Combined Message Passing Algorithms for Iterative Receiver Design in Wireless Communication Systems

Zhang, Chuanzong

DOI (link to publication from Publisher):
10.5278/VBN.PHD.ENGSCI.00123

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

COMBINED MESSAGE PASSING ALGORITHMS FOR ITERATIVE RECEIVER DESIGN IN WIRELESS COMMUNICATION SYSTEMS

BY
CHUANZONG ZHANG

DISSERTATION SUBMITTED 2016

AALBORG UNIVERSITY
DENMARK

Combined Message Passing Algorithms for Iterative Receiver Design in Wireless Communication Systems

Ph.D. Dissertation

Chuanzong Zhang

Aalborg University
Department of Electronic Systems
Fredrik Bajers Vej 7B
DK-9220 Aalborg

Thesis submitted:	June, 2016
PhD Supervisor:	Prof. Dr. Sc. Techn. Bernard Henri Fleury Aalborg University
Assistant PhD Supervisor:	Assoc. Prof. Ph.D. Carles Navarro Manchón Aalborg University
PhD Committee:	Associate Professor Daniel Enrique Lucani Roetter (chairman) Department of Electronic Systems Aalborg University Professor Luc Vandendorpe Institute of Information and Communication Technologies Electronics and Applied Mathematics Université catholique de Louvain (UCL) Belgium Professor Christoph Mecklenbräuker Institute of Telecommunications Vienna University of Technology (VUT) Austria
PhD Series:	Faculty of Engineering and Science, Aalborg University

ISSN: (online):2246-1248
ISBN: (online):978-87-7112-744-7

Published by:
Aalborg University Press
Skjernvej 4A, 2nd floor
DK – 9220 Aalborg Ø
Phone: +45 99407140
aauf@forlag.aau.dk
forlag.aau.dk

Printed in Denmark by Rosendahls, 2016

Abstract

Statistical inference using message passing on factor graphs provides a useful and versatile tool for the design of iterative receivers in wireless communications, as shown by the large number of research articles proposing such solutions during the last decade. Among the different methods, belief propagation (BP), the mean field (MF) approximation, and expectation propagation (EP) have been prevalent. Each of these methods is especially suited for different types of problems, which has motivated the use of algorithms combining two or more of them. These combinations, whether heuristic or based on well-founded theoretical grounds, allow for overcoming tractability and complexity issues present in the individual methods.

In this thesis, we research the application of message passing methods and – combination thereof – to the design of receivers for various wireless communication systems. BP is firstly considered as it typically leads to better performance, while MF or EP can be used when the computation of BP messages is highly complex or intractable. Among others, we study the design of message passing receivers for turbo-equalization of inter-symbol interference channels, frequency domain turbo-equalization, channel estimation and decoding in multicarrier systems, and phase noise estimation and decoding. By appropriately combining message computation rules belonging to different frameworks, we obtain designs that are superior to the state-of-art counterparts in decoding performance, computational complexity, or both. Moreover, in order to obtain feasible receiver algorithms for these concrete problems, we propose approximations of intractable messages produced by BP, that are shown to be practical, have low-complexity, and do not degrade the performance of the receivers significantly.

Based on the good performance of our proposed receivers, we conclude that there is the room and the need for further research towards the theoretical formalization of statistical inference algorithms combining two or more message passing methods. This would further expand the set of tools available for finding the best possible compromise between receiver performance and computational complexity in future wireless receivers.

Resumé

Statistisk inferens ved brug af message passing metoder baseret på faktor grafer er et brugbart og fleksibelt værktøj til design af iterative modtager algoritmer til trådløs kommunikation, hvilket bevidnes af det store antal forskningsartikler der er udgivet om emnet gennem det sidste årti. Adskellige typer af metoder har været anvendt, hvoraf modtagere baseret på belief propagation (BP), mean-field (MF) approksimationen samt expectation propagation (EP) har været fremherskende. Hver af disse metoder er særligt egnede for forskellige typer problemer, hvilket har motiveret brugen af hybridalgoritmer hvor flere metoderne kombineres. Disse hybridalgoritmer, hvad enten de er baseret på heuristik eller et velfunderet teoretisk grundlag, gør det muligt at reducere beregnings- og kompleksitets problemer ved de oprindelige algoritmer.

I denne afhandling, undersøger vi anvendelsen af message passing metoder, særligt hybride algoritmer, til modtagerdesign for forskellige trådløse kommunikationssystemer. BP betragtes først, da den typisk giver bedre resultater, mens MF eller EP kan bruges ved de noder hvor BP beskederne er meget komplekse eller analytisk uhåndterlige. Yderligere undersøges en metode til at approksimation af beskederne for at til nedbringelse af algoritmens beregningskompleksitet. Her undersøges bland andet design af message passing modtagere til turbo-equalisering af kanaler med inter-symbol interferens, frekvens-domæne turbo-equalisering, kanalestimering og dekodning i systemer med flere bærebølger, samt estimering af fasestøj og dekodning. Ved at kombinere forskellige message passing metoder, opnås modtagere med højere dekodningspræcision og lavere kompleksitet. For at opnå brugbare modtageralgoritmer til disse konkrete problemer, foreslår vi metoder til at tilnærme de uhåndterlige beskeder der opstår i BP algoritmer. Det vises at disse tilnærmede metoder er praktisk anvendelige, har lav kompleksitet og at de kun medfører en ubetydelig reduktion af modtagernes ydeevne.

Baseret på den gode ydeevne der opnås af vores foreslåede modtagere, konkluderer vi at der er mulighed og behov for yderligere studier der søger at finde en teoretisk formalisme af message passing metoder der kombinerer flere principper for inferens. Dette vil yderligere øge udvalget af tilgængelige

værktøjer til at opnå det bedste kompromis mellem algoritmisk kompleksitet og modtager ydeevne ved design af fremtidige modtager algoritmer.

Contents

Thesis Details

Thesis Title: Combined Message Passing Algorithms for Iterative Receiver Design in Wireless Communication Systems
Ph.D. Student: Chuanzong Zhang
Supervisors: Prof. Dr. Sc. Techn. Bernard Henri Fleury, Aalborg University
Assoc. Prof. Ph.D. Carles Navarro Manchón, Aalborg University

The main body of this thesis consists of the following papers.

[A] Peng Sun, Chuanzong Zhang, Zhongyong Wang, Carles Navarro Manchón and Bernard Henri Fleury, "Iterative receiver design for ISI channels using combined belief- and expectation-propagation," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1733–1737, 2015.

[B] Chuanzong Zhang, Zhongyong Wang, Peng Sun, Qinghua Guo, and Bernard Henri Fleury, "Turbo equalization using partial Gaussian approximation", accepted by *IEEE Signal Processing Letters*, 2016. Available online: http://arxiv.org/abs/1603.04142

[C] Chuanzong Zhang, Carles Navarro Manchón, Zhongyong Wang and Bernard Henri Fleury "Message-passing receivers for single carrier systems with frequency-domain equalization," *IEEE Signal Processing Letters*, vol. 24, no. 4, pp. 404–207, 2015.

[D] Chuanzong Zhang, Zhengdao Yuan, Zhongyong Wang, and Qinghua Guo, "Low complexity sparse Bayesian learning using combined BP and MF with a stretched factor graph," submitted to *Signal processing (Elsevier)*, 2016. Available online: http://arxiv.org/abs/1602.07762

[E] Zhengdao Yuan, Chuanzong Zhang, Zhongyong Wang, Qinghua Guo, Sheng Wu, and Xingye Wang, "A low complexity OFDM receiver with combined GAMP and MF message passing," submitted to *IEEE Trans. Veh. Technol.*, 2016 Available online: http://arxiv.org/abs/1601.05856

[F] Wei Wang, Zhongyong Wang, Chuanzong Zhang, Qinghua Guo, Peng Sun, and Xingye Wang, "A BP-MF-EP based iterative receiver for joint phase noise estimation, equalization and decoding," resubmitted to *IEEE Signal Processing Letters*, 2016.
Available online: http://arxiv.org/abs/1603.04163

In addition to the main papers, the following publication has also been made.

[G] Jianhua Cui, Zhongyong Wang, Chuanzong Zhang, Zhengyu Zhu, and Peng Sun, "Variational message passing based localization algorithm with Taylor expansion for wireless sensor networks," accepted by *IET Communications*, 2016.

This thesis has been submitted for assessment in partial fulfillment of the PhD degree. The thesis is based on the submitted or published scientific papers which are listed above. Parts of the papers are used directly or indirectly in the extended summary of the thesis. As part of the assessment, co-author statements have been made available to the assessment committee and are also available at the Faculty. The thesis is not in its present form acceptable for open publication but only in limited and closed circulation as copyright may not be ensured.

Preface

This thesis is submitted to the Doctoral School of Engineering and Science at Aalborg University, Denmark, in partial fulfilment of the requirements for the degree of doctor of philosophy. The first part of the thesis contains an introduction and a brief description of the contributions obtained throughout the study. As an outcome, six journal papers are attached. The doctoral work has been supported in part by the China scholarship council (CSC) and the National Natural Science Foundation of China.

I would have not finished this thesis without help. I am very grateful to my supervisor, Professor Bernard Henri Fleury, for sharing his deep insights in many research fields, for setting up a high research standard for me, for always being patient, and for investing consistent efforts into making me a better technical writer. I am also very very grateful to my co-supervisor, Carles Navarro Manchón, who has always been there to listen to me as a friend and who has helped me formulate my work in a more precise way. I am very grateful to Professor Zhongyong Wang from Zhengzhou University, China, who has given me invaluable guidance throughout my studies. In particular, I would like to thank him for supporting my research in Zhengzhou University. I would like to acknowledge my present and former colleagues and fellow PhD students both at Aalborg University and Zhengzhou University for many pleasant and inspiring discussions. I would like to thank our secretaries and their colleagues for the help and assistance throughout the years. Special thanks to my lovely wife, Xiaoxiao Li. Thanks for being there all the time.

Chuanzong Zhang
Aalborg University, June 29, 2016

Part I

Introduction

Chapter 1

Introduction

The ultimate goal of a digital receiver is to optimally recover a sequence of information bits that modulate a signal sent from a transmitter through a propagation channel. Since these bits are unknown in the first place, and the transmitted signal is distorted by an unknown propagation channel, unknown interferences and noise before it reaches the receiver, a common approach used to design the receiver is to consider such entities –information bits, channel response, various types of interferences, noise, and other unknowns– as random variables or processes. This probabilistic modeling forms the baseline for a mathematical formalization of the process of recovery of the transmitted information using methods from statistical inference, and allows us to define what optimal recovery means in this context.

The probabilistic model describing a modern digital communication system often includes both discrete random variables, such as information bits, codewords and modulated data symbols, and continuous random variables, such as channel attenuations and noise. While most engineers in information technology areas, such as communications, signal processing, *etc.*, prefer to distinguish between the concepts of estimation of continuous variables and detection of discrete variables [1, 2], in this doctoral thesis we deliberately avoid such a distinction. We exploit unified inference methods to globally deal with continuous and discrete variables in the probabilistic model describing the whole digital communication system. We call on Bayesian inference [3, 4], which will allow us to formulate a suitable optimization problem, the solution of which will give our desired optimal data recovery. Hence, before considering the design of practical receivers, we briefly introduce the background of it.

There are two important optimality criteria in Bayesian inference: the minimum mean square error (MMSE) criterion and the maximum a posteriori (MAP) criterion [1, 5]. To compute the optimal (MMSE or MAP) estimate of a given variable from some observation, one requires the knowledge of the posterior probability density function (pdf) of that variable given said observation – the received signal in the context of digital communications. When the observed data also statistically depends on several unknown random variables, the computation of the desired posterior pdf involves marginalization operations over all unknown variables but the variable of interest. One generally computes the marginal posterior pdf by means of the total probability theorem and Bayes theorem. Computing this pdf is referred to as ***exact inference*** [1, 3, 4]. Unfortunately, modern communication systems may involve thousands or even more random variables, which renders exact inference impractical at best, and simply infeasible in most cases. Indeed, one can typically not obtain a closed-form expression of the desired posterior pdf except for the simplest types of probabilistic models, which often do not faithfully represent the reality observed in practical communication systems.

One popular approach for coping with such cases is based on particles representation of probability distributions. The Markov chain Monte Carlo (MCMC) method [6–8] is a typical implementation of this approach. Particle-based methods operate with samples drawn from the variables' distributions. These samples are used to compute approximations of cumbersome distributions, and/or their characteristics, e.g. their moments. To achieve a sufficient accuracy of the estimates obtained with such methods, however, requires a large number of samples drawn; hence these methods involve high computational complexity. Our focus in this thesis is on an alternative methodology for approximate statistical inference that is based on ***variational inference methods*** [9, 10] and their applications in the design of receivers for communication systems [11–16].

Variational inference methods can often be iteratively implemented by means of algorithms passing messages along the edges of a factor graph [17] that represents the probabilistic (system) model under consideration. Factor graphs are particularly well-suited for representing large probabilistic systems. Out of all message passing techniques, we focus on three main methods: belief propagation (BP) [17], expectation propagation (EP) [18, 19], and the mean field (MF) [20–22] approximation. All three methods have in common the fact that they are derived from the stationary point equations of the variational free energy or approximations thereof, subject to specific constraints [10, 23]. Recently, unified message passing frameworks in which BP and MF [23] or EP [24, 25] can be combined on a single factor graph have been proposed. These combined schemes keep the virtues of each of the methods but avoid their respective drawbacks.

In this thesis, we investigate combined message passing methods and their application to the unified design of receivers for different wireless communication systems. The receivers obtained in this way exhibit an iterative structure in which different blocks implementing specific tasks – such as channel estimation, noise precision (inverse variance) estimation, phase noise estimation, channel equalization or decoding – exchange information. Since, occasionally, direct application of the message passing methods yields receivers with impractically high computational complexity, we also explore approximate techniques aimed to reduce the computational burden. For simplicity, we restrict our study to single-input single-output wireless communication links operating in interference-free scenarios. Hence, the effects of multi-user and co-channel interferences are not taken into account in the considered systems. We emphasize, however, that the proposed techniques can be extended to multi-antenna and/or multi-user systems by appropriately modifying the probabilistic models of the studied problems.

This thesis is organized as a collection of scientific articles, found in Part II. To put the contributions of these papers in a common context and provide necessary background information, a brief introduction to the articles is provided in Chapters 2 through 5 of Part I. We introduce the background of variational inference in Chapter 2. The message update rules for BP, EP and MF are presented in Chapter 3. In this chapter we also describe combined message passing methods and some approximate techniques to simplify the computation of problematic messages. Chapter 4 introduces the state of the art on receivers for the considered communication systems and summarizes the key contributions of the PhD thesis. Conclusions and outlook are given in Chapter 5. Papers A-F in Part II are the articles listed in Chapter 4. They contain the main scientific contributions of this thesis.

Chapter 2

Variational Inference

In this chapter, we shortly review the fundamental concepts related to variational Bayesian inference on large probabilistic systems. First, we set the notation that we will use henceforth to describe a generic probabilistic system or model, and introduce the graphical representation of it via factor graphs. Then we briefly discuss Bayesian inference on such a model before addressing the approximate variational inference methods that we will use in this work.

1 Large Probabilistic Systems, Factor Graph Representation and Exact Inference

Bayesian inference is an important method of statistical inference whereby the variables of interest in the underlying probabilistic system are considered to be random. In this section, we will describe the framework of Bayesian inference and two important Bayesian estimators both for continuous and for discrete variables. Standard works on Bayesian theory can be found in [1, 5].

1.1 Large Probabilistic Systems

Let $\{X_i; i \in \mathcal{I}\}$ be a finite family of random variables, indexed by $\mathcal{I} = [1 : N]$. Let $x_i \in \mathcal{X}_i$ represent a realization of random variable X_i, which takes values in the range $\mathcal{X}_i$. Define the vectors $\boldsymbol{X} = [X_i; i \in \mathcal{I}]^{\mathrm{T}}$ and $\boldsymbol{x} = [x_i; i \in \mathcal{I}]^{\mathrm{T}}$. We often do not observe the instance $\boldsymbol{x}$ of $\boldsymbol{X}$ directly, but only through an observation $\boldsymbol{y} \in \mathcal{Y}$ of a random vector $\boldsymbol{Y}$. The joint pdf of $\boldsymbol{X}$ and $\boldsymbol{Y}$ is denoted by $p_{\boldsymbol{X},\boldsymbol{Y}}(\boldsymbol{x}, \boldsymbol{y}) : \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_N \times \mathcal{Y} \to \mathbb{R}$. A common task on such models is to infer the value of the realizations of $\boldsymbol{X}$ given the observation of a realization of $\boldsymbol{Y}$. Typically this is done based on the a-posteriori pdf of $\boldsymbol{X}$, which according to Bayes' theorem can be expressed as

$$p_{\boldsymbol{X}|\boldsymbol{Y}}(\boldsymbol{x}|\boldsymbol{y}) = \frac{p_{\boldsymbol{X},\boldsymbol{Y}}(\boldsymbol{x},\boldsymbol{y})}{p_{\boldsymbol{Y}}(\boldsymbol{y})}.$$

For a fixed observation $\boldsymbol{Y} = \boldsymbol{y}$, $\boldsymbol{y}$ can be treated as a constant and the above pdf can be represented by the more general relation:

$$p_{\boldsymbol{X}|\boldsymbol{Y}}(\boldsymbol{x}|\boldsymbol{y}) = \frac{1}{Z} \prod_{a\in\mathcal{A}} f_a(\boldsymbol{x}_a). \tag{2.1}$$

where the explicit dependence on $\boldsymbol{y}$ has been omitted on the right hand side, and only dependence on $\boldsymbol{x}$ is explicitly denoted. In the general expression (2.1), $\prod_{a\in\mathcal{A}} f_a(\boldsymbol{x}_a)$ represents a given factorization of the joint pdf $p_{\boldsymbol{X},\boldsymbol{Y}}(\boldsymbol{x},\boldsymbol{y})$, while Z takes the place of $p_{\boldsymbol{Y}}(\boldsymbol{y})$, which is just a normalization constant for a fixed observation $\boldsymbol{Y} = \boldsymbol{y}$. The set $\mathcal{A} = [1 : M]$ is the indices set of the factors, and factor $f_a(\boldsymbol{x}_a)$ has as argument the vector $\boldsymbol{x}_a$ of which the entries are a subset of $\{x_i; i \in \mathcal{I}\}$. Without loss of generality, we assume that the index sets fulfill $\mathcal{A} \cap \mathcal{I} = \emptyset$.

1.2 A Short Introduction to Factor Graphs

A factor graph provides a graphical representation of the functional relationships between the variables in a probabilistic model that can be expressed as a product of factors as in (2.1). The factor graph representation has been widely applied in coding, signal and image processing, statistical machine learning and artificial intelligence in recent years [17, 26]. It is a useful tool to design efficient iterative message passing algorithms within various variational inference frameworks. The factor graph representation is a simple and intuitive alternative to Markov random fields [27] and Bayesian (belief) networks [28].

A *factor graph* is a bipartite graph that expresses the relationships between the random variables $X_i, i \in \mathcal{I}$ and the factors $f_a, a \in \mathcal{A}$ in the factorization (2.1). A factor graph has a variable node for each variable X_i, typically depicted by a circle, a factor node for each factor f_a, commonly drawn as a square, and an edge – connecting variable node X_i to factor node f_a, if, and only if, the variable x_i is an argument of the factor f_a.

We give an example of a joint pdf and its factor graph next.

Example 1 (Factor Graph Representation). *Let $p_{\boldsymbol{X}}(x_1, x_2, x_3, x_4, x_5)$ be the joint pdf of five random variables X_1, X_2, X_3, X_4, X_5 and suppose that $p_{\boldsymbol{X}}$ can be factorized as*

$$p_{\boldsymbol{X}}(x_1, x_2, x_3, x_4, x_5) = \frac{1}{Z} f_A(x_1) f_B(x_2) f_C(x_1, x_2, x_3) f_D(x_3, x_4) f_E(x_3, x_5). \tag{2.2}$$

The factor graph representing the factorization (2.2) *is depicted in Fig. 2.1.*

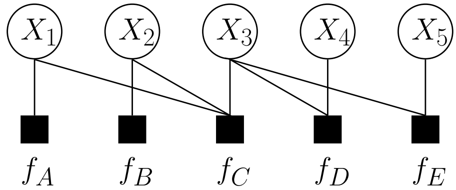


Fig. 2.1: A factor graph representing the factorization of the joint pdf (2.2)

For convenience of description, we use $N(i) \subseteq \mathcal{A}$ to denote the set of indices of factors connected to the variable node X_i by an edge, and $N(a) \subseteq \mathcal{I}$ stands for the set of indices of variables nodes connected to the factor node f_a.

1.3 Optimal Estimators and Exact Inference

Given the probabilistic system described in Subsection 1.1, a common task is the estimation of the random variables X_i, $i \in \mathcal{I}$ or a selection of them given the observation $\boldsymbol{y}$. In Bayesian theory, estimators can be formulated based on different optimality criteria. Two of the most widely used estimators are the MMSE and the MAP estimators [1, 5]. The former minimizes the mean square error (MSE), while the latter maximizes the probability that the estimate is correct. Specifically, the MMSE and MAP estimators of $X_i, i \in \mathcal{I}$, are formulated as

$$\hat{x}_i(\boldsymbol{y})_{\text{MMSE}} \triangleq \arg\min_{\hat{x}_i} \mathbb{E}[\|X_i - \hat{x}_i\|^2|\boldsymbol{y}] = \int_{\mathcal{X}_i} x_i p_{X_i|\boldsymbol{Y}}(x_i|\boldsymbol{y}) dx_i \quad (2.3)$$

$$\hat{x}_i(\boldsymbol{y})_{\text{MAP}} \triangleq \operatorname*{argmax}_{\hat{x}_i} \Pr(X_i = \hat{x}_i|\boldsymbol{y}) = \operatorname*{argmax}_{x_i \in \mathcal{X}_i} p_{X_i|\boldsymbol{Y}}(x_i|\boldsymbol{y}). \quad (2.4)$$

From these expressions, we can see that both estimators require the knowledge of the posterior pdf $p_{X_i|\boldsymbol{Y}}(x_i|\boldsymbol{y})$ of the unknown variable X_i. This posterior pdf can be obtained by marginalizing the joint posterior $p_{\boldsymbol{X}|\boldsymbol{Y}}(\boldsymbol{x}|\boldsymbol{y})$ with respect to all unknown variables but X_i:

$$p_{X_i|\boldsymbol{Y}}(x_i|\boldsymbol{y}) = \int_{\boldsymbol{X}_{\bar{i}}} p_{\boldsymbol{X}|\boldsymbol{Y}}(\boldsymbol{x}|\boldsymbol{y}) d\boldsymbol{x}_{\bar{i}} \quad (2.5)$$

where $\boldsymbol{X}_{\bar{i}}$ and $\boldsymbol{x}_{\bar{i}}$ represent the vectors $\boldsymbol{X}$ and $\boldsymbol{x}$ with their i-th entry removed.

The methods that compute (2.5) perform ***exact inference*** [1, 3, 4]. When exact inference is feasible, the optimal estimators in (2.3) and (2.4) can be computed exactly. Unfortunately, this is seldom the case for large probabilistic systems, i.e. M and N large, as those representing communication systems. For those systems, one needs to resort to some approximation of $p_{X_i|\boldsymbol{Y}}(x_i|\boldsymbol{y})$ instead.

Remark: When $p_{X_i|\boldsymbol{Y}}(x_i|\boldsymbol{y})$ is a Gaussian distribution, its mode coincides with its mean. Thus, the MAP estimate coincides with the MMSE estimate in this case.

2 Principle of Variational Inference

As mentioned above, exact inference in large probabilistic systems is often intractable, and approximate solutions need to be sought instead. Variational approximation methods [9, 10, 20] compute approximations of the marginal pdfs of variables of interest from a joint pdf. These methods attempt to minimize some objective function under some specific constraints. In this section, we introduce the objective functions associated with the methods that we will consider in the thesis, i.e. the variational free energy and its region-based approximations. In the next chapter we will derive the update equations of these methods that result from the stationary point equations of the objective functions subject to the specified constraints.

2.1 Variational Free Energy

Assume that $p_{\boldsymbol{X}}(\boldsymbol{x}) = \frac{1}{Z}\prod_{a\in\mathcal{A}} f_a(\boldsymbol{x}_a)$ is the pdf of a large probabilistic system. The exact computation of marginals from $p_{\boldsymbol{X}}(\boldsymbol{x})$ is often difficult. An avenue to compute approximations of the sought marginals consists of postulating a simple trial function $b(\boldsymbol{x})$, often called "belief". The belief $b(\boldsymbol{x})$ is used in place of the true system's pdf and, as such, should be normalized and obey $0 \leq b(\boldsymbol{x}) \leq 1$ for all $\boldsymbol{x}$. The variational free energy (also called Gibbs free energy) associated to $b(\boldsymbol{x})$ [10] is defined as

$$\mathcal{F}(b) = U(b) - H(b) \tag{2.6}$$

where $U(b) = -\sum_{\boldsymbol{x}} b(\boldsymbol{x}) \sum_{a\in\mathcal{A}} \ln f_a(\boldsymbol{x}_a)$ is called the variational average energy and $H(b) = -\sum_{\boldsymbol{x}} b(\boldsymbol{x}) \ln b(\boldsymbol{x})$ is the variational entropy.

From the above definitions of variational free energy $\mathcal{F}(b)$, variational average energy $U(b)$, and variational entropy $H(b)$, we readily obtain the identity

$$\mathcal{F}(b) = \mathcal{F}_H + \mathrm{KLD}(b\|p_{\boldsymbol{X}}) \tag{2.7}$$

where $\mathcal{F}_H = -\ln Z$ is a constant –called the Helmholtz free energy of the system in statistical mechanics– and

$$\mathrm{KLD}(b\|p_{\boldsymbol{X}}) \triangleq \sum_{\boldsymbol{x}} b(\boldsymbol{x}) \ln \frac{b(\boldsymbol{x})}{p_{\boldsymbol{X}}(\boldsymbol{x})}$$

is the Kullback-Leibler (KL) divergence of $p_{\boldsymbol{X}}(\boldsymbol{x})$ from $b(\boldsymbol{x})$. Since the KL divergence is non-negative, we have the inequality $\mathcal{F}(b) \geq \mathcal{F}_H$, with equality if, and only if, $b(\boldsymbol{x}) = p_{\boldsymbol{X}}(\boldsymbol{x})$. From this result, we clearly see that minimizing the variational free energy $\mathcal{F}(b)$ in (2.7) with respect to the belief $b(\boldsymbol{x})$ is equivalent to minimizing the KL divergence of $p_{\boldsymbol{X}}(\boldsymbol{x})$ from $b(\boldsymbol{x})$. Thus, if there are no restrictions on the trial pdf $b(\boldsymbol{x})$, one obtains the exact $\mathcal{F}_H$ and recover $p_{\boldsymbol{X}}(\boldsymbol{x})$ by computing the minimizer of $\mathcal{F}(b)$.

Instead, one can impose some factorization constraint on the belief function $b(\boldsymbol{x})$ so that the computation of marginals from it becomes tractable. In this case, the minimizer of the free energy is the belief fulfilling the constraint that minimizes the KL divergence. One simplifying assumption commonly employed in variational inference is that the belief function is of the factorized form

$$b_{\mathrm{MF}}(\boldsymbol{x}) \quad = \quad \prod_{i \in \mathcal{I}} b_i(x_i). \tag{2.8}$$

Substituting (2.8) into (2.6) yields the MF free energy

$$\mathcal{F}_{\mathrm{MF}} = \sum_{i \in \mathcal{I}} \sum_{x_i} b_i(x_i) \ln b_i(x_i) - \sum_{a \in \mathcal{A}} \sum_{\boldsymbol{x}_a} \prod_{i \in N(a)} b_i(x_i) \ln f_a(\boldsymbol{x}_a). \tag{2.9}$$

Note that each factor $b_i(x_i)$ satisfies the normalization constraint $\int b_i(x_i) dx_i = 1$.

The belief function (of the form (2.8)) that minimizes (2.9) is called the MF approximation.

This approach assumes that the marginal beliefs of the variables $X_i, i \in \mathcal{I}$ are independent. Hence, the minimizer of the MF free energy is already a product of the sought approximate marginals, and no further operations are needed to compute them.

2.2 Region-Based Free Energy

While the variational free energy (or KL divergence) is often a useful objective function for approximate inference, there are systems in which other objective functions may be preferred. Here we introduce an approximation of the Gibbs free energy, called region-based free energy [10, Sec. IV].

The region-based approximation method is based on the principle of dividing the factor graph of the probabilistic system into different regions. A region $R \triangleq (\mathcal{I}_R, \mathcal{A}_R)$ consists of subsets of indices $\mathcal{I}_R \subseteq \mathcal{I}$ and $\mathcal{A}_R \subseteq \mathcal{A}$ with the restriction that $a \in \mathcal{A}$ implies that $N(a) \subseteq \mathcal{I}_R$. There is also a counting number $c_R \in \mathbb{Z}$ associated to each region R. A set $\mathcal{R} \triangleq \{(R, c_R)\}$ of regions and associated counting numbers is called valid if, and only if,

$$\sum_{(R,c_R) \in \mathcal{R}} c_R I_{\mathcal{A}_R}(a) = \sum_{(R,c_R) \in \mathcal{R}} c_R I_{\mathcal{I}_R}(i) = 1 \tag{2.10}$$

for all $a \in \mathcal{A}$ and $i \in \mathcal{I}$, where $I_{\mathcal{S}}(x)$ is the set-membership indicator function, equal to 1 if $x \in \mathcal{S}$ and equal to 0 otherwise.

For a valid set $\mathcal{R}$ of regions and associated counting numbers, the region-based free energy approximation is defined as [10]

$$\mathcal{F}_{\mathcal{R}}(\{b_R\}) \triangleq U_{\mathcal{R}}(\{b_R\}) - H_{\mathcal{R}}(\{b_R\}), \tag{2.11}$$

where

$$\begin{aligned} U_{\mathcal{R}}(\{b_R\}) &\triangleq -\sum_{(R,c_R)\in\mathcal{R}} c_R \int b_R(\boldsymbol{x}_R) \sum_{a\in\mathcal{A}_R} \ln f_a(\boldsymbol{x}_a) d\boldsymbol{x}_R \\ H_{\mathcal{R}}(\{b_R\}) &\triangleq -\sum_{(R,c_R)\in\mathcal{R}} c_R \int b_R(\boldsymbol{x}_R) \ln b_R(\boldsymbol{x}_R) d\boldsymbol{x}_R. \end{aligned}$$

A typical example of a valid region-based approximation is the one considered in Bethe's method used to derive BP and EP. Its regions are defined as follows:

1) small regions $R_i \triangleq (\{i\}, \emptyset)$, with $c_i = c_{R_i} = 1 - |N(i)|$ for all $i \in \mathcal{I}$;

2) large regions $R_a \triangleq (N(a), \{a\})$, with $c_a = c_{R_a} = 1$ for all $a \in \mathcal{A}$.

The above regions assignment fulfils (2.10). This yields a valid set of regions and associated counting numbers:

$$\mathcal{R}_{\text{Bethe}} \triangleq \{(R_i, c_i); i \in \mathcal{I}\} \cup \{(R_a, c_a); a \in \mathcal{A}\}. \tag{2.12}$$

With this selection, (2.11) becomes

$$\begin{aligned} \mathcal{F}_{\text{Bethe}} &= \sum_{a\in\mathcal{A}} \int b_a(\boldsymbol{x}_a)) \ln \frac{b_a(\boldsymbol{x}_a)}{f_a(\boldsymbol{x}_a)} d\boldsymbol{x}_a \\ &\quad - \sum_{i\in\mathcal{I}} (|N(i)| - 1) \int b_i(x_i) \ln b_i(x_i) dx_i. \end{aligned} \tag{2.13}$$

This approximation of the variational free energy is commonly called the Bethe free energy [10, 19]. The beliefs of the regions in $\mathcal{R}_{\text{Bethe}}$ are non-negative and are normalized, i.e.

$$\begin{aligned} \int b_a(\boldsymbol{x}_a) d\boldsymbol{x}_a &= 1, \quad \forall a \in \mathcal{A} \\ \int b_i(x_i) dx_i &= 1, \quad \forall a \in \mathcal{I}. \end{aligned} \tag{2.14}$$

One can easily show that if the factor graph is a tree, then $\mathcal{F}_{\text{Bethe}} = \mathcal{F}(b)$ with

$$b(\boldsymbol{x}) = \frac{\prod_{a\in\mathcal{A}} b_a(\boldsymbol{x}_a)}{\prod_{i\in\mathcal{I}} b_i(x_i)^{c_i - 1}},$$

i.e. the Bethe free energy coincides with the Gibbs free energy of $b(\boldsymbol{x})$.

In this thesis a ***variational inference method*** is any method that computes the minimizer of the MF free energy or a region-based free energy or an approximation of this minimizer.

Chapter 3

Variational Inference Via Message Passing Algorithms

In this chapter, we present the variational inference tools that we will use in the rest of this thesis. Actually, we present their implementation as message passing algorithms. Messages passing algorithms are iterative. At each iteration, functions that are interpreted as messages passed along edges of the factor graph representing the underlying probabilistic model are computed. Update rules determine how these messages are computed from the messages obtained at the previous iteration. The message passing algorithms considered in the thesis attempt to compute beliefs that minimize a given free energy while fulfilling some constraints (e.g. normalization and marginalization or moment matching constraints). These algorithms are obtained by using the following procedure:

Procedure 1 (Derivation of message passing algorithms). *In a first step the stationary point equations of the Lagrangian function obtained by adding terms that account for the beliefs' constraints to the considered free energy are obtained. In a second step these stationary point equations and the matching constraints are reformulated as implicit equations involving messages. These implicit equations are the fixed point equations of the update rules, or–put differently–the update equations are selected so that their fixed point are solutions to the implicit equations.*

Below we briefly discuss the three message passing algorithms that will be the baseline of all message passing algorithms considered in the thesis: BP, EP and MF.

BP, also known as sum-product algorithm [17], has been widely used in the design of wireless receivers. It is shown in [10] that the fixed point equations of BP are obtained by applying **Procedure 1** while considering the Bethe free energy, normalization constraints, and marginalization constraints that relate the beliefs of the factor nodes and the beliefs of the variable nodes. BP computes exact marginals when the factor graph is free of cycles. Its remarkable performance, especially when applied to linear Gaussian or discrete probabilistic models and the factor graph has no short cycles, justifies its popularity. However, the complexity of BP may become intractable in certain application contexts, e.g. when the probabilistic model includes both discrete and continuous random variables. In this case, one must resort to alternative methods.

The MF approximation [29] have been initially used in quantum and statistical physics. The fixed point equations of MF are obtained by applying **Procedure 1** while considering the Gibbs free energy, some constraint on how $b(\boldsymbol{x})$ factorizes, and the fact that the integral of this belief is equal to one. The MF approximation has also been formulated as a message passing algorithm, referred to as variational message passing (VMP) [9, 21, 22]. It has simple update rules, in particular for conjugate-exponential models and it always converges. However, MF is not compatible with hard constraints.

EP [18, 19] can be seen as a relaxed version of BP: The same procedure is used except the marginalization constraints that are replaced by looser constraints requiring some moments of the beliefs of the factor nodes and the beliefs of the variable nodes to match. First- and second-moments are typically considered. EP is typically regarded as an approximation of BP, in which the beliefs of the variable nodes are approximated by pdfs belonging to specific exponential families (e.g. Gaussian). The approximate beliefs are obtained so that the expectations of their natural statistics coincide with those of the non approximated BP beliefs. In this way, EP solutions can circumvent the high complexity of BP-based algorithms in certain applications.

Recently, a unified message passing framework which combines BP with MF [23] on a same factor graph has been proposed, which keeps the virtues of BP and MF but avoids their respective drawbacks. Following an analogous methodology, we formulate another unified framework which combines BP and EP [24] (**Paper A**) on a single factor graph in this chapter.

Notation: In the sequel, we define an m-type message $m_{f_a \to x_i}(x_i)$ and an n-type message $n_{x_i \to f_a}(x_i)$ to be a message from factor node f_a to variable node x_i and a message from variable node x_i to factor node f_a, respectively. We use this default definition in the rest of thesis, unless otherwise specified.

1 Mean Field Message Passing

A message passing representation of the MF approximation was derived in [20] using Markov random fields, in [21] using Bayesian networks, and in [9, 22] on factor graphs.

Considering the MF free energy (2.9) and the normalization constraints $\int b_i(x_i)dx_i = 1, \forall i \in \mathcal{I}$, we use the method described in **Procedure 1** to derive the fixed point equations of MF:

$$\begin{aligned} m_{f_a \to x_i}(x_i) &= \exp\left\{ \int \log f_a(\boldsymbol{x}_a) \prod_{j \in N(a) \setminus i} n_{x_j \to f_a}(x_j) dx_j \right\}, \\ & \qquad\qquad\qquad\qquad a \in \mathcal{A}, i \in N(a) \qquad (3.1) \\ n_{x_i \to f_a}(x_i) &= \prod_{b \in N(i)} m_{f_b \to x_i}(x_i), \qquad i \in \mathcal{I}. \qquad (3.2) \end{aligned}$$

The beliefs of the variable nodes are obtained as

$$b_i(x_i) \propto \sum_{a \in N(i)} m_{f_a \to x_i}(x_i), \qquad i \in \mathcal{I}. \tag{3.3}$$

The update equations of MF result by interpreting the "=" and "∝" signs in (3.1) - (3.3) as the assignment operator (":=").

2 Belief Propagation

BP [17], also called sum-product algorithm, computes the exact marginal distribution $p_i(x_i)$ of the variable x_i when the factor graph is a tree. In [10, Sec. VI] the fixed point equations of BP are obtained by using the method described in **Procedure 1** while considering the Bethe free energy, the normalization constraints (2.14), and the following marginalization constraints.

The beliefs of the small regions $R_i, i \in \mathcal{I}$, and their neighbouring large regions $R_a, a \in \mathcal{A}$, are related via the marginalization constraints

$$b_i(x_i) = \int b_a(\boldsymbol{x}_a) d\boldsymbol{x}_a \backslash x_i, \quad i \in \mathcal{I}, a \in N(i). \tag{3.4}$$

The resulting fixed point equations of BP read

$$\begin{aligned} m_{f_a \to x_i}(x_i) &= \int f_a(\boldsymbol{x}_a) \prod_{j \in N(a) \setminus i} n_{x_j \to f_a}(x_j) dx_j, \quad a \in \mathcal{A}, i \in N(a) \qquad (3.5) \\ n_{x_i \to f_a}(x_i) &= \prod_{b \in N(i) \setminus a} m_{f_b \to x_i}(x_i), \qquad i \in \mathcal{I}, a \in N(i). \qquad (3.6) \end{aligned}$$

The beliefs of the variable nodes are obtained as

$$b_i(x_i) \propto \prod_{b \in N(i)} m_{f_b \to x_i}(x_i), \qquad i \in \mathcal{I}. \tag{3.7}$$

Interpreting the "=" and "∝" signs in (3.5) - (3.7) as the assignment operator yields the update equations of BP.

3 Expectation Propagation

EP [18, 19] constrains the beliefs $b_i(x_i), x_i \in \mathcal{X}_i, i \in \mathcal{I}$, to belong to special exponential families $\mathcal{E}_i$, yielding lower complexity than BP. Heskes et. al. [19] derived the fixed point equations of EP by applying **Procedure 1** while considering the Bethe free energy (2.13), the normalization constraints (2.14), and the following expectation constraints.

The expectations of the sufficient statistics $\boldsymbol{\phi}_i(x_i)$ over the beliefs of the small regions R_i and their neighboring large regions $R_a, a \in N(i)$, match. Specifically:

$$\int \boldsymbol{\phi}_i(x_i) b_i(x_i) dx_i = \int \boldsymbol{\phi}_i(x_i) b_a(\boldsymbol{x}_a) d\boldsymbol{x}_a, \quad i \in \mathcal{I}, a \in N(i). \tag{3.8}$$

Following **Procedure 1**, we obtain the stationary point equations for the beliefs:

$$\begin{aligned} b_i(x_i) &\propto \exp\left\{ \frac{1}{|N(i)|-1} \sum_{a \in N(i)} \boldsymbol{\mu}_{a,i}^{\mathrm{T}} \boldsymbol{\phi}_i(x_i) \right\}, i \in \mathcal{I} & (3.9) \\ b_a(\boldsymbol{x}_a) &\propto f_a(\boldsymbol{x}_a) \prod_{i \in N(a)} \exp\left\{ \boldsymbol{\mu}_{a,i}^{\mathrm{T}} \boldsymbol{\phi}_i(x_i) \right\}, \quad a \in \mathcal{A}, & (3.10) \end{aligned}$$

where $\boldsymbol{\mu}_{a,i}^{\mathrm{T}}$ is the Lagrangian multiplier associated with the expectation constraints (3.8).

From these equations and the moment constraints we obtain the fixed point equations of EP messages

$$\begin{aligned} m_{f_a \to x_i}(x_i) &= \frac{\mathrm{Proj}_{\mathcal{E}_i}\left\{ \int f_a(\boldsymbol{x}_a) \prod_{j \in N(a) \setminus i} n_{x_j \to f_a}(x_j) dx_j n_{x_i \to f_a}(x_i) \right\}}{n_{x_i \to f_a}(x_i)}, \\ &\qquad a \in \mathcal{A}, i \in N(a) & (3.11) \\ n_{x_i \to f_a}(x_i) &= \prod_{b \in N(i) \setminus a} m_{f_b \to x_i}(x_i), \quad i \in \mathcal{I}, \ a \in N(i). & (3.12) \end{aligned}$$

In (3.11), $\mathrm{Proj}_{\mathcal{E}}\{\cdot\}$ is the projection operator on the class of exponential pdfs $\mathcal{E}$:

$$\begin{aligned} Q(x) &= \mathrm{Proj}_{\mathcal{E}}\{f(x)\} \\ &\triangleq \arg\min_{Q'(x) \in \mathcal{E}} \mathrm{KLD}\{Q'(x) || f(x)\} \\ &= \arg\min_{Q'(x) \in \mathcal{E}} \int Q'(x) \log \frac{Q'(x)}{f(x)} dx. \end{aligned}$$

The projection $Q(x) = \arg\min_{Q'(x)\in\mathcal{E}} \mathrm{KLD}\{Q'(x)||f(x)\}$ is the element in $Q'(x) \in \mathcal{E}$ such that the expectations of $\boldsymbol{\phi}(x)$ over $Q(x)$ and $f(x)$ are equal:

$$\int \boldsymbol{\phi}(x)Q(x)dx = \int \boldsymbol{\phi}(x)f(x)dx.$$

The beliefs of the variable nodes are given by

$$b_i(x_i) \propto \prod_{b\in N(i)} m_{f_b\to x_i}(x_i), \qquad i \in \mathcal{I}. \tag{3.13}$$

It is easily seen that due to (3.11) and (3.12) the belief $b_i(x_i)$ in (3.13) belongs to the exponential family $\mathcal{E}_i$, and fulfills (3.9).

The update equations of EP are obtained by interpreting the "=" and "$\propto$" signs in (3.11) - (3.13) as the assignment operator.

Notice that in EP the variable (small region) beliefs match the factor (large-region) beliefs in terms of their moments, according to (3.8). In BP the former beliefs match the latter in the sense that the former are marginals of the latter, as expressed by (3.4). This is the distinction between BP and EP. The different types of constraints imposed on the minimization of the Bethe free energy –(3.8) for BP and (3.4) for EP– constitute the only difference in the derivation of the two message passing algorithms.

4 Combined Message Passing Frameworks

In this section, we present two message passing methods each of which combines two of the previously discussed methods: BP-MF and BP-EP. In Subsection 4.1 we present the derivation of combined BP-MF [23], which we later apply to frequency domain turbo equalization in **Paper C** and sparse Bayesian learning in **Paper D**. In Subsection 4.2 we derive BP-EP by minimizing the Bethe free energy. We apply this algorithm in **Paper A** for turbo equalization. Other unified message passing frameworks combining more than two techniques, such as combined BP-EP-MF, can be deduced straightforwardly.

4.1 BP-MF [23]

We first group the local functions $f_a(\boldsymbol{x}_a), a \in \mathcal{A}$ in factorization (2.1) into two disjoint sets, a BP set $\{f_a(\boldsymbol{x}_a); a \in \mathcal{A}_{\mathrm{BP}}\}$ and a MF set $\{f_a(\boldsymbol{x}_a); a \in \mathcal{A}_{\mathrm{MF}}\}$. Here, $\mathcal{A}_{\mathrm{BP}}$ and $\mathcal{A}_{\mathrm{MF}}$ stand for the index sets of the factors in the BP and MF sets, respectively. Furthermore, we set

$$\mathcal{I}_{\mathrm{MF}} \triangleq \bigcup_{a\in\mathcal{A}_{\mathrm{MF}}} N(a), \qquad \mathcal{I}_{\mathrm{BP}} \triangleq \bigcup_{a\in\mathcal{A}_{\mathrm{BP}}} N(a)$$

and

$$N_{\text{MF}}(i) \triangleq \mathcal{A}_{\text{MF}} \cap N(i), \qquad N_{\text{BP}}(i) \triangleq \mathcal{A}_{\text{BP}} \cap N(i).$$

Next, we define the following regions and counting numbers:

1) one MF region $R_{\text{MF}} \triangleq (\mathcal{I}_{\text{MF}}, \mathcal{A}_{\text{MF}})$, with $c_{R_{\text{MF}}} = 1$;

2) $|\mathcal{I}_{\text{BP}}|$ small regions $R_i \triangleq (\{i\}, \emptyset)$, with $c_{R_i} = 1 - |N_{\text{BP}}(i)| - I_{\mathcal{I}_{\text{MF}}}(i)$ for all $i \in \mathcal{I}_{\text{BP}}$;

3) $|\mathcal{A}_{\text{BP}}|$ large regions $R_a \triangleq (N(a), \{a\})$, with $c_{R_a} = 1$ for all $a \in \mathcal{A}_{\text{BP}}$.

This yields the valid set of regions and associated counting numbers

$$\mathcal{R}_{\text{BP,MF}} \triangleq \{(R_i, c_{R_i}); i \in \mathcal{I}_{\text{BP}}\} \cup \{R_a, c_{R_a}; a \in \mathcal{A}_{\text{BP}}\} \cup \{R_{\text{MF}}, c_{R_{\text{MF}}}\}$$

and its region-based free energy

$$\begin{aligned} \mathcal{F}_{\text{BP,MF}} \quad = \quad & \sum_{a \in \mathcal{A}_{\text{BP}}} \int b_a(\boldsymbol{x}_a) \ln \frac{b_a(\boldsymbol{x}_a)}{f_a(\boldsymbol{x}_a)} d\boldsymbol{x}_a \\ & - \sum_{a \in \mathcal{A}_{\text{MF}}} \int \prod_{i \in N(a)} b_i(x_i) \ln f_a(\boldsymbol{x}_a) d\boldsymbol{x}_a \\ & - \sum_{i \in \mathcal{I}} (|N_{\text{BP}}(i) - 1|) \int b_i(x_i) \ln b_i(x_i) dx_i. \end{aligned} \tag{3.14}$$

Normalization constraints are given by

$$\begin{aligned} \int b_i(x_i) dx_i \quad &= \quad 1, \qquad i \in \mathcal{I}_{\text{MF}} / \mathcal{I}_{\text{BP}}, \\ \int b_a(\boldsymbol{x}_a) d\boldsymbol{x}_a \quad &= \quad 1, \qquad a \in \mathcal{A}_{\text{BP}}, \end{aligned} \tag{3.15}$$

and marginalization constraints for BP regions are

$$b_i(x_i) \quad = \quad \int b_a(\boldsymbol{x}_a) d\boldsymbol{x}_a / x_i, \quad a \in \mathcal{A}_{\text{BP}}, i \in N(a). \tag{3.16}$$

Once again, **Procedure 1** is invoked using the region-based free energy in (3.14), the normalization constraints in (3.15), and the marginalization constraints in (3.16), to obtain the fixed point equations of BP-MF:

$$m^{\text{BP}}_{f_a \to x_i}(x_i) = \int f_a(\boldsymbol{x}_a) \prod_{j \in N(a) \setminus i} n_{x_j \to f_a}(x_j) dx_j, \quad a \in \mathcal{A}_{\text{BP}}, i \in N(a) \tag{3.17}$$

$$m^{\text{MF}}_{f_a \to x_i}(x_i) = \exp \left\{ \int \log f_a(\boldsymbol{x}_a) \prod_{j \in N(a) \setminus i} n_{x_j \to f_a}(x_j) dx_j \right\}, \; a \in \mathcal{A}_{\text{MF}}, i \in N(a) \tag{3.18}$$

$$n_{x_i \to f_a}(x_i) = \prod_{b \in N(i) \cap \mathcal{A}_{\text{BP}} \setminus a} m^{\text{BP}}_{f_b \to x_i}(x_i) \prod_{c \in N(i) \cap \mathcal{A}_{\text{MF}}} m^{\text{MF}}_{f_c \to x_i}(x_i), i \in \mathcal{I}, a \in N(i). \tag{3.19}$$

The variables' beliefs are given by

$$b_i(x_i) \quad \propto \quad \prod_{a \in N(i) \bigcup \mathcal{A}_{\mathrm{BP}}} m^{\mathrm{BP}}_{f_a \to x_i}(x_i) \prod_{b \in N(i) \bigcup \mathcal{A}_{\mathrm{MF}}} m^{\mathrm{MF}}_{f_b \to x_i}(x_i). \qquad (3.20)$$

Interpreting the "=" and "$\propto$" signs in (3.17) - (3.20) as the assignment operator results in the update equations of BP-MF.

4.2 BP-EP

The variable nodes x_i, $i \in \mathcal{I}$ of a given factor graph are grouped into two disjoint BP and EP sets. The sets $\mathcal{I}_{\mathrm{BP}}$ and $\mathcal{I}_{\mathrm{EP}}$ denote the indices sets of the variables in the BP and EP sets, respectively, with

$$\mathcal{I} = \mathcal{I}_{\mathrm{BP}} \cup \mathcal{I}_{\mathrm{EP}}, \quad \mathcal{I}_{\mathrm{BP}} \cap \mathcal{I}_{\mathrm{EP}} = \emptyset. \qquad (3.21)$$

We impose the normalization constraints (2.14) for every factor node and variable node, the marginalization constraints (3.4) for the variables in the BP set and their neighbouring factors, and the expectation constraints (3.8) for the variables in the EP set and their neighbouring factors.

The Lagrangian function of the Bethe free energy in (2.13) that accounts for all the above constraints is of the form

$$\begin{aligned} \mathcal{L}_{\mathrm{Bethe}} \quad = \quad & \mathcal{F}_{\mathrm{Bethe}} - \sum_{a \in \mathcal{A}} \sum_{i \in N(a) \cap \mathcal{I}_{\mathrm{BP}}} \int \lambda_{a,i}(x_i) \left(b(x_i) - \int b_a(\boldsymbol{x}_a) d\boldsymbol{x}_a / x_i \right) dx_i \\ & - \sum_{a \in \mathcal{A}} \sum_{i \in N(a) \cap \mathcal{I}_{\mathrm{EP}}} \boldsymbol{\mu}^{\mathrm{T}}_{a,i} \left(\int \boldsymbol{\phi}_i(x_i) b(x_i) dx_i - \int \boldsymbol{\phi}_i(x_i) b_a(\boldsymbol{x}_a) d\boldsymbol{x}_a \right) \\ & - \sum_{i \in \mathcal{I}} \gamma_i \left(\int b_i(x_i) dx_i - 1 \right) - \sum_{a \in \mathcal{A}} \gamma_a \left(\int b_a(\boldsymbol{x}_a) d\boldsymbol{x}_a - 1 \right), \qquad (3.22) \end{aligned}$$

where $\lambda_{a,i}$, $\boldsymbol{\mu}_{a,i}$, γ_i and γ_a, $i \in \mathcal{I}, a \in \mathcal{A}$, denote Lagrange multipliers.

Taking the derivatives of $\mathcal{L}_{\mathrm{Bethe}}$ in (3.22) with respect to $b_i(x_i), i \in \mathcal{I}$ and $b_a(\boldsymbol{x}_a), a \in \mathcal{A}$, respectively, and equating the derivatives to zero, yields the stationary point equations

$$(|N(i)| - 1) \ln b_i(x_i) = \sum_{a \in N(i)} \lambda_{a,i}(x_i) + \gamma_i - |N(i)| + 1, \quad i \in \mathcal{I}_{\mathrm{BP}} \qquad (3.23)$$

$$(|N(i)| - 1) \ln b_i(x_i) = \sum_{a \in N(i)} \boldsymbol{\mu}^{\mathrm{T}}_{a,i} \boldsymbol{\phi}_i(x_i) + \gamma_i - |N(i)| + 1, \; i \in \mathcal{I}_{\mathrm{EP}} \qquad (3.24)$$

$$\begin{aligned} \ln b_a(\boldsymbol{x}_a) = \ln f_a(\boldsymbol{x}_a) + & \sum_{i \in N(a) \cap \mathcal{I}_{\mathrm{BP}}} \lambda_{a,i}(x_i) \\ & + \sum_{i \in N(a) \cap \mathcal{I}_{\mathrm{EP}}} \boldsymbol{\mu}^{\mathrm{T}}_{a,i} \boldsymbol{\phi}_i(x_i) + \gamma_a - 1, \quad a \in \mathcal{A}. \qquad (3.25) \end{aligned}$$

From these equations and the marginalization and expectation constraints we obtain the fixed point equations of BP-EP:

$$m^{\mathrm{BP}}_{f_a \to x_i}(x_i) = \int f_a(\boldsymbol{x}_a) \prod_{j \in N(a)/i} n_{z_j \to f_a}(x_j)\, dx_j, \quad i \in \mathcal{I}, a \in N(i) \tag{3.26}$$

$$m^{\mathrm{EP}}_{f_a \to x_i}(x_i) = \frac{\mathrm{Proj}_{\mathcal{E}_i}\left\{ m^{\mathrm{BP}}_{f_a \to x_i}(x_i)\, n_{x_i \to f_a}(x_i) \right\}}{n_{x_i \to f_a}(x_i)}, \quad i \in \mathcal{I}_{\mathrm{EP}}, a \in N(i) \tag{3.27}$$

$$n_{x_i \to f_a}(x_i) = \prod_{b \in N(i)/a} m^{\mathrm{BP}}_{f_b \to x_i}(x_i), \quad i \in \mathcal{I}_{\mathrm{BP}}, a \in N(i) \tag{3.28}$$

$$n_{x_i \to f_a}(x_i) = \prod_{b \in N(i)/a} m^{\mathrm{EP}}_{f_b \to x_i}(x_i), \quad i \in \mathcal{I}_{\mathrm{EP}}, a \in N(i). \tag{3.29}$$

The beliefs of the variable nodes are given by

$$b_i(x_i) \propto \prod_{a \in N(i)} m^{\mathrm{BP}}_{f_a \to x_i}(x_i), \quad i \in \mathcal{I}_{\mathrm{BP}} \tag{3.30}$$

$$b_i(x_i) \propto \prod_{a \in N(i)} m^{\mathrm{EP}}_{f_a \to x_i}(x_i), \quad i \in \mathcal{I}_{\mathrm{EP}}. \tag{3.31}$$

The update equations of BP-EP are obtained by interpreting the "$=$" and "$\propto$" signs in (3.26) - (3.31) as the assignment operator.

5 Gaussian Approximate Message Techniques

BP, EP and MF can sometimes still result in high complexity as some of the messages may involve cumbersome expressions which, in turn, make their computation intractable. Gaussian messages are parametric functions of their mean and variance. They are particularly well suited when these parameters can be easily computed. Hence, several techniques to approximate messages by Gaussian functions have been proposed in the literature. Here, we shortly discuss two of them. The first is based on minimizing a KL divergence and the second relies on a second-order Taylor's expansion. In some applications, having the ability to select some messages that are to be approximated by Gaussian functions allows for a flexible trade-off between complexity and performance. The partial Gaussian approximation (PGA) principle [30] is a good example of this strategy. We have applied it to design an efficient turbo equalization algorithm in **Paper B**.

5.1 Minimizing Kullback-Leibler Divergence

A message $m(x)$ can be approximated by a Gaussian function $m^{\mathrm{G}}(x)$ as follows:

$$m^{\mathrm{G}}(x) = \arg\min_{m'(x)\in\mathcal{N}} \mathrm{KLD}\{m'(x)\|m(x)\} \approx m(x). \tag{3.32}$$

where $\mathcal{N}$ denotes the Gaussian family.

The method matches the first and second moments of $m(x)$ and $m^{\mathrm{G}}(x)$, i.e. $\mathbb{E}[X]_{m(x)} = \mathbb{E}[X]_{m^{\mathrm{G}}(x)}$ and $\mathbb{E}[X^2]_{m(x)} = \mathbb{E}[X^2]_{m^{\mathrm{G}}(x)}$. When one is not able to obtain $\mathbb{E}[X]_{m(x)}$ and $\mathbb{E}[X^2]_{m(x)}$ in closed form, the minimization in (3.32) can be performed by using numerical methods, such as gradient descent method, steepest descent method and Newton's method [31].

5.2 Second Order Taylor's Expansion

Suppose that a message $m(x)$ can be expressed in the exponential form $m(x) = e^{-f(x)}$, where $f(x)$ has first and second derivatives. We can expand $f(x)$ in a second-order Taylor's series at a given point x^*,

$$f(x) \approx f(x^*) + f'(x^*)(x - x^*) + f''(x^*)(x - x^*)^2. \tag{3.33}$$

Using this result, a Gaussian message approximating $m(x)$ can be obtained as

$$m^{\mathrm{G}}(x) \propto \exp\{-f'(x^*)(x - x^*) - f''(x^*)(x - x^*)^2\} \approx m(x). \tag{3.34}$$

This method is one of the main contributions of **Paper F**.

6 Other Approximate Message Passing Algorithms

In our work we have focused on the message passing methods and the approximation techniques discussed in the previous sections of this chapter. There are, however, other alternative methods that can be used for the design of iterative signal processing algorithms. We close this chapter by shortly mentioning two methods that have attracted considerable interest from the research community in recent years: approximate message passing (AMP) [32], and its more general form, generalized AMP (GAMP) [33].

The AMP algorithm was proposed as a computationally efficient and accurate tool to perform inference on signal models of the form

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w} \tag{3.35}$$

where the entries of a unknown vector $\boldsymbol{x}$ are observed through a known mixing matrix $\boldsymbol{A}$ of large dimensions, with some additive white Gaussian noise (AWGN) $\boldsymbol{w}$. The mixing matrix is typically a dense matrix, so that the factor graph representation of the model is densely connected graph. Keeping this factor graph representation in mind, the AMP algorithm can be interpreted as EP message passing [34] on the densely connected graph, in which the large dimensions of $\boldsymbol{A}$ are used to simplify the algorithm using large system argumentations. GAMP is derived based on similar principles, but overcoming the assumption of AWGN channel. Both techniques have been extensively applied since they were proposed [14, 15, 35], especially within the context of compressed sensing and sparse estimation problems.

We have used GAMP to benchmark some of our proposed receiver designs, as e.g. in **Paper C**. We also embedded it into the BP-MF framework to design a low-complexity OFDM receiver in **Paper E**.

Chapter 4

Design of Wireless Receivers: State-of-the-Art and Thesis Contributions

With the principles of variational inference and message passing algorithms laid out in Chapters 2 and 3, we proceed in this chapter to describe their application to the design of receiver algorithms in digital communications systems. We start with a short description of the problem of receiver design for digital communications. After this generic discussion, we review in more depth the specific problems that have been approached in this thesis, detailing the state-of-art and our contributions to solve them. These are collected in **Papers A-F**, and constitute the main contribution of this thesis.

Digital communications deal with the transmission of binary information from a transmitter to a receiver. The transmitter converts the binary information to an analogue waveform and sends this waveform over a physical medium, such as a wire or open space, which we call the channel. At the receiver, the received signal appears distorted by channel and interference and is further corrupted due to thermal noise. Not only has the receiver to recover the original binary information, but it must also deal with channel and interference effects, thermal noise, and synchronization issues. In addition, channel coding is often used to mitigate the effects of intersymbol interference (ISI), so the particular code structure needs to be accounted for as well. In such a context, iterative signal processing based on message passing techniques emerges as a feasible tool for receiver design, as it allows to perform approximate inference in large probabilistic systems including both continuous and discrete random variables.

The idea of iterative turbo processing dates back to 1993, when Berrou *et* *al.* [36] proposed an encoding structure concatenating two simple encoders (turbo-codes) in parallel, associated with a feasible decoding procedure (turbo-decoding) that exchanges soft information between two soft-in-soft-out (SISO) decoders operating in parallel. The drastic performance improvement achieved with this turbo-code and turbo-decoder has lead to a paradigm shift in the coding community. It has also sparkled feverish activities worldwide [37–41] aiming at generalizing the turbo-principle to incorporate –in addition to the decoding procedure– other functionalities of receivers, such as channel estimation, channel equalization, synchronization, multi-user interference cancellation, *etc.* Well-investigated applications today are turbo-equalization as well as joint channel estimation and decoding in multi-user systems. It has been recently recognized that turbo-algorithms are particular instances of the variational Bayesian method. For instance, Berrou's turbo-decoding is known to be an instance of BP algorithm [42]. Other principles, such as VMP or EP, have also been widely applied to design iterative receiver structures.

In this thesis, we focus on the design of iterative receivers for different communication systems integrating tasks such as channel estimation, phase noise estimation, equalization, and decoding. With this aim, we rely on established message passing frameworks, namely BP, the MF approximation, and EP. For each of the studied problems, a probabilistic model of the system is developed. Subsequently, the message passing tool (or combination of tools) most suitable to perform variational inference in such model is chosen, applied, and the performance of the resulting algorithm is evaluated against previously proposed solutions. In the following, we describe each of the problems that have been studied in the thesis, including a short review of their state-of-art, and specify the contributions made for each of the areas.

1 Turbo Equalization

Iterative equalization and decoding of ISI channels has been thoroughly studied over the last two decades, and many currently well known iterative equalizer structures have been derived, e.g. decision feedback equalizers [43] and turbo-equalizers [39, 41]. More recently, the popularization of message passing algorithms and factor graphs [17] has inspired the proposal of various message passing solutions. For instance, a BP-based turbo equalizer has been proposed in [44]. BP-based equalizers suffer, however, from a large computational complexity that scales exponentially with the modulation order and the amount of channel taps L.

Other message passing algorithms, alternative to BP, have been proposed to circumvent the aforementioned complexity problem. In [45], the residual interference plus noise component is approximated as a Gaussian variable, similarly to [11], in an approach that can be interpreted as direct approximation of BP messages by Gaussian messages. While this Gaussian approximation successfully reduces the complexity of the receiver to $\mathcal{O}(L^2)$ per symbol, this is achieved at the expense of a significant degradation in bit-error-rate (BER). The trade-off between complexity and BER can be adjusted by resorting to algorithms which apply the same Gaussian approximation as [11], but only on a subset of the interfering symbols. This approach, presented in [30], is coined partial Gaussian approximation (PGA). The PGA algorithm for a generic channel matrix was developed in [30]. Another alternative to obtain Gaussian messages and, consequently, low-complexity equalization is to use Gaussian EP, as proposed in [25] and **Paper A**. As shown in **Paper A**, applying EP yields a significant improvement in the equalizer's performance, compared to directly approximating BP messages as Gaussian.

The computational complexity ($\mathcal{O}(L^2)$ per symbol [11, 24, 39, 41, 45]) of the linear MMSE SISO equalizer in time domain is still too high when the length of channel taps L is long. For instance, in broadband wireless and underwater acoustic communications, L may often be dozens or hundreds [46, 47]. This has motivated researchers to pursue alternative equalization algorithms and, in particular, algorithms that perform in the frequency domain, rather than the time domain. Along these lines, single carrier frequency domain equalization (SC-FDE) technique is an attractive technology for wireless communications due to its ability to cope with the temporal dispersion introduced by multipath channels. It preserves the performance, efficiency and low complexity benefits of its OFDM counterpart, while being less sensitive to power amplifier nonlinearities and carrier frequency offsets, in addition to exhibiting a lower peak-to-average transmitted power ratio [46]. For these reasons, SC-FDE has been selected as the access scheme for the uplink of the 3GPP long term evolution (LTE) and LTE advanced standards [48].

Recently, the linear MMSE equalizer has been implemented in the frequency domain (with or without the assistance of cyclic prefixing), reducing the equalizer's complexity to logarithmic level [49–52]. Although its low complexity makes it attractive, the frequency domain linear MMSE (FD-LMMSE) equalizer may suffer from significant performance loss when the transmitted signal is severely distorted by the ISI channel. Guo *et. al.* developed a turbo frequency domain equalization (FDE) algorithm [53] based on GAMP [33], which can achieve significant performance gain with slight complexity increase compared to FD-LMMSE equalization. This turbo FDE algorithm can be regarded as the state-of-the-art solution for equalization in frequency domain.

In this thesis, we apply the message passing frameworks to design receivers performing iterative equalization and decoding in time domain (see **Papers A** and **B**) and frequency domain (see **Paper C**).

Paper A: Iterative Receiver Design for ISI Channels Using Combined Belief- and Expectation-Propagation In this contribution, we formulate an approximate inference method combining BP and EP and apply it to design a turbo equalization and decoding receiver for ISI channels. The proposed receiver, using Gaussian message passing approximated by EP in channel equalization part, avoids the exponential complexity problem of BP-based turbo equalizers. The numerical assessment of our proposed receiver illustrates the advantages of applying the combined BP-EP framework over receivers using solely BP and using BP with direct Gaussian approximation of its messages.

Paper B: Turbo-Equalization Using Partial Gaussian Approximation We further develop the BP-EP receiver in **Paper A**, by applying the PGA principle proposed in [30] to modify the output messages from equalization. Since PGA allows the receiver to tune the number of symbols that are considered as strong interferers, the proposed receiver enables a flexible performance-complexity tradeoff. The simulation results illustrate the merits of the new turbo equalization receiver compared to the receiver we proposed in **Paper A** and other benchmarks.

Paper C: Message-Passing Receivers for Single Carrier Systems with Frequency Domain Equalization In this contribution, we design a turbo equalization receiver based on the BP-MF framework for SC-FDE systems. Two receiver algorithms with, respectively, parallel and sequential message passing schedules are proposed in the MF part for channel equalization. Monte Carlo simulations show that our proposed design outperforms a similar structure derived using GAMP, and performs very closely to the matched filter bound.

2 Sparse Channel Estimation

Turbo equalizers in time or frequency domain are designed under the assumption that exact channel state information (CSI) is known. However, it is typically unknown and should be estimated in communication receivers. It is well-known that the accuracy of channel estimation is a crucial factor determining the overall performance in wireless communication systems and networks, in terms of BER and throughput. In addition, as the communication bandwidth increases, multipath propagation channels are usually dominated by a small number of significant paths, resulting in most of the channel coefficients being either zero or nearly zero, and therefore compressive sensing and sparse signal reconstruction become very powerful tools for the design of channel estimators. Consequently, we also investigate sparse channel estimation which can provide CSI for detection and equalization in OFDM or SC-FDE receivers.

Various Bayesian and non-Bayesian approaches have been proposed for sparse signal reconstruction in the literature. Greedy constructive algorithms such as orthogonal matching pursuit (OMP) [54] and compressive sampling MP (CoSaMP) [55] share the common characteristic that a proper iteration number or a predetermined sparsity is required to stop the iteration. Moreover, these greedy algorithms are non-Bayesian, and are therefore not very suitable to be embedded in receiver structures derived using Bayesian formalisms. There are many convex optimization based methods, such as the very popular LASSO regression [56, 57] using l_1 norm or Laplace prior, FOCUSS (focal undetermined system solver) algorithm [58] using l_2 norm or Gaussian prior and smooth l_0 (SL0) algorithms [59, 60] using smoothly approximate of the l_0 pseudo norm. These convex optimization based methods can usually be recast into a Bayesian MAP estimation form, by designing appropriate prior pdfs for the entries of the sparse vector that play the role of the different regularization terms in the convex optimization formulation. Sparse Bayesian learning (SBL) [61–64] has also been proposed for sparse signal reconstruction methods. Instead of working directly with a prior, SBL approaches often model a two-layer (2-L) or three-layer (3-L) hierarchical structure using random hyper-parameters.

Given that direct computation of a Bayesian estimate is often intractable for most choices of a prior pdf, the hierarchical modeling approach allows for the use of iterative estimation approaches instead. For instance, the generalized expectation-maximization (EM) algorithms has been a popular choice for this task, along with variational inference approaches. Most recently, SBL has been efficiently implemented using BP [65, 66] and AMP [35, 67]. However, these methods assume that the power of noise is known, which may not be true in many applications.

Using 2-L and 3-L hierarchical prior models, Pedersen et al. proposed a MF-based algorithm [68] to approximate a sparse Bayesian estimate of radio channels in OFDM systems. Since the MF-based iterative algorithm updates the estimates of all channel taps with length L at once –i.e. it jointly updates the full vector of channel taps– its computational complexity is as high as $\mathcal{O}(L^3)$ per iteration. Such large complexity stems from the inversion of an $L \times L$ matrix required at each iteration. A low complexity MF-based SBL algorithm, which decomposes the inversion of said matrix into a set of inversions of matrices with smaller dimension, was later developed to reduce the high complexity for OFDM systems [69]. The tradeoff between complexity and performance of channel estimation can be adjusted by selecting the size of such matrix inversions, implying that the reduction of complexity is at the cost of performance loss. Recently, a low complexity MF-based SBL algorithm which completely avoids matrix inversions has been applied to estimate both gains and delays of channel propagation paths [16].

In our work, we have sought to find low-complexity and accurate SBL algorithms that, in addition, include the estimation of the noise variance. Our main contribution is described next.

Paper D: Low Complexity Sparse Bayesian Learning Using Combined BP and MF with a Stretched Factor Graph In this contribution, a low complexity SBL algorithm is proposed based on the BP-MF framework for in underdetermined linear systems. We also simplify the BP messages passed on the densely connected subgraph by approximating some BP messages to further reduce the computational complexity, yielding an approximate BP-MF SBL algorithm. The proposed SBL algorithms present better MSE performance and lower complexity than the original algorithm derived using solely MF.

3 Receivers Performing Iterative Estimation, Equalization and Decoding

Many diverse iterative receiver architectures performing joint estimation (including channel response, noise power and/or phase noise), equalization and decoding (JEED) for a multiplicity of communication systems have been proposed by signal processing researchers over the last 20 years. While many of the early designs [38, 51, 52, 70–73] were based on ad-hoc extensions of the turbo equalization principle to include estimation, which was designed individually as described in Section 2 of this chapter, holistic design methodologies [12–16, 74–77] have dominated the scene in the recent past. Among the latter, the use of message passing frameworks in which a global objective function is iteratively optimized has been one of the main approaches. We shortly review some of them next.

As we discussed in Chapter 3, BP often presents good performance and therefore has been widely applied for designing receivers iteratively performing joint estimation, equalization and decoding in communication systems [14, 74, 75, 77]. However, such modern systems usually involve a complicated signal model with a densely connected factor graph in which both discrete and continuous random variables coexist, along with non-linear functions, cumbersome distributions, such as large mixtures of distributions. Those lead to the computations of messages being significantly complex and even intractable when BP is applied directly. Due to this fact, researchers have sought to exploit approximate message techniques, often involving Gaussian approximations, to achieve feasible solutions.

Considering the respective advantages and disadvantages of BP, MF and EP, the combinations of two or more of them can be exploited to design receivers in modern communication systems in complicated scenarios. The BP-MF framework has been applied to JEED receivers for OFDM systems [12, 16, 76] and MIMO-OFDM systems [13]. Paper [15] uses combined BP-EP to design JEED receiver for OFDM systems and also investigates methods to reduce the computational complexity by approximating some of the messages.

In this thesis, we propose designs for JEED receivers in two different communication scenarios. These contributions are detailed next.

Paper E: A Low Complexity OFDM Receiver with Combined GAMP and

MF Message Passing In this contribution, we apply the approximate BP-MF algorithm proposed in **Paper D** to carry out channel estimation in time-frequency domain for a JEED OFDM receiver. Our numerical assessment demonstrates the advantages the proposed receiver in terms of BER performance, complexity and convergence rate.

Paper F: A BP-MF-EP Based Iterative Receiver for Joint Phase Noise Estimation, Equalization and Decoding This contribution is the extension of BP-EP-based turbo equalization proposed in **Paper A** to the scenario of unknown phase noise. We exploit the BP-MF-EP framework to design an iterative receiver for joint phase noise estimation, equalization and decoding. In addition, a second-order Taylor expansion is chosen to approximate some MF messages, so that it provides a Gaussian prior for the phase noise estimation subgraph. Simulation results confirm that remarkable performance gain can be achieved with the proposed approach.

Chapter 5

Conclusions and Outlook

In this thesis, we exploit the formulation of signal processing problems in digital communications as approximate inference problems in probabilistic models. This, in turn, allows for the design of iterative signal processing solutions via message passing algorithms in (probabilistic) factor graphs. This general methodology has the advantage of providing large flexibility in the design of the signal processing algorithms. This flexibility is due to two main reasons:

- On the one hand, there is flexibility in the modeling of each particular problem, i.e. the definition of the probabilistic system that represents the problem. In turn, this results in flexibility regarding the definition of the factor graph representation of the system.
- On the other hand, there is as well flexibility with respect to the approximate inference methods, i.e. the particular message passing technique used to perform estimation in the probabilistic system and associated factor graph.

By finding appropriate combinations of the two aspects mentioned above, receiver algorithms that attain remarkable performance can be derived.

While the advantages of using message passing techniques to design digital receivers are nowadays well established in the signal processing community, most instances of receivers are derived using only a single message passing technique. For some specific systems, such designs may yield solutions with intractably high computational complexity or unsatisfactory performance. An important contribution of this thesis is the proposal of signal processing algorithms that combine different message passing approaches. We have shown that, for many problems, algorithms derived following a combination of different message passing methods lead to better designs than algorithms derived based on a single inference approach. Depending on the particular problem, the superiority of our receivers is in terms of performance, or in terms of computational complexity and convergence speed. For instance, a low complexity iterative equalization and decoding receiver has been proposed for ISI channels based on combined BP-EP (see the method in **Paper A**), which performs better than approximate BP-based receiver, and has lower complexity than standard BP-based receiver. In addition, we have also proposed approaches to approximate messages that lower the computational complexity of the designed receivers without severely degrading their performance, as done e.g. in **Papers E and F**.

To sum up, the insights obtained through the work described in this thesis can be synthesized in the following procedure to design a message passing based receiver for digital communications:

1. In a first step, BP message passing is considered as the first priority.
2. Different probabilistic descriptions of the system and, correspondingly, different factor graph representations should be explored.
3. For each of the models obtained in 2, if BP does not provide tractable solutions, an appropriate combination of techniques needs to be sought. A common solution is to replace BP by EP or MF for those graph nodes leading to messages with intractable complexity.
4. If the above step still results in high complexity, then approximate message passing approaches should be considered.

To finalize, the above discussion leads to the conclusion that further research is needed in the theoretical formalization of message passing approaches combining two or more inference principles. Some of the mixed message passing approaches applied in this work have been obtained by heuristic combinations of existing techniques. As such, the resulting algorithms cannot be guaranteed to optimize any known, global objective function. In spite of this, the ad-hoc combination of message updating rules has proved to result in receiver algorithms with excellent performance when an appropriate combination for a particular problem is found. This fact hints at the idea that it may be possible to justify such mixtures of inference approaches by sound theoretical arguments. While the BP-MF framework or, more generally, region-based approaches have provided a significant step towards that goal, there are still many open aspects. For instance, the embedding of approximate message passing algorithms (AMP or GAMP) with other techniques such as BP or MF is still an open problem that should be addressed in the future.

References

[1] S. M. Kay, *Fundamentals of Statistical Processing, Volume I: Estimation Theory*, 1st ed. Prentice-Hall, April 1993.

[2] ——, *Fundamentals of Statistical Processing, Volume II: Detection Theory*, 1st ed. Prentice-Hall, Feb. 1998.

[3] D. J. C. MacKay, *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003.

[4] C. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[5] H. Wymeersch, *Iterative receiver design*, 1st ed. Cambridge University Press, Sept. 2007.

[6] J. Besag and P. J. Green, "Spatial statistics and Bayesian computation," *Journal of the Royal Statistical Society, Series B*, vol. 55, no. 1, pp. 25–37, 1993.

[7] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov chain Monte Carlo in Practice*, 1st ed. New York: Chapman and Hall/CRC, Dec. 1996.

[8] C. P. Robert and G. Casella, *Monte Carlo statistical methods. Springer Texts in Statistics*, 2nd ed. New York: Springer, June 2005.

[9] M. J. Wainwright and M. I. Jordan, *Graphical Models, Exponential Families, and Variational Inference, Foundations and Trends(r) in Machine Learning (Book 1)*, 1st ed. Now Publishers Inc., Dec. 2008.

[10] J. Yedidia, W. Freeman, and Y. Weiss, "Constructing free-energy approximations and generalized belief propagation algorithms," *IEEE Trans. Inform. Theory*, vol. 51, no. 7, pp. 2282–2312, Jul. 2005.

[11] Q. Guo and L. Ping, "LMMSE turbo equalization based on factor graphs," *IEEE J. Select. Areas Commun.*, vol. 26, no. 2, pp. 311–319, Feb. 2008.

[12] M.-A. Badiu, G. E. Kirkelund, C. Navarro Manchón, E. Riegler, and B. H. Fleury, "Message-passing algorithms for channel estimation and decoding using approximate inference," in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT 2012)*, July 2012, pp. 2376–2380.

[13] C. Navarro Manchón, G. E. Kirkelund, E. Riegler, L. P. B. Christensen, and B. H. Fleury, "Receiver architectures for MIMO-OFDM based on a combined VMP-SP algorithm," *2011, arXiv:1111.5848 [stat.ML]*.

[14] P. Schniter, "A message-passing receiver for BICM-OFDM over unknown clustered-sparse channels," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 8, pp. 1462–1474, Dec. 2011.

[15] S. Wu, L. Kuang, Z. Ni, J. Lu, D. Huang, and Q. Guo, "Expectation propagation approach to joint channel estimation and decoding for OFDM systems," in *IEEE Int. Conference on Acustic, Speech and Signal Processing (ICASSP 2014)*, 4-9 May 2014, pp. 1941–1945.

[16] T. L. Hansen, P. B. Jørgensen, M. Badiu, and B. H. Fleury, "Joint sparse channel estimation and decoding: Continuous and discrete domain sparsity," *CoRR*, vol. abs/1507.02954, 2015. [Online]. Available: http://arxiv.org/abs/1507.02954

[17] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[18] T. P. Minka, "Expectation propagation for approximate Bayesian inference," in *Proceedings of the 17th Conf. on Uncertainty in Artificial Intelligence*, 2001, pp. 362–369.

[19] T. Heskes, M. Opper, W. Wiegerinck, O. Winther, and O. Zoeter, "Approximate inference techniques with expectation constraints," *Journal of Statistical Mechanics – Theory and Experiment*, Sep. 2005.

[20] M. Beal, "Variational algorithms for approximate inference," Ph.D. dissertation, University of Cambridge, May 2003.

[21] J. Winn and C. Bishop, "Variational message passing," *Journal of Machine Learning Research*, vol. 6, pp. 661–694, 2005.

[22] J. Dauwels, "On variational message passing on factor graphs," in *Proc. IEEE International Symposium on Information Theory (ISIT 2007)*, Jun. 2007, pp. 2546–2550.

[23] E. Riegler, G. E. Kirkelund, C. Navarro Manchón, M.-A. Badiu, and B. H. Fleury, "Merging belief propagation and the mean field approximation: A free energy approach," *IEEE Trans. Inform. Theory*, vol. 59, no. 1, pp. 588–602, Jan. 2013.

[24] P. Sun, C. Zhang, Z. Wang, C. Navarro Manchón, and B. H. Fleury, "Iterative receiver design for ISI channels using combined belief- and expectation-propagation," *IEEE Signal Processing Lett.*, vol. 22, no. 10, pp. 1733–1737, Oct. 2015.

[25] 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. Communication, Control, and Computing*, Sep. 2006, pp. 27–29.

[26] H.-A. Loeliger, "An introduction to factor graphs," *IEEE Trans. Signal Processing*, pp. 28–40, Jan. 2004.

[27] V. Isham, "An introduction to spatial point processes and Markov random fields," *Int. Stat. Rev.*, vol. 49, pp. 21–43, 1981.

[28] F. V. Jensen, *An Introduction to Bayesian Networks*. New York: Springer-Verlag, 1996.

[29] G. Parisi, *Statistical field theory*. Perseus Books, 1988.

[30] Q. Guo, D. Huang, L. Ping, S. Nordholm, J. Xi, and P. Li, "Soft-in soft-out detection using partial Gaussian approximation," *IEEE Access*, vol. 2, pp. 427–436, May 2014.

[31] S. Boyd and L. Vandenberghe, *Convex Optimization*, 1st ed. Cambridge University Press, March 2004.

[32] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inform. Theory*, vol. 57, no. 2, pp. 764–785, Feb. 2011.

[33] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT 2011)*, Aug. 2011, pp. 2168–2172.

[34] B. Cakmak, O. Winther, and B. H. Fleury, "S-AMP: Approximate message passing for general matrix ensembles," in *2014 IEEE Information Theory Workshop (ITW)*, Nov. 2014, pp. 192–196.

[35] M. Al-Shoukairi and B. Rao, "Sparse Bayesian learning using approximate message passing," in *2014 48th Asilomar Conference on Signals, Systems and Computers*, Nov. 2014, pp. 1957–1961.

[36] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo codes," in *Proc. IEEE International Conference on Communications, (ICC'93)*, May 1993, pp. 1064–1070.

[37] C. Laot, A. Glavieux, and J. Labat, "Turbo equalization: adaptive equalization and channel decoding jointly optimized," *IEEE J. Select. Areas Commun.*, vol. 19, no. 9, pp. 1744–1752, Sep 2001.

[38] B. Lu and X. Wang, “Bayesian blind turbo receiver for coded OFDM systems with frequency offset and frequency-selective fading,” *IEEE J. Select. Areas Commun.*, vol. 19, no. 12, pp. 2516–2527, Dec 2001.

[39] M. Tüchler, R. Koetter, and A. Singer, “Turbo equalization: Principles and new results,” *IEEE Trans. Commun.*, vol. 50, pp. 754–767, 2002.

[40] T. Abe and T. Matsumoto, “Space-time turbo equalization in frequency-selective MIMO channels,” *IEEE Trans. Veh. Technol.*, vol. 52, no. 3, pp. 469–475, May 2003.

[41] R. Koetter, A. C. Singer, and M. Tücher, “Turbo equalization,” *IEEE Signal Processing Mag.*, vol. 21, pp. 67–80, Jan. 2004.

[42] R. McEliece, D. MacKay, and J. Cheng, “Turbo decoding as an instance of Pearl’s belief “propagation” algorithms,” *IEEE J. Select. Areas Commun.*, vol. 16, no. 2, pp. 140–152, Feb. 1998.

[43] S. L. Ariyavisitakul and Y. Li, “Joint coding and decision feedback equalization for broadband wireless channels,” *IEEE J. Select. Areas Commun.*, vol. 16, no. 9, pp. 1670–1678, Dec. 1998.

[44] 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, 2005.

[45] X. Wang and H. Poor, “Iterative (turbo) soft interference cancellation and decoding for coded CDMA,” *IEEE Trans. Commun.*, vol. 47, pp. 1046–1061, Jul. 1999.

[46] D. Falconer, S. Ariyavisitakul, A. Benyamin-Seeyar, and B. Eidson, “Frequency domain equalization for single-carrier broadband wireless systems,” *IEEE Commun. Mag.*, vol. 40, no. 4, pp. 58–66, Apr. 2002.

[47] M. Stojanovic, “Underwater acoustic communications: Design considerations on the physical layer,” in *2008 Fifth Annual Conference on Wireless on Demand Network Systems and Services (WONS 2008)*, Germany, Jan. 2008, pp. 1–10.

[48] E. Dahlman, S. Parkvall, J. Sköld, and P. Beming, *3G Evolution: HSPA and LTE for Mobile Broadband*, 2nd ed. Academic Press, 2008.

[49] M. Tüchler and J. Hagenauer, “Turbo equalization using frquency domain equalizers,” in *Proc. Allerton Conf., Monticello*, IL, USA, Oct. 2000, pp. 1234–1243.

[50] A. Gusmão, P. Torres, R. Dinis, and N. Esteves, "A turbo FDE technique for reduced-CP SC-based block transmission systems," *IEEE Trans. Commun.*, vol. 55, no. 1, pp. 16–20, Jan. 2007.

[51] H. Liu and P. Schniter, "Iterative frequency-domain channel estimation and equalization for single-carrier transmissions without cyclic-prefix," *IEEE Transactions on Wireless Communications*, vol. 7, no. 10, pp. 3686–3691, October 2008.

[52] Q. Guo, L. Ping, and D. Huang, "A low-complexity iterative channel estimation and detection technique for doubly selective channels," *IEEE Trans. Commun.*, vol. 8, no. 8, pp. 4340–4349, Aug. 2009.

[53] Q. Guo, D. Huang, S. Nordholm, J. Xi, and Y. Yu, "Iterative frequency domain equalization with generalized approximate message passing," *IEEE Signal Processing Lett.*, vol. 20, no. 6, pp. 559–562, June 2013.

[54] J. A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Trans. Inform. Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.

[55] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[56] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. of the Royal Statistical Society*, vol. 58, no. 1, pp. 267–288, 1996.

[57] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basic pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, Aug. 1998.

[58] B. D. Rao, "Analysis and extensions of the FOCUSS algorithm," in *Conference Record of the Thirtieth Asilomar Conference on Signals, Systems and Computers, 1996.*, Nov. 1996, pp. 1218–1223 vol.2.

[59] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed l^0 norm," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 289–301, Jan. 2009.

[60] M. M. Hyder and K. Mahata, "An improved smoothed l^0 approximation algorithm for sparse representation," *IEEE Trans. Signal Processing*, vol. 58, no. 4, pp. 2194–2205, April 2010.

[61] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, June 2001.

[62] M. E. Tipping and A. C. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," *Proc. 2003 9th international Workshop on Artificial Intelligence and Statistics,*, 2003.

[63] D. Shutin, T. Buchgraber, S. R. Kulkarni, and H. V. Poor, "Fast variational sparse Bayesian learning with automatic relevance determination for superimposed signals," *IEEE Trans. Signal Processing*, vol. 59, no. 12, pp. 6257–6261, Dec. 2011.

[64] N. L. Pedersen, C. Navarro Manchón, M.-A. Badiu, D. Shutin, and B. H. Fleury, "Sparse estimation using Bayesian hierarchical prior modeling for real and complex linear models," *Signal Processing*, vol. 115, no. 0, pp. 94 – 109, Oct. 2015.

[65] X. Tan and J. Li, "Computationally efficient sparse Bayesian learning via belief propagation," in *2009 Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*, Nov. 2009, pp. 1566–1570.

[66] D. Baron, S. Sarvotham, and R. Baraniuk, "Bayesian compressive sensing via belief propagation," *IEEE Trans. Signal Processing*, vol. 58, no. 1, pp. 269–280, Jan. 2010.

[67] S. Som and P. Schniter, "Compressive imaging using approximate message passing and a Markov-tree prior," *IEEE Trans. Signal Processing*, vol. 60, no. 7, pp. 3439–3448, July 2012.

[68] N. L. Pedersen, C. Navarro Manchón, D. Shutin, and B. H. Fleury, "Application of Bayesian hierarchical prior modeling to sparse channel estimation," pp. 3487–3492, June 2012.

[69] N. L. Pedersen, C. Navarro Manchón, and B. H. Fleury, "Low complexity sparse Bayesian learning for channel estimation using generalized mean field," *20th European Wireless Conference*, pp. 838–843, June 2014.

[70] M. Nicoli, S. Ferrara, and U. Spagnolini, "Soft-iterative channel estimation: Methods and performance analysis," *IEEE Transactions on Signal Processing*, vol. 55, no. 6, pp. 2993–3006, June 2007.

[71] Y. Wu, X. Zhu, and A. K. Nandi, "Low complexity adaptive turbo frequency-domain channel estimation for single-carrier multi-user detection," *IEEE Transactions on Wireless Communications*, vol. 7, no. 11, pp. 4094–4099, November 2008.

[72] H. Nguyen-Le, T. Le-Ngoc, and C. C. Ko, "Turbo processing for joint channel estimation, synchronization, and decoding in coded MIMO-OFDM systems," *Eurasip Journal on Wireless Communications and Networking*, vol. 2009, no. 1, pp. 53–56, 2009.

[73] Y. Yapici and A. O. Yilmaz, "Low-complexity iterative channel estimation and tracking for time-varying multi-antenna systems," in *2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, 2009, pp. 1317–1321.

[74] T. Wo, P. 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.

[75] C. Knievel, P. A. Hoeher, A. Tyrrell, and G. Auer, "Multi-dimensional graph-based soft iterative receiver for MIMO-OFDM," *IEEE Trans. Commun.*, vol. 60, no. 6, pp. 1599–1609, June 2012.

[76] M.-A. Badiu, C. Navarro Manchón, and B. H. Fleury, "Message-passing receiver architecture with reduced-complexity channel estimation," *IEEE Commun. Lett.*, vol. 17, no. 7, pp. 1404–1407, July 2013.

[77] 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 Processing*, vol. 62, no. 6, pp. 1576–1589, March 2014.

Part II

Papers

Paper A

Iterative Receiver Design for ISI Channels Using Combined Belief- and Expectation-Propagation

Peng Sun, Chuanzong Zhang, Zhongyong Wang, Carles Navarro Manchón, and Bernard H. Fleury

The paper has been published in the
IEEE Signal Processing Letters Vol. 22(10), pp. 1733–1737, 2015.

Abstract

In this letter, a message-passing algorithm that combines belief propagation and expectation propagation is applied to design an iterative receiver for intersymbol interference channels. We detail the derivation of the messages passed along the nodes of a vector-form factor graph representing the underlying probabilistic model. We also present a simple but efficient method to cope with the "negative variance" problem of expectation propagation. Simulation results show that the proposed algorithm outperforms, in terms of bit-error-rate and convergence rate, a LMMSE turbo-equalizer based on Gaussian message passing with the same order of computational complexity.

1 Introduction

Since optimal detection of data transmitted across an intersymbol interference (ISI) channel, like the multipath wireless channel, is typically impractical, suboptimal receiver structures that approach the performance of the optimal detector have been proposed, with turbo equalization [1] being the most emblematic instance. In turbo equalization, the turbo principle –originally used for decoding concatenated codes [2]– is applied by regarding the ISI channel as an encoder acting on the transmitted symbols.

The above "turbo"-processing algorithms are instances of belief propagation (BP) applied on a factor graph representing the underlying probabilistic model [3]. Additional equalizer structures, which implement other variants of BP, have been proposed, e.g. [4]. However, BP-based equalizers suffer from an inherent drawback: their complexity grows exponentially with the channel length or the number of non-zero coefficients (depending on the selected factor graph representation) and the modulation order.

Different approaches have been proposed to circumvent the aforementioned complexity issue. Basically, they introduce approximations that make the messages passed in the subgraph representing the ISI channel Gaussian. In [5] this is achieved by assuming the interference plus noise component with respect to each modulation symbol to be Gaussian and exploiting a relationship between the extrinsic values of the symbols when the channel is driven by these symbols and the LMMSE symbol estimates when the channel is driven by Gaussian inputs. This approach turns out to be equivalent to that in [6] when applied to turbo-equalization [5]. In [7] a combined use of Gaussian expectation propagation (EP) [8, 9] and BP is proposed. The use of EP, however, leads to an unstable algorithm due to the fact that computed Gaussian EP messages may have a negative variance. In [7] the authors propose to circumvent this problem by replacing each EP message with a geometric mixture of said message and a standard Gaussian message, parameterized with a damping/mixing factor. However, “good" sequences of values of the damping factor versus the iteration index of the algorithm need to be tuned in advance via simulations, which severely limits the practicability of the proposed approach.

In this paper we formulate an approximate inference method combining BP and EP and apply it to a vector-form factor graph representation of the probabilistic model for ISI channels to design a receiver algorithm performing joint equalization of ISI channels and data detection. The obtained design is similar to that presented in [7]. We propose a simple solution to avoid the instability problem of EP that leads to a fast converging algorithm. We present a detailed derivation of the turbo-equalizer and a numerical evaluation that compares its performance with that of the receiver proposed in [5]. The simulation results show that for the same complexity our design performs better and converges faster than that in [5], while avoiding the practical issues inherent to that in [7].

Notation- Boldface lowercase and uppercase letters denote vectors and matrices, respectively. The identity matrix of size M is represented by $\boldsymbol{I}_M$. Superscript $(\cdot)^{\mathrm{T}}$ indicates transposition of a vector or matrix. The probability density function (pdf) of a multivariate Gaussian distribution with mean vector $\boldsymbol{m}$ and covariance matrix $\boldsymbol{V}$ is represented by $\mathcal{N}(\boldsymbol{x};\boldsymbol{m},\boldsymbol{V})$. The relation $f(x) = cg(x)$ for some positive constant c is written as $f(x) \propto g(x)$.

2 System Model

The information bit vector $\boldsymbol{b} = [b_1, \ldots, b_K]^{\mathrm{T}}$ is encoded and interleaved, yielding the codeword vector $\boldsymbol{c} = [c_1, \ldots, c_N]^{\mathrm{T}}$. The coded bits are then mapped onto a binary phase shift keying (BPSK) constellation, resulting in the vector of modulated symbols $\boldsymbol{x} = [x_1, \ldots, x_N]^{\mathrm{T}}$, which are then transmitted over a frequency-selective channel corrupted with AWGN. The (baseband discrete-time) signal observed at the receiver is described by the vector $\boldsymbol{r} = [r_1, \ldots, r_{N+L-1}]^{\mathrm{T}}$ with entries

$$r_i = \sum_{l=0}^{L-1} h_l x_{i-l} + n_i = \boldsymbol{h}^{\mathrm{T}} \boldsymbol{s}_i + n_i. \tag{A.1}$$

Here, $\boldsymbol{s}_i = [x_{i-L+1}, \ldots, x_i]^{\mathrm{T}}$ with $x_i = 0$ for $i < 1$ and $i > N$, $\boldsymbol{h} = [h_{L-1}, \ldots, h_0]^{\mathrm{T}}$ denotes the vector of channel weights, and n_i is the ith sample of a white Gaussian noise vector with component variance σ^2.

2.1 Probabilistic Model and Factor Graph

The posterior probability mass function (pmf) of vectors $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{x}$ and $\boldsymbol{s}$ given the received signal vector $\boldsymbol{r}$ reads

$$\begin{aligned} p\left(\boldsymbol{b},\boldsymbol{c},\boldsymbol{x},\boldsymbol{s}|\boldsymbol{r}\right) \propto & \prod_{k=1}^{K} f_{b_k}\left(b_k\right) \times f_c\left(\boldsymbol{c},\boldsymbol{b}\right) \\ & \times \prod_{i=1}^{N} f_{r_i}\left(r_i,\boldsymbol{s}_i\right) f_{G_i}\left(\boldsymbol{s}_i,\boldsymbol{s}_{i-1},x_i\right) f_{M_i}\left(x_i,c_i\right) \\ & \times \prod_{i=N+1}^{N+L-1} f_{r_i}\left(r_i,\boldsymbol{s}_i\right) f_{G_i}\left(\boldsymbol{s}_i,\boldsymbol{s}_{i-1},0\right). \end{aligned} \tag{A.2}$$

In this expression $f_{b_k}(b_k)$ is the uniform prior pmf of the kth information bit, $f_c(\boldsymbol{c},\boldsymbol{b})$ stands for the coding and interleaving constraints, $f_{r_i}(r_i,\boldsymbol{s}_i) \triangleq p(r_i|\boldsymbol{s}_i) \propto \mathcal{N}(r_i;\boldsymbol{h}^{\mathrm{T}}\boldsymbol{s}_i,\sigma^2)$ denotes the likelihood term for $\boldsymbol{s}_i$, and $f_{M_i}(x_i,c_i)$ represents the modulation mapping. Finally, $f_{G_i}\left(\boldsymbol{s}_i,\boldsymbol{s}_{i-1},x_i\right)$ expresses the deterministic relationship between $\boldsymbol{s}_i$, $\boldsymbol{s}_{i-1}$ and x_i, given by

$$\boldsymbol{s}_i = \boldsymbol{G}\boldsymbol{s}_{i-1} + \boldsymbol{e}x_i \tag{A.3}$$

with the $L \times L$ matrix $\boldsymbol{G} = [\mathbf{0} \quad \boldsymbol{I}_{L-1}; \quad 0 \quad \mathbf{0}^{\mathrm{T}}]$ and the L vector $\boldsymbol{e} = [\mathbf{0}^{\mathrm{T}} \quad 1]^{\mathrm{T}}$, where $\mathbf{0}$ is a zero column vector with length $L-1$. Note that $\boldsymbol{G}$ factorizes as $\boldsymbol{G} = \boldsymbol{G}''\boldsymbol{G}'$ with $\boldsymbol{G}'' = [\ \boldsymbol{I}_{L-1} \quad \mathbf{0}\ ]^{\mathrm{T}}$ and $\boldsymbol{G}' = [\ \mathbf{0} \quad \boldsymbol{I}_{L-1}\ ]$ [5].

The vector-form factor graph representation [5] of the posterior pmf in (A.2) is depicted in Fig. A.1. It will be used for the derivation of the BP-EP-based receiver described in Section 4. Note that in this representation the subgraph representing the ISI channel (left part) in Fig. A.1 has a tree structure[1].

3 Combined BP-EP Message-passing Rule

We consider a factor graph of a generic probabilistic model made of a set of factor nodes $\mathcal{F}$, and a set of variable nodes $\mathcal{Z}$. The variable nodes are grouped into two disjoint subsets $\mathcal{Z}^{\mathrm{BP}}$ and $\mathcal{Z}^{\mathrm{EP}}$, i.e. $\mathcal{Z}^{\mathrm{BP}} \cup \mathcal{Z}^{\mathrm{EP}} = \mathcal{Z}$ and $\mathcal{Z}^{\mathrm{BP}} \cap \mathcal{Z}^{\mathrm{EP}} = \emptyset$. Let $m_{f\to z}(z)$ denote the messages from a factor node $f \in \mathcal{F}$ to a variable node $z \in \mathcal{Z}$, and $n_{z\to f}\left(z\right)$ be the message from variable node z

[1]Cycles appearing in "channel" subgraph of the scalar-form factor graph representation of the probabilistic model (A.2) are absorbed in the vector-form representation [5].

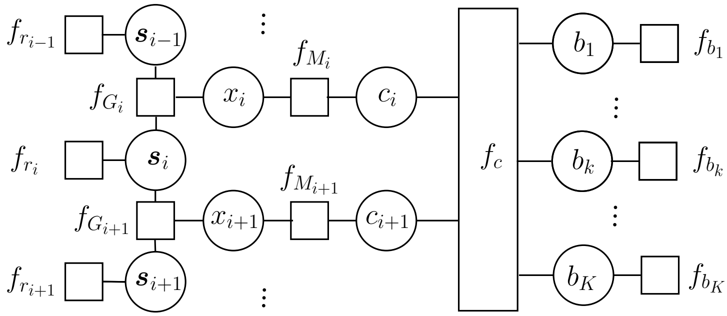


Fig. A.1: Vector-form factor graph representation of the probabilistic model (A.2).

to factor node f. With these definitions, the message update rules read

$$m_{f\to z}(z) = \sum_{\sim\{z\}} f(\boldsymbol{z}) \prod_{z'\in\mathcal{N}(f)\setminus\{z\}} n_{z'\to f}(z'), z \in \mathcal{Z}^{\mathrm{BP}} \tag{A.4}$$

$$m_{f\to z}(z) = \frac{\mathrm{Proj}_{\mathcal{E}_z}\left[m^{\mathrm{BP}}_{f\to z}(z)\, n_{z\to f}(z)\right]}{n_{z\to f}(z)}, \quad z \in \mathcal{Z}^{\mathrm{EP}} \tag{A.5}$$

$$n_{z\to f}(z) = \prod_{f'\in\mathcal{N}(z)\setminus\{f\}} m_{f'\to z}(z), \quad z \in \mathcal{Z}. \tag{A.6}$$

Here, $\sum_{\sim\{z\}}$ is the sum over all variables of $f = f(\boldsymbol{z})$ excluding z, $\mathcal{N}(z)$ and $\mathcal{N}(f)$ denote respectively the set of factor nodes connected to variable node z and the set of variable nodes connected to the factor node f. The superscript BP of the message in the right-hand expression in (A.5) indicates that this message from factor f to $z \in \mathcal{Z}^{\mathrm{EP}}$ is computed using the BP rule, i.e. (A.4). Moreover in this expression $\mathrm{Proj}_{\mathcal{E}_z}[\cdot]$ is the projection of the pdf given as an argument on a specified exponential family $\mathcal{E}_z$ [2]. Note that computing the messages from any factor node to any variable node requires the computation of a BP message. Moreover, messages passed from and to a variable node $z \in \mathcal{Z}^{\mathrm{BP}}$ ($z \in \mathcal{Z}^{\mathrm{EP}}$) are computed using the BP (EP) rule.

4 Iterative Receiver Design

In this section we derive a receiver that performs joint equalization and decoding for ISI channels by passing messages along the edges of the factor graph depicted in Fig. A.1. The complexity of standard BP applied on this factor graph grows exponentially with L, the dimension of the state vectors $\boldsymbol{s}_i$,

[2] For a pdf $b(z)$ $\mathrm{Proj}_{\mathcal{E}}[b(z)] = \arg\min_{b\prime(z)\in\mathcal{E}} D\left(b(z)||b\prime(z)\right)$, with $D(\cdot||\cdot)$ denoting the Kullback-Leibler divergence, and $\mathcal{E}$ being a specific exponential family.
As indicated by the indexing, $\mathcal{E}_z$ might depend on the variable node.

$\forall i$. Such intractable complexity can be reduced by approximating messages passed along the edges of the channel part of the graph (to the left of and including the variable nodes x_i, $\forall i$) with Gaussian messages. This can be done by approximating the messages from x_i to f_{G_i}, $\forall i$, with Gaussian messages. The EP framework provides an elegant and efficient tool to do so. Similarly to [7] we split the variable nodes in the graph as follows: $\mathcal{Z}^{\mathrm{EP}} = \{x_i; \forall i\}$ and $\mathcal{Z}^{\mathrm{BP}} = \mathcal{Z} \setminus \mathcal{Z}^{\mathrm{EP}}$. Moreover we set $\mathcal{E}_{x_i} = \mathcal{G}$, $\forall i$, where $\mathcal{G}$ is the Gaussian family.

4.1 Calculation of Messages

Equalization–Input Messages

Assuming that, for the ith symbol, the message from f_{M_i} to x_i can be expressed as $m^{\mathrm{BP}}_{f_{M_i}\to x_i}(x_i) = \beta_{i,1}\delta(x_i+1) + \beta_{i,2}\delta(x_i-1)$ and the message from x_i to f_{M_i} has the form $n_{x_i\to f_{M_i}}(x_i) \propto \mathcal{N}(x_i; \vec{m}_{x_i}, \vec{v}_{x_i})$, the belief $b(x_i) \propto m^{\mathrm{BP}}_{f_{M_i}\to x_i}(x_i)\, n_{x_i\to f_{M_i}}(x_i)$ of x_i has mean and variance

$$m^p_{x_i} = \frac{\beta_{i,2}\exp\{2\vec{m}_{x_i}/\vec{v}_{x_i}\} - \beta_{i,1}}{\beta_{i,2}\exp\{2\vec{m}_{x_i}/\vec{v}_{x_i}\} + \beta_{i,1}}, \tag{A.7}$$

$$v^p_{x_i} = 1 - (m^p_{x_i})^2. \tag{A.8}$$

The message $m_{f_{M_i}\to x_i}(x_i)$ is computed from (A.5) to be

$$\begin{aligned} m_{f_{M_i}\to x_i}(x_i) &= \frac{\mathrm{Proj}_{\mathcal{G}}[m^{\mathrm{BP}}_{f_{M_i}\to x_i}(x_i)\, n_{x_i\to f_{M_i}}(x_i)]}{n_{x_i\to f_{M_i}}(x_i)} \\ &\propto \mathcal{N}(x_i; \overleftarrow{m}_{x_i}, \overleftarrow{v}_{x_i}) \end{aligned} \tag{A.9}$$

where

$$\overleftarrow{v}_{x_i} = [(v^p_{x_i})^{-1} - \vec{v}^{-1}_{x_i}]^{-1} \tag{A.10}$$

$$\overleftarrow{m}_{x_i} = \overleftarrow{v}_{x_i}[(v^p_{x_i})^{-1} m^p_{x_i} - \vec{v}^{-1}_{x_i}\vec{m}_{x_i}]. \tag{A.11}$$

We further have $n_{x_i\to f_{G_i}}(x_i) = m_{f_{M_i}\to x_i}(x_i)$ by (A.6).

When running the BP-EP algorithm, it can be observed that the variance parameter $\overleftarrow{v}_{x_i}$ in (A.10) and (A.11) sometimes takes negative values, which results in a bad performance, see also [7]. To avoid this problem, the variance $\overleftarrow{v}_{x_i}$ is replaced by its absolute value $|\overleftarrow{v}_{x_i}|$ in both (A.10) and (A.11). We will see in the numerical evaluations that this simple "trick" is very efficient and provides a viable alternative to the damping method proposed in [7], see Section 1.

Equalization–Downward Messages

Assuming that the message $n_{\boldsymbol{s}_{i-1}\to f_{G_i}}(\boldsymbol{s}_{i-1}) \propto \mathcal{N}(\boldsymbol{s}_{i-1}; \boldsymbol{m}^{\downarrow}_{\boldsymbol{s}_{i-1}}, \boldsymbol{V}^{\downarrow}_{\boldsymbol{s}_{i-1}})$ is known, the message $m_{f_{G_i}\to \boldsymbol{s}_i}(\boldsymbol{s}_i)$ is obtained via (A.4) to be

$$m_{f_{G_i}\to \boldsymbol{s}_i}(\boldsymbol{s}_i) \propto \exp\left\{-\frac{1}{2}(\boldsymbol{s}_i - \boldsymbol{m}^{\Downarrow}_{\boldsymbol{s}_i})^{\mathrm{T}} \boldsymbol{V}^{\Downarrow -1}_{\boldsymbol{s}_i}(\boldsymbol{s}_i - \boldsymbol{m}^{\Downarrow}_{\boldsymbol{s}_i})\right\} \tag{A.12}$$

with

$$\begin{aligned} \boldsymbol{m}^{\Downarrow}_{\boldsymbol{s}_i} &= \boldsymbol{G}\boldsymbol{m}^{\downarrow}_{\boldsymbol{s}_{i-1}} + \boldsymbol{e}\overleftarrow{m}_{x_i} && \text{(A.13)}\\ \boldsymbol{V}^{\Downarrow}_{\boldsymbol{s}_i} &= \boldsymbol{G}\boldsymbol{V}^{\downarrow}_{\boldsymbol{s}_{i-1}}\boldsymbol{G}^{\mathrm{T}} + \boldsymbol{e}\boldsymbol{e}^{\mathrm{T}}\overleftarrow{v}_{x_i}. && \text{(A.14)} \end{aligned}$$

The message $n_{\boldsymbol{s}_i\to f_{G_{i+1}}}(\boldsymbol{s}_i)$ is calculated from (A.6) to be

$$\begin{aligned} n_{\boldsymbol{s}_i\to f_{G_{i+1}}}(\boldsymbol{s}_i) &= m_{f_{G_i}\to \boldsymbol{s}_i}(\boldsymbol{s}_i) m_{f_{r_i}\to \boldsymbol{s}_i}(\boldsymbol{s}_i) && \text{(A.15)}\\ &\propto \exp\left\{-\frac{1}{2}(\boldsymbol{s}_i - \boldsymbol{m}^{\downarrow}_{\boldsymbol{s}_i})^{\mathrm{T}} \boldsymbol{V}^{\downarrow -1}_{\boldsymbol{s}_i}(\boldsymbol{s}_i - \boldsymbol{m}^{\downarrow}_{\boldsymbol{s}_i})\right\} \end{aligned}$$

where

$$\boldsymbol{m}^{\downarrow}_{\boldsymbol{s}_i} = \boldsymbol{m}^{\Downarrow}_{\boldsymbol{s}_i} + \tfrac{1}{\sigma^2 + \boldsymbol{h}^{\mathrm{T}}\boldsymbol{V}^{\Downarrow}_{\boldsymbol{s}_i}\boldsymbol{h}}(r_i - \boldsymbol{h}^{\mathrm{T}}\boldsymbol{m}_i{}^{\Downarrow})\boldsymbol{V}^{\Downarrow}_{\boldsymbol{s}_i}\boldsymbol{h} \tag{A.16}$$

$$\boldsymbol{V}^{\downarrow}_{\boldsymbol{s}_i} = \boldsymbol{V}^{\Downarrow}_{\boldsymbol{s}_i} - \tfrac{1}{\sigma^2 + \boldsymbol{h}^{\mathrm{T}}\boldsymbol{V}^{\Downarrow}_{\boldsymbol{s}_i}\boldsymbol{h}}\boldsymbol{V}^{\Downarrow}_{\boldsymbol{s}_i}\boldsymbol{h}\boldsymbol{h}^{\mathrm{T}}\boldsymbol{V}^{\Downarrow}_{\boldsymbol{s}_i} \tag{A.17}$$

and $m_{f_{r_i}\to \boldsymbol{s}_i}(\boldsymbol{s}_i) = f_{r_i}(r_i, \boldsymbol{s}_i)$.

Equalization–Upward Messages

With the message from variable node $\boldsymbol{s}_{i+1}$ to factor node $f_{G_{i+1}}$ being of the form $n_{\boldsymbol{s}_{i+1}\to f_{G_{i+1}}}(\boldsymbol{s}_{i+1}) \propto \mathcal{N}(\boldsymbol{s}_{i+1}; \boldsymbol{m}^{\uparrow}_{\boldsymbol{s}_{i+1}}, \boldsymbol{V}^{\uparrow}_{\boldsymbol{s}_{i+1}})$, the message $m_{f_{G_{i+1}}\to \boldsymbol{s}_i}(\boldsymbol{s}_i)$ from $f_{G_{i+1}}$ to $\boldsymbol{s}_i$ is obtained as

$$\begin{aligned} &m_{f_{G_{i+1}}\to \boldsymbol{s}_i}(\boldsymbol{s}_i) \propto \\ &\quad \exp\left\{-\tfrac{1}{2}(\boldsymbol{G}\boldsymbol{s}_i - \boldsymbol{m}^{\Uparrow}_{\boldsymbol{s}_i})^{\mathrm{T}} \boldsymbol{V}^{\Uparrow -1}_{\boldsymbol{s}_i}(\boldsymbol{G}\boldsymbol{s}_i - \boldsymbol{m}^{\Uparrow}_{\boldsymbol{s}_i})\right\} && \text{(A.18)} \end{aligned}$$

with

$$\begin{aligned} \boldsymbol{V}^{\Uparrow -1}_{\boldsymbol{s}_i}\boldsymbol{m}^{\Uparrow}_{\boldsymbol{s}_i} &= -\frac{\boldsymbol{V}^{\uparrow -1}_{\boldsymbol{s}_{i+1}}\boldsymbol{e}(\boldsymbol{e}^{\mathrm{T}}\boldsymbol{V}^{\uparrow -1}_{\boldsymbol{s}_{i+1}}\boldsymbol{m}^{\uparrow}_{\boldsymbol{s}_{i+1}} + \overleftarrow{v}^{-1}_{x_{i+1}}\overleftarrow{m}_{x_{i+1}})}{\overleftarrow{v}^{-1}_{x_{i+1}} + \boldsymbol{e}^{\mathrm{T}}\boldsymbol{V}^{\uparrow -1}_{\boldsymbol{s}_{i+1}}\boldsymbol{e}} \\ &\quad + \boldsymbol{V}^{\uparrow -1}_{\boldsymbol{s}_{i+1}}\boldsymbol{m}^{\uparrow}_{\boldsymbol{s}_{i+1}} && \text{(A.19)}\\ \boldsymbol{V}^{\Uparrow -1}_{\boldsymbol{s}_i} &= \boldsymbol{V}^{\uparrow -1}_{\boldsymbol{s}_{i+1}} - \frac{\boldsymbol{V}^{\uparrow -1}_{\boldsymbol{s}_{i+1}}\boldsymbol{e}\boldsymbol{e}^{\mathrm{T}}\boldsymbol{V}^{\uparrow -1}_{\boldsymbol{s}_{i+1}}}{\overleftarrow{v}^{-1}_{x_{i+1}} + \boldsymbol{e}^{\mathrm{T}}\boldsymbol{V}^{\uparrow -1}_{\boldsymbol{s}_{i+1}}\boldsymbol{e}}. && \text{(A.20)} \end{aligned}$$

As a consequence, the message $n_{\boldsymbol{s}_i \to f_{G_i}}(\boldsymbol{s}_i)$ reads

$$n_{\boldsymbol{s}_i \to f_{G_i}}(\boldsymbol{s}_i) \propto \exp\left\{-\frac{1}{2}(\boldsymbol{s}_i - \boldsymbol{m}^{\uparrow}_{\boldsymbol{s}_i})^{\mathrm{T}} \boldsymbol{V}^{\uparrow -1}_{\boldsymbol{s}_i}(\boldsymbol{s}_i - \boldsymbol{m}^{\uparrow}_{\boldsymbol{s}_i})\right\} \tag{A.21}$$

where

$$\boldsymbol{V}^{\uparrow -1}_{\boldsymbol{s}_i}\boldsymbol{m}^{\uparrow}_{\boldsymbol{s}_i} = \boldsymbol{G}^{\mathrm{T}}\boldsymbol{V}^{\Uparrow -1}_{\boldsymbol{s}_i}\boldsymbol{m}^{\Uparrow}_{\boldsymbol{s}_i} + \frac{\boldsymbol{h}r_i}{\sigma^2} \tag{A.22}$$

$$\boldsymbol{V}^{\uparrow -1}_{\boldsymbol{s}_i} = \boldsymbol{G}^{\mathrm{T}}\boldsymbol{V}^{\Uparrow -1}_{\boldsymbol{s}_i}\boldsymbol{G} + \frac{\boldsymbol{h}\boldsymbol{h}^{\mathrm{T}}}{\sigma^2}. \tag{A.23}$$

Equalization–Output Messages

The message $m^{\mathrm{BP}}_{f_{G_i} \to x_i}(x_i)$ reads

$$m^{\mathrm{BP}}_{f_{G_i} \to x_i}(x_i) \propto \exp\left\{-\frac{(x_i - \vec{m}_{x_i})^2}{2\vec{v}_{x_i}}\right\} \tag{A.24}$$

with

$$\begin{aligned}\vec{m}_{x_i} &= \boldsymbol{e}^{\mathrm{T}}\boldsymbol{m}^{\uparrow}_{\boldsymbol{s}_i} + \boldsymbol{e}^{\mathrm{T}}\boldsymbol{V}^{\uparrow}_{\boldsymbol{s}_i}\boldsymbol{G}'' \left[\boldsymbol{G}'\boldsymbol{V}^{\downarrow}_{\boldsymbol{s}_{i-1}}\boldsymbol{G}'^{\mathrm{T}} + \boldsymbol{G}''^{\mathrm{T}}\boldsymbol{V}^{\uparrow}_{\boldsymbol{s}_i}\boldsymbol{G}''\right]^{-1} \\ &\quad \times \left[\boldsymbol{G}'\boldsymbol{m}^{\downarrow}_{\boldsymbol{s}_{i-1}} - \boldsymbol{G}''^{\mathrm{T}}\boldsymbol{m}^{\uparrow}_{\boldsymbol{s}_i}\right] \qquad \text{(A.25)} \\ \vec{v}_{x_i} &= \boldsymbol{e}^{\mathrm{T}}\boldsymbol{V}^{\uparrow}_{\boldsymbol{s}_i}\boldsymbol{e} - \boldsymbol{e}^{\mathrm{T}}\boldsymbol{V}^{\uparrow}_{\boldsymbol{s}_i}\boldsymbol{G}'' \left[\boldsymbol{G}'\boldsymbol{V}^{\downarrow}_{\boldsymbol{s}_{i-1}}\boldsymbol{G}'^{\mathrm{T}} + \boldsymbol{G}''^{\mathrm{T}}\boldsymbol{V}^{\uparrow}_{\boldsymbol{s}_i}\boldsymbol{G}''\right]^{-1} \\ &\quad \times \boldsymbol{G}''^{\mathrm{T}}\boldsymbol{V}^{\uparrow}_{\boldsymbol{s}_i}\boldsymbol{e}. \qquad \text{(A.26)}\end{aligned}$$

Because both messages $m^{\mathrm{BP}}_{f_{G_i} \to x_i}(x_i)$ and $n_{x_i \to f_{G_i}}(x_i)$ in (A.9) are Gaussian, we obtain from (A.5)

$$m_{f_{G_i} \to x_i}(x_i) = m^{\mathrm{BP}}_{f_{G_i} \to x_i}(x_i). \tag{A.27}$$

Decoding

Decoding is performed by using the BCJR algorithm, which is an instance of BP [10]. After completion of the forward/backward processing the BCJR decoder returns the messages $m^{\mathrm{BP}}_{f_{M_i} \to x_i}(x_i)$, $\forall i$. We remark that any other code that can be decoded using a BP-based algorithm, e.g. a turbo- or LDPC code, could be used instead within the proposed BP-EP framework.

4.2 Scheduling of the Messages

After initializing $m_{f_{M_i} \to x_i}(x_i)$, $\forall i$, the messages $m_{f_{G_i} \to \boldsymbol{s}_i}(\boldsymbol{s}_i)$ and $n_{\boldsymbol{s}_i \to f_{G_{i+1}}}(\boldsymbol{s}_i)$, with i ranging from 1 to $N + L - 1$, are calculated in the downward recursion using (A.12) and (A.15) respectively. Likewise, $m_{f_{G_{i+1}} \to \boldsymbol{s}_i}(\boldsymbol{s}_i)$ and $n_{\boldsymbol{s}_i \to f_{G_i}}(\boldsymbol{s}_i)$, with i ranging from $N + L - 1$ to 1, are obtained from (A.18) and (A.21), respectively, in the upward recursion[3]. Equations (A.24) and (A.27) are used to get the messages $m_{f_{G_i} \to x_i}(x_i)$, $\forall i$, which are then passed to the BCJR decoder. The decoder outputs the messages $m^{\text{BP}}_{f_{M_i} \to x_i}(x_i)$, $\forall i$, and finally $m_{f_{M_i} \to x_i}(x_i)$, $\forall i$, are updated via (A.9).

4.3 Reduction of Complexity

Since they are performed in the update of each symbol, the two matrix inversions in (A.25) and (A.26) make up a significant part of the computational complexity of the BP-EP-based algorithm. To reduce the complexity, the approach proposed in [5] can be adopted. We calculate the belief of variable $\boldsymbol{s}_i$, i.e. $b(\boldsymbol{s}_i) \propto m_{f_{G_i} \to \boldsymbol{s}_i}(\boldsymbol{s}_i)\, n_{\boldsymbol{s}_i \to f_{G_i}}(\boldsymbol{s}_i) \propto \mathcal{N}(\boldsymbol{s}_i; \boldsymbol{m}_i, \boldsymbol{V}_i)$, where

$$\begin{aligned} \boldsymbol{V}_i &= (\boldsymbol{V}^{\Downarrow}_{\boldsymbol{s}_i}{}^{-1} + \boldsymbol{V}^{\uparrow}_{\boldsymbol{s}_i}{}^{-1})^{-1} \\ \boldsymbol{m}_i &= \boldsymbol{V}_i(\boldsymbol{V}^{\Downarrow}_{\boldsymbol{s}_i}{}^{-1}\boldsymbol{m}^{\Downarrow}_{\boldsymbol{s}_i} + \boldsymbol{V}^{\uparrow}_{\boldsymbol{s}_i}{}^{-1}\boldsymbol{m}^{\uparrow}_{\boldsymbol{s}_i}). \end{aligned} \tag{A.28}$$

According to the deterministic relationship given in (A.3), the messages from factor node $f_{G_{i-l}}$ to variable nodes x_{i-l}, $l = 0, 1, \ldots, L-1$, are obtained as

$$\begin{aligned} \widetilde{m}^{\text{BP}}_{f_{G_{i-l}} \to x_{i-l}}(x_{i-l}) &\propto \frac{\int b(\boldsymbol{s}_i)\delta\left(s_{i,L-l} - x_{i-l}\right) d\boldsymbol{s}_i}{n_{x_{i-l} \to f_{G_{i-l}}}(x_{i-l})} \\ &\propto \exp\left\{ -\frac{(x_{i-l} - \vec{m}_{x_{i-l}})^2}{2\vec{v}_{x_{i-l}}} \right\} \end{aligned} \tag{A.29}$$

where

$$\vec{v}_{x_{i-l}} = (V^{-1}_{i,L-l} - \overleftarrow{m}^{-1}_{x_{i-l}})^{-1} \tag{A.30}$$

$$\vec{m}_{x_{i-l}} = \vec{v}_{x_{i-l}}(V^{-1}_{i,L-l} m_{i,L-l} - \overleftarrow{v}^{-1}_{x_{i-l}} \overleftarrow{m}_{x_{i-l}}) \tag{A.31}$$

with $s_{i,L-l}$ and $m_{i,L-l}$ representing the $(L-l)$th element of vector $\boldsymbol{s}_i$ and $\boldsymbol{m}_i$ respectively, and $V_{i,L-l}$ denoting the $(L-l)$th diagonal entry of the matrix $\boldsymbol{V}_i$. Using this, only two matrix inversions are needed in the update of each block of L symbols [5]. Replacing the messages in (A.24) by those in (A.29) reduces the complexity order from $\mathcal{O}(L^3)$ to $\mathcal{O}(L^2)$. The proof of the equivalence of these messages is provided in Appendix A.

[3]Note that these recursions coincide with those of a Kalman smoother [7].

5 Simulation Results

We evaluate the performance of the communication system described in Section 2 by means of Monte Carlo simulations. Two different lengths of information bit vectors are considered: $K = 32768$ (long) and $K = 8192$ (short). The information bits are coded using a $1/2$ rate convolutional code $(23, 35)_8$. The vector of channel weights is set to $\boldsymbol{h} = [0.227\ 0.460\ 0.668\ 0.460\ 0.227]^{\mathrm{T}}$, which corresponds to a severe time-dispersive (5-tap) channel [11].

Fig. A.2 and Fig. A.3 depict the performance of the investigated algorithms: the proposed algorithm (BP-EP), the algorithm presented in [5] (GABP), the algorithm implementing MAP equalization (BP) (reproduced from Fig. 5 in [5]), and a receiver operating in an ISI-free channel (AWGN). In Fig. A.2, the BER performance after 30 receiver iterations is shown when the SNR ranges from 4dB to 6dB. We observe that BP-EP significantly outperforms GABP. It also performs close to BP, the loss expressed in terms of the SNR value where the threshold effect occurs being about 0.3 dB. In Fig. A.3, the BER performance at 5.5dB SNR of BP-EP and GABP is depicted as a function of the iteration index. We observe that BP-EP converges much faster and is less sensitive to shorter codeword lengths than GABP.

Both BP-EP and GABP receivers exhibit the same complexity order per symbol. They differ only in their respective equalization parts, both having $\mathcal{O}(L^2)$ order of complexity per symbol. The former algorithm approximates the messages from f_{M_i} to x_i based on the messages passed by both the decoder and the equalizer, while the latter only makes use of the messages passed by the decoder for doing this. The observed superior performance indicates that the BP-EP approximation is better.

6 Appendix A: Proof of the Equivalence Between (A.24) and (A.29)

The proof is by induction. Thus, we merely need to show the equivalence for $l = 0$ and $l = 1$.

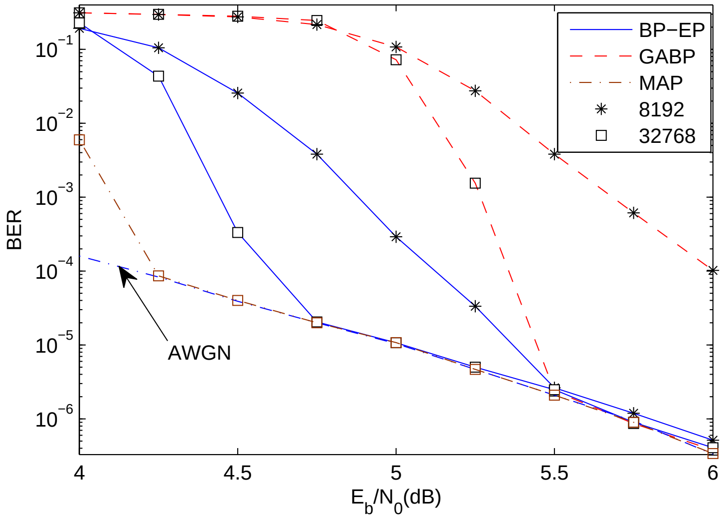


Fig. A.2: BER performance versus E_b/N_0 of the investigated receivers.

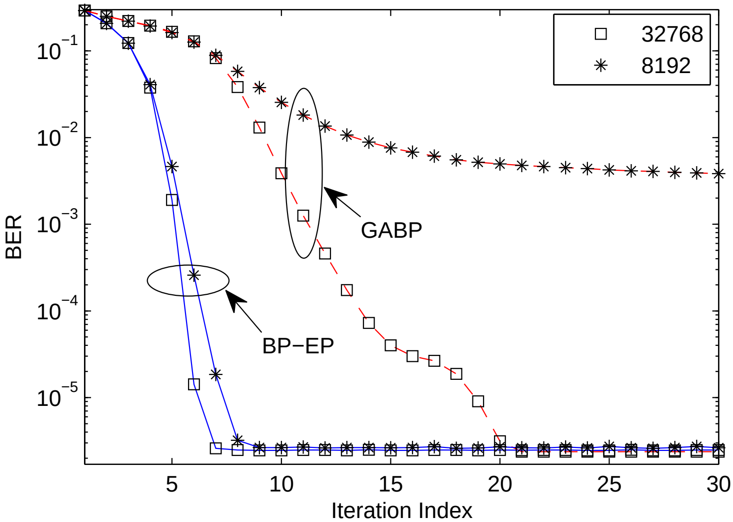


Fig. A.3: BER performance versus iteration index of the investigated receivers.

For $l = 0$ we have according to (A.29)

$$
\begin{aligned}
\widetilde{m}^{\mathrm{BP}}_{f_{G_i}\to x_i}(x_i) &\propto \int b(\mathbf{s}_i)\delta\left(s_{i,L}-x_i\right) d\mathbf{s}_i / n_{x_i\to f_{G_i}}(x_i) \\
&\propto \int n_{\mathbf{s}_{i-1}\to f_{G_i}}\left(\mathbf{s}_{i-1}\right) n_{x_i\to f_{G_i}}(x_i) f_{G_i}\left(\mathbf{s}_i,\mathbf{s}_{i-1},x_i\right) d\mathbf{s}_{i-1}dx_i \\
&\quad \times n_{\mathbf{s}_i\to f_{G_i}}\left(\mathbf{s}_i\right)\delta\left(s_{i,L}-x_i\right) d\mathbf{s}_i / n_{x_i\to f_{G_i}}(x_i) \\
&= \int n_{\mathbf{s}_{i-1}\to f_{G_i}}\left(\mathbf{s}_{i-1}\right) n_{\mathbf{s}_i\to f_{G_i}}\left(\mathbf{s}_i\right) f_{G_i}\left(\mathbf{s}_i,\mathbf{s}_{i-1},x_i\right) d\mathbf{s}_{i-1}d\mathbf{s}_i \\
&= m^{\mathrm{BP}}_{f_{G_i}\to x_i}(x_i).
\end{aligned}
\tag{A.32}
$$

For $l = 1$ we first obtain from the BP rule (A.4)

$$\int b(\boldsymbol{s}_i)\delta\left(s_{i,L-1} - x_{i-1}\right) d\boldsymbol{s}_i \\ \propto \int b(\boldsymbol{s}_{i-1})\delta\left(s_{i-1,L} - x_{i-1}\right) d\boldsymbol{s}_{i-1}. \tag{A.33}$$

Then, using (A.32) and (A.33) yields

$$\begin{aligned} \widetilde{m}^{\mathrm{BP}}_{f_{G_{i-1}} \to x_{i-1}}(x_{i-1}) &\propto \frac{\int b(\boldsymbol{s}_i)\delta\left(s_{i,L-1} - x_{i-1}\right) d\boldsymbol{s}_i}{n_{x_{i-1} \to f_{G_{i-1}}}(x_{i-1})} \\ &\propto \frac{\int b(\boldsymbol{s}_{i-1})\delta\left(s_{i-1,L} - x_{i-1}\right) d\boldsymbol{s}_{i-1}}{n_{x_{i-1} \to f_{G_{i-1}}}(x_{i-1})} \\ &\propto m^{\mathrm{BP}}_{f_{G_{i-1}} \to x_{i-1}}(x_{i-1}). \end{aligned}$$

References

[1] R. Koetter, A. C. Singer, and M. Tücher, "Turbo equalization," *IEEE Signal Processing Mag.*, vol. 21, pp. 67–80, Jan. 2004.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo codes," in *Proc. IEEE International Conference on Communications, (ICC'93)*, May 1993, pp. 1064–1070.

[3] R. McEliece, D. MacKay, and J. Cheng, "Turbo decoding as an instance of Pearl's belief "propagation" algorithms," *IEEE J. Select. Areas Commun.*, vol. 16, no. 2, pp. 140–152, Feb. 1998.

[4] 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.

[5] Q. Guo and L. Ping, "LMMSE turbo equalization based on factor graphs," *IEEE J. Select. Areas Commun.*, vol. 26, no. 2, pp. 311–319, Feb. 2008.

[6] X. Wang and H. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, pp. 1046–1061, Jul. 1999.

[7] 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. Communication, Control, and Computing*, Sep. 2006, pp. 27–29.

[8] T. Heskes, M. Opper, W. Wiegerinck, O. Winther, and O. Zoeter, "Approximate inference techniques with expectation constraints," *Journal of Statistical Mechanics – Theory and Experiment*, Sep. 2005.

[9] T. P. Minka, "Expectation propagation for approximate Bayesian inference," in *Proceedings of the 17th Conf. on Uncertainty in Artificial Intelligence*, 2001, pp. 362–369.

[10] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[11] J. Proakis, *Digital Communications*, 4th ed. McGraw-Hill, Aug. 2000.

Paper B

Turbo-Equalization Using Partial Gaussian Approximation

Chuanzong Zhang, Zhongyong Wang,
Carles Navarro Manchón, Peng Sun, Qinghua Guo
and Bernard Henri Fleury

The paper has been accepted by the
IEEE Signal Processing Letters, 2016.

The layout has been revised.

Abstract

This paper deals with turbo-equalization for coded data transmission over intersymbol interference (ISI) channels. We propose a message-passing algorithm that uses the expectation-propagation rule to convert messages passed from the demodulator-decoder to the equalizer and computes messages returned by the equalizer by using a partial Gaussian approximation (PGA). We exploit the specific structure of the ISI channel model to compute the latter messages from the beliefs obtained using a Kalman smoother/equalizer. Doing so leads to a significant complexity reduction compared to the initial PGA implementation. Results from Monte Carlo simulations show that the proposed approach leads to a significant performance improvement compared to state-of-the-art turbo-equalizers and allows for trading performance with complexity.

1 Introduction

Historically, turbo equalization of coded data transmission across a known inter-symbol interference (ISI) channel found its inspiration from turbo-decoding of turbo-codes, see [1] and references therein. Since its introduction turbo equalization has prevailed over more traditional equalization techniques available at that time due to its tremendous performance gain. Turbo-equalization is a collective name for joint data decoding and channel equalization algorithms that pass messages iteratively along the edges of a factor graph representing the probabilistic model of the considered transmission system. The most prominent message-passing algorithm – inherited from turbo-decoding of turbo-codes – is the sum-product algorithm [2, 3], which is also known as belief propagation (BP) [4].

Different factor graphs representing the ISI channel can be drawn, which lead to different message-passing algorithms for equalization, see e.g. [3, 5, 6]. In this letter, we use a tree graph [3] that explicitly represents the channel state evolution. Applying BP on this graph yields the BCJR algorithm [1], [2], the complexity of which scales exponentially with the modulation order and the channel memory. Proposed solutions that circumvent this complexity problem convert the discrete messages returned by the demodulator-decoder into Gaussian functions that are passed as messages to the equalizer [7–9]. Due to the linearity of the channel and the Gaussianity of additive noise, the equalizer processes Gaussian messages, i.e. it coincides with a Kalman smoother. The discrete-to-Gaussian conversion can be done in two ways: either directly by matching the first and second moments of the discrete messages [8], or by using the formal rule of expectation propagation (EP) [10], [7], [9]. Numerical studies have shown that the latter conversion leads to better performance [7], [9].

Inspired by the partial Gaussian approximation (PGA) proposed in [11] we modify the messages returned by the Kalman smoother/equalizer and passed to the demodulator-decoder in [9]. To equalize each channel symbol the new equalizer combines discrete messages from the demodulator-decoder for the symbols strongly interfering with said symbol and Gaussian-converted messages for the weakly interfering symbols. In this way the equalizer enforces the modulation constellation of strong interfering symbols, unlike the Kalman smoother that uses Gaussian-converted messages for all symbols. The reported simulation results show that doing so leads to a significant performance improvement compared to the turbo-equalizers in [8], [11] and [9]. Our turbo-equalizer allows for trading complexity with performance by varying the set of symbols that are regarded as "strong" interferers.

Our turbo-equalizer differs from the PGA-based one in [11] in two respects. First, in the former the Gaussian conversion of the discrete messages returned by the demodulator-decoder is done by using the formal EP rule, while direct conversion is employed in the latter. Second, due to the particular structure of the ISI channel model, the messages returned by our equalizer can be obtained in a simple way from beliefs computed with the Kalman equalizer. This leads to a significant complexity reduction compared to the turbo-equalizer in [11].

Notation- For a natural number N, we write $[N] = \{1, \ldots, N\}$. Boldface lowercase and uppercase letters denote vectors and matrices, respectively. The identity matrix of size M is represented by $\boldsymbol{I}_M$. Superscript $(\cdot)^{\mathrm{T}}$ designates transposition of a vector or matrix. We write $\mathrm{N}(\boldsymbol{x}; \boldsymbol{m}, \boldsymbol{V})$ for the pdf of a multivariate Gaussian distribution with mean vector $\boldsymbol{m}$ and covariance matrix $\boldsymbol{V}$. Depending on the context $\delta(\cdot)$ denotes either the Dirac delta function or the Kronecker delta. The relation $f(\cdot) = cg(\cdot)$ for some positive constant c is written as $f(\cdot) \propto g(\cdot)$. The notations $\sum_{\boldsymbol{x} \setminus \boldsymbol{y}} f(\boldsymbol{x})$ and $\int f(\boldsymbol{x})\, \mathrm{d}(\boldsymbol{x} \setminus \boldsymbol{y})$ denote respectively the partial summation and partial integration of the function $f(\boldsymbol{x})$ with respect to all entries of the vector $\boldsymbol{x}$ except those entries common to $\boldsymbol{x}$ and $\boldsymbol{y}$.

2 System Model

The vector $\boldsymbol{b} = [b_1, \ldots, b_K]^{\mathrm{T}}$ of information bits is encoded and interleaved, yielding the codeword $\boldsymbol{c} = [\boldsymbol{c}_1^{\mathrm{T}}, \ldots, \boldsymbol{c}_N^{\mathrm{T}}]^{\mathrm{T}}$ with $\boldsymbol{c}_i = [c_i^1, \cdots, c_i^Q]^{\mathrm{T}}$. The coded bits are then mapped onto a 2^Q-order modulation alphabet $\mathcal{X} \subseteq \mathbb{R}$ [1], resulting in the vector of symbols $\boldsymbol{x} = [x_1, \ldots, x_N]^{\mathrm{T}} \in \mathcal{X}^N$. These symbols are transmitted over a linear, time-invariant, frequency-selective channel corrupted by additive white Gaussian noise (AWGN). The received vector $\boldsymbol{r} =$

[1]For simplicity we consider a real baseband model. The extension to a complex model is straightforward.

$[r_1, \ldots, r_{N+L-1}]^{\mathrm{T}}$ has entries

$$r_i = \sum_{l=0}^{L-1} h_l x_{i-l} + n_i = \boldsymbol{h}^{\mathrm{T}} \boldsymbol{s}_i + n_i, \quad i \in [N+L-1] \tag{B.1}$$

where $\boldsymbol{s}_i = [x_{i-L+1}, \ldots, x_i]^{\mathrm{T}}$ with $x_i = 0$ for $i < 1$ and $i > N$, $\boldsymbol{h} = [h_{L-1}, \ldots, h_0]^{\mathrm{T}}$ represents the channel impulse response, and $\boldsymbol{n} = [n_1, \ldots, n_{N+L-1}]^{\mathrm{T}}$ is a white noise vector with component variance σ^2.

2.1 Probabilistic Model and Factor Graph

The posterior probability mass function (pmf) of vectors $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{x}$ and $\boldsymbol{s}$ given the received signal $\boldsymbol{r}$ reads

$$\begin{aligned} p\left(\boldsymbol{b}, \boldsymbol{c}, \boldsymbol{x}, \boldsymbol{s} | \boldsymbol{r}\right) \propto & \prod_{k=1}^{K} f_{\mathrm{B}_k}\left(b_k\right) \times f_{\mathrm{C}}\left(\boldsymbol{c}, \boldsymbol{b}\right) \\ & \times \prod_{i=1}^{N} f_{\mathrm{O}_i}\left(r_i, \boldsymbol{s}_i\right) f_{\mathrm{T}_i}\left(\boldsymbol{s}_i, \boldsymbol{s}_{i-1}, x_i\right) f_{\mathrm{M}_i}\left(x_i, \boldsymbol{c}_i\right) \\ & \times \prod_{i=N+1}^{N+L-1} f_{\mathrm{O}_i}\left(r_i, \boldsymbol{s}_i\right) f_{\mathrm{T}_i}\left(\boldsymbol{s}_i, \boldsymbol{s}_{i-1}, 0\right) \end{aligned} \tag{B.2}$$

where $f_{\mathrm{B}_k}(b_k)$ is the uniform prior pmf of the kth information bit, $f_{\mathrm{C}}(\boldsymbol{c}, \boldsymbol{b})$ stands for the coding and interleaving constraints, $f_{\mathrm{O}_i}(r_i, \boldsymbol{s}_i) \triangleq p(r_i|\boldsymbol{s}_i) = \mathrm{N}(r_i; \boldsymbol{h}^{\mathrm{T}}\boldsymbol{s}_i, \sigma^2)$ denotes the likelihood of $\boldsymbol{s}_i$, and $f_{\mathrm{M}_i}(x_i, \boldsymbol{c}_i)$ represents the modulation mapping. Finally, $f_{\mathrm{T}_i}\left(\boldsymbol{s}_i, \boldsymbol{s}_{i-1}, x_i\right)$ expresses the deterministic relationship between $\boldsymbol{s}_i$, $\boldsymbol{s}_{i-1}$ and x_i, i.e.,

$$f_{\mathrm{T}_i}\left(\boldsymbol{s}_i, \boldsymbol{s}_{i-1}, x_i\right) = \delta(\boldsymbol{G}\boldsymbol{s}_{i-1} + \boldsymbol{e}x_i - \boldsymbol{s}_i) \tag{B.3}$$

with the $L \times L$ matrix $\boldsymbol{G} = \left[\begin{array}{cc} \boldsymbol{0} & \boldsymbol{I}_{L-1}; \ 0 & \boldsymbol{0}^{\mathrm{T}} \end{array}\right]$, $\boldsymbol{e} = \left[\begin{array}{c} \boldsymbol{0}; \ 1 \end{array}\right]$ and $\boldsymbol{0}$ being a zero column vector of length $L-1$.

Fig. B.1 depicts the factor graph [2] representing the factorization of the posterior pmf in (B.2). The factorization and its graph will be used as the baseline for the derivation of the turbo-equalizer described in Section 3. To ease the subsequent discussions we identify two subgraphs. The channel subgraph includes the nodes of the channel symbols x_i, $i \in [N]$ and all factor nodes, variable nodes and edges "to the left" of these symbol nodes. The transmitter subgraph includes the channel symbol nodes and all factor nodes, variable nodes, and edges "to the right" of these symbol nodes.

3 Design of the Iterative Receiver

In a nutshell, we obtain the new turbo-equalizer by replacing the messages passed from the Kalman smoother/equalizer to the demodulator-decoder

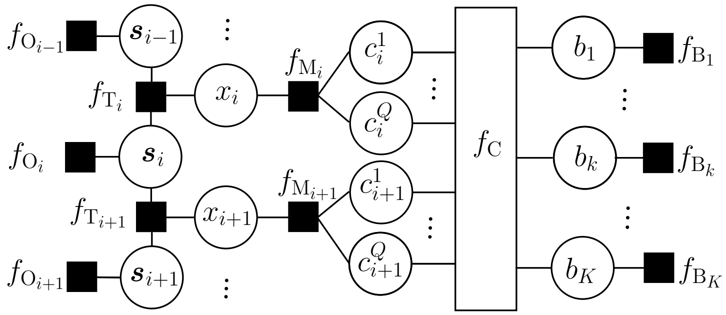


Fig. B.1: Factor graph representing the probabilistic model (B.2).

(from nodes f_{T_i} to nodes x_i, $i \in [N]$) in the turbo-equalizer in [9] with messages computed using the PGA approach in [11]. The latter message can be computed at low complexity from beliefs calculated in the Kalman smoother. The next two subsections describe the messages computed in the new turbo-equalizer. The last subsection sketches the scheduling of these messages.

3.1 Equalization and Demodulation-decoding

Equalization and demodulation-decoding are implemented by passing messages along the edges of the channel subgraph and the transmitter subgraph respectively. Unless otherwise stated, these messages are computed using the BP rule [4].

Demodulation-decoding

The variables in the transmitter subgraph are discrete and so are the computed messages and beliefs. Decoding of the convolution code is done using the BCJR algorithm, an instance of BP. The messages from the modulator nodes to the channel symbol nodes are of the form $m_{f_{\mathrm{M}_i} \to x_i}(x_i) \propto \sum_{x \in \mathcal{X}} \beta_x \delta\left(x_i - x\right)$ with $\beta_x \geq 0$, $x_i \in \mathcal{X}$, $i \in [N]$.

Equalization

The latent variables $\boldsymbol{s}_i$, $i \in [N]$ in the channel subgraph are approximated as Gaussian variables. Since the channel is linear and noise is additive and Gaussian, the messages and beliefs are Gaussian functions. We write for the belief of node $\boldsymbol{s}_i$ ($i \in [N]$),

$$b^{\mathrm{G}}(\boldsymbol{s}_i) \propto \mathrm{N}(\boldsymbol{s}_i; \boldsymbol{m}_{\boldsymbol{s}_i}, \boldsymbol{V}_{\boldsymbol{s}_i}). \tag{B.4}$$

The computation of this belief is given in [8] and [9, Eq. (28)].

3.2 Messages Exchanged Between the Equalizer and the Demodulator-decoder

Demodulator-decoder (D) $\to$ Equalizer (E)

The EP rule [10] is used to convert the discrete messages $m_{f_{\mathrm{M}_i}\to x_i}(x_i)$, $i \in [N]$ into Gaussian messages [7], [9, Eq. (29)]:

$$\begin{aligned} m^{\mathrm{G}}_{f_{\mathrm{M}_i}\to x_i}(x_i) &= \frac{\mathrm{Proj}_{\mathcal{G}}[m_{f_{\mathrm{M}_i}\to x_i}(x_i) n^{\mathrm{G}}_{x_i\to f_{\mathrm{M}_i}}(x_i)]}{n^{\mathrm{G}}_{x_i\to f_{\mathrm{M}_i}}(x_i)} \\ &\propto \mathrm{N}(x_i; m_{x_i}, v_{x_i}), \quad i \in [N]. \end{aligned} \tag{B.5}$$

For a pdf $b(z)$, $\mathrm{Proj}_{\mathcal{G}}[b(z)] = \arg\min_{b'(z)\in\mathcal{G}} \mathrm{D}\left(b(z)||b'(z)\right)$, with $\mathrm{D}(\cdot||\cdot)$ denoting the Kullback-Leibler divergence and $\mathcal{G}$ being the family of Gaussian pdfs. The parameters m_{x_i} and v_{x_i} in (B.5) are given by [9, Eq. (10) & (11)]. With this conversion, Gaussian messages $n^{\mathrm{G}}_{x_i\to f_{\mathrm{T}_i}}(x_i) = m^{\mathrm{G}}_{f_{\mathrm{M}_i}\to x_i}(x_i)$, $i \in [N]$ are passed to the equalizer.

E $\to$ D

This is where the new turbo-equalizer differs from the one described in [7], [9].

In [7], [9] the Gaussian messages from f_{T_i} to x_i, $i \in [N]$ are converted into discrete messages according to [2]

$$m_{f_{\mathrm{T}_i}\to x_i}(x_i) \propto m^{\mathrm{G}}_{f_{\mathrm{T}_i}\to x_i}(x_i) \quad , \quad i \in [N]. \tag{B.6}$$

The messages $n_{x_i\to f_{\mathrm{M}_i}}(x_i) = m_{f_{\mathrm{T}_i}\to x_i}(x_i)$, $i \in [N]$ are then passed to the demodulator-decoder.

Consider a specific symbol x_i ($i \in [N]$). Clearly the computation of $m_{f_{\mathrm{T}_i}\to x_i}(x_i)$ using (B.6) makes use of the Gaussian approximation of the messages from the other symbols, i.e. $n^{\mathrm{G}}_{x_j\to f_{\mathrm{T}_j}}(x_j)$, $j \in [N]\setminus\{i\}$ by the conversion (B.5). The idea is to use the original discrete messages rather than their Gaussian approximation for a selected subset of channel symbols which significantly interfere with x_i. It is inspired from the PGA proposed in [11].

First we identify those channel symbols "significantly" interfering with symbol x_i. Let $q_k = \sum_{l=0}^{L-1} h_l h_{l+k}$, $k \in \mathbb{Z}$ with $h_l = 0$ whenever $l \in \mathbb{Z}\setminus\{0,\ldots,L-1\}$ be the autocorrelation function of the channel impulse response. Define $\mathbb{K}_\rho = \{k \in \{-(L-1),\ldots,L-1\} : |q_k| > \rho q_0\}$ the set of lag indices at which the magnitude of the autocorrelation function is larger than ρq_0, $\rho \in [0,1)$. Then $\mathbb{I}^{\mathrm{D}}_i = \{i+k : k \in \mathbb{K}_\rho\} \subseteq \mathbb{I}_i = \{i-(L-1),\ldots,i+L-1\}$

[2]Note that (**??**) means that $m_{f_{\mathrm{T}_i}\to x_i}(x_i)$ is proportional to the restriction to $\mathcal{X}$ of $m^{\mathrm{G}}_{f_{\mathrm{T}_i}\to x_i}(x_i)$.

contains the indices of the modulation symbol x_i and those symbols that interfere with x_i at correlation level ρ. We collect these symbols in the M-dimensional vector $\boldsymbol{x}_i^{\mathrm{D}} = [x_j : j \in \mathbb{I}_i^{\mathrm{D}}]^{\mathrm{T}}$, with $M = |\mathbb{K}_\rho|$. We assume that $\bar{k} = \max \mathbb{K}_\rho$ fulfills $1 + 2\bar{k} \leq L$. Then we can readily show that all entries in $\boldsymbol{x}_i^{\mathrm{D}}$ are components of $\boldsymbol{s}_{i'}$ whenever $i + \bar{k} \leq i' \leq i + (L-1) - \bar{k}$. Notice that the assumption on $\bar{k}$ guarantees that $i + \bar{k} \leq i + (L-1) - \bar{k}$.

With the above definitions we can now specify the message from f_{T_i} to x_i:

$$m^{\mathrm{PG}}_{f_{\mathrm{T}_i} \to x_i}(x_i) = \sum_{\boldsymbol{x}_i^{\mathrm{D}} \setminus x_i} \frac{\prod_{\kappa \in \mathbb{I}_i^{\mathrm{D}} \setminus i} n_{x_\kappa \to f_{\mathrm{T}_\kappa}}(x_\kappa)}{\prod_{k \in \mathbb{I}_i^{\mathrm{D}}} n^{\mathrm{G}}_{x_k \to f_{\mathrm{T}_k}}(x_k)} b^{\mathrm{G}}_{i'}(\boldsymbol{x}_i^{\mathrm{D}}) \tag{B.7}$$

where $n_{x_\kappa \to f_{\mathrm{T}_\kappa}}(x_\kappa) = m_{f_{\mathrm{M}_\kappa} \to x_\kappa}(x_\kappa)$ and $b^{\mathrm{G}}_{i'}(\boldsymbol{x}_i^{\mathrm{D}}) = \int b^{\mathrm{G}}(\boldsymbol{s}_{i'})\, \mathrm{d}(\boldsymbol{s}_{i'} \setminus \boldsymbol{x}_i^{\mathrm{D}})$ with $b^{\mathrm{G}}(\boldsymbol{s}_{i'})$ given in (B.4). The latter term is the belief of $\boldsymbol{x}_i^{\mathrm{D}}$ obtained by marginalization of the belief $b^{\mathrm{G}}(\boldsymbol{s}_{i'})$. The index i' in $b^{\mathrm{G}}_{i'}(\boldsymbol{x}_i^{\mathrm{D}})$ indicates that this belief depends on the time instant i', $i + \bar{k} \leq i' \leq i + (L-1) - \bar{k}$. Notice that the selection $i' = i + \bar{k}$ minimizes the time instant ahead of i to wait for computing $m_{f_{\mathrm{T}_i} \to x_i}(x_i)$. The derivation of (B.7) is provided in the appendix.

All Gaussian functions occurring in (B.7) combine as

$$\left[\prod_{\kappa \in \mathbb{I}_i^{\mathrm{D}}} n^{\mathrm{G}}_{x_\kappa \to f_{\mathrm{T}_\kappa}}(x_\kappa) \right]^{-1} b^{\mathrm{G}}_{i'}(\boldsymbol{x}_i^{\mathrm{D}}) \propto \mathrm{N}(\boldsymbol{x}_i^{\mathrm{D}}; \boldsymbol{m}^e_{\boldsymbol{x}_i^{\mathrm{D}}}, \boldsymbol{V}^e_{\boldsymbol{x}_i^{\mathrm{D}}}) \tag{B.8}$$

with

$$\boldsymbol{V}^e_{\boldsymbol{x}_i^{\mathrm{D}}} = \left[(\boldsymbol{P}_{i'} \boldsymbol{V}_{\boldsymbol{s}_{i'}} \boldsymbol{P}_{i'}^{\mathrm{T}})^{-1} - (\boldsymbol{V}_{\boldsymbol{x}_i^{\mathrm{D}}})^{-1} \right]^{-1}$$
$$\boldsymbol{m}^e_{\boldsymbol{x}_i^{\mathrm{D}}} = \boldsymbol{V}^e_{\boldsymbol{x}_i^{\mathrm{D}}} \left[(\boldsymbol{P}_{i'} \boldsymbol{V}_{\boldsymbol{s}_{i'}} \boldsymbol{P}_{i'}^{\mathrm{T}})^{-1} \boldsymbol{P}_{i'} \boldsymbol{m}_{\boldsymbol{s}_{i'}} - (\boldsymbol{V}_{\boldsymbol{x}_i^{\mathrm{D}}})^{-1} \boldsymbol{m}_{\boldsymbol{x}_i^{\mathrm{D}}} \right].$$

In these expressions, the $M \times L$ selection matrix $\boldsymbol{P}_{i'}$ extracts the vector $\boldsymbol{x}_i^{\mathrm{D}}$ from $\boldsymbol{s}_{i'}$, i.e. $\boldsymbol{x}_i^{\mathrm{D}} = \boldsymbol{P}_{i'} \boldsymbol{s}_{i'}$. The entries of the vector $\boldsymbol{m}_{\boldsymbol{x}_i^{\mathrm{D}}}$ and the diagonal entries of the diagonal matrix $\boldsymbol{V}_{\boldsymbol{x}_i^{\mathrm{D}}}$ are the first moments m_{x_κ} and the second central moments v_{x_κ} respectively of the messages $n^{\mathrm{G}}_{x_\kappa \to f_{\mathrm{T}_\kappa}}(x_\kappa)$, $\kappa \in \mathbb{I}_i^{\mathrm{D}}$. Inserting (B.8) into (B.7) yields the PGA-based messages

$$m_{f_{\mathrm{T}_i} \to x_i}(x_i) \propto \sum_{\boldsymbol{x}_i^{\mathrm{D}} \setminus x_i} \mathrm{N}(\boldsymbol{x}_i^{\mathrm{D}}; \boldsymbol{m}^e_{\boldsymbol{x}_i^{\mathrm{D}}}, \boldsymbol{V}^e_{\boldsymbol{x}_i^{\mathrm{D}}}) \prod_{\kappa \in \mathbb{I}_i^{\mathrm{D}} \setminus i} n_{x_\kappa \to f_{\mathrm{T}_\kappa}}(x_\kappa), \qquad i \in [N] \tag{B.9}$$

which replace the messages in (B.6) in the new equalizer.

3.3 Messages Scheduling

The turbo-equalizer implements the following scheduling:

S1: *Initialization:* $n_{x_i \to f_{\mathrm{T}_i}}(x_i) \propto 1$ and $n^{\mathrm{G}}_{x_i \to f_{\mathrm{T}_i}}(x_i) = \mathcal{N}(x_i; 0, 1)$, $i \in [N]$.

S2: *Equalization:* The messages $m^{\mathrm{G}}_{f_{\mathrm{T}_i} \to \boldsymbol{s}_i}(\boldsymbol{s}_i)$ and $n^{\mathrm{G}}_{\boldsymbol{s}_i \to f_{\mathrm{T}_{i+1}}}(\boldsymbol{s}_i)$, $i \in [N+L-1]$ are recursively computed using (12) and (15) respectively in [9]. In parallel, the messages $m^{\mathrm{G}}_{f_{\mathrm{T}_i} \to \boldsymbol{s}_{i-1}}(\boldsymbol{s}_{i-1})$ and $n^{\mathrm{G}}_{\boldsymbol{s}_{i-1} \to f_{\mathrm{T}_{i-1}}}(\boldsymbol{s}_i)$, $i \in \{N+L-1, \ldots, 1\}$ are recursively calculated from (18) and (21) respectively in [9]. Finally, the beliefs $b^{\mathrm{G}}(\boldsymbol{s}_i)$, $i \in [N]$ (see (B.4)) are obtained by (28) in [9].

S3: *$E \to D$:* The messages $m^{\mathrm{PG}}_{f_{\mathrm{T}_i} \to x_i}(x_i)$, $i \in [N]$ are obtained from (B.8) and (B.9).

S4: *Demodulation-decoding:* The messages $m^{\mathrm{PG}}_{f_{\mathrm{T}_i} \to x_i}(x_i)$, $i \in [N]$ are passed to the demodulator. The BCJR algorithm, an instance of BP, is run in the decoder, yielding the discrete messages $n_{x_i \to f_{\mathrm{T}_i}}(x_i) = m_{f_{\mathrm{M}_i} \to x_i}(x_i)$, $i \in [N]$.

S5: *$D \to E$:* The Gaussian messages $n^{\mathrm{G}}_{x_i \to f_{\mathrm{T}_i}}(x_i) = m^{\mathrm{G}}_{f_{\mathrm{M}_i} \to x_i}(x_i)$, $i \in [N]$ are updated using (B.5).

Steps S2–S5 constitute an iteration that is repeated until a maximum number of iterations is reached.

4 Analysis, Performance and Complexity

4.1 Comparison with Existing Turbo-equalizers

We compare the performance of the new turbo-equalizer (we denote it as BP-EP-PGA) with that of four other turbo-equalizers by means of Monte Carlo simulations: (a) BP: the MAP-based turbo equalizer obtained by applying solely BP, which runs the BCJR algorithm for both equalization and decoding [2]; (b) BP-EP: the combined BP-EP algorithm in [9]; (c) BP-PGA: an implementation of the PGA algorithm in [11] for the equalization of ISI channels; (d) BP-GA: the LMMSE-based turbo-equalizer, which is equivalent to Gaussian-approximated BP [8]. In our implementation BP-PGA is obtained from BP-EP-PGA by substituting the EP rule (B.5) with a direct Gaussian approximation of the discrete messages from f_{M_i} to x_i, $i \in [N]$. For the BP turbo-equalizer, no conversion of messages between the demodulator-decoder and equalizer is performed. The other four turbo-equalizers solely differ in the types of messages exchanged between the equalizer and the demodulator-decoder. The table below reports these distinctive features.

Turbo-equalizer	$D \rightarrow E$	$E \rightarrow D$
BP	No conversion	No conversion
BP-GA [8]	Direct conversion	GA
BP-EP [9]	EP-rule	GA
BP-PGA [11]	Direct conversion	PGA
BP-EP-PGA (new)	EP-rule	PGA

As the selected threshold ρ in BP-EP-PGA approaches 1, $\mathbb{I}_i^{\mathrm{D}}$ typically shrinks to the singleton $\{i\}$ ($M = 1$). With this configuration, the messages $m^{\mathrm{PG}}_{f_{\mathrm{T}_i} \to x_i}$, $i \in [N]$ in (B.9) coincide with the messages $m^{\mathrm{G}}_{f_{\mathrm{T}_i} \to x_i}$, $i \in [N]$ in (B.6) and, consequently, BP-EP-PGA and BP-EP become equivalent. Notice that both schemes compute the same messages in stage S2-Equalization. They solely differ in S5-$D \rightarrow E$.

4.2 Computational Complexity

The complexity of the PGA algorithm in [11], which was designed for generic channel matrices, is $\mathcal{O}(N^2 + M^2 2^{QM})$ per symbol. The main contribution to the complexity of the BP-EP-PGA is at (B.4), which requires a $L \times L$ matrix inversion (see [9, (28)]), and at (B.7). Thus, the complexity is $\mathcal{O}(L^3 + M^2 2^{QM})$ per symbol. The complexity reduction method described in [9, Subsec. IV.C] can, however, also be applied to BP-PGA and BP-EP-PGA. Since the beliefs $b^{\mathrm{G}}_{i'}(\boldsymbol{x}_i^{\mathrm{D}})$ (see (B.7)) are obtained from the beliefs $b^{\mathrm{G}}(\boldsymbol{s}_{i'})$, $i \in [N]$ (B.4) needs only be computed once every $(L - M + 1)$ symbols when $\mathbb{K}_\rho = \{-\bar{k}, \ldots, \bar{k}\}$ ($M = 1 + 2\bar{k}$). In this case, the complexity of BP-PGA and BP-EP-PGA is $\mathcal{O}(L^3/(L - M + 1) + M^2 2^{QM})$ per symbol. For this latter case, we summarize the complexities of all discussed turbo-equalizers in the table below [3]:

Turbo-equalizer	Equalizer Complexity per Symbol
BP	$\mathcal{O}(2^{QL})$
BP-GA [8]	$\mathcal{O}(L^2)$
BP-EP [9]	$\mathcal{O}(L^2)$
BP-PGA [11]	$\mathcal{O}(L^3/(L - M + 1) + M^2 2^{QM})$
BP-EP-PGA (new)	$\mathcal{O}(L^3/(L - M + 1) + M^2 2^{QM})$

4.3 Numerical Assessment

We compare the BER performance of the above turbo-equalizers and a receiver designed for and operating in a non-dispersive AWGN channel. A sequence of 2048 information bits is encoded using a 1/2 rate convolutional code with generator polynomials $(23, 35)_8$. The coded bits are interleaved and then mapped onto BPSK symbols ($\mathcal{X} = \{-1, +1\}$), which are transmitted over a severely distorted ISI channel with impulse response $\boldsymbol{h} =$

[3] As all turbo-equalizers use the same type of demodulation-decoding, we focus solely on the complexity due to equalization and message conversion.

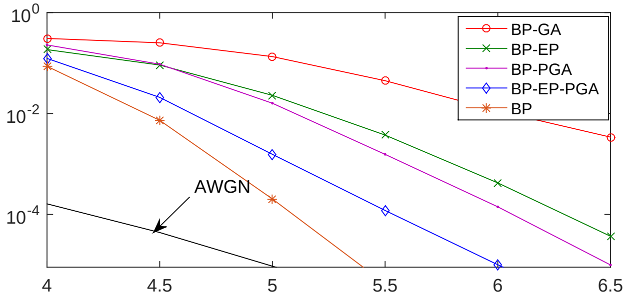


Fig. B.2: BER performance of the considered turbo-equalizers.

$[0.227\ 0.460\ 0.668\ 0.460\ 0.227]^{\mathrm{T}}$. The BER performance is evaluated after 30 iterations. For BP-PGA and BP-EP-PGA, we set ρ so that $M = 3$.

The results are depicted in Fig. B.2. As expected, the BP turbo-equalizer performs best and approaches the AWGN bound as the SNR increases, although at the expense of a large computational complexity. We observe a remarkable performance improvement of BP-EP-PGA compared to the other turbo-equalizers. We attribute this improvement to the fact that BP-EP-PGA combines the advantages of both BP-EP and BP-PGA. Firstly, implementing the EP-based conversion (B.5) instead of a direct conversion of the discrete messages from f_{M_i} to x_i, $i \in [N]$ provides an advantage over BP-GA and BP-PGA. Secondly, implementing (B.9) leads to better performance than when computing the right-hand messages in (B.6) at the expense of a complexity increase, as can be seen by comparing BP-EP and BP-EP-PGA. Since BP-EP can be seen as an instance of our proposed BP-EP-PGA with the setting $M = 1$, we conclude that the tuning of the parameter M (or equivalently ρ) allows for trading performance and computational complexity in the receiver.

5 Appendix

We compute (B.7) for a specific symbol x_i ($i \in [N]$). Select $\boldsymbol{s}_{i'}$ with i' satisfying $i + \bar{k} \leq i' \leq i + (L-1) - \bar{k}$, see $E \to D$ in Subsection 3.2. By applying the BP rule we obtain for the Gaussian belief of $\boldsymbol{s}_{i'}$

$$b^{\mathrm{G}}(\boldsymbol{s}_{i'}) = m^{\mathrm{G}}_{f_{\mathrm{T}_{i'}} \to \boldsymbol{s}_{i'}}(\boldsymbol{s}_{i'}) m^{\mathrm{G}}_{f_{\mathrm{O}_{i'}} \to \boldsymbol{s}_{i'}}(\boldsymbol{s}_{i'}) m^{\mathrm{G}}_{f_{\mathrm{T}_{i'+1}} \to \boldsymbol{s}_{i'}}(\boldsymbol{s}_{i'}). \tag{B.10}$$

To compute $m^{\mathrm{G}}_{f_{\mathrm{T}_{i'}} \to \boldsymbol{s}_{i'}}(\boldsymbol{s}_{i'})$ we use the BP rule in a forward recursion along the variable and factor nodes $f_{\mathrm{T}_{i'-L+1}}, \boldsymbol{s}_{i'-L+1}, \ldots, \boldsymbol{s}_{i'-1}, f_{\mathrm{T}_{i'}}$. Doing so and inserting in (B.10) yields the expression in (B.11). Notice that the product in the first pair of brackets and the integral are functions of $\boldsymbol{s}_i = [x_{i-L+1}, \ldots, x_i]^{\mathrm{T}}$. From the choice of i', the entries of $\boldsymbol{x}^{\mathrm{D}}_i = [x_j : j \in \mathbb{I}^{\mathrm{D}}_i]^{\mathrm{T}}$ are also entries of $\boldsymbol{s}_{i'}$, see $E \to D$ in Subsection 3.2. Thus, the product in the first bracket in (B.11) contains as factors the messages $n^{\mathrm{G}}_{x_j \to f_{\mathrm{T}_j}}(x_j)$, $j \in \mathbb{I}^{\mathrm{D}}_i$.

$$b^{\mathrm{G}}(\boldsymbol{s}_{i'}) = m^{\mathrm{G}}_{f_{\mathrm{T}_{i'+1}} \to \boldsymbol{s}_{i'}}(\boldsymbol{s}_{i'})\, m^{\mathrm{G}}_{f_{\mathrm{O}_{i'}} \to \boldsymbol{s}_{i'}}(\boldsymbol{s}_{i'}) \left[\prod_{l=0}^{L-1} n^{\mathrm{G}}_{x_{i'-l} \to f_{\mathrm{T}_{i'-l}}}(x_{i'-l})\right]$$
$$\times \int \left[\prod_{l=1}^{L-1} m^{\mathrm{G}}_{f_{\mathrm{O}_{i'-l}} \to \boldsymbol{s}_{i'-l}}(\boldsymbol{s}_{i'-l})\right] n^{\mathrm{G}}_{\boldsymbol{s}_{i'-L} \to f_{\mathrm{T}_{i'-L+1}}}(\boldsymbol{s}_{i'-L})\, \mathrm{d}\boldsymbol{s}_{i'-L} \tag{B.11}$$

We implement a PGA by substituting these messages with their discrete counterparts $n_{x_j \to f_{\mathrm{T}_j}}(x_j)$, $j \in \mathbb{I}^{\mathrm{D}}_i$. This substitution can be formally expressed as

$$b^{\mathrm{PG}}(\boldsymbol{s}_{i'}) = \prod_{\kappa \in \mathbb{I}^{\mathrm{D}}_i} \frac{n_{x_\kappa \to f_{\mathrm{T}_\kappa}}(x_\kappa)}{n^{\mathrm{G}}_{x_\kappa \to f_{\mathrm{T}_\kappa}}(x_\kappa)}\, b^{\mathrm{G}}(\boldsymbol{s}_{i'}). \tag{B.12}$$

By using the marginalization constraint of BP we can write

$$m^{\mathrm{PG}}_{f_{\mathrm{T}_i} \to x_i}(x_i) n_{x_i \to f_{\mathrm{T}_i}}(x_i) = \sum_{\boldsymbol{x}^{\mathrm{D}}_i \backslash x_i} \int b^{\mathrm{PG}}(\boldsymbol{s}_{i'})\, \mathrm{d}(\boldsymbol{s}_{i'} \backslash \boldsymbol{x}^{\mathrm{D}}_i). \tag{B.13}$$

Notice that the right-hand term is a marginal belief of x_i. Solving for $m^{\mathrm{PG}}_{f_{\mathrm{T}_i} \to x_i}(x_i)$ in (B.13) yields (B.7).

References

[1] M. Tüchler, R. Koetter, and A. Singer, "Turbo equalization: Principles and new results," *IEEE Trans. Commun.*, vol. 50, pp. 754–767, 2002.

[2] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[3] H.-A. Loeliger, "An introduction to factor graphs," *IEEE Trans. Signal Processing*, pp. 28–40, Jan. 2004.

[4] J. Pearl, "Reverend Bayes on inference engines: a distributed hierarchical approach," in *in Proceedings of the National Conference on Artificial Intelligence*, 1982, pp. 133–136.

[5] 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, 2005.

[6] G. Colavolpe, D. Fertonani, and A. Piemontese, "SISO detection over linear channels with linear complexity in the number of interferers," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 8, pp. 1475–1485, Dec. 2011.

[7] 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. Communication, Control, and Computing*, Sep. 2006, pp. 27–29.

[8] Q. Guo and L. Ping, "LMMSE turbo equalization based on factor graphs," *IEEE J. Select. Areas Commun.*, vol. 26, no. 2, pp. 311–319, Feb. 2008.

[9] P. Sun, C. Zhang, Z. Wang, C. Navarro Manchón, and B. H. Fleury, "Iterative receiver design for ISI channels using combined belief- and expectation-propagation," *IEEE Signal Processing Lett.*, vol. 22, no. 10, pp. 1733–1737, Oct. 2015.

[10] T. P. Minka, "Expectation propagation for approximate Bayesian inference," in *Proceedings of the 17th Conf. on Uncertainty in Artificial Intelligence*, 2001, pp. 362–369.

[11] Q. Guo, D. Huang, L. Ping, S. Nordholm, J. Xi, and P. Li, "Soft-in soft-out detection using partial Gaussian approximation," *IEEE Access*, vol. 2, pp. 427–436, May 2014.

Paper C

Message-Passing Receivers for Single Carrier Systems with Frequency-Domain Equalization

Chuanzong Zhang, Carles Navarro Manchón,
Zhongyong Wang, and Bernard Henri Fleury

The paper has been published in the
IEEE Signal Processing Letters Vol. 22(4), pp. 404–407, 2015.

Abstract

In this paper, we design iterative receiver algorithms for joint frequency-domain equalization and decoding in a single carrier system assuming perfect channel state information. Based on an approximate inference framework that combines belief propagation (BP) and the mean field (MF) approximation, we propose two receiver algorithms with, respectively, parallel and sequential message-passing schedules in the MF part. A recently proposed receiver based on generalized approximate message passing (GAMP) is used as a benchmarking reference. The simulation results show that the BP-MF receiver with sequential passing of messages achieves the best BER performance at the expense of higher computational complexity compared to that of the GAMP receiver. The parallel BP-MF receiver has complexity similar to that of GAMP, but its low convergence rate yields poor performance, especially under high signal-to-noise ratio conditions.

1 Introduction

Single carrier system with frequency domain equalization (SC-FDE) technique is an attractive technology for wireless communications due to its ability to cope with the temporal dispersion introduced by multipath channels. It has the performance, efficiency and low complexity advantages over its orthogonal frequency division multiplexing (OFDM) counterpart, while being less sensitive to power amplifier nonlinearities and carrier frequency offsets, in addition to exhibiting a lower peak-to-average transmitted power ratio [1]. For these reasons, SC-FDE has been selected as the access scheme for the uplink of the 3GPP long term evolution (LTE) and LTE advanced standards [2].

Belief propagation (BP) on factor graphs, also known as sum-product algorithm [3], is a message-passing inference technique that has been widely used in the design of iterative wireless receivers. Its remarkable performance, especially when applied to discrete probabilistic models, justifies its popularity. However, its complexity may become intractable in certain application contexts, e.g. when the probabilistic model includes both discrete and continuous random variables.

As an alternative to BP, variational methods based on the mean field (MF) approximation [4] have been initially used in quantum and statistical physics. The MF approximation has also been formulated as a message passing algorithm, referred to as variational message passing (VMP) algorithm [5]. It has primarily been used on continuous probabilistic conjugate-exponential models. Recently, a method that combines BP and MF [6] as a unified message passing framework on a same factor graph has been proposed, which keeps the virtues of BP and MF but avoids their respective drawbacks. The BP-MF method was applied to joint channel estimation and decoding in MIMO-

OFDM systems [7].

As an alternative approach to deal with the high complexity problem of BP, researchers are also pursuing approximate BP methods. The approximate message passing (AMP) approach was derived from BP by approximating some messages to be Gaussian by invoking the central limit theorem and Taylor expansions, and was initially proposed for compressed sensing [8]. Recently, Rangan extended AMP to the general estimation problem with linear mixing and developed the so-called the generalized approximated message-passing (GAMP) algorithm [9]. GAMP operates as a parallel message passing scheme. It has been previously used for turbo sparse channel estimation and frequency-domain equalization in OFDM systems [10], turbo equalization in SC-FDE systems [11], and iterative channel estimation and detection in OFDM systems impaired by impulsive noise [12].

In this paper, based on the combined inference technique presented in [6], we develop a parallel and a sequential message-passing receiver for a SC-FDE system and compare its performance with an analogous iterative receiver, inspired by [11], implementing GAMP in the equalization part.

Notation- Boldface lowercase and uppercase letters denote vectors and matrices, respectively, while superscripts $(\cdot)^*$, $(\cdot)^{\mathrm{T}}$ and $(\cdot)^{\mathrm{H}}$ represent conjugation, transposition and Hermitian transposition, respectively. The expectation operator with respect to a density $g(x)$ is expressed by $\langle f(x)\rangle_{g(x)} = \int f(x)g(x)dx / \int g(x')dx'$, while $\mathrm{var}[x]_{g(x)} = \langle |x|^2\rangle_{g(x)} - |\langle x\rangle_{g(x)}|^2$ represents the variance. The pdf of a complex Gaussian distribution with mean μ and variance ν is represented by $\mathcal{CN}(x;\mu,\nu)$. The relation $f(x) = cg(x)$ for some positive constant c is written as $f(x) \propto g(x)$. We use $\|\cdot\|$ to stand for Euclidian norm. The $N \times N$ normalized discrete Fourier matrix is denoted by $\boldsymbol{F}$, with entries $F_{ai} = 1/\sqrt{N}e^{-\jmath 2\pi(a-1)(i-1)/N}$.

2 System Model

The finite sequence of information bits $\boldsymbol{b} = [b_1, \ldots, b_K]^{\mathrm{T}}$ of an SC-FDE block is encoded and interleaved by using a rate R channel code and a random interleaver, yielding the codeword vector $\boldsymbol{c} = [\boldsymbol{c}_1^{\mathrm{T}}, \ldots, \boldsymbol{c}_N^{\mathrm{T}}]^{\mathrm{T}}$, where the ith sub-vector is $\boldsymbol{c}_i = [c_i^1, \ldots, c_i^Q]^{\mathrm{T}}$ with Q denoting the modulation order. The codeword $\boldsymbol{c}$ is complex modulated, resulting in the vector of data symbols $\boldsymbol{x} = [x_1, \ldots, x_N]^{\mathrm{T}}$, which is transmitted over the wireless channel after the addition of a cyclic prefix (CP). At the receiver end, the CP is removed and the received signal is Fourier transformed, yielding

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{z} + \boldsymbol{w} = \boldsymbol{H}\boldsymbol{F}\boldsymbol{x} + \boldsymbol{w}, \tag{C.1}$$

where $\boldsymbol{H} = \mathrm{diag}(\boldsymbol{h})$ with $\boldsymbol{h} = [h_1, h_2, \ldots, h_N]^{\mathrm{T}}$ representing the vector of frequency-domain channel weights, $\boldsymbol{z} = \boldsymbol{F}\boldsymbol{x}$ is a vector containing the equiv-

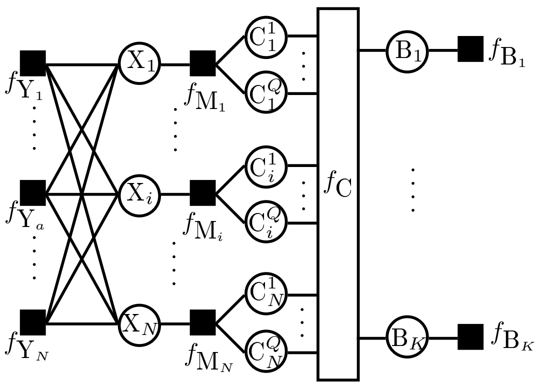


Fig. C.1: Factor graph representing the probabilistic model in (C.2).

alent frequency-domain data symbols and $\boldsymbol{w}$ is a complex additive white Gaussian noise vector with covariance matrix $\lambda^{-1}\boldsymbol{I}$.

2.1 Probabilistic Representation and Factor Graph

Based on (C.1), we can factorize the joint pdf of all unknown random variables conditioned on the observation $\boldsymbol{y}$ as

$$p(\boldsymbol{x}, \boldsymbol{c}, \boldsymbol{b}|\boldsymbol{y}) \propto \prod_{a=1}^{N} f_{\mathrm{Y}_a}(y_a, \boldsymbol{x}) \prod_{i=1}^{N} f_{\mathrm{M}_i}(x_i, \boldsymbol{c}_i) f_{\mathrm{C}}(\boldsymbol{c}, \boldsymbol{b}) \prod_{k=1}^{K} f_{\mathrm{B}_k}(b_k) \tag{C.2}$$

where $f_{\mathrm{Y}_a}(y_a, \boldsymbol{x}) \triangleq p(y_a|\boldsymbol{x}) = \mathcal{CN}\left(y_a; h_a \boldsymbol{F}_a \boldsymbol{x}, \lambda^{-1}\right)$ with $\boldsymbol{F}_a$ being the ath row of $\boldsymbol{F}$ and h_a being the ath entry of $\boldsymbol{h}$, $f_{\mathrm{M}_i}(x_i, \boldsymbol{c}_i)$ and $f_{\mathrm{C}}(\boldsymbol{c}, \boldsymbol{b})$ stand for the modulation and coding constraints, and $f_{\mathrm{B}_k}(b_k) \triangleq p(b_k)$ is the prior of the kth information bit. In (C.2), we have used the fact that, given $\boldsymbol{x}$, $\boldsymbol{y}$ is conditionally independent of $\boldsymbol{c}$ and $\boldsymbol{b}$. The factor graph [3] shown on Fig. C.1 represents the above factorization.

3 Combined BP and MF Framework

In this section, we derive the messages passed on the factor graph shown in Fig. C.1 by using the BP-MF message updating rules [6] and discuss their scheduling.

We group the factor nodes in two disjoint subsets, a BP part for demodulation and decoding, and an MF part for equalization,

$$\begin{aligned}\mathcal{A}_{\mathrm{MF}} &= \{f_{\mathrm{Y}_a}; a \in [1:N]\}\\ \mathcal{A}_{\mathrm{BP}} &= \{f_{\mathrm{M}_i}; i \in [1:N]\} \cup \{f_{\mathrm{C}}\} \cup \{f_{\mathrm{B}_k}; k \in [1:K]\},\end{aligned}$$

where $\mathcal{A}_{\mathrm{MF}}$ and $\mathcal{A}_{\mathrm{BP}}$ denote the sets of factor nodes in the MF and BP parts, respectively. For factor nodes in the BP part, we calculate the messages to neighboring variable nodes using the sum-product rule, and send extrinsic messages. For factor nodes in the MF part, messages to neighboring variable nodes are computed by the VMP rule, and beliefs are passed.

Due to the factor nodes $\{f_{\mathrm{M}_i}\}$ being in the BP part, the extrinsic messages passed from X_i to f_{M_i}, $i = 1, \ldots, N$, read

$$\begin{aligned}n_{\mathrm{X}_i \to f_{\mathrm{M}_i}}(x_i) &\propto \prod_{a=1}^{N} m_{f_{\mathrm{Y}_a} \to \mathrm{X}_i}(x_i)\\ &\triangleq \mathcal{CN}(x_i; \mu_{\mathrm{X}_i \to f_{\mathrm{M}_i}}, \nu_{\mathrm{X}_i \to f_{\mathrm{M}_i}}). \end{aligned} \tag{C.3}$$

Messages $\{m_{f_{\mathrm{M}_i} \to \mathrm{X}_i}(x_i)\}$, yielded by the sum-product rule, come from the soft demodulation part. Messages from X_i to f_{Y_a} in the MF part correspond, for all a, to the belief $b(x_i)$, which is computed by collecting the messages from all neighbors of X_i, i.e.

$$\begin{aligned}b(x_i) &= n_{\mathrm{X}_i \to f_{\mathrm{Y}_a}}(x_i) \propto \prod_{a=1}^{N} m_{f_{\mathrm{Y}_a} \to \mathrm{X}_i}(x_i) m_{f_{\mathrm{M}_i} \to \mathrm{X}_i}(x_i)\\ &\propto n_{\mathrm{X}_i \to f_{\mathrm{M}_i}}(x_i) m_{f_{\mathrm{M}_i} \to \mathrm{X}_i}(x_i), && \text{(C.4)}\\ \mu_{\mathrm{X}_i \to \mathrm{Y}} &\triangleq \langle x_i \rangle_{b(x_i)}, && \text{(C.5)}\\ \nu_{\mathrm{X}_i \to \mathrm{Y}} &\triangleq \mathrm{var}[x_i]_{b(x_i)}. && \text{(C.6)}\end{aligned}$$

We group the parameters $\{\mu_{\mathrm{X}_i \to \mathrm{Y}}\}$ into the vector $\boldsymbol{\mu}_{\mathrm{X} \to \mathrm{Y}} = [\mu_{\mathrm{X}_1 \to \mathrm{Y}}, \mu_{\mathrm{X}_2 \to \mathrm{Y}}, \ldots, \mu_{\mathrm{X}_N \to \mathrm{Y}}]^{\mathrm{T}}$.

Using the VMP rule and the messages $\{n_{\mathrm{X}_i \to f_{\mathrm{Y}_a}}(x_i)\}$, the messages from f_{Y_a} to X_i are obtained as

$$\begin{aligned}m_{f_{\mathrm{Y}_a} \to \mathrm{X}_i}(x_i) &\propto \exp\left\{ \langle \log f_{\mathrm{Y}_a}(y_a, \boldsymbol{x}) \rangle_{\prod_{j \neq i} b(x_j)} \right\}\\ &\propto \mathcal{CN}\left(x_i; \mu_{f_{\mathrm{Y}_a} \to \mathrm{X}_i}, \nu_{f_{\mathrm{Y}_a} \to \mathrm{X}_i}\right),\end{aligned} \tag{C.7}$$

where $\mu_{f_{\mathrm{Y}_a} \to \mathrm{X}_i} \triangleq F_{ai}^* h_a^* (y_a - h_a \boldsymbol{F}_a \boldsymbol{\mu}_{\mathrm{X}_{\bar{i}} \to \mathrm{Y}}) / \left(F_{ai}^* h_a^* h_a F_{ai}\right)$, $\boldsymbol{\mu}_{\mathrm{X}_{\bar{i}} \to \mathrm{Y}}$ is equal to $\boldsymbol{\mu}_{\mathrm{X} \to \mathrm{Y}}$ with its ith entry replaced by 0, and $\nu_{f_{\mathrm{Y}_a} \to \mathrm{X}_i} \triangleq 1/\left(\lambda F_{ai}^* h_a^* h_a F_{ai}\right)$.

Substituting (C.7) into (C.3), we obtain

$$\begin{aligned}\mu_{\mathrm{X}_i \to f_{\mathrm{M}_i}} &= \frac{1}{C} \boldsymbol{F}_i^{\mathrm{H}} \boldsymbol{H}^{\mathrm{H}} \left(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{F}\boldsymbol{\mu}_{\mathrm{X}_{\bar{i}} \to \mathrm{Y}}\right) && \text{(C.8)}\\ \nu_{\mathrm{X}_i \to f_{\mathrm{M}_i}} &= (\lambda C)^{-1} && \text{(C.9)}\end{aligned}$$

where $\boldsymbol{F}_i$ is the ith column of $\boldsymbol{F}$, and $C \triangleq \boldsymbol{F}_i^{\mathrm{H}} \boldsymbol{H}^{\mathrm{H}} \boldsymbol{H} \boldsymbol{F}_i = \frac{\|\boldsymbol{h}\|^2}{N}$.

The factor graph in Fig. C.1 is very densely connected, especially in the MF part. Hence, there is a multitude of different options for scheduling the messages. In this paper, the standard message-passing schedule is chosen for the BP part. Messages are passed from the modulation nodes to the coding node in parallel, followed by a round of decoding using the forward-backward (BCJR) algorithm [3], and the outcome messages are passed to the modulation nodes simultaneously. For the MF part, we propose two kinds of scheduling, a parallel schedule and a sequential schedule, which are described next.

3.1 The Parallel Schedule

The messages $\{n^t_{\mathrm{X}_i \to f_{\mathrm{Y}_a}}(x_i)\}$ (beliefs) from variable nodes $\{\mathrm{X}_i\}$ to factor nodes $\{f_{\mathrm{Y}_a}\}$ are computed from (C.3) and (C.4). The superscript t denotes the iteration index. All the messages $\{m^t_{f_{\mathrm{Y}_a} \to \mathrm{X}_i}(x_i)\}$ from factor nodes $\{f_{\mathrm{Y}_a}\}$ to variable nodes $\{\mathrm{X}_i\}$ are simultaneously computed using (C.7). The aforementioned process is carried out twice per iteration. Subsequently, messages $\{n^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}}(x_i)\}$ from $\{\mathrm{X}_i\}$ to $\{f_{\mathrm{M}_i}\}$ are computed using (C.8) and (C.9) and passed on to the BP part. The above procedure is summarized in **Algorithm 1**.

Algorithm 1 BP-MF with Parallel Scheduling

for all i: initialize $m^0_{f_{\mathrm{M}_i} \to \mathrm{X}_i}(x_i), \mu^0_{\mathrm{X}_i \to f_{\mathrm{M}_i}}, \nu^0_{\mathrm{X}_i \to f_{\mathrm{M}_i}}$.
for all i: $\mu^0_{\mathrm{X}_i \to \mathrm{Y}} \leftarrow \langle x_i \rangle_{\mathcal{CN}\left(x_i; \mu^0_{\mathrm{X}_i \to f_{\mathrm{M}_i}}, \nu^0_{\mathrm{X}_i \to f_{\mathrm{M}_i}}\right) m^0_{f_{\mathrm{M}_i} \to \mathrm{X}_i}(x_i)}$
for $t = 1 \to$Iteration **do**
  $\boldsymbol{\mu}^t_{\mathrm{Z}} \leftarrow \boldsymbol{y} - \boldsymbol{H}\boldsymbol{F}\boldsymbol{\mu}^{t-1}_{\mathrm{X} \to \mathrm{Y}}$
  for all i: $\mu^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}} \leftarrow \frac{1}{C} \boldsymbol{F}_i^{\mathrm{H}} \boldsymbol{H}^{\mathrm{H}} \boldsymbol{\mu}^t_{\mathrm{Z}} + \mu^t_{\mathrm{X}_i \to \mathrm{Y}}$
  for all i: $\nu^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}} \leftarrow \frac{1}{\lambda C}$
  for all i: $\mu^t_{\mathrm{X}_i \to \mathrm{Y}} \leftarrow \langle x_i \rangle_{\mathcal{CN}\left(x_i; \mu^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}}, \nu^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}}\right) m^{t-1}_{f_{\mathrm{M}_i} \to \mathrm{X}_i}(x_i)}$
  $\boldsymbol{\mu}^t_{\mathrm{Z}} \leftarrow \boldsymbol{y} - \boldsymbol{H}\boldsymbol{F}\boldsymbol{\mu}^{t}_{\mathrm{X} \to \mathrm{Y}}$
  for all i: $\mu^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}} \leftarrow \frac{1}{C} \boldsymbol{F}_i^{\mathrm{H}} \boldsymbol{H}^{\mathrm{H}} \boldsymbol{\mu}^t_{\mathrm{Z}} + \mu^t_{\mathrm{X}_i \to \mathrm{Y}}$
  for all i: $\nu^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}} \leftarrow \frac{1}{\lambda C}$
  for all i: $\mathcal{CN}\left(x_i; \mu^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}}, \nu^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}}\right) \to$ BP part
  for all i: $m^t_{f_{\mathrm{M}_i} \to \mathrm{X}_i}(x_i) \leftarrow$ BP part
  for all i: $\mu^t_{\mathrm{X}_i \to \mathrm{Y}} \leftarrow \langle x_i \rangle_{\mathcal{CN}\left(x_i; \mu^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}}, \nu^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}}\right) m^{t}_{f_{\mathrm{M}_i} \to \mathrm{X}_i}(x_i)}$
end for t

3.2 The Sequential Schedule

Similar to the parallel schedule, the messages $\{n^t_{\mathrm{X}_i\to f_{\mathrm{Y}_a}}(x_i)\}$ from variable nodes $\{\mathrm{X}_i\}$ to factor nodes $\{f_{\mathrm{Y}_a}\}$ are computed first. Then, sequentially for each i from 1 to N, the messages $\{m^t_{\mathrm{Y}_a\to\mathrm{X}_i}(x_i)\}$ are computed using messages $\{n^{t-1}_{\mathrm{X}_k\to f_{\mathrm{Y}_a}}(x_k); k\in[i+1:N], a\in[1:N]\}$ and $\{n^t_{\mathrm{X}_j\to f_{\mathrm{Y}_a}}(x_j); j\in[1:i-1], a\in[1:N]\}$. Before calculating messages from factor nodes $\{f_{\mathrm{Y}_a}\}$ to variable node X_{i+1}, the messages $\{n^t_{\mathrm{X}_i\to f_{\mathrm{Y}_a}}(x_i)\}$ are updated using (C.4). Finally, extrinsic messages $\{n^t_{\mathrm{X}_i\to f_{\mathrm{M}_i}}(x_i)\}$ are delivered to the BP part. The above procedure is illustrated in **Algorithm 2**.

Algorithm 2 BP-MF with Sequential Scheduling

1: **for all** i: initialize $m^0_{f_{\mathrm{M}_i}\to\mathrm{X}_i}(x_i), \mu^0_{\mathrm{X}_i\to f_{\mathrm{M}_i}}, \nu^0_{\mathrm{X}_i\to f_{\mathrm{M}_i}}$.
2: **for all** i: $\mu^0_{\mathrm{X}_i\to\mathrm{Y}} \leftarrow \langle x_i\rangle_{\mathcal{CN}\left(x_i;\mu^0_{\mathrm{X}_i\to f_{\mathrm{M}_i}},\nu^0_{\mathrm{X}_i\to f_{\mathrm{M}_i}}\right)m^0_{f_{\mathrm{M}_i}\to\mathrm{X}_i}(x_i)}$
3: **for** $t = 1 \to$ Iteration **do**
4: $\quad\boldsymbol{\mu}^t_{\mathrm{Z}} \leftarrow \boldsymbol{y} - \boldsymbol{H}\boldsymbol{F}\boldsymbol{\mu}^{t-1}_{\mathrm{X}\to\mathrm{Y}}$
5: $\quad$**for** $i = 1 \to N$ **do**
6: $\quad\quad\mu^t_{\mathrm{X}_i\to f_{\mathrm{M}_i}} \leftarrow \frac{1}{C}\boldsymbol{F}_i^{\mathrm{H}}\boldsymbol{H}^{\mathrm{H}}\boldsymbol{\mu}^t_{\mathrm{Z}} + \mu^{t-1}_{\mathrm{X}_i\to\mathrm{Y}}$
7: $\quad\quad\nu^t_{\mathrm{X}_i\to f_{\mathrm{M}_i}} \leftarrow \frac{1}{\lambda C}$
8: $\quad\quad\mu^t_{\mathrm{X}_i\to\mathrm{Y}} \leftarrow \langle x_i\rangle_{\mathcal{CN}\left(x_i;\mu^t_{\mathrm{X}_i\to f_{\mathrm{M}_i}},\nu^t_{\mathrm{X}_i\to f_{\mathrm{M}_i}}\right)m^{t-1}_{f_{\mathrm{M}_i}\to\mathrm{X}_i}(x_i)}$
9: $\quad\quad\boldsymbol{\mu}^t_{\mathrm{Z}} \leftarrow \boldsymbol{\mu}^t_{\mathrm{Z}} + \boldsymbol{H}\boldsymbol{F}_i\left(\mu^t_{\mathrm{X}_i\to\mathrm{Y}} - \mu^{t-1}_{\mathrm{X}_i\to\mathrm{Y}}\right)$
10: $\quad$**end for** i
11: $\quad$**for all** i: $\mathcal{CN}\left(x_i;\mu^t_{\mathrm{X}_i\to f_{\mathrm{M}_i}},\nu^t_{\mathrm{X}_i\to f_{\mathrm{M}_i}}\right) \to$ BP part
12: $\quad$**for all** i: $m^t_{f_{\mathrm{M}_i}\to\mathrm{X}_i}(x_i) \leftarrow$ BP part
13: $\quad$**for all** i: $\mu^t_{\mathrm{X}_i\to\mathrm{Y}} \leftarrow \langle x_i\rangle_{\mathcal{CN}\left(x_i;\mu^t_{\mathrm{X}_i\to f_{\mathrm{M}_i}},\nu^t_{\mathrm{X}_i\to f_{\mathrm{M}_i}}\right)m^t_{f_{\mathrm{M}_i}\to\mathrm{X}_i}(x_i)}$
14: **end for** t

3.3 Noise Precision Estimation

The above algorithms assume that the noise precision λ is known. However, the BP-MF framework allows for estimating the noise precision if necessary. In this case λ is an unknown variable and the probabilistic model (2) is modified as follows: $p(\boldsymbol{x},\boldsymbol{c},\boldsymbol{b}|\boldsymbol{y})$ becomes $p(\boldsymbol{x},\boldsymbol{c},\boldsymbol{b},\lambda|\boldsymbol{y})$, $f_{\mathrm{Y}_a}(y_a,\boldsymbol{x})$ is replaced by $f_{\mathrm{Y}_a}(y_a,\boldsymbol{x},\lambda) \triangleq p(y_a|\boldsymbol{x},\lambda)$ and an additional factor $f_\Lambda(\lambda) \triangleq p(\lambda)$, denoting the prior distribution of λ, is inserted in the factorization. A variable node Λ and a factor node f_Λ are drawn on factor graph Fig. C.1 with f_Λ connected to Λ, and Λ also linked to the factor nodes $\{f_{\mathrm{Y}_a}\}$. With the improper prior pdf $p(\lambda) \propto 1/\lambda$, the update for $\hat{\lambda}$ reads [7]

$$\hat{\lambda} = N/(\|\boldsymbol{\mu}_{\mathrm{Z}}\|^2 + \|\boldsymbol{h}\|^2 \sum_{i=1}^{N} \nu_{\mathrm{X}_i \to \mathrm{Y}}), \tag{C.10}$$

where $\nu_{\mathrm{X}_i \to \mathrm{Y}} = \mathrm{var}\,[x_i]_{\mathcal{CN}\left(x_i; \mu_{\mathrm{X}_i \to f_{\mathrm{M}_i}}, \nu_{\mathrm{X}_i \to f_{\mathrm{M}_i}}\right) m_{f_{\mathrm{M}_i} \to \mathrm{X}_i}(x_i)}$.

If the noise precision is to be estimated, λ is replaced by $\hat{\lambda}^t$ in the algorithms and is updated using (C.10) following Steps 4 and 8 in **Algorithm 1** and Step 4 in **Algorithm 2**.

4 BP and GAMP Method

We briefly present the application of the GAMP algorithm to the SC-FDE system. As for the BP-MF method, the factor graph shown in Fig. C.1 is also divided into two parts, a BP part and a GAMP part. GAMP [9] is implemented for equalization and BP is used for demodulation and decoding. The messages from the BP part $\{m^t_{f_{\mathrm{M}_i} \to \mathrm{X}_i}(x_i)\}$ are used to compute the means $\{\mu^t_{\mathrm{X}_i}\}$ and variances $\{\nu^t_{\mathrm{X}_i}\}$ of the beliefs $\{b(x_i)\}$. The extrinsic messages from nodes $\{\mathrm{X}_i\}$ to nodes $\{f_{\mathrm{M}_i}\}$, $\{m^t_{\mathrm{X}_i \to f_{\mathrm{M}_i}}(x_i) = \mathcal{CN}(x_i; \mu^t_{r_i}, \nu^t_{r_i})\}$, are used for soft demodulation. The BP-GAMP algorithm, proposed in [11], is described in **Algorithm 3**.

5 Simulation Results

We consider an SC-FDE system with $N = 256$ data symbols per block and a bandwidth of $W = N * 15$ KHz. A block of data symbols is obtained from a sequence of information bits encoded using a rate $R = 1/3$ convolutional code with generator polynomials $(133, 171, 165)_8$, or a rate $R = 1/2$ code with $(5, 7)_8$. Random interleaving and QPSK or 16QAM modulation are subsequently applied. The realizations of the channel transfer function are generated using the 3GPP ETU channel model and their samples $\boldsymbol{h}$ are assumed to be known at the receiver side. We assess the performance of receivers implementing three different algorithms: BP-MF with sequential scheduling (BP-MF-s), BP-MF with parallel scheduling (BP-MF-p), and BP-GAMP. For the BP-MF receivers, we allow two different variants: one in which the noise precision (NP) λ is assumed to be known (Known NP) and one in which it is re-estimated at every iteration of the algorithms (Unknown NP). To include an ideal reference, the matched-filter bound (MFB) is also evaluated. The MFB is the performance of a receiver which, when detecting a symbol in an

Algorithm 3 BP-GAMP Receiver

for all i: initialize $\mu^0_{\mathrm{X}_i}, \nu^0_{\mathrm{X}_i}$.
$\boldsymbol{\mu}^0_{\mathrm{S}} \leftarrow \mathbf{0}$; $\boldsymbol{\mu}_{\mathrm{Z}} = \boldsymbol{y}./\boldsymbol{h}$; $\boldsymbol{\nu}_{\mathrm{Z}} = 1./(\lambda \boldsymbol{h} \odot \boldsymbol{h}^*)$
for $t = 1 \rightarrow$ Iteration **do**
  $\boldsymbol{\nu}^t_{\mathrm{P}} \leftarrow \mathbf{1} \sum_{i=1}^{N} \nu^{t-1}_{\mathrm{X}_i} / N$
  $\boldsymbol{\mu}^t_{\mathrm{P}} \leftarrow \boldsymbol{F} \boldsymbol{\mu}^{t-1}_{\mathrm{X}} - \boldsymbol{\mu}^{t-1}_{\mathrm{S}} \odot \boldsymbol{\nu}^t_{\mathrm{P}}$,
  $\boldsymbol{\nu}^t_{\mathrm{S}} \leftarrow 1./(\boldsymbol{\nu}_{\mathrm{Z}} + \boldsymbol{\nu}^t_{\mathrm{P}})$
  $\boldsymbol{\mu}^t_{\mathrm{S}} \leftarrow (\boldsymbol{\mu}_{\mathrm{Z}} - \boldsymbol{\mu}^t_{\mathrm{P}}) \odot \boldsymbol{\nu}^t_{\mathrm{S}}$
  $\boldsymbol{\nu}^t_{\mathrm{R}} \leftarrow \mathbf{1} N / (\sum_{a=1}^{N} \nu^t_{\mathrm{S}_a})$
  $\boldsymbol{\mu}^t_{\mathrm{R}} \leftarrow \boldsymbol{\mu}^{t-1}_{\mathrm{X}} + \boldsymbol{\nu}^t_{\mathrm{R}} \odot (\boldsymbol{F}^{\mathrm{H}} \boldsymbol{\mu}^t_{\mathrm{S}})$
  for all i: $\mathcal{CN}\left(x_i; \mu^t_{\mathrm{R}_i}, \nu^t_{\mathrm{R}_i}\right) \rightarrow$ BP part.
  for all i: $m^t_{f_{\mathrm{M}_i} \rightarrow \mathrm{X}_i}(x_i) \leftarrow$ BP part.
  for all i: $\mu^t_{\mathrm{X}_i} \leftarrow \langle x_i \rangle_{\mathcal{CN}\left(x_i; \mu^t_{\mathrm{R}_i}, \nu^t_{\mathrm{R}_i}\right) m^t_{f_{\mathrm{M}_i} \rightarrow \mathrm{X}_i}(x_i)}$
  for all i: $\nu^t_{\mathrm{X}_i} \leftarrow \mathrm{var}\left[x_i\right]_{\mathcal{CN}\left(x_i; \mu^t_{\mathrm{R}_i}, \nu^t_{\mathrm{R}_i}\right) m^t_{f_{\mathrm{M}_i} \rightarrow \mathrm{X}_i}(x_i)}$
end for t

where $\boldsymbol{\mu}^t_{\mathrm{X}} = [\mu^t_{\mathrm{X}_1}, \ldots, \mu^t_{\mathrm{X}_N}]^{\mathrm{T}}$, $\odot$ and ./ stand for component-wise product and division, respectively, and $\nu^t_{\mathrm{S}_a}$ denotes the ath entry of $\boldsymbol{\nu}^t_{\mathrm{S}}$.

SC-FDE block, has perfect knowledge of every other symbol in the block and the noise precision.

In Fig. C.2, the BER of the receivers operating at a SNR of 10dB is depicted as a function of the iteration index. Two different modulation schemes and coding rates have been selected: a low-rate (R=1/3) system using QPSK modulation, and a high-rate (R=1/2) system employing 16QAM. For the BP-MF receiver, we observe that convergence is dramatically improved by using the sequential schedule as compared to the parallel schedule. The latter requires more than 30 iterations to converge in the low-rate case.[1] The reason for this is that, with the serial schedule, the estimates of the data symbols that have already been obtained in the course of the tth iteration are immediately used to estimate those data symbols which have not yet been estimated during iteration t. Conversely, with the parallel schedule such estimates are not used for equalization until the $(t+1)$th iteration. The BP-GAMP algorithm, for its part, exhibits an erratic BER behavior in the first iterations, before stabilizing to BER values slightly higher than those achieved by the BP-MF-s receiver.

In Fig. C.3, the BER performance of the receivers is shown over a wide range of SNR values, with all receivers running 20 iterations of their respec-

[1] This effect is even more pronounced for the high-rate case. Results have been omitted in the plot.

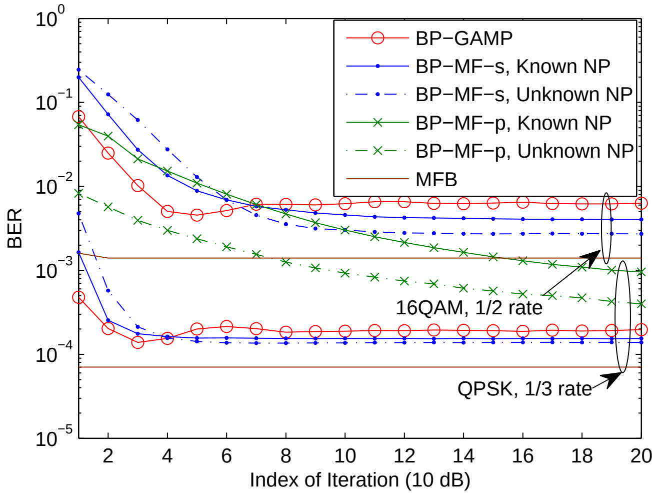


Fig. C.2: BER performance of the considered receivers versus iteration index for an Eb/N0 of 10 dB.

tive algorithms. As already observed in Fig. C.2, the BP-MF-s receiver performs best of all receivers, with gains of 0.5dB and 1dB with respect to the BP-GAMP receiver for the low- and high-rate systems, respectively. The BP-MF-p receiver achieves a performance similar to that of its sequential counterpart in the lowest SNR range, but its convergence is too slow to be used at larger SNR values.

Interestingly, including the estimation of the noise precision in the BP-MF-s receiver has two effects, as seen from Figs. C.2 and C.3: on the one hand, it slightly slows down the convergence speed of the algorithm; on the other hand, the performance obtained after convergence is slightly better than that of the receiver which has knowledge of the true noise precision. As the variance of the estimates of the data symbols are integrated in the noise precision estimates, the algorithm including the noise precision estimation exhibits a more robust behaviour, at the expense of a slightly lower convergence speed.

In terms of computational complexity, we point out that all three receivers differ only in the equalization part. For this part, the BP-MF-p and BP-GAMP receivers have similar complexity, in the order of $\mathcal{O}(N \log_2 N)$ complex operations, as FFT processing can be used due to the passing of messages being parallel. The FFT cannot be used when messages are passed sequentially, which increases the complexity of the BP-MF-s equalization part to $\mathcal{O}(N^2)$ per SC-FDE block.

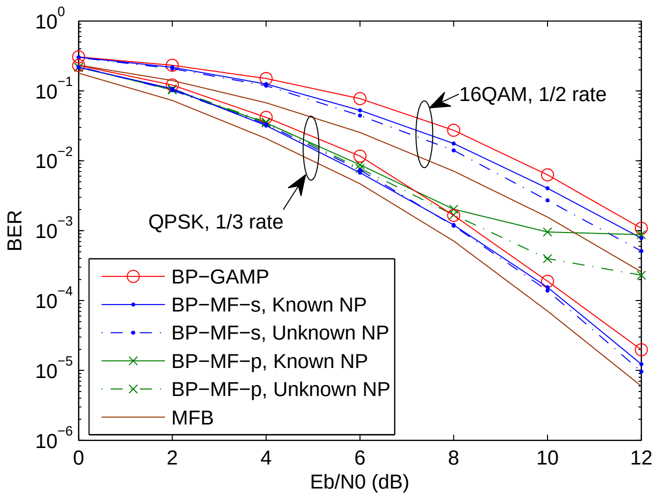


Fig. C.3: BER performance of the considered receivers versus Eb/N0.

6 Conclusion

Based on the BP-MF inference framework, we have developed parallel and sequential message-passing receivers for joint equalization and decoding in an SC-FDE system and compared their performance to that of an analogous receiver using GAMP for equalization.

Our numerical assessment shows that, for the considered SC-FDE system, the receiver using the BP-MF framework with sequential message-passing schedule is superior, in terms of performance, to its parallel counterpart and the receiver using GAMP. This performance improvement comes at the expense of an increase in computational complexity. Additionally, our results show that embedding the estimation of the noise precision parameter in the iterative algorithm improves the receiver's performance even when the true value of this parameter is known beforehand.

References

[1] D. Falconer, S. Ariyavisitakul, A. Benyamin-Seeyar, and B. Eidson, "Frequency domain equalization for single-carrier broadband wireless systems," *IEEE Commun. Mag.*, vol. 40, no. 4, pp. 58–66, Apr. 2002.

[2] E. Dahlman, S. Parkvall, J. Sköld, and P. Beming, *3G Evolution: HSPA and LTE for Mobile Broadband*, 2nd ed. Academic Press, 2008.

[3] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[4] G. Parisi, *Statistical field theory*. Perseus Books, 1988.

[5] J. Winn and C. Bishop, "Variational message passing," *Journal of Machine Learning Research*, vol. 6, pp. 661–694, 2005.

[6] E. Riegler, G. E. Kirkelund, C. Navarro Manchón, M.-A. Badiu, and B. H. Fleury, "Merging belief propagation and the mean field approximation: A free energy approach," *IEEE Trans. Inform. Theory*, vol. 59, no. 1, pp. 588–602, Jan. 2013.

[7] C. Navarro Manchón, G. E. Kirkelund, E. Riegler, L. P. B. Christensen, and B. H. Fleury, "Receiver architectures for MIMO-OFDM based on a combined VMP-SP algorithm," *2011, arXiv:1111.5848 [stat.ML]*.

[8] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inform. Theory*, vol. 57, no. 2, pp. 764–785, Feb. 2011.

[9] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT 2011)*, Aug. 2011, pp. 2168–2172.

[10] P. Schniter, "A message-passing receiver for BICM-OFDM over unknown clustered-sparse channels," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 8, pp. 1462–1474, Dec. 2011.

[11] Q. Guo, D. Huang, S. Nordholm, J. Xi, and Y. Yu, "Iterative frequency domain equalization with generalized approximate message passing," *IEEE Signal Processing Lett.*, vol. 20, no. 6, pp. 559–562, June 2013.

[12] 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 Processing*, vol. 62, no. 6, pp. 1576–1589, March 2014.

Paper D

Low Complexity Sparse Bayesian Learning Using Combined BP and MF with a Stretched Factor Graph

Chuanzong Zhang, Zhengdao Yuan, Zhongyong Wang and Qinghua Guo

The paper has been submitted to the
Signal Processing (Elsevier), 2016.

The layout has been revised.

Abstract

This paper concerns message passing based approaches to sparse Bayesian learning (SBL) with a linear model corrupted by additive white Gaussian noise with unknown variance. With the conventional factor graph, mean field (MF) message passing based algorithms have been proposed in the literature. In this work, instead of using the conventional factor graph, we modify the factor graph by adding some extra hard constraints (the graph looks like being 'stretched'), which enables the use of combined belief propagation (BP) and MF message passing. We then propose a low complexity BP-MF SBL algorithm based on which an approximate BP-MF SBL algorithm is also developed to further reduce the complexity. Thanks to the use of BP, the BP-MF SBL algorithms show their merits compared with state-of-the-art MF SBL algorithms: they deliver even better performance with much lower complexity compared with the vector-form MF SBL algorithm and they significantly outperform the scalar-form MF SBL algorithm with similar complexity.

1 Introduction

Recently, compressed sensing [1, 2] has received tremendous attention and it has found wide applications in a large variety of engineering areas, e.g. biomagnetic imaging, sparse channel estimation, bandlimited extrapolation and spectral estimation, echo cancellation and image restoration [3]. In compressed sensing, a vector $\boldsymbol{\alpha} \in \mathbb{C}^{L\times 1}$ which exhibits sparsity is estimated based on the measurement vector $\boldsymbol{y} \in \mathbb{C}^{N\times 1}$ with the following model

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{\alpha} + \boldsymbol{\omega} \tag{D.1}$$

where $\boldsymbol{\Phi} \in \mathbb{C}^{N\times L}$ is called dictionary matrix and $\boldsymbol{\omega}$ represents an additive white Gaussian noise (AWGN) vector with zero mean and covariance matrix $\lambda^{-1}\boldsymbol{I}$. In this work, we are particularly interested in the case that the variance of the AWGN (or the precision parameter λ) is unknown.

Besides convex [4] and greedy [5] methods, sparse Bayesian learning (SBL) [6–8] is an alternative method of sparse signal estimation, which aims at finding a sparse maximum a posteriori (MAP) estimate $\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmax}}\, p(\boldsymbol{\alpha}|\boldsymbol{y})$ of the vector $\boldsymbol{\alpha}$ by specifying a priori probability density function (pdf) $p(\boldsymbol{\alpha})$. Instead of working directly with a prior $p(\boldsymbol{\alpha})$, SBL typically employs a two-layer (2-L) hierarchical structure [9] that assumes a conditional prior pdf $p(\boldsymbol{\alpha}|\gamma)$ and a hyper-priori pdf $p(\gamma)$, so that $p(\boldsymbol{\alpha}) = \int_\gamma p(\boldsymbol{\alpha}|\gamma)p(\gamma)d\gamma$ has a sparsity-inducing nature. Most recently, SBL has been efficiently implemented using belief propagation (BP) [10, 11] and approximate message passing [12, 13]. However, these methods assume that λ is known, which may not be true in many applications. This work deals with message passing based approaches to SBL with unknown λ.

Mean field (MF) based message passing [14–16], which is also often referred to as variational message passing (VMP), has been widely used for approximate Bayesian inference, especially for exponential distributions. With 2-L or 3-L hierarchical priori structures, Pedersen et al. proposed an MF SBL algorithm (with unknown λ) [17], which was applied to sparse channel estimation in OFDM. As the MF SBL algorithm deals with the sparse signal $\boldsymbol{\alpha}$ in a vector-form, matrix inversion is involved in each iteration and its computational complexity is as high as $\mathcal{O}(L^3)$. To address the issue of complexity, a low complexity MF SBL algorithm [18] is then proposed, where the inverse of a large matrix is decomposed into a number of matrix inverses with smaller size. Flexible trade-off between complexity and performance can be achieved by adjusting the size of smaller matrices, which means that the reduction of complexity comes at the cost of performance loss. Apparently, the size of the smaller matrices can be set to be 1, so that the matrix inverses are avoided and we call it scalar-form MF SBL algorithm. Recently, the scalar-form MF SBL algorithm was used for channel gain and delay estimation in [19]. We note that an efficient hyperprior $p(\boldsymbol{\alpha})$ with 2-L structure was proposed in [6], which performs better than the 2-L and 3-L structures in [17].

Different from MF which supposes all the beliefs of variable nodes are independent, BP considers the joint belief of variable nodes neighbouring a factor node and makes the most of their correlation. BP, which may achieve exact Bayesian inference, is efficient to deal with discrete probability models and linear Gaussian models. However, BP may have a high complexity, when especially dealing with models involving both discrete and continuous random variables. Recently, a unified message passing framework was proposed in [20] where BP and MF are merged to keep the merits of BP and MF while avoid their drawbacks.

In this work, a low complexity BP-MF SBL algorithm with a 2-L hierarchical prior is proposed. Instead of using the conventional factor graph shown in Fig. D.1(a), we modify the factor graph by adding a number of extra hard constraint factors as shown in Fig. D.1(b), i.e., the factor graph looks like being 'stretched'. The hard constraint factors seem redundant, which however facilitates the use of BP in the graph, leading to considerable performance improvement. As we assume that the noise variance λ^{-1} is unknown, MF can be used to tackle the exponential factors, while BP is used to handle the hard constraint factors. As we factorize the signal $\boldsymbol{\alpha}$ in a scalar form, the developed BP-MF SBL algorithm avoids matrix inversion and has a low complexity. Inspired by the derivation of the generalized approximate message passing (GAMP) [21], we further simplify the BP message passing by ignoring some minimal terms and develop an approximate BP-MF SBL algorithm. Numerical examples show that the proposed BP-MF SBL algorithms provide even better mean-square-error (MSE) performance with much lower complexity compared with the vector-form MF SBL algorithm [17], and achieve noticeable MSE performance gain with similar complexity compared with the scalar-form MF SBL algorithm [18, 19].

Notation- Boldface lowercase and uppercase letters denote vectors and matrices, respectively. The expectation operator with respect to a pdf $g(x)$ is expressed by $\langle f(x)\rangle_{g(x)} = \int f(x)g(x)dx / \int g(x')dx'$, while $\text{var}[x]_{g(x)} = \langle |x|^2\rangle_{g(x)} - |\langle x\rangle_{g(x)}|^2$ stands for the variance. The pdf of a complex Gaussian distribution with mean μ and variance ν is represented by $\mathcal{CN}(x;\mu,\nu)$. The relation $f(x) = cg(x)$ for some positive constant c is written as $f(x) \propto g(x)$.

2 Factor Graph Model

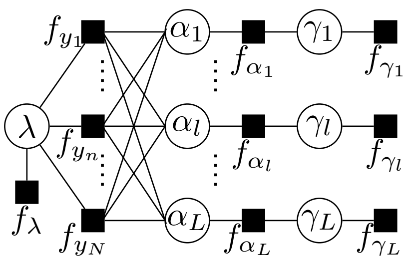
The joint a posteriori pdf of $\boldsymbol{\alpha}, \boldsymbol{\gamma}$ and λ in (D.1) with a 2-L hierarchical prior [9] can be factorized as

$$p(\boldsymbol{\alpha},\boldsymbol{\gamma},\lambda|\boldsymbol{y}) \propto f_\lambda(\lambda)\prod_n f_{y_n}(\boldsymbol{\alpha},\lambda)\prod_l f_{\alpha_l}(\alpha_l,\gamma_l)f_{\gamma_l}(\gamma_l), \tag{D.2}$$

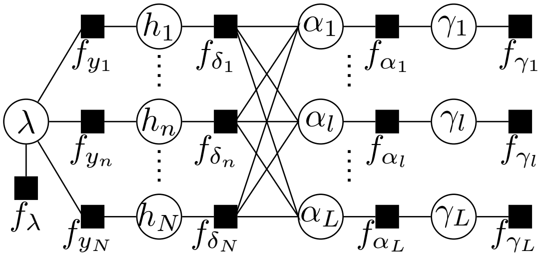
where $f_{y_n}(\boldsymbol{\alpha},\lambda) \triangleq p(y_n|\boldsymbol{\alpha},\lambda) = \mathcal{CN}(y_n;\boldsymbol{\Phi}_n\boldsymbol{\alpha},\lambda^{-1})$, with $\boldsymbol{\Phi}_n$ being the n-th row of matrix $\boldsymbol{\Phi}$, and $f_\lambda(\lambda)$ denotes the prior of noise precision parameter λ. The factor $f_{\alpha_l}(\alpha_l,\gamma_l)$ denotes the conditional pdf $p(\alpha_l|\gamma_l) = \mathcal{CN}(\alpha_l;0,\gamma_l^{-1})$, which is chosen as a Gaussian prior of α_l and $f_{\gamma_l}(\gamma_l)$ represents a hyperprior $p(\gamma_l) = \text{Ga}(\gamma_l;\epsilon,\eta)$[1] of the hyperparameter γ_l. The factorization in (D.2) can be visually depicted on the factor graph [22] shown in Fig. D.1(a), which is similar to those in [18] and [19]. We assume that λ is unknown, and MF can be used to deal with factor nodes $\{f_{y_n}, \forall n \in [1:N]\}$, which leads to the scalar-form MF SBL algorithm [18]. In [17], the vector-form MF SBL algorithm is derived based on a conventional factor graph, where the vector $\boldsymbol{\alpha}$ is treated as a single variable node.

To facilitate the use of both BP and MF, we modify the factor graph in Fig. D.1(a) by adding hard constraint factors $\{f_{\delta_n}(h_n,\boldsymbol{\alpha}) = \delta(h_n - \boldsymbol{\Phi}_n\boldsymbol{\alpha}), \forall n \in [1:N]\}$ with a new variable vector $\boldsymbol{h} = \boldsymbol{\Phi}\boldsymbol{\alpha}$. Therefore, factor f_{y_n} denotes the likelihood function $p(y_n|h_n,\lambda) = \mathcal{CN}(y_n;h_n,\lambda^{-1})$. The new factor graph, shown in Fig. D.1(b), looks like a stretched version of the graph in Fig. D.1(a). In the new graph, MF rules with fixed points equations can be used to compute the messages for the exponential factors, while BP rules, often yielding better performance, can be used to deal with the hard constraint factors. The message computations and scheduling are detailed in the following section.

[1] $\text{Ga}(\cdot;a,b)$ denotes a Gamma pdf with shape parameter a and rate parameter b. Note that, as in [6], we use the Gama prior for the parameter of precision, rather than for variance [17].



(a) Conventional factor graph



(b) Stretched factor graph

Fig. D.1: Two factor graph representations for the probabilistic model (D.2).

3 BP-MF Based SBL

In this section, with the combined BP-MF message update rule [20], we detail the message computations and scheduling on the factor graph shown in Fig. D.1(b) to perform sparse signal estimation. All the factors in Fig. D.1(b) are represented by set $\boldsymbol{A}$, and it is divided into two disjoint subsets, a BP subset and an MF subset, which are denoted by $\boldsymbol{A}_{\mathrm{BP}} = \{f_{\delta_n}, \forall n\}$ and $\boldsymbol{A}_{\mathrm{MF}} = \boldsymbol{A} \setminus \boldsymbol{A}_{\mathrm{BP}}$, respectively.

3.1 Message Computations

The computations for messages passing from left to right (forward) and from right to left (backward) are elaborated. The computations of some forward messages may need relevant backward messages, which we assume are produced from the previous iteration.

Froward message computations

Assuming that the belief $b(\lambda)$, later defined in (D.24), of noise precision λ is known, the message $m_{f_{y_n}\to h_n}(h_n)$ from observation factor $f_{y_n} \in \boldsymbol{A}_{\mathrm{MF}}$ to h_n is calculated by the MF rule, as follows

$$\begin{aligned} m_{f_{y_n}\to h_n}(h_n) &= \exp\left\{\langle \log f_{y_n}(h_n,\lambda)\rangle_{b(\lambda)}\right\} \\ &\propto \mathcal{CN}\left(h_n; y_n, \hat{\lambda}^{-1}\right), \end{aligned} \tag{D.3}$$

where $\hat{\lambda} = \langle\lambda\rangle_{b(\lambda)}$.

The message $m_{f_{\delta_n}\to\alpha_l}(\alpha_l)$ from the hard factor $f_{\delta_n} \in \boldsymbol{A}_{\mathrm{BP}}$ to variable node α_l is computed by the BP rule with the messages $n_{h_n\to f_{\delta_n}}(h_n) = m_{f_{y_n}\to h_n}(h_n)$ and $\{n_{\alpha_{l'}\to f_{\delta_n}}(\alpha'_l), \forall l' \neq l\}$, later defined in (D.18), yielding

$$\begin{aligned} m_{f_{\delta_n}\to\alpha_l}(\alpha_l) &= \langle f_{\delta_n}(h_n,\boldsymbol{\alpha})\rangle_{n_{h_n\to f_{\delta_n}}(h_n)\prod_{l'\neq l} n_{\alpha_{l'}\to f_{\delta_n}}(\alpha_{l'})} \\ &\propto \mathcal{CN}\left(\alpha_l; \hat{\alpha}_{n\to l}, \nu_{\alpha_{n\to l}}\right), \end{aligned} \tag{D.4}$$

where

$$\hat{\alpha}_{n\to l} \triangleq \frac{y_n - \hat{p}_n + \Phi_{nl}\hat{\alpha}_{l\to n}}{\Phi_{nl}} \tag{D.5}$$

$$\nu_{\alpha_{n\to l}} \triangleq \frac{\hat{\lambda}^{-1} + \nu_{p_n} - |\Phi_{nl}|^2\nu_{\alpha_{l\to n}}}{|\Phi_{nl}|^2} \tag{D.6}$$

$$\hat{p}_n \triangleq \sum_l \Phi_{nl}\hat{\alpha}_{l\to n} \tag{D.7}$$

$$\nu_{p_n} \triangleq \sum_l |\Phi_{nl}|^2\nu_{\alpha_{l\to n}}. \tag{D.8}$$

For convenience of description, the product of all the Gaussian messages $\{m_{f_{\delta_n}\to\alpha_l}(\alpha_l), \forall n \in [1:N]\}$ is denoted by

$$\begin{aligned} q_l(\alpha_l) &= \prod_n m_{f_{\delta_n}\to\alpha_l}(\alpha_l) \\ &\propto \mathcal{CN}\left(\alpha_l; \hat{q}_l, \nu_{q_l}\right), \end{aligned} \tag{D.9}$$

where

$$\nu_{q_l} \triangleq \left(\sum_n \frac{1}{\nu_{\alpha_{n\to l}}}\right)^{-1} \tag{D.10}$$

$$\hat{q}_l \triangleq \nu_{q_l}\left(\sum_n \frac{\hat{\alpha}_{n\to l}}{\nu_{\alpha_{n\to l}}}\right). \tag{D.11}$$

Given the message $m_{f_{\alpha_l}\to\alpha_l}(\alpha_l) \propto \mathcal{CN}\left(\alpha_l; 0, \hat{\gamma}_l^{-1}\right)$, later defined in (D.16), the belief $b(\alpha_l)$ of variable α_l is obtained as

$$\begin{aligned} b(\alpha_l) &\propto q_l(\alpha_l) m_{f_{\alpha_l}\to\alpha_l}(\alpha_l) \\ &\propto \mathcal{CN}(\alpha_l; \hat{\alpha}_l, \nu_{\alpha_l}), \end{aligned} \tag{D.12}$$

where

$$\hat{\alpha}_l \triangleq \frac{\hat{q}_l}{1+\nu_{q_l}\hat{\gamma}_l} \tag{D.13}$$

$$\nu_{\alpha_l} \triangleq \left(1/\nu_{q_l} + \hat{\gamma}_l\right)^{-1}. \tag{D.14}$$

Since the factor f_{α_l} is classified into the MF subset, the message $m_{f_{\alpha_l}\to\gamma_l}(\gamma_l)$ is calculated by using the MF rule,

$$\begin{aligned} m_{f_{\alpha_l}\to\gamma_l}(\gamma_l) &= \exp\left\{\langle \log f_{\alpha_l}(\alpha_l, \gamma_l)\rangle_{b(\alpha_l)}\right\} \\ &\propto \gamma_l \exp\left\{-\gamma_l(|\hat{\alpha}_l|^2 + \nu_{\alpha_l})\right\}, \end{aligned} \tag{D.15}$$

so that the belief $b(\gamma_l)$ of hyperparameter γ_l reads

$$\begin{aligned} b(\gamma_l) &\propto m_{f_{\alpha_l}\to\gamma_l}(\gamma_l) f_{\gamma_l}(\gamma_l) \\ &\propto \gamma_l^{\epsilon+1} \exp\left\{-\gamma_l(\eta + |\hat{\alpha}_l|^2 + \nu_{\alpha_l})\right\}. \end{aligned}$$

Backward Message

We firstly compute the message $m_{f_{\alpha_l}\to\alpha_l}(\alpha_l)$ from f_{α_l} to α_l by the MF rule, as follows

$$\begin{aligned} m_{f_{\alpha_l}\to\alpha_l}(\alpha_l) &= \exp\left\{\langle \log f_{\alpha_l}(\alpha_l, \gamma_l)\rangle_{b(\gamma_l)}\right\} \\ &\propto \mathcal{CN}\left(\alpha_l; 0, \hat{\gamma}_l^{-1}\right), \end{aligned} \tag{D.16}$$

where

$$\hat{\gamma}_l = \langle \gamma_l \rangle_{b(\gamma_l)} = \frac{\epsilon+1}{\eta + |\hat{\alpha}_l|^2 + \nu_{\alpha_l}}. \tag{D.17}$$

Since factor node $f_{\delta_n} \in \boldsymbol{A}_{\mathrm{BP}}$, the message $n_{\alpha_l\to f_{\delta_n}}(\alpha_l)$ from variable node α_l to f_{δ_n} is updated by the BP rule,

$$\begin{aligned} n_{\alpha_l\to f_{\delta_n}}(\alpha_l) &= \frac{b(\alpha_l)}{m_{f_{\delta_n}\to\alpha_l}(\alpha_l)} \\ &\propto \mathcal{CN}(\alpha_l; \hat{\alpha}_{l\to n}, \nu_{\alpha l\to n}), \end{aligned} \tag{D.18}$$

where

$$\nu_{\alpha_{l\to n}} \triangleq \left(\frac{1}{\nu_{\alpha_l}} - \frac{1}{\nu_{\alpha_{n\to l}}}\right)^{-1} \tag{D.19}$$

$$\hat{\alpha}_{l\to n} \triangleq \nu_{\alpha_{l\to n}} \left(\frac{\hat{\alpha}_l}{\nu_{\alpha_l}} - \frac{\hat{\alpha}_{n\to l}}{\nu_{\alpha_{n\to l}}}\right). \tag{D.20}$$

Then the message $m_{f_{\delta_n}\to h_n}(h_n)$ can be computed with the BP rule for $f_{\delta_n} \in \boldsymbol{A}_{\mathrm{BP}}$, yielding

$$\begin{aligned} m_{f_{\delta_n}\to h_n}(h_n) &= \langle f_{\delta_n}(h_n, \boldsymbol{\alpha})\rangle_{\prod_l n_{\alpha_l\to f_{\delta_n}}(\alpha_l)} \\ &\triangleq \mathcal{CN}\left(h_n; \hat{p}_n, \nu_{p_n}\right). \end{aligned} \tag{D.21}$$

We compute the belief $b(h_n)$ of variable h_n by

$$\begin{aligned} b(h_n) &\propto m_{f_{\delta_n}\to h_n}(h_n) n_{h_n\to f_{\delta_n}}(h_n) \\ &\propto \mathcal{CN}(h_n; \hat{h}_n, \nu_{h_n}), \end{aligned}$$

where

$$\nu_{h_n} \triangleq \left(\hat{\lambda} + 1/\nu_{p_n}\right)^{-1} \tag{D.22}$$

$$\hat{h}_n \triangleq \nu_{h_n}\left(y_n\hat{\lambda} + \hat{p}_n/\nu_{p_n}\right). \tag{D.23}$$

The message $m_{f_{y_n}\to\lambda}(\lambda) \propto \lambda \exp\{-\langle |y_n - h_n|^2\rangle_{b(h_n)}\}$ is calculated by the MF rule. With the conjugate prior pdf $f_\lambda(\lambda) \propto 1/\lambda$, the belief $b(\lambda)$ is updated by

$$\begin{aligned} b(\lambda) &\propto m_{f_{y_n}\to\lambda}(\lambda) f_\lambda(\lambda) \\ &\propto \lambda^{N-1} \exp\left\{-\lambda \sum_n \left\langle |y_n - h_n|^2 \right\rangle_{b(h_n)}\right\} \end{aligned} \tag{D.24}$$

and the parameter $\hat{\lambda}$ in (D.3) is computed as

$$\hat{\lambda} = \langle\lambda\rangle_{b(\lambda)} = \frac{N}{\sum_n \langle |y_n - h_n|^2\rangle_{b(h_n)}}. \tag{D.25}$$

3.2 Message Scheduling for BP-MF SBL Algorithm

The factors in Fig. D.1(b) are very densely connected and thus there are a multitude of different options for message scheduling. In this paper, we simply choose a schedule, where the messages are sequentially updated in both forward and backward directions, while the messages in vertical direction are simultaneously computed for all $n \in [1:N]$ and $l \in [1:L]$. The BP-MF SBL algorithm with such scheduling is summarized in **Algorithm 4**.

Algorithm 4 BP-MF SBL Algorithm

1: Initialize $\hat{p}_n$, ν_{p_n}, $\hat{\alpha}_{l\to n}$, $\nu_{\alpha_{l\to n}}$, $\hat{\gamma}_l$, $\forall n$, $\forall l$ and $\hat{\lambda}$.
2: **for** $t = 1 \to$ # of Iterations **do**
3: $\quad\forall n, l$: update $\hat{\alpha}_{n\to l}$ and $\nu_{\alpha_{n\to l}}$ by (D.5) and (D.6).
4: $\quad\forall l$: update ν_{q_l} and $\hat{q}_l$ by (D.10) and (D.11).
5: $\quad\forall l$: update $\hat{\alpha}_l$ and ν_{α_l} by (D.13) and (D.14).
6: $\quad\forall l$: update $\hat{\gamma}_l$ by (D.17).
7: $\quad\forall l$: update $\hat{\alpha}_l$ and ν_{α_l} again, by (D.13) and (D.14).
8: $\quad\forall n, l$: update $\nu_{\alpha_{l\to n}}$ and $\hat{\alpha}_{l\to n}$ by (D.19) and (D.20).
9: $\quad\forall n$: update $\hat{p}_n$ and ν_{p_n} by (D.7) and (D.8).
10: $\quad\forall n$: update ν_{h_n} and $\hat{h}_n$ by (D.22) and (D.23).
11: $\quad$update $\hat{\lambda}$ by (D.25), with $b(h_n) = \mathcal{CN}(h_n; \hat{h}_n, \nu_{h_n})$.
12: **end for** t

4 Approximate BP-MF SBL

It is observed that there are NL edges between variable nodes $\{\alpha_l, \forall l\}$ and factor nodes $\{f_{\delta_n}, \forall n\}$, so we have to compute $2NL$ messages (see **Lines 3** and **8** in **Algorithm 4**) for both forward and backward directions in each iteration. To simplify the BP-MF SBL, we approximate the means and variances of Gaussian messages in the BP part by eliminating some small terms, leading to the approximate BP-MF SBL algorithm.

4.1 Approximation of Messages

By substituting (D.14) into (D.19),

$$\nu_{\alpha_{l\to n}} = \left(1/\nu_{q_l} + \hat{\gamma}_l - 1/\nu_{\alpha_{n\to l}}\right)^{-1} \approx \nu_{\alpha_l} \tag{D.26}$$

can be obtained as $1/\nu_{q_l} \gg 1/\nu_{\alpha_{n\to l}}$ from (D.10) when the number N is large enough. Similarly, substituting (D.5) and (D.6) into (D.20), yields[2]

$$\begin{aligned}
\hat{\alpha}_{l\to n} &= \nu_{\alpha_{l\to n}} \left(\frac{\hat{\alpha}_l}{\nu_{\alpha_l}} - \frac{\Phi_{nl}^*(y_n - \hat{p}_n + \Phi_{nl}\hat{\alpha}_{l\to n}^{t-1})}{\hat{\lambda}^{-1} + \nu_{p_n} - |\Phi_{nl}|^2 \nu_{\alpha_{l\to n}}^{t-1}} \right) \\
&\approx \hat{\alpha}_l - \nu_{\alpha_l} \frac{y_n - \hat{p}_n}{\hat{\lambda}^{-1} + \nu_{p_n}} \Phi_{nl}^* \\
&= \hat{\alpha}_l - \nu_{\alpha_l} s_n \Phi_{nl}^*,
\end{aligned} \tag{D.27}$$

where

$$s_n \triangleq \frac{y_n - \hat{p}_n}{\hat{\lambda}^{-1} + \nu_{p_n}}. \tag{D.28}$$

[2]To distinguish the parameters of messages in different iterations, we append a superscript $(t-1)$ to denote the index of the previous iteration.

The above approximation is made by assuming that the length L of variable vector $\boldsymbol{\alpha}$ is very large, so that $\hat{p}_n \gg \Phi_{nl}\hat{\alpha}_{l\to n}$ and $\nu_{p_n} \gg |\Phi_{nl}|^2 \nu_{\alpha_{l\to n}}$ from (D.7) and (D.8).

Substituting (D.26) and (D.27) into (D.8) and (D.7) respectively, we obtain the approximate variance and mean

$$\begin{aligned}
\nu_{p_n} &\approx \sum_l |\Phi_{nl}|^2 \nu_{\alpha_l} && \text{(D.29)}\\
\hat{p}_n &\approx \sum_l \Phi_{nl}\left(\hat{\alpha}_l - \nu_{\alpha_l} s_n \Phi_{nl}^*\right) \\
&\overset{\text{(D.29)}}{\approx} \sum_l \Phi_{nl}\hat{\alpha}_l - s_n \nu_{p_n}. && \text{(D.30)}
\end{aligned}$$

We further substitute (D.6) and (D.5) into (D.10) and (D.11), and approximate them for a large L, as follows,

$$\begin{aligned}
\nu_{q_l} &= \left(\sum_n \frac{|\Phi_{nl}|^2}{\hat{\lambda}^{-1} + \nu_{p_n} - |\Phi_{nl}|^2 \nu_{\alpha_{l\to n}}}\right)^{-1} \\
&\approx \left(\sum_n \frac{|\Phi_{nl}|^2}{\hat{\lambda}^{-1} + \nu_{p_n}}\right)^{-1} && \text{(D.31)}\\
\hat{q}_l &= \nu_{q_l}\left(\sum_n \frac{\Phi_{nl}^*(y_n - \hat{p}_n + \Phi_{nl}\hat{\alpha}_{l\to n})}{\hat{\lambda}^{-1} + \nu_{p_n} - |\Phi_{nl}|^2 \nu_{\alpha_{l\to n}}}\right) \\
&\overset{\text{(D.28)}}{\approx} \nu_{q_l}\sum_n \left(\Phi_{nl}^* s_n + \frac{|\Phi_{nl}|^2}{\hat{\lambda}^{-1} + \nu_{p_n}}\hat{\alpha}_{l\to n}\right) \\
&\overset{\text{(D.27)(D.31)}}{\approx} \hat{\alpha}_l + \nu_{q_l}\sum_n \Phi_{nl}^*\left(s_n - \frac{|\Phi_{nl}|^2}{\hat{\lambda}^{-1} + \nu_{p_n}}\nu_{\alpha_l} s_n\right) \\
&\approx \hat{\alpha}_l + \nu_{q_l}\sum_n \Phi_{nl}^* s_n. && \text{(D.32)}
\end{aligned}$$

The approximation in (D.32) is according to $\frac{|\Phi_{nl}|^2}{\hat{\lambda}^{-1}+\nu_{p_n}} \ll \nu_{\alpha_l}^{-1}$, since $\nu_{\alpha_l}^{-1} = \sum_n \frac{|\Phi_{nl}|^2}{\hat{\lambda}^{-1}+\nu_{p_n}} + \hat{\gamma}_l$ is obtained by inserting (D.31) into (D.14).

4.2 Message Scheduling for Approximate BP-MF SBL Algorithm

We choose the similar message scheduling to BP-MF SBL shown in **Algorithm 4**, where the corresponding parameters are replaced by the above approximate computations. The parameters ν_{q_l} and $\hat{q}_l$ are updated by (D.31) and (D.32) instead of (D.10) and (D.11). The parameters ν_{p_n} and $\hat{p}_n$ are calculated by (D.29) and (D.30) rather than (D.8) and (D.7). In addition, the computations of parameters $\hat{\alpha}_{n\to l}$, $\nu_{\alpha_{n\to l}}$, $\nu_{\alpha_{l\to n}}$ and $\hat{\alpha}_{l\to n}$ in **Lines 3** and **8** of **Algorithm 4** are avoided, while a set of intermediate parameters s_n, $\forall n$, have to be inserted. We summarize the approximate BP-MF SBL in **Algorithm 5**. It is interesting that the message computations for the densely connected BP subgraph as shown in Fig. D.1(b) coincide with the GAMP [21] algorithm.

Algorithm 5 Approximate BP-MF SBL Algorithm

Initialize ν_{p_n}, s_n, $\forall n$, $\hat{\alpha}_l$, $\hat{\gamma}_l$, $\forall l$ and $\hat{\lambda}$.
for $t = 1 \to$ # of Iterations **do**
    $\forall l$: update ν_{q_l} and $\hat{q}_l$ by (D.31) and (D.32).
    $\forall l$: update $\hat{\alpha}_l$ and ν_{α_l} by (D.13) and (D.14).
    $\forall l$: update $\hat{\gamma}_l$ by (D.17).
    $\forall l$: update $\hat{\alpha}_l$ and ν_{α_l} again, by (D.13) and (D.14).
    $\forall n$: update ν_{p_n} and $\hat{p}_n$ by (D.30) and (D.29).
    $\forall n$: update s_n by (D.28).
    $\forall n$: update ν_{h_n} and $\hat{h}_n$ by (D.22) and (D.23).
    update $\hat{\lambda}$ by (D.25), with $b(h_n) = \mathcal{CN}(h_n; \hat{h}_n, \nu_{h_n})$.
end for t

5 Numerical Simulation Results

In this section, we assess the proposed SBL algorithms by means of Monte Carlo simulations. Consider the sparse signal model (D.1) with a random $M \times N (M = 100, N = 200)$ dictionary matrix $\mathbf{\Phi}$, whose entries are independent and identically distributed (i.i.d.) zero-mean complex Gaussian random variables with unit variance. We assume that the length-N vector $\boldsymbol{\alpha}$ has K nonzero elements which are randomly dispersed in vector $\boldsymbol{\alpha}$. In addition, the nonzero elements are i.i.d. and also drawn from a zero-mean complex Gaussian distribution with unit variance. All curves are produced based on 200 Monte-Carlo runs, and for each run with a new realization of the dictionary matrix $\mathbf{\Phi}$, the vector $\boldsymbol{\alpha}$ and the AWGN vector $\boldsymbol{\omega}$ are generated.

We compare the MSE performance of our proposed algorithms and the state-of-the-art algorithms. "BP-MF" and "A-BP-MF" denotes our proposed BP-MF and approximate BP-MF SBL algorithms, i.e., **Algorithms 4** and **5**, respectively. "MF-vector" and "MF-scalar" stand for MF SBL algorithms in vector-form [17] and in scalar-form (sequentially estimating each element of the sparse signal $\boldsymbol{\alpha}$) [19], respectively. For a fair companion, all the above algorithms use 2-L hierarchical structure with the hyperprior proposed in [6]. In addition, we also provide the performance of the vector-form MF algorithm using 3-L hierarchical prior in [17], denoted by "MF-vector-3L".

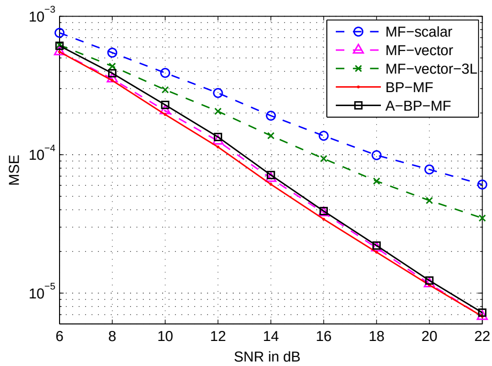


Fig. D.2: MSE performance of different algorithms, where $K = 26$.

In Fig. D.2, the MSE performance of the algorithms is shown over a wide range of signal-to-noise ratios (SNRs), where all algorithms run 20 iterations and the number of nonzero elements $K = 26$. We can observe that the proposed BP-MF and A-BP-MF algorithms deliver slightly better MSE performance than MF-vector, and significantly outperform MF-scalar and MF-vector-3L. Fig. D.3 depicts MSE performance with an SNR of 14dB versus the number of non-zero elements K. It shows that all the algorithms have similar performance when K is small. However, with the increase of K, the MF SBL algorithms exhibit considerable performance loss compared to the proposed BP-MF and A-BP-MF SBL algorithms. It is also seen that BP-MF performs slightly better than A-BP-MF.

Fig. D.4 illustrates the convergence of the algorithms, where SNR $= 14$dB and $K = 26$. We can see that MF-scalar has the fastest convergence rate due to its sequential message updating schedule. Our proposed BP-MF algorithms converge slower but achieve better MSE performance compared to MF-salar and MF-vecotr-3L. It can also be seen that our proposed algorithms have similar convergence rate and performance compared to MF-vector.

In addition, our simulation results in Figs. D.2, D.3 and D.4 also show that the 2-L hierarchical priori structure proposed in [6] outperforms 3-L hierarchical priori structure [17].

5.1 Computational Complexity

As the message computations for updating λ and γ_l are the same for all the algorithms, we only analyze the complexity of message computations related to $\boldsymbol{h}$ and $\boldsymbol{\alpha}$. Due to the matrix inversion involved, MF-vector has a complexity of $\mathcal{O}(L^3)$ per iteration, while MF-scalar $\mathcal{O}(NL)$. Since the proposed BP-MF and A-BP-MF algorithms using scalar-form factor graph shown in

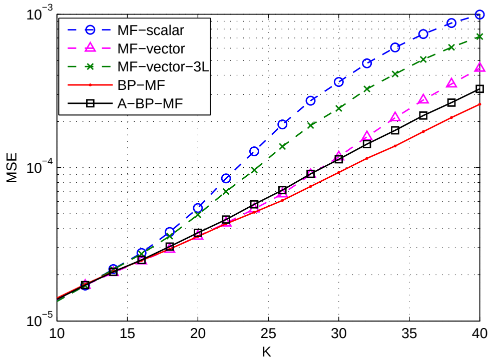


Fig. D.3: MSE performance companions with number of nonzero components K, where SNR $= 14$dB.

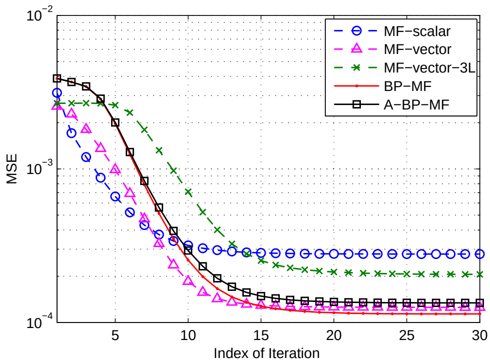


Fig. D.4: MSE performance versus iteration index, where $K = 26$ and SNR $= 14$dB.

Fig. D.1(b), they have similar complexity to MF-scalar. In details, BP-MF needs to compute $\mathcal{O}(NL)$ messages and $\mathcal{O}(NL)$ memory cells to store the parameters (means and variances) of messages (see **Lines 3** and **8** in **Algorithm 4**), while MF-scalar and A-BP-MF only need to update and store $\mathcal{O}(N + L)$ messages. However, in updating the belief $b(\alpha_l), \forall l \in [1 : L]$ MF-scalar with sequential message schedule may take longer running time than BP-MF algorithms.

6 Conclusion

In this paper, we have investigated message passing based approaches to SBL. Two low complexity BP-MF SBL algorithms have been proposed based on a stretched factor graph which is obtained by adding extra hard constraint factors to the conventional factor graph. It has been shown that the BP-MF SBL algorithms outperform the state-of-the-art MF SBL algorithms in terms of computational complexity or performance.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (NSFC 61172086, NSFC U1204607, NSFC 61201251).

References

[1] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inform. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[2] E. Candès and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Mag.*, vol. 25, no. 2, pp. 21–30, March 2008.

[3] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. Signal Processing*, vol. 52, no. 8, pp. 2153–2164, Aug. 2004.

[4] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basic pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, Aug. 1998.

[5] J. A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Trans. Inform. Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.

[6] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, June 2001.

[7] M. E. Tipping and A. C. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," *Proc. 2003 9th international Workshop on Artificial Intelligence and Statistics,*, 2003.

[8] D. Shutin, T. Buchgraber, S. R. Kulkarni, and H. V. Poor, "Fast variational sparse Bayesian learning with automatic relevance determination for superimposed signals," *IEEE Trans. Signal Processing*, vol. 59, no. 12, pp. 6257–6261, Dec. 2011.

[9] N. L. Pedersen, C. Navarro Manchón, M.-A. Badiu, D. Shutin, and B. H. Fleury, "Sparse estimation using Bayesian hierarchical prior modeling for real and complex linear models," *Signal Processing*, vol. 115, no. 0, pp. 94 – 109, Oct. 2015.

[10] X. Tan and J. Li, "Computationally efficient sparse Bayesian learning via belief propagation," in *2009 Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*, Nov. 2009, pp. 1566–1570.

[11] D. Baron, S. Sarvotham, and R. Baraniuk, "Bayesian compressive sensing via belief propagation," *IEEE Trans. Signal Processing*, vol. 58, no. 1, pp. 269–280, Jan. 2010.

[12] S. Som and P. Schniter, "Compressive imaging using approximate message passing and a Markov-tree prior," *IEEE Trans. Signal Processing*, vol. 60, no. 7, pp. 3439–3448, July 2012.

[13] M. Al-Shoukairi and B. Rao, "Sparse Bayesian learning using approximate message passing," in *2014 48th Asilomar Conference on Signals, Systems and Computers*, Nov. 2014, pp. 1957–1961.

[14] E. P. Xing, M. I. Jordan, and S. Russell, "A generalized mean field algorithm for variational inference in exponential families," in *Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence*, ser. UAI'03, San Francisco, CA, USA, 2003, pp. 583–591.

[15] C. M. Bishop and J. Winn, "Structured variational distributions in VIBES," *Proceedings Artificial Intelligence and Statistics*, pp. 3–6, 2003.

[16] J. Dauwels, "On variational message passing on factor graphs," in *Proc. IEEE International Symposium on Information Theory (ISIT 2007)*, Jun. 2007, pp. 2546–2550.

[17] N. L. Pedersen, C. Navarro Manchón, D. Shutin, and B. H. Fleury, "Application of Bayesian hierarchical prior modeling to sparse channel estimation," pp. 3487–3492, June 2012.

[18] N. L. Pedersen, C. Navarro Manchón, and B. H. Fleury, "Low complexity sparse Bayesian learning for channel estimation using generalized mean field," *20th European Wireless Conference*, pp. 838–843, June 2014.

[19] T. L. Hansen, P. B. Jørgensen, M. Badiu, and B. H. Fleury, "Joint sparse channel estimation and decoding: Continuous and discrete domain sparsity," *CoRR*, vol. abs/1507.02954, 2015. [Online]. Available: http://arxiv.org/abs/1507.02954

[20] E. Riegler, G. E. Kirkelund, C. Navarro Manchón, M.-A. Badiu, and B. H. Fleury, "Merging belief propagation and the mean field approximation: A free energy approach," *IEEE Trans. Inform. Theory*, vol. 59, no. 1, pp. 588–602, Jan. 2013.

[21] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT 2011)*, Aug. 2011, pp. 2168–2172.

[22] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

Paper E

A Low Complexity OFDM Receiver with Combined GAMP and MF Message Passing

Zhengdao Yuan, Chuanzong Zhang, Zhongyong Wang,
Qinghua Guo, Sheng Wu and Xingye Wang

The paper has been submitted to the
IEEE Transactions on Vehicular Technology, April 2016.

The layout has been revised.

Abstract

With a unified belief propagation (BP) and mean field (MF) framework, we propose an iterative message passing receiver, which performs joint channel state and noise precision (the reciprocal of noise variance) estimation and decoding for OFDM systems. The recently developed generalized approximate message passing (GAMP) is incorporated to the BP-MF framework, where MF is used to handle observation factor nodes with unknown noise precision and GAMP is used for channel estimation in the time-frequency domain. Compared to state-of-the-art algorithms in the literature, the proposed algorithm either delivers similar performance with much lower complexity, or delivers much better performance with similar complexity. In addition, the proposed algorithm exhibits fastest convergence.

1 Introduction

Due to the excellent performance, especially when applied to discrete probabilistic models, belief propagation (BP) [1] on factor graphs has attracted much attention in the design of iterative receivers for communication systems [2–4]. However, BP may suffer from high or even intractable computational complexity in certain applications [5]. An alternative to BP is the mean field (MF) approximation (also known as variational message passing) [6], which can efficiently deal with continuous probabilistic models involving probability density functions (pdfs) belonging to an exponential family. Another notable approximate inference technique is expectation propagation (EP) [7], which can be seen as an approximation of BP where some beliefs are approximated by pdfs in a specific exponential family. Recently, to take advantage of the merits of different message passing techniques, unified message passing frameworks have been investigated and applied to low complexity communication receiver design, e.g., the combined BP-EP receivers in [8, 9] and the combined BP-MF receivers in [5, 10–13].

With the unified BP and MF framework in [5], a message passing OFDM receiver for joint channel estimation and decoding was proposed in [10], which involves high computational complexity due to the operation of a large matrix inversion required in each iteration. An alternative message passing receiver that allows flexible complexity-performance trade-off was proposed in [11], where groups of contiguous channel weights are assumed to obey a Markov model, leading to an algorithm whose complexity is adjustable by changing the size of each group. In addition, the noise precision is treated as a random variable and estimated by using MF. A similar method for noise precision estimation was also used in [12] and [13].

With combined BP and EP, an OFDM receiver performing joint channel estimation and decoding was designed in [8], where the recently developed generalized approximate message passing (GAMP) [14] is employed to reduce the complexity. GAMP was firstly used in [15] for sparse channel estimation (jointly performed with decoding) in OFDM systems. Compared to the algorithm in [10], the algorithm in [8] achieves better performance with lower complexity. However, the precision of the noise is assumed to be known at receivers in [8] and [15], and the extension of the receivers to the case of unknown noise precision is not straightforward.

This work concerns the design of message passing receiver for joint channel estimation and decoding with unknown noise precision. The unified MF and BP framework in [5] is used, and the GAMP algorithm is incorporated into the BP-MF framework to significantly reduce the complexity. With a stretched factor graph which is inspired by [16], modified GAMP is developed to handle a densely connected subgraph (functioning as channel estimation). In addition, MF is used to deal with observation factor nodes with unknown noise precision, while BP is used for the subgraph of demodulation and decoding. Compared to the BP-EP receiver in [8], the proposed receiver has the capability of noise precision estimation, and can achieve the same performance as the receiver in [8] with known noise precision. Compared to the state-of-the-art BP-MF receivers in [10], the proposed receiver delivers same performance while with much lower complexity. In addition, the proposed receiver can achieve much better performance than the receiver in [9] (for a fair comparison, the group size of the receiver in [11] is adjusted so that it has similar complexity to the proposed receiver). It is also shown that the proposed receiver exhibits fastest convergence compared to the receivers in [8], [10] and [11].

This paper is organized as follows. In Section 2, the OFDM system model is described and a factor graph representation is presented. The new low complexity OFDM receiver is proposed in Section 3. Performance comparisons and complexity analyses are provided in Section 4 and conclusions are drawn in Section 5.

Notation- Boldface lower-case and upper-case letters denote vectors and matrices, respectively. Superscripts $(\cdot)^*$ and $(\cdot)^{\mathrm{T}}$ represent conjugation and transposition, respectively. The expectation operator with respect to a density $g(x)$ is expressed by $\langle f(x)\rangle_{g(x)} = \int f(x)g(x)dx / \int g(x')dx'$. The probability density function (pdf) of a complex Gaussian distribution with mean $\hat{x}$ and variance ν_x is represented by $\mathcal{CN}(x; \hat{x}, \nu_x)$. The relation $f(x) = cg(x)$ for some positive constant c is written as $f(x) \propto g(x)$. The notation $\odot$ represents the element-wise product between two vectors.

2 System Model

Consider an OFDM system employing N data and P pilot subcarriers with disjoint sets of indices $\mathcal{D}$ and $\mathcal{P}$, respectively, where $\mathcal{D} \cup \mathcal{P} = [1 : N + P]$ and $\mathcal{D} \cap \mathcal{P} = \emptyset$. A sequence of K information bits $\boldsymbol{b} = \{b_k, k = 1, ...K\}$ is encoded and interleaved using a rate $R = K/(NQ)$ channel code and a random interleaver, yielding an interleaved codeword vector $\boldsymbol{c}$, where Q denotes the order of modulation. Q coded bits in each sub-vector $\boldsymbol{c}_n$ are mapped to a data symbol $x_{i_n} \in \mathcal{S}_{\mathrm{D}}$, $i_n \in \mathcal{D}$, where $\mathcal{S}_{\mathrm{D}}$ denotes modulation alphabet of size 2^Q. The data symbols $\{x_i, i \in \mathrm{D}\}$ are multiplexed with pilot symbols $\{x_j, i \in \mathrm{P}\}$ which are randomly selected from $\mathcal{S}_{\mathrm{P}}$, resulting in a vector of transmitted symbols $\boldsymbol{x} = \{x_i, i \in \mathcal{D} \cup \mathcal{P}\}^{\mathrm{T}}$. The transmitted symbols are modulated by IFFT and then a cyclic prefix (CP) is added before transmission through a wireless channel with L taps, $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_L)^{\mathrm{T}}$.

After the removal of CP and the Fourier transform at the receiver side, the received signal in the frequency domain can be represented as

$$\boldsymbol{y} = \boldsymbol{h} \odot \boldsymbol{x} + \boldsymbol{\omega} \tag{E.1}$$

where $\boldsymbol{h} = \boldsymbol{\Phi}\boldsymbol{\alpha}$ stands for the vector of frequency-domain channel weights, $\boldsymbol{\Phi}$ represents the first L columns of a $(N + P) \times (N + P)$ discrete Fourier transform matrix, and $\boldsymbol{\omega}$ is an AWGN vector with zero mean and covariance matrix $\lambda^{-1}\boldsymbol{I}$.

2.1 Probabilistic Formulation and Factor Graph Representation

The joint pdf of the collection of observed and unknown variables in the OFDM system can be factorized as

$$\begin{aligned} p(\boldsymbol{y}, \boldsymbol{h}, \boldsymbol{x}, \boldsymbol{c}, \boldsymbol{b}, \lambda) &= f_{\mathrm{M}}(\boldsymbol{x}, \boldsymbol{c}, \boldsymbol{b}) f_\lambda(\lambda) \prod_{i \in \mathcal{D}} f_{\mathrm{D}_i}(x_i, h_i, \lambda) \\ &\times \prod_{j \in \mathcal{P}} f_{\mathrm{P}_j}(h_j, \lambda) \prod_{i \in \mathcal{D} \cup \mathcal{P}} f_{\delta_i}(h_i, \boldsymbol{\alpha}) \prod_{l \in [1:L]} f_{\alpha_l}(\alpha_l) \end{aligned} \tag{E.2}$$

where $f_{\mathrm{D}_i}(x_i, h_i, \lambda) \triangleq p(y_i|x_i, h_i, \lambda) = \mathcal{CN}(h_i x_i; y_i, \lambda^{-1})$ for all $i \in \mathcal{D}$, $f_{\mathrm{P}_j}(h_j, \lambda) \triangleq p(y_j|h_j, \lambda) = \mathcal{CN}(h_j; y_j, \lambda^{-1})$ for all $j \in \mathcal{P}$, $f_{\delta_i}(h_i, \boldsymbol{\alpha}) \triangleq p(h_i|\boldsymbol{\alpha}) = \delta(h_i - \boldsymbol{\Phi}_i\boldsymbol{\alpha})$ with $\boldsymbol{\Phi}_i$ denoting the i-th row of matrix $\boldsymbol{\Phi}$. The local function $f_{\alpha_l}(\alpha_l) \triangleq p(\alpha_l)$ represents the priori pdf of the l-th channel tap, and $f_{\mathrm{M}}(\boldsymbol{x}, \boldsymbol{c}, \boldsymbol{b})$ stands for the modulation, coding and interleaving constraints.

The factorization in (E.2) can be visually depicted by the factor graph shown in Fig. E.1, where $f_{\mathrm{M}}(\boldsymbol{x}, \boldsymbol{c}, \boldsymbol{b})$ is represented by the subgraph in the dashed box. More details about $f_{\mathrm{M}}(\boldsymbol{x}, \boldsymbol{c}, \boldsymbol{b})$ can be found in [10]. It is worth mentioning that, the factor graph used in this paper is a stretched version of

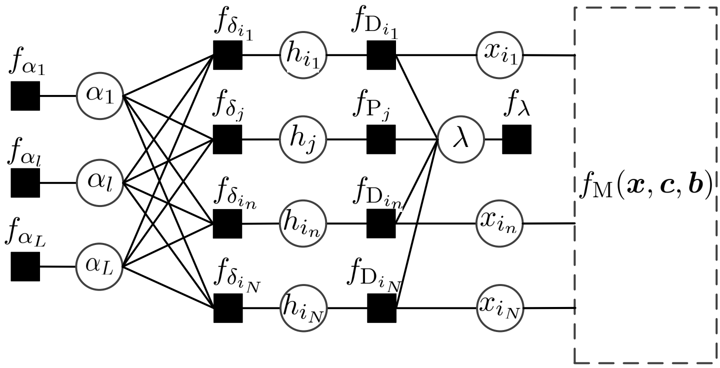


Fig. E.1: Factor graph representation for the factorization in (E.2)

that in [8] where the extra variable nodes $\{h_i\}$ and the corresponding hard constraint factor nodes $\{f_{\delta_i}\}$ are added. This enables the use of combined BP and MF message passing framework. We use MF to handle the observation nodes where the noise precision is treated as a random variable, and use GAMP for message updating in the densely connected subgraph in Fig. 1.

3 Joint Channel State and Noise Precision Estimation and Decoding

In this section, a joint channel state and noise procession estimation and decoding receiver is proposed by using the combined BP-MF message passing framework [5] on the factor graph shown in Fig. E.1.

We denote the set of all factor nodes by $\mathcal{A}$ and divide it into two disjoint subsets, an MF set $\mathcal{A}_{\mathrm{MF}} \triangleq \{f_{\mathrm{D}_i}, i \in \mathcal{D}\} \cup \{f_{\mathrm{P}_j}, j \in \mathcal{P}\}$, and a BP set $\mathcal{A}_{\mathrm{BP}} \triangleq \mathcal{A}/\mathcal{A}_{\mathrm{MF}}$. For factor nodes in the BP part, the messages are updated using the BP rule, and extrinsic messages are passed to their neighbor nodes. For factor nodes in the MF part, messages are computed by the MF rule, and beliefs are used [5].

3.1 Message Passing for Channel Estimation

It can be seen from the graph shown in Fig. 1 that there is a densely connected part between variable nodes $\{\alpha_l, \forall l \in [1:L]\}$ and factor nodes $\{f_{\delta_i}(h_i, \boldsymbol{\alpha})\}, \forall j \in \mathcal{D} \cup \mathcal{P}$. As the relevant factor nodes are in the BP node set, we propose to apply the GAMP algorithm for this part to achieve low complexity message updating. Next, we detail the computations of incoming messages and outgoing messages for this part.

Incoming message (from the observation nodes) computations

We assume that the beliefs of noise precision λ and data symbol x_i are known, which are denoted by $b(\lambda)$ and $b(x_i)$ and given in (E.12) and (E.16) respectively. Then the message $m_{f_{\mathrm{D}_i}\to h_i}(h_i)$, for $j \in \mathcal{D}$, from observation node f_{D_i} to h_i is computed by the MF rule [5] as,

$$\begin{aligned} m_{f_{\mathrm{D}_i}\to h_i}(h_i) &= \exp\left\{ \langle \log f_{\mathrm{D}_i}(h_i, x_i, \lambda) \rangle_{b(x_i)b(\lambda)} \right\} \\ &\propto \mathcal{CN}\left(h_i; \frac{y_i \langle x_i \rangle_{b(x_i)}}{\langle |x_i|^2 \rangle_{b(x_i)}}, \frac{1}{\hat{\lambda}\, \langle |x_i|^2 \rangle_{b(x_i)}} \right) \triangleq \mathcal{CN}\left(h_i; \hat{\theta}_i, \nu_{\theta_i}\right) \end{aligned} \tag{E.3}$$

where $\hat{\lambda} = \langle \lambda \rangle_{b(\lambda)}$.

Since for the observation nodes $m_{f_{\mathrm{P}_j}\to h_j}(h_j)$, for $j \in \mathcal{P}$ the value of x_j is known at the receiver, the message $m_{f_{\mathrm{P}_j}\to h_j}(h_j)$ is computed as

$$\begin{aligned} m_{f_{\mathrm{P}_j}\to h_j}(h_j) &= \exp\left\{ \langle \log f_{\mathrm{P}_j}(h_j, \lambda) \rangle_{b(\lambda)} \right\} \\ &\propto \mathcal{CN}\left(h_j; \frac{y_j}{x_j}, \frac{1}{\hat{\lambda}|x_j|^2} \right) \triangleq \mathcal{CN}\left(h_j; \hat{\theta}_j, \nu_{\theta_j}\right). \end{aligned} \tag{E.4}$$

For the convenience of description, the Gaussian messages $m_{f_{\mathrm{P}_j}\to h_j}(h_j), \forall j \in \mathcal{P}$ and $m_{f_{\mathrm{D}_i}\to h_i}(h_i), \forall i \in \mathcal{D}$ are uniformly denoted as $m_{f_{\mathrm{y}_i}\to h_i}(h_i) \propto \mathcal{CN}(h_i; \hat{\theta}_i, \nu_{\theta_i}), \forall i \in \mathcal{D} \cup \mathcal{P}$.

Outgoing message (to the observation nodes) computations

For the first iteration, we initiate the messages $f_{\delta_i \to h_i}(h_i)$, $\forall i \in \mathcal{D} \cup \mathcal{P}$ as $m_{f_{\delta_i}\to h_i}(h_i) = \mathcal{CN}(h_i; \hat{\xi}_i, \nu_{\xi_i})$, which are later updated in (E.10). The belief $b(\alpha_l)$ for $\alpha_l, \forall l \in [1:L]$ is initiated as $b(\alpha_l) \triangleq \mathcal{CN}(\alpha_l; \hat{\alpha}_l, \nu_{\alpha_l})$, which will be updated by (E.9).

We divide the computations of the messages into the following 5 steps:

S1: Using [14, Eq. (35)], the belief $b(h_i)$ of each frequency-domain channel weight h_i can be calculated as[1]

$$\begin{aligned} b(h_i) &\propto m_{f_{\mathrm{y}_i}\to h_i}(h_i) m_{f_{\delta_i}\to h_i}(h_i) \\ &\propto \mathcal{CN}\left(h_i; \hat{h}_i, \nu_{h_i}\right), \end{aligned}$$

[1]From the probabilistic understanding of message passing, the message $m_{f_{y_i}\to h_i}(h_i) \propto \mathcal{CN}(h_i; \hat{\theta}_i, \nu_{\theta_i})$ characterizes the likelihood function $p(\boldsymbol{y}|h_i)$, which is used to define function f_{out} in [14, Eq. (15a)]. Therefore, the F_{out} in [14, Eq. (26)] is given by $F_{\mathrm{out}}(h_i, \hat{\xi}_i, \nu_{\xi_i}, \hat{\theta}_i, \nu_{\theta_i}) = \log m_{f_{\mathrm{y}_i}\to h_i}(h_i) - \frac{1}{2\nu_{\xi_i}}|h_i - \hat{\xi}_i|^2$, which is equivalent to the belief $b(h_i)$ in log-domain.

where

$$\nu_{h_i} = \left(\frac{1}{\nu_{\theta_i}} + \frac{1}{\nu_{\xi_i}}\right)^{-1}; \quad \hat{h}_i = \nu_{h_i}\left(\frac{\hat{\theta}_i}{\nu_{\theta_i}} + \frac{\hat{\xi}_i}{\nu_{\xi_i}}\right). \tag{E.5}$$

S2: Compute the two intermediate parameters $\hat{s}_i$ and τ_{s_i} for each i by using [14, Eqs. (6a), (6b), (36) and (37)][2]

$$\hat{s}_i = g_{\text{out}}(\hat{\xi}_i, \nu_{\xi_i}, \hat{\theta}_i, \nu_{\theta_i}) = \frac{\hat{h}_i - \hat{\xi}_i}{\nu_{\xi_i}} \tag{E.6}$$

$$\tau_{s_i} = -\frac{\partial}{\partial \hat{\xi}_i} g_{\text{out}}(\hat{\xi}_i, \nu_{\xi_i}, \hat{\theta}_i, \nu_{\theta_i}) = \frac{1}{\nu_{\xi_i}}\left(1 - \frac{\nu_{h_i}}{\nu_{\xi_i}}\right). \tag{E.7}$$

S3: Update the variance ν_{r_l} and mean $\hat{r}_l$ of message $n_{\alpha_l \to f_{\alpha_l}}(\alpha_l) \propto \mathcal{CN}(\alpha_l; \hat{r}_l, \nu_{r_l})$ for each l by using [14, Eqs. (7a) and (7b)],

$$\nu_{r_l} = \left(\sum_{i \in \mathcal{P} \cup \mathcal{D}} \tau_{s_i}\right)^{-1}; \quad \hat{r}_l = \nu_{r_l} \sum_{i \in \mathcal{P} \cup \mathcal{D}} \hat{s}_i \Phi_{il}^* + \hat{\alpha}_l. \tag{E.8}$$

S4: With the Gaussian priori distribution of the channel tap α_l in time-domain $p(\alpha_l) = \mathcal{CN}(\alpha_l; \hat{q}_l, \nu_{q_l})$, calculate the belief $b(\alpha_l)$ of each α_l

$$b(\alpha_l) \propto p(\alpha_l) n_{\alpha_l \to f_{\alpha_l}}(\alpha_l) \triangleq \mathcal{CN}\left(\alpha_l; \hat{\alpha}_l, \nu_{\alpha_l}\right)$$

where

$$\nu_{\alpha_l} = \left(\frac{1}{\nu_{r_l}} + \frac{1}{\nu_{q_l}}\right)^{-1}; \quad \hat{\alpha}_l = \nu_{\alpha_l}\left(\frac{\hat{r}_l}{\nu_{r_l}} + \frac{\hat{q}_l}{\nu_{q_l}}\right). \tag{E.9}$$

The mean and variance coincide those computed by [14, Eqs. (8a), (8b), (31) and (32)][3] in this Gaussian scenario.

S5: The variance ν_{ξ_i} and mean $\hat{\xi}_i$ of each message

$$m_{f_{\delta_i} \to h_i}(h_i) = \mathcal{CN}\left(h_i; \hat{\xi}_i, \nu_{\xi_i}\right) \tag{E.10}$$

is updated by using [14, Eqs. (5a) and (5b)]

$$\nu_{\xi_i} = \sum_l \nu_{\alpha_l}; \quad \hat{\xi}_i = \sum_l \Phi_{il} \hat{\alpha}_l - \hat{s}_i \nu_{\xi_i}. \tag{E.11}$$

It is noted that the computations of $\{\hat{r}_l\}$ and $\{\hat{\xi}_i\}$ in Steps S3 and S5 can be implemented using the inverse fast Fourier transform (IFFT) and fast Fourier transform (FFT) respectively, leading to lower complexity.

[2]Similar to F_{out}, we also denote g_{out}, defined in [14, Eq. (36)], by $g_{\text{out}}(\hat{\xi}_i, \nu_{\xi_i}, \hat{\theta}_i, \nu_{\theta_i})$.

[3]The belief $b(\alpha_l)$ in (E.9) is equivalent to the posterior function defined in [14, Eq. (33)].

3.2 Noise Precision Estimation

The message $m_{f_{\mathrm{P}_j}\to\lambda}(\lambda)$ from pilot observation node $f_{\mathrm{P}_j}(h_j,\lambda)$ to λ, $\forall j\in\mathcal{P}$ is calculated by the MF rule,

$$\begin{aligned} m_{f_{\mathrm{P}_j}\to\lambda}(\lambda) &= \exp\left\{\langle f_{\mathrm{P}_j}(h_j,\lambda)\rangle_{b(h_j)}\right\} \\ &\propto \lambda\exp\left\{-\lambda\langle|y_j-x_jh_j|^2\rangle_{b(h_j)}\right\}. \end{aligned}$$

Analogously, the message $m_{f_{\mathrm{D}_i}\to\lambda}(\lambda)$ from data observation node f_{D_i} to λ, $\forall i\in\mathcal{D}$, can be represented as

$$\begin{aligned} m_{f_{\mathrm{D}_i}\to\lambda}(\lambda) &= \exp\left\{\langle f_{\mathrm{D}_i}(h_i,\lambda)\rangle_{b(h_i)b(x_i)}\right\} \\ &\propto \lambda\exp\left\{-\lambda\langle|y_i-x_ih_i|^2\rangle_{b(h_i)b(x_i)}\right\}. \end{aligned}$$

Supposing that the priori pdf $p(\lambda)$ of λ is set to be $1/\lambda$, the belief $b(\lambda)$ of noise precision λ is updated as,

$$b(\lambda)\propto p(\lambda)\prod_{j\in\mathcal{P}} m_{f_{\mathrm{P}_j}\to\lambda}(\lambda)\prod_{i\in\mathcal{D}} m_{f_{\mathrm{D}_i}\to\lambda}(\lambda) \tag{E.12}$$

and its mean value is given by

$$\hat{\lambda} = \frac{P+N}{\sum\limits_{j\in\mathcal{P}}\langle|y_j-x_jh_j|^2\rangle_{b(h_j)}+\sum\limits_{i\in\mathcal{D}}\langle|y_i-x_ih_i|^2\rangle_{b(h_i)b(x_i)}}. \tag{E.13}$$

3.3 Soft Demodulation and Decoding

The message $m_{f_{\mathrm{D}_i}\to x_i}(x_i)$ from data observation node f_{D_i} to variable node x_i, $\forall i\in\mathcal{D}$, is computed by using the MF rule,

$$\begin{aligned} m_{f_{\mathrm{D}_i}\to x_i}(x_i) &= \exp\left\{\left\langle\log f_{\mathrm{D}_i}(h_i,x_i,\lambda)\right\rangle_{b(h_i)b(\lambda)}\right\} \\ &\propto \mathcal{CN}\left(x_i;\frac{y_i\hat{h}_i}{\nu_{h_i}+|\hat{h}_i|^2},\frac{1}{\hat{\lambda}(\nu_{h_i}+|\hat{h}_i|^2)}\right). \end{aligned} \tag{E.14}$$

Massages $\{n_{x_i\to f_{\mathrm{M}}}(x_i)=m_{f_{\mathrm{D}_i}\to x_i}(x_i),\ \text{for all } i\in\mathcal{D}\}$ are passed to soft demodulation and decoding models, where demodulation is performed by the standard BP message update rule and decoding is implemented with the forward-backward (BCJR) algorithm [1]. Then the discrete extrinsic messages

$$m_{f_{\mathrm{M}}\to x_i}(x_i)=\sum_{s\in\mathcal{S}}\beta_i(s)\delta(x_i-s) \tag{E.15}$$

are passed back, where $\mathcal{S}$ stands for modulation constellation, and $\beta_i(s)$ represent extrinsic information on symbol x_i. At last, the belief $b(x_i)$ of data symbol x_i, $\forall i \in \mathcal{D}$ is updated by

$$b(x_i) \propto n_{x_i \to f_{\mathrm{M}}}(x_i) m_{f_{\mathrm{M}} \to x_i}(x_i). \tag{E.16}$$

3.4 Message Passing Schedule

From the factor graph in Fig. 1 we can find that there are multitude of message passing schedules. In this paper, we firstly perform channel state and noise precision estimation with only pilots, and the number of iterations is denoted by T_p. Secondly, the joint channel state and noise precision estimation and decoding is carried out iteratively using both the pilots and data, and the number of iterations is denoted by T_d. The aforementioned schedule and the corresponding message updating are summarized in **Algorithm 1**, where lines 2-10 correspond to channel and noise precision estimation with only pilots and lines 12-22 correspond to joint channel and noise precision estimation and decoding with both pilots and data. Note that, the message computations in lines 6 and 9 are special forms of (E.8) and (E.13), since only pilots are used in lines 6 and 9.

Table E.1: Parameters setting of the OFDM system

Subcarrier spacing	15KHz
Subcarrier number	512
Number of evenly spacing pilot symbols	32
Modulation for pilot symbols	QPSK
Modulation for data symbols	16QAM
Channel interleaver	Random
Number of channel taps	32

4 Simulation Results

We consider an OFDM system with parameters given in Table E.1, and compare our proposed algorithm and the state-of-the-art algorithms in literatures in terms of BER performance. We use "BP-MF-GAMP" to denote our algorithm, and use "BP-MF-4", "BP-MF-32" and "BP-MF-512" to denote the algorithm in [11] with group size (the state-space dimension of the Markov model) of 4, 32 and 512, respectively. Note that, when the group size G is selected to be 512, it is equivalent to the algorithm proposed in [10]. We also provide a comparison with the receiver [8] denoted by "EP-GAMP", where the noise precision is assumed to be known. As a reference, the performance

Algorithm 6 The Proposed OFDM Receiver

Initialize $\hat{\lambda}$, $\hat{\alpha}_l$, ν_{α_l}, $\hat{\xi}_j$, ν_{ξ_j}, $\forall j \in \mathcal{P}$, $\forall l \in [1:L]$.
for $t = 1 \rightarrow T_p$ **do**
  $\forall j \in \mathcal{P}$: update $\hat{\theta}_j$ and ν_{θ_j} by (E.4).
  $\forall j \in \mathcal{P}$: update ν_{h_j} and $\hat{h}_j$ by (E.5).
  $\forall j \in \mathcal{P}$: update $\hat{s}_j$ and τ_{s_j} by (E.6) and (E.7).
  $\forall l \in [1:L]$: $\nu_{r_l} \leftarrow (\sum_{j \in \mathcal{P}} \tau_{s_j})^{-1}$,
    $\hat{r}_l \leftarrow \nu_{r_l} \sum_{j \in \mathcal{P}} \hat{s}_j \Phi^*_{jl} + \hat{\alpha}_l$.
  $\forall l \in [1:L]$: update ν_{α_l} and $\hat{\alpha}_l$ by (E.9).
  $\forall j \in \mathcal{P}$: update ν_{ξ_j} and $\hat{\xi}_j$ by (E.11).
  $\hat{\lambda} \leftarrow P / \{\sum_{j \in \mathcal{P}} \langle |y_j - x_j h_j|^2 \rangle_{b(h_j)}\}$.
end for t
Initialize $\hat{s}_i$, $\hat{\theta}_i$, ν_{θ_i}, $\forall i \in \mathcal{D}$.
for $t = 1 \rightarrow T_d$ **do**
  $\forall i \in \mathcal{D} \cup \mathcal{P}$: update ν_{ξ_i} and $\hat{\xi}_i$ by (E.11).
  $\forall i \in \mathcal{D} \cup \mathcal{P}$: update ν_{h_i} and $\hat{h}_i$ by (E.5).
  $\forall i \in \mathcal{D}$: update $m_{f_{\mathrm{D}_i} \rightarrow x_i}(x_i)$ by (E.14), send to soft demodulation and decoding part, and then yield $m_{f_{\mathrm{M}} \rightarrow x_i}(x_i)$ using standard BP.
  $\forall i \in \mathcal{D}$: update $b(x_i)$ by (E.16).
  $\forall i \in \mathcal{D}$: update $\hat{\theta}_i$ and ν_{θ_i} by (E.3),
  $\forall j \in \mathcal{P}$: update $\hat{\theta}_j$ and ν_{θ_j} by (E.4).
  $\forall i \in \mathcal{D} \cup \mathcal{P}$: update $\hat{s}_i$ and τ_{s_i} by (E.6) and (E.7).
  $\forall l \in [1:L]$: update ν_{r_l} and $\hat{r}_l$ by (E.8).
  $\forall l \in [1:L]$: update ν_{α_l} and $\hat{\alpha}_l$ by (E.9).
  update $\hat{\lambda}$ by (E.13).
end for t

of the receiver with perfect channel state information $\boldsymbol{h}$ and noise precision λ is also included, denoted by "Perfect CSI". The receivers, except "Perfect CSI", first carry out $T_p = 5$ iterations for channel (and noise precision) estimation with only pilots. Then joint channel (and noise precision) estimation and decoding are performed with $T_d = 15$ iterations.

In Fig. E.2, the BER performance of the receivers versus different SNRs is shown. It can be seen that "BP-MF-GAMP" and "BP-MF-512" perform similar to "EP-GAMP" with known λ. But the performance of "BP-MF-G" (denoting the algorithm in [11] with group size G) deteriorates with the decrease of group size G, and the performance degrades severely when $G = 4$. Note that, the complexity of "BP-MF-GAMP" is approximately the same as "BP-MF-4".

Fig. E.3 shows the performance of the receivers operating at an SNR of 10dB versus the iteration index. We can see that the proposed "BP-MF-GAMP" receiver converges faster than "BP-MF-G" receivers, and even faster than "EP-GAMP" with known λ. It is also observed that, the convergence of "BP-MF-G" also becomes slower with the decrease of the group size G.

4.1 Computational Complexity Comparison

The complexity of the proposed algorithm and those in [10] and [11] is dominated by the channel estimation part, so we only analyze the complexity of channel estimation. In [10], an inverse operation of a large matrix with dimension $(N+P) \times (N+P)$ is required in each iteration, so it has cubic complexity $\mathcal{O}((N+P)^3)$. By assuming that the channel weight obey a Markov model, the large matrix inverse is converted into a number of small matrix inverses (with size G) in [11], and the complexity of the algorithm in [11] is reduced to $\mathcal{O}(G^2(N+P))$.

Designed based on the factor graph in Fig. E.1 where all variables are in scalar form, the proposed receiver avoids matrix inverses and its complexity is $\mathcal{O}((N+P)L)$. Moreover, the computational complexity can be reduced to $\mathcal{O}((N+P)\log(N+P))$ by using the IFFT and FFT for Steps S3 and S5.

5 Conclusion

By incorporating the GAMP algorithm into a unified BP-MF framework, we have designed a low complexity message passing receiver to perform joint channel state and noise precision estimation and decoding. The MF rule is used to tackle the observation factor nodes and GAMP is used to handle the message computations for the densely connected part of the factor graph. It has been shown that, the proposed algorithm outperforms the state-of-the-art algorithms in terms of computational complexity or performance.

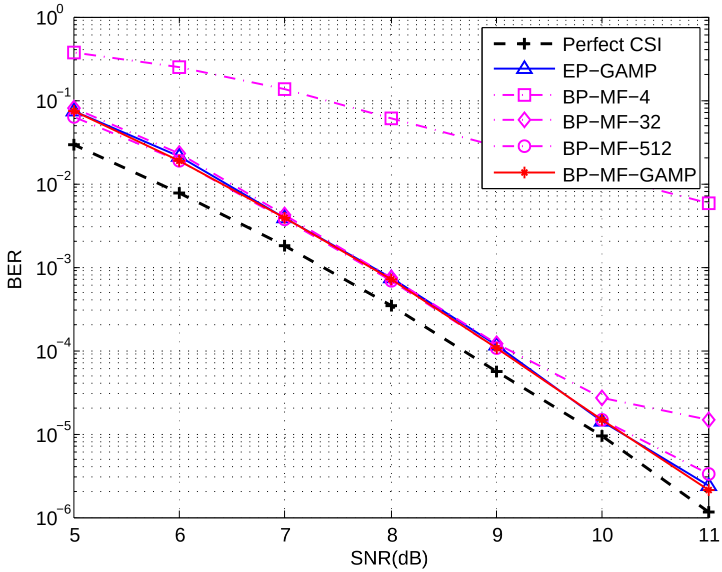


Fig. E.2: BER performance of the receivers versus SNR.

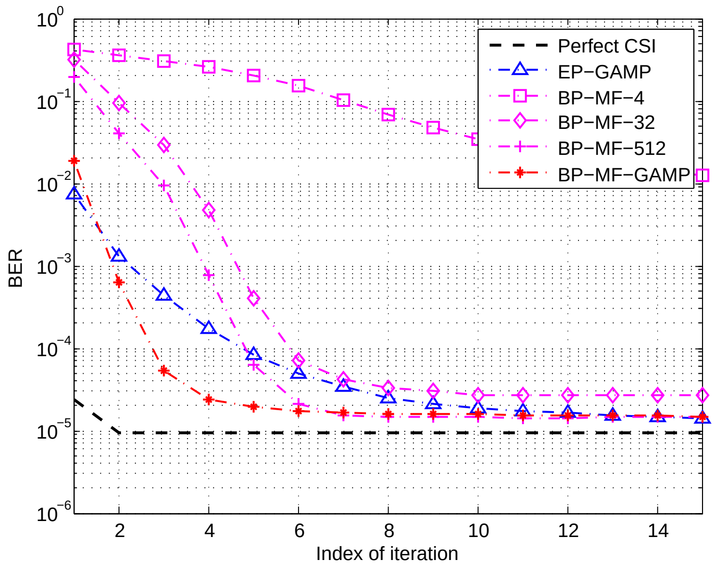


Fig. E.3: BER performance of the receivers versus iteration index for an SNR of 10 dB.

References

[1] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[2] 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, 2005.

[3] Q. Guo, L. Ping, and D. Huang, "A low-complexity iterative channel estimation and detection technique for doubly selective channels," *IEEE Trans. Commun.*, vol. 8, no. 8, pp. 4340–4349, Aug. 2009.

[4] T. Wo, P. 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.

[5] E. Riegler, G. E. Kirkelund, C. Navarro Manchón, M.-A. Badiu, and B. H. Fleury, "Merging belief propagation and the mean field approximation: A free energy approach," *IEEE Trans. Inform. Theory*, vol. 59, no. 1, pp. 588–602, Jan. 2013.

[6] J. Winn and C. Bishop, "Variational message passing," *Journal of Machine Learning Research*, vol. 6, pp. 661–694, 2005.

[7] T. P. Minka, "Expectation propagation for approximate Bayesian inference," in *Proceedings of the 17th Conf. on Uncertainty in Artificial Intelligence*, 2001, pp. 362–369.

[8] S. Wu, L. Kuang, Z. Ni, J. Lu, D. Huang, and Q. Guo, "Expectation propagation approach to joint channel estimation and decoding for OFDM systems," in *IEEE Int. Conference on Acustic, Speech and Signal Processing (ICASSP 2014)*, 4-9 May 2014, pp. 1941–1945.

[9] P. Sun, C. Zhang, Z. Wang, C. Navarro Manchón, and B. H. Fleury, "Iterative receiver design for ISI channels using combined belief- and expectation-propagation," *IEEE Signal Processing Lett.*, vol. 22, no. 10, pp. 1733–1737, Oct. 2015.

[10] M.-A. Badiu, G. E. Kirkelund, C. Navarro Manchón, E. Riegler, and B. H. Fleury, "Message-passing algorithms for channel estimation and decoding using approximate inference," in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT 2012)*, July 2012, pp. 2376–2380.

[11] M.-A. Badiu, C. Navarro Manchón, and B. H. Fleury, "Message-passing receiver architecture with reduced-complexity channel estimation," *IEEE Commun. Lett.*, vol. 17, no. 7, pp. 1404–1407, July 2013.

[12] C. Navarro Manchón, G. E. Kirkelund, E. Riegler, L. P. B. Christensen, and B. H. Fleury, "Receiver architectures for MIMO-OFDM based on a combined VMP-SP algorithm," *2011, arXiv:1111.5848 [stat.ML]*.

[13] C. Zhang, C. Navarro Manchón, Z. Wang, and B. H. Fleury, "Message-passing receivers for single carrier systems with frequency-domain equalization," *IEEE Signal Processing Lett.*, vol. 22, no. 4, pp. 404–407, Apr. 2015.

[14] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," *arXiv:1010.5141 [cs.IT]*, 2010. [Online]. Available: http://arxiv.org/abs/1010.5141

[15] P. Schniter, "A message-passing receiver for BICM-OFDM over unknown clustered-sparse channels," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 8, pp. 1462–1474, Dec. 2011.

[16] 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:1602.07762 [cs.IT]*, 2016. [Online]. Available: http://arxiv.org/abs/1602.07762

Paper F

A BP-MF-EP Based Iterative Receiver for Joint Phase Noise Estimation, Equalization and Decoding

Wei Wang, Zhongyong Wang, Chuanzong Zhang, Qinghua Guo, Peng Sun, and Xingye Wang

The paper has been resubmitted to the
IEEE Signal Processing Letters, 2016.

The layout has been revised.

Abstract

In this work, with combined belief propagation (BP), mean field (MF) and expectation propagation (EP), an iterative receiver is designed for joint phase noise (PN) estimation, equalization and decoding in a coded communication system. The presence of the PN results in a nonlinear observation model. Conventionally, the nonlinear model is directly linearized by using the first-order Taylor approximation, e.g., in the state-of-the-art soft-input extended Kalman smoothing approach (Soft-in EKS). In this work, MF is used to handle the factor due to the nonlinear model, and a second-order Taylor approximation is used to achieve Gaussian approximation to the MF messages, which is crucial to the low-complexity implementation of the receiver with BP and EP. It turns out that our approximation is more effective than the direct linearization in the Soft-in EKS with similar complexity, leading to significant performance improvement as demonstrated by simulation results.

1 Introduction

Local oscillators, which provide a reference signal for time and frequency synchronization, are one of the key modules in a communication system. The instability of oscillators results in phase noise (PN), which may severely affect the system performance [1].

Various Bayesian and non-Bayesian approaches have been proposed to solve the PN problem. Bhatti et al. modelled the PN with a discrete cosine transform (DCT) expansion [2], where the DCT coefficients can be easily estimated. However, the DCT method is a non-Bayesian one, and it does not make use of the time dependence of the PN process. In Bayesian methods such as particle filter [3], Tikhonov parametric estimation [4], and extended Kalman smoothing (EKS) [5], PN is modelled as a Wiener process. The particle filtering method [3] needs to sample the posteriori probability density function (PDF) of continuous-valued PN variables, where a larger number of particles yields better performance at the cost of higher complexity. The Tikhonov parametrization method [4] (or called a von Mises distribution [6]) is an iterative method to deal with the presence of strong PN for AWGN channels. The intractable integral operation associated with continuous variables is circumvented by constraining the PDF to Tikhonov distribution. However, the work in [4] was focused on AWGN channel, and a straightforward extension to the inter-symbol interference (ISI) channel which is allowed by incorporating a MAP equalizer will lead to complexity growing exponentially with the channel memory length. In the soft input EKS (Soft-in EKS) method[1] proposed in [5], the nonlinear observation model is directly linearized by

[1]The EKS method in [5] was proposed for AWGN channels. It can be extended to the case of ISI channels, e.g., by incorporating the BP-EP algorithm [7] to handle ISI channels.

using the first order Taylor expansion. Soft-in EKS has been used in single-input single-output (SISO) and multiple-input multiple-output (MIMO) systems [5, 8–10].

Recently, the message passing techniques, such as belief propagation (BP) [11] and variational message passing (VMP) [12], have been widely used for iterative receivers design. BP is effective for discrete probability models and linear Gaussian models. A BP-based equalizer proposed in [13] has a linear complexity, which is much lower than that of the equalizer in [14]. The VMP method, also referred as mean filed (MF), is especially suitable for handling variables with exponential distributions. Recently, a unified message passing framework was proposed in [15], where BP and MF are merged to keep the virtues of BP and MF while avoid their drawbacks. It has been applied to joint channel estimation and decoding in orthogonal frequency division multiplexing (OFDM) system [16, 17] and single carrier frequency domain equalization (SC-FDE) system [18]. In addition, expectation propagation (EP) [19] has been used to achieve Gaussian approximation to non-Gaussian messages, and combined EP and BP has been applied to flat-fading or ISI channel equalization, e. g., in [7, 20].

In this paper, with combined BP, MF and EP, we propose an iterative approach to joint PN estimation, equalization and decoding for a coded system over ISI channels. BP and EP are used to deal with the linear model for PN process and modulation and coding, while MF is used to handle the factor due to the nonlinear observation model. Furthermore, the non-Gaussian MF messages are approximated to be Gaussian by using the second-order Taylor expansion, which enables low-complexity implementation of the receiver with BP and EP. Our approximation is more effective than the direct linearization of the nonlinear model in the soft-in EKS [5], which is demonstrated by the significant performance gain of the proposed approach in terms of mean-square-error (MSE) of PN estimation and system bit-error-rate (BER) performance.

Notation-The superscriptions $(\cdot)^{\mathrm{T}}$and $(\cdot)^{\mathrm{H}}$denote the transpose and conjugate transpose, respectively. We use $\propto$ to denote equality of functions up to a scale factor, and use $\boldsymbol{I}_N$ to denote an $N \times N$ identity matrix. The real part of a complex quantity is denoted by $\Re[\cdot]$. The functions $\mathcal{N}(x; \hat{x}, \sigma_x^2)$ and $\mathcal{CN}(x; \hat{x}, \sigma_x^2)$ stand for real and proper complex Gaussian probability distributions with mean $\hat{x}$ and variance σ_x^2, respectively.

2 System Model and Factor Graph Representation

We consider a coded communication system. An information bit sequence $\boldsymbol{b} = [b_0, ..., b_{N_{b-1}}]^{\mathrm{T}}$ is encoded and interleaved, yielding an interleaved codeword $\boldsymbol{c} = [c_0, ..., c_{N_{c-1}}]^{\mathrm{T}}$. Then sequence $\boldsymbol{c}$ is mapped to a symbol sequence

$\boldsymbol{x} = [x_0, ..., x_{M-1}]^{\mathrm{T}}$ which is transmitted over an ISI channel with coefficients $\boldsymbol{h} = [h_{L-1}, ..., h_0]^{\mathrm{T}}$. The channel coefficients are assumed to be constant during each transmitted block and they are available to the receiver. By considering the effect of PN, the received baseband signal at time instant k, $(k = 0, 1, ..., M + L - 2)$, can be represented as [21]

$$y_k = e^{j\theta_k} \sum_{l=0}^{L-1} h_l x_{k-l} + n_k = e^{j\theta_k} \boldsymbol{h}^{\mathrm{T}} \boldsymbol{s}_k + n_k \tag{F.1}$$

where $\boldsymbol{s}_k \triangleq [x_{k-L+1}, ..., x_k]^{\mathrm{T}}$ with $x_k = 0$ for $k < 0$ and $k > M - 1$, and n_k is a sample of the complex Gaussian noise with variance σ_n^2. The phase θ_k represents the PN at time instant k, and the PN can be modelled as a random-walk (Wiener) process [4], [5]

$$\theta_k = \theta_{k-1} + \Delta\theta_k \tag{F.2}$$

where $\Delta\theta_k$ is a white real Gaussian process with distribution $\mathcal{N}(\Delta\theta_k; 0, \sigma_\Delta^2)$, and θ_0 is assumed to have a uniform distribution over $[0, 2\pi)$. We define $\boldsymbol{\theta} = [\theta_0, \theta_1, ..., \theta_{M+L-2}]^{\mathrm{T}}$.

The joint probability of $\boldsymbol{b}, \boldsymbol{c}, \boldsymbol{x}, \boldsymbol{s}$ and $\boldsymbol{\theta}$ with given observation $\boldsymbol{y} = [y_0, y_1, \ldots, y_{M+L-2}]^{\mathrm{T}}$ can be expressed as

$$\begin{aligned} p(\boldsymbol{b}, \boldsymbol{c}, \boldsymbol{x}, \boldsymbol{s}, \boldsymbol{\theta} | \boldsymbol{y}) \propto & \prod_{k=0}^{M+L-2} f_{y_k}(\boldsymbol{s}_k, \theta_k) f_{\boldsymbol{s}_k}(\boldsymbol{s}_k, \boldsymbol{s}_{k-1}, x_k) \\ & f_{\theta_0} \prod_{k=1}^{M+L-2} f_{\theta_k}(\theta_k, \theta_{k-1}) f_{\mathrm{X}}(\boldsymbol{x}, \boldsymbol{c}, \boldsymbol{b}) \end{aligned} \tag{F.3}$$

where $f_{y_k}(\boldsymbol{s}_k, \theta_k) \triangleq p(y_k | \boldsymbol{s}_k, \theta_k) \propto \mathcal{CN}(y_k; e^{j\theta_k} \boldsymbol{h}^{\mathrm{T}} \boldsymbol{s}_k, \sigma_n^2)$ denotes the likelihood function of $\boldsymbol{s}_k$ and $\boldsymbol{\theta}_k$, $f_{\theta_k}(\theta_k, \theta_{k-1}) \triangleq p(\theta_k | \theta_{k-1}) = \mathcal{N}(\theta_k; \theta_{k-1}, \sigma_\Delta^2)$ is the conditional PDF of θ_k given θ_{k-1}, and $f_{\mathrm{X}}(\boldsymbol{x}, \boldsymbol{c}, \boldsymbol{b})$ denotes the mapping, interleaving and coding constraints. Function $f_{\boldsymbol{s}_k}(\boldsymbol{s}_k, \boldsymbol{s}_{k-1}, x_k)$ represents the deterministic relationship between $\boldsymbol{s}_k$, $\boldsymbol{s}_{k-1}$ and x_k which is given by $\boldsymbol{s}_k = \boldsymbol{G}\boldsymbol{s}_{k-1} + \boldsymbol{e}x_k$, where the $L \times L$ matrix $\boldsymbol{G} = [\boldsymbol{0}\ \boldsymbol{I}_{L-1};\ 0\ \boldsymbol{0}^{\mathrm{T}}]$, the length-$L$ vector $\boldsymbol{e} = [\boldsymbol{0}^{\mathrm{T}}\ 1]^{\mathrm{T}}$, and $\boldsymbol{0}$ is a zero column vector with length $L - 1$.

A factor graph representation of (F.3) is shown in Fig. F.1, which will be employed to develop a combined BP-MF-EP based receiver to achieve joint PN estimation, equalization and decoding in next section.

3 Iterative Receiver Design with BP-MF-EP

Due to the presence of PN, the observation model in (F.1) is nonlinear. In EKS, the nonlinear model is directly linearized with the first order Taylor

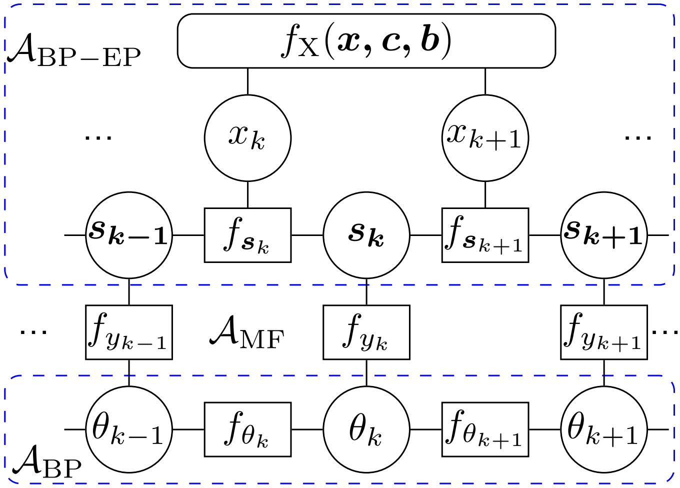


Fig. F.1: Factor-graph representation of the probabilistic model (F.3)

approximation. The nonlinear model is represented by the factors $\{f_{y_k}, \forall k\}$ in Fig. F.1. In this work, we use MF to handle the factors.

As shown in Fig. F.1, we partition the graph into three parts: BP-EP subgraph, MF subgraph and BP subgraph. Accordingly, the factor nodes are classified into three disjoint sets: $\mathcal{A}_{\text{BP-EP}} \triangleq \{f_{s_k}, f_{\mathrm{X}}; \forall k\}$, $\mathcal{A}_{\mathrm{MF}} \triangleq \{f_{y_k}; \forall k\}$ and $\mathcal{A}_{\mathrm{BP}} \triangleq \{f_{\theta_k}; \forall k\}$ with $\mathcal{A}_{\text{BP-EP}} \bigcap \mathcal{A}_{\mathrm{MF}} \bigcap \mathcal{A}_{\mathrm{BP}} = \emptyset$. In the following, we detail the messages updating in each subgraph.

3.1 Message Passing in BP Subgraph

As shown in Fig. F.1, message passing for PN estimation operates in the BP subgraph, where we need to calculate the forward and backward messages and the outgoing messages which are input to the MF subgraph.

We assume that the incoming messages from the MF subgraph are available, and they are Gaussian, i.e., we have $\{m_{f_{y_k}\to\theta_k}(\theta_k) \propto \mathcal{N}(\theta_k; \hat{\theta}_k^{\downarrow}, \sigma^2_{\theta_k^{\downarrow}}), \forall k\}$. The details on the calculations of the incoming messages are delayed to Section 3.3. It is worth mentioning that, with the incoming Gaussian messages, all the messages running in the subgraph are Gaussian.

With the Gaussian message $m_{f_{\theta_{k-1}}\to\theta_{k-1}}(\theta_{k-1}) \propto \mathcal{N}(\theta_{k-1}; \hat{\theta}_{k-1}^{\rightarrow}, \sigma^2_{\theta_{k-1}^{\rightarrow}})$, the message from variable θ_{k-1} to factor f_{θ_k} is calculated as $n_{\theta_{k-1}\to f_{\theta_k}}(\theta_{k-1}) =$

$m_{f_{\theta_{k-1}}\to\theta_{k-1}}(\theta_{k-1})m_{f_{y_{k-1}}\to\theta_{k-1}}(\theta_{k-1})$. The forward message $m_{f_{\theta_k}\to\theta_k}(\theta_k)$ reads

$$\begin{aligned} m_{f_{\theta_k}\to\theta_k}(\theta_k) &\propto \int f_{\theta_k}(\theta_k,\theta_{k-1})n_{\theta_{k-1}\to f_{\theta_k}}(\theta_{k-1})d\theta_{k-1} \\ &\propto \mathcal{N}(\theta_k;\hat{\theta}_k^{\rightarrow},\sigma^2_{\theta_k^{\rightarrow}}), \end{aligned} \tag{F.4}$$

We assume that the initial phase noise θ_0 is absorbed into the channel in the acquisition of the channel state information [8], so the initial message for the forward recursive process $\hat{\theta}_0 = 0, \sigma^2_{\theta_0} = 0$.

Same to the forward messages, the backward message $m_{f_{\theta_{k+1}}\to\theta_k}(\theta_k) \propto \mathcal{N}(\theta_k;\hat{\theta}_k^{\leftarrow},\sigma^2_{\theta_k^{\leftarrow}})$.

According to [15], the outgoing messages input to the MF subgraph should be the belief of θ_k, which can be calculated as

$$\begin{aligned} b(\theta_k) &= m_{f_{y_k}\to\theta_k}(\theta_k)m_{f_{\theta_k}\to\theta_k}(\theta_k)m_{f_{\theta_{k+1}}\to\theta_k}(\theta_k) \\ &\propto \mathcal{N}(\theta_k;\hat{\theta}_k,\sigma^2_{\theta_k}), \end{aligned} \tag{F.5}$$

where

$$\sigma^{-2}_{\theta_k} = \sigma^{-2}_{\theta_k^{\downarrow}} + \sigma^{-2}_{\theta_k^{\leftarrow}} + \sigma^{-2}_{\theta_k^{\rightarrow}} \tag{F.6}$$

$$\hat{\theta}_k = \sigma^2_{\theta_k}(\sigma^{-2}_{\theta_k^{\downarrow}}\hat{\theta}_k^{\downarrow} + \sigma^{-2}_{\theta_k^{\leftarrow}}\hat{\theta}_k^{\leftarrow} + \sigma^{-2}_{\theta_k^{\rightarrow}}\hat{\theta}_k^{\rightarrow}). \tag{F.7}$$

3.2 Message Passing in BP-EP Subgraph

We assume that the incoming messages from the MF subgraph are available, and they are Gaussian. The calculations of the incoming messages will be detailed in Section 3.3. So this subgraph involves the incoming Gaussian messages from the MF subgraph and discrete binary messages from the decoder. For this subgraph, we simply borrow the BP-EP algorithm developed in [7] where the use of EP produces Gaussian messages for x_k, which will in turn lead to Gaussian output messages in the BP-EP subgraph. We refer readers to [7] for the details of the BP-EP algorithm.

With the BP-EP algorithm, we can calculate the messages $m_{f_{s_k}\to s_k}(\boldsymbol{s}_k) \propto \mathcal{CN}(s_k;\hat{\mathbf{s}}_k^{\rightarrow},\Sigma_{s_k^{\rightarrow}})$ and $m_{f_{s_{k+1}}\to s_k}(\boldsymbol{s}_k) \propto (s_k;\hat{\mathbf{s}}_k^{\leftarrow},\Sigma_{s_k^{\leftarrow}})$, which are all Gaussian.

According to [15], the outgoing messages are the belief of $\boldsymbol{s}_k$ denoted by $b(\boldsymbol{s}_k)$, which are Gaussian again and can be expressed as

$$\begin{aligned} n_{s_k\to f_{y_k}}(\boldsymbol{s}_k) &\propto m_{f_{y_k}\to s_k}(\boldsymbol{s}_k)m_{f_{s_k}\to s_k}(\boldsymbol{s}_k)m_{f_{s_{k+1}}\to s_k}(\boldsymbol{s}_k) \\ &\propto \exp\left\{-(\boldsymbol{s}_k-\hat{\mathbf{s}}_k)^{\mathrm{H}}\Sigma^{-1}_{\hat{\mathbf{s}}_k}(\boldsymbol{s}_k-\hat{\mathbf{s}}_k)\right\} \end{aligned} \tag{F.8}$$

where $m_{f_{y_k} \to s_k}(s_k)$ is obtained by (F.15) and

$$\Sigma_{s_k}^{-1} = \Sigma_{s_k^{\rightarrow}}^{-1} + \Sigma_{s_k^{\uparrow}}^{-1} + \Sigma_{s_k^{\leftarrow}}^{-1} \tag{F.9}$$

$$\Sigma_{s_k}^{-1} \hat{s}_k = \Sigma_{s_k^{\rightarrow}}^{-1} \hat{s}_k^{\rightarrow} + \Sigma_{s_k^{\uparrow}}^{-1} \hat{s}_k^{\uparrow} + \Sigma_{s_k^{\leftarrow}}^{-1} \hat{s}_k^{\leftarrow}. \tag{F.10}$$

3.3 Message Passing in the MF Subgraph

As shown by the middle part of the graph in Fig. F.1, the MF subgraph consists of the observation factors f_{y_k}. We need to compute the outgoing messages to the BP-EP subgraph (BP subgraph) based on the incoming messages from the BP subgraph (BP-EP subgraph).

Assume that the incoming message $b(s_k)$ from the BP-EP subgraph is available. According to the rules [15] the outgoing messages to the BP subgraph can be computed as

$$\begin{aligned} m_{f_{y_k} \to \theta_k}(\theta_k) &\propto \exp\left\{ \int \log(f_{y_k}(\theta_k, s_k)) b(s_k) ds_k \right\} \\ &\propto \exp\left\{ \Re[r_k e^{j\theta_k}] \right\} \end{aligned} \tag{F.11}$$

where $r_k \triangleq 2\sigma_n^{-2} y_k^* \boldsymbol{h}^{\mathrm{T}} \hat{s}_k$, and $\hat{s}_k$ is the mean parameter vector of the Gaussian belief $b(s_k)$. Note that the message $m_{f_{y_k} \to \theta_k}(\theta_k)$ yielded in (F.11) is no longer Gaussian. However, Gaussian messages are expected for the BP subgraph for PN estimation, which is crucial to its low complexity implementation. To achieve this, we use the second-order Taylor expansion of $\Re[r_k e^{j\theta_k}]$ at the estimate of θ_k, i.e.,

$$\begin{aligned} & m_{f_{y_k} \to \theta_k}(\theta_k) \\ & \approx \exp\left\{ -\frac{1}{2} \Re[r_k e^{j\hat{\theta}_k}] \theta_k^2 + \Re[r_k e^{j\hat{\theta}_k}(j + \hat{\theta}_k)] \theta_k \right\} \\ & \propto \mathcal{N}(\theta_k; \hat{\theta}_k^{\downarrow}, \sigma_{\theta_k^{\downarrow}}^2) \end{aligned} \tag{F.12}$$

where $\hat{\theta}_k$ denotes the mean of θ_k computed in (F.7), and

$$\sigma_{\theta_k^{\downarrow}}^{-2} = \Re[r_k e^{j\hat{\theta}_k}] \tag{F.13}$$

$$\sigma_{\theta_k^{\downarrow}}^{-2} \hat{\theta}_k^{\downarrow} = \Re[r_k e^{j\hat{\theta}_k}(j + \hat{\theta}_k)]. \tag{F.14}$$

It is noted that the Soft-in-EKS algorithm [5] uses the first-order Taylor expansion to locally linearize model (F.1) directly. In contrast, we use the second order Taylor series to approximate the MF message $m_{f_{y_k} \to \theta_k}(\theta_k)$. It turns out that the performance of our algorithm is much better than that of the Soft-in-EKS algorithm, as demonstrated by simulation results.

Similarly, we also apply MF message update rules to the computation of the outgoing message $m_{f_{y_k}\to s_k}(s_k)$ for the BP-EP subgraph

$$\begin{aligned} m_{f_{y_k}\to s_k}(s_k) &= \exp\left\{\int \log(f_{y_k}(\theta_k, s_k))b(\theta_k)d\theta_k\right\} \\ &\propto \exp\left\{-s_k^{\mathrm{H}}\Sigma_{s_k^{\uparrow}}^{-1}s_k + 2\Re[s_k^{\mathrm{H}}\Sigma_{s_k^{\uparrow}}^{-1}\hat{s}_k^{\uparrow}]\right\} \end{aligned} \tag{F.15}$$

where

$$\Sigma_{\hat{s}_k^{\uparrow}}^{-1} = \sigma_n^{-2}\boldsymbol{h}\boldsymbol{h}^{\mathrm{T}} \tag{F.16}$$

$$\Sigma_{s_k^{\uparrow}}^{-1}\hat{s}_k^{\uparrow} = \sigma_n^{-2}\Re[\, y_k\langle e^{-j\theta_k}\rangle_{b(\theta_k)}\boldsymbol{h}]. \tag{F.17}$$

An approximation to the term $\langle e^{-j\theta_k}\rangle_{b(\theta_k)}$ in (F.17) can be obtained by exploiting the second-order Taylor expansion, and it can be calculated as $\langle e^{-j\theta_k}\rangle_{b(\theta_k)} \approx e^{-j\hat{\theta}_k}(1-0.5\sigma_{\theta_k}^2)$.

3.4 Message Passing Scheduling

The overall message passing schedule for joint PN estimation, equalization and decoding is summarized in **Algorithm 7**.

Algorithm 7 The combined BP-MF-EP Algorithm

1: **input** $\boldsymbol{y}, \boldsymbol{h}, \hat{\theta}_0, \sigma_{\theta_0}^2$
2: **initialize** $n_{\theta_0\to f_{\theta_1}}(\theta_0)$, $m_{f_{y_k}\to\theta_k}(\theta_k), \forall k$
3: **for** $i = 1 \to$Iteration **do**
4: **for** $k = 1 \to M+L-2$, compute $m_{f_{\theta_k}\to\theta_k}(\theta_k)$ using (F.4)
5: **for** $k = M+L-3 \to 1$, compute $m_{f_{\theta_{k+1}}\to\theta_k}(\theta_k)$
6: **for all** k: compute $n_{\theta_k\to f_{y_k}}(\theta_k)$ using (F.5)
7: **for all** k: compute $m_{f_{y_k}\to s_k}(s_k)$ using (F.15)
8: Run the BP-EP algorithm [7] in the BP-EP subgraph
9: **for all** k: update $n_{s_k\to f_{y_k}}(s_k)$ using (F.8)
10: **for all** k: update $m_{f_{y_k}\to\theta_k}(\theta_k)$ using (F.12)
11: **end for** i

3.5 Complexity Analysis

Note that the BP-EP algorithm in [7] is incorporated in both the Soft-in EKS method and the proposed BP-MF-EP method to handle ISI channels. Hence,

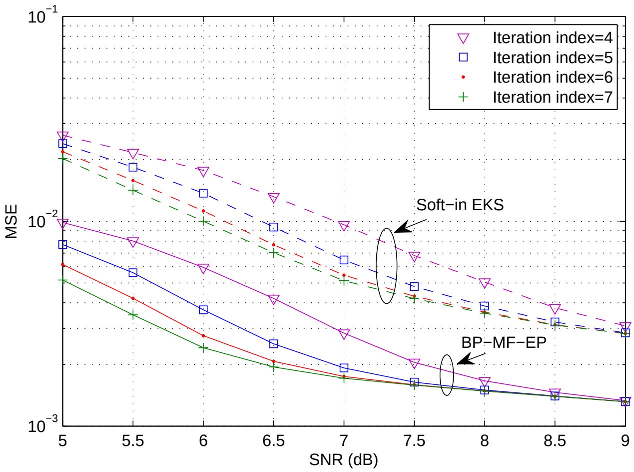


Fig. F.2: MSE performance of the phase noise estimation.

both methods involve the computation of (F.8), which requires a complexity of $O(L^3)$. We can also see that PN estimation in both the Soft-in EKS method and the BP-MF-EP method (i.e., the computation of (F.12) and the message passing in the BP subgraph shown in Fig.F.1) have similar complexity, which is in the order of L. From the above analysis, the BP-MF-EP method and the Soft-in EKS method have similar complexity.

4 Simulation Results

In this section, we evaluate the performance of the proposed method and compare it with the Soft-in EKS method (where the BP-EP algorithm in [7] is incorporated to handle ISI channels) in terms of MSE for PN estimation and BER for the system performance. The system settings are as follows. The length of symbols in each frame is 1024. A rate-1/2 nonsystematic convolutional code with generator $(23, 35)_8$ is used to encode the bits sequence, and the coded sequence is permuted with a pseudo random interleaver. QPSK with Gray mapping is used. In simulations, the phase noise is generated using a Wiener process (F.2) with innovation variance $\sigma_\Delta^2 = 1 \times 10^{-4}$ and the Proakis-C channel with coefficients $\boldsymbol{h} = [0.227, 0.460, 0.668, 0.460, 0.227]^{\mathrm{T}}$ is used to examine the performance of the receiver. As in [4], 5 pilot symbols are inserted every 256 symbols to make the iterative process bootstrap.

We compare the MSE performance of the proposed algorithm with that of the Soft-in EKS algorithm for PN estimation. The results with different number of iterations are shown in Fig. F.2. It can be seen that the proposed

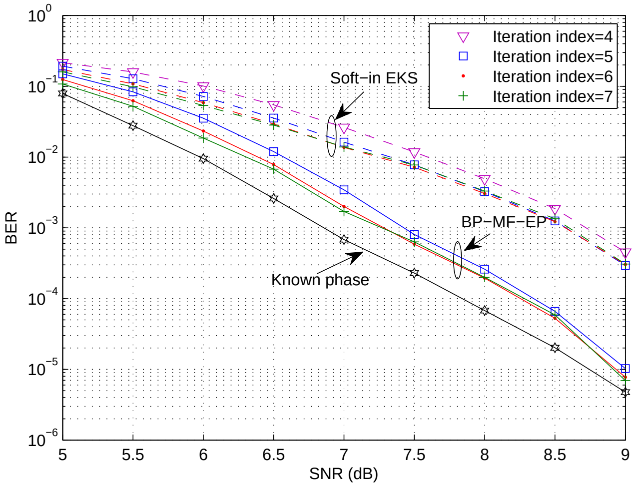


Fig. F.3: BER performance versus SNR.

BP-MF-EP method significantly outperforms the Soft-in EKS method.

The comparisons of system BER performance are shown in Fig. F.3, where the performance with known PN is also included for reference. It can be seen that considerable performance gains can be achieved by the proposed BP-MF-EP method compared to the Soft-in EKS method.

5 Conclusion

In this paper we have proposed an iterative receiver for joint PN estimation, equalization and decoding based on combined BP, MF and EP. In particular, MF is used to tackle the factors due to the nonlinear observation model and the second-order Taylor expansion is used to achieve Gaussian approximation to the MF messages, which is crucial to the low complexity implementation of the receiver. The approximation is more effective than the direct local linearization of the observation model in the Soft-in EKS. As shown by the simulation results, the proposed method significantly outperforms the Soft-in EKS with similar complexity.

References

[1] M. R. Khanzadi, D. Kuylenstierna, A. Panahi, T. Eriksson, and H. Zirath, "Calculation of the performance of communication systems from mea-

sured oscillator phase noise," *IEEE Trans. Circuits Syst. I*, vol. 61, no. 5, pp. 1553–1565, May 2014.

[2] J. Bhatti and M. Moeneclaey, "Iterative soft-decision-directed phase noise estimation from a DCT basis expansion," in *2009 IEEE 20th International Symposium on Personal Indoor and Mobile Radio Communications*, Sep. 2009, pp. 3228–3232.

[3] F. Z. Merli and G. M. Vitetta, "A factor graph approach to the iterative detection of OFDM signals in the presence of carrier frequency offset and phase noise," *IEEE Trans. Wireless Commun.*, vol. 7, no. 3, pp. 868–877, March 2008.

[4] G. Colavolpe, A. Barbieri, and G. Caire, "Algorithms for iterative decoding in the presence of strong phase noise," *IEEE J. Select. Areas Commun.*, vol. 23, no. 9, pp. 1748–1757, Sep. 2005.

[5] M. Nissila and S. Pasupathy, "Adaptive iterative detectors for phase-uncertain channels via variational bounding," *IEEE Trans. Commun.*, vol. 57, no. 3, pp. 716–725, March 2009.

[6] M. Senst and G. Ascheid, "A combined belief propagation and mean field algorithm for soft carrier phase estimation," *2011 8th International Symposium on Wireless Communication Systems, Aachen*, 2011.

[7] P. Sun, C. Zhang, Z. Wang, C. Navarro Manchón, and B. H. Fleury, "Iterative receiver design for ISI channels using combined belief- and expectation-propagation," *IEEE Signal Processing Lett.*, vol. 22, no. 10, pp. 1733–1737, Oct. 2015.

[8] H. Mehrpouyan, A. A. Nasir, S. D. Blostein, T. Eriksson, G. K. Karagiannidis, and T. Svensson, "Joint estimation of channel and oscillator phase noise in MIMO systems," *IEEE Trans. Signal Processing*, vol. 60, no. 9, pp. 4790–4807, Sep. 2012.

[9] A. A. Nasir, H. Mehrpouyan, R. Schober, and Y. Hua, "Phase noise in MIMO systems: Bayesian Cramér-Rao bounds and soft-input estimation," *IEEE Trans. Signal Processing*, vol. 61, no. 10, pp. 2675–2692, Oct. 2013.

[10] R. Krishnan, M. R. Khanzadi, L. Svensson, T. Eriksson, and T. Svensson, "Variational Bayesian framework for receiver design in the presence of phase noise in MIMO systems," in *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, Apr. 2012, pp. 347–352.

[11] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[12] J. Winn and C. Bishop, "Variational message passing," *Journal of Machine Learning Research*, vol. 6, pp. 661–694, 2005.

[13] G. Colavolpe, D. Fertonani, and A. Piemontese, "SISO detection over linear channels with linear complexity in the number of interferers," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 8, pp. 1475–1485, Dec. 2011.

[14] R. Koetter, A. C. Singer, and M. Tücher, "Turbo equalization," *IEEE Signal Processing Mag.*, vol. 21, pp. 67–80, Jan. 2004.

[15] E. Riegler, G. E. Kirkelund, C. Navarro Manchón, M.-A. Badiu, and B. H. Fleury, "Merging belief propagation and the mean field approximation: A free energy approach," *IEEE Trans. Inform. Theory*, vol. 59, no. 1, pp. 588–602, Jan. 2013.

[16] M.-A. Badiu, G. E. Kirkelund, C. Navarro Manchón, E. Riegler, and B. H. Fleury, "Message-passing algorithms for channel estimation and decoding using approximate inference," in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT 2012)*, July 2012, pp. 2376–2380.

[17] M.-A. Badiu, C. Navarro Manchón, and B. H. Fleury, "Message-passing receiver architecture with reduced-complexity channel estimation," *IEEE Commun. Lett.*, vol. 17, no. 7, pp. 1404–1407, July 2013.

[18] C. Zhang, C. Navarro Manchón, Z. Wang, and B. H. Fleury, "Message-passing receivers for single carrier systems with frequency-domain equalization," *IEEE Signal Processing Lett.*, vol. 22, no. 4, pp. 404–407, Apr. 2015.

[19] 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. Communication, Control, and Computing*, Sep. 2006, pp. 27–29.

[20] Y. Qi and T. P. Minka, "Window-based expectation propagation for adaptive signal detection in flat-fading channels," *IEEE Trans. Commun.*, vol. 6, no. 1, pp. 348–355, Jan. 2007.

[21] M. Nissila and S. Pasupathy, "Joint estimation of carrier frequency offset and statistical parameters of the multipath fading channel," *IEEE Trans. Commun.*, vol. 54, no. 6, pp. 1038–1048, June 2006.

ISSN (online): 2246-1248
ISBN (online): 978-87-7112-744-7

AALBORG **UNIVERSITY PRESS**