Local Message Passing on Frustrated Systems

Luca Schmid[1] **Joshua Brenk**[1] **Laurent Schmalen**[1]

[1]Communications Engineering Lab (CEL), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

Abstract

Message passing on factor graphs is a powerful framework for probabilistic inference, which finds important applications in various scientific domains. The most wide-spread message passing scheme is the sum-product algorithm (SPA) which gives exact results on trees but often fails on graphs with many small cycles. We search for an alternative message passing algorithm that works particularly well on such cyclic graphs. Therefore, we challenge the extrinsic principle of the SPA, which loses its objective on graphs with cycles. We further replace the local SPA message update rule at the factor nodes of the underlying graph with a generic mapping, which is optimized in a data-driven fashion. These modifications lead to a considerable improvement in performance while preserving the simplicity of the SPA. We evaluate our method for two classes of cyclic graphs: the 2×2 fully connected Ising grid and factor graphs for symbol detection on linear communication channels with inter-symbol interference. To enable the method for large graphs as they occur in practical applications, we develop a novel loss function that is inspired by the Bethe approximation from statistical physics and allows for training in an unsupervised fashion.

1 INTRODUCTION

Message passing on graphical models is a powerful framework to efficiently solve inference and optimization problems. The most prominent message passing algorithm is the sum-product algorithm (SPA), also known as belief propagation (BP) [Pearl, 1988], which implements exact inference on tree-structured graphs [Kschischang et al., 2001]. Due to its simplicity, the SPA is often applied to cyclic graphs where it becomes an iterative and approximate algorithm. While this works surprisingly well for various applications, such as decoding of low-density parity-check codes [Gallager, 1963], a class of error-correcting codes, the SPA performs poorly on frustrated systems, i.e., on graphs with many cycles and strong coupling between the nodes.

The seminal work of Yedidia et al. [2000] revealed a connection between the SPA and free energy approximations of statistical physics, in particular, the fixed points of BP correspond to stationary points of the Bethe free energy. Based on this insight, alternative message passing methods were proposed which directly minimize the Bethe free energy [Yuille, 2002, Welling and Teh, 2013]. These algorithms are guaranteed to converge to an extremum of the Bethe free energy but are computationally more demanding than plain BP. Wainwright et al. [2003] proposed tree-reweighted BP as a message passing algorithm on the "convexified" Bethe free energy, which is guaranteed to have a global minimum. While this algorithm has stronger convergence guarantees compared to BP, it involves the selection and optimization of so-called edge appearance probabilities, a graph-specific problem that is often non-trivial for practical applications. Yedidia et al. [2000] proposed "generalized BP" as an algorithm that passes messages between regions of nodes instead of single nodes. Larger regions will generally improve the quality of the approximation, however, they also increase the computational complexity.

Recently, model-based deep learning has shown great potential to empower various suboptimal algorithms, such as the SPA on cyclic graphs. Neural BP, proposed by Nachmani et al. [2016], unfolds the iterations of the SPA on its underlying graph and equips the resulting deep network with tunable weights. The GAP algorithm of Schmid and Schmalen [2022] varies the observation model by preprocessing, thereby shaping a graph with more favorable properties with respect to BP performance. Satorras and Welling [2021] extend graph neural networks (GNNs) to factor graphs and propose a hybrid model where BP runs conjointly to a GNN which is structurally identical to the

Accepted for the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023).

original factor graph but has fully parametrized message updates. All these works have in common that they are based on the SPA as a core concept which is vigorously improved using machine learning in order to compensate for its shortcomings on graphs with cycles. In this work, we follow an alternative approach and directly search for alternative message passing algorithms that perform especially well on graphs with cycles, where the SPA tends to fail. To this end, we replace the well-known SPA message update rule with a compact neural network (NN), which is optimized to find a superior local message update rule. Furthermore, we discuss the role of the extrinsic information principle which was originally introduced for tree-structured graphs. Based on the close connection of BP to the Bethe approximation, we propose a novel end-to-end loss function that allows unsupervised and application-agnostic training of new message passing schemes.

2 BACKGROUND

We briefly introduce factor graphs and the SPA as a widespread framework for probabilistic inference on graphical models. We refer the reader to [Kschischang et al., 2001] for an excellent in-depth treatment of the topic.

2.1 FACTOR GRAPHS

Let $f(\mathcal{X})$ be a multivariate function of $\mathcal{X} = \{x_1, \dots, x_N\}$ which factors into a product of *local* functions f_j:

$$f(\mathcal{X}) = \frac{1}{Z} \prod_{j=1}^{J} f_j(\mathcal{X}_j), \quad \mathcal{X}_j \subseteq \mathcal{X}. \tag{1}$$

A *factor graph* visualizes the factorization in (1) as a bipartite graph. Every variable x_n is represented by a unique vertex, a so-called variable node, which we draw as a circle in the graph. Factor nodes represent the local functions f_j and are visualized by squares. The undirected edges of the graph connect a factor node $f_j(\mathcal{X}_j)$ with a variable node x_n if and only if f_j is a function of x_n, i.e., if $x_n \in \mathcal{X}_j$. From a graphical perspective, $\mathcal{X}_j$ thus corresponds to the set of adjacent variable nodes to the factor node f_j. Similarly, we define $\mathcal{N}(x_n)$ to be the set of adjacent factor nodes to the variable node x_n.

In this work, we restrict the variables $x_n \in \{+1, -1\}$ to be binary and the local factors f_j to be either functions of a singular variable x_n or functions of pairs (x_n, x_m) such that the factorization becomes

$$f(\mathcal{X}) = \frac{1}{Z} \prod_{n=1}^{N} \psi_n(x_n) \prod_{(n,m)\in\mathcal{E}} \psi_{n,m}(x_n, x_m), \tag{2}$$

where $\mathcal{E}$ is the set of edges in the graph. Figure 1 shows an exemplary factor graph.

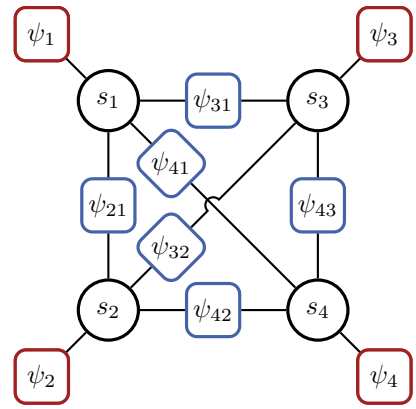

Figure 1: Factor graph representation of (2) with factor nodes of degree 2 (blue) and degree 1 (red). This graph also models the 2×2 fully connected Ising graph of Sec. 2.3.

2.2 SUM-PRODUCT ALGORITHM

The SPA is a message passing algorithm that operates in a factor graph and attempts to determine the marginals of the multivariate function $f(\mathcal{X})$. Messages are propagated between the nodes of the factor graph along its edges and represent interim results of the marginalization. Let $m_{f_j \to x_n}(x_n)$ denote a message sent from a factor node f_j along an edge to a variable node x_n and let $m_{x_n \to f_j}(x_n)$ denote a message on the same edge, but sent in the opposite direction. If the factor graph visualizes a probabilistic model, i.e., if the variable nodes represent random variables, a message $m_{f_j \to x_n}(x_n)$ can be interpreted as a probabilistic statement from node f_j about the random variable x_n to be in one of its possible states [Yedidia et al., 2005]. The SPA defines the updates of the propagating messages at the nodes of the factor graph according to the simple rules [Kschischang et al., 2001]:

$$m_{x \to f_j}(x) = \prod_{f_i \in \mathcal{N}(x) \setminus f_j} m_{f_i \to x}(x) \tag{3}$$

$$m_{f_j \to x}(x) = \sum_{\sim\{x\}} \left(f_j(\mathcal{X}_j) \prod_{x' \in \mathcal{X}_j \setminus x} m_{x' \to f_j}(x') \right). \tag{4}$$

The summary operator $\sum_{\sim\{x\}}$ denotes the marginalization over all variables in $\mathcal{X}_j$ except for x. One key property of the SPA is the *extrinsic information principle* which states that the update of an outgoing message $m_{\mathrm{A}\to\mathrm{B}}$ at node A destined to node B does not depend on the incident message $m_{\mathrm{B}\to\mathrm{A}}$ which travels on the same edge but in opposite direction. For the special case of degree-2 factor nodes $\psi_{n,m}(x_n, x_m)$, the SPA update rule (4) thereby simplifies to

$$m_{\psi_{n,m} \to x_n}(x_n) = \sum_{x_m} \psi_{n,m}(x_n, x_m) \cdot m_{x_m \to \psi_{n,m}}(x_m).$$

Messages at factors nodes $\psi_n(x_n)$ with degree 1 are not updated at all.

Initially, all messages are set to some unbiased state before they are iteratively updated according to a certain schedule. For tree-structured graphs, the messages converge after they have once traveled forward and backward through the entire graph. The result of the SPA, i.e., the marginal functions $f(x_n)$, are finally obtained by a combination of all messages incident to the respective variable nodes:

$$f(x_n) = \prod_{f_i \in \mathcal{N}(x_n)} m_{f_i \to x_n}(x_n).$$

Since the SPA makes no reference to the topology of the factor graph and the message updates are local, the SPA may also be applied to factor graphs with cycles [Yedidia et al., 2003]. On graphs with cycles, the SPA only yields an approximation of the exact marginals. While this approximation works surprisingly well in many cases, even including particular classes of graphs with many small cycles, there are also cases where the results are quite poor or where the SPA does not converge at all [Murphy et al., 1999].

Relation to the Bethe Approximation In their seminal work, Yedidia et al. [2000] showed a revealing connection between the SPA and free energy approximations in statistical physics. From a variational perspective, probabilistic inference can be seen as an optimization problem

$$q = \arg\min_{q \in \mathbb{M}} D_{\text{KL}}(q||p), \tag{5}$$

where we want to find the distribution q from the set $\mathbb{M}$ of all globally valid probability distributions, known as the *marginal polytope* [Wainwright and Jordan, 2008]. Since the Kullback-Leibler (KL) divergence $D_{\text{KL}}(q||p)$ is always non-negative and zero if and only if $q = p$, we reach the minimum exactly for $q = p$. Obviously, optimizing over all possible probability distributions $q \in \mathbb{M}$ is generally intractable. Based on some general assumption, free energy methods simplify the problem in (5) to the minimization of a variational free energy term. We refer the reader to [Yedidia et al., 2005] for a detailed elaboration on this topic.

The Bethe approximation restricts the distribution $q(\mathcal{X})$ to be a product of univariate distributions $b_n(x_n)$ and joint distributions $b_{n,m}(x_n, x_m)$ between pairs $(n, m) \in \mathcal{E}$:

$$q_{\text{Bethe}}(\mathcal{X}) := \prod_{n=1}^{N} b_n(x_n) \prod_{(n,m)\in\mathcal{E}} b_{n,m}(x_n, x_m).$$

This simplification leads to the *Bethe free energy*

$$F_{\text{Bethe}} = \sum_{(n,m)\in\mathcal{E}} \sum_{x_n, x_m} b_{n,m}(x_n, x_m) \log\left(\frac{b_{n,m}(x_n, x_m)}{\phi_{n,m}(x_n, x_m)}\right) - \sum_{n=1}^{N} (|\mathcal{X}_n| - 1) \sum_{x_n} b_n(x_n) \log\left(\frac{b_n(x_n)}{\psi_n(x_n)}\right),$$

with $\phi_{n,m}(x_n, x_m) := \psi_n(x_n)\psi_{n,m}(x_n, x_m)\psi_m(x_m)$. Moreover, the Bethe approximation relaxes the search space in (5) from the marginal polytope $\mathbb{M}$ to the *local polytope*

$$\mathbb{L} = \Big\{\{b_n(x_n), b_{n,m}(x_n, x_m) : \forall n \in [1; N], (n, m) \in \mathcal{E}\}: \sum_{x_n} b_{n,m}(x_n, x_m) = b_m(x_m), \sum_{x_n} b_n(x_n) = 1 \quad \sum_{x_m} b_{n,m}(x_n, x_m) = b_n(x_n)\Big\}.$$

This means that the distributions $b_n(x_n)$ and $b_{n,m}(x_n, x_m)$ only need to locally fulfill consistency in a pairwise sense. In summary, the Bethe approximation converts (5) into the optimization problem

$$q_{\text{Bethe}} = \arg\min_{\{b_n, b_{n,m}\}\in\mathbb{L}} F_{\text{Bethe}}(\{b_n, b_{n,m}\}). \tag{6}$$

Yedidia et al. [2000] showed that the fixed points of BP applied to a factor graph correspond to the stationary points of the respective Bethe free energy. Seen in this light, BP is a suboptimal algorithm to minimize F_{Bethe}. The approximative nature in this sense is twofold: First, there may exist multiple fixed points of the SPA for the same factor graph, i.e., the solution of the (converged) BP might correspond to an extremum of F_{Bethe} other than the global minimum in $\mathbb{L}$ [Knoll et al., 2018]. Second, the beliefs only fulfill the pairwise consistency constraints at the fixed points of BP. This means that the solution only lies within the local polytope $\mathbb{L}$ after BP has converged. However, BP does not necessarily converge and failure of convergence is a major error mode [Yuille, 2002].

Various methods to directly solve (6) or variants thereof were proposed (see [Yedidia et al., 2005] and references therein). Yuille [2002] proposed to decompose the Bethe free energy into concave and convex parts which enables the application of a concave-convex procedure (CCCP). That algorithm consists of a double loop where the outer loop iteratively minimizes F_{Bethe} and the inner loop ensures that the pairwise consistency constraints are fulfilled. Due to the CCCP, the algorithm provably converges to an extremum of the Bethe free energy.

2.3 EXAMPLES

For the remainder of this section, we introduce two important classes of factor graphs which are the basis for the numerical experiments in Sec. 4.

Example 1 - Ising Graphs We consider factor graphs with $N = M^2$ variable nodes, arranged in a square 2D lattice, in which pairs of adjacent variable nodes (x_n, x_m) are symmetrically coupled by the weights $J_{n,m}$ via factor nodes $\psi_{n,m}(x_n, x_m) = \exp(J_{n,m} x_n x_m)$. Additionally,

each variable node x_n has local evidence in the form of a degree-1 factor node $\psi_n(x_n) = \exp(\theta_n x_n)$. The Ising model originates from statistical physics where the binary variables $x_n \in \{+1, -1\}$ represent the orientation of elementary magnets in a lattice [Peierls, 1936]. Each magnet is exposed to a local field θ_n and is influenced by its neighbors via an assigned pairwise coupling $J_{n,m}$. Besides its fundamental significance in statistical physics, the Ising model is a universal mathematical model and finds applications in many other scientific domains such as image processing [Besag, 1986] and modeling of social networks [Banerjee and El Ghaoui, 2008, Wainwright and Jordan, 2008].

Following [Yedidia et al., 2005, Mooij and Kappen, 2007, Knoll et al., 2018], we study the fully connected 2×2 Ising model, i.e., $M = 2$ and $N = 4$, where every pair of variable nodes is connected. A factor graph representation of this model is given in Fig. 1. With more cycles than variable nodes and a girth of 3, this graph can be parametrized to a highly frustrated system and is thus able to highlight the weaknesses of the SPA [Yedidia et al., 2005]. In particular, we consider the Ising spin glass, where the parameters θ_n and $J_{n,m}$ are independent and identically distributed (i.i.d.) random variables, sampled from a uniform distribution $\mathcal{U}[-S, +S]$ with $S \in \mathbb{R}^+$. We are interested in the computation of the marginal functions

$$f(x_n) = \sum_{\sim\{x_n\}} f(x_1, x_2, x_3, x_4), \quad n = 1, 2, 3, 4, \tag{7}$$

which correspond to marginal probability distributions $p(x_n) = f(x_n)$ if the Ising graph represents a probabilistic model. While the direct computation of (7) is still feasible for our example with $N = 4$, the number of summations grows exponentially with N, which calls for alternative methods with lower complexity. Applying the SPA on the factor graph in Fig. 1 yields the single beliefs $b_n(x_n)$ as an approximation of $f(x_n)$ with a complexity that only grows quadratically with N.

Example 2 - Symbol Detection We study the problem of symbol detection in a digital communication system [Proakis and Salehi, 2007]. A transmitter sends a sequence of N independent and uniformly distributed symbols $c_n \in \{+1, -1\}$ over a linear channel with memory, impaired by additive white Gaussian noise (AWGN). The receiver observes the sequence

$$\boldsymbol{y} = \underbrace{\begin{pmatrix} h_0 & & & \\ \vdots & h_0 & \mathbf{0} & \\ h_L & \vdots & \ddots & \\ & h_L & & h_0 \\ & \mathbf{0} & \ddots & \vdots \\ & & & h_L \end{pmatrix}}_{=:\boldsymbol{H}} \underbrace{\begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_N \end{pmatrix}}_{=:\boldsymbol{c}} + \underbrace{\begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_{N+L} \end{pmatrix}}_{=:\boldsymbol{w}}, \tag{8}$$

where $\boldsymbol{h} \in \mathbb{R}^{L+1}$ describes the impulse response of the channel of length $L + 1$ and $w_k \sim \mathcal{CN}(0, \sigma^2)$ are independent noise samples from a complex circular Gaussian distribution. Applying Bayes' theorem, the posterior distribution $p(\boldsymbol{c}|\boldsymbol{y})$ can be expressed in terms of the likelihood:

$$p(\boldsymbol{c}|\boldsymbol{y}) = \frac{1}{Z} p(\boldsymbol{y}|\boldsymbol{c}) = \frac{1}{Z} \exp\left(-\frac{(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{c})^2}{\sigma^2} \right).$$

In the context of symbol detection, we want to infer the transmit symbols c_n based on the channel observation $\boldsymbol{y}$, i.e., we are interested in the marginal distributions $p(c_n|\boldsymbol{y})$. Based on an observation model by Ungerboeck [1974]

$$p(\boldsymbol{y}|\boldsymbol{c}) \propto \exp\left(\frac{2\mathrm{Re}\left\{\boldsymbol{c}^{\mathrm{H}} \boldsymbol{H}^{\mathrm{H}} \boldsymbol{y}\right\} - \boldsymbol{c}^{\mathrm{H}} \boldsymbol{H}^{\mathrm{H}} \boldsymbol{H} \boldsymbol{c}}{\sigma^2} \right),$$

we can factorize the likelihood

$$p(\boldsymbol{y}|\boldsymbol{c}) = \frac{1}{Z} \prod_{n=1}^{N} \left[F_n(c_n) \prod_{\substack{m=1 \\ m<n}}^{N} I_{n,m}(c_n, c_m) \right] \tag{9}$$

into the factors

$$F_n(c_n) := \exp\left(\frac{1}{\sigma^2} \mathrm{Re}\{2 x_n c_n^\star - G_{n,n} |c_n|^2\} \right)$$
$$I_{n,m}(c_n, c_m) := \exp\left(-\frac{2}{\sigma^2} \mathrm{Re}\{G_{n,m} c_m c_n^\star\} \right),$$

where $\boldsymbol{x} := \boldsymbol{H}^{\mathrm{H}} \boldsymbol{y}$ and $\boldsymbol{G} := \boldsymbol{H}^{\mathrm{H}} \boldsymbol{H}$ are the matched filtered versions of the observation and the channel matrix, respectively. Modeling a factor graph based on (9) and applying the SPA yields a low-complexity symbol detection algorithm, originally proposed by Colavolpe et al. [2011].

3 MESSAGE PASSING FOR CYCLIC GRAPHS

Despite its drawbacks on cyclic graphs, the amazing success of the SPA lies in its simplicity and generality: it is only defined by a *local* message update rule which can be applied to any generic factor graph based on a suitable message update schedule. Driven by this elegant concept, we are interested in finding message passing algorithms that perform well on graphs with many cycles where the SPA fails. More specifically, we ask the following questions:

- If the SPA fails to converge, does an alternative *local* message update rule exist that converges (possibly to an extremum of the Bethe free energy) and which provides better results than the SPA?
- If the SPA converges to an extremum of the Bethe free energy, is there a local message update rule which yields superior performance, either because the SPA

converges to a fixed point which only corresponds to a local instead of global minimum of the Bethe free energy, or because the Bethe approximation itself is a bad approximation in this case?

3.1 ON MESSAGE UPDATE RULES

A message update rule defines a mapping from one or multiple incident messages to one outgoing message, which is applied *locally* at the variable or factor nodes of a factor graph. Besides the initialization of the messages and their update schedule, these mappings fully define a graph-based inference algorithm. The SPA update rule at the variable nodes (3) is simply the product of all extrinsic messages. We adopt this quite intuitive aggregation principle and focus on finding a message update rule for the factor nodes, i.e., an alternative to (4). For factor nodes of degree 2, such as in (2), the update rule simplifies to a mapping from one single incident message to one outgoing message:

$$\mathrm{FN_e}(\psi_{n,m}) : m_{x_n \to \psi_{n,m}}(x_n) \mapsto m_{\psi_{n,m} \to x_m}(x_m). \quad (10)$$

If the pairwise factors $\psi_{n,m}(x_n, x_m)$ are symmetric with regard to x_n and x_m, and follow the exponential form

$$\psi_{n,m}(x_n, x_m) = \exp(E_{n,m} x_n x_m),\ x_n, x_m \in \{+1, -1\},$$

we can distill the dependency from the function $\psi_{n,m}$ to the scalar parameter $E_{n,m} \in \mathbb{R}$, which quantifies the repulsive ($E_{n,m} < 0$) or attractive ($E_{n,m} > 0$) coupling between the nodes x_n and x_m. This directly coincides with the pairwise coupling weights $J_{n,m} = E_{n,m}$ of the Ising model in Example 1. The factor nodes $I_{n,m}$ of Example 2 can be reduced to the coupling parameters $E_{n,m} = -2G_{n,m}/\sigma^2$.

Challenging the Extrinsic Principle Most of the existing message passing algorithms follow the extrinsic information principle. For instance in turbo decoding, it is known to be an important property of good message passing decoders [Richardson and Urbanke, 2001]. Ensuring that only extrinsic messages are received, it prevents backcoupling of intrinsic information in tree-structured graphs, which would otherwise lead to a self-enhancement of the messages, also known as "double counting". Thereby, it guarantees that the SPA is exact on trees [Kuck et al., 2020]. We argue that this is in general not valid for cyclic graphs where backcoupling of messages is inevitable due to the very nature of the cycles. Therefore, we propose a second message update rule which operates contradictory to the extrinsic principle: instead of ignoring the intrinsic message, the message update should rather actively leverage this additional information, e.g., to ensure that local consistency between neighboring nodes is fulfilled.

Without the extrinsic principle, we need to reconsider the messages from degree-1 factor nodes which are then also subject to iterative updates. To avoid an increase in complexity due to additional message updates at the degree-1 factor nodes, we apply a clustering approach similar to [Rapp et al., 2022]. We split up the single factors $\psi_n(x_n)$ into $|\mathcal{X}_n|$ parts $\Psi_n(x_n) := (\psi_n(x_n))^{\frac{1}{|\mathcal{X}_n|}}$ and merge them into the adjacent pairwise factors $\psi_{n,m}(x_n, x_m)$, such that the new clustered factors are

$$\Psi_{n,m}(x_n, x_m) := \Psi_n(x_n)\psi_{n,m}(x_n, x_m)\Psi_m(x_m).$$

The overall factorization (2) simplifies to

$$f(x_1, \ldots, x_N) = \frac{1}{Z} \prod_{(n,m)\in\mathcal{E}} \Psi_{n,m}(x_n, x_m),$$

which leads to the non-extrinsic mapping

$$\mathrm{FN}(\Psi_{n,m}) : \begin{pmatrix} m_{x_n \to \Psi_{n,m}}(x_n) \\ m_{x_m \to \Psi_{n,m}}(x_m) \end{pmatrix} \mapsto m_{\Psi_{n,m} \to x_m}(x_m). \quad (11)$$

If the single factors are in exponential form

$$\Psi_n(x_n) = \exp\left(E_n x_n\right), \quad x_n \in \{+1, -1\},$$

the clustered factors $\Psi_{n,m}(x_n, x_m)$ are fully characterized by the three scalars $E_n, E_{n,m}$ and E_m.

3.1.1 Neural Networks as Function Approximators

Finding suitable mappings (10) or (11) such that the overall message passing algorithm performs well is generally non-trivial. We employ feed-forward NNs, known to be efficient universal function approximators [Hornik et al., 1989], to reduce the search space of all possible mappings to a set of weights and biases $\mathcal{P}$, which fully parametrize the NN. At a factor node f_j, the network accepts N_{in} inputs and produces the updated outgoing message $m_{f_j \to x_n}$. For factor graphs with binary variables x_n, the messages $m_{f_j \to x_n}(x_n)$ can be expressed in scalar log-likelihood ratios (LLRs)

$$L_{f_j \to x_n} := \log\left(\frac{m_{f_j \to x_n}(x_n = +1)}{m_{f_j \to x_n}(x_n = -1)}\right).$$

A similar definition holds for the LLRs $L_{x_n \to f_j}$ based on the messages $m_{x_n \to f_j}(x_n)$. For the extrinsic update (10), there are $N_{\text{in}} = 2$ inputs: the LLR of the incoming extrinsic message and the coupling parameter $E_{n,m}$ of the local factor node. Without the extrinsic principle, the NN furthermore accepts the LLR of the intrinsic message as well as E_n and E_m, i.e., in total $N_{\text{in}} = 5$ inputs. Since we only approximate a local mapping from a few scalar inputs to a single output, we can choose a very compact NN structure with a single hidden layer and 7 neurons, as summarized in Table 1.

Having set up the NN structure, we are able to define a convenient message update rule by appropriately tuning the parameterization $\mathcal{P}$ of the NN. We are interested in

Table 1: NN Architecture

Layer (linear)	Activation	Dimension
Input	ReLU	$(N_{\text{in}}, 7)$
Hidden	Tanh	$(7, 7)$
Output	Linear	$(7, 1)$

a local update rule such that the overall message passing performs well. To this end, we optimize $\mathcal{P}$ with respect to an objective function that evaluates the end-to-end performance of the inference task. Therefore we apply a fixed number of message passing iterations and back-propagate the gradient of the objective function in order to iteratively optimize $\mathcal{P}$ using gradient descent based on a representative set of examples. Note that this data-driven approach inevitably leads to a specialization of the learned message update to the data. However, we expect the result to be fairly generic and to have good generalization capabilities since we only optimize very few parameters in an otherwise model-aware system. Moreover, despite the end-to-end optimization, we only use a single message update rule for the entire factor graph, i.e., we employ the same instance of the NN for the message updates at all factor nodes and in each iteration[1].

We note that our approach can be interpreted as a special instance of a GNN as, e.g., described by Yoon et al. [2019]. In comparison, our model passes scalar messages instead of high-dimensional vectors and does not use any hidden states or embeddings at the variable nodes. For this reason, we do not require a second NN with a gated recurrent unit, as used in [Yoon et al., 2019] to update the hidden states based on the aggregated messages. Furthermore, we do not require a third NN which implements a trainable readout function to interpret the final node embeddings.

3.2 END-TO-END OBJECTIVE FUNCTIONS

In the generic context of marginal inference, we hope to find a good approximation of the true marginals. A convenient objective function is the KL divergence which measures a type of statistical distance between the beliefs $b_n(x_n)$ and the exact marginal distributions $p(x_n) = \sum_{\sim\{x_n\}} p(x_1, \ldots, x_N)$:

$$\mathcal{L}_{\text{KL}} := D_{\text{KL}}\left(b_n(x_n) \| p(x_n)\right). \tag{12}$$

For large graphs, the computation of $p(x_n)$ might be infeasible, and $\mathcal{L}_{\text{KL}}$ becomes impractical. Therefore, we propose alternative loss functions in what follows.

[1]As a consequence, the training procedure of the NN is not entirely local because the local copies of the NN at each factor node must be globally synchronized during optimization. However, the local nature of the message updates is still retained.

The training of a symbol detector as in Example 2 is a typical supervised learning scenario where the labels are given by the transmitted symbols c_n. An appropriate performance measure for symbol detection is the bitwise mutual information (BMI) which is an achievable information rate[2] for our scenario [Guillén i Fàbregas et al., 2008]. By a sample mean estimate over D labeled examples $(\boldsymbol{c}, \boldsymbol{y})$ from the data batch $\mathcal{D}$, the BMI can be approximated by

$$\text{BMI} \approx 1 - \frac{1}{DN} \sum_{n=1}^{N} \sum_{(\boldsymbol{c},\boldsymbol{y})\in\mathcal{D}} \log_2\Big(\mathrm{e}^{-c_n L_n(\boldsymbol{y})} + 1\Big),$$

where $L_n(\boldsymbol{y})$ denotes the LLR from the belief $b_n(c_n)$ [Alvarado et al., 2018].

Other applications such as the Ising model in Example 1 relate to the class of unsupervised problems if the true marginals are not accessible. For such scenarios, we consider a novel and application-agnostic objective function in the following. Inspired by the Bethe approximation, which is known to yield excellent results for many applications, even in cases where the SPA performs poorly [Yuille, 2002], we propose a regularized minimization of the Bethe free energy:

$$\mathcal{L}_{\text{Bethe}} := F_{\text{Bethe}} + \alpha \mathcal{L}_{\mathbb{L}}, \quad \alpha \in \mathbb{R}^+. \tag{13}$$

To ensure local consistency, we introduce the *Bethe consistency distance*

$$\mathcal{L}_{\mathbb{L}} := D_{\text{KL}}\left(\sum_{x_m} b_{n,m}(x_n, x_m) \middle\| b_n(x_n)\right) + D_{\text{KL}}\left(\sum_{x_n} b_{n,m}(x_n, x_m) \middle\| b_m(x_m)\right)$$

as a type of distance measure between the solution of the approximative inference $\{b_n, b_{n,m}\}$ and the local polytope $\mathbb{L}$. The weight α in (13) is a hyperparameter that controls how strictly the local consistency is enforced. With this penalty term $\mathcal{L}_{\mathbb{L}}$, we hope to suppress oscillations in the message passing, as they occur in the SPA for graphs with strong coupling.

4 EXPERIMENTS

We consider the examples of Sec. 2.3 for numerical evaluation. To enable a deeper analysis, we fix the number of variable nodes to $N = 4$ such that the computation of the true marginals is feasible. Despite this rather small extent, these models lead to factor graphs with a high density of short cycles and are thus expressive examples to highlight the weaknesses of the SPA. Furthermore, we fix the global

[2]In our case, where the symbols c_n follow a Rademacher distribution, the BMI is equivalent to the mutual information.

Table 2: Behavior of the Novel Message Passing Algorithm cycBP for the 2×2 Spin Glass, Averaged over 10^5 Graphs

Algo.	Loss	$\mathcal{L}_{\text{KL}}$	$\hat{\sigma}_{\mathcal{L}_{\text{KL}}}$	F_{Bethe}	$\mathcal{L}_{\mathbb{L}}$
SPA	-	0.087	0.265	-7.50	0.30
SPA$_\mu$	-	0.035	0.113	-7.49	0.12
CCCP	-	0.044	0.094	-7.24	$2 \cdot 10^{-6}$
cycBP$_\text{e}$	$\mathcal{L}_{\text{KL}}$	0.040	0.068	-7.37	0.17
cycBP	$\mathcal{L}_{\text{KL}}$	0.014	0.023	-7.37	0.48
cycBP$_\text{e}$	$\mathcal{L}_{\text{Bethe}}$	0.030	0.054	-7.38	0.11
cycBP	$\mathcal{L}_{\text{Bethe}}$	0.027	0.057	-7.47	0.027

settings of the message passing to standard choices: all LLR messages are initialized with zero and we perform 10 iterations of a parallel schedule, i.e., each iteration comprises the parallel update of all messages at the factor nodes followed by message updates at all variable nodes.

A common technique to improve the performance of the SPA on graphs with cycles is the use of "momentum", i.e., replacing a message $L^{(t)}$ of the SPA in iteration t with the weighted average $(1-\mu)L^{(t)} + \mu L^{(t-1)}$ [Murphy et al., 1999]. By choosing $0 < \mu < 1$, the idea is to improve the convergence behavior of the message passing scheme compared to the original SPA ($\mu = 0$) while retaining the same fixed points. As in [Murphy et al., 1999], we set $\mu = 0.1$ and use this variant of the SPA as an additional baseline in the following experiments, where we refer to it as SPA$_\mu$.

Besides the SPA, we similarly apply message passing based on the newly proposed update rule (11). We call the resulting inference algorithm cycBP (BP for cyclic graphs). If we use the extrinsic update rule (10), we denote the algorithm with cycBP$_\text{e}$. We also consider the CCCP for the Bethe free energy as defined in [Yuille, 2002], since it gives interesting insights into the quality of the Bethe approximation. For the double loop, we apply 25 outer iterations, each comprising 25 inner iterations.

Ising model We study the 2×2 fully connected spin glass model of Example 1 for $S = 2$, i.e., all parameters θ_n and $J_{n,m}$ are independently sampled from a uniform distribution $\mathcal{U}[-2, +2]$. Table 2 evaluates the behavior of all discussed inference schemes, averaged over 10^5 different graphs. $\hat{\sigma}_{\mathcal{L}_{\text{KL}}}$ denotes the empirical standard deviation of $\mathcal{L}_{\text{KL}}$ of the individual graphs from the empirical mean. We can observe that the SPA does not leverage the full potential of the Bethe approximation, since the average loss $\mathcal{L}_{\text{KL}} = 0.087$ of the SPA is twice as large compared to $\mathcal{L}_{\text{KL}} = 0.044$ for the CCCP. Although the SPA reaches on average a smaller F_{Bethe} than the CCCP, the beliefs of the SPA show local inconsistencies with $\mathcal{L}_{\mathbb{L}} = 0.3$ due to non-convergent behavior. Using "momentum" in the SPA message updates can help to mitigate this behavior: the SPA$_\mu$ shows improved pairwise consistency $\mathcal{L}_{\mathbb{L}} = 0.12$ and also yields in average a better approximation of the true marginals ($\mathcal{L}_{\text{KL}} = 0.035$). The CCCP has a vanishing Bethe consistency distance $\mathcal{L}_{\mathbb{L}}$, i.e., the results of the CCCP lie within the local polytope $\mathbb{L}$. We search for alternative message update rules, by optimizing $\mathcal{P}$ of the NN-based mappings towards minimal $\mathcal{L}_{\text{KL}}$. The training batches are sampled from a spin glass model with $S = 3$ to put more emphasis on graphs with strong coupling, where the SPA is known to be susceptible to convergence errors. The results in Tab. 2 show that there indeed exist superior message update rules to the SPA for this class of cyclic graphs. Using the extrinsic update rule (10), the cycBP$_\text{e}$ algorithm reaches $\mathcal{L}_{\text{KL}} = 0.04$ and thereby outperforms the original SPA as well as the CCCP.

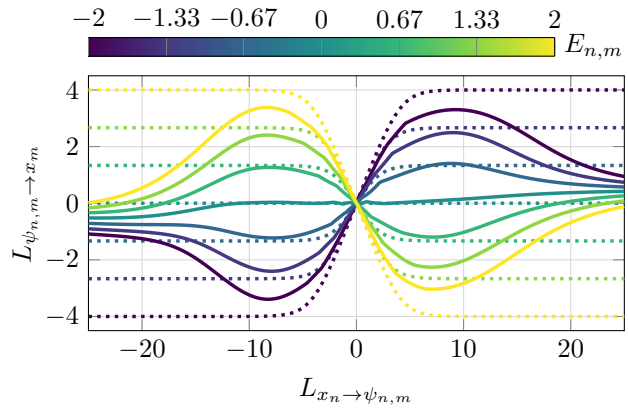

Figure 2: The mapping (10) of the cycBP$_\text{e}$ algorithm, optimized for the 2×2 spin glass with $S = 3$ (solid), in comparison with the mapping of the SPA (dotted), plotted for different coupling $E_{n,m}$.

We visualize the message update rule of the cycBP$_\text{e}$ algorithm in Fig. 2 by plotting the optimized mapping (10) from the incoming LLR message $L_{x_n \to \psi_{n,m}}$ to the outgoing LLR message $L_{\psi_{n,m} \to x_m}$. Similar to the SPA, the mapping is point-symmetric to the origin. The major difference is the behavior for incident LLR messages with high magnitudes $|L_{x_n \to \psi_{n,m}}| > 8$, where the outgoing messages are heavily attenuated. Intuitively, this behavior reduces the potential of oscillation in graphs with strong coupling $E_{n,m}$. We can further improve the inference performance by disabling the extrinsic principle in the message passing procedure. The resulting algorithm cycBP can be interpreted as a generalization of cycBP$_\text{e}$ and outperforms the latter with $\mathcal{L}_{\text{KL}} = 0.014$, as reported in Tab. 2. It also yields a superior approximation of the true marginals compared to the momentum-based SPA$_\mu$, although the Bethe consistency distance $\mathcal{L}_{\mathbb{L}} = 0.48$ is relatively high in this case.

Moreover, we consider unsupervised training towards the proposed loss function $\mathcal{L}_{\text{Bethe}}$. For the cycBP$_\text{e}$ algorithm, the unsupervised training leads to a smaller loss $\mathcal{L}_{\text{KL}} = 0.03$ compared to the supervised training, i.e., the loss function $\mathcal{L}_{\text{Bethe}}$ is better suited for the optimization via stochastic gra-

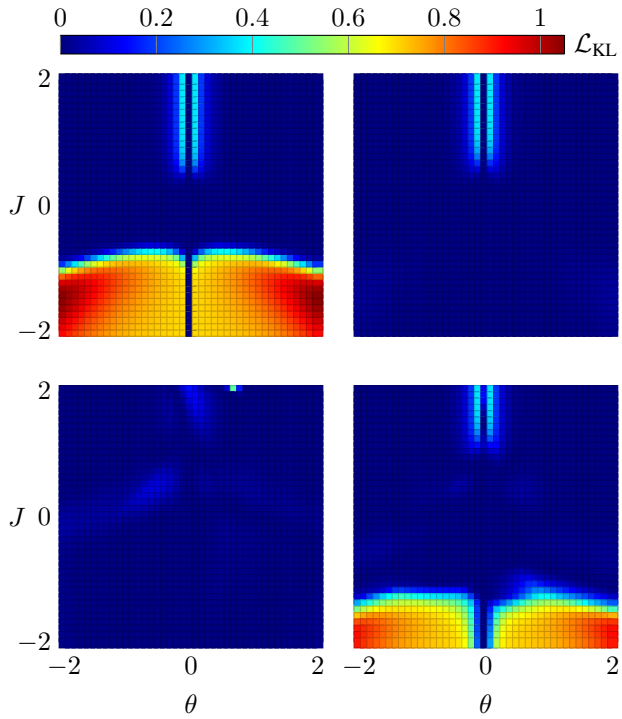

Figure 3: Approximation error $\mathcal{L}_{\mathrm{KL}}$ on the 2×2 Ising model with constant parameters θ and J: SPA (top left), CCCP (top right), cycBP optimized on spin glasses w.r.t. $\mathcal{L}_{\mathrm{KL}}$ (bottom left), and cycBP$_{\mathrm{e}}$ trained with $\mathcal{L}_{\mathrm{Bethe}}$ (bottom right).

dient descent than the loss function $\mathcal{L}_{\mathrm{KL}}$ in this case. In the unsupervised training towards $\mathcal{L}_{\mathrm{Bethe}}$, we observe substantial differences between the two variants cycBP$_{\mathrm{e}}$ and cycBP: while the optimization of cycBP$_{\mathrm{e}}$ converges reliably, the training of cycBP is unstable and the optimization needs to run multiple times with different initializations for $\mathcal{P}$ until a reasonable result is obtained. This behavior is also reflected in the results in Tab. 2, where the cycBP algorithm shows a degraded performance with $\mathcal{L}_{\mathrm{KL}} = 0.027$, compared to the supervised training ($\mathcal{L}_{\mathrm{KL}} = 0.014$). We conjecture that this is accounted for by the local consistency constraint $\mathcal{L}_{\mathbb{L}}$, which can be directly enforced at the message update at the factor nodes if the intrinsic message also takes part in the update. Optimization of the hyperparameter α did not lead to considerable changes in this behavior and we used $\alpha = 25$ for all presented results.

To analyze the convergence properties on highly frustrated systems, we consider the Ising model with constant parameters θ and J. Note that we do not specifically optimize the models for this scenario, but rather use the previous parametrization $\mathcal{P}$ which is optimized for spin glasses with $S = 3$. Figure 3 plots $\mathcal{L}_{\mathrm{KL}}$ over θ and J for different inference algorithms. Knoll et al. [2018] showed that the Bethe free energy has a unique minimum in the complete antiferromagnetic domain ($J < 0$) and in large parts of the ferromagnetic case ($J > 0$), except for a region around $\theta = 0$,

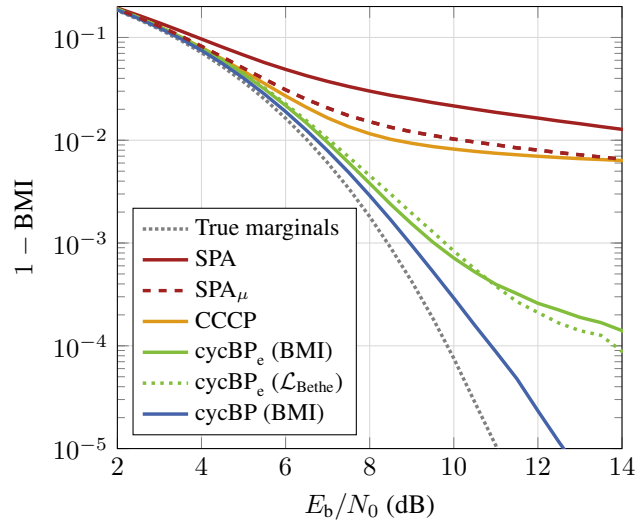

Figure 4: Detection performance of the proposed cycBP algorithm, averaged over 10^7 random channels.

where F_{Bethe} has two minima in $\mathbb{L}$. This coincides with our findings of the approximation error $\mathcal{L}_{\mathrm{KL}}$ for the CCCP in Fig. 3. The SPA shows failure to converge in large parts of the antiferromagnetic region with $J < -1$, where it does not converge to the unique fixed point and produces large approximation errors. The extrinsic message passing scheme cycBP$_{\mathrm{e}}$, optimized towards $\mathcal{L}_{\mathrm{Bethe}}$, shows an improved behavior. However, in the antiferromagnetic case with strong repellings ($J < -1.5$), there are still considerable approximation errors. The non-extrinsic message passing algorithm cycBP shows good inference capabilities over the complete considered region if it is optimized towards $\mathcal{L}_{\mathrm{KL}}$. The unsupervised training with respect to $\mathcal{L}_{\mathrm{Bethe}}$ leads to a similar performance as the CCCP, however, the training procedure is again relatively unstable in this case.

Symbol Detection We further consider the factor graphs of Example 2 for approximate symbol detection on linear channels with memory $L = 2$. To generate random channels, we independently sample each tap h_ℓ of the channel impulse response for every example from a Gaussian distribution with zero mean and unit variance and subsequently normalize each channel to unit energy $\|\boldsymbol{h}\|_2 = 1$. Figure 4 evaluates the detection performance of the considered inference algorithms in terms of the BMI over the signal-to-noise ratio $E_{\mathrm{b}}/N_0 = 1/\sigma^2$. Both, the SPA and the CCCP run into an error floor for high E_{b}/N_0, where the graphs tend to have strong coupling via the factor nodes $I_{n,m}(c_n, c_m)$. The momentum-based message updates of the SPA$_\mu$ enhance the original SPA in the entire E_{b}/N_0 range under consideration and close the performance gap to the CCCP. During the optimization of the new message update rules, the E_{b}/N_0 in dB was sampled from $\mathcal{U}[0, 16]$ for each batch element independently. To help the update rule adapt to different channel realizations, i.e., different

E_b/N_0 and different channel taps $\boldsymbol{h}$, we feed E_b/N_0 and $\boldsymbol{h}$ as additional inputs to the NN. Figure 4 shows that the optimized algorithms cycBP$_e$ and cycBP clearly outperform the SPA and the CCCP, especially for high E_b/N_0. Consistently with our findings on the Ising graphs, the cycBP algorithm performs better than the extrinsic variant cycBP$_e$. The latter can also be trained towards $\mathcal{L}_{\text{Bethe}}$ without degrading the detection performance. This is particularly surprising since it thereby clearly outperforms the CCCP. The optimization of the cycBP algorithm towards $\mathcal{L}_{\text{Bethe}}$ does not converge and is therefore not shown in Fig. 4. However, since training towards the BMI is feasible for large N, the supervised training of the cycBP algorithm yields an attractive algorithm with low complexity and superior performance, which can be highly relevant for practical applications.

4.1 DISCUSSION

To investigate the two central questions which we formulated in Sec. 3 and to show the potential of our method, we investigated compact models with $N = 4$. These models are expressive examples for analysis since they have a high density of short cycles and because the true marginals are available as ground truth data. However, verifying the capability of the proposed cycBP algorithm for practical applications requires extensive numerical evaluation on larger graphs and varying graph structures. This is ongoing work and our preliminary results are promising.

5 CONCLUSION

This work considered message passing for approximate inference and showed the existence of message update rules which perform especially well on cyclic graphs where the SPA fails. We challenged the extrinsic information principle for cyclic graphs and proposed an alternative message update rule which also takes intrinsic information into account. The gain was demonstrated by numerical experiments on two exemplary classes of factor graphs. The learned message update rules generalize well and training is extremely fast since the update rule is defined by a very compact NN that is reused at all factor nodes. We furthermore proposed a novel unsupervised and application-agnostic loss function that follows the idea of the Bethe approximation.

Acknowledgements

This work has received funding in part from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 101001899) and in part from the German Federal Ministry of Education and Research (BMBF) within the project Open6GHub (grant agreement 16KISK010).

References

Alex Alvarado, Tobias Fehenberger, Bin Chen, and Frans M. J. Willems. Achievable information rates for fiber optics: Applications and computations. *J. Lightw. Technol.*, 36(2):424–439, 2018.

Onureena Banerjee and Laurent El Ghaoui. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *J. Mach. Learn. Re.*, 9:485–516, 2008.

Julian Besag. On the statistical analysis of dirty pictures. *J. R. Stat. Soc. Ser. B. Stat. Soc.*, 48(3):259–302, 1986.

Giulio Colavolpe, Dario Fertonani, and Amina Piemontese. SISO detection over linear channels with linear complexity in the number of interferers. *IEEE J. Sel. Topics Signal Process.*, 5(8), 2011.

Robert G. Gallager. *Low-density parity check codes*. PhD thesis, MIT Press, 1963.

Albert Guillén i Fàbregas, Alfonso Martinez, and Giuseppe Caire. Bit-interleaved coded modulation. In *Found. Trends Commun. Inf. Theory*, volume 5. Now Publishers, Delft, NL, 2008.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, January 1989.

Christian Knoll, Dhagash Mehta, Tianran Chen, and Franz Pernkopf. Fixed points of belief propagation – an analysis via polynomial homotopy continuation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(9):2124–2136, 2018.

Frank R. Kschischang, Brendan J. Frey, and Hans-Andrea Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. Inf. Theory*, 47(2):22, 2001.

Jonathan Kuck, Shuvam Chakraborty, Hao Tang, Rachel Luo, Jiaming Song, Ashish Sabharwal, and Stefano Ermon. Belief propagation neural networks. *Adv. Neural Inf. Process. Syst.*, 33:667–678, 2020.

Joris M. Mooij and Hilbert J. Kappen. Sufficient conditions for convergence of the sum-product algorithm. *IEEE Trans. Inf. Theory*, 53(12):4422–4437, 2007.

Kevin Murphy, Yair Weiss, and Michael I. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proc. Innov. Appl. Artif. Intell. Conf.*, 1999.

Eliya Nachmani, Yair Be'ery, and David Burshtein. Learning to decode linear codes using deep learning. In *Proc. Annu. Allerton Conf. Commun., Control, Comput.*, Monticello, IL, 2016.

Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

Rudolf Peierls. On Ising's model of ferromagnetism. *Math. Proc. Camb. Philos. Soc.*, 32:477–481, 1936.

John Proakis and Massoud Salehi. *Digital Communications*. McGraw Hill, 5 edition, 2007.

Lukas Rapp, Luca Schmid, Andrej Rode, and Laurent Schmalen. Structural optimization of factor graphs for symbol detection via continuous clustering and machine learning. *arXiv:2211.11406*, 2022.

Thomas J. Richardson and Rüdiger L. Urbanke. The capacity of low-density parity-check codes under message-passing decoding. *IEEE Trans. Inf. Theory*, 47(2):599–618, 2001.

Victor Garcia Satorras and Max Welling. Neural enhanced belief propagation on factor graphs. In *Proc. Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, pages 685–693, San Diego, CA, USA, 2021.

Luca Schmid and Laurent Schmalen. Low-complexity near-optimum symbol detection based on neural enhancement of factor graphs. *IEEE Trans. Commun.*, 70(11):7562–7575, November 2022.

Gottfried Ungerboeck. Adaptive maximum-likelihood receiver for carrier-modulated data-transmission systems. *IEEE Trans. Commun.*, COM-22(5):624–636, 1974.

Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach.*, 1(1–2):1–305, 2008. Number: 1–2.

Martin J. Wainwright, Tommi S. Jaakkola, and Alan S. Willsky. Tree-reweighted belief propagation algorithms and approximate ML estimation by pseudo-moment matching. *Proc. Int. Workshop Artificial Intelligence and Statistics*, R4(PMLR R4):308–315, 2003.

Max Welling and Yee Whye Teh. Belief optimization for binary networks: A stable alternative to loopy belief propagation. In *Proc. Conf. on Uncertainty in Artificial Intelligence (UAI)*, pages 554–561, 2013.

Jonathan S. Yedidia, William T. Freeman, and Yair Weiss. Generalized belief propagation. *Advances in Neural Information Processing Systems*, 13, 2000.

Jonathan S. Yedidia, William T. Freeman, and Yair Weiss. Understanding belief propagation and its generalizations. In *Exploring Artificial Intelligence in the New Millennium*, volume 8, pages 236–239. Morgan Kaufmann Publishers Inc., 2003.

Jonathan S. Yedidia, William T. Freeman, and Yair Weiss. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Trans. Inform. Theory*, 51(7):2282–2312, 2005.

KiJung Yoon, Renjie Liao, Yuwen Xiong, Lisa Zhang, Ethan Fetaya, Raquel Urtasun, Richard Zemel, and Xaq Pitkow. Inference in probabilistic graphical models by graph neural networks. In *Proc. Asilomar Conf. Signals Syst. Comput.*, 2019.

Alan L. Yuille. CCCP algorithms to minimize the Bethe and Kikuchi free energies: Convergent alternatives to belief propagation. *Neural Computation*, 14(7):1691–1722, 2002.