total throughput follows the capacity of the multiple access channel. We also demonstrate that packet loads as high as 2.95 packets per packet length per spreading factor are achievable and that the presented random access mechanism outperforms the existing random access techniques.

6.1 System Model

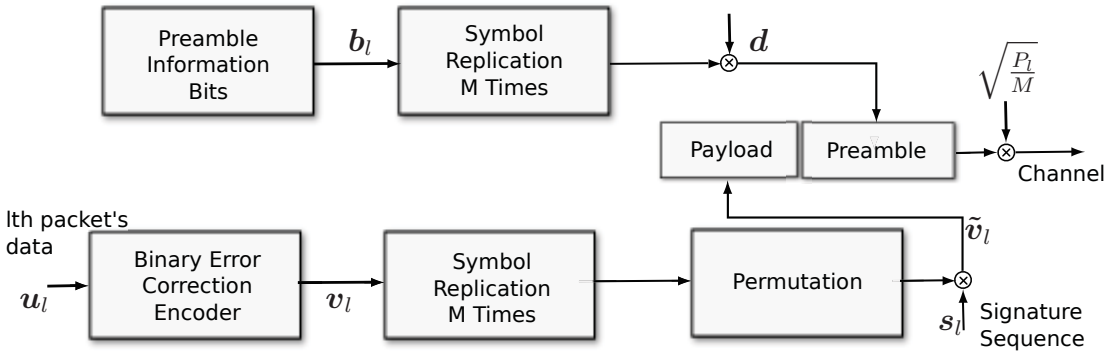

Figure 6.1: Encoding and modulation of one data packet.

We consider a scenario when a number of users send their data packets to the common receiver. The users send packets on demand at random times and create random traffic at the receiver. We assume that all packets are of the same length, the total number of packets transmitted within a frame is L_{pack}, and the ordered set of packet indices is $\mathcal{L}_{\text{pack}} = (1, 2, \cdots, L_{\text{pack}})$. The bit error rates (BER) are computed assuming error-correction codes which are capacity achieving on a binary-input Gaussian channel given post detection SNR. Thus the goal for demodulation is to reduce the SNR to an acceptable threshold in order to pass to the error-correction codes.

Each packet begins with a preamble that will be used for the packet's timing acquisition.

The preamble is followed by a payload consisting of the modulated original data to be sent. To produce the payload of packet $l \in \mathcal{L}_{\text{pack}}$ a transmitter starts with encoding the binary data vector $\boldsymbol{u}_l = u_{1,l}, u_{2,l}, \cdots, u_{K,l}$ with a binary error correction code of rate $R = K/N$. After the encoding each bit of the encoded sequence $\boldsymbol{v}_l = v_{1,l}, v_{2,l}, \cdots, v_{N,l}$, $v_{j,l} \in \{1, -1\}$, is repeated M times and the resulting sequence is permuted using a permutation described by a binary permutation matrix $\boldsymbol{P}_l$. Finally, the permuted sequence is multiplied bit-by-bit by the binary pseudo-random signature sequence $\boldsymbol{s}_l = s_{1,l}, s_{2,l} \cdots, s_{MN,l}$, $s_{j,l} \in \{1, -1\}$, to produce $\tilde{\boldsymbol{v}}_l = \tilde{v}_{1,l}, \tilde{v}_{2,l}, \cdots, \tilde{v}_{MN,l}$. The resulting binary phase shift keying signal is multiplied by a factor $\sqrt{P_l/M}$, where P_l is the transmit power, and is transmitted over the channel (see Fig. 6.1) [1].

Hence, each data bit is spread with a spreading factor M and the resulting spread symbols (chips) are permuted. The modulation format follows the coupling information transmission modulation principle [11], [55], [56], [2] and also resembles interleave division multiple access (IDMA) [14], partition-spreading code-division multiple access (PS-CDMA) [9, 10], generalized modulation [13], and superposition modulation [15]. The modulated signal can be represented as a sparse graph connecting variable nodes corresponding to the data bits $v_{j,l}$ to the channel access nodes corresponding to the times at which M replicas of $v_{j,l}$ are transmitted over the channel. By $\mathcal{T}(j, l)$ we denote the set of times $\tau_1, \tau_2, \cdots, \tau_M$ at which the modulated replicas of bit $v_{j,l}$ are transmitted over the channel.

To help the receiver find the starting time of the packet each transmitter starts its packet with a preamble. Besides its purpose for timing acquisition the preamble also carries information. The information may include the seed of the spreading code and permutation used for modulating the packet's payload. Typically this information is encoded by a strong error correction code. Assume that the preamble of the lth packet contains an information bearing bit sequence $\boldsymbol{b}_l = b_{1,l}, b_{2,l}, \cdots, b_{N_{\text{p}},l}$ where N_{p} is the length of the preamble in bits. To modulate the preamble each bit of the sequence $\boldsymbol{b}_l$ is repeated M times and the resulting sequence is multiplied bit-by-bit by a binary signature sequence $\boldsymbol{d} = d_1, d_2, \cdots, d_{MN_{\text{p}}}$ to produce the preamble sequence $\tilde{\boldsymbol{b}}_l = \tilde{b}_{1,l}, \tilde{b}_{2,l}, \cdots, \tilde{b}_{MN_{\text{p}}}$. The signature sequence $\boldsymbol{d}$ is common to all preambles of all packets. As with the payload, the sequence $\tilde{\boldsymbol{b}}_l$ is finally

[1] To simplify the presentation of the material we consider BPSK modulated packets. The format can be extended to QPSK modulation in a straightforward manner (see [54]).

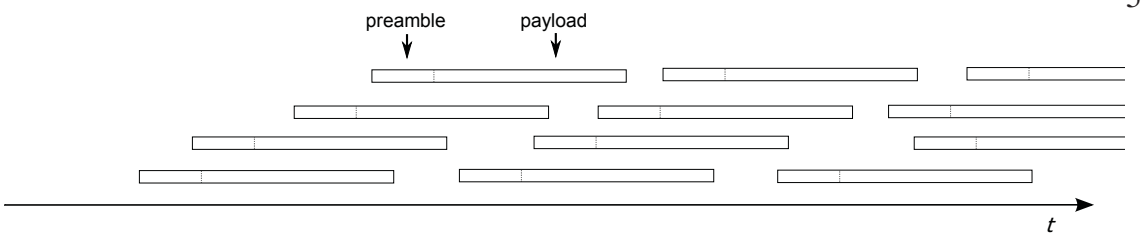

Figure 6.2: A stream of coupled packets with random arrivals of approximately 4 packets/-packet length.

multiplied by the amplitude factor $\sqrt{P_l/M}$ prior to transmission. The pseudo-random signature sequence $\boldsymbol{d}$ is constructed from sequences with low autocorrelation [57] to reduce the rates of false alarms.

Thus, one data packet consists of an $N_{\mathrm{p}}M$-chip preamble followed by an NM-chip payload. The length parameter N_p determines the transmission rate-loss and complexity of the acquisition device, and it is desirable that this preamble length is minimized, especially for the case of transmitting short packets as in VDES.

Combining the payload and preamble data into one vector $\boldsymbol{x}_l = (\boldsymbol{b}_l, \boldsymbol{v}_l)^{\mathrm{T}}$ we can represent the lth modulated packet, $l \in \mathcal{L}_{\text{pack}}$ as a product

$$\tilde{\boldsymbol{x}}_l = \boldsymbol{B}_l \boldsymbol{x}_l \ .$$

The matrix $\boldsymbol{B}_l$ is given by

$$\boldsymbol{B}_l = \sqrt{\frac{P_l}{M}}\, \mathrm{diag}(\boldsymbol{d}, \boldsymbol{s}_l) \begin{pmatrix} \boldsymbol{I}_{MN_{\mathrm{p}}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{P}_l \end{pmatrix} \boldsymbol{R}_{N+N_{\mathrm{p}}} \tag{6.1}$$

$$= \tilde{\boldsymbol{B}}_l \boldsymbol{R}_{N+N_{\mathrm{p}}} \ , \tag{6.2}$$

where $\mathrm{diag}(\boldsymbol{d}, \boldsymbol{s}_l)$ is an $(N+N_{\mathrm{p}})M \times (N+N_{\mathrm{p}})M$ diagonal matrix with sequence $(\boldsymbol{d}, \boldsymbol{s}_l)$ on the main diagonal, $\boldsymbol{I}_{MN_{\mathrm{p}}}$ is an identity matrix of size MN_{p}, and $\boldsymbol{R}_{N+N_{\mathrm{p}}}$ is an $(N+N_{\mathrm{p}})M \times (N+N_{\mathrm{p}})$ matrix which performs M-times replication via right multiplication.

We assume that throughout the frame the users transmit their packets with the overall traffic intensity αM packets/packet length. Hence, at the beginning of the frame the packet traffic increases linearly from an intensity of 0 packets/packet length to αM packets/packet length and stays at that level (see Fig. 6.2). The graphs representing the packets couple in the channel randomly into a chain of graphs. Hence, we obtain a spatially coupled system

on which we will try to exhibit the threshold saturation phenomenon [5][6] via an iterative message-passing-like receiver processing. The fact that the inter-packet interference gradually increases at the beginning of the frame will help to start the detection and decoding process which will then continue successfully for the duration of the entire frame.

We consider communication over the additive white Gaussian noise (AWGN) channel, and so, the received signal $\boldsymbol{y} = (y_1, y_2, y_3, ...)$ is computed as a superposition:

$$y_\tau = \sum_{l=1}^{L(\tau)} \tilde{x}_{\tau,l} + n_\tau \ , \qquad \tau = 0, 1, 2, \cdots$$

where $L(\tau)$ denotes the number of interfering packets at time τ and $n_\tau \sim \mathcal{N}(0, \sigma^2)$ are the iid noise samples.[2] The received signal also allows for representation

$$\boldsymbol{y} = \boldsymbol{A}_{\mathcal{L}_{\text{pack}}} \boldsymbol{x} + \boldsymbol{n} \ , \tag{6.3}$$

where $\boldsymbol{x} = (\boldsymbol{x}_1^{\mathrm{T}}, \boldsymbol{x}_2^{\mathrm{T}}, \cdots, \boldsymbol{x}_{L_{\text{pack}}}^{\mathrm{T}})^{\mathrm{T}}$ is the vector of stacked data and preamble information of all packets, $\boldsymbol{n}$ is the vector of noise samples and

$$\boldsymbol{A}_{\mathcal{L}_{\text{pack}}} = \boldsymbol{C}_{\mathcal{L}_{\text{pack}}} \begin{pmatrix} \boldsymbol{B}_1 \\ \boldsymbol{B}_2 \\ \vdots \\ \boldsymbol{B}_{L_{\text{pack}}} \end{pmatrix} \tag{6.4}$$

where $\boldsymbol{C}_{\mathcal{L}_{\text{pack}}}$ is the coupling matrix representing the addition of all the transmitted packets in the channel. The starting times of the packets $1, 2, \cdots, L_{\text{pack}}$ are denoted by $\theta_1, \theta_2, \cdots, \theta_{L_{\text{pack}}}$. The coupling matrix $\boldsymbol{C}_{\mathcal{L}_{\text{pack}}}$ is depicted in Fig. 6.3. Generalizing this definition, given a set of indices $\mathcal{L} \subset \mathcal{L}_{\text{pack}}$ by $C_{\mathcal{L}}$ we denote a coupling matrix coupling only packets $l \in \mathcal{L}$, by $\boldsymbol{B}_{\mathcal{L}}$ we denote the matrix produced by vertical stacking of matrices $\boldsymbol{B}_l$, $l \in \mathcal{L}$ and by $\boldsymbol{x}_{\mathcal{L}}$ a vector of corresponding packet data $\boldsymbol{x}_l$, $l \in \mathcal{L}$, also stacked vertically. Finally, $\boldsymbol{A}_{\mathcal{L}}$ is computed analogously to (6.4).

[2]Without loss of generality we assume that the received packets are chip-synchronized, otherwise we would consider sample-based detection and interference cancellation processing with the sampling time the receiver would use to oversample the incoming signal.

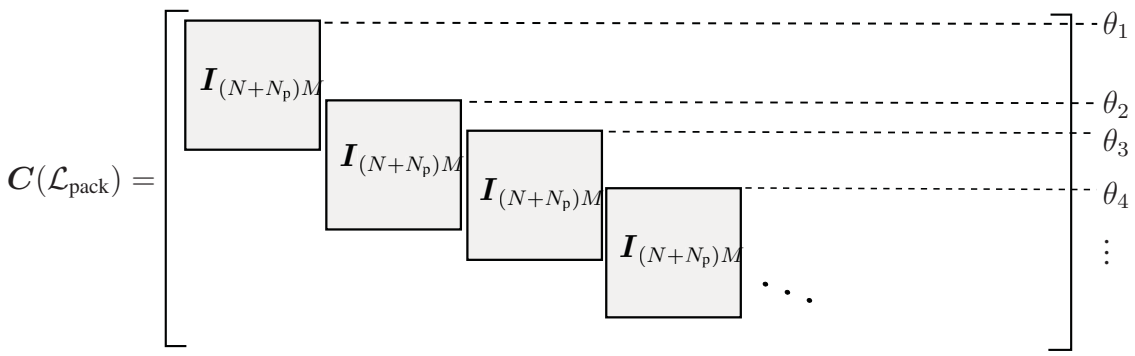

Figure 6.3: The coupling matrix $C(\mathcal{L}_{\text{pack}})$ consists of identity matrices surrounded by zeros. Each identity matrix corresponds to an incoming packet and its row offset corresponds to the packet's starting time.

6.1.1 Preamble Detection

The first iteration starts with the packet detection performed via correlating the preamble's signature sequence $\boldsymbol{d}$ with the incoming signal. The preamble detector slides through the incoming sequence $\boldsymbol{y}$ chip-by-chip and evaluates the correlation detection statistics

$$\eta_\tau = \sum_{j=1}^{N_{\text{p}}} \left| \frac{1}{\sqrt{M}} \sum_{m=1}^{M} y_{\tau+(j-1)M+m-1} d_{(j-1)M+m-1} \right|^2 . \tag{6.5}$$

Since the preamble contains unknown bits we have the inner summation over the chips spreading each bit and compute the energy using the outer summation in (6.5). The statistics η_τ is compared to the decision threshold G and in the case $\eta_\tau \geq G$ a decision about the presence of a packet starting at time τ in the received signal is made.

After the initial detection, some packets, mainly those located at the beginning of the frame, are detected, and will be processed by the data estimation and the interference cancellation block (multiple user detection block) discussed in Section 6.1.2. At the end of the estimation and interference cancellation stage the residual vector $\boldsymbol{z}^{(1)}$ containing the residual interference, undetected packets, and noise will be incoming for the preamble detection at the second iteration. The statistics η_τ will be computed based on $\boldsymbol{z}^{(1)}$ and since $\boldsymbol{z}^{(1)}$ has reduced level of interference compared to $\boldsymbol{y}$ more packets may be detected. The newly detected packets join the estimation and interference cancellation process at the second iteration. The iterations are repeated until all or almost all packets are detected and the interference level is reduced to a level at which the error correction decoders can decode

the packets. A block diagram of the receiver is depicted in Fig. 6.4.

The thresholding parameter G may be determined assuming that the noise and interference experienced by the packets in the multi-user environment can be approximated by a Gaussian random variable. [3] In this case the statistics η_τ follows a Chi-squared distribution with N_p degrees of freedom and non-centrality parameter $N_\mathrm{p}(P+\sigma_\mathrm{I}^2)$ where σ_I^2 is the noise and interference power and P is the power of the packet. If the packet is absent, η_τ follows a central Chi-squared distribution with N_p degrees of freedom. Supposing N_p is large enough, the Central Limit Theorem implies that η_τ may be treated as a Gaussian random variable with expectation and variance

$$\mathbb{E}\eta_\tau = N_\mathrm{p}(P+\sigma_\mathrm{I}^2),\ \ \mathrm{var}(\eta_\tau) = N_\mathrm{p}(P+\sigma_\mathrm{I}^2)\sigma_\mathrm{I}^2$$

for the case of a packet being present at time τ and

$$\mathbb{E}\eta_\tau = N_\mathrm{p}\sigma_\mathrm{I}^2,\ \ \mathrm{var}(\eta_\tau) = N_\mathrm{p}\sigma_\mathrm{I}^4$$

in the case it is absent. With this we may compute the probability of a false alarm as

$$p_\mathrm{fa} = Q\left(\frac{G - N_\mathrm{p}\sigma_\mathrm{I}^2}{\sqrt{N_\mathrm{p}\sigma_\mathrm{I}^4}}\right)\ ,$$

where

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty \exp(-t^2/2)dt\ .$$

Fixing the false alarm probability to some value p_0, and denoting $q_0 = Q^{-1}(p_0)$ allows us to solve for the threshold

$$G = \sigma_\mathrm{I}^2\sqrt{N_\mathrm{p}}(\sqrt{N_\mathrm{p}} + q_0)\ .$$

At iteration i the variance σ_I^2 is estimated directly from the residual vector $\boldsymbol{z}^{(i-1)}$ which is processed for detection. To compute the decision threshold G for time τ the variance σ_I^2 needs to be computed for the part of the vector $\boldsymbol{z}^{(i-1)}$ around the time index τ. The length of this window may be chosen to be $3-5\%$ of the packet length since σ_I^2 varies

[3] This assumption relies on the Central Limit Theorem as we have a collection of a large number of independent interfering packets. The packets' power distribution needs to satisfy Lindeberg Condition.

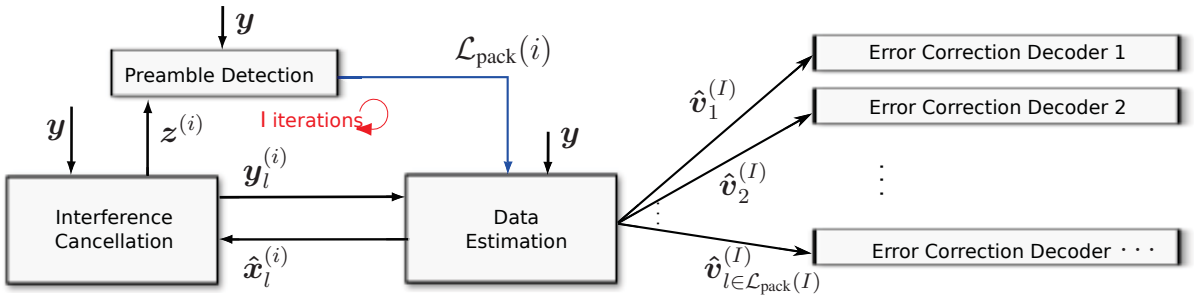

Figure 6.4: Diagram of the iterative receiver that includes joint preamble detection and payload decoding followed by individual error correction decoding.

significantly among different sections of the vector $\boldsymbol{z}^{(i-1)}$. Note also that in practice G has to be adjusted according to the observed false alarm probability, see Section 6.3. Let us denote the set of indices of packets detected by iteration i by $\mathcal{L}_{\text{pack}}(i)$.

The signals produced by the modulators are at the chip level. The modulation format includes a bit-wise multiplication by a known pseudo-random binary sequence prior to transmission. In the above analysis we have assumed that the receiver clock is in sync with each of the transmitters clocks and all the sequences are chip synchronized. However, if that is not the case, both the estimation and the interference cancellation stages of the receiver can still be applied. The interference cancellation would need to be performed at the sample level rather that chip level.

6.1.2 Payload Decoding

One iteration of the payload decoding starts with a data estimation step performed for each bit of the detected packets $l \in \mathcal{L}_{\text{pack}}(1)$. We start with considering estimation of bit $v_{j,l}$ of packet l at the first iteration. At the beginning we form a vector $\boldsymbol{y}_{j,l}$ that consists of M received chips $\{y_\tau\}_{\tau\in\mathcal{T}(j,l)}$, $\mathcal{T}(j,l) = \{\tau_1, \tau_2, \cdots, \tau_M\}$ that contain $v_{j,l}$

$$\boldsymbol{y}_{j,l} = \boldsymbol{h} v_{j,l} + \zeta_{j,l}$$

where $\boldsymbol{h} = \sqrt{P_l/M}(s_{l,\tau_1-\theta_l+1}, s_{l,\tau_2-\theta_l+1}, ..., s_{l,\tau_M-\theta_l+1})^{\mathrm{T}}$ and $\zeta_{j,l}$ is the vector of noise and interference from the other packets. Computing the minimum mean-squared error (MMSE) filtering of $\boldsymbol{y}_{j,l}$ to form an SNR-optimal linear of estimate of $v_{j,l}$ we find

$$w_{j,l} = (\boldsymbol{I} + \boldsymbol{h}^* \boldsymbol{F}_{j,l}^{-1} \boldsymbol{h})^{-1} \boldsymbol{h}^* \boldsymbol{F}_{j,l}^{-1} \boldsymbol{y}_{j,l}$$

where $\boldsymbol{F}_{j,l}$ denotes the covariance matrix of $\zeta_{j,l}$. One may compute the SNR of the signal $w_{j,l}$ as

$$\gamma_{j,l} = \boldsymbol{h}^* \boldsymbol{F}_{j,l}^{-1} \boldsymbol{h}.$$

In other words $w_{j,l}$ is a result of maximum ratio combining (MRC) of M received replicas of $v_{j,l}$. Since each bit $v_{j,l}$ can be 1 or -1, and takes each of the values with probability $1/2$, one can form a conditional expectation estimate $\hat{v}_{j,l}$ of $v_{j,l}$ as

$$\hat{v}_{j,l}^{(1)} = \mathbb{E}(v_{j,l}|w_{j,l}) = \tanh(w_{j,l}\gamma_{j,l}).$$

The estimates $\hat{v}_{j,l}^{(1)}$ of the data bits may then be used instead of the true bits $v_{j,l}$ unknown to the receiver to remodulate the packets. Since our estimation and interference cancellation process is performing message passing on a graph for each bit $v_{j,l}$ we compute M estimates $v_{j,l,m}^{(1)}$, $m = 1, 2, \cdots, M$. An estimate $\hat{v}_{j,l,m}^{(1)}$ is obtained exactly as $\hat{v}_{j,l}^{(1)}$ discussed above but without involving the observation y_{τ_m} in the estimation process (i.e. we start with vector $\boldsymbol{y}_{j,l,m}$ where the y_{τ_m} component is not present). We then produce a vector $\hat{\boldsymbol{v}}_l^{(1)}$ by stacking all the estimates $v_{j,l,m}^{(1)}$, re-modulate the packet and produce

$$\hat{\boldsymbol{x}}_l^{(1)} = \sqrt{\frac{P_l}{M}}\, \mathrm{diag}(\boldsymbol{d}, \boldsymbol{s}_l) \begin{pmatrix} \boldsymbol{I}_{MN_\mathrm{p}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{P}_l \end{pmatrix} \left[\boldsymbol{b}_l \boldsymbol{R}_{N_\mathrm{p}}^{\mathrm{T}}\, \hat{\boldsymbol{v}}_l^{(1)}\right]^{\mathrm{T}} .$$

At the next step we perform full interference cancellation to compute the residual vector of noise and interference

$$\boldsymbol{z}^{(1)} = \boldsymbol{y} - \boldsymbol{A}_{\mathcal{L}_{\text{pack}}(1)} \hat{\boldsymbol{x}}_{\mathcal{L}_{\text{pack}}(1)}^{(1)}$$

which is forwarded to the preamble detector at the beginning of the second iteration. We also compute vectors

$$\boldsymbol{y}_l^{(1)} = \boldsymbol{y} - \boldsymbol{A}_{\mathcal{L}_{\text{pack}}(1)\setminus l} \hat{\boldsymbol{x}}_{\mathcal{L}_{\text{pack}}(1)\setminus l}^{(1)} , \qquad l \in \mathcal{L}_{\text{pack}}(1)$$

that will be used for estimation of data of packets $l \in \mathcal{L}_{\text{pack}}(1)$ at the second iteration. At the beginning of the second iteration the packet detector produces an updated set of detected packet indices $\mathcal{L}_{\text{pack}}(2)$. For the packets newly detected at the second iteration, i.e. $l \in$

$\mathcal{L}_{\text{pack}}(2) \setminus \mathcal{L}_{\text{pack}}(1)$ the vector $\boldsymbol{z}^{(1)}$ will be used for data estimation. The data estimation stage of the second iteration computes $\hat{\boldsymbol{x}}_l^{(2)}$ which is then passed to the interference canceller. The entire process repeats for I iterations until all or almost all packets are detected and mutual inter-packet interference is reduced to (approximately) the noise power level. After that the packets are individually decoded by the error correction decoders. A detailed analysis of convergence for such a system is given in [55] where it is proved that as preamble length increases the system's performance approaches the capacity of the channel.

6.2 AMP Algorithm for Joint Decoding and Detection

We notice that at each iteration we need to compute and store M copies of each estimated packet bit and up to $|\mathcal{L}_{\text{pack}}(i)|$ different cancelled vectors. To reduce this computation burden we will use the AMP framework [16][58]. At each iteration we compute only one estimate for each data bit and each vector, and use correction factors to carefully approximate the original detection and decoding process.

The description of our proposed AMP algorithm based on series expansion and approximation of the passed messages is given in Fig. 6.5. The AMP version has a simple description and saves orders of magnitude in complexity and storage for algorithm implementation. Instead of storing the entire sparse matrix $\boldsymbol{A}_i$ matrix multiplications may be done via explicit modulation as in Fig. 6.1. In practical conditions with no chip synchronicity between the packets steps (6.6) and (6.7) can by done on sample basis.

6.3 Numerical Results

Fig. 6.6 demonstrates the simulated performance of the proposed random access system with a joint detector and decoder. We consider a spreading factor $M = 64$, $N_{\text{p}} = 15$ bit preamble, $N = 390$ bit payload, and set the average rate of the new false alarms per iteration to be $< 10^{-5}$. Equal power distribution of the packets' powers is considered [4] and the transmission is over the AWGN channel with signal-to-noise ratio (SNR) equal to $P\alpha/\sigma^2$. Binary error-correction codes optimal for the SNR observed at the output of the

[4] Unequal power distribution typically results in better iterative multi-user detector performances since power differences facilitate the convergence of the interference cancellation process. However, a modified sliding correlation detector needs to be considered in this case. For each time position detection at a number of threshold levels needs to be attempted before sliding forward.

Algorithm:

$$\text{Initialization:} \quad \boldsymbol{z}^{(0)} = \boldsymbol{y},\ \hat{\boldsymbol{x}}^{(1)} = \mathbf{0}$$

For $i = 1, 2, \cdots, I$

$$\begin{aligned}
\mathcal{L}_{\text{pack}}(i) &= \text{CorrelationDetect}(\boldsymbol{z}^{(i-1)}) && (6.6)\\
\boldsymbol{A}_i &= \boldsymbol{A}_{\mathcal{L}_{\text{pack}}(i)} \\
\boldsymbol{z}^{(i)} &= \boldsymbol{y} - \boldsymbol{A}_i \hat{\boldsymbol{x}}^{(i)} + (\boldsymbol{A}_i * \boldsymbol{A}_i)\, \boldsymbol{v}_i * \frac{\boldsymbol{z}^{(i-1)}}{\text{var}\,(\boldsymbol{z}^{(i-1)})} && (6.7)\\
\boldsymbol{w}_{i+1} &= \boldsymbol{A}_i^{\text{T}} \boldsymbol{z}^{(i)} + \hat{\boldsymbol{x}}^{(i)} * \left(\left(\boldsymbol{A}_i^{\text{T}} * \boldsymbol{A}_i^{\text{T}}\right) \frac{1}{\text{var}\,(\boldsymbol{z}^{(i)})} \right) \\
\hat{\boldsymbol{x}}^{(i+1)} &= \tanh(\boldsymbol{w}_{i+1}) \\
\boldsymbol{v}_{i+1} &= \text{sech}^2(\boldsymbol{w}_{i+1})
\end{aligned}$$

$*$ denotes component-wise multiplication.

Figure 6.5: AMP based joint decoding and detection algorithm. The above iterative process is followed by the individual error correction decoding of the packets.

joint decoder after the final iteration are assumed [5]. The rate R of the error-correction codes approaches 1 at high SNRs. The normalized traffic intensity is in the range $\alpha \in [0.84, 2.95]$ and the achievable throughput is computed for the maximum traffic intensity achievable for a given SNR as αR.

The achievable throughput is given in blue in Fig. 6.6, the theoretic throughput with perfect timing knowledge by the magenta stars curve, its asymptotic limit (as in [11]) by the magenta curve, the channel capacity by the red curve, and the throughput for separate detection and decoding (as in [55]) by the black curve. We notice that packet detection results in a moderate loss (about 2.2dB or smaller) compared to the results with perfect timing knowledge assumption.

The threshold G given by formula (6.6) needs to be adjusted to practical conditions to give specific false alarm probability. For this purpose we have multiplied G computed for the entire vector $\boldsymbol{z}^{(i)}$ by a single scaling factor ranging between 1.4 and 2.6 depending on the load. In addition, the preambles of the detected packets are removed upon detection to reduce the inter-packet interference. The rate of the false alarms that will inevitably occur during the detection process is kept low (suggested rate for VDES is 10^{-4} [18]). We do not consider explicit processing of the false alarms in this work, however, the false alarms may be detected by the decoding the preamble codes or noticing no improvement of the iterative

[5] We have not considered the impact of finite-length capacity.

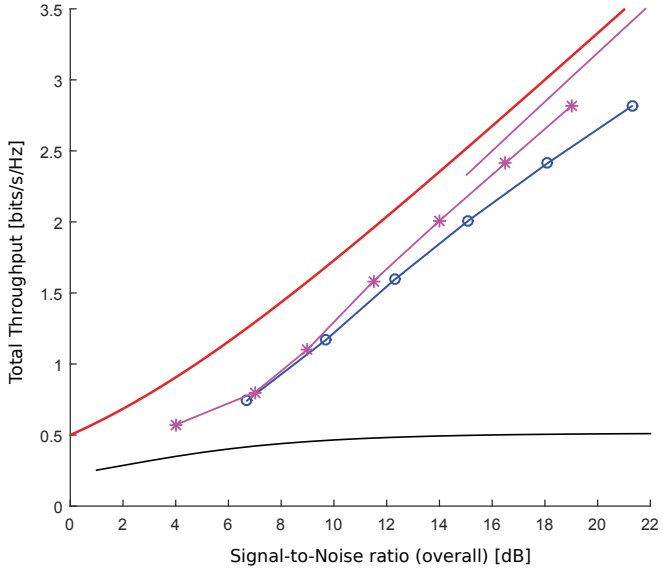

Figure 6.6: The throughput achievable by the proposed random access system in bit/s/Hz as a function of the total system's SNR.

data estimation of a non-existing packet.

Comparing the results with other random access techniques we notice that the achieved traffic load of 2.95 of packets per packet length per spreading factor (bandwidth) is much higher than 1 packets per slot or (packet length for unslotted systems) which is a typical maximum for classic random access systems such as ALOHA [59], carrier-sense multiple access (CSMA), self-organized TDMA used in AIS, contention resolution diversity slotted ALOHA (CRDSA) [60], irregular resolution slotted ALOHA (IRSA) [61], coupled random access [62], and even recently proposed systems such as coded slotted ALOHA and enhanced contention resolution ALOHA (ECRA) [63] that can achieve loads up to 2 packets/packet length.

Chapter 7

Conclusions

In this thesis we have considered an implementation of the coupling information transmission multi-user communication technique based on the spatial graph coupling principle [3]. We have proposed a demodulation technique based on approximate message passing (AMP) proposed by Donoho et al. [16] for compressed sensing problems. A version of the AMP is derived for the multi-user situation to simplify the reception algorithm from the perspective of implementation layout and computational complexity. At the same time we make sure that operation in the "near-capacity" regime for a large number of multi-user loads is preserved. The demodulation complexity per bit is constant per iteration and does not scale with the processing gain, contrary to existing algorithms [2, 11, 12].

The algorithm, which is derived for the case of a theoretic scenario with perfectly organized message arrivals is the extended to a random-access case. An application of a satellite uplink communication for maritime data exchange systems is considered and we demonstrate that the achievable performance is superior that that of competing random access mechanisms such as variations on ALOHA, ECRA, and uncoupled CDMA.

7.1 Future Work

It has been shown that coupling information transmission in conjunction with the derived approximate message passing is an effective tool for communication over the AWGN multiple access channel. There are several directions one could consider for future work:

- **Unequal Power Case :** While the equal power case is more difficult for inter-symbol interference cancellation, in a realistic scenarion multiple user will always have power differences. It is, for example, likely that users will transmit with different power levels due to differing distances from the receiver. While, the algorithm in its present form can accept unequal power messages it would be helpful to examine the effectiveness of the AMP algorithm for the unequal power cases.

- **Alternative Channels :** We have considered the AWGN channel, however satellite communication can be complicated by fading due to the ionosphere or troposphere, possible multi-path propagation, and changes of phase in the signal due to the Doppler effect. It is worthwhile to consider how the AMP algorithm would be modified to handle this situation, such as the inclusion of a bank of correlation detectors and adjustment of the transmitter frequency based on approximate satellite location broadcasted every frame via a bulletin board on the downlink channel.

Bibliography

[1] L. Pérez C. Schlegel. Trellis and turbo coding, 2nd edition. 2004.

[2] Dmitri Truhachev and Christian Schlegel. Coupling data transmission for capacity-achieving multiple-access communications. *arXiv preprint arXiv:1209.5785*, 2012.

[3] Christian Schlegel and Dmitry Truhachev. Multiple access demodulation in the lifted signal graph with spatial coupling. *Information Theory, IEEE Transactions on*, 59(4): 2459–2470, 2013.

[4] Alberto Jimenez Felström and Kamil Sh Zigangirov. Time-varying periodic convolutional codes with low-density parity-check matrix. *Information Theory, IEEE Transactions on*, 45(6):2181–2191, 1999.

[5] Michael Lentmaier, Arvind Sridharan, Daniel J Costello Jr, and Kamil Zigangirov. Iterative decoding threshold analysis for ldpc convolutional codes. *IEEE Transactions on Information Theory*, 56(10):5274–5289, 2010.

[6] Shrinivas Kudekar, Thomas J Richardson, and Rüdiger L Urbanke. Threshold saturation via spatial coupling: Why convolutional ldpc ensembles perform so well over the bec. *Information Theory, IEEE Transactions on*, 57(2):803–834, 2011.

[7] Toshiyuki Tanaka. A statistical-mechanics approach to large-system analysis of cdma multiuser detectors. *Information Theory, IEEE Transactions on*, 48(11):2888–2910, 2002.

[8] Andrea Montanari and David Tse. Analysis of belief propagation for non-linear problems: The example of cdma (or: How to prove tanaka's formula). In *Information Theory Workshop, 2006. ITW'06 Punta del Este. IEEE*, pages 160–164. IEEE, 2006.

[9] Christian Schlegel. Cdma with partitioned spreading. *Communications Letters, IEEE*, 11(12):913–915, 2007.

[10] Dmitri Truhachev, Christian Schlegel, and Lukasz Krzymien. Low-complexity capacity achieving two-stage demodulation/decoding for random matrix channels. In *Information Theory Workshop, 2007. ITW'07. IEEE*, pages 584–589. IEEE, 2007.

[11] Dmitry Truhachev. Universal multiple access via spatially coupling data transmission. In *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on*, pages 1884–1888. IEEE, 2013.

[12] Keigo Takeuchi, Toshiyuki Tanaka, and Tsutomu Kawabata. Performance improvement of iterative multiuser detection for large sparsely spread cdma systems by spatial coupling. *Information Theory, IEEE Transactions on*, 61(4):1768–1794, 2015.

[13] Christian Schlegel, Marat Burnashev, and Dmitri Truhachev. Generalized superposition modulation and iterative demodulation: A capacity investigation. *Journal of Electrical and Computer Engineering*, 2010:1, 2010.

[14] Li Ping, Lihai Liu, KY Wu, and Wai Kong Leung. On interleave-division multiple-access. In *Communications, 2004 IEEE International Conference on*, volume 5, pages 2869–2873. IEEE, 2004.

[15] Peter Adam Hoeher and Tianbin Wo. Superposition modulation: myths and facts. *Communications Magazine, IEEE*, 49(12):110–116, 2011.

[16] David L Donoho, Arian Maleki, and Andrea Montanari. Message-passing algorithms for compressed sensing. *Proceedings of the National Academy of Sciences*, 106(45): 18914–18919, 2009.

[17] Stefan Bober. Vdes - vhf data exchange system overview. In *2nd Workshop on VDES*.

[18] Itu-r working party 5b . http://www.itu.int/en/ITU-R/study-groups/rsg5/rwp5b/Pages/default.aspx. Accessed: 2016-03-30.

[19] *IALA Recommendation for ITU-R M VDES*.

[20] IEEE. Ieee standard, section 20.3.11.6 "802.11n-2009". http://standards.ieee.org/getieee802/download/802.11n-2009.pdf. Accessed: 2016-05-24.

[21] Daniel J Costello and G David Forney. Channel coding: The road to channel capacity. *Proceedings of the IEEE*, 95(6):1150–1177, 2007.

[22] Shrinivas Kudekar, Tom Richardson, and Rudiger L Urbanke. Spatially coupled ensembles universally achieve capacity under belief propagation. *Information Theory, IEEE Transactions on*, 59(12):7761–7813, 2013.

[23] Daniel J Costello, Lara Dolecek, Thomas Fuja, Joerg Kliewer, David Mitchell, and Roxana Smarandache. Spatially coupled sparse codes on graphs: Theory and practice. *Communications Magazine, IEEE*, 52(7):168–176, 2014.

[24] Keigo Takeuchi, Toshiyuki Tanaka, and Tsutomu Kawabata. A phenomenological study on threshold improvement via spatial coupling. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 95(5):974–977, 2012.

[25] Arvind Yedla, Yung-Yih Jian, Phong S Nguyen, and Henry D Pfister. A simple proof of threshold saturation for coupled scalar recursions. In *Turbo Codes and Iterative Information Processing (ISTC), 2012 7th International Symposium on*, pages 51–55. IEEE, 2012.

[26] Arvind Yedla, Yung-Yih Jian, Phong S Nguyen, and Henry D Pfister. A simple proof of threshold saturation for coupled vector recursions. In *Information Theory Workshop (ITW), 2012 IEEE*, pages 25–29. IEEE, 2012.

[27] Wassim M Haddad and VijaySekhar Chellaboina. *Nonlinear dynamical systems and control: a Lyapunov-based approach*. Princeton University Press, 2008.

[28] Joel A Tropp. Just relax: Convex programming methods for identifying sparse signals in noise. *Information Theory, IEEE Transactions on*, 52(3):1030–1051, 2006.

[29] David L Donoho, Michael Elad, and Vladimir N Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *Information Theory, IEEE Transactions on*, 52(1):6–18, 2006.

[30] Emmanuel J Candes, Justin K Romberg, and Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on pure and applied mathematics*, 59(8):1207–1223, 2006.

[31] Emmanuel J Candes and Terence Tao. Decoding by linear programming. *Information Theory, IEEE Transactions on*, 51(12):4203–4215, 2005.

[32] David L Donoho. Compressed sensing. *Information Theory, IEEE Transactions on*, 52(4):1289–1306, 2006.

[33] Emmanuel Candes and Terence Tao. The dantzig selector: statistical estimation when p is much larger than n. *The Annals of Statistics*, pages 2313–2351, 2007.

[34] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *Image Processing, IEEE Transactions on*, 15 (12):3736–3745, 2006.

[35] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *Image Processing, IEEE Transactions on*, 16(8):2080–2095, 2007.

[36] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image restoration by sparse 3d transform-domain collaborative filtering. In *Electronic Imaging 2008*, pages 681207–681207. International Society for Optics and Photonics, 2008.

[37] Matan Protter and Michael Elad. Image sequence denoising via sparse and redundant representations. *Image Processing, IEEE Transactions on*, 18(1):27–35, 2009.

[38] Michael Elad, J-L Starck, Philippe Querre, and David L Donoho. Simultaneous cartoon and texture image inpainting using morphological component analysis (mca). *Applied and Computational Harmonic Analysis*, 19(3):340–358, 2005.

[39] David L Donoho and Xiaoming Huo. Uncertainty principles and ideal atomic decomposition. *Information Theory, IEEE Transactions on*, 47(7):2845–2862, 2001.

[40] Yagyensh Chandra Pati, Ramin Rezaiifar, and PS Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pages 40–44. IEEE, 1993.

[41] Shuchin Aeron, Venkatesh Saligrama, and Manqi Zhao. Information theoretic bounds for compressed sensing. *Information Theory, IEEE Transactions on*, 56(10):5111–5130, 2010.

[42] Galen Reeves and Michael Gastpar. Compressed sensing phase transitions: Rigorous bounds versus replica predictions. In *Information Sciences and Systems (CISS), 2012 46th Annual Conference on*, pages 1–6. IEEE, 2012.

[43] Yihong Wu and Sergio Verdú. Rényi information dimension: Fundamental limits of almost lossless analog compression. *Information Theory, IEEE Transactions on*, 56 (8):3721–3748, 2010.

[44] Yihong Wu and Sergio Verdú. Optimal phase transitions in compressed sensing. *Information Theory, IEEE Transactions on*, 58(10):6241–6263, 2012.

[45] Dongning Guo, Dror Baron, and Shlomo Shamai. A single-letter characterization of optimal noisy compressed sensing. In *Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on*, pages 52–59. IEEE, 2009.

[46] David L Donoho, Adel Javanmard, and Alessandro Montanari. Information-theoretically optimal compressed sensing via spatial coupling and approximate message passing. *Information Theory, IEEE Transactions on*, 59(11):7434–7464, 2013.

[47] Shrinivas Kudekar and Henry D Pfister. The effect of spatial coupling on compressive sensing. In *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on*, pages 347–353. IEEE, 2010.

[48] Florent Krzakala, Marc Mézard, François Sausset, YF Sun, and Lenka Zdeborová. Statistical-physics-based reconstruction in compressed sensing. *Physical Review X*, 2 (2):021005, 2012.

[49] Yonina C Eldar and Gitta Kutyniok. *Compressed sensing: theory and applications*. Cambridge University Press, 2012.

[50] Upamanyu Madhow and Michael B Pursley. Acquisition in direct-sequence spread-spectrum communication networks: an asymptotic analysis. *Information Theory, IEEE Transactions on*, 39(3):903–912, 1993.

[51] Giovanni E Corazza and Vittorio Degli-Esposti. Acquisition-based capacity estimates for cdma with imperfect power control. In *Spread Spectrum Techniques and Applications, 1994. IEEE ISSSTA'94., IEEE Third International Symposium on*, pages 325–329. IEEE, 1994.

[52] Mark C Reed, Leif W Hanlen, and Giovanni E Corazza. Return-link code acquisition for 1-d and 2-d with ds-cdma for high-capacity multiuser systems. *Vehicular Technology, IEEE Transactions on*, 57(1):324–334, 2008.

[53] Bathiya Senanayake, Mark C Reed, and Zhenning Shi. Timing acquisition for multiuser idma. In *Communications, 2008. ICC'08. IEEE International Conference on*, pages 5077–5081. IEEE, 2008.

[54] Dmitri Truhachev. Asymptotic analysis of joint timing acquisition and multiple packet reception. In *Wireless Communications and Networking Conference (WCNC), 2011 IEEE*, pages 1869–1874. IEEE, 2011.

[55] Dmitry Truhachev. Packet acquisition for spatially coupling information transmission. In *Communications Workshops (ICC), 2014 IEEE International Conference on*, pages 495–500. IEEE, 2014.

[56] Dmitri Truhachev. Achieving awgn multiple access channel capacity with spatial graph coupling. *Communications Letters, IEEE*, 16(5):585–588, 2012.

[57] Burkhard Militzer, Michele Zamparelli, and Dieter Beule. Evolutionary search for low autocorrelated binary sequences. *IEEE Transactions on Evolutionary Computation*, 2(1):34–39, 1998.

[58] Alfred M Bruckstein, David L Donoho, and Michael Elad. From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM review*, 51(1): 34–81, 2009.

[59] Norman Abramson. The aloha system: another alternative for computer communications. In *Proceedings of the November 17-19, 1970, fall joint computer conference*, pages 281–285. ACM, 1970.

[60] Enrico Casini, Riccardo De Gaudenzi, and Od R Herrero. Contention resolution diversity slotted aloha (crdsa): An enhanced random access schemefor satellite access packet networks. *Wireless Communications, IEEE Transactions on*, 6(4):1408–1419, 2007.

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

[62] Gianluigi Liva, Enrico Paolini, Michael Lentmaier, and Marco Chiani. Spatially-coupled random access on graphs. In *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*, pages 478–482. IEEE, 2012.

[63] Federico Clazzer and Christian Kissling. Enhanced contention resolution aloha-ecra. In *Systems, Communication and Coding (SCC), Proceedings of 2013 9th International ITG Conference on*, pages 1–6. VDE, 2013.