α Belief Propagation as Fully Factorized Approximation

Dong Liu[1], Nima N. Moghadam[2], Lars K. Rasmussen[1], Jinliang Huang[2], Saikat Chatterjee[1]

[1] KTH Royal Institute of Technology, Stockholm, Sweden.

[2] Huawei Technologies Sweden AB, Stockholm, Sweden.

e-mail: {doli, lkra, sach}@kth.se, {nima.najari.moghadam1, jinliang.huang}@huawei.com

Abstract—Belief propagation (BP) can do exact inference in loop-free graphs, but its performance could be poor in graphs with loops, and the understanding of its solution is limited. This work gives an interpretable belief propagation rule that is actually minimization of a localized α-divergence. We term this algorithm as α belief propagation (α-BP). The performance of α-BP is tested in MAP (maximum a posterior) inference problems, where α-BP can outperform (loopy) BP by a significant margin even in fully-connected graphs.

I. Introduction

Bayesian inference provides a general mathematical framework for many learning tasks such as classification, denoising, object detection, and signal detection. The wide applications include but not limited to imaging processing [1], multi-input-multi-output (MIMO) signal detection in digital communication [2], inference on structured lattice [3], machine learning [4]–[6]. Specifically, statistic properties of a hidden variable $\boldsymbol{x} = \{x_1, \dots, x_N\}$ are of common interests in Bayesian inference. Practical interests usually include finding joint probability $p(\boldsymbol{x})$, marginal probability $p_i(x_i)$, the most probable state $\operatorname{argmax}_{\boldsymbol{x}} p(\boldsymbol{x})$. It can be extended to maximum a posterior (MAP) inference when it is conditional on some observation $(\operatorname{argmax}_{\boldsymbol{x}} p(\boldsymbol{x}|\cdot))$. Direct inference from $p(\boldsymbol{x})$ may be difficult computationally or technically. For instance, in the MAP inference problem, it could be the case that the gradient or subgradient of $p(\boldsymbol{x})$ may not exist and it is computationally prohibitive to search $\boldsymbol{x}$'s whole feasible space.

Probabilistic graphical models as structured graphs provide a framework for modeling the dependency between random variables. Belief propagation (BP) is a general message-passing algorithm for performing inference on graphical models. The intuition of BP is exchange of belief (statistical information) between neighboring nodes [7]. When belief exchange converges, inference can be done by using the converged belief in graphical models. BP can solve inference problems exactly when the graphical model representation of $p(\boldsymbol{x})$ is loop-free or tree-structured [8]. When there are loops or circles in graphical models, BP is still a practical method to do inference approximately (loopy BP) by running it as if there is no loop. But its performance could be deteriorated significantly. In the loopy case, there are attempts to study convergence properties of BP in special cases [9], [10], but (loopy) BP may not converge in general.

Apart from the practical performance issues of BP in loopy graphs, the understanding of it is also limited. [11] shows that BP in loopy graphs approaches to a stationary point of an approximate free energy, the Bethe free energy in statistical physics. Based on this understanding, variants of BP are derived to improve BP. For instance, fractional BP in [12] applies a correction coefficient to each factor, and generalized BP [11] propagates belief between different regions of a graph. Another track is expectation propagation (EP), introduced by Opper and Winther [13] and Minka [14], [15]. In EP, a simpler factorized distribution defined in exponential distribution family is used to approximate the original complex distribution, and an intuitive factor-wise refinement procedure is used to find such an approximate distribution. The method has an intuition of minimizing a localized Kullback-Leibler (KL) divergence. This is discussed further in [16] and it shows an unifying view of message passing algorithms. Following work stochastic EP [17] explores its variant method for applications to large dataset.

In this work, we take the path of Minka's variational methods to improve BP and also to gain better understanding of BP in loopy graphs. We define a surrogate distribution $q(\boldsymbol{x})$ first. $q(\boldsymbol{x})$ is assumed to be fully factorized and each factor of $q(\boldsymbol{x})$ represents a message in the factor graph representation of the original distribution $p(\boldsymbol{x})$. Fully factorization is the only requirement to $q(\boldsymbol{x})$. Then we define a message passing rule that is derived by minimizing a localized α-divergence. This is factor-wise refinement of $q(\boldsymbol{x})$ iteratively. We refer to the obtained algorithm by α-BP. The merits of α are as follows:

a. α-BP has clear intuition as localized minimization of α-divergence between original distribution p and surrogate distribution q.

b. α-BP generalizes the standard BP, since the message rule of BP is a special case of α-BP.

c. α-BP could outperform BP significantly even in full-connected graphs while still maintaining simplicity of BP for inference.

II. Preliminary

In this section, we provide the preliminaries that are needed in this paper. We introduce the α-divergence and a graphical model that we are going to use to explain α-BP.
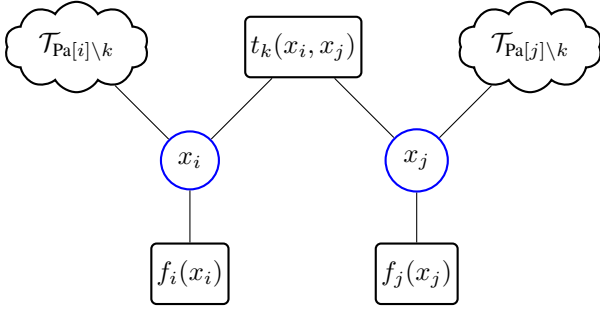
Figure 1. Factor graph illustration of Equation 5.

A. Divergence Measures

As explained in Section I, we are going to minimize α-divergence between p and q, which is defined as follows according to [18] [16]:

$$\mathcal{D}_\alpha(p\|q) = \frac{\int_{\boldsymbol{x}} \alpha p(\boldsymbol{x}) + (1-\alpha)q(\boldsymbol{x}) - p(\boldsymbol{x})^\alpha q(\boldsymbol{x})^{1-\alpha} d\boldsymbol{x}}{\alpha(1-\alpha)}, \tag{1}$$

where α is the parameter of α-divergence, distribution p and q are unnormalized, i.e. $\int_{\boldsymbol{x}} p(\boldsymbol{x}) d\boldsymbol{x} \neq 1$, $\int_{\boldsymbol{x}} q(\boldsymbol{x}) d\boldsymbol{x} \neq 1$.

The classic KL divergence is defined as

$$KL(p\|q) = \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{q(\boldsymbol{x})} d\boldsymbol{x} + \int q(\boldsymbol{x}) - p(\boldsymbol{x}) d\boldsymbol{x} \tag{2}$$

where the $\int q(\boldsymbol{x}) - p(\boldsymbol{x}) d\boldsymbol{x}$ is a correction factor to accommodate unnormalized p and q. The KL divergence is a special case of α-divergence, since $\lim_{\alpha \to 1} \mathcal{D}_\alpha(p\|q) = KL(p\|q)$ and $\lim_{\alpha \to 0} \mathcal{D}_\alpha(p\|q) = KL(q\|p)$, by applying L'Hôpital's rule to Equation 1.

Both α-divergence and KL divergence are equal to zero if $p = q$, and they are non-negative (therefore satisfy the basic property of error measure). Denote KL-projection by

$$\text{proj}[p] = \underset{q \in \mathcal{F}}{\arg\min} \, KL(p\|q), \tag{3}$$

where $\mathcal{F}$ is the distribution family of q.

According to the stationary point equivalence Theorem in [16], proj$[p^\alpha q^{1-\alpha}]$ and $\mathcal{D}_\alpha(p\|q)$ have same stationary points. A heuristic scheme to find q minimizing $\mathcal{D}_\alpha(p\|q)$ is to find its stationary point by a fixed-point iteration:

$$q(\boldsymbol{x})^{\text{new}} = \text{proj}[p(\boldsymbol{x})^\alpha q(\boldsymbol{x})^{1-\alpha}]. \tag{4}$$

B. A Graphic Model

We introduce a pairwise Markov random field (MRF) $p(\boldsymbol{x})$ to explain our algorithm. Variable $\boldsymbol{x} \in \mathcal{A}^N$, where $\mathcal{A}$ is a discrete finite set or subset of $\mathbb{R}$ and N is a positive integer. We factorize the distribution $p(\boldsymbol{x})$ as

$$p(\boldsymbol{x}) \propto \prod_{i=1}^N f_i(x_i) \prod_{k \in \mathcal{K}} t_k(x_i, x_j), \tag{5}$$

where f_i is the *singleton factor*, t_k is *pairwise factor*, $\mathcal{K}$ is the index set of all pairwise factors, and $\propto$ denotes the fact that the only difference between two sides of $\propto$ is a constant factor.

The factor graph representing Equation 5 is shown in Figure 1. In the figure, Pa$[i]$ is the index set of pairwise factors connecting to variable node x_i, i.e. Pa$[i]$ is subset of $\mathcal{K}$, $\backslash$ denotes exclusion. $\mathcal{T}_{\text{Pa}[i]\backslash k}$ is the product of all pairwise factors connecting to x_i except for t_k:

$$\mathcal{T}_{\text{Pa}[i]\backslash k} = \prod_{n \in \text{Pa}[i]\backslash k} t_n. \tag{6}$$

III. α-BP AS FULLY-FACTORIZED APPROXIMATION

In this section, we will show why α-BP as a message-passing algorithm can be used as a fully-factorized approximation to the original distribution $p(\boldsymbol{x})$.

A. α Belief Propagation

1) Fully Factorized Surrogate: Now we formulate a surrogate distribution as

$$q(\boldsymbol{x}) \propto \prod_{i=1}^N \tilde{f}_i(x_i) \prod_{k \in \mathcal{K}} \tilde{t}_k(x_i, x_j), \boldsymbol{x} \in \mathcal{A}^N \tag{7}$$

to approximate $p(\boldsymbol{x})$. We further assume that $q(\boldsymbol{x})$ can be fully factorized, which means that $\tilde{t}_k(x_i, x_j)$ can be factorized as two independent functions of x_i, x_j respectively. We denote this factorization as

$$\tilde{t}_k(x_i, x_j) = m_{k \to i}(x_i) m_{k \to j}(x_j). \tag{8}$$

We use the notation $m_{k \to i}(x_i)$ to denote the factor as a function of x_i due to the intuitive fact that $m_{k \to i}$ is also the message from the factor $t_k(x_i, x_j)$ to variable node x_i. Similarly we have factor $m_{k \to j}(x_j)$. Then the marginal can be formulated straightforwardly as

$$q_i(x_i) \propto \tilde{f}_i(x_i) \prod_{k \in \text{Pa}[i]} m_{k \to i}(x_i). \tag{9}$$

2) Local α-Divergence Minimization: Now, we are going to use the heuristic scheme as in Equation 4 to minimize the information loss by using tractable $q(\boldsymbol{x})$ to represent $p(\boldsymbol{x})$. The information loss is measured by α-divergence $\mathcal{D}_\alpha(p(\boldsymbol{x})\|q(\boldsymbol{x}))$.

We do factor-wise refinement to update the factors of $q(\boldsymbol{x})$ such that $q(\boldsymbol{x})$ approaches $p(\boldsymbol{x})$ asymptotically similar to [14], [16]. Without losing generality, we begin to refine factor $\tilde{t}_k(x_i, x_j)$. Define $q^{\backslash k}(\boldsymbol{x})$ as all other factors except for $\tilde{t}_k(x_i, x_j)$

$$q^{\backslash k}(\boldsymbol{x}) = q(\boldsymbol{x})/\tilde{t}_k(x_i, x_j) \propto \prod_i \tilde{f}_i(x_i) \prod_{n \in \mathcal{K}\backslash k} \tilde{t}_n(x_i, x_j). \tag{10}$$

Similarly, we have $p^{\backslash k}(\boldsymbol{x})$ as all other factors except for $t_k(x_i, x_j)$. Assume that we already have had $q^{\backslash k}(\boldsymbol{x})$ as a good approximation of $p^{\backslash k}(\boldsymbol{x})$, i.e. $q^{\backslash k}(\boldsymbol{x}) \simeq p^{\backslash k}(\boldsymbol{x})$, it is $\tilde{t}_k(x_i, x_j)$ that remains to be refined. Then the problem $\underset{\tilde{t}_k^{\text{new}}}{\arg\min} \, \mathcal{D}_\alpha\left(p^{\backslash k} t_k \| q^{\backslash k} \tilde{t}_k^{\text{new}}\right)$ becomes

$$\underset{\tilde{t}_k^{\text{new}}(x_i, x_j)}{\arg\min} \, \mathcal{D}_\alpha\left(q^{\backslash k}(\boldsymbol{x}) t_k(x_i, x_j) \| q^{\backslash k}(\boldsymbol{x}) \tilde{t}_k^{\text{new}}(x_i, x_j)\right). \tag{11}$$

Using Equation 4, the above problem is equivalent to

$$
\begin{aligned}
& q^{\backslash k}(\boldsymbol{x})\tilde{t}_k^{\text{new}}(x_i, x_j) \\
& \propto \text{proj}\left[\left(q^{\backslash k}(\boldsymbol{x})t_k(x_i, x_j)\right)^\alpha \left(q^{\backslash k}(\boldsymbol{x})\tilde{t}_k(x_i, x_j)\right)^{1-\alpha}\right] \\
& \propto \text{proj}\left[q^{\backslash k}(\boldsymbol{x})t_k(x_i, x_j)^\alpha \tilde{t}_k(x_i, x_j)^{1-\alpha}\right].
\end{aligned}
\tag{12}
$$

Let us refine one message per time in factor $\tilde{t}_k$. Without lose of generality, we update $m_{k\to i}$ and denote

$$
\tilde{t}_k^{\text{new}}(x_i, x_j) = m_{k\to i}^{\text{new}}(x_i)m_{k\to j}(x_j).
\tag{13}
$$

Since KL-projection to a fully factorized distribution reduces to matching the marginals, Equation 12 is reduced to

$$
\sum_{\boldsymbol{x}\backslash x_i} q^{\backslash k}(\boldsymbol{x})\tilde{t}_k^{\text{new}}(x_i, x_j) \propto \sum_{\boldsymbol{x}\backslash x_i} q^{\backslash k}(\boldsymbol{x})t_k(x_i, x_j)^\alpha \tilde{t}_k(x_i, x_j)^{1-\alpha}.
\tag{14}
$$

We use summation here. But it should be replaced by integral if $\mathcal{A}$ is a continuous set. Solving Equation 14 gives the message passing rule as

$$
\begin{aligned}
m_{k\to i}^{\text{new}}(x_i) \propto & \left[\sum_{x_j} t_k(x_i, x_j)^\alpha m_{k\to j}(x_j)^{1-\alpha}m_{j\to k}(x_j)\right] \\
& \cdot m_{k\to i}(x_i)^{1-\alpha},
\end{aligned}
\tag{15}
$$

where

$$
m_{j\to k}(x_j) = \tilde{f}_j(x_j)\prod_{n\in \text{Pa}[j]\backslash k} m_{n\to j}(x_j).
\tag{16}
$$

Similarly, the message from t_k to x_j, $m_{k\to j}(x_j)$, can be updated in similar way.

As for the singleton factor $\tilde{f}_i(x_i)$, we can do the refinement procedure on $\tilde{f}_i(x_i)$ in the same way as we have done for $\tilde{t}_k(x_i, x_j)$. This gives us the update rule of $\tilde{f}_i(x_i)$ as

$$
\tilde{f}_i^{\text{new}}(x_i) \propto f_i(x_i)^\alpha \tilde{f}_i(x_i)^{1-\alpha},
\tag{17}
$$

which is the belief from factor $f_i(x_i)$ to variable x_i. Note, if we initialize $\tilde{f}_i(x_i) = f_i(x_i)$, then it remains the same in all iterations.
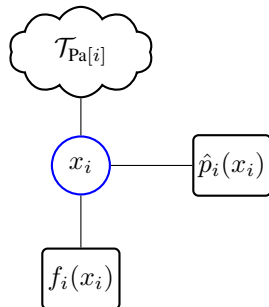
B. Remarks on α-BP

Figure 2. Factor graph illustration with prior factor.

As discussed in Section II, $KL(p\|q)$ is the special case of $\mathcal{D}_\alpha(p\|q)$ when $\alpha \to 1$. When applying $\alpha = 1$ to Equation 15, it gives

$$
m_{k\to i}^{\text{new}}(x_i) \propto \sum_{x_j} t_k(x_i, x_j)m_{j\to k}(x_j),
\tag{18}
$$

which is exactly the messages of BP in Chapter 8 of [7]. From this point of view, α-BP generalizes BP.

Inspired by [19] and assembling methods [20], we can add an extra singleton factor to each x_i as prior information that is obtained from other (usually weak) methods. This factor stands for our belief from exterior estimation. Then run our α-BP. Denote the prior by $\hat{p}_i(x_i)$ for variable node x_i, then the factor graph including this prior belief can be represented as in Figure 2.

We summarize the method into the pseudo-code in Algorithm 1. Though we explain the method with a binary MRF, it is straightforward to replace the factor t_k by a factor involving more than two variables and applies α-BP to general factor graphs.

Algorithm 1 Algorithm of α-BP

Input: Factor graph of $p(\boldsymbol{x})$
1: Initialize $q(\boldsymbol{x})$
2: **if** Prior belief on x_i available **then**
3:     Add prior factor as Figure 2
4: **end if**
5: **while** not converge **do**
6:     **for** each edge of factor graph **do**
7:         Message update by Equation 15 or Equation 17
8:     **end for**
9: **end while**
10: **return** $q(\boldsymbol{x})$

IV. EXPERIMENTAL RESULTS

In this section, we report numerical results on the α-BP. It is well known that performance of BP and its variants deteriorate significantly when loops appear in factor graph. We would like to see if α-BP could relief the deterioration brought by loops in inference. Thus we firstly test the performance of α-BP for MAP inference in a MRF with $\mathcal{A} = \{-1, 1\}$ (Ising model), where we adjust how loopy its corresponding factor graph is.

In addition, we apply the α-BP to a MIMO detection problem, to explore its performance in comparison with (loopy) BP and minimum mean square error (MMSE). At the end, the prior factor trick is used according to discussion in Subsection III-B. This turns out to be MAP inference problem as well.

For the MAP inference, the most probable estimation by α-BP, $\hat{\boldsymbol{x}} = [\hat{x}_1, \cdots, \hat{x}_N]$, is obtained by

$$
\hat{x}_i = \underset{x_i}{\text{argmax}}\ \tilde{f}_i(x_i)\prod_{k\in \text{Pa}[i]} m_{k\to i}(x_i), x_i \in \mathcal{A}.
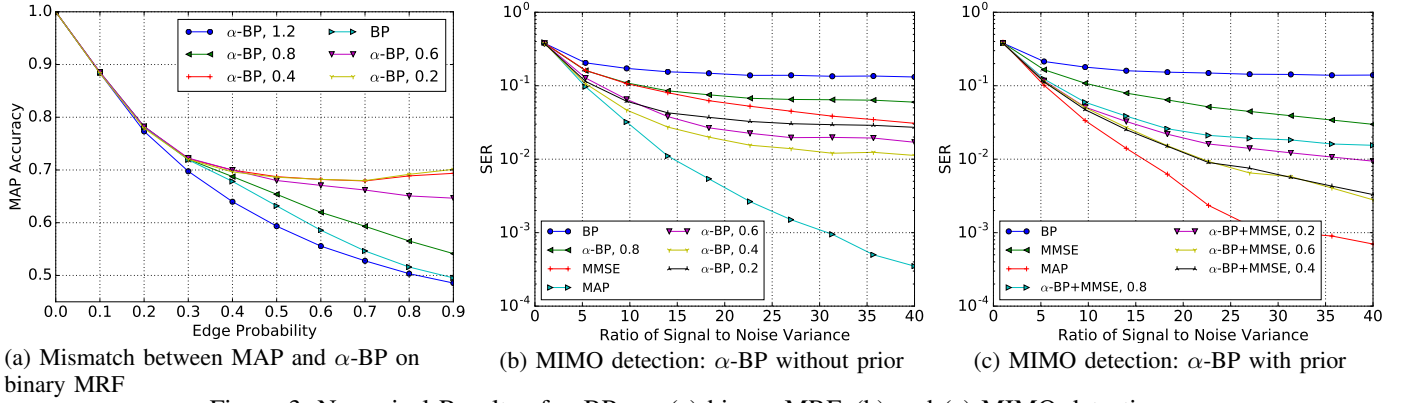\tag{19}
$$

(a) Mismatch between MAP and α-BP on binary MRF

(b) MIMO detection: α-BP without prior

(c) MIMO detection: α-BP with prior

Figure 3. Numerical Results of α-BP on: (a) binary MRF, (b) and (c) MIMO detection.

A. α-BP on binary pairwise MRF

With $t_k(x_i, x_j) = e^{-2J_{i,j}x_i x_j}$ and $f_i(x_i) = e^{-J_{i,i}-b_i x_i}$, Equation 5 can be reformulated as

$$p(\boldsymbol{x}) \propto \exp\{-\boldsymbol{x}^T \boldsymbol{J}\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{x}\}, \boldsymbol{x} \in \mathcal{A}^N, \qquad (20)$$

where $\boldsymbol{x}^T$ is transform of $\boldsymbol{x}$, $J_{i,j}$ is element of symmetrix matrix $\boldsymbol{J}$ at i-th row and j-th column, $\boldsymbol{b} = [b_1, \cdots, b_N]^T$.

For this experiment, we set $\mathcal{A} = \{-1, 1\}$ and $N = 9$. Bias $\boldsymbol{b}$ is sampled from Gaussian, $b_i \sim \mathcal{N}(0, (1/4)^2)$. Since $\boldsymbol{J}$ decides the loopy level of its corresponding factor graph, we use the Erdos-Rnyi model [21] to construct its connectivity. Namely, an element of $\boldsymbol{J}$ is set as non-zero, $J_{i,j} = J_{j,i} \sim \mathcal{N}(0, 1)$, with an *Edge Probability*. Otherwise, $J_{i,j} = J_{j,i} = 0$, which means this is no connection between variable node x_i and x_j. For each test value of Edge Probability, 5000 binary MRF models are generated randomly and mismatch between $\operatorname{argmax}_{\boldsymbol{x}} p(\boldsymbol{x})$ and $\{\boldsymbol{x} | \operatorname{argmax}_{x_i} q_i(x_i)\}$ is computed in each realization. The results are shown in Figure 3a. As the Edge Probability increases, graphs become loopier and α-BP (also BP) has more mismatch with MAP inference. In general, α-BP with $\alpha > 1$ underperforms BP, and α-BP with $\alpha < 1$ outperforms BP. For $\alpha = 0.2, 0.4, 0.6$, α-BP stops deteriorating for Edge Probability increasing over 0.35, while BP continues giving even worse approximations.

B. Application to MIMO Detection

For MIMO system, the observation $\boldsymbol{y}$ is a linear function of channel $\boldsymbol{H} \in \mathbb{R}^{M \times N}$ when unknown signal $\boldsymbol{x}$ need to be estimated:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{e}, \boldsymbol{x} \in \mathcal{A}^N, \qquad (21)$$

where $\boldsymbol{e}$ is noise modeled as Gaussian noise $\boldsymbol{e} \sim \mathcal{N}\left(\boldsymbol{0}, \sigma_w^2 \boldsymbol{I}\right)$. Here $\boldsymbol{I}$ is unitary matrix. In this case, the posterior of $\boldsymbol{x}$ can be written as:

$$p(\boldsymbol{x}|\boldsymbol{y}) \propto \exp\left\{-\frac{1}{2\sigma_w^2}\|\boldsymbol{H}\boldsymbol{x} - \boldsymbol{y}\|^2\right\}$$
$$= \exp\left\{-\frac{1}{2\sigma_w^2}\left[\boldsymbol{x}^T\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{y}^T\boldsymbol{H}\boldsymbol{x} + \boldsymbol{y}^T\boldsymbol{y}\right]\right\} \qquad (22)$$

where $\|\cdot\|$ is the Euclidean norm. Denote $\boldsymbol{S} = \boldsymbol{H}^T\boldsymbol{H}$, $\boldsymbol{h}_i$ as the i-th column of $\boldsymbol{H}$, and apply

$$f_i(x_i) = \exp\left\{-\frac{S_{i,i}x_i^2}{2\sigma_w^2} + \frac{\langle \boldsymbol{h}_i, \boldsymbol{y}\rangle x_i}{\sigma_w^2}\right\}, \qquad (23)$$

$$t_k(x_i, x_j) = \exp\left\{-\frac{x_i S_{i,j} x_j}{\sigma_w^2}\right\}. \qquad (24)$$

Then the Equation 22 is equivalent to Equation 5. We set $\mathcal{A} = \{-1, 1\}$, $N = 8$, and $\boldsymbol{H} \in \mathbb{R}^{8 \times 8}$ sampled from Gaussian.

We test the application of α-BP to the MIMO signal detection numerically. We run the α-BP, without prior trick in Figure 3b and with prior in Figure 3c (legend "α-BP+MMSE") from MMSE. The reference results of MMSE and MAP inference are also reported under the same conditions. MMSE estimator depends on Gaussian posterior $\mathcal{N}(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$ with $\hat{\boldsymbol{\mu}} = (\boldsymbol{H}^T\boldsymbol{H} + \sigma_w^2\boldsymbol{I})^{-1}\boldsymbol{H}^T\boldsymbol{y}$ and $\hat{\boldsymbol{\Sigma}} = (\boldsymbol{H}^T\boldsymbol{H} + \sigma_w^2\boldsymbol{I})^{-1}\sigma_w$. Detection of MMSE is carried out by $\operatorname{argmin}_{x_i \in \mathcal{A}} |x_i - \hat{\mu}_i|$.

Figure 3b shows that BP even underperforms MMSE but α-BP can outperform MMSE by assigning smaller value of α. Note that MMSE requires the matrix inverse computation whose complexity is proportional to N^3, while the complexity of α-BP increases linearly with N. Therefore α-BP is superior to MMSE both performance-wise and complexity-wise. However, there is still a big gap between α-BP (even for $\alpha = 0.4$) and MAP. This gap can be decreased further by using the prior trick discussed in Subsection III-B. Figure 3c exemplifies this effects by using prior belief from MMSE, $\hat{p}_i(x_i) \propto \exp\{-(x_i - \mu_i)^2/(2\boldsymbol{\Sigma}_{i,i})\}$, which comes with legend "α-BP+MMSE". It is shown that larger performance gain is observed when α-BP runs with prior belief.

V. CONCLUSION

In this work, we obtain the α-BP method by doing local α-divergence minimization between model distribution p and surrogate distribution q. α-BP is a practical Bayesian method for message passing. α-BP is a valid and practical algorithm accoinding to our experiments. With prior trick, the performance of α-BP can be further improved. Future works would be to investigate the parallel message passing scheduling of α-BP. It is also interesting to study guideline on choice of α.

REFERENCES

[1] R. Zhang, C. A. Bouman, J. Thibault, and K. D. Sauer, "Gaussian mixture markov random field for image denoising and reconstruction," in *2013 IEEE Global Conference on Signal and Information Processing*, Dec 2013, pp. 1089–1092.

[2] J. Cspedes, P. M. Olmos, M. Snchez-Fernndez, and F. Perez-Cruz, "Expectation propagation detection for high-order high-dimensional mimo systems," *IEEE Transactions on Communications*, vol. 62, no. 8, pp. 2840–2849, Aug 2014.

[3] N. Friel, A. N. Pettitt, R. Reeves, and E. Wit, "Bayesian inference in hidden markov random fields for binary data defined on large lattices," *Journal of Computational and Graphical Statistics*, vol. 18, no. 2, pp. 243–261, 2009. [Online]. Available: http://www.jstor.org/stable/25651244

[4] G. Montufar, "Restricted Boltzmann Machines: Introduction and Review," *ArXiv e-prints*, Jun. 2018.

[5] G. Lin, C. Shen, I. Reid, and A. v. d. Hengel, "Deeply learning the messages in message passing inference," in *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, ser. NIPS'15. Cambridge, MA, USA: MIT Press, 2015, pp. 361–369. [Online]. Available: http://dl.acm.org/citation.cfm?id=2969239.2969280

[6] K. Yoon, R. Liao, Y. Xiong, L. Zhang, E. Fetaya, R. Urtasun, R. S. Zemel, and X. Pitkow, "Inference in probabilistic graphical models by graph neural networks," *CoRR*, vol. abs/1803.07710, 2018. [Online]. Available: http://arxiv.org/abs/1803.07710

[7] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006.

[8] F. R. Kschischang, B. J. Frey, and H. . Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, Feb 2001.

[9] A. T. Ihler, J. W. Fischer III, and A. S. Willsky, "Loopy belief propagation: Convergence and effects of message errors," *J. Mach. Learn. Res.*, vol. 6, pp. 905–936, Dec. 2005. [Online]. Available: http://dl.acm.org/citation.cfm?id=1046920.1088703

[10] J. Du, S. Ma, Y.-C. Wu, S. Kar, and J. M. F. Moura, "Convergence analysis of distributed inference with vector-valued gaussian belief propagation," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6302–6339, Jan. 2017. [Online]. Available: http://dl.acm.org/citation.cfm?id=3122009.3242029

[11] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Generalized belief propagation," in *Proceedings of the 13th International Conference on Neural Information Processing Systems*, ser. NIPS'00. Cambridge, MA, USA: MIT Press, 2000, pp. 668–674. [Online]. Available: http://dl.acm.org/citation.cfm?id=3008751.3008848

[12] W. Wiegerinck and T. Heskes, "Fractional belief propagation," in *Proceedings of the 15th International Conference on Neural Information Processing Systems*, ser. NIPS'02. Cambridge, MA, USA: MIT Press, 2002, pp. 438–445. [Online]. Available: http://dl.acm.org/citation.cfm?id=2968618.2968673

[13] M. Opper and O. Winther, "Gaussian processes for classification: Mean-field algorithms," *Neural Comput.*, vol. 12, no. 11, pp. 2655–2684, Nov. 2000. [Online]. Available: http://dx.doi.org/10.1162/089976600300014881

[14] T. P. Minka, "Expectation propagation for approximate bayesian inference," in *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, ser. UAI '01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 362–369. [Online]. Available: http://dl.acm.org/citation.cfm?id=647235.720257

[15] ——, "A family of algorithms for approximate bayesian inference," Ph.D. dissertation, Cambridge, MA, USA, 2001, aAI0803033.

[16] T. Minka, "Divergence measures and message passing," Tech. Rep. MSR-TR-2005-173, January 2005. [Online]. Available: https://www.microsoft.com/en-us/research/publication/divergence-measures-and-message-passing/

[17] Y. Li, J. M. Hernández-Lobato, and R. E. Turner, "Stochastic expectation propagation," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 2323–2331. [Online]. Available: http://papers.nips.cc/paper/5760-stochastic-expectation-propagation.pdf

[18] H. Zhu and R. Rohwer, "Information geometric measurements of generalisation," Tech. Rep., 1995.

[19] J. Goldberger and A. Leshem, "Pseudo prior belief propagation for densely connected discrete graphs," in *2010 IEEE Information Theory Workshop on Information Theory (ITW 2010, Cairo)*, Jan 2010, pp. 1–5.

[20] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: With Applications in R*. Springer Publishing Company, Incorporated, 2014.

[21] P. Erdos and A. Rényi, "On the evolution of random graphs," *Publ. Math. Inst. Hung. Acad. Sci*, vol. 5, no. 1, pp. 17–60, 1960.