Advances in Iterative Probabilistic Processing for Communication Receivers

Daniel Joseph Jakubisin

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

R. Michael Buehrer, Chair
Claudio R. C. M. da Silva
Dhruv Batra
Inyoung Kim
Robert W. McGwier

April 22, 2016
Blacksburg, Virginia

Keywords: Iterative Receivers, Factor Graphs, Belief Propagation

Advances in Iterative Probabilistic Processing for Communication Receivers

Daniel J. Jakubisin

(ABSTRACT)

As wireless communication systems continue to push the limits of energy and spectral efficiency, increased demands are placed on the capabilities of the receiver. At the same time, the computational resources available for processing received signals will continue to grow. This opens the door for iterative algorithms to play an increasing role in the next generation of communication receivers.

In the context of receivers, the goal of iterative probabilistic processing is to approximate maximum *a posteriori* (MAP) symbol-by-symbol detection of the information bits and estimation of the unknown channel or signal parameters. The sum-product algorithm is capable of efficiently approximating the marginal posterior probabilities desired for MAP detection and provides a unifying framework for the development of iterative receiver algorithms. However, in some applications the sum-product algorithm is computationally infeasible. Specifically, this is the case when both continuous and discrete parameters are present within the model. Also, the complexity of the sum-product algorithm is exponential in the number of variables connected to a particular factor node and can be prohibitive in multi-user and multi-antenna applications.

In this dissertation we identify three key problems which can benefit from iterative probabilistic processing, but for which the sum-product algorithm is too complex. They are (1) joint synchronization and detection in multipath channels with emphasis on frame timing, (2) detection in co-channel interference and non-Gaussian noise, and (3) joint channel estimation and multi-signal detection. This dissertation presents the advances we have made in iterative probabilistic processing in order to tackle these problems. The motivation behind the work is to (a) compromise as little as possible on the performance that is achieved while limiting the computational complexity and (b) maintain good theoretical justification to the algorithms that are developed.

Dedication

To my family,

You bring me so much joy! I look forward to the many adventures we will share.

Acknowledgments

Seven years ago, my wife and I made a house hunting trip to Blacksburg, VA to find a home that we could enjoy for two years while I pursued a masters degree. We quickly realized that the home we chose was much larger than we needed. We never imagined how well we would use it! Now, three kids and one dog later we look back on many memorable days and good friendships. There are many people that have supported me during this time and I hope to convey my gratitude through this acknowledgment.

I am grateful to the professors who have shaped my education at Virginia Tech. I would like to acknowledge my advisor Dr. R. Michael Buehrer. He has been supportive, provided discerning guidance, and been an example to me throughout my Ph.D. study. I am grateful for the insight and guidance of my committee: Dr. Buehrer, Dr. da Silva, Dr. McGwier, Dr. Batra, and Dr. Kim. And for the dedication of the professors from whom I have learned in the classroom: Dr. da Silva, Dr. Ellingson, Dr. Pratt, Dr. Clancy, Dr. Buehrer, Dr. Batra, Dr. House, and Dr. Fraticelli.

I have enjoyed the friendship of many students during my time here (there are too many to name, but I will try): Gautham Chavali, Chris Headley, Yue Yan, Aditya Mundle, Chris Phelps, Benton Thompson, Corey Cooke, Javier Schloemann, Hunter DeJarnette, Dhiraj Amuru, Mahi Abdelbar, Zach Biskaduros, Kevin McDermott, Matt Carrick, Jeff Poston, Ali Gaber, Chris O'Lone, and Thaddeus Czauski. I have relied on four people in particular for discussions of research: Dhiraj Amuru, Javier Schloemann, Chris Phelps, and Matt Carrick. I am thankful to Dhiraj for countless hours of discussion, brainstorming, and proof-reading. He has been my go-to source for advice in the face of many challenges. I am thankful for Javier Shloemann for his encouragement and prayer. A special thanks to Hilda Reynolds and Nancy Goad.

I am thankful for the support and encouragement of my family. My wife and my children have been constantly supportive and encouraging even though this journey has been weary at times. To my parents, brothers, and sisters, (including in-laws!): thank you for your help and encouragement. To the many friends from our church and our homegroup who have blessed our family and prayed for us, thank you.

I am grateful to Jesus Christ for giving up his life so that I might have life in him. Everything that I accomplish is through him.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Advances in wireless communication systems continue to push the limits of both energy efficiency and spectral efficiency. Today's systems already employ capacity-approaching error correction codes, multiple-input multiple-output (MIMO) transmission, higher order modulations, and orthogonal frequency-division multiplexing (OFDM). Requirements for next generation cellular systems (5G) have been defined which seek further improvements in terms of spectral efficiency, peak data rate, mobility, and more [1]. Continued network densification is expected with the introduction of heterogeneous networks [2, 3] and spectrum sharing [4]. In addition, wireless system architecture is expected to transition from a "cell-centric" model in which interference is centrally managed by the base station to a "device-centric" model [3]. For these reasons advanced receiver algorithms are vitally important to current and future wireless systems. Moreover advanced algorithms will be feasible as the computational resources available to receivers increase through technologies such as Cloud RAN [5].

Iterative algorithms are a viable option for the challenging estimation and detection tasks that face next generation communications receivers. Iterative probabilistic processing for communications receivers originates from the popular concept of turbo decoding discovered in the early 1990s [6]. The concept of an *iterative receiver* was developed in which probabilistic information about the unknown data and parameters is passed between the blocks of the receiver (e.g., synchronization, equalization, demodulation, and decoding) in an iterative fashion. From a theoretical standpoint the goal is to approximate maximum *a posteriori* (MAP) symbol-by-symbol detection of the information bits and estimation of the unknown channel or signal parameters.

The sum-product algorithm (SPA) has been shown to be a generalization of turbo decoding and provides a unifying framework for the development of iterative receiver algorithms [7, 8]. The SPA operates by passing messages along the edges of a factor graph model of the underlying joining probability distribution of the received signal. It has been shown to be equivalent to belief

propagation (BP) and is capable of efficiently computing marginal posterior probabilities in the case of tree structured graphs (and approximate marginal posteriors in the case of graphs with cycles). However, in some applications the SPA is computationally infeasible. Specifically, this is the case when both continuous and discrete parameters are present within the model. Complexity is exponential in the number of variables connected to a particular factor node and can be prohibitive in multi-user and multi-antenna applications.

In the next section, the contributions of this dissertation are summarized. In doing so, several problems are identified in which communications receivers can benefit greatly from iterative probabilistic processing, but for which the SPA is too complex. This dissertation presents the advances we have made in iterative probabilistic processing in order to tackle these problems. The motivation behind the work has been to (a) compromise as little as possible on the performance that is achieved while limiting the computational complexity and (b) maintain good theoretical justification to the algorithms that are developed.

1.2 Contributions

This dissertation advances the existing body of work on probabilistic processing for communications receivers with the following four major contributions:

- Chapter 3: Development of an iterative receiver to incorporate code-aided frame synchronization algorithm and a frame pre-processing stage for multipath channels.
- Chapter 4: Development of an approximate MAP algorithm for detection of co-channel signals in non-Gaussian noise.
- Chapter 5: Developement of a probabilistic receiver architecture combining BP, MF, and EP for multi-signal detection
- Chapter 6: Study of the value of iterative parameter estimation

In the following sections the contributions of this work are introduced.

1.2.1 Iterative Receiver Design and Frame Synchronization

Frame synchronization refers to the acquisition of the start time of a frame of transmitted data. Accurate synchronization is particularly important for block codes such as turbo codes and low-density parity-check (LDPC) codes where the information of the whole frame is lost if the frame start time is incorrectly estimated. Frame synchronization is traditionally accomplished by adding a sync word as a preamble to the frame and correlating the received signal with this known sequence [9]. The complexity of the correlation technique is low, but it is sub-optimal in terms of frame

synchronization error probability. The low SNR requirements of advanced error correction codes motivates the development of code-aided frame synchronization techniques. However, several challenges are faced.

MAP frame synchronization for *coded data* requires marginalization over the entire set of codewords. Herzet and Vandendorpe recognized that the sum-product algorithm is capable of efficiently performing the desired marginalization over the coded data when the factor graph of the joint probability distribution is acyclic [10]. In general, however, marginalization over the entire set of codewords is too complex and thus a number of code-aided frame synchronization methods have been proposed in the literature. Code-aided frame synchronization is performed in [11, 12] by computing the syndrome posterior probability (SPP) of the code's parity check relations and in [13] based on the expectation-maximization (EM) algorithm. However, previous work on code-aided frame synchronization predominately assumes an additive white Gaussian noise (AWGN) channel. Furthermore, code-aided frame synchronization techniques require processing each potential frame offset individually and the processing requires decoding or a decoding-like operation. Therefore, the complexity of the receiver as a whole scales linearly with the number of frame offsets considered by the code-aided synchronizer. Robertson proposed a two-stage frame synchronizer, referred to as list synchronization, in which a low-complexity decision rule is used to obtain a list of the most likely frame offsets and then this list is processed by a high-complexity decision rule [14]. However, parameters of the list synchronization method (for instance, the size of the list) are determined through simulation. What is lacking with the list synchronization method is the ability to choose the optimal number of candidate frame offsets to be considered by the high complexity decision rule. Finally, knowledge of the channel, noise distribution, and other synchronization parameters are required for code-aided methods.

In Chapter 3, the problem of frame synchronization is considered for coded signals in *multipath channels*. We consider the case in which the channel is not known *a priori*, and develop an iterative receiver structure for frame synchronization, channel estimation, equalization, demodulation, and decoding. Two code-aided frame synchronization methods are developed for multipath channels based on the SPP and EM metrics. In contrast to Robertson's list synchronizer, the pre-processing stage uses a Bayesian technique to select, on a frame-by-frame basis, the subset of frame offsets to be processed by a code-aided frame synchronizer. The proposed technique selects the minimum number of offsets for a given target performance by computing the highest posterior density (HPD) region of the frame distribution. Numerical results are provided to characterize the expected number of frame offsets in the HPD region because the computational complexity of code-aided frame synchronization is proportional to the number of frame offsets which must be processed.

In Chapter 3, we also explore the complexity-performance trade-off of code-aided frame synchronization in multipath channels. Regions of signal-to-noise ratio (SNR) are identified for which code-aided frame synchronization provides an improvement in performance while maintaining a complexity within a factor of $2\times$ that of conventional techniques. The performance of the receiver is demonstrated in a scenario in which the proposed receiver achieves a gain of up to 3 dB with a complexity increase of only 20%. We also show that code-aided frame synchronization may be reliably performed at SNR values below that required for decoding. This is important, for ex-

ample, to support acknowledgment (ACK)/non-acknowledgment (NACK) and hybrid automatic repeat request (hybrid ARQ) protocols or to achieve reliable synchronization in a collaborative communications scenario where multiple receivers may cooperate to decode a signal at low SNR.

In summary, the contributions of this work are as follows:

- Development of code-aided frame synchronization for multipath channels
- Development of a novel frame pre-processing stage to allow a trade-off between performance and complexity
- Design of an iterative receiver for frame synchronization, channel estimation, equalization, demodulation, and decoding
- Analysis of the complexity-performance trade-off in order to identify regions where code-aided frame synchronization is beneficial
- Analysis of code-aided frame synchronization performance at SNR values below that required for decoding.

1.2.2 Detection of Co-channel Signals in Non-Gaussian Noise

Co-channel interference is a key factor limiting the capacity of densely deployed wireless communication systems. As stated before, continued network densification is expected with the introduction of heterogeneous networks [2, 3] and spectrum sharing [4]; and wireless system architecture is expected to transition from a "cell-centric" model in which interference is centrally managed by the base station to a "device-centric" model [3]. For these reasons, the ability to detect a desired signal in the presence of co-channel interference is vitally important for current and future wireless systems.

Joint maximum *a posteriori* probability (JMAP) symbol-by-symbol detection of the desired and co-channel signals is optimal with respect to minimizing the symbol error probability. However, it is prohibitively complex. Factor graphs and the sum-product algorithm (SPA) provide a theoretically justified approach to approximating the marginal posterior distributions required for JMAP detection [7, 8, 15]. Complexity is greatly reduced by passing local distributions (messages) along the edges of the factor graph. The factor graph approach generalizes and provides justification for algorithms which combine soft detection with single-user soft decoding. Yet, even within the factor graph model the receiver may be prohibitively complex as a result of high-order modulations, numerous users, or inter-symbol interference (ISI).

A particularly challenging case is signal detection using a single receive antenna in the presence of both ISI and co-channel interference. The received signals do not possess a natural separability due to a multiple access technique or the use of multiple antennas. As a result, linear filtering and interference cancellation are ineffective, especially when the signal power levels are similar.

Joint MAP detection in such a signal model is developed and studied in [16], where the detector accounts for the strongest ISI components (ignoring the weaker terms) and separates detection from single-user soft decoding. Even then, the algorithm is exponentially complex in the number of co-channel signals, the considered ISI components, and the modulation order and as a result simulations in [16] are limited to the detection of two users with BPSK modulation.

Moreover, a number of measurement studies report that the noise affecting wireless communication systems is often non-Gaussian—exhibiting an impulsive behavior [17–24]. Non-Gaussian noise may also be used to model co-channel signals with transmission properties which are unknown to the receiver and thus not part of the joint detection [25, 26].

In Chapter 4, we consider detection in co-channel interference in the general case of ISI (similarly, inter-carrier interference in an OFDM system) and noise which is non-Gaussian. We propose a new approximate joint MAP (AMAP) detector which is motivated by a factor graph model of the received signal. The AMAP algorithm approximates the SPA messages by maintaining a discrete domain for the multipath components with the strongest received power and approximating the remaining weaker multipath components with a Gaussian distribution. The motivation behind the AMAP algorithm is to approximate the JMAP detector as closely as possible within the allowable computational complexity of the receiver. The complexity of the algorithm is adjustable and can be set to account for the capabilities of the receiver, the desired performance, or the difficulty of the detection task. Noise is modeled with a Gaussian mixture distribution which is a common model for impulsive noise [27–30]. We also develop an iterative estimation stage based on the EM algorithm [31] to incorporate estimation of the channel coefficients and the Gaussian mixture model (GMM) parameters within the iterative receiver.

The proposed receiver improves upon existing algorithms for detecting co-channel signals in the presence of ISI while also introducing the ability to handle non-Gaussian noise. Applicable detection algorithms in the literature generally assume known channel parameters or perform channel estimation assuming Gaussian noise. In this work, we remove these assumptions and develop an iterative receiver which provides the following contributions:

- advances the state of the art by improving performance with lower complexity,
- operates effectively in the presence of Gaussian and non-Gaussian noise, and
- incorporates iterative channel and noise parameter estimation.

1.2.3 Probabilistic Receiver Architecture for Multi-Signal Detection

In Chapter 5, we consider the development of a probabilistic receiver architecture for joint channel estimation and multi-signal detection. While Chapter 4 is focused on a co-channel detection problem in which only the discrete (digital) signal constellation and the code structure may be exploited for signal separability, Chapter 5 considers a more general case of multi-signal detection

where multiple antennas or a multiple access scheme potentially aide signal separation. Instead, the focus of this work is on developing a combined message passing algorithm applicable to both estimation and detection which passes probabilistic messages throughout the receiver.

In the Bayesian framework, the data (e.g., symbols) and the parameters (e.g., channel coefficients) of the received signal are modeled as random variables. This results in a non-linear observation model which, along with the fact that the model variables are both continuous and discrete, makes exact application of belief propagation (BP) (i.e., the sum-product algorithm) infeasible. Recognizing that BP and the MF approximation have complementary strengths, receiver algorithms have been developed which combine these algorithms [32–34]. Riegler *et al*. provided a theoretical justification for the combined BP-MF message passing framework in [35] which is analogous to that given for BP by Yedidia *et al*. in [15]. In the context of communications receivers, BP is applied to demodulation and decoding and the MF approximation is applied to the non-linear observation model and estimation of parameters such as coefficients of a wireless multipath channel.

Combined BP-MF message passing has been applied to multi-user code-division multiple access (CDMA) [32], multiple-input multiple-output (MIMO) systems [33, 34, 36, 37], and co-channel interference [38].[1] The standard implementation of the BP-MF framework in the multi-signal case places the equalization or multi-user detection task within the MF subgraph. Specifically, in [34], the BP-MF framework was applied to MIMO-OFDM where multi-stream equalization is within the MF subgraph, but the algorithm relies upon a generous initialization point.

In Chapter 5, we show that the MF approximation when applied to detection of multiple signals results in an interference cancellation structure which degrades performance. We develop a new factor graph construction for application of the BP-MF framework to problems involving the detection of multiple signals. In the proposed construction the joint distribution is factored into distinct observation factors and equalization factors which allow BP to be applied to the equalization function (i.e., multi-signal detection), thereby improving the detection capability of the structure. We also develop a low-complexity variant to the proposed construction in which Gaussian BP is applied to the equalization factors. In this case, the factor graph of the joint probability distribution is divided into three subgraphs: (i) a MF subgraph comprised of the observation factors and channel estimation, (ii) a Gaussian BP subgraph which is applied to multi-signal detection, and (iii) a discrete BP subgraph which is applied to demodulation and decoding. This requires that Gaussian approximations of the discrete messages passed from the discrete BP subgraph to the Gaussian BP subgraph be computed. A common approach in the technical literature is to match the mean and variance of the approximating Gaussian distribution with those of the discrete extrinsic or posterior distribution [39–44]. However, expectation propagation (EP) [45, 46] provides a theoretically justified approach to computing Gaussian approximations which has been shown to outperform extrinsic and posterior approximations [47, 48]. Thus, EP is used to approximate discrete distributions with a Gaussian distribution and links the discrete BP and Gaussian BP subgraphs. The result

[1] Some of these works pre-date [35] and have used the terms variational message passing / sum-product algorithm (VMP-SP) and divergence minimization (DM) to refer to algorithms that similarly combine BP and the MF approximation.

is a probabilistic receiver architecture with strong theoretical justification which can be applied to multi-signal detection and, in general, detection in the presence of interference.

Our proposed receiver architecture is suitable for multi-signal detection in a variety of scenarios including co-channel interference (CCI), MIMO, multi-user MIMO, and non-orthogonal multiple access (NOMA). Reasonable complexity is maintained through the use of the MF approximation for the observation factors and Gaussian BP for the equalization factors. Parameter estimation may be included in the MF subgraph or, using MF as a link across the non-linear observation model, other algorithms such as BP or generalized approximate message passing (GAMP) [49] may be applied to estimation [50].

The contributions of this work are summarized as follows:

- development of a factor graph construction for applying BP-MF to multi-signal detection,
- development of a low-complexity variant of the proposed construction combining BP (both Gaussian and discrete), the MF approximation, and EP, and
- derivation of a new MF-based time-domain channel estimator for OFDM signals.

1.2.4 Value of Iterative Parameter Estimation

Iterative receiver algorithms for parameter estimation have been shown to achieve the Cramér-Rao lower bound (CRLB) for the mean square error (MSE) of frequency, phase, and symbol timing estimation of *coded* signals [51]. While the CRLB provides the best estimation performance that the iterative receiver can achieve (given the classical estimation framework), the desired insight in our work is to understand the effect of parameter knowledge on data detection. Iterative parameter estimation has been shown to achieve a bit error rate (BER) and frame error rate (FER) very close to a receiver with perfect estimation. Thus, there are clearly cases in which the iterative approach outperforms traditional parameter estimation. However, in other cases traditional parameter estimation may be sufficient. The current body of work lacks a framework for determining when iterative parameter estimation is valuable, especially for making online decisions about whether or not to perform further iterations. In order to gain insight into the value of iterative parameter estimation, both performance and complexity must be viewed in the context of the performance and complexity of a receiver which relies on traditional parameter estimation (e.g., training-based estimation).

In the literature, EXIT charts provide insight into the convergence of probabilistic information exchanged between two blocks of the receiver (decoders, demodulators, equalizers, estimators, etc.) [52]. When the EXIT chart is constructed between a channel estimation block and the decoding block of the receiver, as done in [53, 54], it provides a tool for computing the expected performance gains of re-estimating the channel in each iteration. However, it is a simulation-based approach which is not able to inform online decisions.

In Chapter 6, we develop a theoretical framework for evaluating the value of parameter knowledge which can provide feedback about the need to re-estimate parameters in real-time. A very influential work by Yedidia *et al.* [55] made a connection between the sum-product algorithm and the Bethe free energy. Specifically, it was discovered that the fixed points of the sum-product algorithm correspond to minima of the Bethe free energy [15, 56]. The Bethe free energy is a measure of divergence which we use to evaluate the quality of our parameter knowledge with respect to data detection.

1.3 Relevant Publications

This dissertation is based on the following publications:

- D. J. Jakubisin, R. M. Buehrer, and C. R. C. M. da Silva, "Probabilistic receiver architecture combining BP, MF, and EP for multi-signal detection," *IEEE Trans. Commun., submitted for publication*. [57]
- D. J. Jakubisin and R. M. Buehrer, "Approximate joint MAP detection of co-channel signals in non-Gaussian noise," *IEEE Trans. Commun., submitted for publication*. [44]
- D. J. Jakubisin and R. M. Buehrer, "Performance, complexity, and receiver design for code-aided frame synchronization in multipath channels," *IEEE Trans. Commun.*, vol. 63, no. 9, pp. 3363–3376, Sep. 2015. [58]
- D. J. Jakubisin, R. M. Buehrer, and C. R. C. M. da Silva, "BP, MF, and EP for joint channel estimation and detection of MIMO-OFDM signals," in *Proc. IEEE GLOBECOM, submitted for publication*, 2016. [59]
- D. J. Jakubisin and R. M. Buehrer, "Approximate joint MAP detection of co-channel signals," in *Proc. IEEE MILCOM*, Tampa, FL, Oct. 2015, pp. 1530–1535. [43]
- D. Jakubisin and R. M. Buehrer, "On the complexity-performance trade-off in code-aided frame synchronization," in *Proc. 15th IEEE Int. Workshop SPAWC*, Toronto, ON, Canada, Jun. 2014, pp. 364–368. [60]
- D. Jakubisin, C. I. Phelps, and R. M. Buehrer, "Iterative joint detection, decoding, and synchronization with a focus on frame timing," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Istanbul, Turkey, Apr. 2014, pp. 446–451. [61]
- D. Jakubisin and R. M. Buehrer, "Improved modulation classification using a factor-graph-based iterative receiver," in *Proc. IEEE MILCOM*, Oct. 2012, pp. 1–6. [62]

Chapter 2

Background

2.1 History

The Turbo Principle

Shannon derived the capacity of a communication link in 1948 [63]. Yet, it was unclear for several decades if and how capacity could be achieved in practice. In 1993, Berrou, Glavieux, and Thitimajshima discovered that near-Shannon capacity could be achieved through turbo decoding of parallel concatenated convolutional codes [6]. The concept behind turbo decoding is to pass extrinsic information between two parallel convolutional decoders. This discovery led to renewed interest in LDPC codes originally developed by Gallager [64–66]. More generally, it became apparent that the *turbo principle* could be applied to receiver functions beyond just decoding including, for example, equalization, multiuser detection, and coded modulation [67]. This led to the development of *iterative receivers* which pass probabilistic information in the form of extrinsic distributions up and down the receiver chain. In contrast, traditional receivers process information once as it is passes sequentially through the receiver.

Graphical Models

In the field of coding, early uses of graphical models include Gallager's work on LDPC codes [64], Tanner graphs [68], and trellis diagrams [69]. In 1988, Pearl developed Bayesian networks and belief propagation (BP) [70] which have been highly successful tools for inference in a broad range of fields including communications. Wiberg showed in [71] that trellis diagrams and Tanner graphs are connected to Pearl's Bayesian networks. This lead to the development of factor graphs and the interpretation of the sum-product algorithm as a message passing algorithm [71,72]. Factor graphs and the sum-product algorithm have been shown to generalize numerous other algorithms such as the Viterbi algorithm, the BCJR algorithm, and the turbo-decoding algorithm [71–73].

Introductions to factor graphs and the sum-product algorithm are available in [7, 74, 75].

Another form of graphical model are Markov random fields which are well known for their applicability to the Ising model as well as image processing and computer vision [76, 77]. One feature that differentiates factor graphs from Bayesian networks and Markov random fields is the degree of specificity provided about the probability distribution [78, Chp. 4]. As the name implies, factor graphs explicitly define the structure of the factors of the joint probability distribution. Yet, with appropriately defined models, the sum-product algorithm, belief propagation, and inference in Markov random field are equivalent.

Iterative Receivers

Factor graphs and the sum-product algorithm have served as a foundation for development of iterative receiver algorithms [7, 8, 74, 75, 79]. In particular, the sum-product algorithm is effective for decoding, demodulation, multi-user detection, and equalization, that is, tasks which involve discrete variables and probability distributions with hard constraints. Wymeersch provides an overview on iterative receiver design based on the factor graph approach [79]. Theoretical justification was provided for iterative code-aided synchronization based on the EM algorithm [31] in [80]. The EM algorithm justifies the use of posterior expectations for estimation of channel and synchronization parameters and was further explored in [81–84], among others. An exaustive list of iterative receiver developments is not possible, but we include here a number important works in synchronization [82, 85, 86], channel estimation [41, 87–93], and equalization [8, 39, 94–99].

Recently, approximate message passing (AMP) algorithms have been developed including Gaussian approximated belief propagation [100] and generalized AMP (GAMP) [49]. GAMP is applied to joint OFDM channel estimation and decoding in impulsive (non-Gaussian) noise in [30]. Another recent algorithm merging belief propagation and the mean field approximation is developed in [35] and is applied to frequency domain equalization in an iterative receiver in [101].

Theoretical Developments in Message Passing Algorithms

A very influential work by Yedidia *et al.* [55] made a connection between the sum-product algorithm (belief propagation) and the Bethe free energy. Specifically, it was discovered that the fixed points of the sum-product algorithm correspond to minima of the Bethe free energy [15, 56]. This connection has lead to further insight into belief propagation algorithms.

It has lead to convergent algorithms which seek to directly minimize the Bethe free energy [56, 102–104]. Other free energy approximations (for example, the Kikuchi free energy) have lead to the development of generalized belief propagation (GBP) algorithms [15]. Insight into the convergence of belief propagation algorithms has been gained [105–107]. More specifically, [106] provides an analysis of the effect of message errors (for example, due to quantization) and from this analysis draws conclusions on convergence. Sufficient conditions for convergence of the sum-

product algorithm are given in [107].

There has been a development of tree re-weighted belief propagation algorithms in which the graphical model is decomposed into a combination of convex tree structures [108,109]. Algorithms have also been developed based on the linear programming relaxations of the problem [110–112]. In the work of Hazan and Shashua [113] *norm-product* algorithms—a class of approximate inference message passing algorithms—are derived. They are based on free energy approximation and generalize the sum-product, max-product, convex sum-product, and convex max-product algorithms as well as globally convergent algorithms for the linear programming relaxation.

A summary of key works is provided in the following list:

- Yedidia 2001 [55]: This paper shows that BP fixed points correspond to stationary points of the Bethe free energy in statistical physics. From this connection, generalized belief propagation (GBP) algorithms are derived based on the Kikuchi free energy approximation. The Kikuchi approximation is more accurate than that of the Bethe free energy. Similarly, the authors show that the GBP algorithms can be significantly more accurate that ordinary BP.
- Yuille 2002 [102]: CCCP convergent algorithm based on minimizing Bethe or Kikuchi free energy.
- Teh 2002 [103]: UPS convergent algorithm.
- Yedidia 2003 [114]: Provides a more popular introduction to GBP.
- Heskes 2003 [56]: Extends the work of Yedidia *et al*. [55] to show that stable fixed points of BP are (not just stationary points, but) minima of the Bethe free energy. Presents a double-loop algorithm for minimizing the Bethe free energy that can converge even when loopy BP does not.
- Heskes 2004 [105]: Sufficient conditions for uniqueness if loopy BP fixed points. (Working hypothesis in literature at this time is that uniqueness of the loopy BP fixed point guarantees convergence).
- Welling 2004 [115]: BP does not provide a means of computing the joint distribution between two non-neighboring nodes in a graph. Presents linear response algorithms for approximate inference specifically related to determining joint distributions of non-neighboring nodes.
- Wainwright 2005 [108, 109]: Idea of decomposing the graphical model into a combination of convex tree structures and the resulting tree reweighted BP algorithms (sum-product and max-product).
- Ihler 2005 [106]: An analysis of the effect of message errors (for example, due to quantization). Conclusions on convergence also drawn.

- Yedidia 2005 [15]: Complete work on the connection between belief propagation and free energy minimization and GBP algorithms.
- Heskes 2006 [104]: Sufficient conditions for the Kikuchi free energy to be convex are derived. A class of converging double-loop algorithms are derived "in which each inner loop correponds to constrained minimization of a convex bound on the Kikuchi free energy, and each outer loop step to a recalculation of this bound" [104]. Connections are made with CCCP [102] and UPS [103].
- Mooij 2007 [107]: Sufficient conditions for convergence of the sum-product algorithm.
- Weiss 2007 [110]: This paper deals with finding the MAP assignment in a graphical model and linear programming relaxations. It defines *convex* BP which refers to belief propagation with a convex free energy approximation. Conditions are given for which the MAP solution can be extracted from the convex BP beliefs.
- Werner 2007 [111]: Motivated by image analysis and pattern recognition, the work explores linear programming approaches to the problem of computing the MAP solution.
- Globerson 2008 [112]: Presents an algorithm for approximating the MAP solution in a graphical model by "block coordinate descent in a dual of the linear programming (LP) relaxation of the MAP" [112] which always converges. As with the max-product algorithm, it takes advantage of the graphical model.
- Hazan 2010 [113]: In this work, *norm-product* algorithms—a class of approximate inference message passing algorithms—are derived. They are based on free energy approximation and generalize the sum-product, max-product, convex sum-product (i.e., convex BP), and convex max-product algorithms as well as globally convergent algorithms for the LP relaxation.
- Komodakis 2011 [116]: Motivated by computer vision, this work presents algorithms for energy minimization in Markov random fields (MRF). They attempt to solve an MRF optimization problem by first decomposing it into a set of subproblems.

2.2 Overview of Algorithms

2.2.1 Factor Graphs

A factor graph is an undirected bipartite graph which represents the factorization of a function. It is constructed by creating a factor node for each of the function's factors and a variable node for each of the associated variables. Wherever a variable appears in the argument of a factor, an edge is drawn between the variable node and the factor node in the graph.

In the context of probability theory, factor graphs are used to represent the factorization of a joint probability distributions and are a type of probabilistic graphical model [78]. Factor graphs enable efficient inference through message passing algorithms which operate by passing messages along the edges of the graph.

Let $\boldsymbol{\chi} = [\chi_1, \chi_2, \ldots, \chi_K]^{\mathrm{T}}$ be a vector of random variables and let x_i represent a possible realization of random variable χ_i. The joint probability distribution $p_{\chi_1,\chi_2,\ldots,\chi_K}(x_1, x_2, \ldots, x_K)$ is expressed using vector notation as $p_{\boldsymbol{\chi}}(\mathbf{x})$. Throughout this work we use x_i to represent both the random variable and the possible realizations and write the joint distribution simply as $p(\mathbf{x})$. A generic factorization of the joint probability distribution is given by

$$p(\mathbf{x}) = \prod_{a \in \mathcal{A}} f_a(\mathbf{x}_a) \tag{2.1}$$

where $\mathcal{A}$ denotes the set of all factors and $\mathbf{x}_a$ is a vector of the variables which appear in the argument of f_a. A factor node and a variable node are said to be neighbors if the variable appears in the argument of the factor, that is, they are connected by an edge in the graph. The neighbors of factor node f_a are given by $\mathcal{N}(a)$ and the neighbors of variable node x_i are given by $\mathcal{N}(i)$.

Consider an example joint distribution $p(x_1, x_2, x_3, x_4, x_5)$ which is factored as follows:

$$p(x_1, x_2, x_3, x_4, x_5) = \underbrace{p(x_1|x_2, x_3)}_{f_a} \underbrace{p(x_2)}_{f_b} \underbrace{p(x_3|x_4)}_{f_c} \underbrace{p(x_4)}_{f_d} \underbrace{p(x_4|x_5)}_{f_e} \underbrace{p(x_5)}_{f_f}. \tag{2.2}$$

The notation $f_{(\cdot)}$ as shown in (2.2) is used to label the factors when their distribution notation is cumbersome. The factor graph of (2.2) is shown in Fig. 2.1. A shown in Fig. 2.1, the factor graph conveys the conditional independence present within the model.

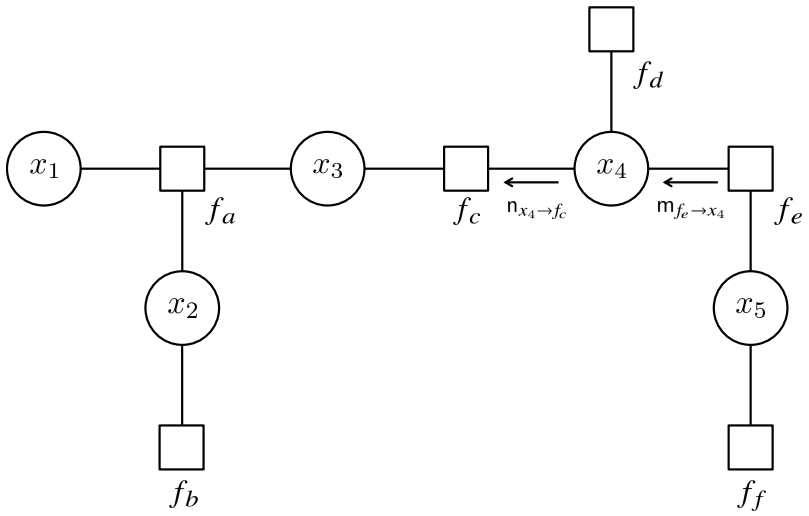

Figure 2.1: Example factor graph

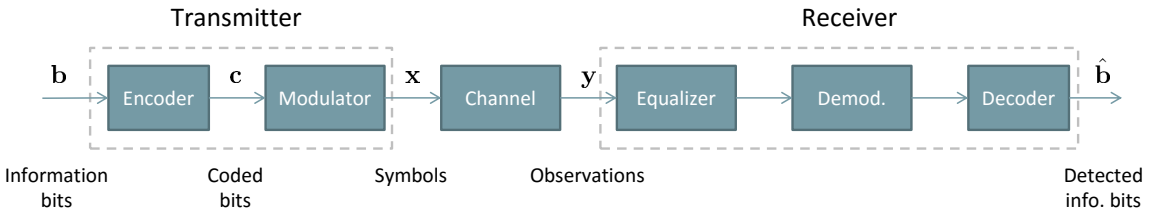

Figure 2.2: Block diagram of an example communication system.

Communications Example

In order to provide an example of a factor graph, consider a communications system in which N_b information bits $\mathbf{b}$ are encoded with an error correction code to produce N_c coded buts $\mathbf{c}$. Subsequently, the coded bits are modulated to form K symbols $\mathbf{x}$ which are transmitted over a wireless channel, for example, a multipath channel. The signal is corrupted by additive white Gaussian noise after which the receiver acquires a set of observations $\mathbf{y}$. A diagram of the communications system is shown in Fig. 2.2.

An initial, high-level factorization is possible taking into account the conditional independence between the information bits, coded bits, symbols, and observations as follows:

$$p(\mathbf{y}, \mathbf{x}, \mathbf{c}, \mathbf{b}) = p(\mathbf{y}|\mathbf{x})p(\mathbf{x}|\mathbf{c})p(\mathbf{c}|\mathbf{b}) \prod_{i=1}^{N_b} p(b_i), \tag{2.3}$$

where we have assumed that the information bits are independent and identically distributed (iid) *a priori*. The factor graph associated with the factorization in (2.3) is provided in Fig. 2.3. As a detailed example of factorization, consider two-tap multipath channel, QPSK modulation, and a 'simple' systematic LDPC code. The factored distribution is given by

$$p(\mathbf{y}, \mathbf{x}, \mathbf{c}, \mathbf{b}) = \underbrace{\prod_{k=1}^{K+1} p(y_k|x_k, x_{k-1})}_{p(\mathbf{y}|\mathbf{x})} \underbrace{\prod_{k=1}^{K} p(x_k|c_{2k-1}, c_{2k})}_{p(\mathbf{x}|\mathbf{c})} \underbrace{\prod_{i=1}^{N_b} p(c_i|b_i) \prod_{i=1}^{N_c-N_b} p(\mathbf{c}_i)}_{p(\mathbf{c}|\mathbf{b})} \prod_{i=1}^{N_b} p(b_i), \tag{2.4}$$

where $\mathbf{c}_i$ is a subset of the coded bits corresponding to the ith parity check $p(\mathbf{c}_i)$. The factor graph is shown in Fig. 2.4. The factors $p(x_k|c_{2k-1}, c_{2k})$ are conditional distributions which convey the modulation constraints. Similarly, the factors $p(c_i|b_i)$ and $p(\mathbf{c}_i)$ convey the code constraints. These factors are hard constraints because of the deterministic relationship between the information bits, coded bits, and symbols.

The high-level factorization shown in Fig. 2.3 is useful for abstracting away some of the details of the factor graph and the message passing algorithm. The factors in (2.3) are further factored when implementing the probabilistic model. However, it will be beneficial to use high-level factors for terms which are well known and not the focus of this work.

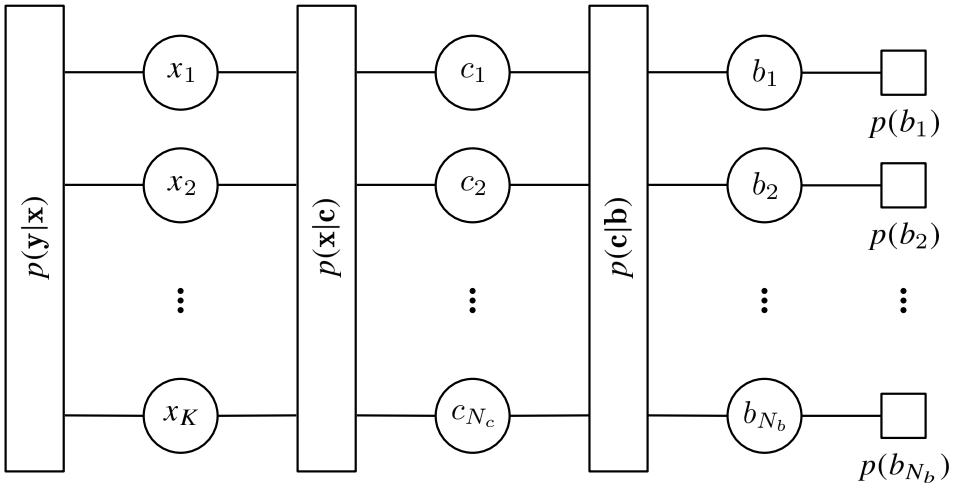

Figure 2.3: High-level factorization of an example joint probability distribution.

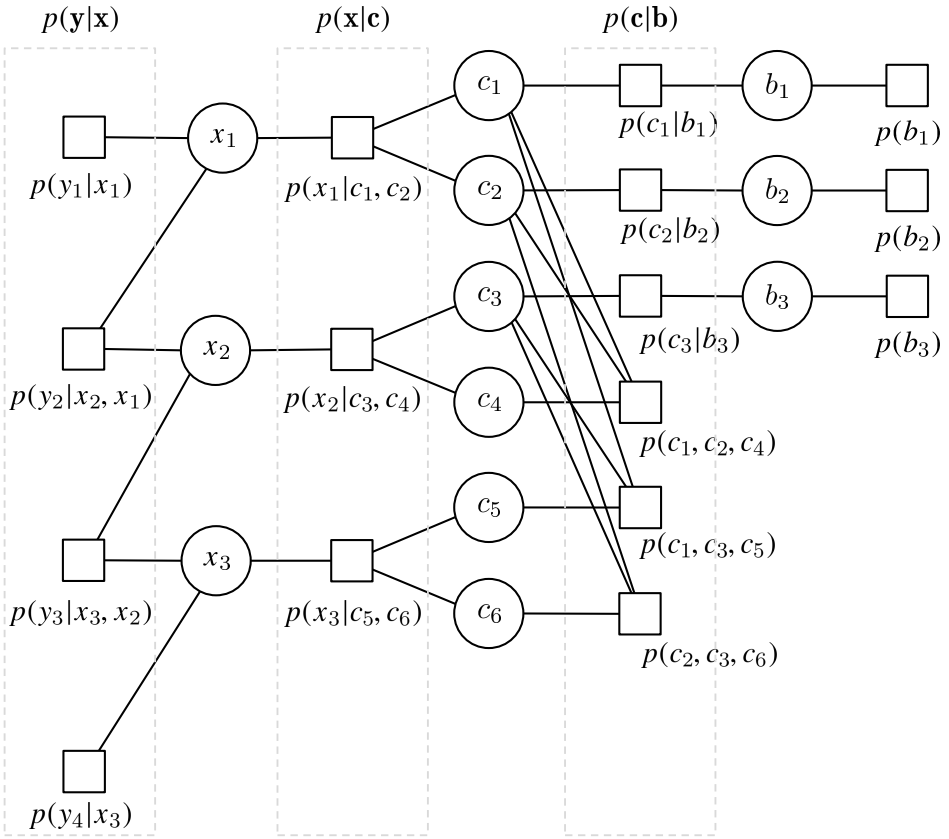

Figure 2.4: Factor graph of (2.4) with $N_b = 3$, $N_c = 6$, and $K = 3$.

2.2.2 Sum-Product Algorithm (SPA)

The sum-product algorithm (SPA) is used to efficiently compute marginal posterior probabilities from a joint distribution[1] by making local summations at the nodes of the factor graph and by passing these summations in the form of "messages" along the edges of the graph. Messages are labeled according to the node and edge they are associated with and are functions of the variable. When the factor graph has a tree structure, the posterior distributions computed by the SPA are exact. However, when the graph contains cycles, the result of the SPA approximates the posterior distributions. Following the convention in Bayesian statistics, we refer to the marginal posterior distributions (whether exact or approximate) as *beliefs*. The SPA is suited for inference in graphs were all the variables are discrete (as implied by the use of the "sum" to perform marginalization).

The messages are denoted with sans-serif fonts where $\mathsf{m}_{f_a \to x_i}$ denotes the message from factor node f_a to variable node x_i and $\mathsf{n}_{x_i \to f_a}$ denotes the message from variable node x_i to factor node f_a. The belief for variable x_i is denoted by $b(x_i)$ and can be computed from the SPA messages. The SPA computation rules for the messages and beliefs are given as follows.

Factor update The message from factor node f_a to variable node x_i is computed according to

$$\mathsf{m}_{f_a \to x_i}(x_i) = \sum_{\mathbf{x}_a \setminus x_i} f_a(\mathbf{x}_a) \prod_{j \in \mathcal{N}(a) \setminus i} \mathsf{n}_{x_j \to f_a}(x_j). \tag{2.5}$$

Variable update The message from variable node x_i to factor node f_a is computed according to

$$\mathsf{n}_{x_i \to f_a}(x_i) = \prod_{c \in \mathcal{N}(i) \setminus a} \mathsf{m}_{f_c \to x_i}(x_i). \tag{2.6}$$

Belief The marginal posterior probability or *belief* is computed according to

$$b(x_i) = \prod_{c \in \mathcal{N}(i)} \mathsf{m}_{f_c \to x_i}(x_i). \tag{2.7}$$

Returning to the example of (2.2) and Fig. 2.1, we demonstrate how the sum-product algorithm and factor graph model enable computation of the marginal distribution for x_3. The desired marginal distribution is given by

$$b(x_3) = \left(\sum_{x_1, x_2} p(x_1|x_2, x_3) p(x_2) \right) \left(\sum_{x_4} p(x_3|x_4) \underbrace{p(x_4) \underbrace{\left(\sum_{x_5} p(x_4|x_5) p(x_5) \right)}_{\mathsf{m}_{f_e \to x_4}}}_{\mathsf{n}_{x_4 \to f_c}} \right). \tag{2.8}$$

[1] This is possible since the posterior distribution is proportional to the joint distribution.

According to the SPA, the message from node f_e to variable x_4 is given by

$$\mathsf{m}_{f_e \to x_4}(x_4) = \sum_{x_5} p(x_4|x_5)\mathsf{n}_{x_5 \to f_e}(x_5). \tag{2.9}$$

Recognizing that $\mathsf{n}_{x_5 \to f_e}(x_5) = p(x_5)$, it can be seen from (2.8) that $\mathsf{m}_{f_e \to x_4}(x_4)$ contains all the information about x_5, f_e, and f_f that is necessary for computing the belief. The message from variable node x_4 to factor node f_c is given by

$$\mathsf{n}_{x_4 \to f_c}(x_4) = \mathsf{m}_{f_d \to x_4}(x_4)\mathsf{m}_{f_e \to x_4}(x_4). \tag{2.10}$$

Again, recognizing that $\mathsf{m}_{f_d \to x_4}(x_4) = p(x_4)$, we observe that $\mathsf{n}_{x_4 \to f_c}(x_4)$ corresponds to a term in (2.8). Finally, the belief $b(x_3)$ is given by $b(x_3) = \mathsf{m}_{f_a \to x_3}(x_3)\mathsf{m}_{f_c \to x_3}(x_3)$. For a complete introduction to factor graphs and the sum-product algorithm please refer to [7, 74].

2.2.3 Gaussian Belief Propagation (GaBP)

Belief propagation for continuous parameters is feasible when all involved distributions are Gaussian. Gaussian BP (GaBP) is a generalization of a number of algorithms in the literature (e.g., Kalman filtering and linear MMSE filtering). Since Gaussian distributions are fully characterized by their mean and variance, GaBP message passing involves passing these statistics on the edges of the factor graph model. As an example, GaBP for state space models leads to passing mean $\vec{\boldsymbol{\mu}}$ and covariance $\vec{\Sigma}$ in the forward recursion and passing $\overleftarrow{\Sigma}^{-1}\overleftarrow{\boldsymbol{\mu}}$ and $\overleftarrow{\Sigma}^{-1}$ in the reverse recursion [117]. Loeliger *et al*. provide an introduction to GaBP in [75].

2.2.4 Expectation Maximization (EM)

The EM algorithm [31] provides a theoretical framework for learning the parameters of a model when there exists a set of unobserved or hidden data. Let the observed data be denoted by $\mathbf{y}$, the hidden data by $\mathbf{x}$ (generally assumed to be discrete), and the complete data by $\mathbf{z} = \{\mathbf{y}, \mathbf{x}\}$. The unknown parameters are denoted by vector $\boldsymbol{\theta}$. For the problem at hand, the expectation (E) and maximization (M) steps are given by the following expressions:

$$\text{E step: } Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)}) = \sum_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) \ln p(\mathbf{z}|\boldsymbol{\theta}) d\mathbf{x} \tag{2.11}$$

$$\text{M step: } \hat{\boldsymbol{\theta}}^{(p)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)}), \tag{2.12}$$

where $\hat{\boldsymbol{\theta}}^{(p)}$ represents the estimates of the parameters in the pth iteration. Expectation conditional-maximization (ECM) is a variant of the EM algorithm in which maximization is performed sequentially for each parameter while conditioning on the most recent estimate of the other parameters [118].

The EM or ECM algorithms are integrated with the factor graph model and the SPA for estimation of parameters of the factor functions. The SPA is used to compute marginal posteriors (beliefs) conditioned on the current estimates of the parameters (i.e., the terms which result from $p(\mathbf{x}|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)})$ when simplifying (2.11)). The beliefs from the SPA are used to evaluate the expectation in (2.11) and re-estimate the parameters (2.12). EM is an iterative algorithm which is known to converge to a maximum of the likelihood function.

2.2.5 Expectation Propagation (EP)

The expectation propagation (EP) algorithm is a method of performing approximate inference which is especially useful when BP is infeasible [45, 119, 120]. EP has a message passing interpretation and can be shown to be a generalization of BP. When computation of a marginal belief $b(x_i)$ from a joint distribution $p(\mathbf{x})$ is computational infeasible, EP provides an approach to finding an approximating distribution $q(\mathbf{x})$. The distribution $q(\mathbf{x})$ is constrained to a family of distributions such that its marginals $q(x_i)$ can be easily computed. If an approximate distribution can be found which well approximates the true joint distribution, we can expect to achieve good approximations of the beliefs as well.

In the following a brief description of EP is provided. Consider the joint distribution given in (2.1). The approximating distribution is similarly factored into a collection of approximating terms

$$q(\mathbf{x}) = \prod_{a \in \mathcal{A}} \tilde{f}_a(\mathbf{x}_a). \tag{2.13}$$

Let all terms other than $\tilde{f}_a(\mathbf{x}_a)$ be denoted by

$$q^{\setminus a}(\mathbf{x}) = \frac{q(\mathbf{x})}{\tilde{f}_a(\mathbf{x}_a)} = \prod_{c \in \mathcal{A} \setminus a} \tilde{f}_c(\mathbf{x}_c). \tag{2.14}$$

In order to approximate factor $\tilde{f}_a(\mathbf{x}_a)$, we seek to minimize the divergence $D(p(\mathbf{x})||\tilde{f}_a(\mathbf{x}_a)q^{\setminus a}(\mathbf{x}))$ with respect to $\tilde{f}_a(\mathbf{x}_a)$. In order to make this tractable, it is assumed that $\tilde{f}_c(\mathbf{x}_c)$ $\forall c \neq a$ well approximate their respective terms and we instead minimize divergence as follows:

$$\tilde{f}_a(\mathbf{x}_a) = \underset{\tilde{f}_a(\mathbf{x}_a)}{\arg\min}\, D\left(f_a(\mathbf{x}_a)q^{\setminus a}(\mathbf{x})||\tilde{f}_a(\mathbf{x}_a)q^{\setminus a}(\mathbf{x})\right). \tag{2.15}$$

This approximation is refined by iteratively computing (2.15) for all factors $a \in \mathcal{A}$.

EP can be interpreted a message passing algorithm when a fully-factorized approximating distribution is assumed [120]. The message update rule for factor nodes is given by

$$\mathsf{m}_{f_a \to x_i}(x_i) = \frac{1}{\mathsf{n}_{x_i \to f_a}(x_i)} \text{proj}\left[\mathsf{n}_{x_i \to f_a}(x_i) \int \cdots \int f_a(\mathbf{x}_a) \prod_{j \in \mathcal{N}(a) \setminus i} \mathsf{n}_{x_j \to f_a}(x_j) dx_j\right], \tag{2.16}$$

where proj[·] denotes projection of the argument into the family of approximating distributions. The projection is performed by minimizing the KL divergence. Two observations provide intuition into EP. First, we observe that the term

$$\int \cdots \int f_a(\mathbf{x}_a) \prod_{j \in \mathcal{N}(a) \setminus i} \mathsf{n}_{x_j \to f_a}(x_j) dx_j$$

in (2.16) is the message computation rule for BP. Thus, the distribution which is to be projected is the posterior belief of x_i (i.e., $b(x_i) = \mathsf{n}_{x_i \to f_a}(x_i)\mathsf{m}_{f_a \to x_i}(x_i)$). Rather than project the BP message into the approximating family. The belief is projected and then the message $\mathsf{n}_{x_i \to f_a}(x_i)$ is removed. Second, when the distribution of the belief is in the approximating family of distributions, (2.16) reduces to BP. This is due to the fact that the projection does not modify the distribution and $\mathsf{n}_{x_i \to f_a}(x_i)$ cancels from the expression.

2.2.6 Mean Field (MF) Approximation

The mean field approximation comes from statistical physics and is based on approximating the joint distribution with a product of marginal beliefs as given by

$$q(\mathbf{x}) = \prod_{i=1}^{K} b(x_i). \tag{2.17}$$

The MF approximation leads to a type of variational message passing. The message computations for factor nodes and variables node according to the MF approximation are given as follows.

Factor update The message from factor node f_a to variable node x_i is computed according to

$$\mathsf{m}_{f_a \to x_i}(x_i) = \exp\left\{ \int \cdots \int \ln f_a(\mathbf{x}_a) \prod_{j \in \mathcal{N}(a) \setminus i} \mathsf{n}_{x_j \to f_a}(x_j) dx_j \right\}. \tag{2.18}$$

Variable update The message from variable node x_i to factor node f_a is computed according to

$$\mathsf{n}_{x_i \to f_a}(x_i) = \prod_{c \in \mathcal{N}(i)} \mathsf{m}_{f_c \to x_i}(x_i). \tag{2.19}$$

When the factors $f_a(\mathbf{x}_a)$ are within the exponential family, the message computation in (2.18) performs integration over the argument of the exponential (i.e., expectation with respect to $\ln f_a(\mathbf{x}_a)$). Another difference between MF and BP is that posterior beliefs are passed from the variable nodes to the factor nodes as shown in (2.19) rather than extrinsic distributions as shown in (2.6). The MF approximation is well suited for continuous variables (or when both continuous and discrete

variables are present in the model). However, it is not capable of handling hard constraints found in the demodulation and decoding factors. Recognizing the complementary advantages of the MF approximation and BP, a combined BP-MF message passing algorithm has been proposed in the literature which is presented in the next section.

2.2.7 Combined BP-MF Framework

Riegler *et al.* provided a theoretical justification for combining BP and MF in [35]. The justification is based on defining a region-based free energy with respect to the factorization of the probability distribution. Consider the factorization of (2.1). The factor graph (equivalently, the factors) are partitioned into a MF subgraph $\mathcal{A}_{\text{MF}}$ and a BP subgraph $\mathcal{A}_{\text{BP}}$ where $\mathcal{A}_{\text{MF}} \cap \mathcal{A}_{\text{BP}} = \varnothing$ and $\mathcal{A} \triangleq \mathcal{A}_{\text{MF}} \cup \mathcal{A}_{\text{BP}}$. The variables associated with each portion of the graph are given by

$$\mathcal{I}_{\text{MF}} \triangleq \bigcup_{a \in \mathcal{A}_{\text{MF}}} \mathcal{N}(a) \quad \text{and} \quad \mathcal{I}_{\text{BP}} \triangleq \bigcup_{a \in \mathcal{A}_{\text{BP}}} \mathcal{N}(a),$$

respectively. From this definition, the region based free energy is constructed from the following regions [35]

1. A single MF region containing all factors and variables in the MF portion with a counting number of 1.
2. Large regions from the Bethe free energy $R_a \triangleq (\mathcal{N}(a), \{a\})$, with a counting number of 1.
3. Small regions from the Bethe free energy $R_i \triangleq (\{i\}, \varnothing)$, with counting number $1 - |\mathcal{N}_{\text{BP}}(i)| - \text{I}(i \in \mathcal{I}_{\text{MF}})$.

The region based free energy is a function of the joint beliefs $b_a(\mathbf{x}_a)$ and marginal beliefs $b_i(x_i)$ as given by [35]

$$\begin{aligned} F_{\text{BP,MF}} = &\sum_{a \in \mathcal{A}_{\text{BP}}} \sum_{\mathbf{x}_a} b_a(\mathbf{x}_a) \ln \frac{b_a(\mathbf{x}_a)}{f_a(\mathbf{x}_a)} \\ &- \sum_{a \in \mathcal{A}_{\text{MF}}} \sum_{\mathbf{x}_a} \prod_{i \in \mathcal{N}(a)} b_i(x_i) \ln f_a(\mathbf{x}_a) \\ &- \sum_{i \in \mathcal{I}} (|\mathcal{N}_{\text{BP}}(i)| - 1) \sum_{x_i} b_i(x_i) \ln b_i(x_i). \end{aligned} \tag{2.20}$$

Constraints for the factorization of the MF portion beliefs as well as normalization and marginalization constraints are detailed in [35].

The combined BP-MF message passing rules are as follows. The messages from factor nodes to variable nodes within the MF subgraph are given by [35]

$$\mathsf{m}^{\text{MF}}_{f_a \to x_i}(x_i) = \exp\left(\sum_{\mathbf{x}_a \setminus x_i} \prod_{j \in \mathcal{N}(a) \setminus i} n_{x_j \to f_a}(x_j) \ln f_a(\mathbf{x}_a) \right) \tag{2.21}$$

for all $a \in \mathcal{A}_{\text{MF}}, i \in \mathcal{N}(a)$. The messages from factor nodes to variable nodes within the BP subgraph are given by [35]

$$\mathsf{m}^{\text{BP}}_{f_a \to x_i}(x_i) = \sum_{\mathbf{x}_a \setminus x_i} f_a(\mathbf{x}_a) \prod_{j \in \mathcal{N}(a) \setminus i} n_{x_j \to f_a}(x_j) \tag{2.22}$$

for all $a \in \mathcal{A}_{\text{BP}}, i \in \mathcal{N}(a)$. Finally, messages passed from variable nodes to factor nodes throughout the entire graph are given by [35]

$$\mathsf{n}_{x_i \to f_a}(x_i) = \prod_{c \in \mathcal{N}_{\text{BP}}(i) \setminus a} \mathsf{m}^{\text{BP}}_{f_c \to x_i}(x_i) \prod_{c \in \mathcal{N}_{\text{MF}}(i)} \mathsf{m}^{\text{MF}}_{f_c \to x_i}(x_i) \tag{2.23}$$

for all $a \in \mathcal{A}, i \in \mathcal{N}(a)$ where $\mathcal{N}_{\text{BP}}(i) = \mathcal{N}(i) \cap \mathcal{A}_{\text{BP}}$ and $\mathcal{N}_{\text{MF}}(i) = \mathcal{N}(i) \cap \mathcal{A}_{\text{MF}}$. In (2.23), the messages to factor nodes in the BP subgraph are extrinsic messages (as denoted by the exclusion of a in the first product). On the other hand, the messages to factor nodes in the MF subgraph are posterior beliefs. Thus, when a message is directed from or to a *factor node* in the MF subgraph, the MF computation rules apply. Similarly, when a message is directed from or to a *factor node* in the BP subgraph, the BP computation rules apply. The message passing algorithm is determined by how the graph is partitioned into BP and MF portions. The fixed points of this combined message passing algorithm correspond to minima of (2.20) which provides justification to the framework.

Chapter 3

Iterative Receiver Design and Frame Synchronization

3.1 Introduction

Next generation wireless communications systems are pushing the limits of both energy efficiency and spectral efficiency. Energy efficiency is aided by advanced error correction codes while spectral efficiency is aided by multiple-input multiple-output (MIMO) transmission, higher order modulations, adaptive rate control, orthogonal frequency-division multiplexing (OFDM), and the reduction of training overhead. However, these advances present a challenge at the receiver when it comes to accomplishing tasks such as synchronization, channel estimation, and equalization. This has motivated the development of code-aided algorithms in the technical literature (see, for example, [8, 67, 82, 95]). These algorithms are generally iterative—making use of probabilistic information from the decoder to improve the performance of earlier receiver tasks.

Frame synchronization refers to the acquisition of the start time of a frame of transmitted data. Accurate synchronization is particularly important for block codes such as turbo codes and low-density parity-check (LDPC) codes where the information of the whole frame is lost if the frame start time is incorrectly estimated. Frame synchronization is traditionally accomplished by adding a sync word as a preamble to the frame and correlating the received signal with this known sequence [9]. The complexity of the correlation technique is low, but it is sub-optimal in terms of frame synchronization error probability. Optimal maximum *a posteriori* probability (MAP) frame synchronization for *uncoded data* is derived for binary phase-shift keying (BPSK) in [121] and for higher-order modulations in [122] where significant gains in frame synchronization performance are shown when compared to the correlation technique.

MAP frame synchronization for *coded data* requires marginalization over the entire set of codewords. Herzet and Vandendorpe recognized that the sum-product algorithm is capable of efficiently performing the desired marginalization over the coded data when the factor graph of the

joint probability distribution is acyclic [10]. In general, however, marginalization over the entire set of codewords is too complex and thus a number of code-aided frame synchronization methods have been proposed in the literature [11–14, 123–132]. Frame synchronization is performed in [11, 12] by computing the syndrome posterior probability (SPP) of the code's parity check relations. This same approach is applied to blind code recognition in [133]. Code-aided frame synchronization based on the expectation-maximization (EM) algorithm is presented in [13]. Further theoretical justification for the EM-motivated algorithm is provided in [128] where it is shown to be an approximation to MAP hypothesis testing (HT). Frame synchronization based on free energy minimization is proposed in [129] taking advantage of recent connections between belief propagation and free energy minimization from statistical physics [15].

In the case of uncoded signals, likelihood based frame synchronization is derived for flat-fading channels in [134] and frequency-selective channels in [135]. However, previous work on code-aided frame synchronization predominately assumes an additive white Gaussian noise (AWGN) channel. Two exceptions include [124] and [130] which develop frame synchronization based on decoder state estimation for convolutional codes and provide results for flat-fading and frequency-selective channels, respectively. Moreover, previous work on code-aided frame synchronization has typically assumed knowledge of the other channel parameters and the noise power.

Code-aided frame synchronization techniques require processing each potential frame offset individually and the processing requires decoding [13, 14, 126, 128] or a decoding-like operation [10–12, 127, 129, 131, 132]. Therefore, the complexity of the receiver as a whole scales linearly with the number of frame offsets considered by the code-aided synchronizer. Robertson proposed a two-stage frame synchronizer, referred to as list synchronization, in which a low-complexity decision rule is used to obtain a list of the most likely frame offsets and then this list is processed by a high-complexity decision rule [14]. Other papers that have applied the list synchronization method in their analysis include [124, 126]. However, parameters of the list synchronization method (for instance, the size of the list) are determined through simulation. What is lacking with the list synchronization method is the ability to choose the optimal number of candidate frame offsets to be considered by the high complexity decision rule.

In this chapter, the problem of frame synchronization is considered for coded signals in *multipath channels*. We consider the case in which the channel is not known *a priori*, and develop an iterative receiver structure for frame synchronization, channel estimation, equalization, demodulation, and decoding. In contrast to Robertson's list synchronizer, the pre-processing stage uses a Bayesian technique to select, on a frame-by-frame basis, the subset of frame offsets to be processed by a code-aided frame synchronizer. The proposed technique selects the minimum number of offsets for a given target performance by computing the highest posterior density (HPD) region of the frame distribution. Numerical results are provided to characterize the expected number of frame offsets in the HPD region because the computational complexity of code-aided frame synchronization is proportional to the number of frame offsets which must be processed.

Two code-aided frame synchronization methods are developed for multipath channels. The first algorithm, based on the SPP method, is designed to have a low-complexity. The second algo-

rithm, based on the EM method, is designed to be an iterative approximation to joint MAP synchronization and data detection. In the case of unknown channel parameters, code-aided frame synchronization is generalized to code-aided hypothesis testing of the frame offset and phase ambiguity. We consider two implementations of the EM-based method in the iterative receiver. The first, HT/ECM, performs EM-based code-aided hypothesis testing of the frame offset and phase ambiguity while iteratively re-estimating the remaining channel parameters using the expectation-conditional maximization (ECM) algorithm [118]. The second, HT, performs EM-based code-aided hypothesis testing without re-estimating the remaining parameters in order to reduce complexity.

This chapter also explores the complexity-performance trade-off of code-aided frame synchronization in multipath channels. Regions of signal-to-noise ratio (SNR) are identified for which code-aided frame synchronization provides an improvement in performance while maintaining a complexity within a factor of $2\times$ that of conventional techniques. The performance of the receiver is demonstrated in a scenario in which the proposed receiver achieves a gain of up to 3 dB with a complexity increase of only 20%. We also show that code-aided frame synchronization may be reliably performed at SNR values below that required for decoding. This is important, for example, to support acknowledgment (ACK)/non-acknowledgment (NACK) and hybrid automatic repeat request (hybrid ARQ) protocols or to achieve reliable synchronization in a collaborative communications scenario where multiple receivers may cooperate to decode a signal at low SNR.

In summary, the contributions of this work are as follows:

- Development of code-aided frame synchronization for multipath channels
- Development of a novel frame pre-processing stage to allow a trade-off between performance and complexity
- Design of an iterative receiver for frame synchronization, channel estimation, equalization, demodulation, and decoding
- Analysis of the complexity-performance trade-off in order to identify regions where code-aided frame synchronization is beneficial
- Analysis of code-aided frame synchronization performance at SNR values below that required for decoding.

This chapter is organized as follows. The system model is given in Section 3.2 and MAP frame synchronization is reviewed in Section 3.3. A brief overview of the proposed receiver structure is presented in Section 3.4. A novel frame pre-processing stage—which guarantees a maximum probability of excluding the true frame offset—is presented in Section 3.5. In Section 3.6, the SPP and EM-based code-aided frame synchronization techniques are developed for multipath channels and numerical results are provided in Section 3.7. The receiver is developed in Section 3.8 for an unknown channel, including the HT/ECM and HT receivers, and numerical results are presented. Finally, the chapter is concluded in Section 3.9.

3.2 System Model

A burst transmission scheme is considered in this work. At the transmitter an error correction code is applied to a sequence of N_b information bits $\mathbf{b}$ to produce a sequence of N_c coded bits $\mathbf{c}$. Subsequently, the coded bits are modulated using a digital phase-amplitude modulation to form a complex sequence of coded data symbols denoted by $\mathbf{a} = [a_0, \ldots, a_{N_a-1}]^{\mathrm{T}}$ with length N_a. The symbol alphabet is given by the set $\{m_i\}_{i=1}^{M}$, where M is the order of the modulation, and is normalized to unit average power (i.e., $\frac{1}{M}\sum_{i=1}^{M}|m_i|^2 = 1$). A frame is constructed from the concatenation of a sync word and the coded data sequence as given by $\mathbf{x} = [\mathbf{s}^{\mathrm{T}}\mathbf{a}^{\mathrm{T}}]^{\mathrm{T}}$ where $\mathbf{s} = [s_0, \ldots, s_{N_s-1}]^{\mathrm{T}}$ is the sync word with length N_s. The frame length is given by $K = N_s + N_a$.

The low-pass equivalent of the transmitted frame is given by

$$q(t) = \sum_{k=0}^{K-1} x_k p(t - kT), \tag{3.1}$$

where $p(t)$ is a real valued pulse with unit energy and T is the symbol period. The multipath channel is modeled with a conventional tapped delay line with tap spacing equal to T [136]. The low-pass equivalent of the channel impulse response is given by

$$h(t) = \sum_{l=0}^{L-1} h_l \delta(t - lT - \tau), \tag{3.2}$$

where h_l are the complex channel coefficients for each of the L resolvable paths, and τ is a time delay. Let the delay be separated into an integer multiple of the symbol period η (i.e., the frame offset) and a fractional multiple of the symbol period ϵ such that $\tau = (\eta + \epsilon)T$ where $0 \leq \epsilon < 1$. The low-pass equivalent of the received signal is given by

$$r(t) = \sum_{l=0}^{L-1} h_l \sum_{k=0}^{K-1} x_k p(t - (k + l + \eta + \epsilon)T) + w(t), \tag{3.3}$$

where $w(t)$ is a spectrally-white complex Gaussian random process representing noise. At the output of the matched filter, the received signal is given by

$$y(t) = r(t) * p(-t) = \sum_{l=0}^{L-1} h_l \sum_{k=0}^{K-1} x_k g(t - (k + l + \eta + \epsilon)T) + v(t), \tag{3.4}$$

where $g(t) = p(t) * p(-t)$ is a Nyquist pulse (for example, a raised cosine pulse) and $v(t) = \int_{-\infty}^{\infty} w(\xi)p(\xi - t)d\xi$ is filtered noise. When the symbol timing ϵ is known, samples y_n of the matched filter output may be taken at $t = (n + \epsilon)T$ as given by

$$y_n = y((n + \epsilon)T) = \sum_{l=0}^{L-1} h_l x_{n-l-\eta} + v_n, \tag{3.5}$$

where x_i is non-zero for indexes $0 \leq i \leq K-1$ and $\{v_n\}_{n=0}^{N-1}$ are independent and identically distributed (iid) circularly-symmetric complex Gaussian random variables with variance given by σ^2. Let the domain of the frame offset be given by $\eta \in \{0, 1, \ldots, H-1\}$ where H is assumed to be large due to prior uncertainty in the start time of the frame. The vector of samples of the matched filtered signal is denoted by $\mathbf{y} = [y_0, \ldots, y_{N-1}]^{\mathrm{T}}$ and has length $N = K + H + L - 2$ to account for the length of the frame, the multipath channel, and all possible frame offsets. Throughout the chapter, SNR is defined as the ratio between the instantaneous power of the multipath channel and the noise power as expressed by

$$\mathrm{SNR} = \frac{\sum_{l=0}^{L-1} |h_l|^2}{\sigma^2}. \tag{3.6}$$

3.3 MAP Frame Synchronization

Following convention, we define the optimum frame synchronizer to be the estimator which maximizes the probability of selecting the correct frame offset [121]. Given an observation of the received signal $\mathbf{y}$, this estimator is the one which maximizes the posterior probability of the frame offset $p(\eta|\mathbf{y})$. According to Bayes' rule the posterior probability is given by

$$p(\eta|\mathbf{y}) = \frac{p(\mathbf{y}|\eta)p(\eta)}{p(\mathbf{y})} \propto p(\mathbf{y}|\eta)p(\eta). \tag{3.7}$$

The distribution $p(\mathbf{y}|\eta)$ is expressed as the marginalization of the joint distribution $p(\mathbf{y}, \mathbf{x}|\eta)$ over $\mathbf{x}$ producing the following MAP estimator:

$$\hat{\eta}_{\mathrm{MAP}} = \underset{\eta \in \{0,\ldots,H-1\}}{\arg\max} \sum_{\mathbf{x}} p(\mathbf{y}|\mathbf{x}, \eta)p(\mathbf{x})p(\eta), \tag{3.8}$$

where $\sum_{\mathbf{x}}$ denotes marginalization over the domain of $\mathbf{x}$.

Given knowledge of the channel and noise parameters, the likelihood function for observation of the symbol-synchronous samples from (3.5) is given by

$$\begin{aligned} p(\mathbf{y}|\mathbf{x}, \eta) \propto & \prod_{i=0}^{\eta-1} \exp\left\{-\frac{1}{\sigma^2}|y_i|^2\right\} \prod_{i=0}^{L-2} \exp\left\{-\frac{1}{\sigma^2}\left|y_{i+\eta} - \sum_{l=0}^{i} h_l x_{i-l}\right|^2\right\} \\ & \cdot \prod_{i=L-1}^{K-1} \exp\left\{-\frac{1}{\sigma^2}\left|y_{i+\eta} - \sum_{l=0}^{L-1} h_l x_{i-l}\right|^2\right\} \\ & \cdot \prod_{i=K}^{K+L-2} \exp\left\{-\frac{1}{\sigma^2}\left|y_{i+\eta} - \sum_{l=i-K+1}^{L-1} h_l x_{i-l}\right|^2\right\} \prod_{i=K+L+\eta-1}^{N-1} \exp\left\{-\frac{1}{\sigma^2}|y_i|^2\right\}. \end{aligned} \tag{3.9}$$

In (3.9), the first and fifth products are distributions on the noise-only samples before and after the frame, respectively. The second, third, and fourth products are distributions on the signal (with

intersymbol interference due to multipath) and noise. The second and fourth products take into account the effects of multipath at the beginning and end of the frame, respectively. In order to simplify the notation, let $\alpha_i = \max(i - K + 1, 0)$ and $\beta_i = \min(i, L - 1)$ represent lower and upper limits, respectively, on the summation of the multipath terms within the exponential functions of (3.9). Further, recognizing that the term $\prod_{i=0}^{N-1} \exp\left\{-\frac{1}{\sigma^2}|y_i|^2\right\}$ is a proportionality constant with respect to the frame offset, the likelihood function can be simplified as follows:

$$p(\mathbf{y}|\mathbf{x}, \eta) \propto \prod_{i=0}^{K+L-2} \exp\left\{ \frac{2}{\sigma^2} \Re\left[y_{i+\eta}^* \sum_{l=\alpha_i}^{\beta_i} h_l x_{i-l} \right] - \frac{1}{\sigma^2} \left| \sum_{l=\alpha_i}^{\beta_i} h_l x_{i-l} \right|^2 \right\}. \tag{3.10}$$

Returning to (3.8), the prior probability of the data sequence $p(\mathbf{x})$ can be decomposed into a product of prior probabilities for the sync word and the coded symbols $p(\mathbf{s})p(\mathbf{a})$. The sync word is deterministic (i.e., $p(\mathbf{s}) = 1$ for the true sync word and is zero for all other sequences). Since the prior probability $p(\mathbf{a})$ is zero for all symbol sequences which do not correspond to a valid codeword, the complexity of computing (3.8) is $O(2^{N_b})$, i.e., it scales with the number of codewords[1]. In the case of an uncoded signal, the complexity is $O(M^L)$ due to the multipath terms in (3.10) and the marginalization can be performed using the Bahl-Cocke-Jelinek-Raviv (BCJR) algorithm [137].

The following three factors contribute to the complexity of the MAP estimator in (3.8):

- *The size of the search space $\eta \in \{0, \ldots, H-1\}$:* The time between burst transmission may create a large search space for the frame start time. A low complexity pre-processing stage is developed in Section 3.5 to address this challenge.
- *The exponential complexity of the marginalization:* As described above, the MAP estimator is too complex to implement. Therefore, in Section 3.6, we develop code-aided frame synchronization algorithms for multipath channels which approximate the MAP estimator.
- *The presence of unknown parameters:* The complex channel coefficients $\mathbf{h} = [h_0, \ldots, h_{L-1}]^{\mathrm{T}}$, the fractional (symbol) delay ϵ, and the noise power σ^2 are required for the pre-processing stage and code-aided algorithms developed in this chapter. An extension of the work to the unknown channel case is given in 3.8.

First, we provide a brief overview of the receiver structure in the following section.

3.4 Receiver Design

In the proposed iterative receiver structure, shown in Fig. 3.1, frame synchronization is performed in two stages: (1) frame pre-processing and (2) code-aided frame synchronization. Blind estima-

[1]The term O denotes the standard big O notation which is used to describe the asymptotic behavior of a function—in our case the number of computations as a function of the signal parameters.

tion is used to obtain coarse estimates of the symbol timing and channel parameters prior to the frame pre-processing stage. Knowledge of the sync word (but not of the code structure) is assumed in the frame pre-processing stage in order to narrow the search for the frame offset to the HPD region. Code-aided frame synchronization is performed for each frame offset in the HPD region as represented by the duplicate receiver chains in Fig. 3.1. When the HPD region contains a single frame offset, this offset is selected without the need to perform code-aided frame synchronization. Further processing is performed on the iterative receiver chain corresponding to the selected frame offset (as shown by the shaded region in Fig. 3.1) in order to detect the information bits. In the iterative receiver, fine estimation of the continuous channel parameters (ϵ, $\mathbf{h}$, and σ^2) is accomplished with the ECM algorithm. The sum-product algorithm [7] is utilized to perform iterative, probabilistic equalization, demodulation, and decoding in order to compute posterior probabilities of the symbols and information bits used for fine channel estimation and data detection, respectively.

3.5 Pre-Processing Stage

The frame pre-processing stage shown in Fig. 3.1 is developed in this section. Due to the burst transmission scheme, we assume that the search space for the frame offset $\{0, \ldots, H-1\}$ is large. Instead of processing all frame offsets with a code-aided algorithm, a pre-processing stage identifies a subset $\mathcal{S}$ of the most probable frame offsets based on the assumption that the data is uncoded. The pre-processing stage modifies the search space of the MAP estimator in (3.8) so that the approximate MAP estimate is given by

$$\hat{\eta}_{\text{MAP}} \approx \arg\max_{\eta \in \mathcal{S}} \sum_{\mathbf{x}} p(\mathbf{y}|\mathbf{x}, \eta) p(\mathbf{x}) p(\eta). \tag{3.11}$$

3.5.1 HPD Region-Based Pre-Processing

Given a subset of the frame offsets denoted by $\mathcal{S}$ with size $|\mathcal{S}|$, we define P_{ex} to be the probability that the true offset has been excluded from $\mathcal{S}$ (i.e., $\eta_{true} \notin \mathcal{S}$). We desire to find the smallest subset for which the probability that the true offset is included in this set is greater than or equal to $(1 - P_{ex})$. This set is known as the highest posterior density (HPD) region [138]. For example, to limit the probability of exclusion to $P_{ex} = 0.01$, the set of frame offsets $\mathcal{S}$ which must be processed by the code-aided method is given by the 0.99 HPD region. An example posterior distribution is shown in Fig. 3.2 where the 0.99 HPD region is shown with marker "x". In this example, the HPD region contains seven offsets: $\{15, 22, 23, 24, 25, 26, 27\}$. The number of offsets is dynamically chosen when determining the HPD region and, as seen in Fig. 3.2, the offsets need not be contiguous. Mathematically, we can express the $(1 - P_{ex})$ HPD region as the smallest set $\mathcal{S}$ such that

$$\sum_{\eta \in \mathcal{S}} p_{\text{uncoded}}(\eta|\mathbf{y}) \geq 1 - P_{ex}, \tag{3.12}$$

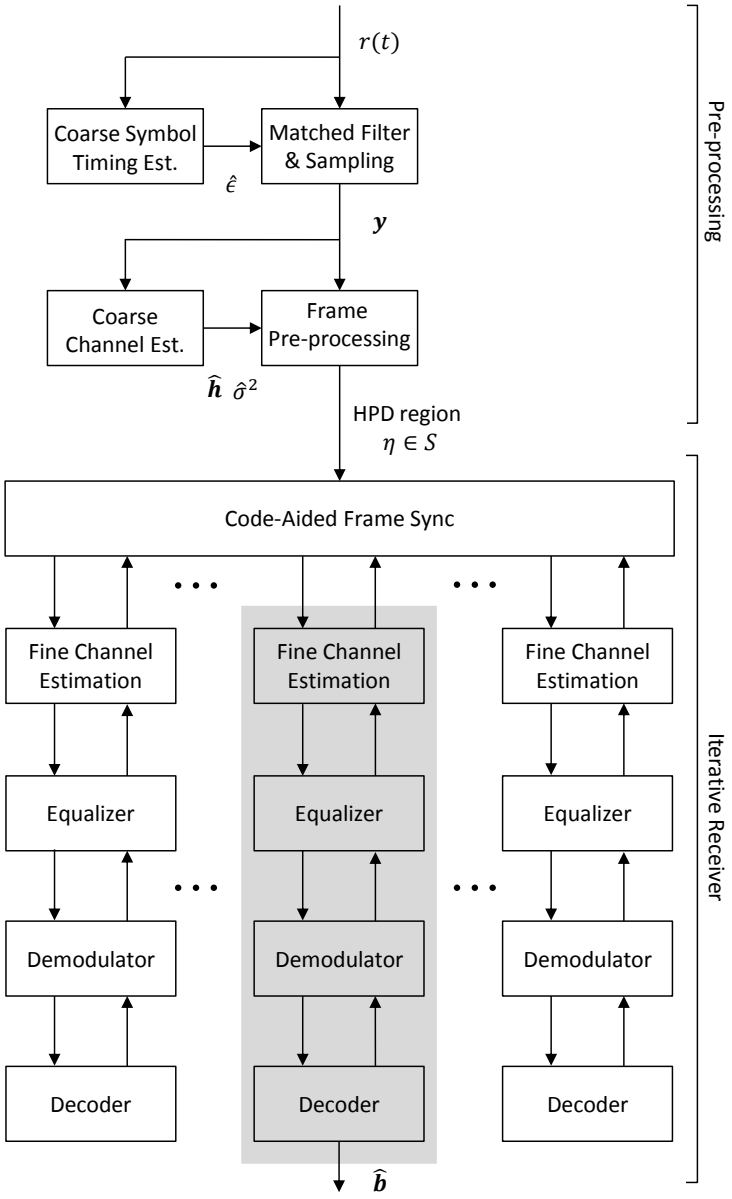

Figure 3.1: Proposed iterative receiver design for implementing code-aided frame synchronization with a pre-processing stage.

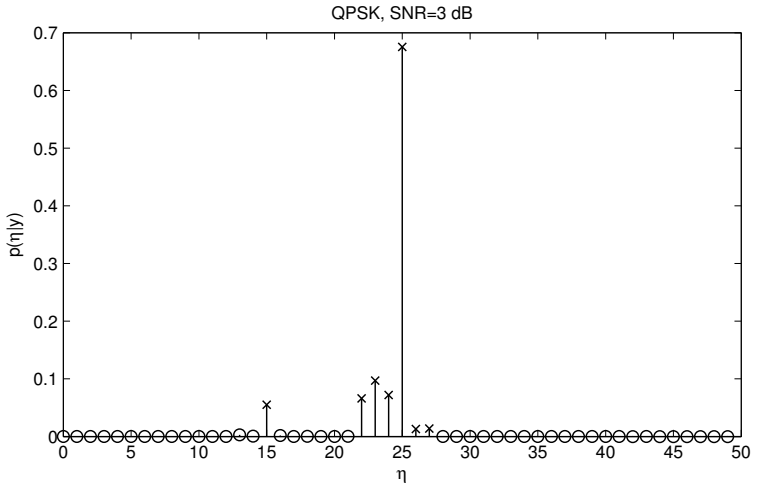

Figure 3.2: Example posterior distribution and 0.99 highest posterior density region (shown with 'x') with true offset $\eta = 25$.

where p_{uncoded} denotes the distribution under the *uncoded* signal model. If $|\mathcal{S}| > 1$, the frame offsets contained in the HPD region are passed to the code-aided algorithm in order to make a reliable decision.

Using the HPD region allows the receiver to choose on a frame by frame basis how many and which frame offsets to process with the code-aided method. Subsequently, the number of frame offsets processed by the code-aided method is minimized for a specified P_{ex}. Note that the frame synchronization error rate (FSER) of the receiver will be lower-bounded by P_{ex} which will in turn lower bound the achievable frame error rate (FER) of data detection.

In the MAP estimator given in (3.8), the *a priori* distribution of the frame offset $p(\eta)$ is assumed to be uniform over the domain $\eta \in \{0, \ldots, H-1\}$. The prior on the coded symbols $p(\mathbf{a})$ is non-zero and uniform for all symbol sequences which correspond to valid codewords. However, under the assumption of iid data symbols (i.e., the assumption of an uncoded signal) the prior reduces to $\prod_{i=0}^{N_a-1} p(a_i)$ where $p(a_i)$ is assumed to be a uniform distribution. In the case of an AWGN channel (given by (3.5) with $L = 1$) the assumption of the uncoded signal model simplifies the posterior distribution to

$$p_{\text{uncoded}}(\eta|\mathbf{y}) \propto \prod_{i=0}^{N_s-1} \exp\left\{\frac{2}{\sigma^2}\Re\left[y_{i+\eta}^* h_0 s_i\right]\right\} \cdot \prod_{i=0}^{N_a-1} \sum_{a_i} \exp\left\{\frac{2}{\sigma^2}\Re\left[y_{i+\eta+N_s}^* h_0 a_i\right] - \frac{1}{\sigma^2}|h_0 a_i|^2\right\}, \tag{3.13}$$

where the computational complexity is $O(N_s + N_a M)$.

In the case of the multipath channel, the assumption of an uncoded signal model reduces the complexity of computing the posterior distribution to $O(N_s L + N_a M^L)$. The complexity is still

exponential in the number of channel taps due to the second term in the likelihood function of (3.10): $\left|\sum_{l=\alpha_i}^{\beta_i} h_l x_{i-l}\right|^2$. By dropping these terms from the likelihood function and assuming an uncoded signal model, the complexity is further reduced. Although these terms are not functions of the frame offset η, they cannot be factored from the likelihood function because they appear within the summation over $\mathbf{x}$. Thus, their removal results in the following approximation of the likelihood function[2]:

$$p(\mathbf{y}|\mathbf{x},\eta) \overset{\propto}{\sim} \prod_{i=0}^{K+L-2} \exp\left\{\frac{2}{\sigma^2}\Re\left[y_{i+\eta}^* \sum_{l=\alpha_i}^{\beta_i} h_l x_{i-l}\right]\right\}. \tag{3.14}$$

A similar approximation is used in [135] for frame synchronization of uncoded signals. Substituting (3.14) into the posterior distribution for uncoded signals and grouping the intersymbol interference (ISI) terms by symbol leads to the following approximate posterior distribution:

$$\begin{aligned} p_{\text{uncoded}}(\eta|\mathbf{y}) \overset{\propto}{\sim} & \prod_{i=0}^{N_s-1} \exp\left\{\frac{2}{\sigma^2}\Re\left[s_i \sum_{l=0}^{L-1} y_{i+l+\eta}^* h_l\right]\right\} \\ & \cdot \prod_{i=0}^{N_a-1} \sum_{a_i} \exp\left\{\frac{2}{\sigma^2}\Re\left[a_i \sum_{l=0}^{L-1} y_{i+l+\eta+N_s}^* h_l\right]\right\}, \end{aligned} \tag{3.15}$$

with computational complexity $O(N_s L + N_a M L)$. In the section that follows, (3.15) is shown to provide a good approximation to the posterior distribution for the purpose of computing the HPD region.

3.5.2 HPD Region Characterization

We desire to characterize the HPD region in order to evaluate the impact that the pre-processing stage has on the complexity of the receiver. The code-aided algorithms will process each frame offset in the HPD region. Thus, understanding the expected size of the HPD region provides insight into the complexity of the receiver. We also benchmark the performance by comparing the probability of exclusion P_{ex} of the HPD region with the FSER of the MAP estimate under the uncoded signal model (from now on referred to as the *uncoded MAP estimate*).

The mean number of frame offsets in the HPD region is determined through Monte Carlo simulation. We consider BPSK modulation and m-sequences of length 7, 15, 31, and 63 for the sync word. Since P_{ex} provides a lower bound on the FER of the receiver, we set $P_{ex} = 10^{-3}$ with the perspective that FERs of 10^{-2} to 10^{-3} are typical for wireless communication systems in multipath channels. In the case of multipath, the results are averaged over realizations of a $L = 4$ tap channel drawn from complex Gaussian random variables where the relative power in each tap is given by $[0.644, 0.237, 0.087, 0.032]$. The results are presented with respect to the instantaneous SNR as defined in (3.6).

[2]We use $\overset{\propto}{\sim}$ to denote that the right hand side is proportional to an approximation of the likelihood function.

In the results that follow, the HPD region is computed using the posterior distribution in (3.13) for AWGN channels and the approximate posterior distribution in (3.15) for multipath channels. The mean HPD region size for a probability of exclusion of $P_{ex} = 10^{-3}$ is shown for AWGN and multipath in Figs. 3.3 and 3.4, respectively. The HPD region is compared to list synchronization [14, 126] in Figs. 3.3 and 3.4 by determining, through simulation, the required size of a fixed size list to achieve the same target P_{ex}.

Remark 1. Using the HPD region reduces the complexity by a factor of 2-3 over list based synchronization and can guarantee a P_{ex} value without requiring simulations to tune the list size.

The FSER of the uncoded MAP synchronizer and the P_{ex} achieved[3] by the HPD region are compared in Figs. 3.5 and 3.6 for AWGN and multipath channels, respectively. The uncoded MAP estimate is found by taking the maximum over the posterior distributions in (3.13) and (3.15) for AWGN and multipath, respectively. Because the channel is known, we expect the performance in the AWGN channel and the multipath channel to be similar. However, for short sync word lengths, the use of the approximate likelihood function for the multipath channel leads to a higher P_{ex} and FSER and a larger required fixed list size. It is seen that the size of the HPD region is also affected by this approximation although the impact on the achieved P_{ex} in the multipath channel is small.

Remark 2. In Fig. 3.6, there is only a slight degradation in the P_{ex} achieved by the HPD region (i.e., P_{ex} is slightly greater than 10^{-3}) due to the approximation made in the likelihood function for multipath channels. Thus, the approximation is useful for the purposes of the HPD region-based pre-processing stage.

Remark 3. As SNR increases, the P_{ex} of the HPD region improves below the target value. Since the uncoded MAP estimate is always included in the HPD region, the FSER of the uncoded MAP synchronizer provides an upper bound on the P_{ex}—although in these simulations the P_{ex} maintains an order of magnitude or more improvement.

Remark 4. There is a sharp transition in the HPD region size as a function of SNR. For each sync word length, the transition occurs at an SNR which is lower than that required for the uncoded MAP synchronizer to achieve the target P_{ex}.

We observe in Figs. 3.3 and 3.4 that a shorter sync word length increases the size of the HPD region. When the length of the sync word is reduced by (approximately) half, there is about a 2 dB loss for the mean size of the HPD region (see Figs. 3.3 and 3.4), while there is a 3 dB loss in the uncoded MAP synchronizer's performance (see Figs. 3.5 and 3.6). Thus, possible gains from code-aided frame synchronization versus conventional synchronization are greatest for short sync word lengths.

We conclude that the HPD region provides a useful tool in minimizing the complexity of the proposed iterative receiver. By comparing the HPD region size with the FSER of the uncoded MAP estimate, we can begin to identify a range of SNR over which the complexity of the iterative

[3] The achieved P_{ex} is different from the target P_{ex} due to the discrete number of frame offsets in the HPD region and the use of an approximate likelihood function in the case of multipath.

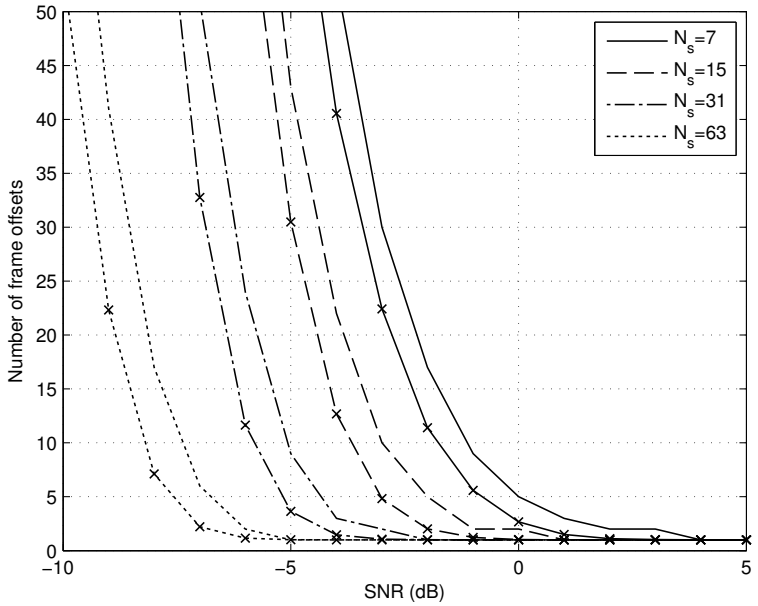

Figure 3.3: Mean number of frame offsets in the $(1-10^{-3})$ HPD region ("x" marker) and in a fixed size list (no marker) for an AWGN channel.

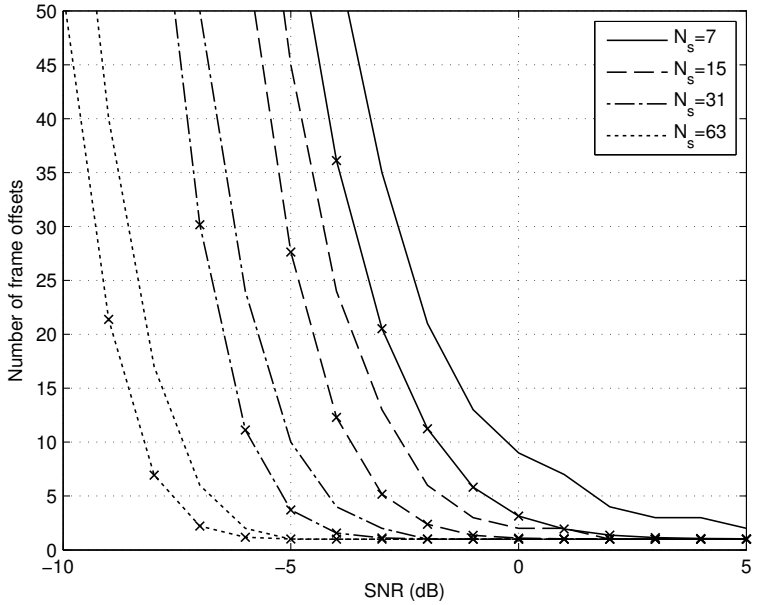

Figure 3.4: Mean number of frame offsets in the $(1-10^{-3})$ HPD region ("x" marker) and in a fixed size list (no marker) for a multipath channel.

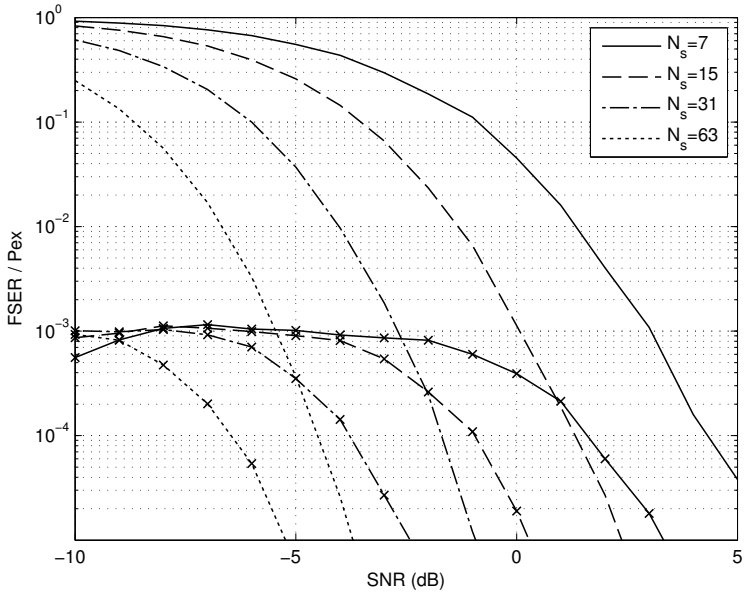

Figure 3.5: Comparison between the FSER of the uncoded MAP estimate (no marker) and the achieved P_{ex} of the $(1 - 10^{-3})$ HPD region ("x" marker) for an AWGN channel.

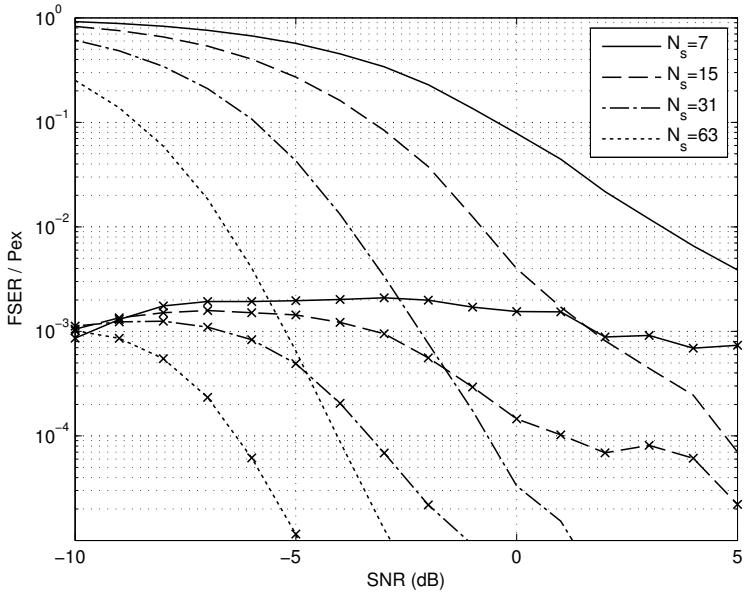

Figure 3.6: Comparison between the FSER of the uncoded MAP estimate (no marker) and the achieved P_{ex} of the $(1 - 10^{-3})$ HPD region ("x" marker) for a multipath channel.

receiver is reasonable and the performance of a conventional receiver based on uncoded MAP synchronization does not reach the target. We will return to this idea in Section 3.7 after considering the complexity of the code-aided algorithms.

3.6 Code-Aided Synchronization

After the pre-processing stage, $\mathcal{S}$ (the set of frame offsets in the HPD region) is conveyed to the code-aided frame synchronization stage as shown in Fig. 3.1. This stage initiates code-aided frame synchronization (as described in this section) for each of the frame offsets in the HPD region as shown by the duplicate receiver chains in Fig. 3.1. While the code structure is ignored in the frame pre-processing stage, it is utilized in this stage in order to make the final frame offset estimate. After this estimate has been made, further processing is executed on the receiver chain corresponding to the selected frame offset in order to estimate the information bits.

In this section, we present the following code-aided frame synchronization methods for the multipath channel:

- maximization of the syndrome posterior probability (SPP) of the code parity checks and
- approximate MAP hypothesis testing motivated by the EM algorithm.

3.6.1 SPP Algorithm

One of the properties of a linear code is the existence of a parity check matrix $\mathbf{H}$ which defines the dependencies between the coded bits. A binary sequence $\mathbf{c}$ is a valid codeword if it satisfies the parity check relations or, in other words, when the syndrome of the operation $\mathbf{Hc}$ is the all zeros vector. There are $N_p = N_c - N_b$ parity check relations represented by the rows of $\mathbf{H}$. Let the number of nonzero elements in the ith row of $\mathbf{H}$ be given by J_i and let $\phi_i(j)$ for all $j = 1, 2, \ldots, J_i$ provide the index of the jth nonzero element in the ith row of $\mathbf{H}$. The ith parity check relation is written as given by

$$c_{\phi_i(1)} \oplus c_{\phi_i(2)} \oplus \cdots \oplus c_{\phi_i(J_i)} = 0, \tag{3.16}$$

where $\oplus$ denotes modulus two addition.

The presence of the frame at an offset of η implies that the code structure is present in the received samples at this offset. Let $E_i(\eta)$ denote the event that the ith parity check is satisfied when the frame delay is given by η. Rather than approach the MAP estimator by applying Bayes' rule as in Section 3.3, the posterior probability of η is equated to the joint probability that the parity checks are satisfied at an offset of η [11, 133]. Mathematically, this is expressed as given by

$$p(\eta|\mathbf{y}) = \mathrm{P}\left[\bigcap_{i=1}^{N_p} E_i(\eta)\middle|\mathbf{y}\right]. \tag{3.17}$$

An assumption of independence between the events is made as given by

$$p(\eta|\mathbf{y}) \approx \prod_{i=1}^{N_p} \mathrm{P}\left[E_i(\eta)|\mathbf{y}\right] \tag{3.18}$$

which has similarly been made by [11, 133].

The posterior probability of parity check $E_i(\eta)$ is given by

$$\begin{aligned}\mathrm{P}\left[E_i(\eta)|\mathbf{y}\right] &= \sum_{c_{\phi_i(1)}} \cdots \sum_{c_{\phi_i(J_i)}} \mathrm{P}\left[E_i(\eta), c_{\phi_i(1)}, \ldots, c_{\phi_i(J_i)}|\mathbf{y}\right] \\ &= \sum_{c_{\phi_i(1)}} \cdots \sum_{c_{\phi_i(J_i)}} \mathrm{P}\left[E_i(\eta)|c_{\phi_i(1)}, \ldots, c_{\phi_i(J_i)}\right] \cdot p(c_{\phi_i(1)}, \ldots, c_{\phi_i(J_i)}|\mathbf{y})\end{aligned} \tag{3.19}$$

where the posterior probability of the coded bits is approximated as a multiplication of the coded bit likelihoods

$$p(c_{\phi_i(1)}, \ldots, c_{\phi_i(J_i)}|\mathbf{y}) \approx \prod_{j=1}^{J_i} p(\mathbf{y}|c_{\phi_i(j)}). \tag{3.20}$$

Converting these operations into the log domain, the logarithm of the posterior probability can be expressed as a function of the log-likelihood ratios (LLRs) of the coded bits. The logarithm of the event posterior probability is given by

$$\ln \mathrm{P}\left[E_i(\eta)|\mathbf{y}\right] = -\ln(1 + e^{-\gamma_i(\eta)}) \tag{3.21}$$

where $\gamma_i(\eta)$ is the log-likelihood ratio for the ith event $E_i(\eta)$. In this case, $\gamma_i(\eta)$ is given by the box-plus operation [139]

$$\gamma_i(\eta) = \lambda_{\phi_i(1)}(\eta) \boxplus \lambda_{\phi_i(2)}(\eta) \boxplus \cdots \boxplus \lambda_{\phi_i(J_i)}(\eta) \tag{3.22}$$

where $\lambda_{\phi_i(j)}(\eta)$ is the log-likelihood ratio for $c_{\phi_i(j)}$. Simplifications to the box-plus operation enable efficient computation of the syndrome probability without significant loss in performance [133, 139].

Substituting (3.21) into (3.18), we approximate the logarithm of the posterior event probability $\ln p(\eta|\mathbf{y})$ by

$$\Lambda_{\mathrm{SPP}}(\eta) = -\sum_{i=1}^{N_p} \ln(1 + e^{-\gamma_i(\eta)}) \tag{3.23}$$

and the estimate of the frame offset is given by

$$\hat{\eta}_{\mathrm{SPP}} = \arg\max_{\eta \in \mathcal{S}} \Lambda_{\mathrm{SPP}}(\eta). \tag{3.24}$$

The method requires computing the log-likelihood ratio of each coded bit in the presence of inter-symbol interference from multipath. In order to compute the coded bit log-likelihood ratios, we

begin by computing the likelihood function of the symbols. In the case of an AWGN channel, the symbol likelihoods are given by

$$p(\mathbf{y}|a_i, \eta) \propto \exp\left\{-\frac{1}{\sigma^2}|y_{i+\eta+N_s} - h_0 a_i|^2\right\}. \tag{3.25}$$

In the case of a multipath channel the computation of the symbol likelihoods can be performed by using the BCJR algorithm [137]. However, to keep the complexity of this method low, we propose a linear minimum mean square error (MMSE) equalizer followed by computation of the symbol likelihoods given the equalizer output. MMSE equalization is performed once and the output is used for all frame offsets in the HPD region. Let the output of the equalizer be given by $\hat{x}_{\eta+k}$ for all $k = 0, \ldots, K-1$ and $\eta \in \mathcal{S}$. The symbol likelihoods are given by

$$p(\mathbf{y}|a_i, \eta) \approx p(\hat{x}_{i+\eta+N_s}|a_i) \propto \exp\left\{-\frac{1}{\sigma^2}|\hat{x}_{i+\eta+N_s} - a_i|^2\right\}. \tag{3.26}$$

Soft demodulation is performed to compute the coded bit LLRs from the symbol likelihoods. Let the function $\psi(i)$ return the index of the coded data symbol $a_{\psi(i)}$ corresponding to the ith coded bit. For example, $\log_2(M)$ coded bits including c_i are modulated to generate symbol $a_{\psi(i)}$. The coded bit log-likelihood ratio is given by

$$\lambda_i(\eta) = \ln \frac{\sum_{a_{\psi(i)} \in \mathcal{A}^0} p(\mathbf{y}|a_{\psi(i)}, \eta)}{\sum_{a_{\psi(i)} \in \mathcal{A}^1} p(\mathbf{y}|a_{\psi(i)}, \eta)} \tag{3.27}$$

where $\mathcal{A}^0$ and $\mathcal{A}^1$ denote subsets of the symbol alphabet for which $c_i = 0$ and $c_i = 1$, respectively.

In summary, the SPP algorithm is implemented by computing symbol estimates using an MMSE equalizer, the symbol likelihoods from (3.26), and the coded bit LLRs from (3.27). These operations are performed once to evaluate the coded bit LLRs necessary for all the frame offsets in the HPD region. The parity check LLRs for a given frame offset in $\mathcal{S}$ are computed from (3.22) using the coded bit LLRs corresponding to that offset. Finally, the SPP metric is computed from (3.23) and the maximum is taken as the frame offset estimate as shown in (3.24). A diagram of this procedure is shown in Fig. 3.7. Subsequently, the iterative receiver is used to recover the information bits at this frame offset.

3.6.2 EM-Based Algorithm

The EM algorithm [31] provides a means of iteratively estimating parameter η from incomplete data $\mathbf{y}$ when there exists a set of unobserved or missing data $\mathbf{x}$. Maximization is performed on the complete data $\mathbf{z} = [\mathbf{y}, \mathbf{x}]$ which typically simplifies computation. The EM algorithm is given by the following expectation (E) and maximization (M) steps:

$$\begin{aligned} &\text{E step: } Q(\eta, \hat{\eta}^{(p-1)}) = \int_{\mathbf{z}} p(\mathbf{z}|\mathbf{y}, \hat{\eta}^{(p-1)}) \ln p(\mathbf{z}|\eta) d\mathbf{z} \\ &\text{M step: } \hat{\eta}^{(p)} = \arg\max_{\eta} Q(\eta, \hat{\eta}^{(p-1)}), \end{aligned} \tag{3.28}$$

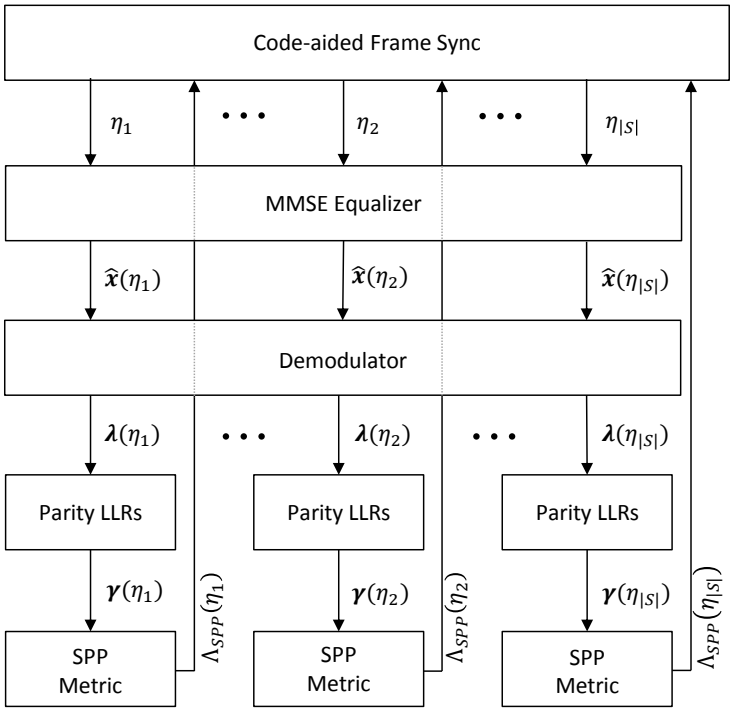

Figure 3.7: SPP code-aided frame synchronization block diagram. The HPD region is labeled $\eta_1, \ldots, \eta_{|\mathcal{S}|}$. The vectors $\hat{\mathbf{x}}(\eta_i)$, $\boldsymbol{\lambda}(\eta_i)$, and $\boldsymbol{\gamma}(\eta_i)$ are the symbol estimates, coded bits LLRs, and parity check LLRs, respectively, for frame offset η_i.

where p is the iteration number. However, the EM algorithm is not equipped to handle discrete parameters which has motivated a modification proposed by Wymeersch *et al.* where the expectation step is computed for all discrete values [13]. The EM-motivated algorithm is given a more rigorous mathematical explanation in [128] where it is shown to be an approximation to MAP hypothesis testing.

The discrete EM method is formulated for the frame synchronization problem as follows:

$$\hat{\eta}_{\text{EM}} = \arg\max_{\eta \in \mathcal{S}} Q(\eta), \tag{3.29}$$

where

$$Q(\eta) = \sum_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}, \eta) \ln p(\mathbf{y}|\mathbf{x}, \eta). \tag{3.30}$$

As given in (3.29), maximization is performed over all frame offsets in the HPD region $\mathcal{S}$. The EM motivated algorithm computes the expectation of the log-likelihood function with respect to the data symbols rather than performing marginalization over the data symbols as required in the MAP estimator given in (3.8).

In the case of an AWGN channel, the log-likelihood $\ln p(\mathbf{y}|\mathbf{x}, \eta)$ from (3.30) can be simplified as given by

$$\ln p(\mathbf{y}|\mathbf{x}, \eta) \propto \sum_{i=0}^{K+L-2} \frac{2}{\sigma^2} \Re\left[y_{i+\eta}^* h_0 x_i\right] - \frac{1}{\sigma^2} |h_0 x_i|^2 . \tag{3.31}$$

Substituting (3.31) into (3.30) and rearranging the summations provides the following expression for the expectation step:

$$Q(\eta) = \sum_{i=0}^{K+L-2} \frac{2}{\sigma^2} \Re\left[y_{i+\eta}^* h_0 \sum_{x_i} x_i^* p(x_i|\mathbf{y}, \eta)\right] - \frac{1}{\sigma^2} |h_0|^2 \sum_{x_i} |x_i|^2 p(x_i|\mathbf{y}, \eta). \tag{3.32}$$

We observe that the EM framework has reduced the summation over coded symbol sequences in (3.30) to symbol-by-symbol summations.

In the case of ISI from a multipath channel, the log-likelihood is given by

$$\begin{aligned} \ln p(\mathbf{y}|\eta, \mathbf{x}) &\propto \sum_{i=0}^{K+L-2} \frac{2}{\sigma^2} \Re\left[y_{i+\eta}^* \left(\sum_{l=\alpha_i}^{\beta_i} h_l x_{i-l}\right)\right] - \frac{1}{\sigma^2} \left|\sum_{l=\alpha_i}^{\beta_i} h_l x_{i-l}\right|^2 \\ &\propto \frac{2}{\sigma^2} \Re\left[\sum_{i=0}^{K-1} x_i \sum_{l=0}^{L-1} y_{i+l+\eta}^* h_l\right] - \frac{1}{\sigma^2} \sum_{i=0}^{K+L-2} \sum_{l_1=\alpha_i}^{\beta_i} \sum_{l_2=\alpha_i}^{\beta_i} h_{l_1} h_{l_2}^* x_{i-l_1} x_{i-l_2}^*. \end{aligned} \tag{3.33}$$

Substituting (3.33) into (3.30), evaluation requires marginal posterior probabilities of the symbols $p(x_i|\eta, \mathbf{y})$ as well as pairwise joint posterior probabilities $p(x_i, x_j|\eta, \mathbf{y})$. Thus, the complexity has been reduced to being linear in the codeword length as a result of the EM-motivated approximation.

The sync word is naturally included in this method where the posterior probabilities associated with the sync symbols are unit impulses.

The posterior probabilities for the coded data symbols are obtained from the decoder. Thus, the EM-based method requires decoding all frame offsets in the HPD region. The receiver performs iterative MAP equalization and decoding based on the sum-product algorithm. Marginal and pairwise joint symbol posterior probabilities are estimated using the beliefs from the sum-product algorithm. Code-aided estimation of the other channel parameters may also be incorporated into the receiver framework as shown in Fig. 3.1. A viable alternative to the MAP equalizer is turbo equalization as proposed in [95] which reduces the computational complexity for large L.

3.7 Numerical Results

3.7.1 Receiver Performance

Simulations are presented which quantify the performance and complexity of the frame synchronization methods. The frame is constructed from a 15 symbol m-sequence and a 1/4-rate LDPC codeword of length 2164 symbols. The modulation is BPSK and the target P_{ex} is set to 10^{-3}. The multipath channel has $L = 4$ taps with coefficients drawn from complex Gaussian random variables as described in Section 3.5.2. In this section, we quantify the performance and complexity of the receiver under the assumption that all channel parameters other than the frame offset are known. In the next section, we will relax this assumption and include estimation of all channel parameters.

In Fig. 3.8, the FSER performance of the receiver is shown for the SPP and EM-based code-aided methods. The SPP frame synchronizer is based on the evaluation of (3.23) and decision of (3.24). The EM-based frame synchronizer is based on evaluation of (3.33), substituted into (3.30), and the decision of (3.29). Comparison is also made to a conventional receiver which performs frame synchronization with the uncoded MAP estimator, i.e., the use of (3.15) to approximate the argument of (3.8). In the case of the EM-based method, the receiver performs two iterations of the sum-product algorithm for each frame offset. The EM-based algorithm makes use of the sync word in addition to the code structure. Thus, the performance of this method converges to the uncoded MAP synchronization (i.e., conventional) performance at low SNR.

Although further iterations of the sum-product algorithm are required in order to reliably detect the data, good *synchronization* performance is seen with just two iterations for the EM-based algorithm. The SPP method is less complex than the EM-based method, yet its performance is comparable (at moderate to high SNR). Due to the limitations of linear equalizers, if the MMSE equalizer output is used for data detection (i.e., decoding), the performance is poor. For example, losses of 3-5 dB are shown for the MMSE linear equalizer compared to a MAP equalizer in a non-iterative case [95]. However, when the MMSE linear equalizer is used for frame synchronization in the SPP method, a loss of 1 dB or less is observed with respect to the EM-based method.

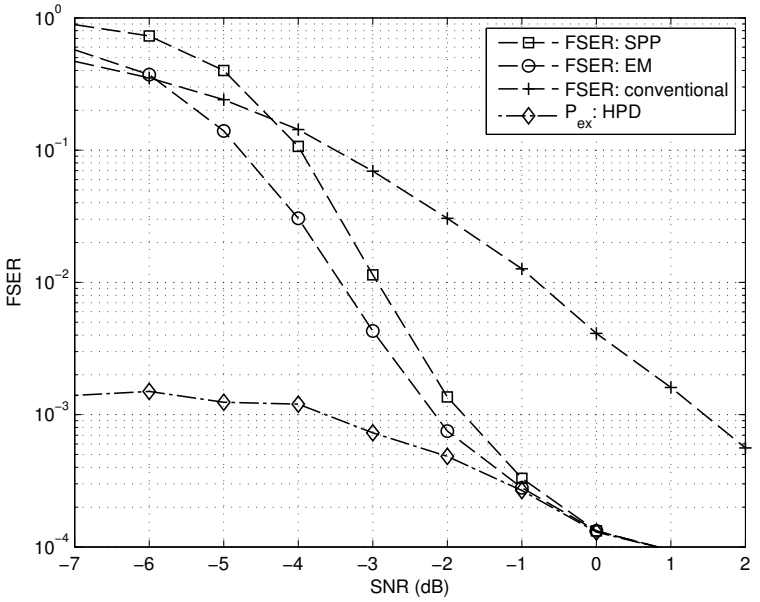

Figure 3.8: FSER vs. instantaneous SNR of the code-aided frame synchronization methods for a multipath channel.

The FER performance of the receiver with the EM-based synchronization method is compared to that of conventional (uncoded MAP) synchronization in Fig. 3.9. The FER in perfect synchronization is based on iterative MAP equalization and decoding with the sum-product algorithm. We observe in Fig. 3.9 that the FER of the proposed receiver is within 0.3 dB of the ideal performance at a FER of 10^{-3}. Further, the proposed receiver achieves a performance gain of about 3 dB over the conventional receiver. We observe in Fig. 3.9 that the frame synchronization performance of the EM-based method is not the limiting factor in the FER performance of the receiver. The same is not true of the conventional receiver. This performance improvement is achieved by computing the expectation in (3.30) for each frame offset in the HPD region. Therefore, an understanding of the complexity can be gained by observing the mean number of frame offsets in the HPD region for $N_s = 15$, found in Fig. 3.4. For example, at SNR $= -3$ dB, the mean HPD region size is $|\mathcal{S}| = 5.2$.

3.7.2 Complexity

The complexity of the SPP method and of a single iteration of decoding are both $\mathcal{O}(N_p)$. In the case of the EM-based method, two iterations of the sum-product algorithm were performed in order to compute the required posterior probabilities. As a benchmark, we consider a conventional receiver which processes a single frame offset and performs 20 iterations of the sum-product algorithm for data detection. Let Z denote the complexity of a single iteration of the sum-product algorithm based equalizer, demodulator, and decoder. Thus, $20Z$ is the complexity of the con-

ventional receiver. The complexity of the SPP-based receiver is $(|\mathcal{S}| + 20)Z$ and the complexity of the EM-based receiver is $(2|\mathcal{S}| + 20)Z$. Expressed as a multiple of the conventional receiver's complexity, the SPP and EM complexities are $\frac{|\mathcal{S}|}{20} + 1$ and $\frac{|\mathcal{S}|}{10} + 1$, respectively. For example, for an HPD region size of $|\mathcal{S}| = 5$, the complexity of the SPP method is 1.25× the complexity of the conventional receiver and the complexity of the EM-based method is 1.5× the complexity of the conventional receiver. The complexity of the proposed receiver is shown in Fig. 3.10 as a multiple of the conventional receiver's complexity. We observe that the complexity of the receiver based on each of the code-aided methods is very reasonable, especially in the operating region of the code. For example, when the FER of the iterative receiver approaches the target of 10^{-3} at an SNR of -2 dB, the complexity is a factor of 1.1× and 1.2× for the SPP and EM-based methods, respectively.

3.7.3 Low-SNR Synchronization Performance

In a wireless communication system we desire to support ACK/NACK and hybrid ARQ protocols even when the frame cannot be decoded. This goal motivates improving frame synchronization at low SNR values. For example, in a hybrid ARQ system with soft combining, additional parity bits can be transmitted when the initial message was not successfully decoded. In order to combine the additional parity bits with the original message, frame synchronization must be obtained even when the frame is not decoded properly. In Fig. 3.11, the FSER of the SPP code-aided method is shown for five codeword lengths in both AWGN and multipath channels. In this simulation, the code is a 1/2-rate LDPC code, the modulation is BPSK, and the frame is transmitted without a sync word. As a point of reference, the channel capacity for binary signaling in AWGN is shown in Fig. 3.11. In either channel, every time the codeword length is doubled, the performance of the synchronization algorithm improves by about 1 dB. The results demonstrate the ability of the code-aided method to perform synchronization below the SNR required for recovery of the information bits. Even when accurate decoding is not possible, if a sufficient number of parity checks are likely to be satisfied at the true offset, accurate frame synchronization is possible.

3.7.4 Value of Code-Aided Synchronization

As discussed in Section 3.7.2, the complexity of the proposed receiver depends on the size of the HPD region as well as the computational requirements of the chosen frame synchronization method. Now that we have considered the complexity of the code-aided methods, we interpret the results in Figs. 3.3 and 3.4 in light of their impact on the complexity and value of the proposed receiver. We consider the proposed receiver to provide value over the conventional receiver when (a) the proposed receiver achieves the target performance while the conventional does not and (b) the complexity of the proposed receiver is within a factor of 2× the complexity of the conventional receiver.

The conventional receiver performs 10 and 20 decoder iterations in the AWGN and multipath channels, respectively. For the EM-based code-aided method, one and two iterations are performed in

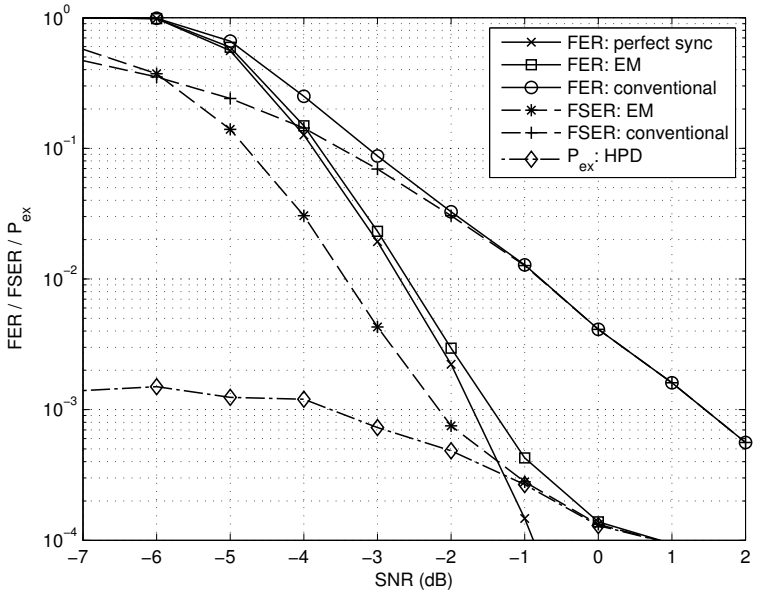

Figure 3.9: FER vs. instantaneous SNR of the proposed receiver with the EM-based code-aided frame synchronization method for a multipath channel. Note that E_b/N_0 = SNR + 6 dB.

the AWGN and multipath channels, respectively. The value of SNR at which the complexity falls below a factor of 2 (with respect to a conventional receiver) is recorded in Table 3.1. In addition, the value of SNR at which the FSER of the uncoded MAP estimator reaches the target performance (10^{-3}) is recorded in Table 3.1. Even when the conventional receiver reaches the target performance, the code-aided methods may continue to provide an improvement in performance as observed in Figs. 3.5 and 3.6, but additional gains might not be needed.

Remark 5. Table 3.1 provides regions of SNR where the code-aided methods outperform conventional frame synchronization while maintaining a reasonable complexity. For a given amount of overhead, due to the sync word, the proposed receiver is capable of operating at an SNR which is anywhere from 2.8 to 10.2 dB below that of the conventional receiver.

When comparing the proposed receiver to the conventional receiver, the most substantial gains are observed in the multipath channel. Since, the conventional receiver estimates the frame offset using the uncoded signal model, the approximation made in the likelihood function in (3.15) has a more pronounced impact on the performance. As the length of the sync word increases, the presence of multipath in the channel does not substantially impact either receiver. We see that the proposed receiver begins to behave very similarly in AWGN and multipath channels as the sync word length increases. With the length 63 m-sequence, the proposed receiver remains capable of providing a significant performance improvement over the conventional receiver ($\geq$ 2.8 dB). The choice of sync word impacts the size of the HPD region and determines the regions of SNR over which the proposed receiver provides value, as given in Table 3.1. The choice of code rate and modulation will determine the desired operating SNR of the receiver and thus will also indirectly impact the complexity of the proposed receiver.

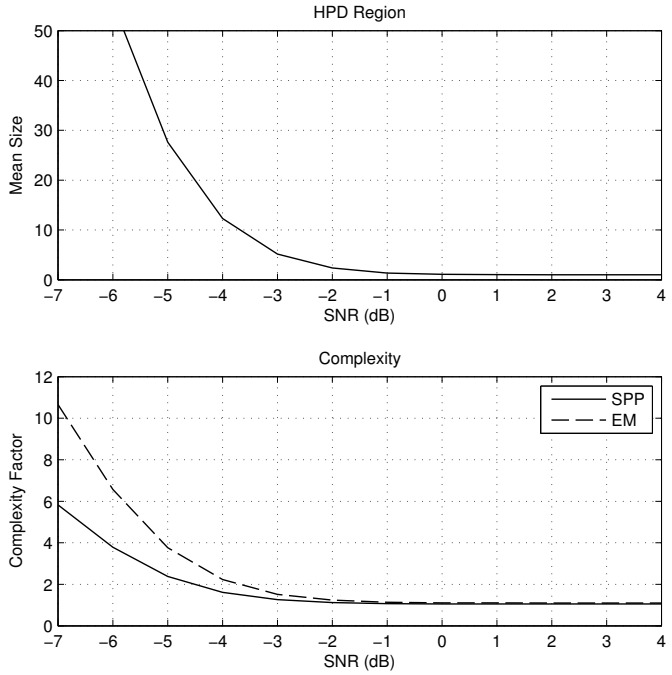

Figure 3.10: Number of frame offsets in the HPD region and the resulting complexity of the proposed SPP and EM-based receivers (relative to the conventional receiver) for a multipath channel.

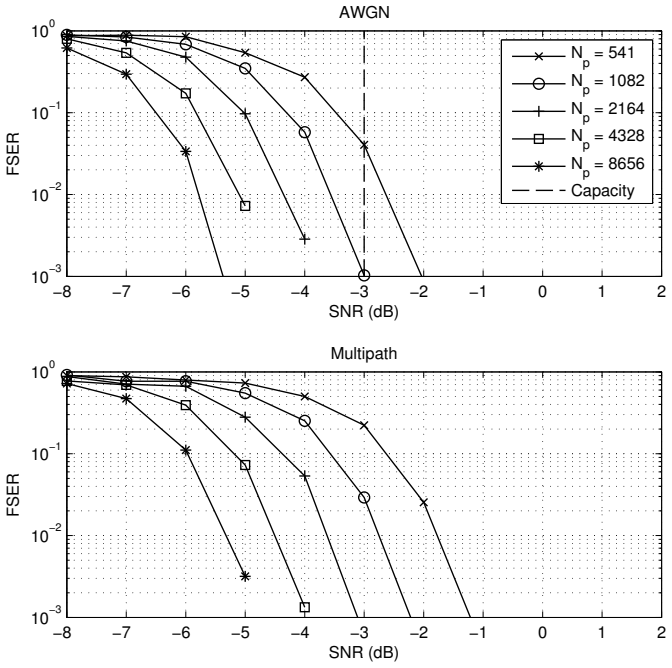

Figure 3.11: FSER of the SPP code-aided frame synchronization method for five codeword lengths in an AWGN channel and a multipath channel ($\mathbf{h} = [\,-0.09 - 0.53j,\ 0.60 - 0.34j,\ 0.17 + 0.20j,\ 0.01 - 0.42j\,]^{\mathrm{T}}$). Note that E_b/N_0 = SNR + 3 dB.

Table 3.1: Regions of value for the proposed receiver (values in dB)

AWGN Channel			
N_s	SPP & EM		Conventional
(m-seq. length)	(SNR for complexity $\leq 2\times$)		(SNR for FSER $\leq 10^{-3}$)
7	-1.8		3.0
15	-3.8		0.1
31	-5.9		-2.7
63	-8.3		-5.5
Multipath Channel			
N_s	SPP	EM	Conventional
(m-seq. length)	(SNR for complexity $\leq 2\times$)		(SNR for FSER $\leq 10^{-3}$)
7	-3.0	-1.8	7.2
15	-4.6	-3.8	1.7
31	-6.6	-5.9	-2.2
63	-8.9	-8.3	-5.2

3.8 Extension to an Unknown Channel

In this section we consider the case of unknown channel and noise parameters. The two purposes of this extension are (a) to integrate code-aided frame synchronization into an iterative receiver structure for synchronization, channel estimation, equalization, demodulation, and decoding and (b) to show the impact of imperfect channel information on the frame pre-processing stage. Before the frame pre-processing stage, non-data-aided (blind) channel estimation is used to obtain coarse estimates of the channel and noise parameters.

Due to the rotational symmetry of common digital phase-amplitude modulations, a phase ambiguity will be unresolved by blind channel estimation. Therefore, the estimated channel taps will be equal to the true channel taps rotated by the phase ambiguity plus an error term as given by

$$\hat{\mathbf{h}} = \mathbf{h}e^{j\frac{2\pi}{\Psi}\psi} + \mathbf{e}, \tag{3.34}$$

where $\hat{\mathbf{h}}$ is the estimated channel, Ψ is the number of rotational symmetries in the modulation, $\psi \in \{0, \ldots, \Psi - 1\}$ is the phase ambiguity, and $\mathbf{e}$ is the estimation error. For phase shift keying (PSK) constellations, Ψ is equal to the modulation order and for quadrature amplitude modulation (QAM) constellations $\Psi = 4$.

The iterative receiver structure presented in Fig. 3.1 is used to perform joint estimation of the

channel, symbol timing, frame timing, and noise power and detection of the information bits. The receiver performs the following functions: a) blind coarse estimation of the continuous parameters (channel gain, phase offset, symbol timing, and noise power), b) computation of the HPD region to select the frame offsets processed by the iterative receiver, c) EM-based hypothesis testing of the discrete parameters (phase ambiguity and frame offset), d) EM fine estimation of the continuous parameters, and e) posterior symbol and bit probability computation using the sum-product algorithm. While we focus on the EM-based code-aided method in this section, a complementary receiver can be designed with the SPP algorithm.

3.8.1 Iterative Receiver Structure

Each function in the receiver is briefly described in the following:

a) *Coarse estimation*: We use blind estimators to obtain initial, coarse estimates of the channel coefficients $\mathbf{h}$, symbol timing ϵ, and noise power σ^2 as shown in the "Coarse Symbol Timing Est." and "Coarse Channel Est." blocks of Fig. 3.1. The Oerder and Meyr (O&M) timing detector [140], moment-based amplitude, phase, and noise power estimators for AWGN [141], and the blind multipath channel estimators of [142] are examples of potential coarse estimators.[4]

b) *HPD region pre-processing*: The computation of the HPD region is conditioned on the coarse estimates $\{\hat{\mathbf{h}}, \hat{\epsilon}, \hat{\sigma^2}\}$ in an analogous way to the quasi hybrid likelihood ratio approach taken for detection and classification problems [141]. The phase ambiguity may be handled in the pre-processing stage in one of two ways. The approach taken in our work is to perform marginalization over the phase ambiguity when computing the (uncoded) posterior distribution for the frame offset. This case is appropriate when there is no sync word present in the transmitted signal (which we consider in the results of this section). Alternatively, the HPD region may be computed for the joint (uncoded) posterior distribution of the frame offset and phase ambiguity. In this case, the "Frame Pre-processing" block of Fig. 3.1 is a pre-processing stage for both the frame and phase ambiguity.

c) *Code-aided hypothesis testing*: The code-aided frame synchronization block of the receiver in Fig. 3.1 is generalized to code-aided hypothesis testing. The EM-based code-aided method presented in Section 3.6.2 is applied to both discrete parameters: the frame offset and the phase ambiguity. This requires computation of (3.33) and the associated symbol posterior probabilities for each combination of frame offset $\eta \in \mathcal{S}$ and phase ambiguity $\psi \in \{0, \ldots, \Psi - 1\}$. The fine channel estimates, discussed in the following bullet, are used in the evaluation of the code-aided metric. Fine estimates of the continuous parameters are obtained *per* hypothesis, making this a composite hypothesis test [144].

[4]The effect of a carrier frequency offset can be handled in the proposed receiver structure in a similar manner. For instance, coarse frequency synchronization is performed in the pre-processing stage using a blind estimator [143]. If necessary, residual frequency offset is removed using a code-aided approach in the fine channel estimation stage [82].

d) *Fine channel estimation*: Fine estimation of the continuous parameters $\mathbf{h}$, ϵ, and σ^2 is performed for each hypothesis $\{\eta, \psi\}$ as represented by the "Fine Channel Estimation" blocks in Fig. 3.1. We make use of the ECM algorithm, a variant of the EM algorithm, to sequentially perform maximization over each parameter while conditioning on the remaining parameters [118]. Posterior symbol probabilities required for code-aided hypothesis testing are also used for fine estimation of the continuous parameters which enables the fine estimation stage to be efficiently implemented. Derivations for the ECM estimation equations in an AWGN channel are provided in Appendix A.

e) *Sum-product algorithm*: The expectation step of both the EM-based hypothesis test and the ECM algorithm requires computation of the marginal symbol posterior probabilities. Also, MAP bit-wise detection is performed using marginal posterior probabilities of the information bits. The sum-product algorithm performs efficient computation of these probabilities by applying message passing on a factor graph representation of the equalizer, demodulator, and decoder shown in Fig. 3.1.

3.8.2 Numerical Results

The performance of the proposed iterative receiver in the presence of an unknown AWGN channel is evaluated for a transmission length of 1000 information bits, 16QAM modulation, and a 1/2-rate turbo code ($N_a = 500$). No sync word is included in the transmission, requiring the receiver to perform training-less recovery ($N_s = 0$, $K = 500$). The unknown parameters are h_0, ϵ, σ^2, ψ, and η. Energy detection is used to find the maximum energy frame offset of the received signal, and the domain of the frame offset is set relative to this point. We found $H = 51$ to be sufficiently large to ensure that the true offset is always included in the frame offset characterization. In order to minimize complexity without significant performance loss, the target P_{ex} was set to be $1/10$th of the frame error rate achieved with perfect synchronization. For each hypothesis, 10 iterations of the ECM algorithm are performed, each of which includes a single iteration of the sum-product algorithm.

Simulation results of the FER are presented for four cases: (a) iterative demodulation and decoding with perfect synchronization (Perfect sync.); (b) hypothesis testing of ψ, η with no iterative estimation of the other parameters (HT); (c) the full iterative receiver with hypothesis testing and the ECM algorithm (HT/ECM); and (d) an iterative receiver with an uncoded MAP frame estimator (Uncoded MAP Frame Estimate). The performance of each case is shown in Fig 3.12. The iterative receiver demonstrates performance very near to that of perfect synchronization. In other words, the iterative receiver's ability to perform code-aided synchronization is on the same order as the decoder's ability to successfully decode the information in a known channel. Limiting the iterative receiver to only hypothesis testing results in a loss of 0.1-0.2 dB. The performance of the fourth case—considering only the uncoded MAP frame offset estimate—quickly becomes limited by frame synchronization errors.

In the iterative receiver, the frame offset posterior distribution is characterized using the coarse

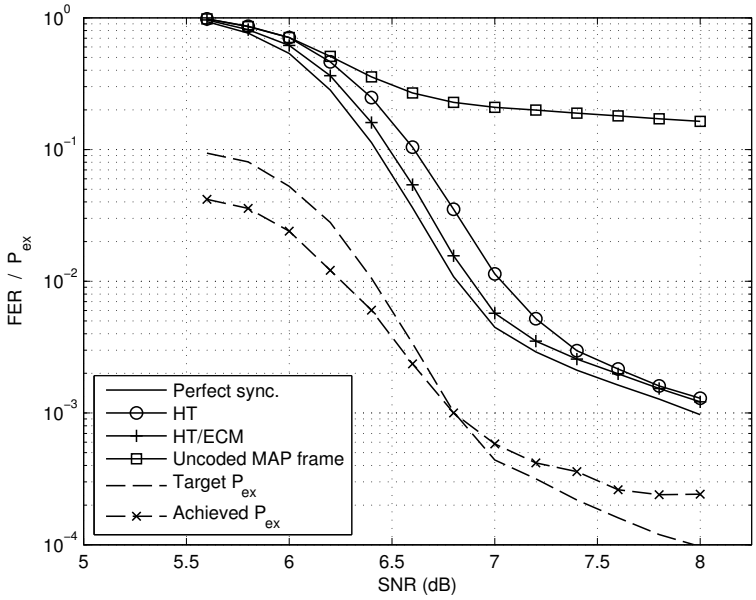

Figure 3.12: Results for the FER for Perfect sync. (solid), HT (solid,circle), HT/ECM (solid,plus), and Uncoded MAP Frame Estimate (solid,square) as well as the target P_{ex} (dash) and the achieved P_{ex} (dash,'x') for the pre-processing stage of the HT and HT/ECM receivers. Note that $E_b/N_0 =$ SNR $-$ 3 dB.

estimates of h_0, ϵ, and σ^2. In Fig. 3.12, the achieved P_{ex} for the iterative receiver performance simulation is shown. When the target P_{ex} is below 10^{-3}, we observe an increase in the probability of excluding the true frame offset from the HPD region as a result of errors in the coarse estimation stage, although the increase is slight.

Remark 6. The use of coarse estimates in the frame pre-processing stage slightly degrades the effectiveness of the HPD region. However, by setting the P_{ex} to 1/10th of the ideal FER, the decoder (rather than the HPD region) becomes the limiting factor in the receiver's performance.

To provide insight into limiting factors in the performance of the examined receivers, Table 3.2 lists the number of frame errors according to the source of the error at an SNR of 7 dB. The number of frame errors is shown by receiver type. Frame errors may originate from excluding the true frame offset from $\mathcal{S}$, choosing an incorrect hypothesis when the true offset has been included in $\mathcal{S}$, and incorrect detection of one or more information bits when the true hypothesis has been chosen. The first two cases (frame synchronization errors) generally lead to half the bits being in error (i.e., the bit error rate is significantly affected). The third case generally results in fewer bit errors.

In the pre-processing stage, the true offset was excluded from the HPD region in 30 simulations (out of a total of 47 683 total simulated frames), which affects both the HT and HT/ECM receivers. When the true offset is included, the HT/ECM receiver always selected the correct hypothesis (i.e., no frame synchronization errors) while the HT receiver chose an incorrect hypothesis in 32 simulations.

Remark 7. Fine estimation of the continuous parameters with the ECM algorithm enables more reliable estimation of the frame offset and phase ambiguity, and improves data detection.

Table 3.2: Breakdown of Frame Errors by the Source of the Error

		$\eta_{\text{true}} \notin \mathcal{S}$	$\eta_{\text{true}} \in \mathcal{S}$	
Receiver Type	Total No. Frame Errors[†]	Exclusion from HPD	Wrong Hypothesis	Parameter Error & Noise
Perfect Sync.	200	0	0	200
HT	521	30	32	459
HT/ECM	254	30	0	224

[†]The total number of frames in the simulation is 47 683.

3.9 Conclusion

In this chapter, we presented the development of (a) SPP and EM-based code-aided frame synchronization methods for multipath channels; (b) a novel frame pre-processing stage based on the HPD region; and (c) the HT/ECM iterative receiver structure which supports frame synchronization, channel estimation, equalization, demodulation, and decoding. Table 3.1 was a main result from the work where it was shown that the proposed receiver is capable of operating at an SNR from 2.8 to 10.2 dB below that of the conventional receiver while maintaining a complexity within 2× the conventional receiver's complexity. The performance of the proposed receiver was demonstrated in a scenario in which a gain of about 3 dB was achieved with a complexity increase of only 20%. Finally, the proposed receiver design was shown to reliably operate without prior knowledge of the channel and noise parameters. In this case, fine estimation of the continuous parameters in the HT/ECM receiver improved the performance of code-aided frame synchronization and data detection. Overall, code-aided frame synchronization is shown to provide value over conventional techniques while introducing a reasonable increase in complexity.

Chapter 4

Detection of Co-channel Signals in Non-Gaussian Noise

4.1 Introduction

Co-channel interference is a key factor limiting the capacity of densely deployed wireless communication systems. Continued network densification is expected with the introduction of heterogeneous networks [2, 3] and spectrum sharing [4]. In addition, wireless system architecture is expected to transition from a "cell-centric" model in which interference is centrally managed by the base station to a "device-centric" model [3]. For these reasons, the ability to detect a desired signal in the presence of co-channel interference is vitally important for current and future wireless systems. Receiver algorithms generally fit into one of three categories: linear filtering, interference cancellation, and joint detection.

The optimal approach to detection in co-channel interference is to jointly detect the desired and co-channel signals which can be accomplished by the joint maximum likelihood sequence estimator (JMLSE) [145] or the joint maximum *a posteriori* probability (JMAP) symbol-by-symbol detector [6, 137, 146], depending on the choice of optimality criterion. Both receiver structures are prohibitively complex which has motivated research into sub-optimal receiver algorithms including [91, 147–153], among many others.[1] Factor graphs and the sum-product algorithm (SPA) provide a theoretically justified approach to approximating the marginal posterior distributions required for JMAP detection [7, 8, 15]. Complexity is greatly reduced by passing local distributions (messages) along the edges of the factor graph. The factor graph approach generalizes and provides justification for algorithms which combine soft detection with single-user soft decoding. Yet, even within the factor graph model the receiver may be prohibitively complex as a result of high-order modulations, numerous users, or inter-symbol interference (ISI).

[1] For an overview of works on detection in interference the reader is referred to [154, 155] for single carrier systems, [156] for CDMA, [157] for multiuser MIMO, and [158] for array processing techniques.

A particularly challenging case is signal detection using a single receive antenna in the presence of both ISI and co-channel interference. The received signals do not possess a natural separability due to a multiple access technique or the use of multiple antennas. As a result, linear filtering and interference cancellation are ineffective, especially when the signal power levels are similar. Joint MAP detection in such a signal model is developed and studied in [16]. Signal separability is achieved due to both frame and symbol timing offsets and the presence of an error correction code. The JMAP detector of [16] is still an approximation to true JMAP detection as it only accounts for the strongest ISI components and separates detection from single-user soft decoding. Even then, the algorithm is exponentially complex in the number of co-channel signals, the considered ISI components, and the modulation order and in [16] simulations are limited to the detection of two users with BPSK modulation.

Jiang and Li consider single antenna detection of co-channel signals in the presence of ISI which results from multipath channels [42]. They propose a multiuser detection algorithm named concurrent MAP (CMAP) in which single-user MAP equalization is performed for the multipath channel and a Gaussian approximation is made for co-channel interference. The CMAP algorithm is shown to perform well compared to joint MAP detection, the Rake Gaussian method proposed in [159], and soft interference cancellation with MAP equalization [42]. However, the Gaussian assumption is a poor approximation for dominant multipath components from the co-channel interference. At the same time, CMAP remains exponentially complex in the number of multipath components. It also makes inefficient use of computational resources for weak multipath components. From these observations, we proposed a new approximate joint MAP (AMAP) detector which is motivated by a factor graph model of the received signal in our previous work [43]. The AMAP algorithm approximates the SPA messages by maintaining a discrete domain for the multipath components with the strongest received power and approximating the remaining weaker multipath components with a Gaussian distribution. The motivation behind the AMAP algorithm is to approximate the joint MAP detector as closely as possible within the allowable computational complexity of the receiver. The complexity of the algorithm is adjustable and can be set to account for the capabilities of the receiver, the desired performance, or the difficulty of the detection task.

A number of measurement studies report that the noise affecting wireless communication systems is often non-Gaussian—exhibiting an impulsive behavior [17–24]. Non-Gaussian noise may also be used to model co-channel signals with transmission properties which are unknown to the receiver and thus not part of the joint detection [25, 26]. Algorithms for detection in non-Gaussian noise are presented for single-user single-carrier transmissions in [23], for single-user OFDM transmissions in [30, 160, 161], and for multiuser CDMA systems in [27, 28]. In [28] Gibbs sampling is used to perform joint parameter estimation and data detection in a synchronous multiuser CDMA system affected by non-Gaussian noise. However, much of the literature on joint channel estimation and detection in co-channel interference assumes Gaussian noise (e.g., [32, 41, 162–164], among others). In this chapter, we consider detection in co-channel interference in the general case of ISI (similarly, inter-carrier interference in an OFDM system) and noise which is non-Gaussian. We develop a factor graph-based iterative receiver which has at its core the AMAP detector which we initially presented for Gaussian noise in [43]. In the proposed receiver,

noise is modeled with a two-term Gaussian mixture distribution which is well-justified [26] and commonly used for impulsive noise [27–30]. We also develop an iterative estimation stage based on the EM algorithm [31] to incorporate estimation of the channel coefficients and the Gaussian mixture model (GMM) parameters within the iterative receiver. The EM algorithm provides a theoretically justified approach to performing iterative joint estimation and detection in the receiver [58, 61, 82, 165].

The proposed receiver improves upon existing algorithms for detecting co-channel signals in the presence of ISI while also handling non-Gaussian noise. Applicable detection algorithms in the literature generally assume known channel parameters or perform channel estimation assuming Gaussian noise. In this work, we remove these assumptions and develop an iterative receiver which provides the following contributions:

- advances the state of the art by improving performance with lower complexity,
- operates effectively in the presence of Gaussian and non-Gaussian noise, and
- incorporates iterative channel and noise parameter estimation.

The chapter is organized as follows. The system model is presented in Section 4.2 followed by the development of the factor graph receiver in Section 4.3. In Section 4.4, we present our proposed approximate MAP detection algorithm based on the factor graph model. In Section 4.5, an iterative parameter estimation algorithm is presented to operate alongside the proposed detection algorithm. The message passing schedule of the combined SPA and EM algorithm receiver is discussed in Section 4.6. Numerical results are presented in Section 4.7 and conclusions are drawn in Section 4.8.

4.2 System Model

In this work we consider the task of detecting a desired signal from among a collection of U co-channel signals. Let $\mathbf{b}^{(u)}$ denote an $N_b \times 1$ vector of information bits corresponding to the uth signal. The information bits of each user are modeled as random variables assumed to be independent and identically distributed (iid) with uniform prior probabilities. The information bits of the uth signal are encoded with an error correction code to produce an $N_c \times 1$ vector of coded bits $\mathbf{c}^{(u)}$. Subsequently, the coded bits of the uth signal are modulated using a digital phase-amplitude modulation resulting in a complex symbol sequence which is denoted by

$$\mathbf{x}^{(u)} = \left[x_0^{(u)}, x_1^{(u)}, \ldots, x_{K-1}^{(u)}\right]^{\mathrm{T}}, \tag{4.1}$$

where K is the length of the transmission (assumed to be the same for all signals). The set of symbols in the modulation is given by $\{m_i\}_{i=1}^{M}$, where M is the order of the modulation, and is normalized to unit average power (i.e., $\frac{1}{M}\sum_{i=1}^{M}|m_i|^2 = 1$). We define the collection of the information

bits, coded bits, and symbols for all signals as

$$\mathbf{B} = [\mathbf{b}^{(1)}, \ldots, \mathbf{b}^{(U)}], \tag{4.2}$$

$$\mathbf{C} = [\mathbf{c}^{(1)}, \ldots, \mathbf{c}^{(U)}], \text{ and} \tag{4.3}$$

$$\mathbf{X} = [\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(U)}], \tag{4.4}$$

respectively. The signals are assumed to arrive at the receiver via a multipath channel which we model as a tapped delay line with the number of taps (i.e., multipath components) given by L. Thus, the nth sample of the received signal is given by

$$y_n = \sum_{u=1}^{U} \sum_{l=0}^{L-1} h_l^{(u)} x_{n-l}^{(u)} + w_n, \tag{4.5}$$

where $h_l^{(u)}$ denotes the lth multipath component of the uth signal and w_n represents noise. We define a vector $\mathbf{h}^{(u)}$ to denote the multipath channel for signal u as given by

$$\mathbf{h}^{(u)} = \left[h_0^{(u)}, h_1^{(u)}, \ldots, h_{L-1}^{(u)}\right]^{\mathrm{T}} \tag{4.6}$$

and the collection of the multipath channels for all signals as

$$\mathbf{h} = \left[\mathbf{h}^{(1)^{\mathrm{T}}}, \mathbf{h}^{(2)^{\mathrm{T}}}, \ldots, \mathbf{h}^{(U)^{\mathrm{T}}}\right]^{\mathrm{T}}. \tag{4.7}$$

The symbols of signal u which are present in sample y_n are collected into a vector as given by

$$\mathbf{x}_{[n]}^{(u)} = \left[x_n^{(u)}, x_{n-1}^{(u)}, \ldots, x_{n-L+1}^{(u)}\right]^{\mathrm{T}}, \tag{4.8}$$

and the collection of these terms for all signals is denoted by

$$\mathbf{x}_{[n]} = \left[\mathbf{x}_{[n]}^{(1)^{\mathrm{T}}}, \mathbf{x}_{[n]}^{(2)^{\mathrm{T}}}, \ldots, \mathbf{x}_{[n]}^{(U)^{\mathrm{T}}}\right]^{\mathrm{T}}. \tag{4.9}$$

This enables us to express the received signal in a condensed form as given by

$$y_n = \mathbf{h}^{\mathrm{T}} \mathbf{x}_{[n]} + w_n. \tag{4.10}$$

The collection of all received samples is denoted as $\mathbf{y} = [y_0, \ldots, y_{N-1}]^{\mathrm{T}}$, where $N = K + L - 1$. In (4.5) and (4.8) we define $x_i^{(u)} = 0$ for $i < 0$ or $i > K - 1$.

For the sake of notational simplicity, the model provided in (4.10) makes a number of assumptions about the received signal—for example, that the channel duration L is identical for all signals and that the received signal is sampled at a single sample per symbol. We will use the model of (4.10) for clarity of presentation throughout the chapter. However, the detection algorithm presented in this chapter is applicable to more general cases.

The noise is modeled as a random variable distributed according to a Γ-component Gaussian mixture model (GMM). The distribution of each mixture component is modeled as a circularly-symmetric complex Gaussian random variable. Therefore, the pdf of w_n is given by

$$p(w_n) = \sum_{i=1}^{\Gamma} \frac{\lambda_i}{\pi\sigma_i^2} \exp\left\{-\frac{|w_n|^2}{\sigma_i^2}\right\}, \tag{4.11}$$

where λ_i is the probability that w_n is distributed according to the ith mixture component and σ_i^2 is the variance of the ith mixture component. Since λ_i is a probability, we have the constraint $\sum_{i=1}^{\Gamma} \lambda_i = 1$. Additionally, w_n is assumed to be iid for all $n = 0, 1, \ldots, N-1$.

We define a random variable $q_n \in [1, \ldots, \Gamma]$ to indicate the mixture component from which w_n is generated. The vector of indicator variables is denoted by $\mathbf{q} = [q_0, \ldots, q_{N-1}]^{\mathrm{T}}$. The indicator variables are important for parameter estimation in Section 4.5.

In summary, the probabilistic model is comprised of random variables $\mathbf{B}$, $\mathbf{C}$, $\mathbf{X}$, and $\mathbf{q}$; observation $\mathbf{y}$; and model parameters $\mathbf{h}$, $\boldsymbol{\lambda}$, and $\boldsymbol{\sigma}^2$ where $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_\Gamma]^{\mathrm{T}}$ and $\boldsymbol{\sigma}^2 = [\sigma_1^2, \ldots, \sigma_\Gamma^2]^{\mathrm{T}}$.

4.3 Factor Graph of the Joint Distribution

The goal of the receiver is to detect the information bits $\mathbf{B}$ given observation of $\mathbf{y}$. To that end, we desire to perform MAP symbol-by-symbol (in our case, bit-by-bit) detection by computing the marginal posterior distributions of the information bits conditioned on the observation and the system parameters. The MAP detector of the ith bit of signal u is given by

$$\hat{b}_i^{(u)} = \arg\max_{b_i^{(u)}} p(b_i^{(u)} | \mathbf{y}, \mathbf{h}, \boldsymbol{\lambda}, \boldsymbol{\sigma}^2). \tag{4.12}$$

Because $\mathbf{y}$ is observed, (4.12) is equivalent to

$$\hat{b}_i^{(u)} = \arg\max_{b_i^{(u)}} p(\mathbf{y}, b_i^{(u)} | \mathbf{h}, \boldsymbol{\lambda}, \boldsymbol{\sigma}^2), \tag{4.13}$$

where the term $1/p(\mathbf{y}|\mathbf{h}, \boldsymbol{\lambda}, \boldsymbol{\sigma}^2)$ is a constant which has been removed. By the total probability theorem, (4.13) can equivalently be expressed as a marginalization over the full joint distribution as given by

$$\hat{b}_i^{(u)} = \arg\max_{b_i^{(u)}} \sum_{\mathbf{X},\mathbf{C},\mathbf{B},\mathbf{q} \setminus b_i^{(u)}} p(\mathbf{y}, \mathbf{X}, \mathbf{C}, \mathbf{B}, \mathbf{q} | \mathbf{h}, \boldsymbol{\lambda}, \boldsymbol{\sigma}^2), \tag{4.14}$$

where $\sum_{\mathbf{X},\mathbf{C},\mathbf{B},\mathbf{q} \setminus b_i^{(u)}}$ denotes summation over the domain of all variables except $b_i^{(u)}$. Due to the complexity of the marginalization in (4.14), it cannot be performed exactly. However, in many cases an iterative implementation of the sum-product algorithm is well suited to approximate the desired marginal posteriors [7, 8, 58, 79, 166].

The sum-product algorithm operates by passing messages along the edges of the factor graph of the joint distribution. Consider the following high level factorization of the joint distribution:

$$p(\mathbf{y}, \mathbf{X}, \mathbf{C}, \mathbf{B}, \mathbf{q}|\mathbf{h}, \boldsymbol{\lambda}, \boldsymbol{\sigma}^2) = p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2) \prod_{n=0}^{N-1} p(q_n|\boldsymbol{\lambda}) \prod_{u=1}^{U} p(\mathbf{x}^{(u)}|\mathbf{c}^{(u)}) p(\mathbf{c}^{(u)}|\mathbf{b}^{(u)}) \prod_{i=1}^{N_b} p(b_i^{(u)}). \tag{4.15}$$

The corresponding factor graph is shown in Fig. 4.1. The first factor on the right-hand side of (4.15) $p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2)$ is the likelihood function of the received samples. Further factorization of the likelihood function will be considered in Section 4.4 and will form the co-channel detection sub-graph of the factor graph. As a result of the assumption of iid noise samples, the prior probability of the indicator variables $\mathbf{q}$ factors into a multiplication across n as shown in the second term of (4.15). The third term $p(\mathbf{x}^{(u)}|\mathbf{c}^{(u)})$ and the fourth term $p(\mathbf{c}^{(u)}|\mathbf{b}^{(u)})$ of (4.15) are hard constraints [15] corresponding to the modulation and code constraints, respectively. These factors enforce equality between the values of the information bits and the coded bits according to the chosen error correction code and between the values of the coded bits and the symbols according to the chosen modulation. The modulation and code constraints are further factored when implementing the SPA. We do not provide their factorizations since they are available for common modulations and codes in the literature (for example, [7, 79]). The final term in (4.15) is a multiplication of the uniform prior probabilities of the information bits which factor due to the assumption of iid data.

In the proposed factor graph-based receiver, the sum-product algorithm is employed to accomplish tasks of soft detection, soft demodulation, and soft decoding in an iterative (turbo) fashion. An advantage of the sum-product algorithm is that marginal posteriors are readily available for all variables of the factor graph.

4.4 Joint Detection in Co-Channel Interference

4.4.1 Joint MAP Detection

We first develop joint MAP detection of the co-channel signals. The factorization of the joint distribution has a direct impact on the message passing algorithm. Therefore, we begin by exploring the factorization of $p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2)$ from (4.15). The factors of this function make up the co-channel detection sub-graph.

First consider a factor graph based on a direct factorization of the likelihood function. From (4.5)

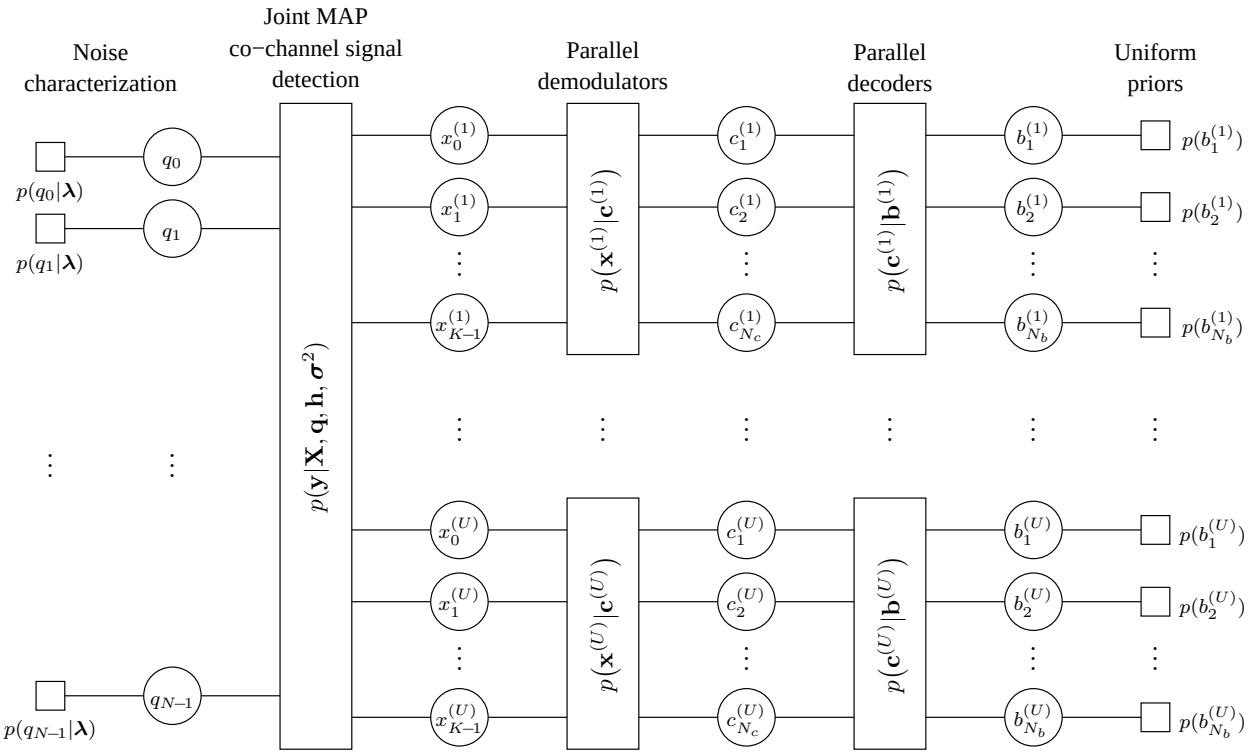

Figure 4.1: Factor graph of the distribution given in (4.15). The model is conditioned on the channel and noise parameters which are iteratively estimated using the EM algorithm.

and (4.10) the likelihood function $p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2)$ may be factored as given by

$$\begin{aligned} p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2) &= \prod_{n=0}^{N-1} p\left(y_n|\mathbf{x}_{[n]}, q_n, \mathbf{h}, \boldsymbol{\sigma}^2\right) \\ &= \prod_{n=0}^{N-1} p\left(y_n|x_n^{(1)}, \ldots, x_{n-L+1}^{(1)}, \ldots, x_n^{(U)}, \ldots, x_{n-L+1}^{(U)}, q_n, \mathbf{h}, \boldsymbol{\sigma}^2\right). \end{aligned} \tag{4.16}$$

A slice of the co-channel detection sub-graph with the factorization given in (4.16) is shown in Fig. 4.2. The notation f_{y_n} denotes the factor $p\left(y_n|x_n^{(1)}, \ldots, x_{n-L+1}^{(1)}, \ldots, x_n^{(U)}, \ldots, x_{n-L+1}^{(U)}, q_n, \mathbf{h}, \boldsymbol{\sigma}^2\right)$. As a result of this factorization, we observe numerous cycles with girth 4 in the factor graph which has a negative impact on convergence [166].

Cycles are avoided in an equalization setting using the BCJR algorithm [137]. The BCJR algorithm was extended to the case of joint detection of a desired and a co-channel signal in ISI by Moon and Gunther [16]. The BCJR algorithm may be generalized to the sum-product algorithm where the factorization of the joint distribution is based on a state-space model (SSM) [79]. This implementation has been referred to as joint MAP (JMAP) detection, because it is the MAP detector when the signals are uncoded (i.e., iid symbols).

Even with these factorizations, the sum-product algorithm may have a complexity which is prohibitive. The complexity of computing the outgoing messages for a factor node in Fig. 4.2, is $O(M^{UL})$. In the JMAP detector (SSM graph), the complexity is also $O(M^{UL})$. In the graph of Fig. 4.2, the messages between the detection nodes f_{y_n} and the symbols variables $x_k^{(u)}$ have dimensions M. On the other hand, the messages for the state variables in the SSM graph have dimensions M^{UL} which may be quite large. Thus, a SSM graph would avoid cycles within the co-channel detection sub-graph at the expense of a slight increase in complexity and possibly a substantial increase in memory.

4.4.2 Approximate Sum-Product Computations

The computational complexity of the sum-product algorithm with the fully factored graph may be prohibitive, but the dimension of the messages is very reasonable (equal to the modulation order M). The received samples y_n are comprised of UL signal components due to the multipath channel and co-channel interference. The complexity of MAP detection increases exponentially with the number of signal components. At the same time, with increased numbers of signal components, the distribution of the received samples approaches a Gaussian distribution. A number of methods in the literature have made use of a Gaussian approximation for co-channel interference or inter-symbol interference (ISI) even for as few as two arriving signals. In the Rake Gaussian method, the likelihood for a given symbol is computed assuming all other symbols are Gaussian distributed random variables [159]. In the Concurrent MAP method, MAP equalization of a particular signal is performed while modeling the symbols from the co-channel interference as Gaussian random variables [42]. One requirement of the central limit theorem is that the summed terms have

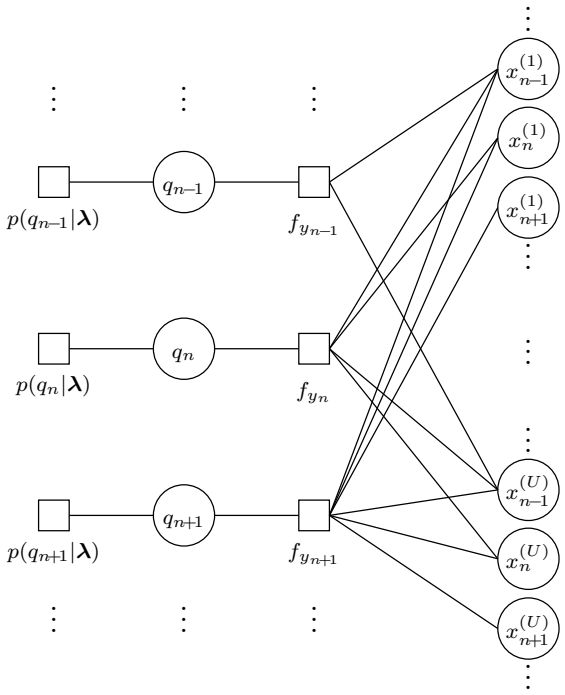

Figure 4.2: Co-channel detection sub-graph of $p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2)p(\mathbf{q}|\boldsymbol{\lambda})$ based on the factorization in (4.16).

identical statistics. This condition is violated when a small number of the multipath components dominate (i.e., their channel coefficient has higher power). Based on our study, the sum-product algorithm messages from the detection factor nodes to the variable nodes are best approximated when the dominant signal components are modeled with a discrete domain and the weaker components are approximated as Gaussian random variables. However, approximating one or more of the dominant signal components as a Gaussian random variable reduces the accuracy of the resulting messages when compared to the sum-product algorithm.

In the proposed algorithm, the sum-product computations for the outgoing messages at the detection factor nodes are approximated by performing marginalization over the strongest signal components and assuming the remaining weaker components are Gaussian distributed. The power of the associated channel coefficient is used to measure the strength of a signal component.

Consider the factor node $f_{y_n} = p\left(y_n|x_n^{(1)}, \ldots, x_{n-L+1}^{(1)}, \ldots, x_n^{(U)}, \ldots, x_{n-L+1}^{(U)}, q_n, \mathbf{h}, \boldsymbol{\sigma}^2\right)$ of Fig. 4.2. For simplicity of notation, let the elements of $\mathbf{h}$ (as defined in (4.6) and (4.7)) be indexed by $d = 1, \ldots, D$ where $D = UL$. Also, we map the multipath and co-channel symbols of the factor to a new vector as defined by

$$\mathbf{s} = [s_1, \ldots, s_D]^{\mathrm{T}} \triangleq [x_n^{(1)}, \ldots, x_{n-L+1}^{(1)}, \ldots, x_n^{(U)}, \ldots, x_{n-L+1}^{(U)}]^{\mathrm{T}} = \mathbf{x}_{[n]}. \tag{4.17}$$

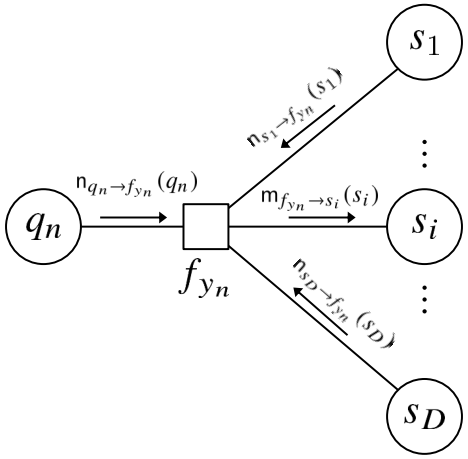

Figure 4.3: Factor graph of f_{y_n} and its associated variables.

Using this new notation, the received sample y_n is expressed as given by

$$y_n = \sum_{d=1}^{D} h_d s_d + w_n \tag{4.18}$$

and the factor associated with the received sample y_n is given by[2]

$$p(y_n|s_1, \ldots, s_D, q_n, \mathbf{h}, \boldsymbol{\sigma}^2) = \mathcal{CN}\left(y_n; \sum_{d=1}^{D} h_d s_d, \, \sigma_{q_n}^2\right). \tag{4.19}$$

A diagram of the factor node and the associated (re-labeled) variables is shown in Fig. 4.3.

We desire to compute the outward messages from factor node f_{y_n} to each of the associated variables s_d for all $d = 1, \ldots, D$. The messages from factor node f_{y_n} to variable nodes s_d and q_n are denoted $\mathsf{m}_{f_{y_n} \to s_d}(s_d)$ and $\mathsf{m}_{f_{y_n} \to q_n}(q_n)$, respectively. Similarly, the messages from variable nodes s_d and q_n to factor node f_{y_n} are denoted $\mathsf{n}_{s_d \to f_{y_n}}(s_d)$ and $\mathsf{n}_{q_n \to f_{y_n}}(q_n)$, respectively. The sum-product computation for the message $\mathsf{m}_{f_{y_n} \to s_i}(s_i)$ is expressed as given by

$$\mathsf{m}_{f_{y_n} \to s_i}(s_i) = \sum_{\mathbf{s} \setminus s_i} \sum_{q_n=1}^{\Gamma} p(y_n|s_1, \ldots, s_D, q_n, \mathbf{h}, \boldsymbol{\sigma}^2) \prod_{d \neq i} \mathsf{n}_{s_d \to f_{y_n}}(s_d) \mathsf{n}_{q_n \to f_{y_n}}(q_n). \tag{4.20}$$

The messages associated with the computation in (4.20) are shown in Fig. 4.3. The complexity of the computation of (4.20) is $O(M^D)$. The number of mixture components is typically $\Gamma = 2$ and does not significantly impact the complexity order. Note that $\Gamma = 1$ corresponds to Gaussian noise.

[2] $\mathcal{CN}(y; \mu, \sigma^2)$ denotes the complex Gaussian probability density function evaluated at y with mean μ and variance σ^2.

In order to reduce the complexity of the detector, a modification to the sum-product rule in (4.20) is proposed. Rather than performing local marginalization over all variables at the factor node, a subset of the variables are treated as Gaussian random variables. The mean and variance of the assumed Gaussian distribution are obtained from the input messages. The number of symbols included in marginalization is set by parameter C which represents the number of symbols over which the marginalization takes place including the output message symbol since its domain is included in the computation.

In the proposed algorithm, the message $\mathsf{m}_{f_{y_n}\to s_i}(s_i)$ is computed as follows:

1. All symbols with the exception of the symbol of interest (i.e., $\{s_d\}_{d\neq i}$) are sorted by their channel coefficient power $|h_d|^2$. The indices of the variables associated with the $C-1$ strongest channel coefficients are grouped into set $\mathcal{A}$. These variables remain a part of the local marginalization as given in (4.20). The number of variables in the set $\mathcal{A}$ will depend on the acceptable complexity in the implementation. The indices of the remaining variables associated with the weaker channel coefficient power are included in the set $\mathcal{B}$ and the distributions of these variables are approximated by Gaussian random variables to eliminate the marginalization over these variables. Let the variables associated with sets $\mathcal{A}$ and $\mathcal{B}$ be given by $\mathbf{s}_{\mathcal{A}}$ and $\mathbf{s}_{\mathcal{B}}$, respectively.

2. The mean and variance of the input messages are computed for all $l \in \mathcal{B}$ according to

$$\mu_{s_l} = \sum_{s_l} s_l \mathsf{n}_{s_l\to f_{y_n}}(s_l) \tag{4.21}$$

and

$$\sigma^2_{s_l} = \sum_{s_l} |s_l - \mu_{s_l}|^2 \mathsf{n}_{s_l\to f_{y_n}}(s_l), \tag{4.22}$$

respectively. The distribution of the summation of these terms is Gaussian with mean and variance given by

$$\mu_{\mathcal{B}} = \sum_{l\in\mathcal{B}} h_l \mu_{s_l} \tag{4.23}$$

and

$$\sigma^2_{\mathcal{B}} = \sum_{l\in\mathcal{B}} |h_l|^2 \sigma^2_{s_l}. \tag{4.24}$$

3. The message is computed with the following approximate sum-product computation:

$$\mathsf{m}_{f_{y_n}\to s_i}(s_i) = \sum_{\mathbf{s}_{\mathcal{A}}} \sum_{q_n=1}^{\Gamma} \tilde{p}(y_n|s_i, \mathbf{s}_{\mathcal{A}}, q_n, \mathbf{h}, \boldsymbol{\sigma}^2) \prod_{d\in\mathcal{A}} \mathsf{n}_{s_d\to f_{y_n}}(s_d) \mathsf{n}_{q_n\to f_{y_n}}(q_n), \tag{4.25}$$

where

$$\tilde{p}(y_n|s_i, \mathbf{s}_{\mathcal{A}}, q_n, \mathbf{h}, \boldsymbol{\sigma}^2) = C\mathcal{N}\left(y_n\,;\; h_i s_i + \sum_{d\in\mathcal{A}} h_d s_d + \mu_{\mathcal{B}}\,,\; \sigma^2_{q_n} + \sigma^2_{\mathcal{B}}\right). \tag{4.26}$$

This algorithm is applied to the computation of the sum-product messages at each of the detection factors in Fig. 4.2. The complexity of the proposed algorithm is $O(M^C)$ for each factor where C is the number of symbols included in the set $\mathcal{A}$ plus the output message variable. Thus, by choosing the size of $\mathcal{A}$, the complexity of the algorithm may be adjusted to match the computational capability of the receiver or the performance requirements.

When the parameters of the noise are unknown, specifically the mixture probabilities, the message $\mathsf{m}_{f_{y_n} \to q_n}$ is important for estimating the mixture probabilities. For equivalent complexity, this message can be computed while marginalizing over C symbols. In this case, the C symbols with the strongest channel coefficients are chosen. As before, the remaining symbols are treated as having a Gaussian distribution. The message from the factor node to the mixture distribution variable is given by

$$\mathsf{m}_{f_{y_n} \to q_n}(q_n) = \sum_{\mathbf{s}_{\mathcal{A}}} \tilde{p}(y_n | \mathbf{s}_{\mathcal{A}}, q_n, \mathbf{h}, \boldsymbol{\sigma}^2) \prod_{d \in \mathcal{A}} \mathsf{n}_{s_d \to f_{y_n}}(s_d), \tag{4.27}$$

where

$$\tilde{p}(y_n | \mathbf{s}_{\mathcal{A}}, q_n, \mathbf{h}, \boldsymbol{\sigma}^2) = \mathcal{CN}\left(y_n \,;\, \sum_{d \in \mathcal{A}} h_d s_d + \mu_{\mathcal{B}} \,,\, \sigma^2_{q_n} + \sigma^2_{\mathcal{B}}\right). \tag{4.28}$$

In summary, the computation of the messages $\mathsf{m}_{f_{y_n} \to s_i}(s_i)$ is presented in Algorithm 1 and computation of the message $\mathsf{m}_{f_{y_n} \to q_n}(q_n)$ is presented in Algorithm 2.

Algorithm 1 Approximate sum-product algorithm computations for symbol distributions

Compute $\mu_{s_d}, \sigma^2_{s_d}$ according to (4.21), (4.22) $\forall d = 1, \ldots, D$
for $i = 1, \ldots, D$ **do**
  $\mathcal{I} \leftarrow \{1, \ldots, D\} \backslash \{i\}$
  $\mathcal{A} \leftarrow \emptyset$
  while $|\mathcal{A}| < C - 1$ **do**
    $a \leftarrow \arg\max_{d \in \mathcal{I}} |h_d|^2$
    $\mathcal{A} \leftarrow \mathcal{A} \cup \{a\}$
    $\mathcal{I} \leftarrow \mathcal{I} \backslash \{a\}$
  end while
  $\mathcal{B} \leftarrow \mathcal{I}$
  Compute $\mu_{\mathcal{B}}, \sigma^2_{\mathcal{B}}$ according to (4.23), (4.24)
  Compute $\mathsf{m}_{f_{y_n} \to s_i}(s_i)$ according to (4.25), (4.26)
end for

4.5 Iterative Parameter Estimation

Thus far we have assumed knowledge of the channel coefficients and of the parameters of the Gaussian mixture distribution. In this section, we address the task of learning the parameters of

Algorithm 2 Approximate sum-product algorithm computations for mixture distribution

Compute μ_{s_d}, $\sigma^2_{s_d}$ according to (4.21), (4.22) $\forall d = 1, \ldots, D$
$\mathcal{I} \leftarrow \{1, \ldots, D\}$
$\mathcal{A} \leftarrow \emptyset$
while $|\mathcal{A}| < C$ **do**
  $a \leftarrow \arg\max_{d \in \mathcal{I}} |h_d|^2$
  $\mathcal{A} \leftarrow \mathcal{A} \cup \{a\}$
  $\mathcal{I} \leftarrow \mathcal{I} \backslash \{a\}$
end while
$\mathcal{B} \leftarrow \mathcal{I}$
Compute $\mu_{\mathcal{B}}$, $\sigma^2_{\mathcal{B}}$ according to (4.23), (4.24)
Compute $\mathsf{m}_{f_{yn} \to q_n}(q_n)$ according to (4.27), (4.28)

the model within the iterative receiver. This task is particularly challenging due to the inter-symbol interference, the co-channel interference, and the impulsive noise. Thus, we rely both *training symbols* included in the transmission and *probabilistic information* for the data symbols obtained from the sum-product algorithm.

The EM algorithm [31] provides a theoretical framework for learning the parameters of a model when there exists a set of unobserved or hidden data. The observed data is $\mathbf{y}$, the hidden data is $\{\mathbf{X}, \mathbf{q}\}$, and therefore the complete data is $\{\mathbf{y}, \mathbf{X}, \mathbf{q}\}$. The unknown channel and noise parameters are $\boldsymbol{\theta} = \{\mathbf{h}, \boldsymbol{\lambda}, \boldsymbol{\sigma}^2\}$. For the problem at hand, the expectation (E) and maximization (M) steps are given by the following expressions:

$$\text{E step: } Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)}) = \sum_{\mathbf{X},\mathbf{q}} \underbrace{p(\mathbf{X}, \mathbf{q} | \mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)})}_{\text{from the SPA}} \ln p(\mathbf{y}, \mathbf{X}, \mathbf{q} | \boldsymbol{\theta}) \tag{4.29}$$

$$\text{M step: } \hat{\boldsymbol{\theta}}^{(p)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)}), \tag{4.30}$$

where $\hat{\boldsymbol{\theta}}^{(p)}$ represents the estimates of the parameters in the pth iteration. Closed form update equations for the maximization step may be obtained by applying the ECM algorithm [29, 118]. The ECM is a variant of the EM algorithm in which maximization is performed sequentially for each parameter while conditioning on the most recent estimate of the other parameters. The EM algorithm is integrated with the SPA as shown in Fig. 4.4.

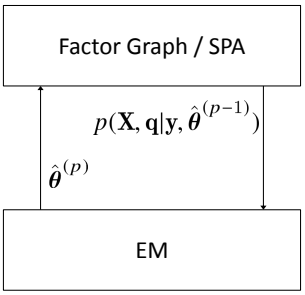

Figure 4.4: Interaction of the SPA and EM algorithm in the iterative receiver. The posterior $p(\mathbf{X}, \mathbf{q}|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)})$ represents the marginal and joint posteriors summarized in Table 4.1.

4.5.1 Expectation Step

By expanding the posterior distribution in (4.29), the expectation step is given by

$$\begin{aligned} Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)}) &= \sum_{\mathbf{X},\mathbf{q}} p(\mathbf{X}, \mathbf{q}|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) \ln(p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2) p(\mathbf{X}) p(\mathbf{q}|\boldsymbol{\lambda})) \\ &= \sum_{\mathbf{X},\mathbf{q}} p(\mathbf{X}, \mathbf{q}|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) \ln(p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2)) + \sum_{\mathbf{X}} p(\mathbf{X}|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) \ln(p(\mathbf{X})) \\ &\quad + \sum_{\mathbf{q}} p(\mathbf{q}|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) \ln(p(\mathbf{q}|\boldsymbol{\lambda})). \end{aligned} \tag{4.31}$$

The factor $\ln(p(\mathbf{X}))$ does not depend on the parameters $\boldsymbol{\theta}$ and thus the second term on the right-hand side of (4.31) is a constant which can be ignored. The likelihood function in the first term of (4.31) is expanded as follows:

$$\begin{aligned} p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2) &= \prod_{n=0}^{N-1} p(y_n|\mathbf{x}_{[n]}, q_n, \mathbf{h}, \boldsymbol{\sigma}^2) \\ &= \prod_{n=0}^{N-1} \frac{1}{\pi\sigma_{q_n}^2} \exp\left\{ -\frac{1}{\sigma_{q_n}^2} \left| y_n - \mathbf{h}^{\mathrm{T}} \mathbf{x}_{[n]} \right|^2 \right\} \\ &= \prod_{n=0}^{N-1} \frac{1}{\pi\sigma_{q_n}^2} \exp\left\{ -\frac{1}{\sigma_{q_n}^2} \left(|y_n|^2 - y_n^* \mathbf{h}^{\mathrm{T}} \mathbf{x}_{[n]} - y_n \mathbf{x}_{[n]}^{\mathrm{H}} \mathbf{h}^* + \mathbf{h}^{\mathrm{T}} \mathbf{x}_{[n]} \mathbf{x}_{[n]}^{\mathrm{H}} \mathbf{h}^* \right) \right\} \end{aligned} \tag{4.32}$$

and the log-likelihood is given by

$$\begin{aligned} &\ln(p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2)) \\ &= -N \ln(\pi) - \sum_{n=0}^{N-1} \ln(\sigma_{q_n}^2) - \frac{1}{\sigma_{q_n}^2} |y_n|^2 + \frac{1}{\sigma_{q_n}^2} y_n^* \mathbf{h}^{\mathrm{T}} \mathbf{x}_{[n]} + \frac{1}{\sigma_{q_n}^2} y_n \mathbf{x}_{[n]}^{\mathrm{H}} \mathbf{h}^* - \frac{1}{\sigma_{q_n}^2} \mathbf{h}^{\mathrm{T}} \mathbf{x}_{[n]} \mathbf{x}_{[n]}^{\mathrm{H}} \mathbf{h}^*. \end{aligned} \tag{4.33}$$

Substituting (4.33) into the first term of (4.31) produces

$$\begin{aligned}\sum_{\mathbf{X},\mathbf{q}} p(\mathbf{X},\mathbf{q}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\ln(p(\mathbf{y}|\mathbf{X},\mathbf{q},\mathbf{h},\boldsymbol{\sigma}^2)) =& \Delta_1 - \sum_{n=0}^{N-1}\sum_{q_n} p(q_n|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\ln(\sigma_{q_n}^2) \\ &- \sum_{n=0}^{N-1}\sum_{q_n} p(q_n|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\frac{1}{\sigma_{q_n}^2}|y_n|^2 \\ &+ \mathbf{h}^{\mathrm{T}}\left(\sum_{n=0}^{N-1}\sum_{\mathbf{x}_{[n]}}\sum_{q_n} p(q_n,\mathbf{x}_{[n]}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\frac{1}{\sigma_{q_n}^2}y_n^*\mathbf{x}_{[n]}\right) \\ &+ \left(\sum_{n=0}^{N-1}\sum_{\mathbf{x}_{[n]}}\sum_{q_n} p(q_n,\mathbf{x}_{[n]}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\frac{1}{\sigma_{q_n}^2}y_n\mathbf{x}_{[n]}^{\mathrm{H}}\right)\mathbf{h}^* \\ &- \mathbf{h}^{\mathrm{T}}\left(\sum_{n=0}^{N-1}\sum_{\mathbf{x}_{[n]}}\sum_{q_n} p(q_n,\mathbf{x}_{[n]}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\frac{1}{\sigma_{q_n}^2}\mathbf{x}_{[n]}\mathbf{x}_{[n]}^{\mathrm{H}}\right)\mathbf{h}^*,\end{aligned} \tag{4.34}$$

where Δ_1 is a constant with respect to $\boldsymbol{\theta}$.

The third term on the right-hand side of (4.34) contains an expectation of the vector $\mathbf{x}_{[n]}$. We define a vector $\boldsymbol{\psi}(n,q_n)$ for the result of the expectation. Each element of the vector $\boldsymbol{\psi}(n,q_n)$ can be computed using the joint posterior probabilities of the mixture distribution and a single symbol as given by

$$\boldsymbol{\psi}(n,q_n) = \sum_{\mathbf{x}_{[n]}} p(q_n,\mathbf{x}_{[n]}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\mathbf{x}_{[n]} = \begin{bmatrix} \sum_{x_n^{(1)}} p(q_n,x_n^{(1)}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})x_n^{(1)} \\ \vdots \\ \sum_{x_{n-L+1}^{(U)}} p(q_n,x_{n-L+1}^{(U)}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})x_{n-L+1}^{(U)} \end{bmatrix}. \tag{4.35}$$

In the fourth term of (4.34), the expectation over $\mathbf{x}_{[n]}$ is given by

$$\boldsymbol{\psi}^{\mathrm{H}}(n,q_n) = \sum_{\mathbf{x}_{[n]}} p(q_n,\mathbf{x}_{[n]}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\mathbf{x}_{[n]}^{\mathrm{H}}. \tag{4.36}$$

The fifth term of (4.34), requires computation of an expectation with respect to joint posterior probabilities of the mixture term and two symbols. We define $\boldsymbol{\Psi}(n,q_n)$ to be a matrix of the expectation with respect to the symbols as given by

$$\boldsymbol{\Psi}(n,q_n) = \sum_{\mathbf{x}_{[n]}} p(q_n,\mathbf{x}_{[n]}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\mathbf{x}_{[n]}\mathbf{x}_{[n]}^{\mathrm{H}}, \tag{4.37}$$

where the element of $\boldsymbol{\Psi}(n,q_n)$ corresponding to the u_1, l_1 symbol in the first dimension and the u_2, l_2 symbols in the second dimension is given by

$$[\boldsymbol{\Psi}(n,q_n)]_{(u_1-1)L+l_1+1,\,(u_2-1)L+l_2+1} = \sum_{x_{n-l_1}^{(u_1)}}\sum_{x_{n-l_2}^{(u_2)}} p(q_n,x_{n-l_1}^{(u_1)},x_{n-l_2}^{(u_2)}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})x_{n-l_1}^{(u_1)}(x_{n-l_2}^{(u_2)})^*, \tag{4.38}$$

where $[\cdot]_{i,j}$ denotes the element in the ith row and jth column of the matrix.

With this set of notation, (4.34) is rewritten as

$$\begin{aligned}\sum_{\mathbf{X},\mathbf{q}} p(\mathbf{X},\mathbf{q}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\ln(p(\mathbf{y}|\mathbf{X},\mathbf{q},\mathbf{h},\boldsymbol{\sigma}^2)) =& \Delta_1 - \sum_{n=0}^{N-1}\sum_{q_n} p(q_n|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\ln(\sigma^2_{q_n}) \\ &- \sum_{n=0}^{N-1}\sum_{q_n} p(q_n|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\frac{1}{\sigma^2_{q_n}}|y_n|^2 \\ &+ \mathbf{h}^{\mathrm{T}}\left(\sum_{n=0}^{N-1}\sum_{q_n}\frac{1}{\sigma^2_{q_n}}y_n^*\boldsymbol{\psi}(n,q_n)\right) \\ &+ \left(\sum_{n=0}^{N-1}\sum_{q_n}\frac{1}{\sigma^2_{q_n}}y_n\boldsymbol{\psi}^{\mathrm{H}}(n,q_n)\right)\mathbf{h}^* \\ &- \mathbf{h}^{\mathrm{T}}\left(\sum_{n=0}^{N-1}\sum_{q_n}\frac{1}{\sigma^2_{q_n}}\boldsymbol{\Psi}(n,q_n)\right)\mathbf{h}^*. \end{aligned} \tag{4.39}$$

Finally, the third term of (4.31) is given by

$$\sum_{\mathbf{q}} p(\mathbf{q}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\ln(p(\mathbf{q}|\boldsymbol{\lambda})) = \sum_{n=0}^{N-1}\sum_{q_n} p(q_n|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\ln(\lambda_{q_n}). \tag{4.40}$$

The required posterior distributions for the expectation step are summarized in Table 4.1. The posteriors are computed from the messages in the factor graph of Fig. 4.1 using the algorithm of Section 4.4.2. Marginalization is performed on the strongest components up to a limit of C symbols. The remaining symbols are modeled as Gaussian random variables.

4.5.2 Maximization Step

Following the expectation step, maximization is sequentially performed on the parameters of the model: $\boldsymbol{\lambda} \rightarrow \boldsymbol{\sigma}^2 \rightarrow \mathbf{h}$. With each update, we condition on most recent estimate of the other parameters.

Mixture Probabilities

Only the third term in (4.31) is required to update $\boldsymbol{\lambda}$ as the other to terms are not functions of $\boldsymbol{\lambda}$. The constraint $\sum_{q_n}\lambda_{q_n} = 1$ is incorporated into the parameter update with a Lagrangian multiplier as follows

$$\frac{\partial}{\partial\lambda_{q_n}}\sum_{n=0}^{N-1}\sum_{q_n} p(q_n|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)})\ln(\lambda_{q_n}) - \alpha\left(\sum_{q_n}\lambda_{q_n} - 1\right) = 0. \tag{4.41}$$

The derivatives with respect to λ_{q_n} and α are given by

$$\sum_{n=0}^{N-1} p(q_n|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) \frac{1}{\lambda_{q_n}} - \alpha = 0 \quad \forall q_n = 1, \ldots, \Gamma \tag{4.42}$$

and

$$\sum_{q_n=1}^{\Gamma} \lambda_{q_n} - 1 = 0, \tag{4.43}$$

respectively. Solving for the λ_{q_n}, we derive the following system of equations:

$$\lambda_{q_n} = \frac{1}{\alpha} \sum_{n=0}^{N-1} p(q_n|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) \quad \forall q_n = 1, \ldots, \Gamma \tag{4.44}$$

$$\sum_{q_n=1}^{\Gamma} \lambda_{q_n} = 1. \tag{4.45}$$

Substituting (4.44) for the λ_{q_n} in (4.45) and solving for α, we find that $\alpha = N$. Therefore, the parameter update equation is

$$\hat{\lambda}_{q_n}^{(p)} = \frac{1}{N} \sum_{n=0}^{N-1} p(q_n|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}). \tag{4.46}$$

Mixture Variances

The first term of (4.31) is a function of the Gaussian mixture component variances. Taking the derivative of this term (as given in (4.39)) and setting it equal to zero results in the following expression:

$$\begin{aligned} \frac{\partial}{\partial \sigma_{q_n}^2} \sum_{\mathbf{X},\mathbf{q}} p(\mathbf{X}, \mathbf{q}|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) \ln(p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2)) &= -\sum_{n=0}^{N-1} p(q_n|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) \frac{1}{\sigma_{q_n}^2} \\ &+ \frac{1}{\sigma_{q_n}^4} \left[\sum_{n=0}^{N-1} p(q_n|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) |y_n|^2 - 2\Re\left\{ \mathbf{h}^{\mathrm{T}} \left(\sum_{n=0}^{N-1} y_n^* \boldsymbol{\psi}(n, q_n) \right) \right\} + \mathbf{h}^{\mathrm{T}} \left(\sum_{n=0}^{N-1} \boldsymbol{\Psi}(n, q_n) \right) \mathbf{h}^* \right] \\ &= 0. \end{aligned} \tag{4.47}$$

Solving for $\sigma_{q_n}^2$ results in the following update equation:

$$\hat{\sigma_{q_n}^2}^{(p)} = \frac{\left[\sum_{n=0}^{N-1} p(q_n|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) |y_n|^2 - 2\Re\left\{ \mathbf{h}^{\mathrm{T}} \left(\sum_{n=0}^{N-1} y_n^* \boldsymbol{\psi}(n, q_n) \right) \right\} + \mathbf{h}^{\mathrm{T}} \left(\sum_{n=0}^{N-1} \boldsymbol{\Psi}(n, q_n) \right) \mathbf{h}^* \right]}{\sum_{n=0}^{N-1} p(q_n|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)})}, \tag{4.48}$$

where $\mathbf{h}$ is set to the most recent estimate $\hat{\mathbf{h}}^{(p-1)}$.

Channel Coefficients

Finally, we derive an update equation for the channel coefficient vector $\mathbf{h}$. In order to simplify notation we define the following variables:

$$\mathbf{z} = \sum_{n=0}^{N-1} y_n^* \sum_{q_n} \frac{1}{\sigma_{q_n}^2} \boldsymbol{\psi}(n, q_n) \tag{4.49}$$

$$\mathbf{Z} = \sum_{n=0}^{N-1} \sum_{q_n} \frac{1}{\sigma_{q_n}^2} \boldsymbol{\Psi}(n, q_n), \tag{4.50}$$

where $\sigma_{q_n}^2$ is set to its most recent estimate $\hat{\sigma}_{q_n}^{2\,(p)}$. With these definitions the first term of (4.31) (as given in (4.39)) is simplified to

$$\sum_{\mathbf{X},\mathbf{q}} p(\mathbf{X}, \mathbf{q}|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) \ln(p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2)) = \mathbf{h}^{\mathrm{T}}\mathbf{z} + \mathbf{z}^{\mathrm{H}}\mathbf{h}^* - \mathbf{h}^{\mathrm{T}}\mathbf{Z}\mathbf{h}^* + \Delta_2, \tag{4.51}$$

where Δ_2 represents terms independent of $\mathbf{h}$. Taking the derivative of (4.51) with respect to $\mathbf{h}^*$ (where $\mathbf{h}$ and $\mathbf{h}^*$ are considered unique) results in the following expression

$$\begin{aligned}
&\frac{\partial}{\partial \mathbf{h}^*} \sum_{\mathbf{X},\mathbf{q}} p(\mathbf{X}, \mathbf{q}|\mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)}) \ln(p(\mathbf{y}|\mathbf{X}, \mathbf{q}, \mathbf{h}, \boldsymbol{\sigma}^2)) \\
&= \frac{\partial}{\partial \mathbf{h}^*} \left(\mathbf{h}^{\mathrm{T}}\mathbf{z} + \mathbf{z}^{\mathrm{H}}\mathbf{h}^* - \mathbf{h}^{\mathrm{T}}\mathbf{Z}\mathbf{h}^* \right) \\
&= \mathbf{z}^{\mathrm{H}} - \mathbf{h}^{\mathrm{T}}\mathbf{Z} \\
&= 0.
\end{aligned} \tag{4.52}$$

Solving for $\mathbf{h}$, the update equation is given by

$$\hat{\mathbf{h}}^{(p)} = (\mathbf{Z}^{\mathrm{T}})^{-1}\mathbf{z}^*. \tag{4.53}$$

A summary of the required computations for parameter estimation is given in Table 4.1.

4.6 Message Passing Schedule

Iterative channel estimation is integrated with the sum-product algorithm. An iteration of the iterative receiver includes an iteration of message passing for the sum-product algorithm and re-estimation of the parameters based on the ECM algorithm. At a high level the sum-product algorithm message passing schedule includes performing message passing within the co-channel

Table 4.1: Computations for parameter estimation

E-step:	$p(q_n \vert \mathbf{y}, \hat{\theta})\ \forall n$	
	$p(q_n, x_{n-l}^{(u)} \vert \mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)})\ \forall n, u, l$	
	$p(q_n, x_{n-l_1}^{(u_1)}, x_{n-l_2}^{(u_2)} \vert \mathbf{y}, \hat{\boldsymbol{\theta}}^{(p-1)})\ \forall n, u_1, u_2, l_1, l_2$	
	$\boldsymbol{\psi}(n, q_n)$	(4.35)
	$\boldsymbol{\Psi}(n, q_n)$	(4.37)
M-step:	$\lambda_{q_n}\ \forall q_n = 1, \ldots, \Gamma$	(4.46)
	$\sigma_{q_n}^2\ \forall q_n = 1, \ldots, \Gamma$	(4.48)
	$\mathbf{h}$	(4.49), (4.50), (4.53)

detection sub-graph followed by message passing back and forth through each user's demodulation and decoding sub-graphs. The message passing iteration ends when messages have been updated from the demodulation and decoding sub-graphs to the symbols variables.

Within the co-channel detection sub-graph a forward-backward message passing structure is implemented. That is, messages are computed from factor node f_{y_n} to the associated symbols variables beginning with $n = 0$ and continuing on to $n = N - 1$. Then the same messages are recomputed in the reverse order from $n = N - 1$ to $n = 0$. Before computing the messages for each factor node, all incoming messages from the associated variable nodes are updated.

We denote the total number of iterations of message passing as N_{rx}. Within an iteration of message passing, each parameter is updated once. Cycles exist within the user's demodulation and decoding sub-graphs, either within the decoder sub-graph or between the demodulation and decoding factors. Thus, we consider the option of performing multiple iterations of message passing within each user's demodulation and decoding sub-graph *per* iteration of the overall receiver. We denote the number of demodulation and decoding iterations performed *per* overall receiver iteration as N_{dd}.

Since a large number of short cycles (i.e., with a girth of 4) are present in the co-channel detection subgraph, damping is applied to the message updates in this subgraph. The *new* message from an observation node f_{y_n} to a variable node $x_k^{(u)}$ is computed from the *old* message and the *full* message where the full message update is computed according to Section 4.4.2. After computing the full message update the damped message update is given by

$$\mathsf{m}^{\text{new}}_{f_{y_n} \to x_k^{(u)}}\left(x_k^{(u)}\right) = \left(\mathsf{m}^{\text{old}}_{f_{y_n} \to x_k^{(u)}}\left(x_k^{(u)}\right)\right)^{\eta} \left(\mathsf{m}^{\text{full}}_{f_{y_n} \to x_k^{(u)}}\left(x_k^{(u)}\right)\right)^{1-\eta}, \tag{4.54}$$

where *eta* is the damping factor which is constrained to $0 \leq \eta < 1$.

Finally, it is useful to remove a user from the factor graph model if it has been successfully detected. This allows the marginalization to be focused on the remaining signal(s). The receiver is not able to determine if the information bits of a user have been correctly detected, so we propose a stopping condition based on the beliefs within the factor graph. Specifically we look for the strongest

posterior probabilities of all the symbols of a user to be greater than a threshold as given by the condition

$$\min_{k=0,\ldots,K-1}\left\{\max_{x_k^{(u)}} p\left(x_k^{(u)}|\mathbf{y},\hat{\boldsymbol{\theta}}^{(p-1)}\right)\right\} > \mathcal{P}.$$

We found that $\mathcal{P} = 0.999$ performed well in simulations. When the stopping condition is met, the user's estimated signal is canceled from the received signal and the iterative receiver continues processing the remaining signals.

4.7 Numerical Results

The proposed receiver is demonstrated in the presence of both Gaussian noise and non-Gaussian noise. A 1/2-rate Turbo code is used throughout the results and several digital phase-amplitude modulations are considered. The code is constructed from two recursive systematic 1/2-rate convolutional codes with generator polynomials of 23 and 35. The interleaver is a pseudo-random permutation. Every-other parity bit of each convolutional code is punctured in order to obtain the 1/2-rate Turbo code. One of the convolutional codes is terminated to the all zeros state, while the termination bits of the other are punctured. The coded bits are interleaved prior to modulation where each user's interleaver is a unique pseudo-random permutation.

Several multipath power profiles are considered. Multipath profiles including the number of taps and the average (relative) power per tap are given in Table 4.2.[3] The channel coefficients are assumed to be independent complex Gaussian random variables with variance given by the power profiles. While the multipath profiles provide average power distributions for the channel coefficients relative to one another, the total power in the channel is normalized.

In the results that follow, we consider (a) Gaussian noise and (b) Gaussian mixture noise as defined in (4.11). More specifically, we consider impulsive noise which is often modeled as a two-component Gaussian mixture ($\Gamma = 2$). The first component represents thermal noise and typically has a higher probability and lower variance. The second component represents the impulsive component and typically has a lower probability and significantly higher variance. Case (a) can be considered a one-component ($\Gamma = 1$) version of case (b).

We define the signal-to-noise ratio (SNR) as the ratio between the normalized power of the signal of interest and the variance of the $q = 1$ component (i.e., the thermal noise component). Let the SNR of signal u be denoted by SNR_u as given by

$$\text{SNR}_u = \frac{\sum_{l=0}^{L-1}|h_l^{(u)}|^2}{\sigma_1^2}. \tag{4.55}$$

The signal-to-interference ratio (SIR) is defined as the ratio of the normalized power of the signal of interest (in this work signal 1) to the normalized power of the interference. For a $U = 2$ user

[3]The power distribution for profile B is taken from [167].

Table 4.2: Multipath profiles for simulations

Profile	No. taps (L)	Average power per tap
A	4	[0.644, 0.237, 0.087, 0.032]
B	5	[0.052, 0.212, 0.473, 0.212, 0.052]
C	6	[0.167, 0.167, 0.167, 0.167, 0.167, 0.167]

system model, as is considered here, SIR is given by

$$\text{SIR} = \frac{\sum_{l=0}^{L-1} |h_l^{(1)}|^2}{\sum_{l=0}^{L-1} |h_l^{(2)}|^2}. \tag{4.56}$$

4.7.1 Receiver Performance: Known Parameters

In this section simulations are performed under the assumption that the channel and noise parameters are known.

Gaussian Noise

In Figs. 4.5 and 4.6 the FER of the proposed approximate MAP (AMAP) algorithm is compared with the FER of the CMAP algorithm from [42], serial interference cancellation (SIC) with MAP single user equalization, and JMAP detection are compared for $\text{SNR}_1 = 6$ and 10 dB, respectively. In this simulation we set receiver complexity parameter to $C = 4$ so that the AMAP and CMAP algorithms have a similar complexity order. From Fig. 4.5 we observe that the AMAP algorithm outperforms CMAP and that the most significant improvement in FER is achieved when the signals have similar power levels ($-3 \leq \text{SIR} \leq 3$ dB) and the SNR is high. Similar power levels is our range of interest because when the signals have a significantly disparity power levels, joint detection is not required (as demonstrated by SIC with MAP single-user equalization). The proposed technique thus provides a substantial improvement in the state of the art.

In Fig. 4.7, the FER is shown with respect to the number of iterations where we observe that the proposed algorithm converges in approximately 2-3 iterations fewer than CMAP.[4] Thus, the proposed algorithm reduces computational complexity by 20–40% due to faster convergence and simultaneously achieves a lower FER. Similar improvements in convergence are observed for AMAP for the moderate and high SNR values considered in Figs. 4.5 and 4.6.

For the same simulation setup, we consider the FER convergence for three message passing cases in Fig. 4.8: no damping, damping, and model mismatch. We observe that damping with $\eta = 0.4$

[4] We define convergence as reaching a FER which is within 20% of the minimum FER that the respective algorithms achieve.

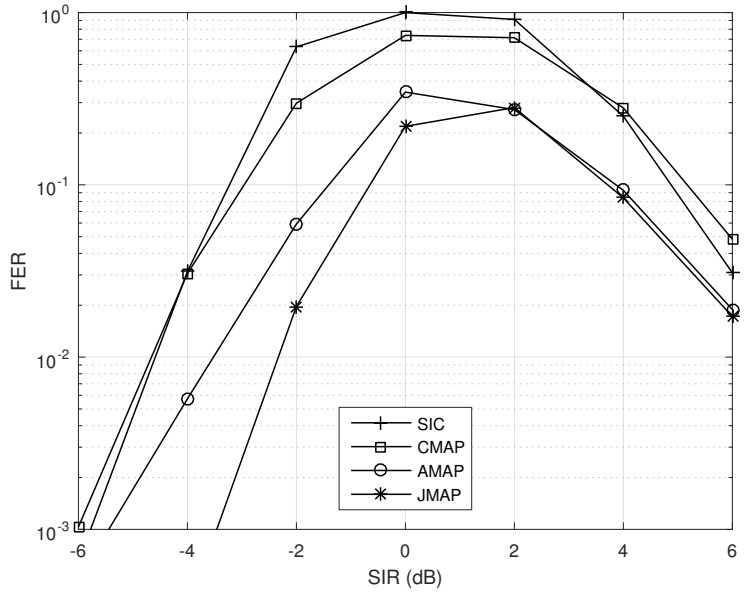

Figure 4.5: FER of the proposed AMAP and the CMAP algorithms with respect to SIR. The FER of the desired signal ($u = 1$) is shown. Simulation parameters: $U = 2$, $K = 250$, QPSK, profile A, Gaussian noise, $\text{SNR}_1 = 6$ dB, $C = 4$, $N_{rx} = 20$, $N_{dd} = 2$, and $\eta = 0.4$.

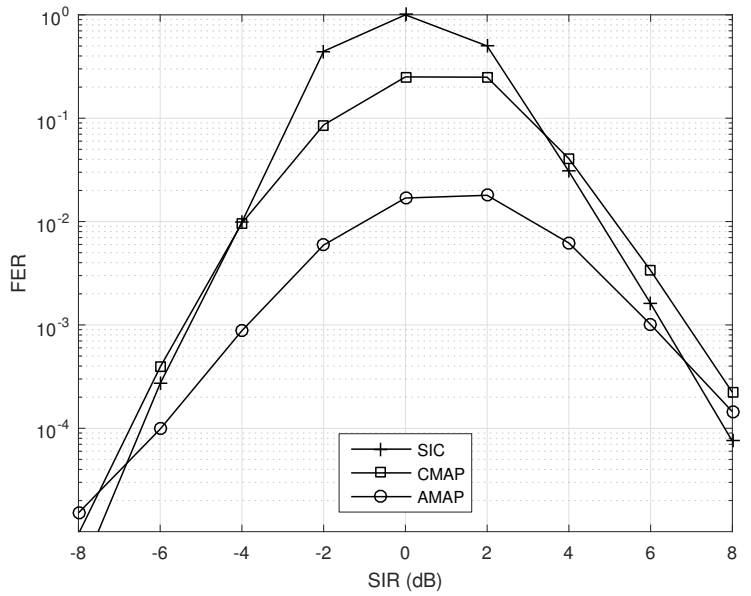

Figure 4.6: FER of the proposed AMAP and the CMAP algorithms with respect to SIR. The FER of the desired signal ($u = 1$) is shown. Simulation parameters: $U = 2$, $K = 250$, QPSK, profile A, Gaussian noise, $\text{SNR}_1 = 10$ dB, $C = 4$, $N_{rx} = 20$, $N_{dd} = 2$, and $\eta = 0.4$.

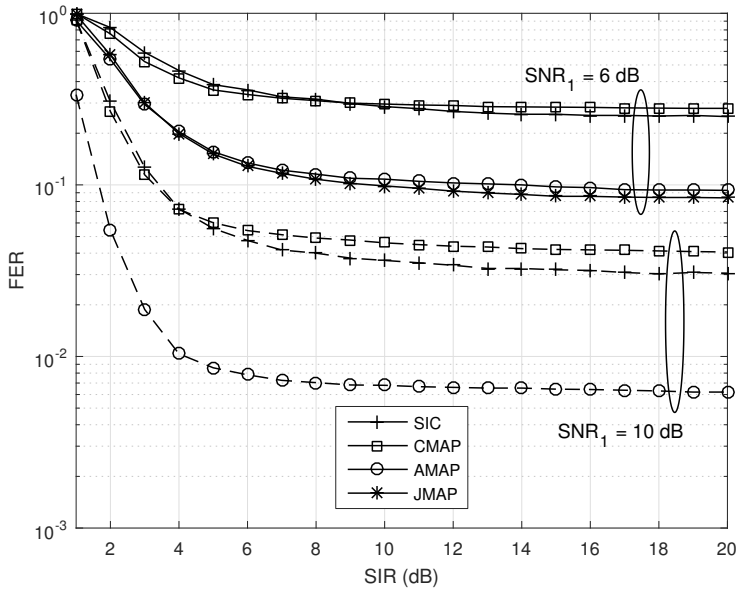

Figure 4.7: FER of the proposed AMAP and the CMAP algorithms with respect to the number of iterations performed. Simulation parameters: $U = 2$, $K = 250$, QPSK, profile A, Gaussian noise, SIR = 4 dB, $C = 4$, $N_{dd} = 2$, and $\eta = 0.4$.

provides a significant FER improvement and does not slow the rate of convergence versus iteration. In addition, we observe from Fig. 4.8 that when utilizing the GMM receiver model (with $\lambda_2 = 0.1$) the performance is actually improved even though the noise distribution is in fact Gaussian. This is because, with the Gaussian model and without damping, the beliefs can become biased too strongly toward low entropy distributions. The effect of the GMM is to slow the biasing of the beliefs, similar to damping. If we conclude that over-estimating the number of GMM components in the noise model (i.e., assuming a two-term GMM when the noise is actually Gaussian) does not negatively affect performance. This conclusion is important because the noise model is not necessarily known *a priori*.

The performance of the proposed receiver with 16QAM modulation is shown in Fig. 4.9. We set $C = 3$ to compensate for the increase in the modulation order. The CMAP algorithm is not considered due to its complexity.[5] Again a significant improvement in FER is obtained by modeling the Gaussian noise with a GMM and applying damping at high SNR (left-hand side of Fig. 4.9). The results demonstrate that the proposed receiver is capable of performing adequate detection, but requires a greater separation in signal power between the two users than with lower order constellations.

[5]With 16QAM and a 4-tap channel, the complexity of CMAP is on the order of $M^{L+1} = 1\,048\,576$ per received sample.

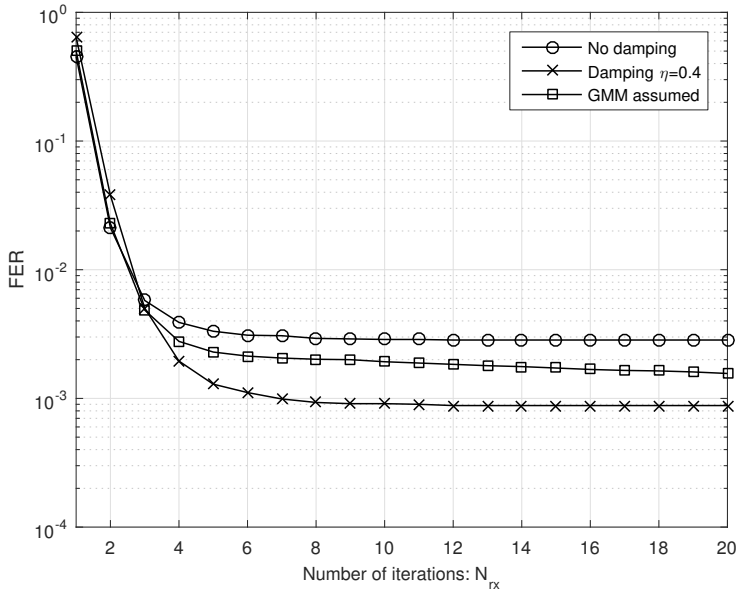

Figure 4.8: FER of the AMAP algorithm for three message passing cases: no damping, damping, and model mismatch. Simulation parameters: $U = 2$, $K = 250$, QPSK, profile A, Gaussian noise, $\text{SNR}_1 = 10$ dB, SIR $= -4$ dB, and $C = 4$.

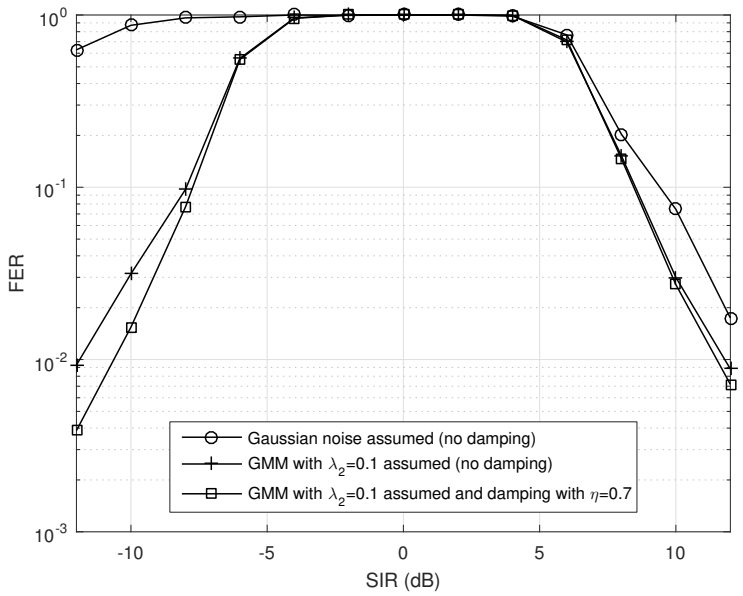

Figure 4.9: FER of the AMAP algorithm for a case in which complexity is higher due to 16QAM modulation. Simulation parameters: $U = 2$, $K = 250$, 16QAM, profile A, Gaussian noise, $\text{SNR}_1 = 20$ dB, $C = 3$, $N_{rx} = 20$, and $N_{dd} = 2$.

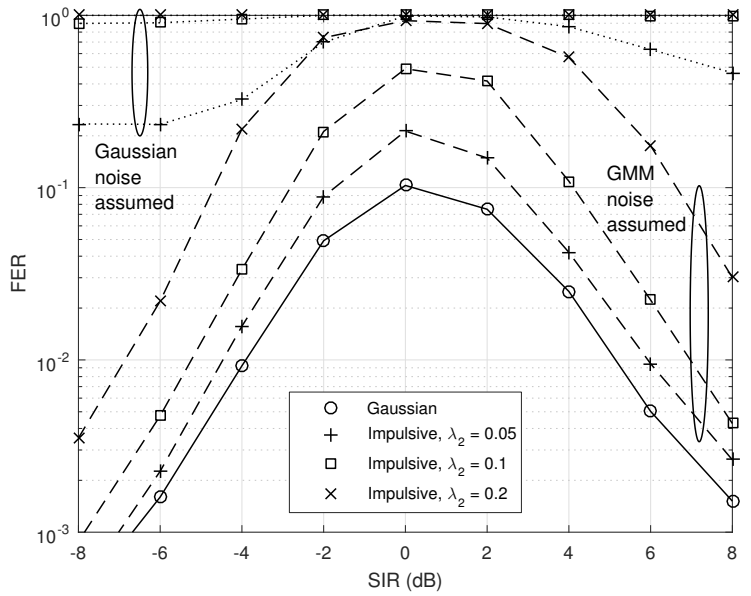

Figure 4.10: FER of the AMAP algorithm for four noise environments. Simulation parameters: $U = 2$, $K = 250$, QPSK, profile B, $\text{SNR}_1 = 10$ dB, $C = 4$, $N_{rx} = 20$, $N_{dd} = 3$, and $\eta = 0.4$.

Non-Gaussian Noise

A $U = 2$ signal scenario is considered in a non-Gaussian noise environment. A two-term ($\Gamma = 2$) mixture model is assumed where the first term is considered to be a thermal component and the second term an impulsive component. The ratio of the variance of the impulsive component to the thermal component is $\sigma_2^2/\sigma_1^2 = 100$. The following three sets of mixture probabilities are considered in the simulation:

- $\lambda_1 = 0.95$, $\lambda_2 = 0.05$
- $\lambda_1 = 0.9$, $\lambda_2 = 0.1$
- $\lambda_1 = 0.8$, $\lambda_2 = 0.2$

The FER performance of the proposed algorithm is shown in Fig. 4.10 for each noise model and for Gaussian noise. For comparison, the performance of the proposed algorithm where the receiver assumes that the noise is Gaussian distributed is also provided in Fig. 4.10. We observe significant gains by designing the receiver to handle non-Gaussian noise. In contrast to the previous section, a significant loss in performance results from under-estimating the number of GMM components (i.e., assuming Gaussian noise when two-term Gaussian mixture noise is present).

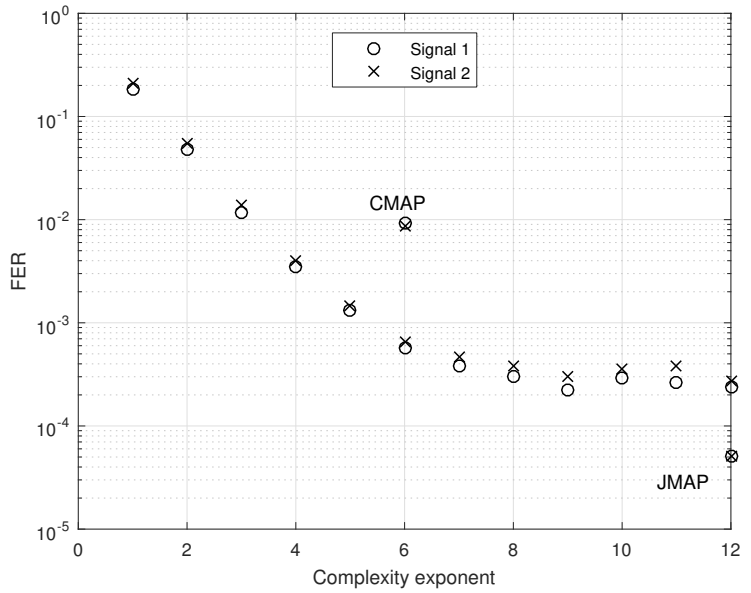

Figure 4.11: FER as a function of the complexity exponent of the algorithm. Simulation parameters: $U = 2$, $K = 500$, BPSK, profile C, Gaussian noise, $\text{SNR}_1 = 4$ dB, SIR = 1 dB (i.e., $\text{SNR}_2 = 3$ dB), $N_{rx} = 5$, $N_{dd} = 2$, and $\eta = 0.4$.

Complexity Analysis

As our goal is to create a receiver which most effectively uses the available computational capability, we evaluate the performance versus the complexity parameter C. In Fig. 4.11, the FER is shown for BPSK modulation and multipath profile C ($L = 6$). FER is shown versus the complexity exponent which is given by C for the AMAP algorithm. The number of symbols in the marginalization C is varied from 1 to 12. For $C = 1$ the only symbol modeled with a discrete domain is the symbol for which we are computing the message. All other symbols are modeled as Gaussian random variables, making $C = 1$ equivalent to the Rake Gaussian method generalized to higher order modulations [159]. For $C = 12$, the proposed AMAP algorithm is equivalent to the sum-product algorithm. As a point of reference, we show the performance of the CMAP and full JMAP algorithms. The complexity of CMAP is $\mathcal{O}(M^L)$, thus its complexity exponent is L. Similarly the complexity of JMAP detection is $\mathcal{O}(M^{UL})$ and, thus, its complexity exponent is UL. We observe that the proposed algorithm outperforms CMAP for a similar complexity. Also, the rate of decrease in the FER is nearly linear for $C = 1, \ldots, 6$ and levels off for $C > 6$ validating the proposed receiver design.

Remark 8. Including the dominant signal components in the local marginalization performed by the SPA is important for reliable detection, but there are diminishing returns for the symbols with weaker channel coefficients.

The complexity of the AMAP algorithm is unique in that it is not dependent on the number of

signals U or multipath components L. The complexity of the JMAP algorithm is exponential in both U and L and the complexity of the CMAP algorithm is linear in U and exponential in L. In contrast, the complexity of the AMAP algorithm is controlled by C and does not necessarily need to increase with increased U or L. However, there is a trade-off between complexity and performance as shown in Fig. 4.11. If the number of signals or multipath components is increased, it is expected that an increase in C would be necessary to avoid degraded performance.

There has been significant development in graphics processing units (GPUs) in recent years [168]. While their growth has largely been driven by graphics applications, GPUs are increasingly being used for massive parallelization in scientific applications (e.g., deep learning). Massive parallelization can be utilized for the co-channel detection subgraph. A reduction in processing time on the order of K (the number of observations) is possible with a parallel schedule for the co-channel detection subgraph. Additionally, the demodulation and decoding subgraphs of each signal can be processed in parallel.

4.7.2 Receiver Performance: Estimated Parameters

We again use a 1/2-rate turbo code, but the length of the transmission is increased to $K = 1031$ symbols: a 31 symbol training sequence and 1000 data symbols. We consider two signals each with QPSK modulation and a receiver with complexity parameter $C = 4$ and $N_{rx} = 20$ iterations of message passing/parameter re-estimation. Results are provided for both Gaussian noise and non-Gaussian noise ($\Gamma = 2$). In both cases, the starting estimate of the mixture probability is $\hat{\lambda} = [0.95, 0.05]^{\mathrm{T}}$. Thus, in the case of Gaussian noise, the receiver structure initially models the noise with a GMM. We set $\hat{\sigma}_1^2$ equal to the initial estimate of total noise power and $\hat{\sigma}_2^2 = 20\hat{\sigma}_1^2$. The initial channel estimates are made using a linear least squares estimator in which the received samples that are estimated to have noise generated from the impulsive component are blanked [169].

The FER of the proposed receiver in Gaussian noise is shown in Fig. 4.12. Similarly, the FER of the proposed receiver in non-Gaussian noise ($\lambda_1 = 0.9$, $\sigma_2^2/\sigma_1^2 = 100$) is shown in Fig. 4.13. For comparison, we also provide the detection performance for perfect knowledge of the channel and noise parameters and the detection performance for a receiver that uses the initial estimates of the parameters.

The iterative receiver generally achieves a FER which is within an order of magnitude (and often within a factor of two) of the FER when the parameters are known. In all cases the iterative receiver outperformed the receiver which relied on initial parameter estimates. We observe this most notably in Fig. 4.13 with non-Gaussian noise. Estimation error in the initial parameter estimates limits the receiver performance across SIR.

Remark 9. The proposed receiver is capable of utilizing the coarse estimates to establish convergence toward the desired solution. However, the coarse estimates themselves are not sufficient for reliable data detection.

Remark 10. The receiver does not require prior knowledge of the noise distribution—whether it is

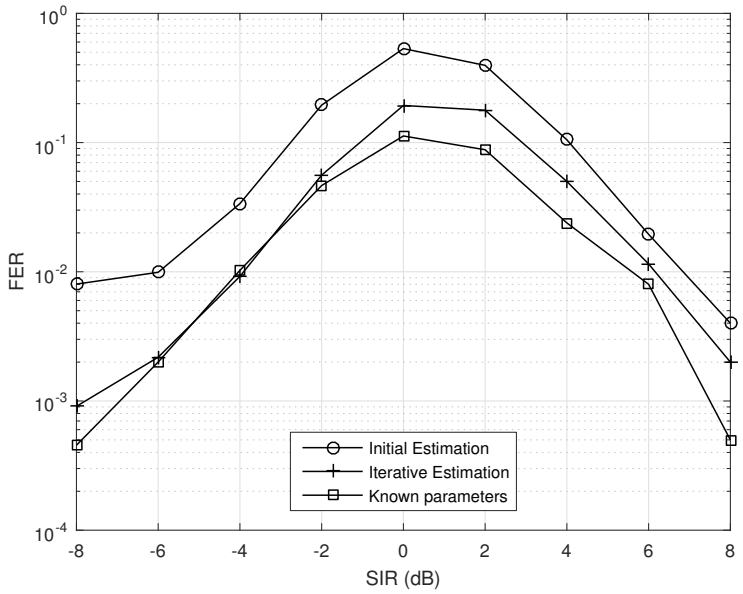

Figure 4.12: FER with channel estimation in Gaussian noise. Simulation parameters: $U = 2$, $K = 1031$, QPSK, $\text{SNR}_1 = 10$ dB, $C = 4$, $N_{rx} = 20$, $N_{dd} = 3$, and $\eta = 0.4$.

Gaussian or non-Gaussian. This is evidenced by the fact that the same initial GMM parameters are effective in both Gaussian and non-Gaussian noise.

4.8 Conclusion

The complexity of joint MAP detection of co-channel signals is prohibitive. Therefore, in this work we proposed an approximation to joint MAP detection for a factor graph-based iterative receiver. The proposed receiver was shown to outperform state-of-the-art receiver algorithms while having a *lower* complexity. In addition, the receiver was shown to perform well in multipath channels and non-Gaussian noise environments when the parameters of the channel and noise were not known *a priori*. Applying iterative parameter estimation in the receiver resulted in only a minimal loss in detection performance when compared to a receiver with perfect knowledge of the parameters.

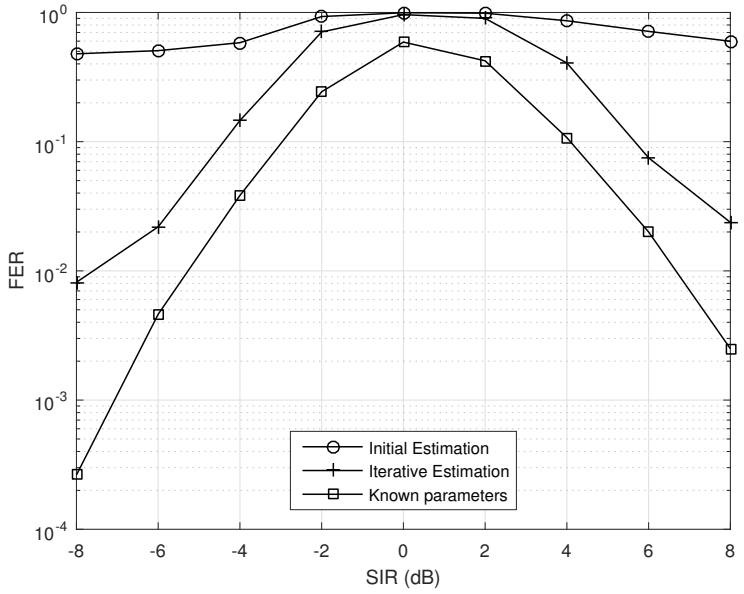

Figure 4.13: FER with channel estimation in non-Gaussian noise. Simulation parameters: $U = 2$, $K = 1031$, QPSK, $\text{SNR}_1 = 10$ dB, $C = 4$, $N_{rx} = 20$, $N_{dd} = 3$, and $\eta = 0.4$.

Chapter 5

Probabilistic Receiver Architecture for Multi-Signal Detection

5.1 Introduction

Belief propagation (BP)—also known as the sum-product algorithm—is an effective inference tool for many of the tasks performed by a communications receiver. Notable examples include decoding, demodulation, and multi-user detection [7, 8, 79]. In the Bayesian framework, the data (e.g., symbols) and the parameters (e.g., channel coefficients) of the received signal are modeled as random variables. This results in a non-linear observation model which, along with the fact that the model variables are both continuous and discrete, makes exact application of BP infeasible. To handle the non-linear observation model, some of the work in the literature has approximated the BP messages by making Gaussian assumptions on the message distributions where needed [54,92]. However, a more common approach has been to work with parameter estimates (rather than distributions) computed from "soft" symbol estimates. The expectation maximization (EM) algorithm was shown to provide theoretical justification for this approach where soft symbols estimates take the form of posterior expectations [58, 61, 82, 165]. The trade-off with these approaches is the loss of probabilistic information. While the estimates (i.e., the messages) are computed taking into account the underlying probability distributions, they do not convey probabilistic information (e.g., our confidence in the estimate).

Variational message passing based on the MF approximation is another viable alternative to BP for estimation of continuous variables and for handling the non-linear observation model [35, 170]. In contrast to EM, messages computed according to the MF approximation *do* convey probabilistic information. In fact, EM is a special case of the MF approximation where the messages are given by Dirac delta functions [32]. A disadvantage of the MF approximation is that it is not suited for demodulation and decoding tasks where the factor nodes contain hard constraints. Recognizing that BP and the MF approximation have complementary strengths, receiver algorithms have been

developed which combine these algorithms [32–34]. Riegler *et al.* provided a theoretical justification for the combined BP-MF message passing framework in [35]. The justification is based on the construction of a region-based free energy approximation analogous to that given for BP by Yedidia *et al.* in [15]. Specifically, in [35] it is shown that fixed points of the combined message passing algorithm correspond to stationary points of a constrained region-based free energy approximation. A notable result from Riegler's work is that the BP-MF framework provides a consistent rule for when to pass *a posteriori* probabilities (posterior beliefs) and when to pass extrinsic messages based on the constructed model.

The combined message passing algorithm is particularly applicable to communications receivers. In this setting, BP is a generalization of iterative decoding of error correction codes and the MF approximation is applicable to the estimation of parameters such as coefficients of a wireless multipath channel. The combined message passing algorithm is demonstrated in [35] for channel estimation and equalization in an orthogonal frequency-division multiplexing (OFDM) system with demodulation and decoding. Subsequent works have applied the BP-MF framework to various channel estimation scenarios [50, 90, 171–176]. Combined BP-MF message passing has also been applied to multi-user code-division multiple access (CDMA) [32], multiple-input multiple-output (MIMO) systems [33, 34, 36, 37], co-channel interference [38], and frequency-domain equalization [101].[1] Joint channel estimation is included in the majority of these works as well [32–34, 36–38]. In scenarios involving detection of multiple signals, the standard application of the BP-MF framework is to include the equalization task within the MF subgraph.[2] As we will show in this chapter, the MF approximation is a poor choice for detection tasks involving signal separation (i.e., signal models with interference-corrupted observations). In [34], the BP-MF framework was applied to MIMO-OFDM where multi-stream equalization is within the MF subgraph, but the algorithm relies upon a generous initialization point.

In this chapter, we develop a receiver architecture for multi-signal detection based on the BP-MF framework. We show how factorization of the joint distribution into distinct observation factors and equalization factors greatly improves the detection capability of the structure. This is because the equalization function of the receiver is now chosen to be within the BP subgraph. A consequence of this factorization is that the computational complexity of BP-based equalization is exponential in the number of arriving signals components and the modulation order. This problem may be circumvented by approximating the domain of the symbol variables to be continuous random variables and the messages returned from BP-based demodulation and decoding as Gaussian distributions. These two approximations lead to Gaussian BP for the equalization factors which has a complexity independent of the modulation order and which is polynomial with respect to the number of interfering signal components. Thus, in the proposed receiver architecture the factor graph is divided into three subgraphs:

[1] Some of these works pre-date [35] and have used the terms variational message passing / sum-product algorithm (VMP-SP) and divergence minimization (DM) to refer to algorithms that similarly combine BP and the MF approximation.

[2] We use the term equalization to refer to the un-doing of both multi-stream and multi-user interference.

- MF subgraph: applied to the observation factors and parameter estimation. The MF algorithm can also serve as a link to BP based estimation of parameters.
- Gaussian BP: when the complexity of discrete BP is too high, Gaussian BP is applied to multi-signal detection. The symbol variables are treated as continuous random variables.
- Discrete BP: the sum-product algorithm is applied to demodulation and decoding where the factor functions have hard constraints and the variables are discrete.

The question that remains is how to approximate the discrete distributions passed from the symbol variables to the equalization factors with Gaussian distributions. A common approach in the technical literature is to match the mean and variance of the approximating Gaussian distribution with those of the discrete extrinsic or posterior distribution [39–44]. However, expectation propagation (EP) [45, 46] provides a theoretically justified approach to computing Gaussian approximations which has been shown to outperform extrinsic and posterior approximations [47,48]. Combined BP-EP has also been applied more broadly to OFDM channel estimation [177] and massive MIMO [178], but lacks the flexibility of the MF approximation (for example, to incorporate estimation of the noise variance [50]). In our receiver architecture, EP is used to link the Gaussian and discrete BP subgraphs. From these developments, we propose a probabilistic message passing receiver architecture for multi-signal detection which utilizes BP (both discrete and Gaussian), MF, and EP. Very recently a pre-print has appeared which combines BP, the MF approximation, and EP for the purpose of joint phase noise estimation and equalization of inter-symbol interference [179]. While [179] bears a conceptual similarity to our work, it differs substantially in that it does not handle multi-signal detection and assumes knowledge of the channel coefficients.

Our proposed receiver architecture is suitable for multi-signal detection in a variety of scenarios including co-channel interference (CCI), MIMO, multi-user MIMO, and non-orthogonal multiple access (NOMA). Reasonable complexity is maintained through the use of the MF approximation for the observation factors and Gaussian BP for the equalization factors. Parameter estimation may be included in the MF subgraph or, using MF as a link across the non-linear observation model, other algorithms such as BP or generalized approximate message passing (GAMP) [49] may be applied to estimation [50].

The contributions of our work are summarized as follows:

- Development of a factor graph construction for applying BP-MF to multi-signal detection
- Development of a low-complexity variant of the proposed construction combining BP (both Gaussian and discrete), the MF approximation, and EP.
- Derivation of a new MF-based time-domain channel estimator for OFDM signals.

In presenting these contributions, the chapter is organized as follows. We begin in Section 5.2 by providing an exemplary multi-signal system model. In Section 5.3, we develop a factor graph

construction which maintains the benefits of the MF approximation and BP in the case of the multi-signal model. In Section 5.4, we develop a receiver architecture based on MF, Gaussian BP, and Discrete BP. Although the architecture makes an approximation on the domain of the symbol variables, a solid theoretical foundation is maintained by applying EP to compute messages between the Gaussian and Discrete BP subgraphs. In Section 5.5, we apply the developed receiver architecture to a MIMO-OFDM signal and in Section 5.6 numerical results are provided that demonstrate the performance of the approach. Finally, the chapter is concluded in Section 5.7.

Notation

Column vectors and matrices are denoted by boldface lowercase and uppercase letters, respectively. We use $(\cdot)^{\mathrm{T}}$ and $(\cdot)^{\mathrm{H}}$ to denote transpose and conjugate transpose, respectively. The multivariate complex Gaussian pdf of $\mathbf{x}$ is denoted by $\mathcal{CN}(\mathbf{x}; \mu, \Sigma)$ where μ is a vector of the means and Σ is the covariance matrix. The size of set $\mathcal{A}$ is denoted by $|\mathcal{A}|$. The indicator function is denoted $\mathrm{I}(\cdot)$ and returns a value of 1 when the argument is true and 0 otherwise. Messages passed along the edges of a factor graph are denoted by sans-serif fonts where $\mathsf{m}_{f_A \to a}$ denotes the message from factor node f_A to variables node a and $\mathsf{n}_{a \to f_A}$ denotes the message from variables node a to factor node f_A.

5.2 Exemplary Multi-Signal System Model

Consider the reception of N signals where column vector $\mathbf{b}_i$ denotes the information bits corresponding to the ith signal. The information bits $\mathbf{b}_i$ are encoded with an error correction code to produce a vector of coded bits $\mathbf{c}_i$ and, subsequently, modulated using a digital phase-amplitude modulation. The resulting complex symbol sequence is denoted by $\mathbf{x}_i = [x_i(0), x_i(1), \ldots, x_i(K-1)]^{\mathrm{T}}$. The information bits, coded bits, and symbols for all signals are denoted by $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_N]$, $\mathbf{C} = [\mathbf{c}_1, \ldots, \mathbf{c}_N]$, and $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]$, respectively.

The kth observation y_k is comprised of N interfering signal components $\mathbf{x}(k) = [x_1(k), \ldots, x_N(k)]^{\mathrm{T}}$ and white Gaussian noise as given by

$$y(k) = \sum_{n=1}^{N} h_n x_n(k) + w(k) \tag{5.1}$$

where $\mathbf{h} = [h_1, \ldots, h_N]^{\mathrm{T}}$ are the channel coefficients corresponding to the N signals and $w(k)$ are independent identically distributed (iid) circularly-symmetric complex Gaussian random variables with variance γ^{-1}. Let the vector of observations be denoted by $\mathbf{y} = [y(0), \ldots, y(K-1)]^{\mathrm{T}}$ and the symbols associated with the kth observation be denoted by $\mathbf{x}(k) = [x_1(k), \ldots, x_N(k)]^{\mathrm{T}}$. The joint

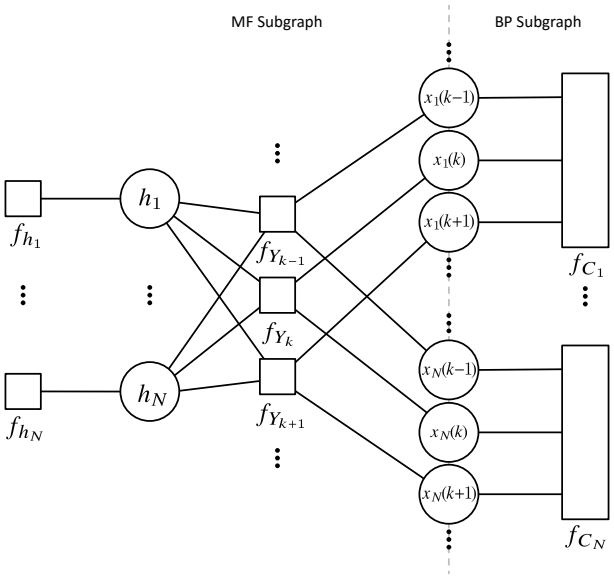

Figure 5.1: Factor graph of (5.2) based on original construction of BP-MF for multi-signal problems. In the graph, f_{h_i} labels the factor $p(h_i)$, f_{Y_k} labels the factor $p(y(k)|\mathbf{x}(k), \mathbf{h})$, and f_{C_i} labels the factors $p(\mathbf{x}_i|\mathbf{c}_i)p(\mathbf{c}_i|\mathbf{b}_i)p(\mathbf{b}_i)$.

distribution is factored as follows:

$$p(y, \mathbf{X}, \mathbf{C}, \mathbf{B}, \mathbf{h}) = \prod_{k=0}^{K-1} \underbrace{p(y(k)|\mathbf{x}(k), \mathbf{h})}_{f_{Y_k}} \left(\prod_{i=1}^{N} \underbrace{p(\mathbf{x}_i|\mathbf{c}_i)p(\mathbf{c}_i|\mathbf{b}_i)p(\mathbf{b}_i)}_{f_{C_i}} \right) \left(\prod_{i=1}^{N} \underbrace{p(h_i)}_{f_{h_i}} \right). \tag{5.2}$$

The factors $p(\mathbf{x}_i|\mathbf{c}_i)$ and $p(\mathbf{c}_i|\mathbf{b}_i)$ are hard constraints corresponding to the modulation and code constraints, respectively. Further factorization of these terms are available for common modulations and codes in the literature [79]. The factor graph of the joint distribution is shown in Fig. 5.1. This model is representative of interference corrupted observations due to co-channel interference or a non-orthogonal multiple access scheme. The developments presented with this model are applicable to other multi-signal or interference models. In comparison to Chapter 4, this signal model is very general and includes multi-antenna and multi-carrier scenarios.

The application of the BP-MF framework to the multi-signal scenario is determined by the partitioning of the factor graph into BP and MF subgraphs. The model of Fig. 5.1 shows the original approach to factor graph construction and partitioning [34, 37]. The partitioning falls along the symbols variables as shown by the dashed line in Fig. 5.1. As a formal definition of this partitioning we have the following sets:

$$\mathcal{A}_{\text{BP}} = \{f_{C_i} | i \in [1 : N]\} \tag{5.3}$$

$$\mathcal{A}_{\text{MF}} = \{f_{Y_k} | k \in [0 : K-1]\} \cup \{f_{h_n} | n \in [1 : N]\} \tag{5.4}$$

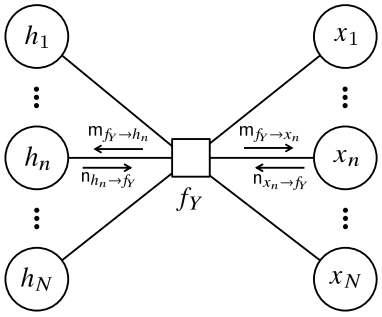

Figure 5.2: Factor graph of a *typical* interference-corrupted observation model based on the original BP-MF construction.

with the associated sets of variables given by

$$\mathcal{I}_{BP} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \cup \{\mathbf{c}_1, \ldots, \mathbf{c}_N\} \cup \{\mathbf{b}_1, \ldots, \mathbf{b}_N\} \tag{5.5}$$

$$\mathcal{I}_{MF} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \cup \{\mathbf{h}\}. \tag{5.6}$$

5.3 Factor Graph Construction for BP-MF

In this section, we develop a factor graph construction which enables efficient inference using the BP-MF framework. We begin by showing the limitations of the original approach, and then we develop a factor graph construction which is successful for multi-signal detection.

5.3.1 Original BP-MF Approach

Based on the BP-MF implementation found in the prior art, a *typical* observation factor is shown in Fig. 5.2 as a point of reference for the following work. The reference to subscript k has been removed in order to simply the notation. The factor function for f_Y in Fig. 5.2 is the likelihood function which is given by

$$\begin{aligned} p(y|\mathbf{x}, \mathbf{h}) &= \frac{\gamma}{\pi} \exp\left(-\gamma \left| y - \sum_{n=1}^{N} h_n x_n \right|^2\right) \\ &\propto \exp\left(-\gamma \left(-2\Re\left\{ y \sum_{n=1}^{N} h_n^* x_n^* \right\} + \sum_{n_1=1}^{N} \sum_{n_2=1}^{N} h_{n_1} h_{n_2}^* x_{n_1} x_{n_2}^* \right)\right). \end{aligned} \tag{5.7}$$

A summary of notation is provided in Table 5.1 for the parameters of the messages used in the following sections. The parameters of the messages have a subscript which identifies the associated variable and an arrow which identifies the associated message according to the direction it is passed in Figs. 5.2 and 5.3.

Table 5.1: Summary of notation

Messages Fig. 5.2	Messages Fig. 5.3	Mean	Covariance	Correlation
$\mathsf{m}_{f_Y \to h_n}$	$\mathsf{m}_{f_Q \to h_n}$	$\overleftarrow{\mu}_{h_n}$	$\overleftarrow{\sigma}^2_{h_n}$	
$\mathsf{n}_{h_n \to f_Y}$	$\mathsf{n}_{h_n \to f_Q}$	$\vec{\mu}_{h_n}$	$\vec{\sigma}^2_{h_n}$	$\vec{\rho}_{h_n}$
	$\mathsf{m}_{f_Q \to \mathbf{q}}$	$\vec{\vec{\mu}}_{\mathbf{q}}$	$\vec{\vec{\Sigma}}_{\mathbf{q}}$	$\vec{\vec{\mathbf{R}}}_{\mathbf{q}}$
	$\mathsf{n}_{\mathbf{q} \to f_Q}$	$\overleftarrow{\overleftarrow{\mu}}_{\mathbf{q}}$	$\overleftarrow{\overleftarrow{\Sigma}}_{\mathbf{q}}$	$\overleftarrow{\overleftarrow{\mathbf{R}}}_{\mathbf{q}}$
	$\mathsf{m}_{f_Y \to \mathbf{q}}$	$\overleftarrow{\mu}_{\mathbf{q}}$	$\overleftarrow{\Sigma}_{\mathbf{q}}$	$\overleftarrow{\mathbf{R}}_{\mathbf{q}}$
	$\mathsf{n}_{\mathbf{q} \to f_Y}$	$\vec{\mu}_{\mathbf{q}}$	$\vec{\Sigma}_{\mathbf{q}}$	$\vec{\mathbf{R}}_{\mathbf{q}}$
	$\mathsf{m}_{f_Y \to \mathbf{s}}$	$\vec{\mu}_{\mathbf{s}}$	$\vec{\Sigma}_{\mathbf{s}}$	$\vec{\mathbf{R}}_{\mathbf{s}}$
	$\mathsf{n}_{\mathbf{s} \to f_Y}$	$\overleftarrow{\mu}_{\mathbf{s}}$	$\overleftarrow{\Sigma}_{\mathbf{s}}$	$\overleftarrow{\mathbf{R}}_{\mathbf{s}}$
	$\mathsf{m}_{f_S \to \mathbf{s}}$	$\overleftarrow{\overleftarrow{\mu}}_{\mathbf{s}}$	$\overleftarrow{\overleftarrow{\Sigma}}_{\mathbf{s}}$	$\overleftarrow{\overleftarrow{\mathbf{R}}}_{\mathbf{s}}$
	$\mathsf{n}_{\mathbf{s} \to f_S}$	$\vec{\vec{\mu}}_{\mathbf{s}}$	$\vec{\vec{\Sigma}}_{\mathbf{s}}$	$\vec{\vec{\mathbf{R}}}_{\mathbf{s}}$
$\mathsf{m}_{f_Y \to x_n}$	$\mathsf{m}_{f_S \to x_n}$	$\vec{\mu}_{x_n}$	$\vec{\sigma}^2_{x_n}$	
$\mathsf{n}_{x_n \to f_Y}$	$\mathsf{n}_{x_n \to f_S}$	$\overleftarrow{\mu}_{x_n}$	$\overleftarrow{\sigma}^2_{x_n}$	$\overleftarrow{\rho}_{x_n}$

According to the MF approximation, the messages from the observation factor to the channel coefficients are given by

$$\mathsf{m}_{f_Y \to h_n}(h_n) = \exp\left(\int \cdots \int \sum_{i=1}^{N} \sum_{x_i} \ln(p(y|\mathbf{x}, \mathbf{h})) \mathsf{n}_{x_i \to f_Y}(x_i) \prod_{j \neq n} \mathsf{n}_{h_j \to f_Y}(h_j) dh_j \right). \tag{5.8}$$

The MF rule is less complex than BP because the expectation is taken on the argument of the exponential function due to $p(y|\mathbf{x}, \mathbf{h})$ being in the exponential family. After performing the expectations the message is proportional to a complex Gaussian distribution as given by

$$\begin{aligned} \mathsf{m}_{f_Y \to h_n}(h_n) &\propto \exp\left(-\gamma \left(\overleftarrow{\rho}_{x_n} |h_n|^2 - 2\Re \left\{ h_n^* \overleftarrow{\mu}^*_{x_n} \left(y - \sum_{n' \neq n} \vec{\mu}_{h_{n'}} \overleftarrow{\mu}_{x_{n'}} \right) \right\} \right) \right) \\ &\propto \mathcal{CN}\left(h_n; \overleftarrow{\mu}_{h_n}, \overleftarrow{\sigma}^2_{h_n} \right), \end{aligned} \tag{5.9}$$

where the mean and variance are $\overleftarrow{\mu}_{h_n} = \overleftarrow{\rho}^{-1}_{x_n} \overleftarrow{\mu}^*_{x_n} \left(y - \sum_{n' \neq n} \vec{\mu}_{h_{n'}} \overleftarrow{\mu}_{x_{n'}} \right)$ and $\overleftarrow{\sigma}^2_{h_n} = (\gamma \overleftarrow{\rho}_{x_n})^{-1}$, respectively. Similarly, the messages from the observation factor to the symbols are given by

$$\begin{aligned} \mathsf{m}_{f_Y \to x_n}(x_n) &\propto \exp\left(-\gamma \left(\vec{\rho}_{h_n} |x_n|^2 - 2\Re \left\{ x_n^* \vec{\mu}^*_{h_n} \left(y - \sum_{n' \neq n} \vec{\mu}_{h_{n'}} \overleftarrow{\mu}_{x_{n'}} \right) \right\} \right) \right) \\ &\propto \mathcal{CN}\left(x_n; \vec{\mu}_{x_n}, \vec{\sigma}^2_{x_n} \right), \end{aligned} \tag{5.10}$$

where the mean and variance are $\vec{\mu}_{x_n} = \vec{\rho}_{h_n}^{-1}\vec{\mu}_{h_n}^{*}\left(y - \sum_{n'\neq n}\vec{\mu}_{h_{n'}}\overleftarrow{\mu}_{x_{n'}}\right)$ and $\vec{\sigma}_{x_n}^{2} = \gamma\vec{\rho}_{h_n}$, respectively. We conclude from (5.9) and (5.10) that the MF approximation leads to an interference cancellation structure when computing the outgoing messages. The uncertainty in the variables (i.e., the variance) is not considered in the interference cancellation part; only the mean is used for terms $n' \neq n$ in (5.8) and (5.10). Interference cancellation, which only accounts for the mean, are known to be inferior to approaches which use both the mean and variance of the incoming messages [40, 42]. When estimation of the noise variance (or precision) is included in the MF framework, it naturally accounts for errors in the interference cancellation and, therefore, indirectly the uncertainty is accounted for. However, a single variance parameter does not effectively capture the variance of each individual symbol when performing cancellation. In simulations, we found that due to the interference cancellation structure, the MF approximation was a particularly poor choice for equalization in the presence of interference. This has motivated us to explore and propose an alternative model for multi-signal detection.

5.3.2 Joint Auxiliary Variables

The proposed factor graph model is based on separation of the equalization and channel estimation functions from the observations. The goal is to apply more effective BP for these functions while still maintaining the advantages of the MF approximation in regards to the non-linear observation model. To separate the equalization and channel estimation functions, we introduce two auxiliary variables: a joint symbol variable and a joint channel coefficient variable as defined by

$$\mathbf{s} \triangleq \mathbf{x} = [x_1, \ldots, x_N]^{\mathrm{T}} \text{ and } \mathbf{q} \triangleq \mathbf{h} = [h_1, \ldots, h_N]^{\mathrm{T}}, \tag{5.11}$$

respectively. This enables us to factor the joint distribution into distinct observation, equalization, and channel estimation factors as given by

$$p(y, \mathbf{x}, \mathbf{h}, \mathbf{s}, \mathbf{q}) = \underbrace{p(y|\mathbf{s}, \mathbf{q})}_{f_Y}\underbrace{p(\mathbf{s}|\mathbf{x})}_{f_S}\underbrace{p(\mathbf{q}|\mathbf{h})}_{f_Q}\prod_{i=1}^{N}p(x_i)\prod_{i=1}^{N}p(h_i). \tag{5.12}$$

The factor graph model is shown in Fig. 5.3.

With the introduction of joint auxiliary variables, the MF subgraph becomes a link between estimation of the channel coefficients and equalization of the symbols. Summary statistics (i.e., the first and second order moments) of the channel coefficient based on incoming message $\mathsf{n}_{\mathbf{q}\to f_Y}$ are used for detection of the data. Similarly, first and second order moments of the symbols based on the incoming message $\mathsf{n}_{\mathbf{s}\to f_Y}$ are used for channel estimation. The key difference is that joint distributions for all symbols (or channel coefficients) flow away from the observation nodes. Thus, the interference cancellation structure has been eliminated. As we will show, this allows us to perform equalization or channel estimation using belief propagation which is more successful. These points are demonstrated by the following message derivations.

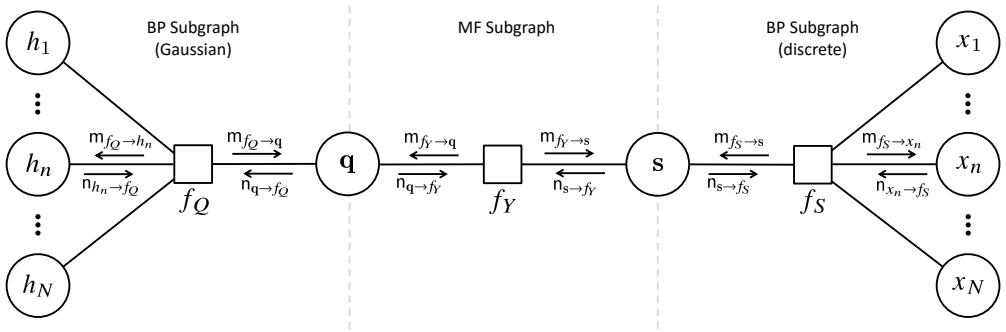

Figure 5.3: Factor graph of a *typical* interference corrupted observation model based on the proposed BP-MF construction with auxiliary variables. In the graph, f_Q labels the factor $p(\mathbf{q}|h_1, \ldots, h_N)$ and f_S labels the factor $p(\mathbf{s}|x_1, \ldots, x_N)$.

The likelihood function is expressed in terms of the vector notation defined in (5.11) and is given by

$$p(y|\mathbf{s}, \mathbf{q}) \propto \exp\left(-\gamma \left|y - \mathbf{q}^{\mathrm{T}}\mathbf{s}\right|^2\right). \tag{5.13}$$

The message from the observation to the auxiliary symbol variable is derived as follows:

$$\begin{aligned}
&\mathsf{m}_{f_Y \to \mathbf{s}}(\mathbf{s}) \\
&\quad \propto \exp\left\{-\gamma\left(-y\mathbf{s}^{\mathrm{H}}\vec{\mu}_{\mathbf{q}}^{*} - y^{*}\vec{\mu}_{\mathbf{q}}^{\mathrm{T}}\mathbf{s} + \mathbf{s}^{\mathrm{H}}\vec{\mathbf{R}}_{\mathbf{q}}\mathbf{s}\right)\right\} && (5.14)\\
&\quad \propto \mathcal{CN}\left(\mathbf{s}; \vec{\mu}_{\mathbf{s}}, \vec{\Sigma}_{\mathbf{s}}\right), && (5.15)
\end{aligned}$$

where $\vec{\mu}_{\mathbf{s}} = \vec{\mathbf{R}}_{\mathbf{q}}^{-1} y \vec{\mu}_{\mathbf{q}}^{*}$ and $\vec{\Sigma}_{\mathbf{s}} = \left(\gamma \vec{\mathbf{R}}_{\mathbf{q}}\right)^{-1}$. The message from the observation to the auxiliary channel variable is derived similarly as given by

$$\begin{aligned}
&\mathsf{m}_{f_Y \to \mathbf{q}}(\mathbf{q}) \\
&\quad \propto \exp\left\{-\gamma\left(-y\mathbf{q}^{\mathrm{H}}\overleftarrow{\mu}_{\mathbf{s}}^{*} - y^{*}\overleftarrow{\mu}_{\mathbf{s}}^{\mathrm{T}}\mathbf{q} + \mathbf{q}^{\mathrm{H}}\vec{\mathbf{R}}_{\mathbf{s}}\mathbf{q}\right)\right\} \\
&\quad \propto \exp\left\{-\gamma\left(\mathbf{q} - \overleftarrow{\mathbf{R}}_{\mathbf{s}}^{-1} y \overleftarrow{\mu}_{\mathbf{s}}^{*}\right)^{\mathrm{H}} \overleftarrow{\mathbf{R}}_{\mathbf{s}}\left(\mathbf{q} - \overleftarrow{\mathbf{R}}_{\mathbf{s}}^{-1} y \overleftarrow{\mu}_{\mathbf{s}}^{*}\right)\right\} \\
&\quad \propto \mathcal{CN}\left(\mathbf{q}; \overleftarrow{\mu}_{\mathbf{q}}, \overleftarrow{\Sigma}_{\mathbf{q}}\right), && (5.16)
\end{aligned}$$

where $\overleftarrow{\mu}_{\mathbf{q}} = \overleftarrow{\mathbf{R}}_{\mathbf{s}}^{-1} y \overleftarrow{\mu}_{\mathbf{s}}^{*}$ and $\overleftarrow{\Sigma}_{\mathbf{q}} = \left(\gamma \overleftarrow{\mathbf{R}}_{\mathbf{s}}\right)^{-1}$. Descriptions for the parameters can be found in Table 5.1.

5.3.3 BP-Based Channel Estimation

In contrast to the interference cancellation structure of the MF approximation, Gaussian BP follows the structure of LMMSE filtering where the incoming messages ($\mathsf{n}_{h_n \to f_Q}(h_n)$) are treated as prior

distributions. We derive the messages associated with the channel estimation BP subgraph in Section 5.8. Rather than rely only on "soft" symbol estimates, the proposed algorithm makes use of first and second order moments of the symbols as a result of the MF approximation.

5.3.4 BP-Based Equalization

Because the joint auxiliary symbol variable has a discrete domain, the message $\mathsf{m}_{f_Y \to \mathbf{s}}(\mathbf{s})$ is a discrete distribution which is computed by evaluating (5.14) for all vectors $\mathbf{s}$. The sum-product rule is applied to messages from the factor f_S to the neighboring variables. In this context, discrete BP (or the sum-product algorithm) is a (local) joint MAP detector where incoming messages from the symbols are treated as prior distributions. We derive the messages associated with the equalization BP subgraph in Section 5.9.

5.4 Combined BP-MF-EP Receiver Architecture

In the previous section, we applied the BP-MF framework to detection of multiple signals. The advancement in our work is a factor graph model that maintains the benefits of the MF approximation and enables BP to be used for equalization and channel estimation. The complexity of BP for discrete variables is a function of the variable's domain size and number of signal components in the observation. For example, if an observation is comprised of N signal components with domain $\mathcal{X}$, the complexity of BP is $\mathcal{O}(|\mathcal{X}|^N)$. In other words, it is exponential in the number of components and number of bits per symbol (i.e., bits per signal component). While this is not prohibitive for detection of several signals with low-order modulation (for example, 2 signals with QPSK modulation has a complexity order of 16), it is a barrier when more signal components are present or have high-order modulations (for example, 4 signals with 16 QAM modulation has a complexity order of 65 536)

We develop a reduced complexity receiver for the proposed model of Section 5.3 by making the assumption that the symbols and the joint auxiliary symbol variable are continuous random variables and that the messages returned from demodulation and decoding are Gaussian distributed. With these assumptions, Gaussian BP is applied to the equalization factor f_S and associated variables. The Gaussian BP algorithm for equalization is identical to Gaussian BP for channel estimation. As with channel estimation, the Gaussian BP equalizer has the form of LMMSE filtering with prior information and makes use of first and second order moments of the channel coefficients according to the MF messages. Gaussian BP has been applied to equalization in iterative receivers before and has been shown to be equivalent to LMMSE filtering where feedback from the decoder is treated as prior information [117].

The question that remains is how to compute Gaussian distributed messages from the discrete messages returned from demodulation and decoding. In similar applications, a Gaussian distribution

whose mean and variance match the mean and variance of the extrinsic distribution has been proposed [41]. In some cases it was found to be more effective to match the mean and variance to the posterior discrete distribution [39]. Senst and Ascheid provided a theoretically justified approach to computing the Gaussian messages based on EP [47]. Their work also provides insight into why posterior distributions are more effective than extrinsic distributions.

5.4.1 Gaussian Approximation

The joint auxiliary symbol variable is approximated as having a continuous domain. Similarly, the symbols $x_1, \ldots, x_N$ are approximated as having continuous domains for factor f_S (i.e., $p(\mathbf{s}|x_1, \ldots, x_N)$) and maintain a discrete domain in the factors $f_{C_1}, \ldots, f_{C_N}$. The messages from the symbols to the equalization factors $\mathsf{n}_{x_n \to f_S}(x_n)$ are now continuous and are approximated as Gaussian distributions. Gaussian BP is used to compute messages involving the equalization factor in a similar manner to Section 5.3.3 and Section 5.8.

5.4.2 Expectation Propagation

From Fig. 5.4, messages $\mathsf{m}_{f_S \to x_n}(x_n)$ and $\mathsf{n}_{x_n \to f_S}(x_n)$ are Gaussian distributions while $\mathsf{m}_{f_{C_n} \to x_n}(x_n)$ and $\mathsf{n}_{x_n \to f_{C_n}}(x_n)$ are discrete distributions. Computing $\mathsf{n}_{x_n \to f_{C_n}}(x_n)$ from $\mathsf{m}_{f_S \to x_n}(x_n)$ is straightforward: the Gaussian distribution is evaluated for each value in the domain of the symbol. In order to approximate discrete distribution $\mathsf{m}_{f_{C_n} \to x_n}(x_n)$ with Gaussian distribution $\mathsf{n}_{x_n \to f_S}(x_n)$, we apply EP.

EP is implemented by first computing the exact belief from the discrete distributions as given by

$$b(x_n) = \mathsf{m}_{f_{C_n} \to x_n}(x_n) \mathsf{n}_{x_n \to f_{C_n}}(x_n). \tag{5.17}$$

Subsequently, the belief is approximated with a Gaussian distribution as given by

$$\tilde{b}(x_n) = \mathcal{CN}(x_n; \mu_{x_n}, \sigma^2_{x_n}), \tag{5.18}$$

where μ_{x_n} and $\sigma^2_{x_n}$ *without arrows* denote parameters of the belief computed from (5.17). Finally, the parameters of the Gaussian distribution for $\mathsf{n}_{x_n \to f_S}(x_n)$ are computed by dividing $\tilde{b}(x_n)$ by $\mathsf{m}_{f_S \to x_n}(x_n)$. Thus, the mean and variance of $\mathsf{n}_{x_n \to f_S}(x_n)$ are given by

$$\overleftarrow{\sigma}^2_{x_n} = \left(\frac{1}{\sigma^2_{x_n}} - \frac{1}{\vec{\sigma}^2_{x_n}} \right)^{-1} \tag{5.19}$$

and

$$\overleftarrow{\mu}_{x_n} = \overleftarrow{\sigma}^2_{x_n} \left(\frac{\mu_{x_n}}{\sigma^2_{x_n}} - \frac{\vec{\mu}_{x_n}}{\vec{\sigma}^2_{x_n}} \right), \tag{5.20}$$

respectively. It is possible that the variance computed in (5.19) is negative. In this case, we apply the solution used in [47] and approximate the message by the Gaussian belief, i.e., we set $\overleftarrow{\mu}_{x_n} = \mu_{x_n}$ and $\overleftarrow{\sigma}^2_{x_n} = \sigma^2_{x_n}$.

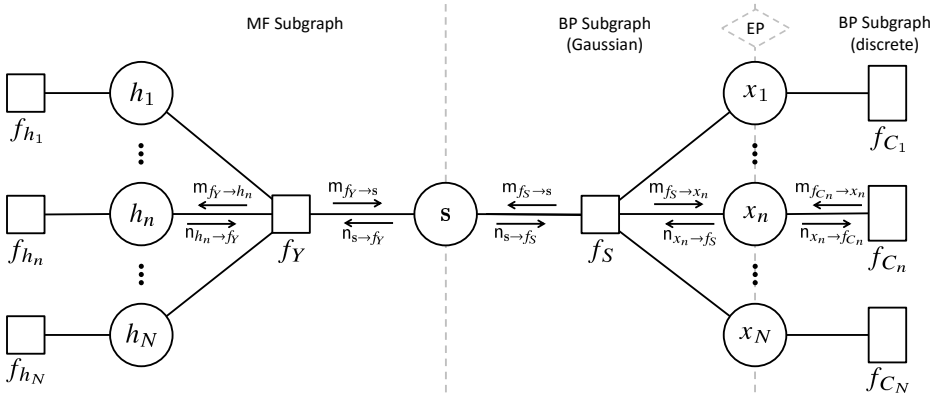

Figure 5.4: Factor graph of a *typical* observation based on the proposed BP-MF-EP receiver architecture for low-complexity implementation.

5.4.3 Receiver Architecture

In summary we propose a receiver architecture for multi-signal detection which combines Gaussian and discrete BP, the MF approximation, and EP. The architecture is built upon the factor graph model of the joint distribution including choice of auxiliary variables and partitioning of the graph into subgraphs. We divide the graph into three regions as follows:

- MF subgraph including the observation and channel estimation factors and corresponding variables.
- Gaussian BP subgraph including the equalization factors, and
- Discrete BP subgraph including the modulation and coding constraint factors.

Channel estimation can be separated into its own subgraph applying Gaussian BP or GAMP. Additionally, channel estimation can be accomplished using the EM algorithm as a special case of the MF approximation. A diagram of the receiver architecture is shown in Fig. 5.4 In the next section, we demonstrate this receiver architecture by applying it to MIMO-OFDM.

5.5 Application to MIMO-OFDM

In this section, the proposed receiver architecture is applied to reception of MIMO-OFDM signals. This model can be used to accomplish multi-signal detection in general (e.g., single antenna multiuser detection or multiuser MIMO schemes).

5.5.1 System Model

Consider a MIMO-OFDM transmission scheme which utilizes N transmit antennas, M receive antennas, and K subcarriers. The information bits, coded bits, and symbols for the ith stream transmitted on the ith antenna (similarly, the ith user) are denoted by $\mathbf{b}_i$, $\mathbf{c}_i$, and $\mathbf{x}_i$, respectively. The symbols transmitted on the kth subcarrier across all antennas are collected into vector $\mathbf{x}(k) = [x_1(k), \ldots, x_N(k)]^{\mathrm{T}}$.

The multipath channel between each pair of transmitter and receiver antennas is modeled with a conventional tapped delay line with L taps spaced at the OFDM symbol sample rate. The channel coefficients associated with the nth transmitter antenna and the mth receiver antenna are denoted by the $L\times 1$ vector $\mathbf{h}_{mn} = [h_{mn}(0), h_{mn}(1), \ldots, h_{mn}(L-1)]^{\mathrm{T}}$. The collection of the channel coefficients for all pairs of transmitter and receiver antennas is given by

$$\mathbf{h} = \left[\left[\mathbf{h}_{11}^{\mathrm{T}}, \ldots, \mathbf{h}_{M1}^{\mathrm{T}}\right], \ldots, \left[\mathbf{h}_{1N}^{\mathrm{T}}, \ldots, \mathbf{h}_{MN}^{\mathrm{T}}\right]\right]^{\mathrm{T}}.$$

The channel coefficients $h_{mn}(l)$ for all $m = 1, \ldots, M$, $n = 1, \ldots, N$, and $l = 0, \ldots, L-1$ are assumed to be independent.

The frequency domain channel coefficients for the K subcarriers are obtained through the Fourier transform. We define a $K \times L$ DFT matrix $\mathbf{D}$ where the k,lth element is given by $d_{kl} = e^{-j2\pi kl/K}$. The frequency domain channel coefficients are defined as given by

$$\tilde{h}_{mn}(k) = \sum_{l=0}^{L-1} h_{mn}(l) d_{kl} \tag{5.21}$$

for all $m = 1, \ldots, M$, $n = 1, \ldots, N$, and $k = 0, \ldots, K-1$. The $M \times N$ MIMO channel matrix for the kth subcarrier is given by

$$\tilde{\mathbf{H}}(k) = \begin{bmatrix} \tilde{h}_{11}(k) & \tilde{h}_{12}(k) & \cdots & \tilde{h}_{1N}(k) \\ \tilde{h}_{21}(k) & \tilde{h}_{22}(k) & & \vdots \\ \vdots & & \ddots & \\ \tilde{h}_{M1}(k) & \cdots & & \tilde{h}_{MN}(k) \end{bmatrix}. \tag{5.22}$$

We define both scalar and vector forms for the received signal as follows. The $M \times 1$ received signal vector for the kth subcarrier is given by

$$\mathbf{y}(k) = \tilde{\mathbf{H}}(k)\mathbf{x}(k) + \mathbf{w}(k) \quad \forall k = 0, 1, \ldots, K-1, \tag{5.23}$$

where and $\mathbf{w}(k)$ is a $M \times 1$ vector of iid circularly symmetric complex Gaussian random variables representing noise. The per antenna noise precision is denoted by γ (i.e., the noise variance is given by $1/\gamma$). The scalar form of the received signal is given by

$$y_m(k) = \sum_{n=1}^{N} x_n(k) \sum_{l=0}^{L-1} h_{mn}(l) d_{kl} + w_m(k), \tag{5.24}$$

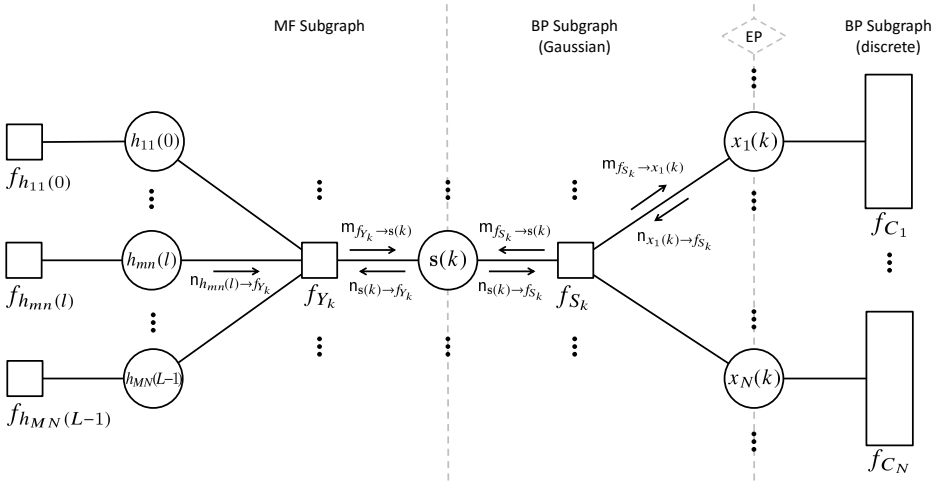

Figure 5.5: Factor graph of (5.25) with auxiliary variables $\mathbf{s}(k)$ enabling separation between channel estimation and detection.

where $y_m(k)$ and $w_m(k)$ are the mth elements of $\mathbf{y}(k)$ and $\mathbf{w}(k)$, respectively. The concatenation of the received signal vectors and channel matrices for all subcarriers are denoted as $\mathbf{y} = [\mathbf{y}(0)^{\mathrm{T}}, \ldots, \mathbf{y}(K-1)^{\mathrm{T}}]^{\mathrm{T}}$ and $\tilde{\mathbf{H}} = [\tilde{\mathbf{H}}(0)^{\mathrm{T}}, \ldots, \tilde{\mathbf{H}}(K-1)^{\mathrm{T}}]^{\mathrm{T}}$, respectively.

5.5.2 Factor Graph

We introduce an auxiliary variable $\mathbf{s}(k)$ for each observation $\mathbf{y}(k)$ which represents the joint symbol vector for that observation (i.e., $x_1(k), \ldots, x_N(k)$). The collection of all auxiliary variables is denoted as $\mathbf{s} = [\mathbf{s}(0)^{\mathrm{T}}, \ldots, \mathbf{s}(K-1)^{\mathrm{T}}]^{\mathrm{T}}$. The joint distribution is factored as follows:

$$p(\mathbf{y}, \mathbf{s}, \mathbf{x}_1, \ldots, \mathbf{x}_N, \mathbf{c}_1, \ldots, \mathbf{c}_N, \mathbf{b}_1, \ldots, \mathbf{b}_N, \mathbf{h}) = \prod_{k=0}^{K-1} p(\mathbf{y}(k)|\mathbf{s}(k), \mathbf{h}) p(\mathbf{s}(k)|x_1(k), \ldots, x_N(k)) \prod_{i=1}^{N} p(\mathbf{x}_i, \mathbf{c}_i, \mathbf{b}_i) \prod_{m=1}^{M} \prod_{n=1}^{N} \prod_{l=0}^{L-1} p(h_{mn}(l)). \quad (5.25)$$

The distributions $p(\mathbf{x}_i, \mathbf{c}_i, \mathbf{b}_i)$ may be further factored based on the modulation and code constraints which has been considered extensively in past work [79]. The factor graph of the joint distribution is shown in Fig. 5.5.

Two receiver architectures are constructed by defining the partitioning of the factor into subgraphs according to the message passing algorithm to be applied. First, the *exact* implementation applies BP-MF as constructed in Section 5.3. The exact implementation is defined by the following

subgraphs:

$$\mathcal{A}_{\text{BP}} = \{f_{S_k} | k \in [0 : K-1]\} \cup \{f_{C_i} | i \in [1 : N]\} \tag{5.26}$$
$$\mathcal{A}_{\text{MF}} = \{f_{Y_k} | k \in [0 : K-1]\} \cup \{f_{h_{mn}(l)} | m \in [1 : M], n \in [1 : N], l \in [0 : L-1]\} \tag{5.27}$$
$$\mathcal{I}_{BP} = \{\mathbf{s}(k) | k \in [0 : K-1]\} \cup \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \cup \{\mathbf{c}_1, \ldots, \mathbf{c}_N\} \cup \{\mathbf{b}_1, \ldots, \mathbf{b}_N\} \tag{5.28}$$
$$\mathcal{I}_{MF} = \{\mathbf{s}(k) | k \in [0 : K-1]\} \cup \{\mathbf{h}\}. \tag{5.29}$$

Second, the approximate implementation applies BP-MF-EP as constructed in Section 5.4. The approximate implementation is defined by the following subgraphs:

$$\mathcal{A}_{\text{dBP}} = \{f_{C_i} | i \in [1 : N]\} \tag{5.30}$$
$$\mathcal{A}_{\text{GBP}} = \{f_{S_k} | k \in [0 : K-1]\} \tag{5.31}$$
$$\mathcal{A}_{\text{MF}} = \{f_{Y_k} | k \in [0 : K-1]\} \cup \{f_{h_{mn}(l)} | m \in [1 : M], n \in [1 : N], l \in [0 : L-1]\} \tag{5.32}$$
$$\mathcal{I}_{dBP} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \cup \{\mathbf{c}_1, \ldots, \mathbf{c}_N\} \cup \{\mathbf{b}_1, \ldots, \mathbf{b}_N\} \tag{5.33}$$
$$\mathcal{I}_{GBP} = \{\mathbf{s}(k) | k \in [0 : K-1]\} \cup \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \tag{5.34}$$
$$\mathcal{I}_{MF} = \{\mathbf{s}(k) | k \in [0 : K-1]\} \cup \{\mathbf{h}\}, \tag{5.35}$$

where $\mathcal{A}_{\text{dBP}}$ denotes the discrete BP subgraph and $\mathcal{A}_{\text{GBP}}$ denotes the Gaussian BP subgraph. We handle channel estimation differently than previous work [35, 50, 90, 173] by estimating the time-domain channel taps with the MF approximation. Derivations for the messages within the MF subgraph are provided in Section 5.10.

5.6 Numerical Results

In this section, we provide numerical results for the MIMO-OFDM receiver using Monte Carlo simulation in order to validate the proposed probabilistic receiver architecture. The simulation parameters are summarized in Table 5.2. Although internal loops are present within the factor graph, we do not perform any sub-iterations within a full iteration of the receiver.

We present simulation results for four receiver algorithms. In all four cases, we use our proposed MF-based MIMO-OFDM channel estimation developed in Section 5.5 and Section 5.10. Diffuse Gaussian prior distributions are chosen for the channel coefficients (where the prior variance is several orders of magnitude greater than the channel coefficient power). The algorithms differ in how the BP-MF framework is applied to multi-signal detection. A description of each algorithm and the complexity of the equalization function is provided as follows:

(a) *BP-MF original*: the BP-MF implementation found in the prior art in which multi-signal detection is included in the MF subgraph (e.g., [34, 37]). The equalization function has a computational complexity $O(N)$ as shown in (5.10).

Table 5.2: Summary of the MIMO-OFDM simulation parameters

Parameter	Description
Transmit antennas (N)	4
Receive antennas (M)	4
Subcarriers (K)	180
OFDM Symbols/packet	7
Coding	1/2 PCCC
Modulation	QPSK
Reference Signals	3GPP LTE (antenna ports 0–3)
Channel	$L = 10$ taps/antenna pair (iid)

(b) *BP-MF exact*: our proposed BP-MF implementation in which discrete BP is applied to multi-signal detection (subgraphs defined in (5.26)–(5.29)). The computational complexity of discrete BP for equalization of multiple signals is $O(|\mathcal{X}|^N)$ where $|\mathcal{X}|$ is the modulation order.

(c) *BP-MF-EP*: our proposed low-complexity BP-MF implementation with Gaussian BP and Gaussian approximations based on EP (subgraphs defined in (5.30)–(5.35)). The computational complexity is $O(N^3)$ due to the matrix inversion required for equalization.

(d) *BP-MF approximate (ext)*: for comparison purposes a BP-MF implementation with Gaussian BP equalization and Gaussian approximations based on extrinsic distributions (subgraphs defined in (5.30)–(5.35)). The computational complexity is also $O(N^3)$.

The performance with known channel coefficients and joint MAP (JMAP) detection is also simulated to provide a lower bound. The JMAP receiver with known channel coefficients applies discrete BP and is identical to the *BP-MF exact* receiver in its equalization, demodulation, and decoding functions.

The bit error rate (BER) performance is shown with respect to SNR and with respect to the number of receiver iterations in Figs. 5.6 and 5.7, respectively. The results support our claim that the interference cancellation structure of the MF approximation (used in the original implementation) is not ideal for handling multi-signal detection. This is because variance of a symbol's messages (i.e., the degree of uncertainty about a particular symbol's value) is not accounted for when applying the MF approximation to multi-signal detection. Applying BP to the MIMO equalization task (as done in the exact implementation of our proposed BP-MF construction) yields more than 1 dB improvement in performance and is about 1 dB away from the best achievable performance with perfect parameter knowledge. However, our exact implementation of BP-MF comes at the cost of a computational complexity which is exponential in the number of signals. The performance of the receivers which use Gaussian BP lies in between these two. Specifically, Gaussian BP with extrinsic-based Gaussian approximations does not significantly improve performance versus the

original BP-MF algorithm. On the other hand, because EP is effective in computing Gaussian messages from the discrete distributions passed to the Gaussian BP subgraph, the *BP-MF-EP* receiver performs very close to the *BP-MF exact* receiver with discrete BP. Thus, with polynomial complexity order (like *BP-MF original*), we are able to achieve nearly the performance of the exact implementation. In fact, we observe in Fig. 5.7, that for iterations 1–13, *BP-MF-EP* slightly outperforms the exact implementation. This is likely due to the fact that *BP-MF-EP* does not follow a strict use of extrinsic information which leads to faster convergence but an increased chance of "hardening" the distributions toward the wrong decisions (as seen in the cross-over of their BER curves in Fig. 5.7). In general, the value of MIMO detection is clearly seen in comparison to a single antenna, uncoded 16QAM transmission which achieves the same rate as the 4×4 MIMO system.

The mean square error (MSE) channel estimation performance is shown in Figs. 5.8 and 5.9. In these figures, the Cramér-Rao lower bound (CRLB) is shown for the case in which both the reference signals and the data symbols are known. Thus, it provides a lower bound on the MSE that we would expect to achieve through iterative estimation. The only unexpected result is that, at lower SNR, the original BP-MF implementation outperforms Gaussian BP with extrinsic-based Gaussian approximations. This is further evidence that basing the Gaussian messages on the extrinsic distributions is a poor approximation. As all receiver algorithms perform channel estimation in the same way, the difference in channel estimation performance is a consequence of the quality of the information about the data in each receiver. The BP-MF-EP receiver again converges more quickly than the exact BP-MF receiver. However, both receiver algorithms converge to the same estimation performance after about 12 iterations.

5.7 Conclusion

In this chapter, we have developed a probabilistic receiver architecture for detection of multiple signals based on the BP-MF framework. By introducing auxiliary variables into the factor graph model, we maintain the benefits of the MF approximation while avoiding an undesirable interference cancellation structure. In scenarios in which the complexity of discrete BP for equalization is prohibitive, we proposed Gaussian BP for multi-signal detection and a combined BP-MF-EP message passing algorithm. The proposed low-complexity algorithm is shown to perform nearly as well as the exact implementation for a MIMO-OFDM signal detection. As a result of this work, we have developed a probabilistic receiver architecture with strong theoretical justification which can be applied to multi-signal detection and, in general, detection in the presence of interference. We have also developed a new MF-based time-domain channel estimation approach for MIMO-OFDM. While we have focused on MF-based channel estimation, the factor graph construction also enables BP, GAMP, or EM to be applied to channel estimation.

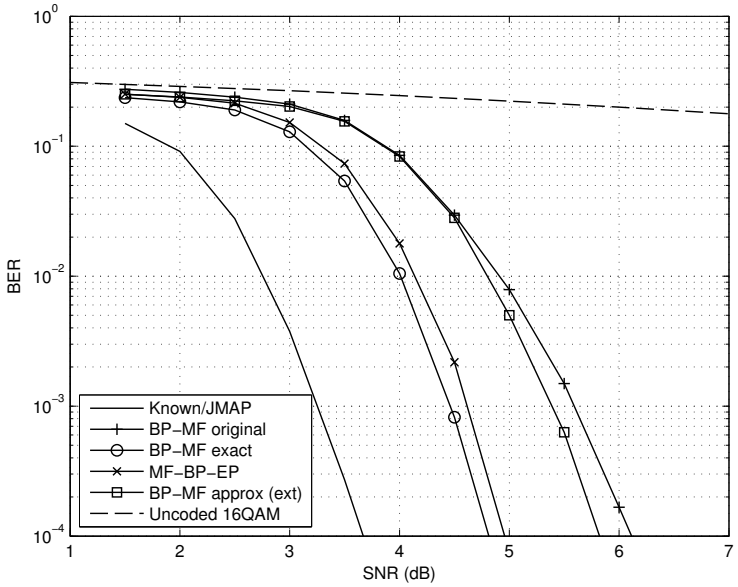

Figure 5.6: BER vs. SNR per antenna with 20 iterations.

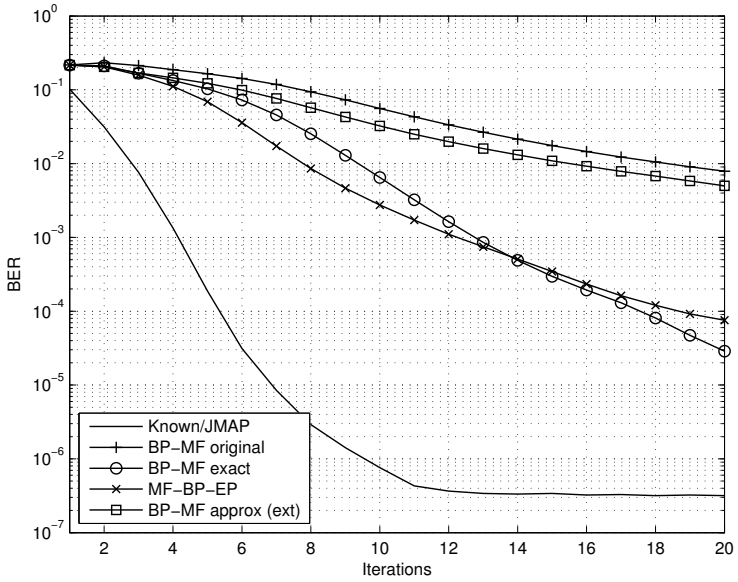

Figure 5.7: BER vs. iteration for SNR = 5 dB per antenna.

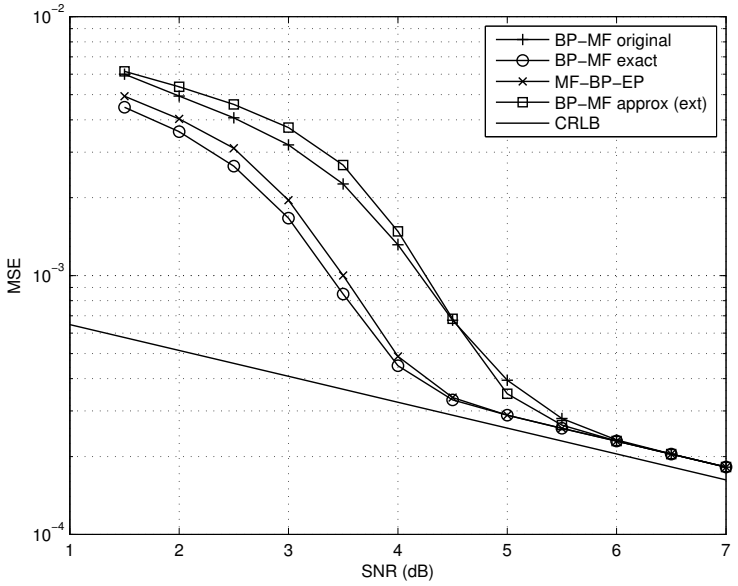

Figure 5.8: Channel estimation MSE vs. SNR per antenna with 20 iterations.

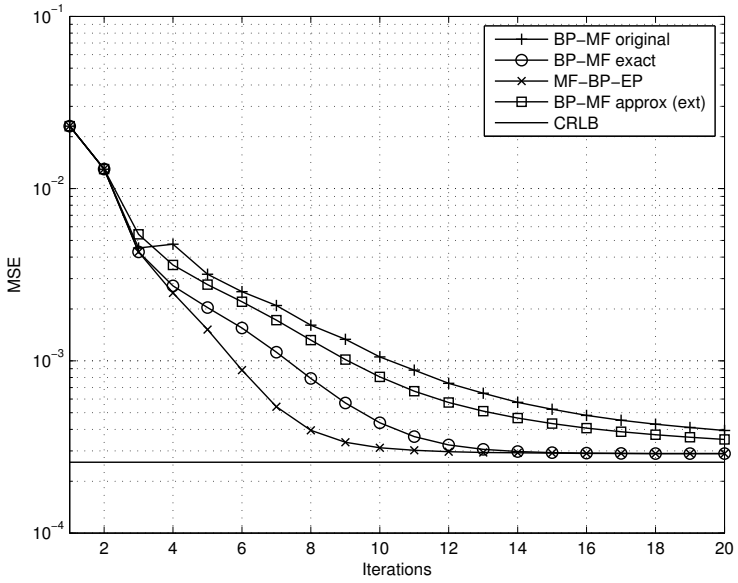

Figure 5.9: Channel estimation MSE vs. iteration for SNR = 5 dB per antenna.

5.8 Gaussian BP for Channel Estimation

The factor f_Q is within a BP subgraph. Thus, the message passed from the joint auxiliary channel variable to this factor is an extrinsic message as given by

$$\mathsf{n}_{\mathbf{q}\to f_Q}(\mathbf{q}) = \mathsf{m}_{f_Y\to\mathbf{q}}(\mathbf{q}). \tag{5.36}$$

Similarly, the messages from the channel coefficient variables to f_Q are extrinsic messages which carry the prior distributions for the channel coefficients. The factor f_Q enforces equality between the channel coefficients and the corresponding terms within the joint auxiliary variable. The factor function is given by the following hard constraint:

$$p(\mathbf{q}|\mathbf{h}) = \prod_{i=1}^{N} \mathrm{I}(q_i = h_i). \tag{5.37}$$

When the prior distributions are Gaussian (e.g., for Rayleigh fading channels), Gaussian BP is used to perform the computations for this factor node.

The joint auxiliary variable $\mathbf{q}$ is a concatenation of the channel coefficients. Thus, the BP message from f_Q to the joint auxiliary variable is given by a concatenation of the input messages from the channel coefficients. That is, the message is given by

$$\mathsf{m}_{f_Q\to\mathbf{q}}(\mathbf{q}) = \mathcal{CN}\left(\mathbf{q}; \vec{\vec{\mu}}_{\mathbf{q}}, \vec{\vec{\Sigma}}_{\mathbf{q}}\right), \tag{5.38}$$

where

$$\vec{\vec{\mu}}_{\mathbf{q}} = [\vec{\mu}_{h_1}, \ldots, \vec{\mu}_{h_N}]^{\mathrm{T}} \tag{5.39}$$

and

$$\vec{\vec{\Sigma}}_{\mathbf{q}} = \mathrm{diag}\left(\vec{\sigma}^2_{h_1}, \ldots, \vec{\sigma}^2_{h_N}\right). \tag{5.40}$$

The message from f_Q to the channel coefficients is computed according to the sum-product rule. In order to work with vector notation, the incoming messages from the channel coefficients are combined into a mean and covariance matrix as given by

$$\mu_{\sim h_n} = [\vec{\mu}_{h_1}, \ldots, \vec{\mu}_{h_{n-1}}, 0, \vec{\mu}_{h_{n+1}}, \ldots, \vec{\mu}_{h_N}]^{\mathrm{T}} \tag{5.41}$$

and

$$V_{\sim h_n} = \mathrm{diag}\left(\frac{1}{\vec{\sigma}^2_{h_1}}, \ldots, \frac{1}{\vec{\sigma}^2_{h_{n-1}}}, 0, \frac{1}{\vec{\sigma}^2_{h_{n+1}}}, \ldots, \frac{1}{\vec{\sigma}^2_{h_N}}\right), \tag{5.42}$$

where h_n is excluded due to the sum-product rule. With these definitions, the sum-product compu-

tation is as follows:

$$\begin{aligned}\mathsf{m}_{f_Q\to h_n}(h_n) &= \int\cdots\int p(\mathbf{q}|\mathbf{h})\mathsf{n}_{\mathbf{q}\to f_Q}(\mathbf{q})\prod_{i\neq n}\mathsf{n}_{h_i\to f_Q}(h_i)dh_i d\mathbf{q}.\\ &\propto \int\cdots\int \exp\left\{-\left(\mathbf{h}-\overleftarrow{\mu}_{\mathbf{q}}\right)^{\mathrm{H}}\overleftarrow{\Sigma}_{\mathbf{q}}^{-1}\left(\mathbf{h}-\overleftarrow{\mu}_{\mathbf{q}}\right)\right\}\\ &\quad\cdot\exp\left\{-\left(\mathbf{h}-\mu_{\sim h_n}\right)^{\mathrm{H}}V_{\sim h_n}\left(\mathbf{h}-\mu_{\sim h_n}\right)\right\}\prod_{i\neq n}dh_i\\ &\propto \mathcal{CN}\left(h_n;\overleftarrow{\mu}_{h_n},\overleftarrow{\sigma}^2_{h_n}\right),\end{aligned} \tag{5.43}$$

where

$$\overleftarrow{\sigma}^2_{h_n} = \left[\left(\overleftarrow{\Sigma}_{\mathbf{q}}^{-1}+V_{\sim h_n}\right)^{-1}\right]_{n,n} \tag{5.44}$$

and

$$\overleftarrow{\mu}_{h_n} = \left[\left(\overleftarrow{\Sigma}_{\mathbf{q}}^{-1}+V_{\sim h_n}\right)^{-1}\left(\overleftarrow{\Sigma}_{\mathbf{q}}^{-1}\overleftarrow{\mu}_{\mathbf{q}}+V_{\sim h_n}\mu_{\sim h_n}\right)\right]_n. \tag{5.45}$$

An efficient implementation is to compute the joint posterior, marginalize, and remove the input distribution to obtain marginal extrinsic messages for each channel coefficient. The joint posterior is computed once for the messages to all channel coefficient variables and, therefore, a single matrix inversion is required.

Finally, the message from the joint channel auxiliary variable to the observation factor is the posterior distribution for $\mathbf{q}$ since f_Y is in the MF subgraph. The posterior is given by

$$\begin{aligned}\mathsf{n}_{\mathbf{q}\to f_Y}(\mathbf{q}) &= \mathsf{m}_{f_Q\to\mathbf{q}}(\mathbf{q})\mathsf{m}_{f_Y\to\mathbf{q}}(\mathbf{q})\\ &\propto \exp\left\{-\left(\mathbf{h}-\vec{\mu}_{\mathbf{q}}\right)^{\mathrm{H}}\vec{\Sigma}_{\mathbf{q}}^{-1}\left(\mathbf{h}-\vec{\mu}_{\mathbf{q}}\right)\right\}\\ &\quad\cdot\exp\left\{-\left(\mathbf{h}-\overleftarrow{\mu}_{\mathbf{q}}\right)^{\mathrm{H}}\overleftarrow{\Sigma}_{\mathbf{q}}^{-1}\left(\mathbf{h}-\overleftarrow{\mu}_{\mathbf{q}}\right)\right\}\end{aligned} \tag{5.46}$$

and the parameters of the distribution for $\mathsf{n}_{\mathbf{q}\to f_Y}(\mathbf{q})$ are found to be

$$\vec{\Sigma}_{\mathbf{q}} = \left(\vec{\Sigma}_{\mathbf{q}}^{-1}+\overleftarrow{\Sigma}_{\mathbf{q}}^{-1}\right)^{-1}$$

and

$$\vec{\mu}_{\mathbf{q}} = \vec{\Sigma}_{\mathbf{q}}\left(\vec{\Sigma}_{\mathbf{q}}^{-1}\vec{\mu}_{\mathbf{q}}+\overleftarrow{\Sigma}_{\mathbf{q}}^{-1}\overleftarrow{\mu}_{\mathbf{q}}\right).$$

5.9 Discrete BP for Equalization

Messages from auxiliary symbol variables to factor nodes

The factors f_Y receive posterior beliefs from the auxiliary variables as given by

$$\mathsf{n}_{\mathbf{s}\to f_Y}(\mathbf{s}) \propto \mathsf{m}_{f_Y\to\mathbf{s}}(\mathbf{s})\mathsf{m}_{f_S\to\mathbf{s}}(\mathbf{s}). \tag{5.47}$$

On the other hand, messages passed from the auxiliary variables to the detection factors f_S are in the form of extrinsic distributions as given by

$$\mathsf{n}_{\mathbf{s}\to f_S}(\mathbf{s}) \propto \mathsf{m}_{f_Y\to\mathbf{s}}(\mathbf{s}). \tag{5.48}$$

Messages from equalization node to auxiliary variables

The BP rule (sum-product algorithm) leads to the following message:

$$\begin{aligned}\mathsf{m}_{f_S\to\mathbf{s}}(\mathbf{s}) &= \sum_{x_1}\cdots\sum_{x_N} p(\mathbf{s}|x_1,\ldots,x_N)\prod_{i=1}^{N}\mathsf{n}_{x_i\to f_S}(x_i)\\ &= \prod_{i=1}^{N}\mathsf{n}_{x_i\to f_S}(s_i),\end{aligned} \tag{5.49}$$

where s_i is the ith element of $\mathbf{s}$ and $p(\mathbf{s}|x_1,\ldots,x_N)$ enforces equality between the symbol variables and the associated components of the auxiliary symbol variable as given by

$$p(\mathbf{s}|x_1,\ldots,x_N) = \prod_{i=1}^{N}\mathrm{I}(s_i = x_i). \tag{5.50}$$

5.10 MF Message Derivations

Here we derive the messages involving the time-domain channel estimation in the MF subgraph for the MIMO-OFDM model.

Observations to channel coefficients

For the derivation of the message $\mathsf{m}_{f_{Y_k}\to h_{mn}(l)}(h_{mn}(l))$, we first consider the factor function. The factor function for factor node f_{Y_k} is the likelihood function of observation $\mathbf{y}(k)$. Specifically we consider the likelihood function based on signal model (5.24) where the channel coefficients are

expressed in terms of the time-domain taps as given by (5.21). The likelihood function is given by

$$\begin{aligned}
&p(\mathbf{y}(k)|\mathbf{s}(k),\mathbf{h}) \\
&= \left(\frac{\gamma}{\pi}\right)^M \exp\left\{-\gamma \sum_{m=1}^{M} \left| y_m(k) - \sum_{n=1}^{N} s_n(k) \sum_{l=0}^{L-1} h_{mn}(l) d_{kl} \right|^2 \right\} \\
&= \left(\frac{\gamma}{\pi}\right)^M \exp\left\{-\gamma \sum_{m=1}^{M} \left(|y_m(k)|^2 \; - 2\Re\left[y_m(k) \sum_{n=1}^{N} s_n(k)^* \sum_{l=0}^{L-1} h_{mn}(l)^* d_{kl}^* \right] \right.\right. \\
&\qquad \left.\left. + \sum_{n_1=1}^{N} \sum_{n_2=1}^{N} s_{n_1}(k) s_{n_2}(k)^* \sum_{l_1=0}^{L-1} \sum_{l_2=0}^{L-1} h_{mn_1}(l_1) h_{mn_2}(l_2)^* d_{kl_1} d_{kl_2}^* \right) \right\}. \qquad (5.51)
\end{aligned}$$

The factor function is simplified by removing all terms which are constant with respect to $h_{mn}(l)$ as given by

$$\begin{aligned}
&p(\mathbf{y}(k)|\mathbf{s}(k),\mathbf{h}) \\
&\propto \exp\left\{-\gamma \left(- 2\Re\left[y_m(k) s_n(k)^* h_{mn}(l)^* d_{kl}^* \right] + |s_n(k)|^2 |h_{mn}(l)|^2 \right.\right. \\
&\qquad \left.\left. + 2\Re\left[\sum_{n' \neq n} s_{n'}(k) s_n(k)^* \sum_{l'=0}^{L-1} h_{mn'}(l') h_{mn}(l)^* d_{kl'} d_{kl}^* + |s_n(k)|^2 \sum_{l' \neq l} h_{mn}(l') h_{mn}(l)^* d_{kl'} d_{kl}^* \right] \right) \right\} \\
&\propto \exp\left\{-\gamma \left(|s_n(k)|^2 |h_{mn}(l)|^2 \; - 2\Re\left[h_{mn}(l)^* \left(y_m(k) s_n(k)^* d_{kl}^* \right.\right.\right.\right. \\
&\qquad \left.\left.\left.\left. - \sum_{n' \neq n} s_{n'}(k) s_n(k)^* d_{kl}^* \sum_{l'=0}^{L-1} h_{mn'}(l') d_{kl'} \; - |s_n(k)|^2 d_{kl}^* \sum_{l' \neq l} h_{mn}(l') d_{kl'} \right) \right] \right) \right\}. \qquad (5.52)
\end{aligned}$$

The message from observation factor node f_{Y_k} to channel coefficient $h_{mn}(l)$ is computed according

to the MF approximation. The derivation for the message is as follows:

$$
\begin{aligned}
&\mathsf{m}_{f_{Y_k}\to h_{mn}(l)}(h_{mn}(l)) \\
&\quad = \exp\left\{ \int\cdots\int \sum_{\mathbf{s}(k)} \mathsf{n}_{\mathbf{s}(k)\to f_{Y_k}}(\mathbf{s}(k)) \ln\left(p(\mathbf{y}(k)|\mathbf{s}(k),\mathbf{h})\right) \right. \\
&\qquad \left. \prod_{\substack{m',n',l' \\ \{m',n',l'\}\neq\{m,n,l\}}} \mathsf{n}_{h_{m'n'}(l')\to f_{Y_k}}(h_{m'n'}(l'))dh_{m'n'}(l') \right\} \\
&\quad \propto \exp\left\{ -\gamma \left(\overleftarrow{\rho}_{\mathbf{s}(k)n,n}|h_{mn}(l)|^2 - 2\Re\left[h_{mn}(l)^* \left(y_m(k)\overleftarrow{\mu}^*_{\mathbf{s}(k)n} d^*_{kl} \right.\right.\right.\right. \\
&\qquad \left.\left.\left.\left. - \sum_{n'\neq n} \overleftarrow{\rho}_{\mathbf{s}(k)n',n} d^*_{kl} \sum_{l'=0}^{L-1} \mu_{h_{mn'}(l')} d_{kl'} - \overleftarrow{\rho}_{\mathbf{s}(k)n,n} d^*_{kl} \sum_{l'\neq l} \mu_{h_{mn}(l')} d_{kl'} \right) \right] \right) \right\} \\
&\quad \propto \mathcal{CN}\left(h_{mn}(l);\, \phi_{mn}(l,k), \psi_{mn}(l,k)^{-1} \right),
\end{aligned} \tag{5.53}
$$

where the mean is given by

$$
\phi_{mn}(l,k) = \overleftarrow{\rho}^{-1}_{\mathbf{s}(k)n,n} \left(y_m(k)\overleftarrow{\mu}^*_{\mathbf{s}(k)n} d^*_{kl} - \sum_{n'\neq n} \overleftarrow{\rho}_{\mathbf{s}(k)n',n} d^*_{kl} \sum_{l'=0}^{L-1} \mu_{h_{mn'}(l')} d_{kl'} - \overleftarrow{\rho}_{\mathbf{s}(k)n,n} d^*_{kl} \sum_{l'\neq l} \mu_{h_{mn}(l')} d_{kl'} \right) \tag{5.54}
$$

and the precision (inverse variance) is given by

$$
\psi_{mn}(l,k) = \gamma \overleftarrow{\rho}_{\mathbf{s}(k)n,n}. \tag{5.55}
$$

In the above expression, the mean $\overleftarrow{\mu}_{\mathbf{s}(k)i}$ is the ith element of $\overleftarrow{\mu}_{\mathbf{s}(k)}$ and cross-correlation $\overleftarrow{\rho}_{\mathbf{s}(k)i,j}$ is the i,jth element of matrix $\overleftarrow{\mathbf{R}}_{\mathbf{s}(k)}$.

Channel coefficients to observations

Since the observation factors are contained within the MF portion of the graph, posterior beliefs are returned to them from the channel coefficient variables. The message (posterior) is given by

$$\begin{aligned}
&\mathsf{n}_{h_{mn}(l)\to f_{Y_k}}(h_{mn}(l)) \\
&= \prod_{k=0}^{K-1} \mathsf{m}_{f_{Y_k}\to h_{mn}(l)}(h_{mn}(l))\mathsf{m}_{f_{h_{mn}(l)}\to h_{mn}(l)}(h_{mn}(l)) \\
&\propto \exp\left\{-\left[\sum_{k=0}^{K-1}\psi_{mn}(l,k)|h_{mn}(l)-\phi_{mn}(l,k)|^2+\psi^p_{mn}(l)|h_{mn}(l)-\phi^p_{mn}(l)|^2\right]\right\} \\
&\propto \exp\left\{-|h_{mn}(l)|^2\left(\sum_{k=0}^{K-1}\psi_{mn}(l,k)+\psi^p_{mn}(l)\right)\right. \\
&\left.+2\Re\left[h_{mn}(l)\left(\sum_{k=0}^{K-1}\psi_{mn}(l,k)\phi^*_{mn}(l,k)+\psi^p_{mn}(l)\phi^{p*}_{mn}(l)\right)\right]\right\} \\
&\propto \mathcal{CN}\left(h_{mn}(l);\mu_{h_{mn}(l)},\sigma^2_{h_{mn}(l)}\right),
\end{aligned} \tag{5.56}$$

where the mean and variance of the message are given by

$$\mu_{h_{mn}(l)} = \frac{\sum_{k=0}^{K-1}\psi_{mn}(l,k)\phi_{mn}(l,k)+\psi^p_{mn}(l)\phi^p_{mn}(l)}{\sum_{k=0}^{K-1}\psi_{mn}(l,k)+\psi^p_{mn}(l)} \tag{5.57}$$

and

$$\sigma^2_{h_{mn}(l)} = \left(\sum_{k=0}^{K-1}\psi_{mn}(l,k)+\psi^p_{mn}(l)\right)^{-1}, \tag{5.58}$$

respectively. It is useful to denote the mean and variance of the frequency-domain channel coefficients per subcarrier. According to 5.21, the mean and variance of $\tilde{h}_{mn}(k)$ are given by

$$\mu_{\tilde{h}_{mn}(k)} = \sum_{l=0}^{L-1}\mu_{h_{mn}(l)}d_{kl} \tag{5.59}$$

and

$$\sigma^2_{\tilde{h}_{mn}(k)} = \sum_{l=0}^{L-1}\sigma^2_{h_{mn}(l)}, \tag{5.60}$$

respectively.

Observations to auxiliary variables

The likelihood function based on signal model (5.23) is expressed as given by

$$\begin{aligned} & p(\mathbf{y}(k)|\mathbf{s}(k), \tilde{\mathbf{H}}(k)) \\ & \quad = \exp\left\{-\gamma \left(\mathbf{y}(k) - \tilde{\mathbf{H}}(k)\mathbf{s}(k)\right)^{\mathrm{H}} \left(\mathbf{y}(k) - \tilde{\mathbf{H}}(k)\mathbf{s}(k)\right)\right\} \\ & \quad = \exp\left\{-\gamma \left(\mathbf{y}(k)^{\mathrm{H}}\mathbf{y}(k) - 2\Re\left[\mathbf{y}(k)^{\mathrm{H}}\tilde{\mathbf{H}}(k)\mathbf{s}(k)\right] + \mathbf{s}(k)^{\mathrm{H}}\tilde{\mathbf{H}}(k)^{\mathrm{H}}\tilde{\mathbf{H}}(k)\mathbf{s}(k)\right)\right\}. \end{aligned} \tag{5.61}$$

In the constructed factor graph, the MF approximation message passing rule leads to the following message:

$$\begin{aligned} & \mathsf{m}_{f_{Y_k} \to \mathbf{s}(k)}(\mathbf{s}(k)) \\ & \quad = \exp\left\{\int \cdots \int \prod_{m,n,l} \mathsf{n}_{h_{mn}(l) \to f_{Y_k}}(h_{mn}(l)) \ln p(\mathbf{y}(k)|\mathbf{s}(k), \tilde{\mathbf{H}}(k)) d\mathbf{h}\right\} \\ & \quad \propto \exp\left\{-\gamma \left(\mathbf{s}(k)^{\mathrm{H}}\mathbf{W}(k)\mathbf{s}(k) - 2\Re\left[\mathbf{y}(k)^{\mathrm{H}}\Xi(k)\mathbf{s}(k)\right]\right)\right\} \qquad (5.62) \\ & \quad \propto \mathcal{CN}\left(\mathbf{s}(k); \mathbf{W}(k)^{-1}\Xi(k)^{\mathrm{H}}\mathbf{y}(k), \gamma^{-1}\mathbf{W}(k)^{-1}\right), \qquad (5.63) \end{aligned}$$

where

$$\begin{aligned} \Xi(k) &= \int \cdots \int \prod_{m,n,l} \mathsf{n}_{h_{mn}(l) \to f_{Y_k}}(h_{mn}(l)) \tilde{\mathbf{H}}(k) d\mathbf{h} \\ &= \begin{bmatrix} \mu_{\tilde{h}_{11}(k)} & \mu_{\tilde{h}_{12}(k)} & \cdots & \mu_{\tilde{h}_{1N}(k)} \\ \mu_{\tilde{h}_{21}(k)} & \mu_{\tilde{h}_{22}(k)} & & \vdots \\ \vdots & & \ddots & \\ \mu_{\tilde{h}_{M1}(k)} & \cdots & & \mu_{\tilde{h}_{MN}(k)} \end{bmatrix} \end{aligned} \tag{5.64}$$

and

$$\begin{aligned} \mathbf{W}(k) &= \int \cdots \int \prod_{m,n,l} \mathsf{n}_{h_{mn}(l) \to f_{Y_k}}(h_{mn}(l)) \tilde{\mathbf{H}}(k)^{\mathrm{H}} \tilde{\mathbf{H}}(k) d\mathbf{h} \\ &= \Xi(k)^{\mathrm{H}}\Xi(k) + \sum_{m=1}^{M} \mathrm{diag}\left(\sigma^2_{\tilde{h}_{m1}(k)}, \sigma^2_{\tilde{h}_{m2}(k)}, \ldots, \sigma^2_{\tilde{h}_{mN}(k)}\right). \end{aligned} \tag{5.65}$$

In the case in which $\mathbf{s}(k)$ is a discrete variable, the message may be determined (up to a multiplicative constant) by evaluating (5.62) with respect to $\mathbf{s}(k)$ which avoids the matrix inversion of (5.63).

Chapter 6

Value of Iterative Parameter Estimation

6.1 Introduction

The capability of iterative probabilistic processing to solve difficult inference problems is clearly demonstrated by the success of turbo decoding of parallel concatenated convolutional codes (PCCC) and sum-product algorithm-based decoding of LDPC codes [6,65,66]. Further study of these algorithms has lead to connections being made to Bayesian belief propagation [71] and the development of factor graphs [7,72]. A factor graph is a bipartite graph with variable nodes and factor nodes representing the factorization of a function (in our case a probability distribution). Factor graphs and the sum-product algorithm provide a unifying representation for a number of receiver algorithms developed in the technical literature and have motivated the development of new algorithms, not only for decoding, but also for larger portions of the receiver. For example, demodulation, equalization, and multiuser detection have all been cast in a factor graph representation [7, 79].

Similarly, iterative receiver algorithms for synchronization and channel estimation have been explored in the literature [8, 61, 82, 85, 86, 165, 180]. Some of these algorithms are based on factor graph representations and the sum-product algorithm [8, 86, 180]. However, the sum-product algorithm is computationally complex for continuous parameters and a more practical approach to including parameter estimation in the iterative receiver has been through the expectation-maximization (EM) algorithm [31]. As developed in [82, 165], the EM algorithm is applied to the iterative receiver as follows. *E step:* the sum-product algorithm is used to compute marginal probability distributions *conditioned* on the current parameter estimates. *M step:* the parameters are re-estimated using the marginal probabilities of the data from the sum-product algorithm.

Many of the papers on iterative parameter estimation examine the mean squared error (MSE) performance (for example, [82, 85, 165]). Noels *et al.* showed that iterative parameter estimators achieved the Cramér-Rao lower bound (CRLB) for the MSE of frequency, phase, and symbol timing estimation of *coded* signals [51]. While the CRLB provides the best estimation performance that the iterative receiver can achieve (given the classical estimation framework), the desired in-

sight in our work is to understand the effect of parameter knowledge on data detection. Iterative parameter estimation has been shown to achieve a bit error rate (BER) and frame error rate (FER) very close to a receiver with perfect estimation. Thus, there are clearly cases in which the iterative approach outperforms traditional parameter estimation. However, in other cases traditional parameter estimation may be sufficient. The current body of work lacks a framework for determining when iterative parameter estimation is valuable, especially for making online decisions. In order to gain insight into the value of iterative parameter estimation, both performance and complexity must be viewed in the context of the performance and complexity of a receiver which relies on traditional parameter estimation (i.e., blind, training-based, or decision feedback methods, etc.).

EXIT charts provide insight into the convergence of probabilistic information exchanged between two blocks of the receiver (decoders, demodulators, equalizers, estimators, etc.) [52]. When the EXIT chart is constructed between a channel estimation block and the decoding block of the receiver, as done in [53, 54], it provides a simulation-based tool which is able to show the expected performance gains of re-estimating the channel in each iteration. In contrast to the simulation approach of EXIT charts, we develop a theoretical framework for evaluating the value of parameter knowledge which can provide feedback about the need to re-estimate parameters in real-time.

Our goal is to develop a framework for assessing the value of the iterative receiver as a whole and specifically iterative parameter estimation. Thus, we present an approach to explore the value of parameter knowledge for iterative detection of coded data. A very influential work by Yedidia *et al.* [55] made a connection between the sum-product algorithm (belief propagation) and the Bethe free energy. Specifically, it was discovered that the fixed points of the sum-product algorithm correspond to minima of the Bethe free energy [15, 56]. The Bethe free energy is a measure of divergence which we will use to evaluate the quality of our parameter knowledge with respect to data detection.

Consider the task of making inference about the transmitted data $\mathbf{x} = [x_1, \ldots, x_K]^{\mathrm{T}}$ given observation of the received signal $\mathbf{y} = [y_1, \ldots, y_L]^{\mathrm{T}}$. In the Bayesian framework, we are interested in the posterior distribution $p(\mathbf{x}|\mathbf{y})$. The factorization of the posterior distribution may be represented with a factor graph where inference can be efficiently performed using the sum-product algorithm. Let $\Theta = \{\theta_1, \ldots, \theta_N\}$ denote the collection of parameters of the model—for example, channel parameters. In this chapter, examples of carrier phase offset ϕ and channel gain α will be considered. In the Bayesian framework, the parameters are modeled as random variables leading to a joint posterior distribution $p(\mathbf{x}, \Theta|\mathbf{y})$.

6.2 Kullback-Leibler Divergence

In order to minimize the probability of error in detecting the transmitted bits, we are interested in computing marginal posterior probabilities. Thus, the receiver is fundamentally performing density estimation. The quality of the estimated (or trial) distribution can be measured by the Kullback-Leibler (KL) divergence [181]. The KL divergence is a standard metric for evaluating the quality

of using a trial distribution $q(\mathbf{x}, \Theta|\mathbf{y})$ to approximate the true posterior distribution $p(\mathbf{x}, \Theta|\mathbf{y})$ and is given by

$$\begin{aligned} &D(\, q(\mathbf{x}, \Theta|\mathbf{y}) \,||\, p(\mathbf{x}, \Theta|\mathbf{y}) \,) \\ &\quad = \int_{\Theta} \sum_{\mathbf{x} \in \mathcal{X}} q(\mathbf{x}, \Theta|\mathbf{y}) \ln \frac{q(\mathbf{x}, \Theta|\mathbf{y})}{p(\mathbf{x}, \Theta|\mathbf{y})} d\Theta \end{aligned} \tag{6.1}$$

where $\mathcal{X}$ is the domain of $\mathbf{x}$. The KL divergence between two distributions is zero only when the distributions are identical.

According to Bayes rule, the true distribution has the following form:

$$p(\mathbf{x}, \Theta|\mathbf{y}) = p(\mathbf{y}|\mathbf{x}, \Theta) p(\mathbf{x}) \prod_{i=1}^{N} p(\theta_i) / p(\mathbf{y}), \tag{6.2}$$

where we have assumed that the parameters are independent *a priori*. Consider the following form for the approximate distribution

$$q(\mathbf{x}, \Theta|\mathbf{y}) = q_{\mathbf{x}}(\mathbf{x}) \prod_{i=1}^{N} q_{\theta_i}(\theta_i), \tag{6.3}$$

where dependence on the observation $\mathbf{y}$ is implied. The KL divergence is given by

$$\begin{aligned} &D(\, q(\mathbf{x}, \Theta|\mathbf{y}) \,||\, p(\mathbf{x}, \Theta|\mathbf{y}) \,) = \\ &\int_{\Theta} \sum_{\mathbf{x}} q_{\mathbf{x}}(\mathbf{x}) \prod_{i=1}^{N} q_{\theta_i}(\theta_i) \ln \frac{q_{\mathbf{x}}(\mathbf{x}) \prod_{i=1}^{N} q_{\theta_i}(\theta_i)}{p(\mathbf{y}|\mathbf{x}, \Theta) p(\mathbf{x}) \prod_{i=1}^{N} p(\theta_i) / p(\mathbf{y})} d\Theta. \end{aligned} \tag{6.4}$$

For the purpose of analytical tractability, let the trial distributions for the parameters take the form of an impulse at an estimated value (e.g., $q_{\theta_i}(\theta_i) = \delta(\theta_i - \hat{\theta}_i)$). This approach is also motivated by the use of parameter estimates within a receiver—both traditional receivers and iterative receivers based on the EM algorithm and its variants [61, 82, 85, 143, 165]. Applying this assumption, the divergence is proportional to

$$\begin{aligned} &D(\, q(\mathbf{x}, \Theta|\mathbf{y}) \,||\, p(\mathbf{x}, \Theta|\mathbf{y}) \,) \propto \\ &\quad \sum_{\mathbf{x}} q_{\mathbf{x}}(\mathbf{x}) \ln q_{\mathbf{x}}(\mathbf{x}) - \sum_{\mathbf{x}} q_{\mathbf{x}}(\mathbf{x}) \ln p(\mathbf{y}|\mathbf{x}, \hat{\Theta}) \\ &\quad - \sum_{\mathbf{x}} q_{\mathbf{x}}(\mathbf{x}) \ln p(\mathbf{x}) - \sum_{i=1}^{N} \ln p(\hat{\theta}_i) \end{aligned} \tag{6.5}$$

where we assume that the entropy of the dirac delta is zero (i.e., $\int_{\theta_i} \delta(\theta_i - \hat{\theta}_i) \ln \delta(\theta_i - \hat{\theta}_i) d\theta_i = 0$).

6.3 Bethe Free Energy

Let the factorization of $p(\mathbf{y}|\mathbf{x},\hat{\Theta})p(\mathbf{x})$ be represented by $\prod_\alpha f_\alpha(\mathbf{x}_\alpha)$ where α is an index of the factors and the argument $\mathbf{x}_\alpha$ is a subset of $\mathbf{x}$. The trial distribution of the data $q_\mathbf{x}(\mathbf{x})$ may be restricted to the following form (which is exact for tree structured distributions) [15]:

$$q_\mathbf{x}(\mathbf{x}) = \frac{\prod_\alpha b_\alpha(\mathbf{x}_\alpha)}{\prod_k b_k(x_k)^{d_k-1}} \tag{6.6}$$

where $b_\alpha(\mathbf{x}_\alpha)$ and $b_k(x_k)$ are marginal beliefs for the respective variables and d_k is the number of factors which contain x_k (i.e., the degree of variable x_k in the graphical model of the distribution). The distribution given in (6.6) is fundamental to the Bethe approximation of the free energy [15, 56]. With substitution of (6.6) into (6.5) and with non-informative prior distributions (i.e, $p(\theta_i) = 1$ for all $i = 1,\ldots,N$), it can be verified that the divergence is equivalent to the Bethe free energy (F_{BETHE}) of $p(\mathbf{y}|\mathbf{x},\hat{\Theta})p(\mathbf{x})$.

A connection between the sum-product algorithm (belief propagation) and the Bethe free energy was made by Yedidia *et al.* in [55]. Specifically, it was discovered that the fixed points of the sum-product algorithm corresponded to minima of the Bethe free energy [15,56]. This connection has lead to further insight into belief propagation algorithms [106, 107, 110, 112, 113]. The Bethe free energy is a variational free energy based on the tree-like trial probability distribution given in (6.6).

Our goal in this work is to determine the value of accurate parameter information on the iterative receiver. To that end it is useful to know the impact a parameter or set of parameters has on the effectiveness of the receiver in estimating the marginal probabilities. Due to the error correction code, the true joint probability distribution and the true marginal probabilities are not tractable. Therefore, we use the Bethe free energy which provides a tool for determining how close the estimated distribution (based on the beliefs obtained from the sum-product algorithm) is to the true distribution without requiring knowledge of the true marginal probabilities. One difficulty, however, is that while an incorrect parameter estimate may still lead to correct marginal probabilities, it will increase the divergence value. In order to isolate the impact of the beliefs, we propose to normalize the likelihood function. This is accomplished by replacing $f_\alpha(\mathbf{x}_\alpha)$ with $\frac{f_\alpha(\mathbf{x}_\alpha)}{\sum_{\mathbf{x}_\alpha} f_\alpha(\mathbf{x}_\alpha)}$ for all factors of the likelihood function.

6.4 Carrier Phase Estimation Example

As an example, consider the problem of carrier phase estimation and coded data detection. The received signal is comprised of coded data, an unknown phase offset, and (complex) Gaussian noise. The signal model is given by

$$y_k = x_k e^{j\phi} + w_k \;\; \forall\, k = 1,\ldots,K \tag{6.7}$$

where y_k is the kth received sample, x_k is the kth coded bit, $\phi \in \{-\pi, \pi\}$ is the phase offset, and w_k is iid complex Gaussian noise. The posterior distribution may be factored as follows:

$$p(\mathbf{x}, \phi|\mathbf{y}) \propto \prod_{k=1}^{K} p(y_k|x_k, \phi) \prod_{\beta=1}^{P} p(\mathbf{x}_\beta) p(\phi). \tag{6.8}$$

where $p(\mathbf{x}) = \prod_{\beta=1}^{P} p(\mathbf{x}_\beta)$ represents the factored code constraints. For example, with an LDPC code, $p(\mathbf{x}_\beta)$ represents the βth parity check and $\mathbf{x}_\beta$ are the bits included in the parity check. A non-informative prior (uniform distribution) is assumed for $p(\phi)$. The Bethe approximation is made for $q_{\mathbf{x}}(\mathbf{x})$ in the trial distribution. The Bethe free energy is minimized with respect to the beliefs when the beliefs are fixed points of the sum-product algorithm. Thus, given a particular estimate $\hat{\phi}$, the Bethe free energy is evaluated with the beliefs obtained from the sum-product algorithm. For the given example of phase offset, the Bethe free energy may be written as given by

$$\begin{aligned} F_{BETHE} \propto & \sum_{\beta=1}^{P} \sum_{\mathbf{x}_\beta} b_\beta(\mathbf{x}_\beta) \ln b_\beta(\mathbf{x}_\beta) \\ & - \sum_{k=1}^{K} (d_k - 2) \sum_{x_k} b_k(x_k) \ln b_k(x_k) \\ & - \sum_{k=1}^{K} \sum_{x_k} b_k(x_k) \ln p(y_k|x_k, \hat{\phi}) \\ & - \sum_{\beta=1}^{P} \sum_{\mathbf{x}_\beta} b_\beta(\mathbf{x}_\beta) \ln p(\mathbf{x}_\beta) \end{aligned} \tag{6.9}$$

As discussed in the previous section, we propose to normalize the likelihood terms by replacing $f(y_k|x_k, \hat{\phi})$ with $\frac{f(y_k|x_k,\hat{\phi})}{\sum_{x_k} f(y_k|x_k,\hat{\phi})}$. The normalization term is not a function of x_k so the relative value of F_{BETHE} as a function of x_k is unchanged. However, the normalization is a function of $\hat{\phi}$ and thus, comparison of F_{BETHE} at two different values of $\hat{\phi}$ has changed. The impact of this normalization, as is shown in the results, is that the Bethe free energy is strongly influenced by the beliefs.

6.5 Results

In this section, results are presented in order to demonstrate the usefulness of the KL Divergence and (normalized) Bethe free energy. Specifically these tools

- evaluate the impact of the parameter on the task of estimating the marginal posterior probabilities,
- evaluate the capability of the iterative receiver to estimate the unknown parameter, and

- indicate whether or not to devote resources to iterative parameter estimation in an 'online' scenario.

A 1/2-rate LDPC code which excludes the 'all ones' codeword is designed (i.e., there are parity checks with odd degree). This ensures that flipping the bits (e.g., as a result of a π phase rotation for modulations such as BPSK and QPSK) will not result in a valid codeword. SNR is measured as the ratio of signal power to the variance of the complex valued noise.

In Figs. 6.1–6.4, four results are given for the example problem where the modulation is BPSK, the code is a 1/2-rate LDPC code, SNR = 5 dB, and $\phi_{true} = 0$. The KL divergence and the Bethe free energy with normalized likelihood terms are shown in Fig. 6.1 and Fig. 6.2, respectively. In order to understand these metrics, two additional figures have been included. First, in Fig. 6.3 we show the average KL divergence between the marginal posterior beliefs obtained with the true phase offset[1] ϕ_{true} and the marginal posterior beliefs obtained at the estimated phase offset $\hat{\phi}$ which is expressed as given by

$$\frac{1}{K}\sum_{k=1}^{K} D\left(b_k^{\phi_{true}}(x_k) \,||\, b_k^{\hat{\phi}}(x_k) \right). \tag{6.10}$$

The beliefs are of interest because they are the basis for estimating the transmitted information bits. The average KL divergence in (6.10) provides a measure of error in the estimation of the beliefs as a result of parameter error. Second, Fig. 6.4 shows the fraction of the information bits which are in error. This is displayed in order to determine how well the other metrics correlate with the receiver's goal of minimizing the number of errors made in detecting the information bits.

By comparing Fig. 6.1 with Fig. 6.3, we observe that error in the parameter estimate more strongly influences the KL divergence than it does the marginal beliefs. Fig. 6.3 demonstrates that the beliefs are unchanged over a range of about $(-1.1, 1.1)$ radians. However, this fact is not apparent in the KL divergence plot. On the other hand, the normalized Bethe free energy does highlight this behavior. A *bowl* is observed in the normalized Bethe free energy that corresponds well with the marginal posterior beliefs divergence and bit error performance.

The KL divergence does indicate the direction in which the receiver will update the phase offset parameter. As expected, the direction of update is correct for a starting estimate of about $-\pi/2 \leq \hat{\phi} \leq \pi/2$, that is, for a starting estimate within the phase ambiguity of the modulation. The shaded region in Figs. 6.1–6.4 highlights the range over which the receiver through iterative parameter estimation can recover the data while the traditional receiver has failed to completely recover the data (see Fig. 6.4). Recovery of the data for a phase estimate outside this range requires the phase ambiguity to be resolved.

In Figs. 6.5 and 6.6, the normalized Bethe free energy is shown for the problem of phase estimation and amplitude estimation, respectively. We observe that the amplitude parameter has a very similar

[1]The sum-product algorithm is approximate for graphs with cycles. Thus, we use the term beliefs to differentiate between the true marginals (which are unknown) and the result of the sum-product algorithm at the true phase offset (the 'true' beliefs).

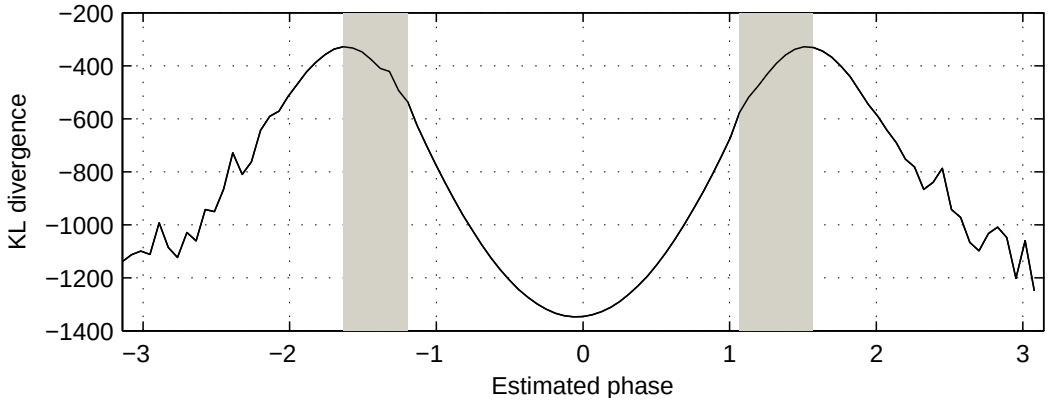

Figure 6.1: KL divergence from (6.9) vs. estimated phase parameter. Shaded region: data is incorrectly detected with the current phase estimate *and* is correctly detected with iterative phase estimation.

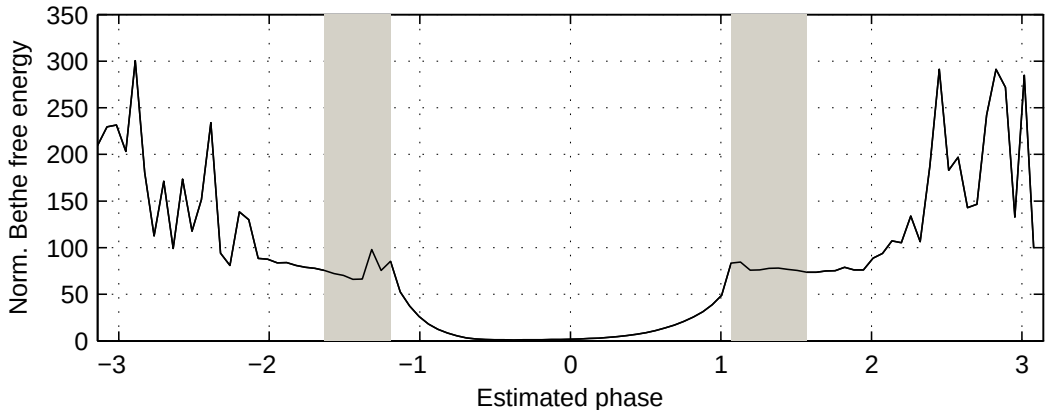

Figure 6.2: Bethe free energy with normalized likelihoods. Shaded region: data is incorrectly detected with the current phase estimate *and* is correctly detected with iterative phase estimation.

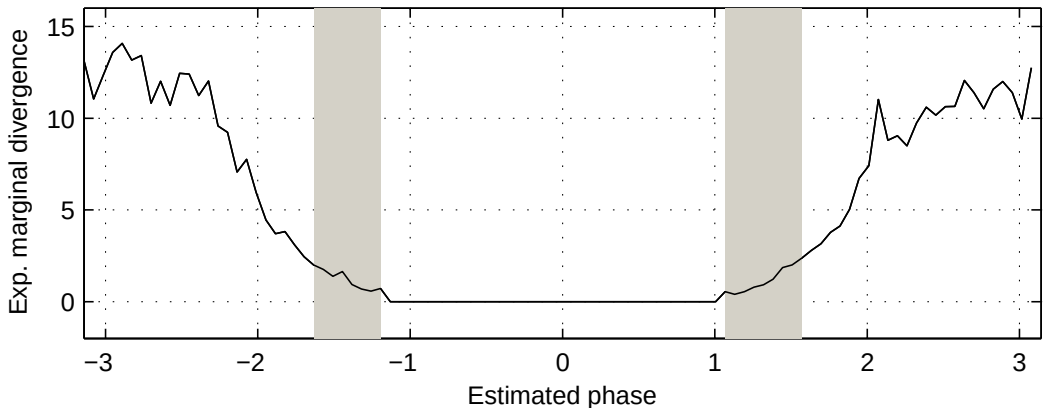

Figure 6.3: Expected divergence between marginal posterior beliefs from (6.10). Shaded region: data is incorrectly detected with the current phase estimate *and* is correctly detected with iterative phase estimation.

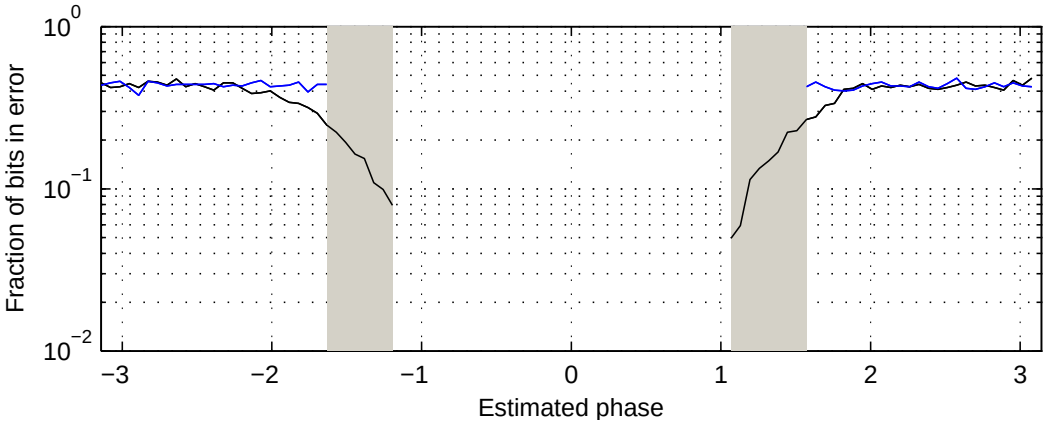

Figure 6.4: Fraction of bits in error with fixed phase estimate (black) and with iterative EM algorithm phase estimation (blue). The EM algorithm is able to correct the errors made in the shaded region. Shaded region: data is incorrectly detected with the current phase estimate *and* is correctly detected with iterative phase estimation.

behavior to the phase parameter. At higher SNR, the bowl in the normalized Bethe free energy is wider—signifying a greater tolerance to parameter error. These figures show the impact of the parameter estimate on the performance of the receiver. Specifically, we can assess the importance of accurate parameter information. This result provides a starting point for determining the value of the iterative receiver. When parameter estimation using a traditional receiver does not put us within the bowl of the normalized Bethe free energy, we expect errors to be made in data detection and that there is value to performing iterative parameter estimation.

In Fig. 6.6, the curve for an SNR of 5 dB (i.e., $E_b/N_0 = 2$ dB for 16QAM and a 1/2-rate code) does not have a bowl region. This is because the SNR is below the required value for the code to achieve reliable detection. The curve indicates that there is little value in improving the amplitude estimate which is intuitive because, at this SNR, the frame error rate will be high given perfect parameter knowledge.

At high SNR, the normalized Bethe free energy is zero when the phase parameter is near the true value. This would indicate that the trial distribution is equal to the true distribution. In these cases, the likelihoods are strongly biased toward one of the symbols and the normalized likelihood values tend towards either 0 or 1. In addition, the beliefs also tend towards 0 or 1. Finally, the code and modulation constraints are enforced by the probability distributions $p(\mathbf{x}_\beta)$ for all β. These constraints are deterministic: forcing the value of $p(\mathbf{x}_\beta)$ to be either 0 or 1. Thus, at high SNR, the terms $b_k(x_k)$, $b_\beta(\mathbf{x}_\beta)$, $p(y_k|x_k, \hat{\phi})$, and $p(\mathbf{x}_\beta)$ of (6.9) take on values of 0 or 1. When the code and modulation constraints are satisfied and the beliefs are in agreement with the likelihood information, the normalized Bethe free energy will be comprised of terms $1 \cdot \ln 1 = 0$ and $0 \cdot \ln 0 = 0$. A value of zero (or near zero) indicates that the beliefs have polarized and that a valid codeword has been found.

The absolute value of the normalized Bethe free energy is also significant. Regardless of SNR, the normalized Bethe free energy outside of the bowl is consistently higher than it is within the bowl. In Figs. 6.5 and 6.6, the dashed horizontal line displays the normalized Bethe free energy when the beliefs have uniform distributions (maximum entropy). When the parameter estimate is outside of the bowl region or when the SNR is insufficient, the Bethe free energy is generally at or above this boundary. This motivates an online approach to determining whether or not to perform iterative parameter estimation by comparing the normalized Bethe free energy with a threshold below the uniform belief boundary.

6.6 Conclusion

In this chapter, we have introduced a approach to evaluate the value of iterative parameter estimation for communications receivers. The approach is based on divergence and free energy functions associated with belief propagation. The normalized Bethe free energy demonstrates the impact of the estimated parameter on the performance of the receiver. In the ongoing work, these results will be interpreted within the context of (a) the estimation performance of the traditional receiver

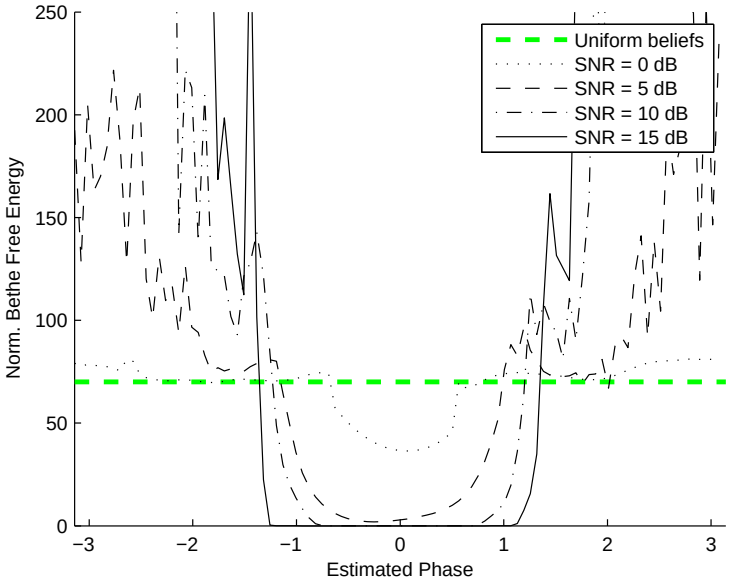

Figure 6.5: Normalized Bethe free energy as a function of the estimated phase value. The modulation is BPSK and the true phase is $\phi_{true} = 0$.

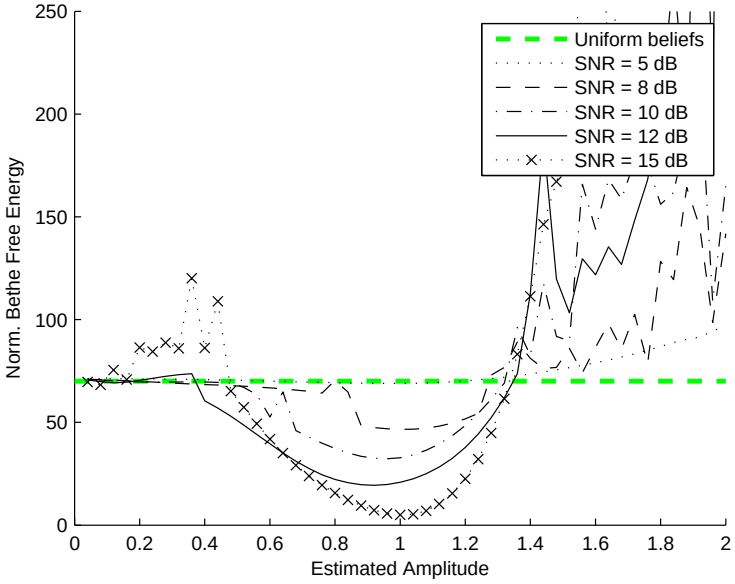

Figure 6.6: Normalized Bethe free energy as a function of the estimated amplitude value. The modulation is 16 QAM and the true amplitude is $\alpha_{true} = 1$.

and (b) the complexity associated with improving the estimate within an iterative receiver. The normalized Bethe free energy can be computed online, while metrics such as the average marginal divergence cannot. Therefore, the normalized Bethe free energy is promising as a means of making online decisions about the value of further iterative processing and determining whether or not additional energy should be expended.

Chapter 7

Conclusion

In this dissertation we identified three key problems which can benefit from iterative probabilistic processing, but for which the SPA is too complex. They are (1) joint synchronization and detection in multipath channels with an emphasis on frame timing, (2) detection in the presence of co-channel interference and non-Gaussian noise, and (3) joint channel estimation and multi-signal detection. Advances in the area of iterative probabilistic processing were presented in order to tackle these problems. The motivation behind the work was to (a) compromise as little as possible on the performance that is achieved while limiting the computational complexity and (b) maintain good theoretical justification to the algorithms that are developed.

In Chapter 3, we developed an iterative receiver structure for frame synchronization, channel estimation, equalization, demodulation, and decoding. We addressed complexity by developing code-aided frame synchronization approaches based on the EM and SPP metrics as well as by developing a frame pre-processing stage. The receiver was shown to perform nearly as well as a receiver with perfect knowledge of the synchronization parameters. Furthermore, the proposed receiver achieved significant gains over traditional frame synchronization while maintaining a reasonable complexity.

In Chapter 4, we identified a detection problem that is relevant to future wireless systems, but also poses a significant challenge to reliable detection. Specifically the combination of co-channel interference, inter-symbol interference, and non-Gaussian noise. By approximating the signal components with weaker received power with Gaussian distributions and maintaining discrete domains for the signal components with the strongest received power, we found that the sum-product messages could be effectively approximated. The proposed algorithm improved upon the performance of the state-of-the-art with lower complexity. By making use of the EM algorithm for estimation, the receiver was able to achieve a FER very close to that of a receiver with prior knowledge of the channel and noise parameters.

In Chapter 5, we developed a factor graph construction for applying the BP-MF framework to multi-signal detection and channel estimation. The exact implementation of our proposed BP-MF

construction applies discrete BP to the multi-signal detection part of the receiver which is exponentially complex in the number of received signals. Therefore, we also developed a low-complexity variant which applies Gaussian BP to the equalization (i.e., multi-signal detection) function. This implementation maintains solid theoretical basis by using the EP algorithm to compute Gaussian approximations to the discrete messages passed from the discrete BP subgraph to the Gaussian BP subgraph. Both the exact implementation and the low-complexity alternative are shown to achieve excellent performance in a MIMO-OFDM system. This contribution, in particular, opens the door to future work in applying the developed probabilistic receiver architecture to other multi-signal detection problems. We develop and demonstrate a MF-based time-domain channel estimator for MIMO-OFDM. The architecture also enables EM, BP, or GAMP-based estimators to be incorporated. There is much room to explore the choice of estimation approach for a variety of estimation problems within the proposed architecture.

Finally, in Chapter 6, we present an approach to studying the value of iterative parameter estimation and to making "online" decisions of whether or not to continue to perform iterative parameter estimation. The metric we used for this study is the Bethe free energy with a normalization of the likelihood function. It was shown that this metric provided useful information about our ability to determine the data given the current parameter estimates. Further work is required to interpret the results within the context of (a) the estimation performance of the traditional receiver and (b) the complexity associated with improving the estimate within an iterative receiver. The normalized Bethe free energy can be computed online, while metrics such as the average marginal divergence cannot. Therefore, the normalized Bethe free energy is promising as a means of making online decisions about the value of further iterative processing and determining whether or not additional energy should be expended.

In addition to the specific lessons learned from each contribution, the following insights have been gained from the work as a whole:

- *Performance-complexity trade-offs may be exploited in the design of receiver algorithms*: Even through BP is in some cases prohibitively complexity, excellent performance can be achieved through probabilistic processing with manageable complexity. Examples from our work include the probability of exclusion P_{ex} for the HPD region and the complexity parameter C in the AMAP algorithm. Additionally, the number of iterations applied to iterative processing impacts complexity and performance which the work of Chapter 6 seeks to optimize.

- *Iterative probabilistic processing is particularly valuable for detection in co-channel interference*: Traditionally communication systems rely on orthogonality between multiple users in order to aid signal separation. However, these techniques reduce spectral efficiency. As improved spectral efficiency continues to be demanded from future wireless systems, it has become apparent that multiple access protocols which allow for interference are required. Iterative algorithms provide a means of signal separation which is unavailable with traditional sequential processing. In this dissertation, receiver algorithms have been developed which

have improved performance, lowered complexity, or both with respect to the existing body of iterative receiver work.

- *Iterative receivers can leverage advancements in machine learning*: On its face, the sum product algorithm provides a unified, formulaic solution to the design of an iterative receiver for any communications problem. However, digging deeper it is apparent that the SPA suffers from complexity in many practical problems (e.g., involving continuous variables and large discrete domains). On the other hand lower complexity alternatives from machine learning often suffer significantly in performance. In this dissertation we have shown that excellent performance may be maintained with manageable complexity while utilizing theoretically justified algorithms. We have consistently relied on algorithms which are broadly applicable to inference problems including the EM algorithm, the MF approximation, Gaussian BP, EP, and carefully chosen Gaussian approximations. As a result, the developed receiver structures are capable of leveraging advances in probabilistic processing. As both software and hardware developments take place, receivers will stand to benefit.

Appendix A

Iterative Synchronization in AWGN Channels

A.1 Introduction

Following the success of iterative (turbo) decoding of channel codes, iterative algorithms have been developed for the purpose of *code-aided* or *turbo* synchronization or, more generally, incorporating synchronization into an *iterative receiver*. These iterative algorithms seek to approximate maximum likelihood (ML) or maximum *a posteriori* (MAP) joint detection of the information bits and estimation of the synchronization parameters. In general, the receiver may be responsible for estimation of the channel gain, carrier phase, symbol timing, frame timing, noise power, and more. Traditionally, in packet based systems, a known preamble enables reliable synchronization through data-aided methods. In contrast, the structure inherent in the channel code can be exploited within the context of the iterative receiver to perform reliable synchronization without a preamble.

The work of Noels *et al.* in [80] and [82] provides a theoretical framework for an iterative receiver for synchronization based on the expectation-maximization (EM) algorithm. Frequency, symbol timing, phase, and amplitude estimation are considered in the theoretical development of [80] and [82]. However, simulations are limited to phase estimation in [80] and joint frequency and phase estimation in [82]. The application of the EM algorithm to symbol timing synchronization is considered in [81] and [85].

The channel time delay may, in general, be greater than one symbol period and the rotational symmetry which exists in common phase-amplitude constellations will lead to an ambiguity in the phase offset. In order to handle this, the time offset is divided into a frame offset (integer multiples of the symbol period) and symbol timing offset (within the symbol period) and the phase offset is divided into a phase ambiguity (integer multiples of the rotational ambiguity in the constellation) and a continuous phase offset (within the ambiguity). Determining the frame offset and phase ambiguity becomes a detection problem which does not immediately fit into the EM algorithm's

framework. Frame timing and phase ambiguity are approached from a hypothesis testing perspective in [13] where one of the presented methods is analogous to the EM algorithm and is given a more rigorous mathematical explanation in [128].

Unlike estimation of the other parameters, non-data-aided detection of the frame timing and phase ambiguity do not have the advantage of improving with the sequence length. In fact, without any knowledge of the data[1], detection of the phase ambiguity is not possible and detection of the frame timing (for example, using energy detection) is unreliable even at moderate signal-to-noise ratio (SNR). Furthermore, an incorrect estimate of the frame offset or phase ambiguity results in complete uncertainty about the information bits. Taking a hypothesis testing approach, the number of phase ambiguities that must be considered is well defined since this is determined by the modulation scheme. However, the number of frame offsets which must be considered is not obvious. In previous work on code-aided frame timing estimation, an assumption has been made regarding the uncertainty of the frame offset. For example, in [13] it is assumed that the number of possible frame offsets is fixed and simulations are performed assuming three possible offsets. In [127], it is assumed that the frame offset is known to be within a fixed window.

The results of [13] and [127] demonstrate that reliable frame timing estimation can be performed in an iterative receiver. However, due to the hypothesis testing approach, this comes at the cost of individually decoding each considered frame offset. Therefore, the cost of performing code-aided frame timing estimation is linearly dependent on the number of frame offsets which must be processed by the iterative receiver. The first contribution of this work is to characterize the frame offset distribution in order to evaluate the complexity of code-aided frame timing estimation. In a practical system, the receiver does not have prior knowledge of the values that the frame offset may take. Therefore, a method is provided for determining, on a frame-by-frame basis, *how many* and *which* frame offsets to process in the iterative receiver.

The second contribution of this work is to consider the design of an iterative receiver to perform joint estimation of the channel gain, carrier phase, symbol timing, frame timing, and noise power and detection of the information bits. The receiver performs the following functions: (a) non-data-aided coarse estimation of the continuous parameters (channel gain, phase offset, symbol timing, and noise power), (b) frame offset characterization to select the frame offsets processed by the iterative receiver, (c) hypothesis testing of the discrete parameters (phase ambiguity and frame offset), (d) EM-based fine estimation of the continuous parameters, and (e) posterior symbol and bit probability computation using the sum-product algorithm (SPA). In the numerical results, attention is given to the complexity of the iterative receiver.

[1] Knowledge of the data can come from pilot symbols or through feedback from the decoder when rotations by the phase ambiguity do not lead to valid codewords.

A.2 System Model

We consider a system in which the information bits are encoded with an error-correction code and modulated with a digital phase-amplitude modulation to produce a symbol sequence $\mathbf{x} = [x_0, x_1, \ldots, x_{K-1}]$. Packets containing a single codeword (K symbols) are transmitted in bursts. Thus, synchronization must be performed on each packet. The channel introduces unknown attenuation α, carrier phase shift ϕ, delay τ, and additive white Gaussian noise. In the context of the multipath channel of Chapter 3, $h_0 = \alpha e^{j\phi}$. The received signal is expressed as

$$r(t) = \alpha e^{j\phi} \sum_{k=0}^{K-1} x_k p(t - kT - \tau) + w(t), \tag{A.1}$$

where $p(t)$ is a unit-energy pulse shape for which $g(t) = p(t) * p(-t)$ satisfies the Nyquist condition for zero intersymbol interference, T is the symbol period, and $w(t)$ is a complex Gaussian random process.

As explained in Section A.1, the phase offset is separated into an integer component ψ corresponding to a multiple of the phase ambiguity (that is, the phase by which a constellation can be rotated and maintain an identical form) and a continuous component χ such that $\phi = (\frac{2\pi}{\Psi}\psi + \chi)$. For phase shift keying (PSK) constellations, Ψ is equal to the modulation order and for quadrature amplitude modulation (QAM) constellations $\Psi = 4$. Furthermore, let the delay be separated into an integer multiple of the symbol period η and a fractional symbol period ϵ such that $\tau = (\eta + \epsilon)T$ where $-1/2 \leq \epsilon < 1/2$. Making these substitutions, the output of the matched filter is given by

$$\begin{aligned} y(t) &= r(t) * p(-t) \\ &= \alpha e^{j(\frac{2\pi}{\Psi}\psi + \chi)} \sum_{k=0}^{K-1} x_k g(t - (k + \eta + \epsilon)T) + n(t) \end{aligned} \tag{A.2}$$

where $n(t) = \int_{-\infty}^{\infty} w(a)p(a - t)da$ is filtered noise.

A.3 Frame Timing

The iterative receiver structure described in Section A.4 will take advantage of feedback from the decoder in order to perform reliable frame offset estimation. In this section, we consider the problem of choosing which frame offsets to process in the iterative receiver. We apply Bayesian techniques to obtain a posterior distribution on η which is used to decide on the number of frame offsets to consider. We then supply the iterative receiver with the highest probability offsets.

We introduce characterization of the frame offset distribution *assuming knowledge of the other parameters*. Furthermore, for the purpose of frame characterization, we assume that the symbols are independent and identically distributed (iid). Let the number of possible frame offsets be H

such that $\eta \in (0, 1, \ldots, H-1)$. After matched filtering, samples taken at the (known) symbol offset are given by

$$\begin{aligned} y_l &= y((l+\epsilon)T) \\ &= \alpha e^{j(\frac{2\pi}{\Psi}\psi+\chi)} \sum_{k=0}^{K-1} x_k g((l-k-\eta)T) + n_l \\ &= \begin{cases} \alpha e^{j(\frac{2\pi}{\Psi}\psi+\chi)} x_{l-\eta} + n_l & l = \eta, \ldots, \eta+K-1 \\ n_l & \text{otherwise} \end{cases}, \end{aligned} \tag{A.3}$$

where $n_l = n((l+\epsilon)T)$ are iid complex Gaussian random variables with variance σ^2. The sequence of samples is denoted $\mathbf{y} = [y_0, \ldots, y_l, \ldots, y_{L-1}]$ where $L = K + H - 1$. We assume that the number of offsets included in the sampling process H is large to ensure that the true offset has been included and to allow proper characterization of the frame offset's posterior distribution. For this reason, it is impractical (and unnecessary) to decode all H offsets.

The posterior distribution of the frame offset is expressed as

$$p(\eta|\mathbf{y}) \propto f(\mathbf{y}|\eta)p(\eta), \tag{A.4}$$

where, for the time being, all other parameters are assumed to be known and the symbols are assumed equally likely. The prior distribution for η is selected to be uniform over the domain of η. That is, $p(\eta) = \frac{1}{H}$ which, being a constant, may be dropped. Removing proportional constants and integrating out the phase ambiguity (due to the assumption of equally likely symbols), the posterior distribution of η is given by

$$\begin{aligned} &p(\eta|\mathbf{y}) \propto \\ &\prod_{k=0}^{K-1} \sum_{x_k \in \mathcal{X}} \exp\left\{ \frac{1}{\sigma^2} \left(2\alpha \Re\left\{ e^{-j\chi} y_{k+\eta} x_k^* \right\} - \alpha^2 |x_k|^2 \right) \right\}, \end{aligned} \tag{A.5}$$

where $\mathcal{X}$ is the set of all constellation points. To determine the probability of a particular offset, the posterior distribution is normalized so that it sums to one.

Given a subset of the H frame offsets, the probability that the true offset has been excluded from this subset is denoted P_{ex}. We desire to find the smallest subset of frame offsets which has a P_{ex} less than or equal to a specified value. This set is known as the highest posterior density (HPD) region which we will denote by $\mathcal{S}$ with size $|\mathcal{S}|$. For example, to limit the probability of exclusion to $P_{ex} = 0.01$, the set of frame offsets $\mathcal{S}$ which must be processed by the iterative receiver is given by the 99% HPD region. An example posterior distribution is shown in Fig. A.1 where the 99% HPD is shown with marker "x". In this example, the HPD region contains seven offsets: $\{15, 22, 23, 24, 25, 26, 27\}$. The number of offsets is dynamically chosen when determining the HPD and, as seen in Fig. A.1, need not be contiguous.

Since, α, χ, ϵ, and σ^2 are not known, we use non-data-aided estimates to characterize the frame offset distribution in the receiver. This is analogous to the Quasi Hybrid Likelihood Approach

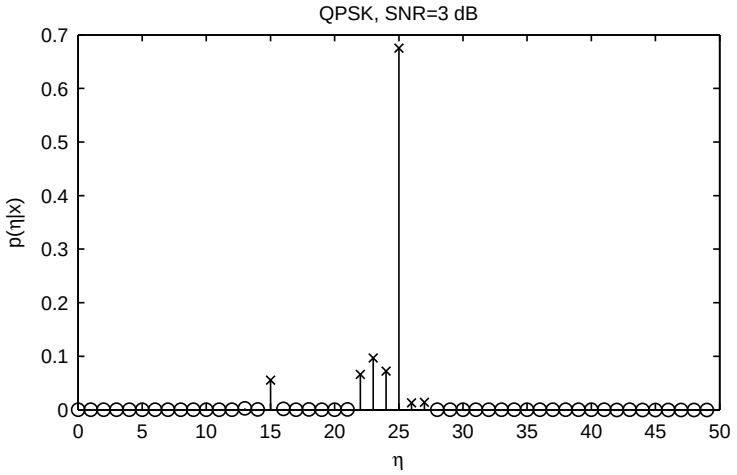

Figure A.1: Example posterior distribution and 99% highest posterior density region (shown with 'x') with true offset $\eta = 25$.

taken for detection and classification problems [141]. When simulating the iterative receiver's performance in Section A.5.2, we use this approach.

A.4 Iterative Receiver Structure

The iterative receiver is based on the expectation maximization (EM) algorithm [31] which provides a means of estimating parameter $\boldsymbol{\theta}$ from incomplete data $\mathbf{r}$ when there exists a set of unobserved or missing data $\mathbf{x}$. Maximization is performed on the complete data $\mathbf{z} = [\mathbf{r}, \mathbf{x}]$ which typically simplifies computation. The benefit of the EM algorithm, in our application, is that it enables the receiver to work with marginal symbol probabilities rather than symbol sequence probabilities as would be required in a maximum likelihood receiver. The EM algorithm is given by the following expectation (E) and maximization (M) steps:

$$\text{E step: } Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)}) = \int_{\mathbf{z}} f(\mathbf{z}|\mathbf{r}, \hat{\boldsymbol{\theta}}^{(p-1)}) \ln f(\mathbf{z}|\boldsymbol{\theta}) d\mathbf{z} \tag{A.6}$$

$$\text{M step: } \hat{\boldsymbol{\theta}}^{(p)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)}) \tag{A.7}$$

Since we are concerned with maximization, the E step can be simplified to [80]

$$Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)}) = \sum_{\mathbf{x}} f(\mathbf{x}|\mathbf{r}, \hat{\boldsymbol{\theta}}^{(p-1)}) \ln f(\mathbf{r}|\mathbf{x}, \boldsymbol{\theta}). \tag{A.8}$$

In our case, let $\mathbf{r}$ represent the projection of $r(t)$ onto a complete orthonormal basis; let $\mathbf{y}$ be the transmitted symbol sequence; and define $\boldsymbol{\theta} = [\alpha, \chi, \epsilon, \sigma^2]$. In the expectation step, posterior probabilities of the symbols are obtained for the previous estimate $\boldsymbol{\theta}^{(p-1)}$ using the sum-product algorithm SPA operating on a factor graph of the demodulator and decoder. Since maximization over $\boldsymbol{\theta}$

is still complex, we make use of the expectation conditional maximization (ECM) algorithm [118]. The ECM algorithm is a variant of EM which involves sequentially performing maximization over each parameter while conditioning on the remaining parameters. Because parameters ψ and η are discrete, a hypothesis testing approach is taken as considered in [128]. The ECM algorithm is applied for each set $\{\tilde{\psi}, \tilde{\eta}\}$ where $\tilde{\psi} \in \{0, 1, \ldots, \Psi - 1\}$ and $\tilde{\eta} \in \mathcal{S}$.

The three components of the receiver: expectation, conditional maximization, and hypothesis testing are considered in the three sections which follow. To simplify the notation we define $\mathbf{y}_\eta(\epsilon) = [y_\eta(\epsilon), \ldots, y_{\eta+K-1}(\epsilon)]$ where $y_l(\epsilon) = y((l+\epsilon)T)$.

A.4.1 Expectation

The expectation step requires computing posterior probabilities $p(x_k|\mathbf{r}, \hat{\boldsymbol{\theta}}^{(p-1)})$. From these probabilities, the expectations required for maximization in Section A.4.2 are

$$\bar{x}_k^{(p)} = \sum_{x_k \in \mathcal{X}} x_k p(x_k|\mathbf{r}, \hat{\boldsymbol{\theta}}^{(p-1)}), \text{ and} \tag{A.9}$$

$$\bar{x^2}_k^{(p)} = \sum_{x_k \in \mathcal{X}} |x_k|^2 p(x_k|\mathbf{r}, \hat{\boldsymbol{\theta}}^{(p-1)}). \tag{A.10}$$

The SPA and factor graphs provide a means of computing the posterior symbol probabilities. This step also provides posterior probabilities for the information bits which are necessary for detection. The factor graph receives as input symbol likelihoods which are calculated with respect to samples taken at the symbol timing estimate, i.e., $p(y_{k+\tilde{\eta}}(\hat{\epsilon}^{(p-1)})|x_k, \hat{\boldsymbol{\theta}}^{(p-1)})$. A thorough discussion of factor graph representations of modulation and coding is presented in [79].

A.4.2 Conditional maximization

We begin by developing the conditional maximization of α, χ, and ϵ. The log-likelihood function of these parameters is known to be equivalent to the following [143]:

$$\begin{aligned}
&\ln f(\mathbf{r}|\mathbf{x}, \alpha, \chi, \epsilon) \propto \\
&- \int_{-\infty}^{\infty} \left| r(t) - \alpha e^{j(\frac{2\pi}{\Psi}\tilde{\psi} + \chi)} \sum_{k=0}^{K-1} x_k p(t - (k + \tilde{\eta} + \epsilon)T) \right|^2 dt \\
&\propto 2\alpha \Re \left\{ e^{-j\left(\frac{2\pi}{\Psi}\tilde{\psi} + \chi\right)} \sum_{k=0}^{K-1} x_k^* y_{k+\tilde{\eta}}(\epsilon) \right\} - \alpha^2 \sum_{k=0}^{K-1} |x_k|^2
\end{aligned} \tag{A.11}$$

Substituting (A.11) into the E step expression from (A.8) produces

$$Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)}) = \\ 2\alpha \Re \left\{ e^{-j\left(\frac{2\pi}{\Psi}\tilde{\psi}+\chi\right)} \sum_{k=0}^{K-1} \left(\bar{x}_k^{(p)}\right)^* y_{k+\tilde{\eta}}(\epsilon) \right\} - \alpha^2 \sum_{k=0}^{K-1} \bar{x^2}_k^{(p)}. \tag{A.12}$$

The maximization step for ϵ is computed using a Newton-Raphson method as proposed by [81]. The pth update of ϵ is obtained from

$$\hat{\epsilon}^{(p)} = \hat{\epsilon}^{(p-1)} - \left. \frac{\left(\frac{\partial Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)})}{\partial \epsilon}\right)}{\left(\frac{\partial^2 Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)})}{\partial \epsilon^2}\right)} \right|_{\epsilon=\hat{\epsilon}^{(p-1)}} \tag{A.13}$$

Finite difference approximations to the first and second order partial derivatives are calculated by sampling the matched filtered signal at $t = (k + \hat{\epsilon}^{(p-1)} - \Delta)T$ and $t = (k + \hat{\epsilon}^{(p-1)} + \Delta)T$ where $0 < \Delta << 1$.

After obtaining $\hat{\epsilon}^{(p)}$, estimates of α and χ are calculated conditioned on the new symbol timing estimate. The parameters χ and α are updated during the same conditional maximization step because their updates do not depend on each other. The estimates of α and χ which conditionally maximize $Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(p-1)})$ are

$$\hat{\alpha}^{(p)} = \frac{\left|\sum_{k=0}^{K-1} \left(\bar{x}_k^{(p)}\right)^* y_{k+\tilde{\eta}}(\hat{\epsilon}^{(p)})\right|}{\sum_{k=0}^{K-1} \bar{x^2}_k^{(p)}} \text{ and} \tag{A.14}$$

$$\hat{\chi}^{(p)} = \arg \left\{ e^{-j\frac{2\pi}{\Psi}\tilde{\psi}} \sum_{k=0}^{K-1} \left(\bar{x}_k^{(p)}\right)^* y_{k+\tilde{\eta}}(\hat{\epsilon}^{(p)}) \right\}, \tag{A.15}$$

respectively.

The noise power estimate is necessary to determine the symbol likelihoods used by the SPA as discussed in Section A.4.1. Since the maximization of σ^2 is conditioned on ϵ, we use the log-likelihood function of the sampled matched filter output which is given by

$$\ln f(\mathbf{y}_{\tilde{\eta}}(\hat{\epsilon}^{(p)})|\mathbf{x}, \sigma^2) \\ \propto -K \ln \pi\sigma^2 - \frac{1}{\sigma^2}\left(\sum_{k=0}^{K-1} |y_{k+\tilde{\eta}}(\hat{\epsilon}^{(p)})|^2 - 2\hat{\alpha}^{(p)} \Re \left\{ e^{-j\left(\frac{2\pi}{\Psi}\tilde{\psi}+\hat{\chi}^{(p)}\right)} x_k^* y_{k+\tilde{\eta}}(\hat{\epsilon}^{(p)}) \right\} + \hat{\alpha}^{(p)^2} |x_k|^2 \right). \tag{A.16}$$

Finally, the noise power update equation is given by

$$\hat{\sigma^2}^{(p)} = \frac{1}{K}\left(\sum_{k=0}^{K-1} |y_{k+\tilde{\eta}}(\hat{\epsilon}^{(p)})|^2 - 2\hat{\alpha}^{(p)} \Re \left\{ e^{-j\left(\frac{2\pi}{\Psi}\tilde{\psi}+\hat{\chi}^{(p)}\right)} \left(\bar{x}_k^{(p)}\right)^* y_{k+\tilde{\eta}}(\hat{\epsilon}^{(p)}) \right\} + \hat{\alpha}^{(p)^2} \bar{x^2}_k^{(p)} \right). \tag{A.17}$$

A.4.3 Hypothesis testing

The authors of [128] present a code-aided hypothesis test which is analogous to the EM algorithm. Following their work, the E step requires evaluation of (A.12) for each hypothesis $\{\tilde{\psi}, \tilde{\eta}\}$. In order to handle the continuous parameters $\boldsymbol{\theta}$, we apply a composite hypothesis test approach. For each hypothesis, after performing P iterations, estimates $\boldsymbol{\theta}^{(P)}$ and expectations $\bar{x}_k^{(P)}$ and $\bar{x^2}_k^{(P)}$ are used to evaluate (A.12). The hypothesis test is given by

$$[\hat{\psi}, \hat{\eta}] = \arg\max_{\tilde{\psi},\tilde{\eta}} Q\left(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(P-1)} | \tilde{\psi}, \tilde{\eta}\right)\Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}^{(P)}}, \tag{A.18}$$

where the dependence on $\{\tilde{\psi}, \tilde{\eta}\}$ has been added.

A.5 Numerical Results

A.5.1 Frame timing

Results are first presented for the characterization of the HPD region of the frame offset. Simulations are presented in Figs. A.2 and A.3 for 16QAM and the assumption that all other synchronization parameters are known. In Fig. A.2 the mean sizes of the 99% HPD, 99.9% HPD, and 99.99% HPD regions ($P_{ex} = 10^{-2}, 10^{-3}, 10^{-4}$, respectively) are presented. In Fig. A.3 the P_{ex} achieved in simulation is shown for the three HPDs. The achieved P_{ex} is lower than the target value because an integer number of offsets must be chosen. The results demonstrate the effectiveness of the proposed approach to selecting frame offsets to process in the iterative receiver.

The best decision that a non-iterative receiver can make—assuming equally likely symbols—is to choose the frame offset that maximizes the likelihood, i.e., the ML offset. In contrast, recall that the iterative receiver seeks to approximate the MAP solution to the *joint* detection and synchronization problem using feedback from the decoder. For comparison, Fig. A.3 also shows the P_{ex} of the ML estimate (i.e., $Pr(\eta \neq \hat{\eta}_{\mathrm{ML}})$) achieved by the non-iterative receiver. The main conclusion from this comparison is that, even for relatively high SNR, the P_{ex} of the HPD region is at least two orders of magnitude lower than that of the ML estimate.

Due to the hypothesis testing approach for frame offset detection in the iterative receiver, each frame offset must be demodulated and decoded. Thus, processing two offsets represents approximately a doubling of complexity versus processing a single offset. From Figs. A.2 and A.3, we can determine the complexity required for a given lower bound on performance. For $P_{ex} = 10^{-4}$ the mean HPD size is 8.1 at 4.8 dB (the Shannon bound for 2 bits/channel use) while at a moderate SNR of 8 dB the mean HPD size is 3.6.

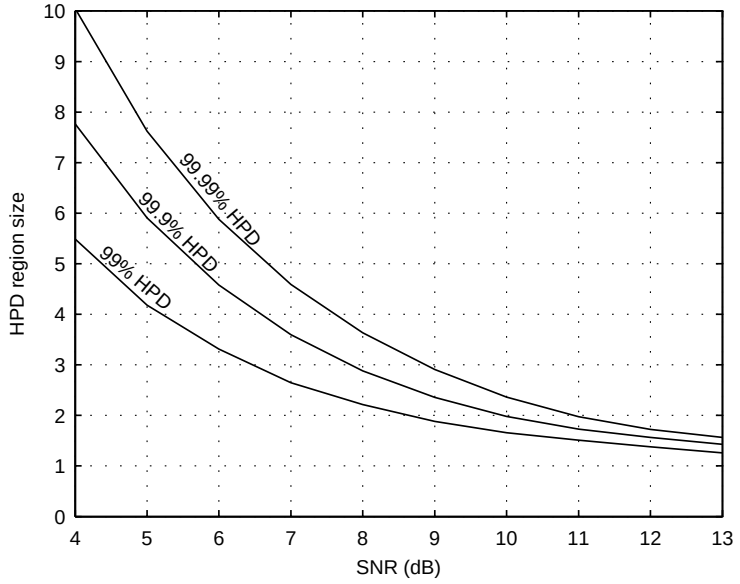

Figure A.2: Mean HPD region size with 16QAM modulation.

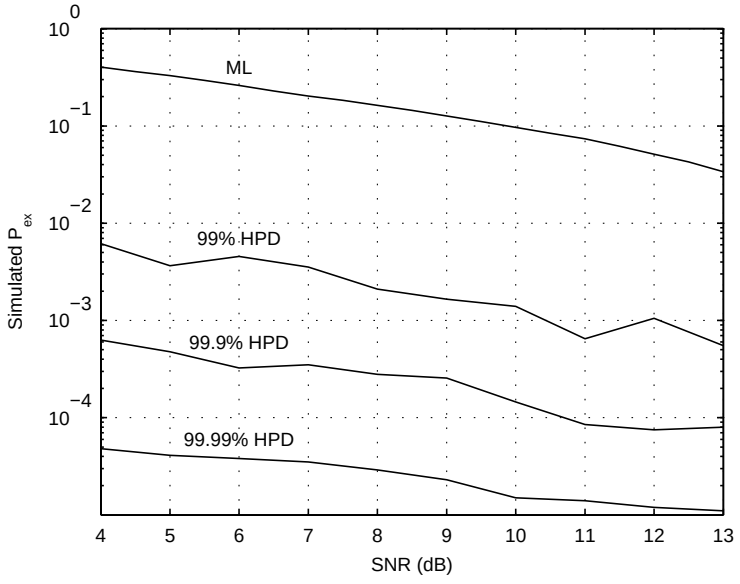

Figure A.3: P_{ex} of the HPD regions and of the ML estimate with 16QAM modulation.

A.5.2 Iterative Receiver Performance

The performance of the iterative receiver is evaluated for a transmission length of 1000 information bits, 16QAM modulation and a 1/2-rate turbo code ($K = 500$). The Oerder and Meyr (O&M) timing detector is used to obtain a coarse estimate of the symbol timing [140] and moment based estimates are obtained for α, χ, and σ^2 [141]. Energy detection is used to find the maximum energy frame offset of the received signal. Then the domain of the frame offset is set relative to this point. We found $H = 51$ to be sufficiently large to ensure that the true offset is always included in the frame offset characterization. In order to minimize complexity without significant performance loss, the target P_{ex} was set to be $1/10$th of the codeword error rate with perfect synchronization. For each hypothesis, $P = 10$ iterations of the ECM algorithm are performed, each of which include a single iteration of the SPA[2].

Simulation results of the frame error rate (FER) are presented for four cases: (1) iterative demodulation and decoding with perfect synchronization (Perfect sync.); (2) hypothesis testing of ψ, η with no iterative estimation of the other parameters (HT); (3) the full iterative receiver with hypothesis testing and the ECM algorithm (HT/ECM); and (4) the full iterative receiver which only considers the ML estimate from the frame offset characterization stage (HT/ECM, ML frame offset). The performance of each case is shown in Fig A.4. The iterative receiver demonstrates performance very near to that of perfect synchronization. In other words, the iterative receiver's ability to perform code-aided synchronization is on the same order as the decoder's ability to successfully decode the information in an ideal channel. Limiting the iterative receiver to only hypothesis testing results in a loss of 0.1-0.2 dB. The performance of the fourth case—considering only the ML frame offset estimate—quickly becomes limited by the P_{ex} (as observed in Fig. A.3).

In the iterative receiver, the frame offset posterior distribution is characterized using the coarse estimates of α, χ, ϵ, and σ^2. In Fig. A.4, the achieved P_{ex} for the iterative receiver performance simulation is shown. When the target P_{ex} is below 10^{-3}, we observe an increase in the probability of excluding the true frame offset from the HPD region as a result of errors in the coarse estimation stage, although the increase is slight.

Table A.1 highlights the frame error performance of the receivers at SNR=7 dB. The number of frame errors is shown by receiver type. Frame errors may originate from excluding the true frame offset from $\mathcal{S}$, choosing an incorrect hypothesis when the true offset has been included in $\mathcal{S}$, and incorrect detection of one or more information bits when the true hypothesis has been chosen. The first two cases generally lead to half the bits being in error (i.e., the BER is significantly affected). The third case generally results in fewer bit errors.

During the frame offset characterization, the true offset was excluded from the HPD region in 30 simulations, which affects both the HT and HT/ECM receivers. When the true offset is included, the HT/ECM receiver always selected the correct hypothesis. Thus, the limiting factor is the frame characterization. We found that the improved estimates obtained by the ECM algorithm enables

[2]Construction of the factor graph and SPA computations were performed using the Dimple probabilistic processing software [182].

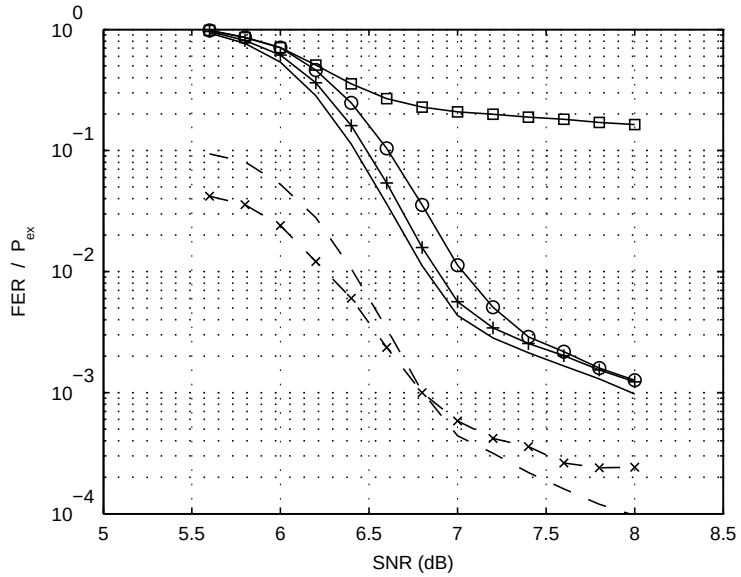

Figure A.4: Results for the FER for Perfect sync. (solid), HT (solid,circle), HT/ECM (solid,plus), and HT/ECM, ML frame offset (solid,square) as well as the target P_{ex} (dash) and the achieved P_{ex} (dash,'x').

more reliable selection of the hypothesis.

In Fig. A.5, the bit error rate (BER) performance of the receiver is shown. On the bit level, there is a significant difference in performance between perfect synchronization and the iterative receiver. This is primarily due to cases where the true offset was excluded from the HPD region. This is verified in Fig. A.6 which shows only those simulations in which the true frame offset is included in $\mathcal{S}$. In Fig. A.6, the BERs of perfect synchronization and HT/ECM are nearly identical while the BER of the HT receiver is higher because hypothesis errors are still made when the true offset is included. This agrees with the data presented in Table A.1. To achieve the BERs shown in Fig. A.6, a larger HPD region is necessary, thereby increasing complexity.

A.5.3 Iterative Receiver Complexity

The dominant source of complexity is from individually demodulating and decoding each hypothesis. The iterative receiver processes $\Psi \cdot |\mathcal{S}|$ hypotheses. Returning to the example from Section A.5.1 with SNR = 8 dB, the average number of hypotheses is 14.4 ($\Psi = 4$, $|\mathcal{S}| = 3.6$). With $P = 10$, the iterative receiver requires a total of 144 decoder iterations on average in comparison to 10 for a traditional receiver (a factor of 14.4).

However, we found through simulation that after performing a single iteration of the ECM algorithm for each hypothesis, detection of the phase ambiguity and frame offset is reliable. After the hypothesis is chosen, further iterations of ECM are run to correctly detect the information

Table A.1: Number of frame errors according to the source of the error at SNR=7 dB

Receiver Type	Total No. Frame Errors†	Exclusion	Inclusion	
		$\eta \notin \mathcal{S}$	Wrong Hypothesis	Noise
Perfect Sync.	200	0	0	200
HT	521	30	32	459
HT/ECM	254	30	0	224

†The total number of frames in the simulation is 47 683.

bits. Thus, from our example, an iterative receiver would require an average of 14.4 iterations for hypothesis testing and 9 iterations to complete ECM algorithm for the selected hypothesis. A total of 23.4 decoder iterations are required on average, making the complexity of the iterative receiver 2.34× that of the traditional receiver. The ECM algorithm does not significantly increase the complexity of the receiver because it is embedded into the decoding process [80].

A.6 Conclusion

In this appendix, an iterative receiver capable of performing synchronization without the use of a preamble was presented. The structure, performance, and complexity of the receiver was considered with specific attention given to frame timing synchronization. A method based on the highest posterior density region was proposed for determining how many and which frame offsets to process in the iterative receiver. This method provides control over the complexity and performance of the receiver. The results demonstrate that the iterative receiver achieves a FER near to that of perfect synchronization with reasonable complexity in comparison to a traditional receiver aided by a preamble.

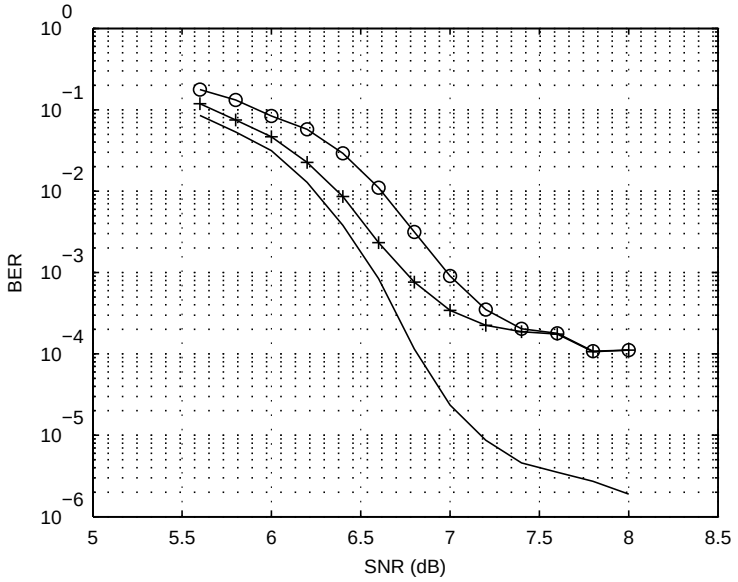

Figure A.5: BER of Perfect sync. (solid), HT (solid,circle), and HT/ECM (solid,plus) for all simulations.

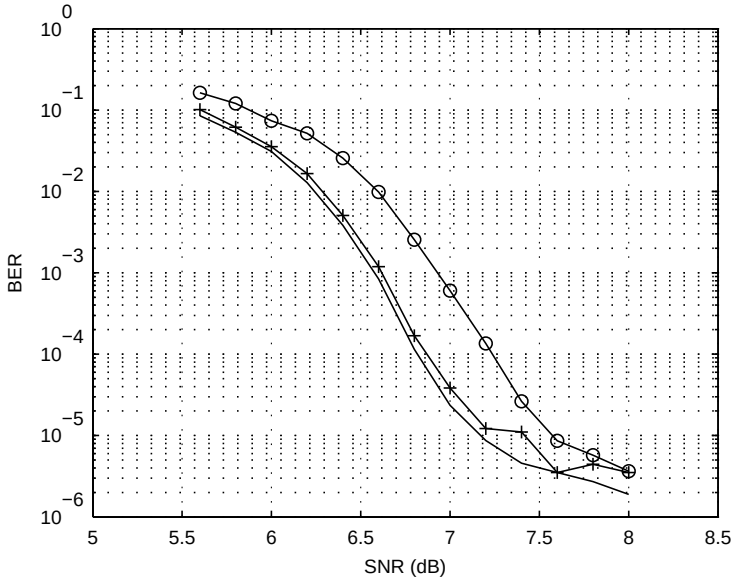

Figure A.6: BER of Perfect sync. (solid), HT (solid,circle), and HT/ECM (solid,plus) where we have excluded simulations in which the true offset is not in $\mathcal{S}$.

Appendix B

Modulation Classification

B.1 Introduction

Traditionally, receivers have been designed to complete tasks such as synchronization, channel estimation, demodulation, and decoding in a sequential fashion. Optimal maximum *a posteriori* (MAP) or maximum likelihood (ML) detection of the information bits requires that these tasks be completed jointly which is prohibitively complex. The term "iterative receiver" refers to a communication receiver in which the MAP or ML estimate of the transmitted message is approximated using an iterative algorithm (i.e., using the turbo principle [67]). In the context of turbo codes, empirical results have shown that iterative decoding provides performance near the Shannon capacity bound [6]. Iterative algorithms for joint synchronization and decoding have been shown to achieve performance near that of decoding with perfect synchronization [13]. Significant contributions have been made to iterative receiver design in the areas of channel estimation [8, 162], equalization [95], and synchronization [13, 82, 180]. Numerous concepts from these areas were joined together in the work by Wymeersch [79].

In this work, we propose incorporating automatic modulation classification into the iterative receiver structure. To the best of our knowledge, this concept has not been previously explored. Adaptive modulation systems are attractive because they provide a means of optimizing link throughput according to channel conditions. Generally, information about the selected modulation scheme must be conveyed to the receiver which wastes bandwidth and energy. In order to avoid this loss, it is desirable that the receiver have the ability to automatically classify the modulation. Automatic modulation classification has also been the focus of significant research effort [183].

The concept behind the proposed iterative receiver is to improve classification by exploiting extrinsic information obtained from soft decoding. The extrinsic information from the soft decoder is treated as *a priori* coded bit probabilities (or equivalently the symbol probabilities). This is in contrast to past research into ML classification in which the symbols are assumed to be equally likely *a priori*. ML modulation classification with equally likely symbols has been shown to have

arbitrarily low probability of error when the number of symbols used by the classifier goes to infinity [184]. In this work we consider relatively short frame lengths and low signal to noise ratio (SNR). We demonstrate through simulation, that the iterative receiver achieves significant performance gains over ML classification performed separately from demodulation and decoding which we will term the "traditional receiver".

Joint modulation classification and decoding requires a receiver structure which supports each modulation scheme simultaneously. Factor graphs provide a useful tool in constructing such a system. The sum-product algorithm is a message passing algorithm for computing marginals on a factor graph [74]. We implement the sum-product algorithm on the factor graph to iteratively solve the joint problem.

In Section B.2 we present background on the system model, factor graphs, and ML classification. In Section B.3 we present the proposed iterative receiver. Simulation results are presented in Section B.4 and conclusions are given in Section B.5.

B.2 Background

B.2.1 System model

We consider a communication system which uses bit interleaved coded modulation (BICM) and may employ one of M amplitude-phase modulation schemes. The number of symbols K in a transmitted frame is fixed and the number of coded bits is allowed to vary according to the selected modulation. In order to handle the variable number of coded bits, the length of the channel encoder is set to provide K coded output bits and the bit interleaver is set to interleave K bits at a time. In this way a single encoder and a single bit interleaver are used for BPSK. In general, when the transmitter uses a modulation with a set of S constellation points, $Z = \log_2 S$ encoder/interleaver blocks are required. This structure for the encoder and interleaver makes it possible to design a receiver which can incorporate the de-interleaving and decoding of each modulation within a single structure. As an example, the diagram of the transmitter for 64QAM is shown in Fig. B.1.

The message sequence is divided into Z blocks $\mathbf{b}_z$, each of which is the input to one of the encoders. The output of the zth encoder $\mathbf{c}_z$ is applied to the zth bit interleaver. The output of the bit interleaver $\mathbf{d}_z$ is a pseudo-random permutation (π) of the input bits. We denote the entire block of information, coded, and interleaved bits as $\mathbf{b} = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_Z]$, $\mathbf{c} = [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_Z]$, and $\mathbf{d} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_Z]$, respectively.

The kth bit from the output of each bit interleaver is denoted $\mathbf{d}_{:,k} = [d_{1,k}, d_{2,k}, \ldots, d_{Z,k}]$ and is passed to the modulator in order to generate symbol y_k. Within the modulator the input bits are interleaved. We call this the "intra-symbol interleaver" and we include it so that the net result of the bit interleavers and the intra-symbol interleaver is to fully interleave the coded bits in keeping with the BICM design of Caire et. al. [185] while maintaining the separate encoder and bit inter-

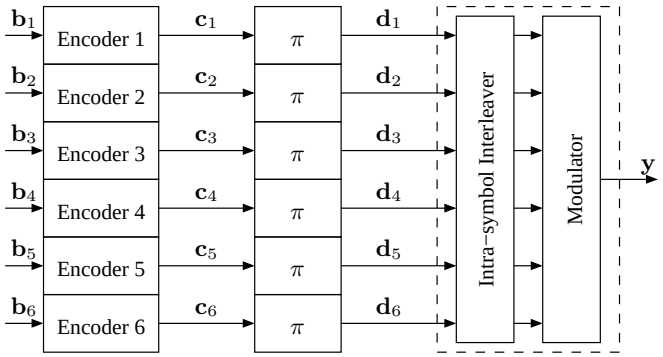

Figure B.1: Diagram of the transmitter structure for 64QAM

leaver blocks. The intra-symbol interleaver mapping is generated randomly for each symbol and is denoted $\psi_{m,k}$ where m refers to the modulation scheme. The modulator employs Gray labeling.

The modulator generates symbols y_k from a unit energy constellation $E[|y_k|^2] = 1$. We assume a system in which the combined response of the pulse shaping filter and receiver's matched filter satisfies the Nyquist condition for zero intersymbol interference. In order to test the concept of the proposed receiver, we simulate transmission of the signal over an additive white Gaussian noise (AWGN) channel. After matched filtering, the signal is sampled at the symbol rate to generate a vector of complex valued samples denoted by $\mathbf{r} = [r_1, r_2, \ldots, r_K]^T$. The complex samples are comprised of symbols and noise as given by $\mathbf{r} = \mathbf{y}+\mathbf{n}$ where $\mathbf{y} = [y_1, y_2, \ldots, y_K]$ and we assume that the noise $\mathbf{n}$ is a vector of independent and identically distributed (iid) complex Gaussian random variables. The noise power is denoted σ_N^2 and is assumed to be known.

B.2.2 Maximum Likelihood Classification

Let $\mathcal{H}_m$ denote the hypothesis that the received frame employs the mth modulation scheme. For an observation of $\mathbf{r}$, the MAP modulation class is the class $\mathcal{H}_m$ which maximizes the probability $\mathrm{P}[\mathcal{H}_m|\mathbf{r}]$. When the modulations are equally likely *a priori*, the MAP and ML classifiers are equivalent as given by [144]

$$\mathrm{P}[\mathcal{H}_m|\mathbf{r}] = \frac{f(\mathbf{r}|\mathcal{H}_m)\mathrm{P}[\mathcal{H}_m]}{f(\mathbf{r})} \propto f(\mathbf{r}|\mathcal{H}_m) \tag{B.1}$$

where we are concerned with the maximum (not the actual probabilities) so multiplicative constants can be ignored.

It is assumed that the symbols are independent [184]. Due to the assumption of iid noise samples and independent symbols, the likelihood function $f(\mathbf{r}|\mathcal{H}_m)$ can be expressed by the product [184]

$$f(\mathbf{r}|\mathcal{H}_m) = \prod_{k=1}^{K} f(r_k|\mathcal{H}_m). \tag{B.2}$$

Let $s_{m,l}$ denote the lth constellation point of the mth modulation and let S_m denote the total number of constellation points in modulation m. The distribution $f(r_k|\mathcal{H}_m)$ is expressed as a marginalization over the joint distribution $f(r_k, s_{m,l}|\mathcal{H}_m)$ [184] which produces

$$\begin{aligned} f(\mathbf{r}|\mathcal{H}_m) &= \prod_{k=1}^{K} \sum_{l=1}^{S_m} f(r_k|s_{m,l}, \mathcal{H}_m) \mathrm{P}[s_{m,l}|\mathcal{H}_m] \\ &= \prod_{k=1}^{K} \sum_{l=1}^{S_m} \frac{1}{\pi\sigma_N^2} \exp\left(-\frac{1}{\sigma_N^2} ||r_k - s_{m,l}||^2\right) \frac{1}{S_m} \\ &\propto \prod_{k=1}^{K} \frac{1}{S_m} \sum_{l=1}^{S_m} \exp\left(-\frac{1}{\sigma_N^2} ||r_k - s_{m,l}||^2\right) \end{aligned} \tag{B.3}$$

where we assume that all symbols within a modulation are equally likely ($p(s_{m,l}|\mathcal{H}_m) = 1/S_m$). From (B.3) we can write the ML classifier as follows

$$\mathcal{H}_{\mathrm{ML}} = \arg\max_{\mathcal{H}_m} \prod_{k=1}^{K} \frac{1}{S_m} \sum_{l=1}^{S_m} \exp\left(-\frac{1}{\sigma_N^2} ||r_k - s_{m,l}||^2\right). \tag{B.4}$$

The result in (B.4) is used for the traditional receiver and will also be useful to verify the sum-product algorithm on the iterative receiver in Section B.3.2.

B.3 Joint Decoding and Classification

The joint probability density function of the presented system model can be factored into several conditional and marginal distributions as given by

$$f(\mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{y}, \mathbf{r}, m) = f(\mathbf{r}|\mathbf{y}) p(\mathbf{y}|\mathbf{d}, m) p(m) p(\mathbf{d}|\mathbf{c}) p(\mathbf{c}|\mathbf{b}) p(\mathbf{b}). \tag{B.5}$$

where $\mathbf{r}$ is the observed sample vector. The modulations are equally likely *a priori*: $p(m) = 1/M$. The likelihood function of the information bits can be determined (up to a scaling factor) as the marginalization of (B.5) over $\mathbf{c}$, $\mathbf{d}$, $\mathbf{y}$, and m. This marginalization is expressed as

$$\begin{aligned} f(\mathbf{r}|\mathbf{b}) &\propto f(\mathbf{r}, \mathbf{b}) \\ &= \sum_{\mathbf{c},\mathbf{d},\mathbf{y},m} f(\mathbf{r}|\mathbf{y}) p(\mathbf{y}|\mathbf{d}, m) p(m) p(\mathbf{d}|\mathbf{c}) p(\mathbf{c}|\mathbf{b}) p(\mathbf{b}) \\ &= p(\mathbf{b}) \sum_{\mathbf{c}} p(\mathbf{c}|\mathbf{b}) \sum_{\mathbf{d}} p(\mathbf{d}|\mathbf{c}) \sum_{\mathbf{y},m} f(\mathbf{r}|\mathbf{y}) p(\mathbf{y}|\mathbf{d}, m) p(m). \end{aligned} \tag{B.6}$$

From the likelihood function of (B.6) we can determine the ML estimate of the information bits. In a similar way, the likelihood function of the modulation can be determined from the marginal-

ization of (B.5) over **b**, **c**, **d**, and **y** as given by

$$\begin{aligned} f(\mathbf{r}|m) &\propto f(\mathbf{r}, m) \\ &= \sum_{\mathbf{b},\mathbf{c},\mathbf{d},\mathbf{y}} f(\mathbf{r}|\mathbf{y}) p(\mathbf{y}|\mathbf{d}, m) p(m) p(\mathbf{d}|\mathbf{c}) p(\mathbf{c}|\mathbf{b}) p(\mathbf{b}) \\ &= p(m) \sum_{\mathbf{y}} f(\mathbf{r}|\mathbf{y}) \sum_{\mathbf{d}} p(\mathbf{y}|\mathbf{d}, m) \sum_{\mathbf{c}} p(\mathbf{d}|\mathbf{c}) \sum_{\mathbf{b}} p(\mathbf{c}|\mathbf{b}) p(\mathbf{b}). \end{aligned} \tag{B.7}$$

The expression for the likelihood function of the modulation (B.7) incorporates information about **b**, **c**, and **d** and therefore is not restricted to the assumption that the symbols are equally likely and independent. The marginalization of (B.6) and (B.7) can be obtained simultaneously from the sum-product algorithm.

B.3.1 Factor graph

In the iterative receiver we require the ability to decode the frame of several modulation schemes. Thus, the number of decoder/de-interleaver blocks is determined by the highest order modulation ($Z = \max_m \log_2 S_m$). The factor graph representing the factorization in (B.5) is given in Fig. B.2. In this appendix we use the 'normal' factor graph representation developed in [73]. Each factor of (B.5) is considered in the paragraphs that follow.

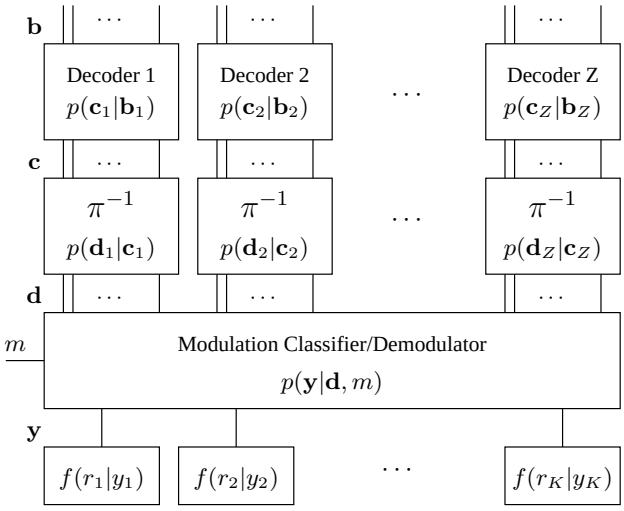

Figure B.2: Iterative receiver factor graph

The encoders and interleavers are deterministic functions of the input. Let the encoder be expressed by the function $g(\cdot)$ such that $\mathbf{c}_z = g(\mathbf{b}_z)$. This allows the conditional distribution $p(\mathbf{c}|\mathbf{b})$ to be expressed

$$p(\mathbf{c}|\mathbf{b}) = \prod_{z=1}^{Z} \mathrm{I}[\mathbf{c}_z = g(\mathbf{b}_z)] \tag{B.8}$$

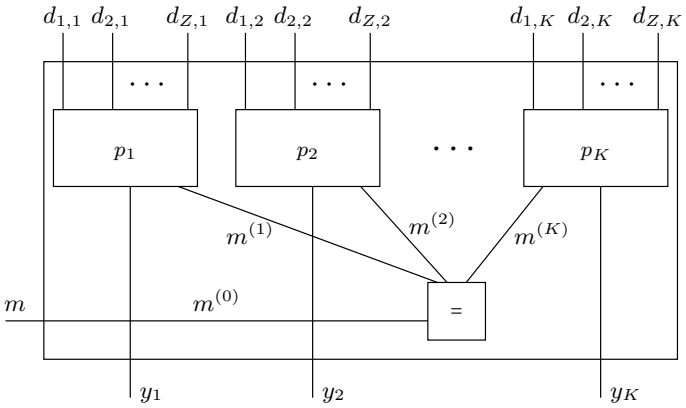

Figure B.3: Factor graph of the joint classifier and demodulator node

where I[·] is the indicator function with value 1 if the expression is true and value 0 otherwise. For additional details on the structure of the decoder and the corresponding sum-product algorithm the reader is referred to [79]. In a similar way, the conditional distribution $p(\mathbf{d}|\mathbf{c})$ may be expressed

$$p(\mathbf{d}|\mathbf{c}) = \prod_{z=1}^{Z} \prod_{k=1}^{K} \mathrm{I}[c_{z,k} = d_{z,\pi(k)}]. \tag{B.9}$$

Using factor graphs, we are able to combine modulation classification and demodulation in a very natural way. The conditional distribution $p(\mathbf{y}|\mathbf{d}, m)$ can be factored into K terms as follows:

$$p(\mathbf{y}|\mathbf{d}, m) = \prod_{k=1}^{K} p_k(y_k|\mathbf{d}_{:,k}, m) \tag{B.10}$$

A detailed view of the classifier/demodulator node is shown in Fig. B.3 where the notation $m^{(k)}$ is used to denote multiple instances of the same modulation variable which are constrained to be equal through the "equals node" [79]. This factorization follows the transmitter's structure which takes an output from each bit interleaver to construct a symbol. The intra-symbol interleaver is not shown in this structure because it is handled within each factor node p_k. Let the function h_m provide a constellation point from the mth modulation given a vector of input bits. This leads to the following function

$$\begin{aligned} p_k(y_k|\mathbf{d}_{:,k}, m) = \mathrm{I}\left[y_k \in \{s_{m,1}, \ldots, s_{m,S_m}\}\right] \\ \cdot \mathrm{I}\left[h_m\left(\psi_{m,k}\left(\mathbf{d}_{:,k}\right)\right) = y_k\right]. \end{aligned} \tag{B.11}$$

The function enforces the structure of the intra-symbol interleaver and modulator for the modulation specified by m.

Finally, the distribution $f(\mathbf{r}|\mathbf{y})$ can be factored into K terms due to the conditional independence of the samples

$$f(\mathbf{r}|\mathbf{y}) = \prod_{k=1}^{K} f(r_k|y_k) = \prod_{k=1}^{K} \exp\left(-\frac{1}{\sigma_N^2} ||r_k - y_k||^2\right). \tag{B.12}$$

B.3.2 Message passing

The high level schedule for the sum-product algorithm on the factor graph of Fig. B.2 begins with computing messages $\mu_{f(r_k|y_k)\to y_k}$. These messages are input to the sum-product algorithm on the classifier/demodulator which will be described in this section. The result of the sum-product algorithm on this block is messages $\mu_{p(\mathbf{y}|\mathbf{d},m)\to m}$ and $\mu_{p(\mathbf{y}|\mathbf{d},m)\to \mathbf{d}}$. The messages $\mu_{p(\mathbf{y}|\mathbf{d},m)\to \mathbf{d}}$ are passed to the de-interleaver which produces $\mu_{p(\mathbf{d}|\mathbf{c})\to \mathbf{c}}$. This provides input to the decoder which provides messages $\mu_{p(\mathbf{c}|\mathbf{b})\to \mathbf{c}}$ and $\mu_{p(\mathbf{c}|\mathbf{b})\to \mathbf{b}}$. The messages from the decoder to the coded bits are passed through the de-interleaver producing $\mu_{p(\mathbf{d}|\mathbf{c})\to \mathbf{d}}$. This provides local *a priori* input to the classifier/demodulator and completes one iteration. Using the new information about the messages $\mu_{p(\mathbf{d}|\mathbf{c})\to \mathbf{d}}$, successive iterations are performed.

We now consider the sum-product algorithm in detail for the classifier/demodulator node. Message passing for the joint classifier and demodulator is outlined in the following steps:

1. Initialize input messages from $\mathbf{y}$ and $\mathbf{d}$ to nodes p_k and from $m^{(0)}$ to the equals node.
2. Compute messages $\mu_{p_k\to m^{(k)}}$ for $k = 1, 2, \ldots, K$.
3. Update the equals node outward messages $\mu_{m^{(k)}\to p_k}$.
4. Compute messages $\mu_{p_k\to d_{z,k}}$ for $z = 1, 2, \ldots, Z$ and $k = 1, 2, \ldots, K$.

In the factor graph framework, the message for a variable out of an equals node is the product of the messages from all other variables connected to the node. Thus, the messages for the modulation variable can be computed as

$$\mu_{eq\to m^{(0)}}(m) = \prod_{k=1}^{K} \mu_{p_k\to m^{(k)}}(m) \tag{B.13}$$

where the marginal is given by

$$p(m|\mathbf{r}) = \mu_{eq\to m^{(0)}} \cdot \mu_{m^{(0)}\to eq} \tag{B.14}$$

and the message $\mu_{m^{(0)}\to eq}$ contains the *a priori* probabilities of the modulation schemes $p(m)$.

In the first iteration of the sum-product algorithm, no prior information about the interleaved bits $\mathbf{d}$ is available. The messages $\mu_{d_{z,k}\to p_k}$ are initialized as uniform distributions. The messages on the edges of the symbol variable y_k contain probabilities for the constellation points in all modulations as given by

$$\mu_{y_k\to p_k}(y_k) = \begin{bmatrix} \exp\left(-\frac{1}{\sigma_N^2}\left\|r_k - s_{1,1}\right\|^2\right) \\ \vdots \\ \exp\left(-\frac{1}{\sigma_N^2}\left\|r_k - s_{m,l}\right\|^2\right) \\ \vdots \\ \exp\left(-\frac{1}{\sigma_N^2}\left\|r_k - s_{M,S_M}\right\|^2\right) \end{bmatrix} \tag{B.15}$$

The message computation for $m^{(k)}$ at each factor p_k is computed according to the sum-product algorithm as follows:

$$\begin{aligned}
&\mu_{p_k \to m^{(k)}}(m) \\
&= \sum_{y_k, \mathbf{d}_{:,k}} p_k(y_k | \mathbf{d}_{:,k}, m) \cdot \mu_{y_k \to p_k}(y_k) \\
&\quad \cdot \mu_{d_{1,k} \to p_k}(d_{1,k}) \cdot \mu_{d_{2,k} \to p_k}(d_{2,k}) \ldots \mu_{d_{Z,k} \to p_k}(d_{Z,k}) \\
&= \frac{1}{2^Z} \sum_{y_k, \mathbf{d}_{:,k}} \mathrm{I}\left[y_k \in \{s_{m,1}, \ldots, s_{m,S_m}\}\right] \\
&\quad \cdot \mathrm{I}\left[g_m\left(\psi_{m,k}\left(\mathbf{d}_{:,k}\right)\right) = y_k\right] \cdot \mu_{y_k \to p_k}(y_k).
\end{aligned} \tag{B.16}$$

For BPSK ($m = 1$) the expression in (B.16) can be further reduced to

$$\begin{aligned}
&\mu_{p_k \to m^{(k)}}(1) \\
&= \frac{1}{2} \sum_{y_k, d_{1,k}} \mathrm{I}\left[y_k \in \{s_{1,1}, s_{1,2}\}\right] \\
&\qquad \cdot \mathrm{I}\left[g_1\left(d_{1,k}\right) = y_k\right] \cdot \mu_{y_k \to p_k}(y_k) \\
&= \frac{1}{2}\left(\exp\left(-\frac{1}{\sigma_N^2} \left\| r_k - s_{1,1} \right\|^2 \right) \right. \\
&\qquad \left. + \exp\left(-\frac{1}{\sigma_N^2} \left\| r_k - s_{1,2} \right\|^2 \right) \right).
\end{aligned} \tag{B.17}$$

Equivalent results are obtained for other modulations. Substituting a general form of these results into (B.13) produces

$$\mu_{eq \to m^{(0)}}(m) = \prod_{k=1}^{K} \frac{1}{S_m} \sum_{l=1}^{S_m} \exp\left(-\frac{1}{\sigma_N^2} \left\| r_k - s_{m,l} \right\|^2 \right) \tag{B.18}$$

which is identical to (B.4) for ML classification and provides a verification of the factor graph. In successive iterations the messages $\mu_{p(\mathbf{d}|\mathbf{c}) \to \mathbf{d}}$ provide feedback from the decoder and are not necessarily uniform.

B.4 Simulation Results

In order to demonstrate the iterative receiver we simulate the design for $M = 4$ with modulations BPSK, QPSK, 16QAM, and 64QAM. The highest order modulation (64QAM) requires $Z = 6$ blocks to be incorporated into the receiver structure. The encoder employs a $1/2$-rate convolutional code with a constraint length of $k_c = 7$ (64 state). The octal generators of the code are (133,

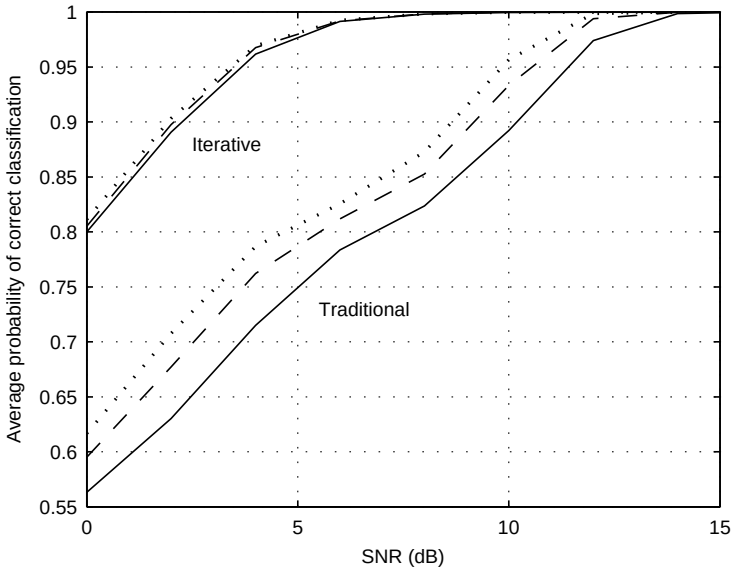

Figure B.4: Average classification performance of the iterative and traditional receivers with frame lengths of $K = 50$ (solid), $K = 100$ (dash), and $K = 150$ (dot) symbols.

171) and provide the maximum free distance [167]. A zero tail is used to convert the convolutional code into a block code. The receiver's decoder utilizes a sum-product implementation of the BCJR algorithm. We consider frame lengths of 50, 100, and 150 symbols in order to determine the influence that the frame size has on the classifier's performance. The receiver performs 10 iterations of the sum-product algorithm. Fig. B.4 displays the probability of correct modulation classification averaged over all modulations. The iterative receiver demonstrates a gain of 5 to 8 dB over ML classification in the traditional receiver.

At low SNR, the most significant gains in performance are for 16QAM which has the worst performance in the traditional receiver. In Table B.1, the confusion tables for the iterative and traditional receivers are given at an SNR of 2 dB. ML classification has a significant bias towards classifying 16QAM as 64QAM. The iterative receiver nearly eliminates this bias while improving overall classification.

The end goal is to correctly detect the transmitted message. Improving the classification performance is useful if this translates to a reduction in the frame error rate (FER) or bit error rate (BER). For our purposes frame errors include both modulation classification errors and frame detection errors. In order to provide a reference point to the performance of the iterative and traditional receivers, we also simulate the performance with known modulation. In all cases iterative demodulation and decoding is performed. Fig. B.5 shows the FER of the iterative and traditional receivers. Improved classification performance in the iterative receiver does in fact lead to improvements in the FER of the system. For BPSK and 64QAM, the iterative receiver achieves ideal performance (that which is achieved with known modulation). For QPSK and 16QAM the system

Table B.1: Confusion table for SNR = 2 dB, $K = 100$, and 10 iterations

Iterative Receiver

True\Hypothesis	BPSK	QPSK	16QAM	64QAM
BPSK	100	0	0	0
QPSK	0	99.07	0.47	0.46
16QAM	0	4.26	78.35	17.39
64QAM	0	4.23	14.17	81.59

Traditional Receiver

True\Hypothesis	BPSK	QPSK	16QAM	64QAM
BPSK	100	0	0	0
QPSK	0	81.10	14.70	4.20
16QAM	0	19.49	36.57	43.94
64QAM	0	14.31	32.50	53.20

is limited by the classifier, but still improves upon the traditional receiver by as much as 3 dB.

When considering FER, the performance of the traditional receiver employing 64QAM is already very near the ideal performance. Therefore a comparison of BER performance versus E_b/N_0 is given in Fig. B.6 where $E_b/N_0 = SNR \cdot 1/\log_2 S_m \cdot K/(K/2-6)$. For the purposes of simulating the BER we compare the transmitted sequence with the appropriate bits at the receiver. In terms of BER, the iterative receiver achieves a 2 dB gain over the traditional receiver for $E_b/N_0 < 8$ dB.

B.5 Conclusion

An iterative receiver for solving the problem of joint modulation classification, demodulation, and decoding is proposed. The iterative receiver is based on factor graphs and the associated sum-product algorithm. The performance of the iterative receiver is quantified with respect to the performance of a receiver which completes the task of modulation classification separately from demodulation and decoding. The iterative receiver is shown to provide greater than 5 dB gain in terms of modulation classification. By improving the reliability of automatic modulation classification, the proposed iterative receiver enables efficient use of the spectrum at low SNR.

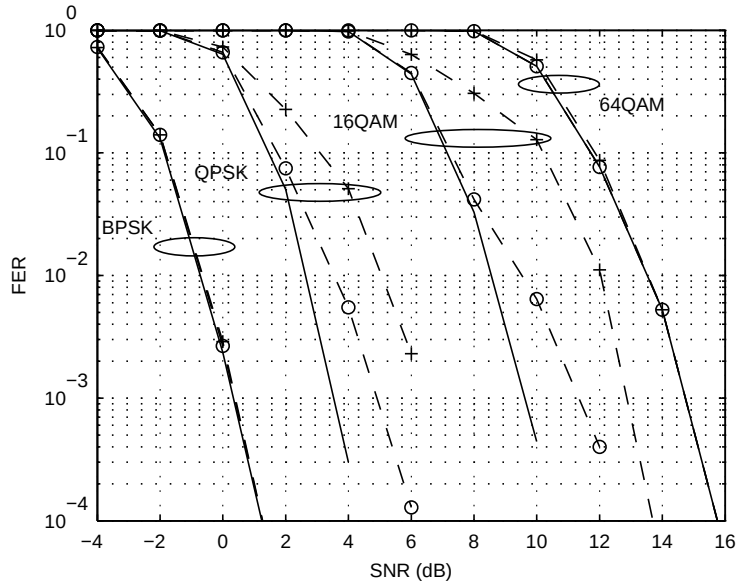

Figure B.5: FER performance by modulation scheme for known modulation (solid), the iterative receiver (dash, circle), and the traditional receiver (dash, plus) for a frame length of $K = 100$.

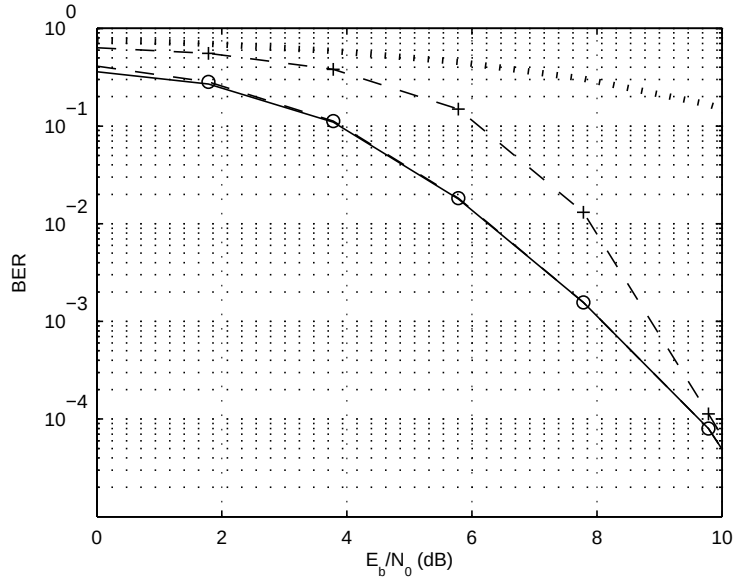

Figure B.6: BER performance of 64QAM for known modulation (solid), the iterative receiver (dash, circle), the traditional receiver (dash, plus), and uncoded transmission (dot) for a frame length of $K = 100$.

Bibliography

[1] ITU-R, "Imt vision–framework and overall objectives of the future development of imt for 2020 and beyond," ITU-R M.2083-0, Sep. 2015.

[2] J. G. Andrews, "Seven ways that HetNets are a cellular paradigm shift," *IEEE Commun. Mag.*, vol. 51, no. 3, pp. 136–144, Mar. 2013.

[3] F. Boccardi, R. Heath, A. Lozano, T. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 74–80, Feb. 2014.

[4] FCC, "Enabling innovative small cell use in 3.5 GHz band NPRM & order," Dec. 2012.

[5] N. Lee, R. W. Heath, D. Morales-Jimenez, and A. Lozano, "Base station cooperation with dynamic clustering in super-dense cloud-RAN," in *Proc. 2013 IEEE Globecom Workshops*, Dec. 2013, pp. 784–788.

[6] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. IEEE ICC*, vol. 2, Geneva, Switzerland, May 1993, pp. 1064–1070.

[7] F. R. Kschischang, B. J. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[8] A. P. Worthen and W. E. Stark, "Unified design of iterative receivers using factor graphs," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 843–849, Feb. 2001.

[9] R. H. Barker, "Group synchronizing of binary digital systems," *Communication Theory*, pp. 273–287, 1953.

[10] C. Herzet and L. Vandendorpe, "Code-aided ML ambiguity resolution," in *Proc. IEEE ICC*, Jun. 2007, pp. 2900–2905.

[11] R. Imad, G. Sicot, and S. Houcke, "Blind frame synchronization for error correcting codes having a sparse parity check matrix," *IEEE Trans. Commun.*, vol. 57, no. 6, pp. 1574–1577, Jun. 2009.

[12] R. Imad and S. Houcke, "Theoretical analysis of a MAP based blind frame synchronizer," *IEEE Trans. Wireless Commun.*, vol. 8, no. 11, pp. 5472–5476, Nov. 2009.

[13] H. Wymeersch, H. Steendam, H. Bruneel, and M. Moeneclaey, "Code-aided frame synchronization and phase ambiguity resolution," *IEEE Trans. Signal Process.*, vol. 54, no. 7, pp. 2747–2757, Jul. 2006.

[14] P. Robertson, "A generalized frame synchronizer," in *Conf. Rec. IEEE GLOBECOM*, Dec. 1992, pp. 365–369.

[15] J. S. Yedidia, W. Freeman, and Y. Weiss, "Constructing free-energy approximations and generalized belief propagation algorithms," *IEEE Trans. Inf. Theory*, vol. 51, no. 7, pp. 2282–2312, Jul. 2005.

[16] T. Moon and J. Gunther, "Multiple-access via turbo joint equalization," *IEEE Trans. Commun.*, vol. 60, no. 10, pp. 3001–3010, Oct. 2012.

[17] D. Middleton, "Statistical-physical models of electromagnetic interference," *IEEE Trans. Electromagn. Compat.*, vol. EMC-19, no. 3, pp. 106–127, Aug. 1977.

[18] K. Blackard, T. Rappaport, and C. Bostian, "Measurements and models of radio frequency impulsive noise for indoor wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 11, no. 7, pp. 991–1001, Sep. 1993.

[19] W. Lauber and J. Bertrand, "Statistics of motor vehicle ignition noise at VHF/UHF," *IEEE Trans. Electromagn. Compat.*, vol. 41, no. 3, pp. 257–259, Aug. 1999.

[20] M. Sanchez, L. de Haro, M. Ramon, A. Mansilla, C. Ortega, and D. Oliver, "Impulsive noise measurements and characterization in a UHF digital TV channel," *IEEE Trans. Electromagn. Compat.*, vol. 41, no. 2, pp. 124–136, May 1999.

[21] M. Sanchez, A. Alejos, and I. Cuinas, "Urban wide-band measurement of the UMTS electromagnetic environment," *IEEE Trans. Veh. Technol.*, vol. 53, no. 4, pp. 1014–1022, Jul. 2004.

[22] M. Nassar, X. Lin, and B. Evans, "Stochastic modeling of microwave oven interference in WLANs," in *Proc. IEEE ICC*, Jun. 2011, pp. 1–6.

[23] M. Nassar, K. Gulati, M. R. DeYoung, B. L. Evans, and K. R. Tinsley, "Mitigating near-field interference in laptop embedded wireless transceivers," *J. Signal Process. Syst.*, vol. 63, no. 1, pp. 1–12, Apr. 2011.

[24] H. A. DeJarnette, "Measuring noise in the VHF band and its effect on low SNR signal detection," Master's thesis, Virginia Polytechnic Institute and State University, 2012.

[25] C. Luschi and B. Mulgrew, "Nonparametric trellis equalization in the presence of non-Gaussian interference," *IEEE Trans. Commun.*, vol. 51, no. 2, pp. 229–239, Feb. 2003.

[26] K. Gulati, B. Evans, J. Andrews, and K. Tinsley, "Statistics of co-channel interference in a field of Poisson and Poisson-Poisson clustered interferers," *IEEE Trans. Signal Process.*, vol. 58, no. 12, pp. 6207–6222, Dec. 2010.

[27] X. Wang and H. Poor, "Robust multiuser detection in non-Gaussian channels," *IEEE Trans. Signal Process.*, vol. 47, no. 2, pp. 289–305, Feb. 1999.

[28] X. Wang and R. Chen, "Adaptive Bayesian multiuser detection for synchronous CDMA with Gaussian and impulsive noise," *IEEE Trans. Signal Process.*, vol. 48, no. 7, pp. 2013–2028, Jul. 2000.

[29] V. Chavali and C. R. C. M. da Silva, "Maximum-likelihood classification of digital amplitude-phase modulated signals in flat fading non-Gaussian channels," *IEEE Trans. Commun.*, vol. 59, no. 8, pp. 2051–2056, Aug. 2011.

[30] M. Nassar, P. Schniter, and B. Evans, "A factor graph approach to joint OFDM channel estimation and decoding in impulsive noise environments," *IEEE Trans. Signal Process.*, vol. 62, no. 6, pp. 1576–1589, Mar. 2014.

[31] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc., Ser. B*, vol. 39, no. 1, pp. 1–38, Jan. 1977.

[32] B. Hu, I. Land, L. Rasmussen, R. Piton, and B. Fleury, "A divergence minimization approach to joint multiuser decoding for coded CDMA," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 3, pp. 432–445, Apr. 2008.

[33] G. Kirkelund, C. Manchon, L. Christensen, E. Riegler, and B. Fleury, "Variational message-passing for joint channel estimation and decoding in MIMO-OFDM," in *Proc. IEEE GLOBECOM*, Dec. 2010, pp. 1–6.

[34] C. N. Manchón, G. E. Kirkelund, E. Riegler, L. P. Christensen, and B. H. Fleury, "Receiver architectures for MIMO-OFDM based on a combined VMP-SP algorithm," *arXiv preprint arXiv:1111.5848*, Nov. 2011.

[35] E. Riegler, G. Kirkelund, C. Manchon, M. Badiu, and B. Fleury, "Merging belief propagation and the mean field approximation: A free energy approach," *IEEE Trans. Inf. Theory*, vol. 59, no. 1, pp. 588–602, Jan. 2013.

[36] C. N. Manchon, G. E. Kirkelund, B. Fleury, P. Mogensen, L. Deneire, T. B. Sorensen, and C. Rom, "Interference cancellation based on divergence minimization for MIMO-OFDM receivers," in *Proc. IEEE GLOBECOM*, Nov 2009, pp. 1–6.

[37] M. A. Badiu, C. N. Manchn, V. Bota, and B. H. Fleury, "Distributed iterative processing for interference channels with receiver cooperation," in *Proc. 7th International Symposium on Turbo Codes and Iterative Information Processing (ISTC)*, Aug. 2012, pp. 140–144.

[38] C. N. Manchon, B. Fleury, G. E. Kirkelund, P. Mogensen, L. Deneire, T. B. Sorensen, and C. Rom, "Channel estimation based on divergence minimization for OFDM systems with co-channel interference," in *Proc. IEEE ICC*, Jun. 2009, pp. 1–6.

[39] M. Witzke, S. Baro, F. Schreckenbach, and J. Hagenauer, "Iterative detection of MIMO signals with linear detectors," in *Conf. Rec. 36th Asilomar Conf. Signals Syst. Comput.*, vol. 1, 2002, pp. 289–293.

[40] L. Ping, L. Liu, K. Wu, and W. K. Leung, "Interleave division multiple-access," *IEEE Trans. Wireless Commun.*, vol. 5, pp. 938–947, Apr. 2006.

[41] P. Salvo Rossi and R. Muller, "Joint twofold-iterative channel estimation and multiuser detection for MIMO-OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 7, no. 11, pp. 4719–4729, 2008.

[42] W. Jiang and D. Li, "Iterative single-antenna interference cancellation: algorithms and results," *IEEE Trans. Veh. Technol.*, vol. 58, no. 5, pp. 2214–2224, Jun. 2009.

[43] D. J. Jakubisin and R. M. Buehrer, "Approximate joint MAP detection of co-channel signals," in *Proc. IEEE MILCOM*, Tampa, FL, Oct. 2015, pp. 1530–1535.

[44] ——, "Approximate joint MAP detection of co-channel signals in non-Gaussian noise," *IEEE Trans. Commun., submitted for publication*.

[45] T. P. Minka, "Expectation propagation for approximate Bayesian inference," in *Proc. 17th Conf. Uncertainty Artificial Intell.*, ser. UAI'01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 362–369.

[46] J. Hu, H.-A. Loeliger, J. Dauwels, and F. Kschischang, "A general computation rule for lossy summaries/messages with examples from equalization," in *Proc. 44th Allerton Conf. Commun. Control Comput.*, Monticello, IL, Sep. 2006.

[47] M. Senst and G. Ascheid, "How the framework of expectation propagation yields an iterative IC-LMMSE MIMO receiver," in *Proc. IEEE GLOBECOM*, Dec. 2011, pp. 1–6.

[48] P. Sun, C. Zhang, Z. Wang, C. Navarro Manchon, and B. Fleury, "Iterative receiver design for ISI channels using combined belief- and expectation-propagation," *IEEE Signal Process. Lett.*, vol. PP, no. 99, pp. 1–1, 2015.

[49] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. Inform. Theory*, Jul. 2011, pp. 2168–2172.

[50] Z. Yuan, C. Zhang, Z. Wang, Q. Guo, S. Wu, and X. Wang, "A low-complexity receiver using combined BP-MF for joint channel estimation and decoding in OFDM systems," *arXiv preprint arXiv:1601.05856*, 2016.

[51] N. Noels, H. Steendam, and M. Moeneclaey, "Carrier and clock recovery in (turbo-) coded systems: Cramér-Rao bound and synchronizer performance," *EURASIP J. Appl. Signal Process.*, vol. 2005, pp. 972–980, Jan. 2005.

[52] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.

[53] Y.-N. Lee, A. Ashikhmin, and J.-T. Chen, "Impact of soft channel construction on iterative channel estimation and data decoding for multicarrier systems," *IEEE Trans. Wireless Commun.*, vol. 7, no. 7, pp. 2762–2770, Jul. 2008.

[54] T. Wo, P. A. Hoeher, and Z. Shi, "Graph-based soft channel estimation for fast fading channels," *IEEE Trans. Wireless Commun.*, vol. 11, no. 12, pp. 4243–4251, Dec. 2012.

[55] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Generalized belief propagation," in *Proc. Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2001, vol. 13, pp. 689–695.

[56] T. Heskes, "Stable fixed points of loopy belief propagation are minima of the Bethe free energy," in *Proc. Advances Neural Inform. Process. Syst. (NIPS)*, vol. 15, 2003, pp. 359–366.

[57] D. J. Jakubisin, R. M. Buehrer, and C. R. C. M. da Silva, "Probabilistic receiver architecture combining BP, MF, and EP for multi-signal detection," *IEEE Trans. Commun., submitted for publication*.

[58] D. J. Jakubisin and R. M. Buehrer, "Performance, complexity, and receiver design for code-aided frame synchronization in multipath channels," *IEEE Trans. Commun.*, vol. 63, no. 9, pp. 3363–3376, Sep. 2015.

[59] D. J. Jakubisin, R. M. Buehrer, and C. R. C. M. da Silva, "BP, MF, and EP for joint channel estimation and detection of MIMO-OFDM signals," in *Proc. IEEE GLOBECOM, submitted for publication*, 2016.

[60] D. Jakubisin and R. M. Buehrer, "On the complexity-performance trade-off in code-aided frame synchronization," in *Proc. 15th IEEE Int. Workshop SPAWC*, Toronto, ON, Canada, Jun. 2014, pp. 364–368.

[61] D. Jakubisin, C. I. Phelps, and R. M. Buehrer, "Iterative joint detection, decoding, and synchronization with a focus on frame timing," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Istanbul, Turkey, Apr. 2014, pp. 446–451.

[62] D. Jakubisin and R. M. Buehrer, "Improved modulation classification using a factor-graph-based iterative receiver," in *Proc. IEEE MILCOM*, Oct. 2012, pp. 1–6.

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

[64] R. G. Gallager, *Low density parity check codes*. MIT Press, 1963.

[65] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.

[66] T. Richardson, M. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.

[67] J. Hagenauer, "The turbo principle: Tutorial introduction and state of the art," in *Proc. Int. Symp. Turbo Codes Relat. Topics*, 1997, pp. 1–11.

[68] R. M. Tanner, "A recursive approach to low complexity codes," *Information Theory, IEEE Transactions on*, vol. 27, no. 5, pp. 533–547, Sep. 1981.

[69] G. D. Forney, "The viterbi algorithm," *Proc. IEEE*, vol. 61, no. 3, pp. 268–278, Mar. 1973.

[70] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*. San Francisco, CA: Morgan Kaufmann, 1988.

[71] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Linköping University, Sweden, 1996.

[72] B. J. Frey, *Graphical models for machine learning and digital communication*. MIT press, 1998.

[73] G. D. Forney, "Codes on graphs: normal realizations," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 520–548, Feb. 2001.

[74] H. A. Loeliger, "An introduction to factor graphs," *IEEE Signal Process. Mag.*, vol. 21, no. 1, pp. 28–41, Jan. 2004.

[75] H. A. Loeliger, J. Dauwels, J. Hu, S. Korl, L. Ping, and F. R. Kschischang, "The factor graph approach to model-based signal processing," *Proceedings of the IEEE*, vol. 95, no. 6, pp. 1295–1322, Jun. 2007.

[76] F. L. Spitzer, "Random fields and interacting particle systems," in *M.A.A. Summer Seminar Notes*. Mathematical Association of America, 1971.

[77] R. Kindermann and J. L. Snell, *Markov Random Fields and Their Applications*. American Mathematical Society, 1980.

[78] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA, USA: The MIT Press, 2009.

[79] H. Wymeersch, *Iterative Receiver Design*. Cambridge Univ. Press, 2007.

[80] N. Noels, C. Herzet, A. Dejonghe, V. Lottici, H. Steendam, M. Moeneclaey, M. Luise, and L. Vandendorpe, "Turbo synchronization: an EM algorithm interpretation," in *Proc. IEEE ICC*, vol. 4, 2003, pp. 2933–2937.

[81] C. Herzet, V. Ramon, L. Vandendorpe, and M. Moeneclaey, "EM algorithm-based timing synchronization in turbo receivers," in *Proc. IEEE ICASSP*, vol. 4, 2003, pp. IV–612–IV–615.

[82] N. Noels, V. Lottici, A. Dejonghe, H. Steendam, M. Moeneclaey, M. Luise, and L. Vandendorpe, "A theoretical framework for soft-information-based synchronization in iterative (turbo) receivers," *EURASIP J. Wireless Commun. Netw.*, vol. 2005, no. 2, pp. 117–129, Apr. 2005.

[83] C. Herzet, X. Wautelet, V. Ramon, and L. Vandendorpe, "Iterative synchronization: EM algorithm versus Newton-Raphson method," in *Proc. IEEE ICASSP*, vol. 4, 2006, pp. IV–IV.

[84] C. Herzet, V. Ramon, and L. Vandendorpe, "A theoretical framework for iterative synchronization based on the sum-product and the expectation-maximization algorithms," *IEEE Trans. Signal Process.*, vol. 55, no. 5, pp. 1644–1658, May 2007.

[85] C. Herzet, H. Wymeersch, M. Moeneclaey, and L. Vandendorpe, "On maximum-likelihood timing synchronization," *IEEE Trans. Commun.*, vol. 55, no. 6, pp. 1116–1119, Jun. 2007.

[86] A. Barbieri, G. Colavolpe, and G. Caire, "Joint iterative detection and decoding in the presence of phase noise and frequency offset," *IEEE Trans. Commun.*, vol. 55, no. 1, pp. 171–179, Jan. 2007.

[87] X. Zhuang and F. W. Vook, "Iterative channel estimation and decoding for a turbo-coded OFDM system via the EM algorithm," in *Proc. IEEE ICASSP*, vol. 3, May 2002, pp. III–2337–III–2340.

[88] E. Jaffrot and M. Siala, "Turbo channel estimation for OFDM systems on highly time and frequency selective channels," in *Proc. IEEE ICASSP*, vol. 5, 2000, pp. 2977–2980.

[89] X. Ma, H. Kobayashi, and S. Schwartz, "An EM-based estimation of OFDM signals," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, vol. 1, Mar. 2002, pp. 228–232.

[90] M. Badiu, C. Manchon, and B. Fleury, "Message-passing receiver architecture with reduced-complexity channel estimation," *IEEE Commun. Lett.*, vol. 17, no. 7, pp. 1404–1407, Jul. 2013.

[91] P. Hammarberg, F. Rusek, and O. Edfors, "Iterative receivers with channel estimation for multi-user MIMO-OFDM: complexity and performance," *EURASIP J. on Wireless Commun. and Netw.*, vol. 2012, pp. 1–17, Mar. 2012.

[92] C. Knievel, P. Hoeher, A. Tyrrell, and G. Auer, "Multi-dimensional graph-based soft iterative receiver for MIMO-OFDM," *IEEE Trans. Commun.*, vol. 60, pp. 1599–1609, Jun. 2012.

[93] Y. Liu, G. Y. Li, H. Hu, and Z. Tan, "Map based iterative channel estimation for OFDM systems: Approach, convergence, and performance bound," *IEEE Trans. Wireless Commun.*, vol. 13, no. 1, pp. 476–485, Jan. 2014.

[94] A. Glavieux, C. Laot, and J. Labat, "Turbo equalization over a frequency selective channel," in *Proc. Int. Symp. Turbo Codes*, 1997, pp. 96–102.

[95] M. Tuchler, R. Koetter, and A. C. Singer, "Turbo equalization: principles and new results," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 754–767, May 2002.

[96] R. Drost and A. Singer, "Factor-graph algorithms for equalization," *IEEE Trans. Signal Process.*, vol. 55, no. 5, pp. 2052–2065, 2007.

[97] B. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. Commun.*, vol. 51, pp. 389–399, Mar. 2003.

[98] S. Haykin, M. Sellathurai, Y. de Jong, and T. Willink, "Turbo-MIMO for wireless communications," *IEEE Commun. Mag.*, vol. 42, pp. 48–53, Oct. 2004.

[99] B. Farhang-Boroujeny, H. Zhu, and Z. Shi, "Markov chain Monte Carlo algorithms for CDMA and MIMO communication systems," *IEEE Trans. Signal Process.*, vol. 54, pp. 1896–1909, May 2006.

[100] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 764–785, Feb. 2011.

[101] C. Zhang, C. N. Manchn, Z. Wang, and B. H. Fleury, "Message-passing receivers for single carrier systems with frequency-domain equalization," *IEEE Signal Process. Lett.*, vol. 22, no. 4, pp. 404–407, Apr. 2015.

[102] A. L. Yuille, "CCCP algorithms to minimize the Bethe and Kikuchi free energies: Convergent alternatives to belief propagation," *Neural Computation*, vol. 14, no. 7, pp. 1691–1722, Jul. 2002.

[103] Y. W. Teh and M. Welling, "The unified propagation and scaling algorithm," in *Advances in Neural Information Processing Systems 14*, T. Dietterich, S. Becker, and Z. Ghahramani, Eds. MIT Press, 2002, pp. 953–960. [Online]. Available: http://papers.nips.cc/paper/2001-the-unified-propagation-and-scaling-algorithm.pdf

[104] T. Heskes, "Convexity arguments for efficient minimization of the Bethe and Kikuchi free energies," *J. Artificial Intell. Research (JAIR)*, vol. 26, pp. 153–190, 2006.

[105] ——, "On the uniqueness of loopy belief propagation fixed points," *Neural Computation*, vol. 16, no. 11, pp. 2379–2413, Nov. 2004.

[106] A. T. Ihler, J. Iii, and A. S. Willsky, "Loopy belief propagation: Convergence and effects of message errors," in *J. Machine Learning Research*, 2005, pp. 905–936.

[107] J. Mooij and H. Kappen, "Sufficient conditions for convergence of the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4422–4437, Dec. 2007.

[108] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky, "A new class of upper bounds on the log partition function," *IEEE Trans. Inf. Theory*, vol. 51, no. 7, pp. 2313–2335, Jul. 2005.

[109] ——, "MAP estimation via agreement on trees: message-passing and linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 11, pp. 3697–3717, Nov. 2005.

[110] Y. Weiss, C. Yanover, and T. Meltzer, "MAP estimation, linear programming and belief propagation with convex free energies," in *Proc. 19th Conf. Uncertainty Artifical Intell. (UAI)*, 2007.

[111] T. Werner, "A linear programming approach to max-sum problem: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 7, pp. 1165–1179, Jul. 2007.

[112] A. Globerson and T. S. Jaakkola, "Fixing max-product: Convergent message passing algorithms for MAP LP-relaxations," in *Proc. Advances Neural Inform. Process. Syst. (NIPS)*, vol. 20, 2008, pp. 553–560.

[113] T. Hazan and A. Shashua, "Norm-product belief propagation: Primal-dual message-passing for approximate inference," *IEEE Trans. Inf. Theory*, vol. 56, no. 12, pp. 6294–6316, Dec. 2010.

[114] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Understanding belief propagation and its generalizations," *Exploring artificial intelligence in the new millennium*, vol. 8, pp. 236–239, 2003.

[115] M. Welling and Y. W. Teh, "Linear response algorithms for approximate inference in graphical models," *Neural Computation*, vol. 16, no. 1, pp. 197–221, Jan. 2004.

[116] N. Komodakis, N. Paragios, and G. Tziritas, "MRF energy minimization and beyond via dual decomposition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 3, pp. 531–552, Mar. 2011.

[117] Q. Guo and L. Ping, "LMMSE turbo equalization based on factor graphs," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 2, pp. 311–319, Feb. 2008.

[118] X. L. Meng and D. B. Rubin, "Maximum likelihood estimation via the ECM algorithm: A general framework," *Biometrika*, vol. 80, no. 2, pp. 267–278, Jun. 1993.

[119] T. P. Minka, "A family of algorithms for approximate Bayesian inference," Ph.D. dissertation, Massachusetts Institute of Technology, 2001.

[120] ——, "Divergence measures and message passing," Microsoft Research, Tech. Rep. MSR-TR-2005-173, Dec. 2005.

[121] J. L. Massey, "Optimum frame synchronization," *IEEE Trans. Commun.*, vol. 20, no. 2, pp. 115–119, Apr. 1972.

[122] G. Lui and H. Tan, "Frame synchronization for Gaussian channels," *IEEE Trans. Commun.*, vol. 35, no. 8, pp. 818–829, Aug. 1987.

[123] P. Robertson, "Improving frame synchronization when using convolutional codes," in *Conf. Rec. IEEE GLOBECOM*, vol. 3, Nov. 1993, pp. 1606–1611.

[124] M. Howlader and B. Woerner, "Decoder-assisted frame synchronization for packet transmission," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 12, pp. 2331–2345, Dec. 2001.

[125] J. Sodha, "Turbo code frame synchronization," *Signal Process. J., Elsevier*, vol. 82, no. 5, pp. 803–809, 2002.

[126] T. Cassaro and C. Georghiades, "Frame synchronization for coded systems over AWGN channels," *IEEE Trans. Commun.*, vol. 52, no. 3, pp. 484–489, Mar. 2004.

[127] J. Sun and M. Valenti, "Optimum frame synchronization for preamble-less packet transmission of turbo codes," in *Conf. Rec. 38th Asilomar Conf. Signals Syst. Comput.*, 2004, pp. 1126–1130.

[128] C. Herzet, H. Wymeersch, F. Simoens, M. Moeneclaey, and L. Vandendorpe, "MAP-based code-aided hypothesis testing," *IEEE Trans. Wireless Commun.*, vol. 7, no. 8, pp. 2856–2860, Aug. 2008.

[129] C. Herzet, K. Woradit, H. Wymeersch, and L. Vandendorpe, "Code-aided maximum-likelihood ambiguity resolution through free-energy minimization," *IEEE Trans. Signal Process.*, vol. 58, no. 12, pp. 6238–6250, Dec. 2010.

[130] H. Huh and J. Krogmeier, "A unified approach to optimum frame synchronization," *IEEE Trans. Wireless Commun.*, vol. 5, no. 12, pp. 3700–3711, Dec. 2006.

[131] D.-U. Lee, H. Kim, C. Jones, and J. Villasenor, "Pilotless frame synchronization for LDPC-coded transmission systems," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 2865–2874, Jul. 2008.

[132] C. Stefanovic, D. Vukobratovic, and D. Bajic, "Low-complexity list-based frame synchronization for LDPC coded transmission," in *Proc. IEEE ICC*, Jun. 2009, pp. 1–5.

[133] R. Moosavi and E. Larsson, "Fast blind recognition of channel codes," *IEEE Trans. Commun.*, vol. 62, no. 5, pp. 1393–1405, May 2014.

[134] P. Robertson, "Maximum likelihood frame synchronization for flat fading channels," in *Conf. Rec. IEEE ICC*, vol. 3, Jun. 1992, pp. 1426–1430.

[135] B. Moon and S. Soliman, "ML frame synchronization for the Gaussian channel with ISI," in *Conf. Rec. IEEE ICC*, vol. 3, Jun. 1991, pp. 1698–1702.

[136] M. K. Simon and M.-S. Alouini, "A unified approach to the performance analysis of digital communication over generalized fading channels," *Proc. IEEE*, vol. 86, no. 9, pp. 1860–1877, Sep. 1998.

[137] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate (corresp.)," *IEEE Trans. Inf. Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[138] J. Gill, *Bayesian methods: A social and behavioral sciences approach*, 2nd ed. CRC press, 2007.

[139] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inf. Theory*, vol. 42, no. 2, pp. 429–445, Mar. 1996.

[140] M. Oerder and H. Meyr, "Digital filter and square timing recovery," *IEEE Trans. Commun.*, vol. 36, no. 5, pp. 605–612, May 1988.

[141] A. Abdi, O. Dobre, R. Choudhry, Y. Bar-Ness, and W. Su, "Modulation classification in fading channels using antenna arrays," in *Proc. IEEE MILCOM*, vol. 1, 2004, pp. 211–217.

[142] L. Tong and S. Perreau, "Multichannel blind identification: from subspace to maximum likelihood methods," *Proc. IEEE*, vol. 86, no. 10, pp. 1951–1968, Oct. 1998.

[143] U. Mengali and A. N. D'Andrea, *Synchronization techniques for digital receivers*. New York: Plenum Press, 1997.

[144] S. M. Kay, *Fundamentals of Statistical Signal Processing: Detection Theory*. Prentice-Hall, 1998.

[145] S. Verdu, "Minimum probability of error for asynchronous Gaussian multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 32, no. 1, pp. 85–96, Jan. 1986.

[146] J. Lee, D. Toumpakaris, and W. Yu, "Interference mitigation via joint detection," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 6, pp. 1172–1184, Jun. 2011.

[147] P. Wolniansky, G. Foschini, G. Golden, and R. Valenzuela, "V-BLAST: an architecture for realizing very high data rates over the rich-scattering wireless channel," in *Proc. URSI Int. Symposium on Signals, Systems, and Electronics (ISSSE)*, Sep. 1998, pp. 295–300.

[148] X. Wang and H. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, no. 7, pp. 1046–1061, Jul. 1999.

[149] C. Tidestav, M. Sternad, and A. Ahlen, "Reuse within a cell—interference rejection or multiuser detection?" *IEEE Trans. Commun.*, vol. 47, no. 10, pp. 1511–1522, Oct. 1999.

[150] H. Arslan and K. Molnar, "Cochannel interference suppression with successive cancellation in narrow-band systems," *IEEE Commun. Lett.*, vol. 5, no. 2, pp. 37–39, Feb. 2001.

[151] J. Hicks, J. Tsai, J. Reed, W. Tranter, and B. Woerner, "Overloaded array processing with MMSE-SIC," in *Proc. IEEE 55th Vehicular Technology Conf.*, vol. 2, 2002, pp. 542–546.

[152] J. Boutros and G. Caire, "Iterative multiuser joint decoding: unified framework and asymptotic analysis," *IEEE Trans. Inf. Theory*, vol. 48, no. 7, pp. 1772–1793, Jul. 2002.

[153] J. Lee, H. Kwon, and I. Kang, "Interference mitigation in MIMO interference channel via successive single-user soft decoding," in *Proc. Inform. Theory Appl. (ITA) Workshop*, 2012, pp. 180–185.

[154] P. Hoeher, S. Badri-Hoeher, W. Xu, and C. Krakowski, "Single-antenna co-channel interference cancellation for TDMA cellular radio systems," *IEEE Wireless Commun. Mag.*, vol. 12, no. 2, pp. 30–37, Apr. 2005.

[155] J. Andrews, "Interference cancellation for cellular systems: a contemporary overview," *IEEE Wireless Commun. Mag.*, vol. 12, no. 2, pp. 19–29, Apr. 2005.

[156] X. Wang and H. V. Poor, *Wireless communication systems: Advanced techniques for signal reception*. Prentice Hall Professional, 2004.

[157] E. Biglieri, R. Calderbank, A. Constantinides, A. Goldsmith, A. Paulraj, and H. V. Poor, *MIMO Wireless Communications*. Cambridge University Press, 2007.

[158] H. L. Van Trees, *Optimum Array Processing - Part IV, Detection, Estimation, and Modulation Theory*. John Wiley & Sons, 2002.

[159] L. Ping, L. Liu, and W. K. Leung, "A simple approach to near-optimal multiuser detection: interleave-division multiple-access," in *Proc. IEEE Wireless Commun. Netw. Conf.*, vol. 1, 2003, pp. 391–396.

[160] C.-H. Yih, "Iterative interference cancellation for OFDM signals with blanking nonlinearity in impulsive noise channels," *IEEE Signal Process. Lett.*, vol. 19, no. 3, pp. 147–150, Mar. 2012.

[161] K. Nieman, M. Nassar, J. Lin, and B. Evans, "FPGA implementation of a message-passing OFDM receiver for impulsive noise channels," in *Proc. Asilomar Conf. Signals Syst. Comput.*, Nov. 2013, pp. 2041–2045.

[162] G. Caire, A. Tulino, and E. Biglieri, "Iterative multiuser joint detection and parameter estimation: a factor-graph approach," in *Proc. IEEE Inform. Theory Workshop*, 2001, pp. 36–38.

[163] M. Kobayashi, J. Boutros, and G. Caire, "Successive interference cancellation with SISO decoding and EM channel estimation," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 8, pp. 1450–1460, Aug. 2001.

[164] Y. Zhu, D. Guo, and M. Honig, "A message-passing approach for joint channel estimation, interference mitigation, and decoding," *IEEE Trans. Wireless Commun.*, vol. 8, pp. 6008–6018, Dec. 2009.

[165] C. Herzet, N. Noels, V. Lottici, H. Wymeersch, M. Luise, M. Moeneclaey, and L. Vandendorpe, "Code-aided turbo synchronization," *Proc. IEEE*, vol. 95, no. 6, pp. 1255–1271, Jun. 2007.

[166] G. Colavolpe and G. Germi, "On the application of factor graphs and the sum-product algorithm to ISI channels," *IEEE Trans. Commun.*, vol. 53, no. 5, pp. 818–825, May 2005.

[167] J. Proakis and M. Salehi, *Digital Communications*, 5th ed. Boston:McGraw-Hill, 2008.

[168] P. Needham, A. W. Gotz, and R. C. Walker, "Why graphics processing units," in *Electronic Structure Calculations on Graphics Processing Units: From Quantum Chemistry to Condensed Matter Physics*, R. C. Walker and A. W. Gotz, Eds. John Wiley & Sons, Ltd, Chichester, UK, 2016.

[169] X. Yu, P. Lin, Z. He, and W. Wu, "OFDM channel estimation with impulse noise cancellation," in *Proc. Int. Conf. Wireless Commun. Networking and Mobile Computing (WiCom)*, Sep. 2007, pp. 330–333.

[170] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," *Found. Trends Mach. Learning*, vol. 1, no. 1-2, pp. 1–305, Jan. 2008.

[171] M. Senst and G. Ascheid, "A combined belief propagation and mean field algorithm for soft carrier phase estimation," in *Proc. 8th Int. Symp. on Wireless Communication Systems (ISWCS)*, Nov 2011, pp. 512–516.

[172] ——, "A message passing approach to iterative Bayesian SNR estimation," in *Proc. Int. Symp. on Signals, Systems, and Electronics (ISSSE)*, Oct. 2012, pp. 1–6.

[173] M. A. Badiu, G. E. Kirkelund, C. N. Manchn, E. Riegler, and B. H. Fleury, "Message-passing algorithms for channel estimation and decoding using approximate inference," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Jul. 2012, pp. 2376–2380.

[174] T. L. Hansen, P. B. Jørgensen, M.-A. Badiu, and B. H. Fleury, "Joint sparse channel estimation and decoding: Continuous and discrete domain sparsity," *arXiv preprint arXiv:1507.02954*, Jul. 2015.

[175] O.-E. Barbu, C. N. Manchn, C. Rom, T. Balercia, and B. H. Fleury, "OFDM receiver for fast time-varying channels using block-sparse Bayesian learning," *IEEE Trans. Veh. Technol., submitted for publication*, 2016.

[176] C. Zhang, Z. Yuan, Z. Wang, and Q. Guo, "Low complexity sparse Bayesian learning using combined BP and MF with a stretched factor graph," *arXiv preprint arXiv:1602.07762*, 2016.

[177] S. Wu, L. Kuang, Z. Ni, J. Lu, D. D. Huang, and Q. Guo, "Expectation propagation approach to joint channel estimation and decoding for OFDM systems," in *Proc. IEEE ICASSP*, May 2014, pp. 1941–1945.

[178] S. Wu, L. Kuang, Z. Ni, Q. Guo, J. Lu *et al.*, "Message-passing receiver for joint channel estimation and decoding in broadband massive MIMO systems," *arXiv preprint arXiv:1509.09059*, 2015.

[179] W. Wang, Z. Wang, C. Zhang, Q. Guo, P. Sun, and X. Wang, "A BP-MF-EP based iterative receiver for joint phase noise estimation, equalization and decoding," *arXiv preprint arXiv:1603.04163*, 2016.

[180] J. Dauwels and H. A. Loeliger, "Phase estimation by message passing," in *Proc. IEEE ICC*, vol. 1, Jun. 2004, pp. 523–527.

[181] T. M. Cover and J. A. Thomas, *Elements of information theory*. Hoboken, N.J. Wiley-Interscience, 2006.

[182] S. Hershey, J. Bernstein, B. Bradley, A. Schweitzer, N. Stein, T. Weber, and B. Vigoda, "Accelerating inference: towards a full language, compiler and hardware stack," *arXiv:1212.2991*, Dec. 2012.

[183] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of automatic modulation classification techniques: classical approaches and new trends," *IET Commun.*, vol. 1, no. 2, pp. 137–156, Apr. 2007.

[184] W. Wei and J. M. Mendel, "Maximum-likelihood classification for digital amplitude-phase modulations," *IEEE Trans. Commun.*, vol. 48, no. 2, pp. 189–193, Feb. 2000.

[185] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 927–946, May 1998.