

Inference on Graphs:  
The Iterative Pseudo Likelihood Maximization  
Algorithm

Cédric Herzet

Some material that is available from this technical report is copyrighted.

IEEE Copyright Notice: This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright.

## CONTENTS

<b>I</b>	<b>Notations</b>	4
<b>II</b>	<b>Introduction</b>	4
<b>III</b>	<b>Computation of the LF with the BP algorithm and Bethe free energy approximation</b>	6
<b>IV</b>	<b>Iterative Pseudo LF Maximization</b>	10
IV-A	Pseudo LF: Definition and Properties . . . . .	10
IV-B	IPLFM Algorithm: Definition and Properties . . . . .	14
<b>V</b>	<b>Ensuring the Convergence: a Constrained Version of the IPLFM Algorithm</b>	18
	<b>Appendix I: Factor-graph representation and belief-propagation algorithm</b>	21
	<b>Appendix II: Variational Formulations of the LF as Free Energy Maximization Problems</b>	24
II-A	The LF as the maximum of the Gibbs free energy of the system . . . . .	24
II-B	The LF as the minimum of the Bethe free energy of the system . . . . .	25
	<b>Appendix III: The Expectation-Maximization Algorithm</b>	30
III-A	Classical Formulation of the EM Algorithm . . . . .	30
III-B	The EM Algorithm as a Gibbs Free Energy Maximization Procedure . . . . .	30
III-C	The extended EM algorithm as a Bethe free energy maximization procedure . . . . .	31
III-D	Another formulation of the extended EM algorithm . . . . .	32
	<b>Appendix IV: Local speed of convergence of algorithms based on iterative maximization</b>	34
	<b>References</b>	36

## I. NOTATIONS

The notational conventions adopted in this paper are as follows:

<i>a</i>	italic lowercase indicates a scalar quantity,
<b>a</b>	boldface lowercase indicates a vector quantity,
$a_k$	the $k$ th element of vector <b>a</b> ,
$A$	capital normal indicates random variables,
<b>A</b>	and boldface letters indicates random vectors,
$\mathcal{A}$	the set of indices of the elements of a vector
$\mathbf{a}_A$	vector made up of the elements of <b>a</b> whose index is in $A$ ,
$\mathcal{A}$	the set of values that a random variable or vector can take on,
$\mathcal{A}^{a_k}$	the set of values of <b>A</b> when $A_k = a_k$ ,
$\mathcal{A}_V$	the set of values of $\mathbf{a}_V$ ,
$ \mathcal{A} $	cardinal of $\mathcal{A}$
$p_{\mathbf{A}}(\mathbf{a})$	probability of a random vector <b>A</b> evaluated at <b>a</b>
$\propto$	equality up to a normalization factor.

## II. INTRODUCTION

In this paper, we consider the problem of maximum a posteriori (MAP) estimation of an unknown vector  $\theta$  from the observation of a vector  $\mathbf{y}$ , i.e.,

$$\hat{\theta}_{MAP} = \arg \max_{\theta} \log p_{\Theta|\mathbf{Y}}(\theta|\mathbf{y}), \quad (1)$$

$$= \arg \max_{\theta} \log p_{\Theta, \mathbf{Y}}(\theta, \mathbf{y}). \quad (2)$$

In the sequel, we will refer to the goal function in (2) as the *log-MAP function* (LF).

In a number of practical scenarios, the observation vector  $\mathbf{Y}$  depends on a random vector  $\mathbf{X} = [X_1, X_2, \dots, X_N]$ . The LF may therefore be rewritten as<sup>1</sup>

$$\log p_{\Theta, \mathbf{Y}}(\theta, \mathbf{y}) = \log \sum_{\mathbf{x} \in \mathcal{X}} p_{\Theta, \mathbf{X}, \mathbf{Y}}(\theta, \mathbf{x}, \mathbf{y}). \quad (3)$$

Unfortunately, due to the dependence of the observations on  $\mathbf{X}$ , the MAP estimation problem (2) has, most of the time, no closed-form solutions. In order to circumvent this problem, powerful numerical methods, enabling to iteratively compute the MAP solution (2), have been proposed in the literature.

<sup>1</sup>If  $\mathbf{X}$  takes on values on a continuous domain, the summation sign has to be understood as an integral.

For example, the expectation-maximization (EM) algorithm [1] or the family of gradient methods [2] are instances of such algorithms. More recently, iterative estimation methods based on the belief-propagation (BP) algorithm [3] have appeared in the literature, see e.g. [4], [5], [6]. Although slightly different in their implementation, these methods have the common feature of computing a sequence  $\{\theta^{(n)}\}_{n=0}^{\infty}$  by increasing at each iteration a "pseudo" log-MAP function (PLF); the latter PLF being built by considering standard BP messages as a priori information on the nuisance parameters. In the sequel, we will therefore refer to this kind of algorithm as iterative PLF maximization (IPLFM) algorithm.

In [6], the authors proposed to maximize the PLF by means of the EM algorithm. Considering this particular implementation, they showed that if only one EM iteration is performed, one recovers the standard implementation of the EM algorithm, proving as a by-product that the fixed points of their IPLFM algorithm must be stationary point [1] of the LF when the FG has *no cycles*. This conclusion was later shown to be valid irrespective of the method used to maximize the PLF in two parallel works [7], [8]: in [7] this result was shown in the particular context of synchronization problems whereas general FGs were considered in [8]. In [8], the author also gives a result for *cyclic* FG, although the proposed result does not enable an easy interpretation of the nature of the fixed points.

In this paper, we relate the fixed points of the IPLFM algorithm to the Bethe free energy [9] associated to the considered FG. In particular, we show the following results: *i)* any fixed points of the IPLFM algorithm is a stationary point of the Bethe free energy; *ii)* the fixed points of the IPLFM algorithm must also be fixed point of the (extended<sup>2</sup>) EM algorithm; *iii)* we give necessary and sufficient conditions for local convergence of the IPLFM algorithm. As a corollary of this result, we show that the IPLFM algorithm can never converge to maxima of the Bethe free energy. Finally, we give a way of combining the IPLFM and the EM algorithms to derive fast-convergence generalized EM algorithm.

The remainder of this paper is organized as follows. In section III, we discuss the evaluation of the LF by means of the BP/FG framework and briefly discuss the complexity associated to standard maximization methods (gradient algorithm, EM algorithm). In section IV, we define the *Pseudo LMF* associated to a covering set of regions of a FG and we emphasize some of its properties. Based on these properties, we then emphasize several important properties of the IPLM algorithm. Finally, in section V, we propose a constrained version of the IPLM algorithm, which is ensured to converge, by using results from the EM-algorithm theory.

Note that we have included a lot of material in the appendices. In Appendix I, we give a short

<sup>2</sup>see section III

introduction to the factor-graph representation and the belief-propagation algorithm. In Appendix II, we review the notion of free energy of a system and make the necessary connections between free energy, LF and BP algorithm. In Appendix III, we give some results pertaining to the EM algorithm framework. Finally, in Appendix IV we give the expression of the rate of converge of a certain family of algorithms.

### III. COMPUTATION OF THE LF WITH THE BP ALGORITHM AND BETHE FREE ENERGY APPROXIMATION

As mentionned in II, the LF may be regarded as the marginal of  $p_{\Theta, \mathbf{X}, \mathbf{Y}}(\theta, \mathbf{x}, \mathbf{y})$ . In particular, we can rewrite (3) as

$$\log p_{\Theta, \mathbf{Y}}(\theta, \mathbf{y}) = \log \sum_{x_i \in \mathcal{X}_i} p_{\Theta, X_i, \mathbf{Y}}(\theta, x_i, \mathbf{y}), \quad (4)$$

where

$$p_{\Theta, X_i, \mathbf{Y}}(\theta, x_i, \mathbf{y}) = \sum_{\mathbf{x} \in \mathcal{X}^{x_i}} p_{\Theta, \mathbf{X}, \mathbf{Y}}(\theta, \mathbf{x}, \mathbf{y}). \quad (5)$$

(5) can be efficiently evaluated using the theoretical framework of the factor graphs (FG) and the belief propagation (BP) algorithm (see Appendix I). Indeed, let

$$p_{\Theta, \mathbf{X}, \mathbf{Y}}(\theta, \mathbf{x}, \mathbf{y}) = \prod_{a=1}^M \Psi_{\mathbf{X}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta), \quad (6)$$

be a particular factorization<sup>3</sup> of  $p_{\Theta, \mathbf{X}, \mathbf{Y}}(\theta, \mathbf{x}, \mathbf{y})$ . Then, if the FG representation of (6) is *cycle free*, we have

$$p_{\Theta, X_i, \mathbf{Y}}(\theta, x_i, \mathbf{y}) = \prod_{a \in P_i} \mathbf{m}_{a \rightarrow i}(x_i). \quad (7)$$

Therefore,

$$\log p_{\Theta, \mathbf{Y}}(\theta, \mathbf{y}) = \log \sum_{x_i \in \mathcal{X}_i} \prod_{a \in P_i} \mathbf{m}_{a \rightarrow i}(x_i; \theta). \quad (8)$$

Since the BP algorithm enables to efficiently compute  $\mathbf{m}_{a \rightarrow i}(x_i; \theta)$  by taking benefit from the factorization of  $p_{\Theta, \mathbf{X}, \mathbf{Y}}(\theta, \mathbf{x}, \mathbf{y})$ , the evaluation of the LF via (8) turns out to be often much less complex than the brute-force evaluation of (4). Note that (8) is valid for any node  $i$  in the FG. Moreover, since we only need the messages at *one* arbitrary node in the FG, the complexity of this approach is equivalent to computing the messsages on all the edges of the FG in only *one* direction. In other words, we see that the complexity associated to (8) is half the complexity associated to the computation of all the marginals

<sup>3</sup>For the sake of conciseness, we have drop the possible dependence of  $\Psi_{\mathbf{X}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta)$  on  $\mathbf{Y}$ .

that of the function that the FG represents (which requires to compute the messages on the edges of the FG in *both* directions).

Note that (8) is valid to evaluate the LF as long as the messages  $\mathbf{m}_{a \rightarrow i}(x_i; \theta)$  are computed on a *cycle-free* FG representation of  $p_{\Theta, \mathbf{X}, \mathbf{Y}}(\theta, \mathbf{x}, \mathbf{y})$ . Unfortunately, in many cases a "low-complexity"<sup>4</sup> cycle-free representation of  $p_{\Theta, \mathbf{X}, \mathbf{Y}}(\theta, \mathbf{x}, \mathbf{y})$  does not exist. In such cases, the implementation of iterative optimization techniques (gradient algorithm, EM algorithm,...) turns out to be complex since the complexity of one iteration of these algorithms is usually of the same order as the evaluation of the LF. For example, the gradient of the LF can be computed as follows [10],

$$\nabla_{\Theta} \log p_{\Theta | \mathbf{Y}}(\theta | \mathbf{y}) = \sum_{a=1}^M \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}} p_{\mathbf{x}_{V_a} | \mathbf{Y}, \Theta}(\mathbf{x}_{V_a} | \mathbf{y}, \theta) \nabla_{\Theta} \log \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta), \quad (9)$$

whereas the function evaluated by the EM algorithm at each iteration (see Appendix III) writes

$$\mathcal{Q}_{\Theta | \Theta'}(\theta | \theta') = \sum_{a=1}^M \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}} p_{\mathbf{x}_{V_a} | \mathbf{Y}, \Theta}(\mathbf{x}_{V_a} | \mathbf{y}, \theta') \log \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta), \quad (10)$$

We see that the evaluation of (9) and (10) both requires the computation of marginals  $p_{\mathbf{x}_{V_a} | \mathbf{Y}, \Theta}(\mathbf{x}_{V_a} | \mathbf{y}, \theta)$ . As a consequence, the complexity of the gradient algorithm or the EM algorithm is roughly equal to twice the complexity associated to the evaluation of the LF via (8). In such cases, if we want to reduce the complexity of the problem there is no other solutions than resorting to approximations.

An approach that we will consider in the rest of this paper is to approximate the LF by the minimum of the Bethe free energy of the system. Indeed, we show in Appendix II that if the FG associated to (6) is cycle free, then (minus) the LF can be seen as the minimum (with respect to some variables  $b_a(\mathbf{x}_{V_a})$  and  $b_i(x_i)$ ) of the Bethe free energy. In the cycle-free case, denoting by  $b_a^*(\mathbf{x}_{V_a})$  and  $b_i^*(x_i)$  the values minimizing the Bethe free energy, we have therefore

$$\log p_{\Theta, \mathbf{Y}}(\theta, \mathbf{y}) = -G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)). \quad (11)$$

If the FG contains cycle, we can consider the minimum of the Bethe free energy as an approximation of  $\log p_{\Theta, \mathbf{Y}}(\theta, \mathbf{y})$ , i.e.

$$\log p_{\Theta, \mathbf{Y}}(\theta, \mathbf{y}) \simeq -G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)). \quad (12)$$

In the rest of this paper will therefore consider the following maximization problem

$$\theta^* = \arg \max_{\theta} L_{\Theta}(\theta), \quad (13)$$

<sup>4</sup>i.e. a representation such that the cardinality of  $\mathcal{X}_i$ 's is small.

where

$$L_{\Theta}(\theta) \triangleq -G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)). \quad (14)$$

With a slight abuse of language, we will refer to  $L_{\Theta}(\theta)$  as the LF in the sequel. The reader should however keep in mind that  $L_{\Theta}(\theta)$  is only an approximation of the LF when the FG contains cycles. To conclude this section, it is interesting to generalize (8) to the more general problem of evaluating  $L_{\Theta}(\theta)$ :

**Result (Relation between the Bethe free energy and  $\gamma_{\Theta}(\theta)$ ):** Let  $b_a^*(\mathbf{x}_{V_a})$  and  $b_i^*(x_i)$  denote the beliefs maximizing the Bethe free energy associated to (6) (see Appendix II). Then, for any  $\theta$ ,

$$L_{\Theta}(\theta) = K_{FG} \log \gamma_{\Theta}(\theta), \quad (15)$$

where

$$K_{FG} \triangleq M - N - \sum_{i=1}^N d_i \begin{cases} = 1 & \text{if the FG contains no cycle,} \\ = 0 & \text{if the FG contains one cycle,} \\ < 0 & \text{if the FG contains more than one cycle.} \end{cases} \quad (16)$$

**Proof:** Let us start from (91). Plugging the expression of  $b_a^*(\mathbf{x}_{V_a})$  and  $b_i^*(x_i)$  defined in (100)-(101) into (90), we get<sup>5</sup>

$$\begin{aligned} G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)) &= \sum_{a=1}^M \log \gamma_a - \sum_{i=1}^N (d_i - 1) \log \gamma_i \\ &- \sum_{a=1}^M \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}} b_a(\mathbf{x}_{V_a}) \sum_{i \in V_a} \log \mathbf{m}_{i \rightarrow a}(x_i) + \sum_{i=1}^N (d_i - 1) \sum_{x_i \in \mathcal{X}_i} b_i(x_i) \log \mathbf{m}_{a \rightarrow i}(x_i). \end{aligned} \quad (17)$$

Using (104), it can then be shown that the last two terms in in (17) cancel out. Moreover using (76), we finally have

$$G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)) = (M - N - \sum_{i=1}^N d_i) \log \gamma_{\Theta}(\theta). \quad (18)$$

It is easy to show that  $M - N - \sum_{i=1}^N d_i$  is equal to 1 if the FG is cycle free, 0 if it contains one cycle and negative otherwise.  $\square$

We see that if the FG is cycle free, we recover (8) since,

$$\log p_{\Theta, \mathbf{Y}}(\theta, \mathbf{y}) = -G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)) \quad (19)$$

$$= \log \gamma_{\Theta}(\theta). \quad (20)$$

<sup>5</sup>We use the shorthand notation:  $\gamma_{\Theta}^a(\theta) = \gamma_a$  and  $\gamma_{\Theta}^i(\theta) = \gamma_i$ .

However, when the FG contains more than one cycle, we have to take minus  $\log \gamma_{\Theta}(\theta)$  to evaluate  $L_{\Theta}(\theta)$ .

#### IV. ITERATIVE PSEUDO LF MAXIMIZATION

In this section, we give a definition of the PLF and the IPLFM algorithm and discuss some their important properties.

##### A. Pseudo LF: Definition and Properties

**Definition:** A region  $\mathcal{R}$  of a FG is defined by a set of factor nodes and the set of *all* variables which are connected to them.

**Definition:** A covering set  $\Omega$  is a set of regions such that all factor nodes in the FG are included in one and only one region of the set.

**Definition:** A variable node  $i$  is said to be a *boundary node* if there exists some  $a$  such that  $a \notin \mathcal{R}$  and  $a \in P_i$ .

**Notations:** In the sequel, we will use the following set of notations:

$V_{\mathcal{R}}$	set of (the indices of the) variable nodes belonging to region $\mathcal{R}$ ,
$V_{\mathcal{R}}^B$	set of (the indices of the) boundary variable nodes belonging to region $\mathcal{R}$ ,
$P_{\mathcal{R}}$	set of (the indices of the) factor nodes belonging to region $\mathcal{R}$ ,
$\mathbf{m}_{a \rightarrow i}(x_i, \theta)$	BP message transmitted from factor node $a$ to variable node $i$ if $\Theta = \theta$ in all the factor nodes.

**Example:** We show in Fig. 1 two possible covering sets of regions in a FG.

**Definition:** Let  $\Omega$  be a covering set of *cycle-free* regions. The pseudo LF (PLF) associated with a covering set of regions  $\Omega$  is defined as<sup>6</sup>

$$G_{\Theta, \Theta'}^{\Omega}(\theta, \theta') \triangleq \sum_{\mathcal{R} \in \Omega} \log \sum_{\mathbf{x}_{\mathcal{R}}} \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta'), \quad (21)$$

where

$$\Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \triangleq \prod_{a \in P_{\mathcal{R}}} \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta), \quad (22)$$

$$\Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \triangleq \prod_{i \in V_{\mathcal{R}}^B} \prod_{a \in P_i \setminus P_{\mathcal{R}}} \mathbf{m}_{ai}(x_i, \theta'). \quad (23)$$

i.e.  $\Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta)$  is equal to the product of the factors belonging to  $\mathcal{R}$  and  $\Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta')$  is equal to the product of the messages entering the boundary variable nodes of  $\mathcal{R}$ .

<sup>6</sup>The definition of the PLF is related to the "Hybrid-EM" update rule defined in [8] when none of the factor nodes depending on  $\Theta$  follow the standard "E-log" rules.

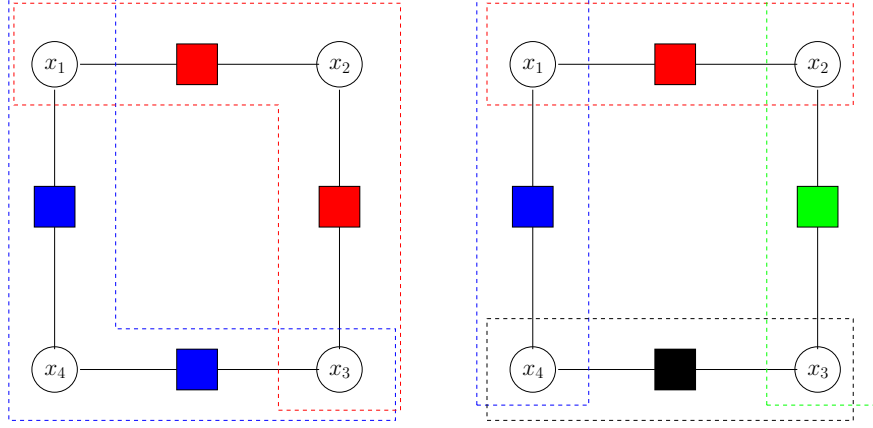


Fig. 1. Two possible covering sets of regions of the FG. In the left-hand-side figure, only  $x_1$  and  $x_3$  are boundary variable nodes whereas in the right-hand-side figures all the variable nodes are boundary nodes.

The appellation ”pseudo” LF is motivated by the following result:

**Property 1:** We have

$$L_{\Theta}(\theta) = \frac{K_{FG}}{|\Omega|} G_{\Theta, \Theta'}^{\Omega}(\theta, \theta), \quad (24)$$

**Proof:** This result is direct consequence of the fact that

$$\log \sum_{\mathbf{x}_{\mathcal{R}}} \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta) = \log \gamma_{\Theta}(\theta). \quad (25)$$

if  $\mathcal{R}$  is cycle free. (This can be shown by showing that

$$\sum_{\mathbf{x}_{\mathcal{R}}} \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta) = \sum_{x_i} \prod_{a \in V_i} m_{a_i}(x_i; \theta) \quad \text{for any } i \in V_{\mathcal{R}}). \quad (26)$$

Therefore, using (15) we get (24).  $\square$

From property 1, we see that the PLF  $G_{\Theta, \Theta'}^{\Omega}(\theta, \theta)$  is equal to  $L_{\Theta}(\theta)$  up to a factor  $\frac{K_{FG}}{|\Omega|}$  when  $\theta = \theta'$ . In fact  $L_{\Theta}(\theta)$  and  $G_{\Theta, \Theta'}^{\Omega}(\theta, \theta')$  have exactly the same mathematical structure; the only difference is that  $L_{\Theta}(\theta)$  allows both  $\Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}$  and  $\Phi_{\mathbf{x}_{\mathcal{R}}, \Theta}$  to vary with  $\theta$  whereas  $G_{\Theta, \Theta'}^{\Omega}$  only allows  $\Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}$  to vary with  $\theta$ . This approximation is equivalent to not taking into account the interactions that factors in different regions of the FG could have. Roughly speaking, this intuitive reasoning tells us that the PLF is likely to behave more and more like the LF when the size of the region increases<sup>7</sup>. Note that since (24) is true for

<sup>7</sup>As a particular case, if  $\Omega$  only contains *one* region which covers the whole FG, the PLF and the LF are equal.

any covering set  $\Omega$  of (cycle-free) regions, it is also true<sup>8</sup> for any linear combination of set of regions, i.e.

$$L_{\Theta}(\theta) = \frac{\text{K}_{FG} \sum_i w_i G_{\Theta, \Theta'}^{\Omega_i}(\theta, \theta)}{\sum_i w_i |\Omega_i|}, \quad (27)$$

with  $\sum_i w_i = 1$ . Considering combination of PLF may probably be interesting in practice to build more accurate approximation of the LF without increasing the complexity of the IPLFM algorithm. For the sake of simplicity and without loss of generality, we will however stick to the case of one single covering set in the remainder of the paper.

The next property shows that the LF and the PLF have locally the same first order behavior:

**Property 2:** We have

$$\nabla_{\Theta} L_{\Theta}(\theta) = \nabla_{\Theta} G_{\Theta, \Theta'}^{\Omega}(\theta, \theta), \quad (28)$$

**Proof:** Using the definition of  $G_{\Theta, \Theta'}^{\Omega}(\theta, \theta)$  and taking the derivative with respect to  $\Theta$ , we get

$$\nabla_{\Theta} G_{\Theta, \Theta'}^{\Omega}(\theta, \theta') = \sum_{\mathcal{R} \in \Omega} \nabla_{\Theta} \log \sum_{\mathbf{x}_{\mathcal{R}}} \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta'), \quad (29)$$

$$= \sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} \frac{\nabla_{\Theta} (\Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta'))}{\sum_{\mathbf{x}_{\mathcal{R}}} \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta')}, \quad (30)$$

$$= \sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta), \quad (31)$$

$$= \sum_{\mathcal{R} \in \Omega} \sum_{a \in P_{\mathcal{R}}} \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta), \quad (32)$$

where we have used the fact that  $\nabla_{\Theta} \log f_{\Theta} = \frac{\nabla_{\Theta} f_{\Theta}}{f_{\Theta}}$  in (30) and (31), and

$$b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \triangleq \frac{\Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta')}{\sum_{\mathbf{x}_{\mathcal{R}}} \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta')} \quad (33)$$

Now, since  $\Omega$  is a *covering* set of regions, we have that  $\sum_{\mathcal{R} \in \Omega} \sum_{a \in P_{\mathcal{R}}} = \sum_{a=1}^M$ . Moreover, since the regions are cycle free, we have

$$\sum_{\mathbf{x}_{\mathcal{R}} \in \mathcal{X}_{\mathcal{R}}^{\mathbf{x}_{V_a}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') = b_{\mathbf{x}_{V_a}, \Theta, \Theta'}(\mathbf{x}_{V_a}, \theta, \theta') = b_a^*(\mathbf{x}_{V_a}), \quad (34)$$

where  $b_a^*(\mathbf{x}_{V_a})$  is the belief minimizing the Bethe free energy of the system. Therefore,

$$\nabla_{\Theta} G_{\Theta, \Theta'}^{\Omega}(\theta, \theta) = \sum_{a=1}^M \sum_{\mathbf{x}_{V_a}} b_a^*(\mathbf{x}_{V_a}) \nabla_{\Theta} \log \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta). \quad (35)$$

<sup>8</sup>This will also be the case for the other properties proved in the sequel.

Finally, using (106) we get (28).  $\square$

This second property of the PLF is very interesting since it states that the Bethe free energy and the PLF have locally the same first order behavior (up to a factor -1). As we will see in the next section, this property will turn out to be key in the characterization of the fixed points of the IPLFM algorithm.

**Property 3:** The Hessian matrix of  $L_{\Theta}(\theta)$  can be expressed as

$$\nabla_{\Theta}^2 L_{\Theta}(\theta) = \nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta, \theta) + \nabla_{\Theta, \Theta'} G_{\Theta, \Theta'}^{\Omega}(\theta, \theta), \quad (36)$$

where

$$\begin{aligned} \nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta, \theta') = & \\ & \sum_{\mathcal{R} \in \Omega} \left( \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta}^2 \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \right. \\ & + \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \\ & \left. - \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \right), \end{aligned} \quad (37)$$

$$\begin{aligned} \nabla_{\Theta, \Theta'} G_{\Theta, \Theta'}^{\Omega}(\theta, \theta') = & \\ & \sum_{\mathcal{R} \in \Omega} \left( \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \nabla_{\Theta'} \log \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \right. \\ & \left. - \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta'} \log \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \right). \end{aligned} \quad (38)$$

**Proof:** Starting from (31), we have

$$\begin{aligned} \nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta, \theta') = & \sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} \nabla_{\Theta} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \\ & + \sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta}^2 \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta). \end{aligned} \quad (39)$$

Now, using the fact that  $\nabla_{\Theta} \log f_{\Theta} = \frac{\nabla_{\Theta} f_{\Theta}}{f_{\Theta}}$ , we have

$$\nabla_{\Theta} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') = b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta} \log b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta'), \quad (40)$$

$$\begin{aligned} &= b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \left( \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \right. \\ &\quad \left. - \nabla_{\Theta} \log \sum_{\mathbf{x}'_{\mathcal{R}}} \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}'_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}'_{\mathcal{R}}, \theta') \right), \end{aligned} \quad (41)$$

$$\begin{aligned} &= b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \left( \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \right. \\ &\quad \left. - \sum_{\mathbf{x}'_{\mathcal{R}}} \frac{\Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}'_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}'_{\mathcal{R}}, \theta')}{\sum_{\mathbf{x}''_{\mathcal{R}}} \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}''_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}''_{\mathcal{R}}, \theta')} \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}'_{\mathcal{R}}, \theta) \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}'_{\mathcal{R}}, \theta') \right), \end{aligned} \quad (42)$$

$$\begin{aligned} &= b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta, \theta') \left( \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \right. \\ &\quad \left. - \sum_{\mathbf{x}'_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta, \Theta'}(\mathbf{x}'_{\mathcal{R}}, \theta, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}'_{\mathcal{R}}, \theta) \right). \end{aligned} \quad (43)$$

Plugging (43) into (39), we get (37). Proceeding in the same way and taking into account that  $\nabla_{\Theta} L_{\Theta}(\theta) = \nabla_{\Theta} G_{\Theta, \Theta'}^{\Omega}(\theta, \theta)$  by (28), we can get similar expressions for  $\nabla_{\Theta, \Theta'} G_{\Theta, \Theta'}^{\Omega}(\theta, \theta)$  and  $\nabla_{\Theta}^2 L_{\Theta}(\theta)$  and prove (36).

### B. IPLFM Algorithm: Definition and Properties

The IPLFM algorithm is define by the following recursion<sup>9</sup>

$$\theta^{(n+1)} = \arg \max_{\theta} G_{\Theta, \Theta'}^{\Omega}(\theta, \theta^{(n)}), \quad (44)$$

i.e. at each iteration we compute a new estimate  $\theta^{(n+1)}$  by maximizing the PLF. In the rest of this section, we will show that the properties of the PLF (see section IV-A) has very interesting properties. In this section, we will show that those properties translates to desirable properties concerning the fixed point and the convergence of the IPLFM algorithm.

**Result:** If  $\theta_f$  is a fixed points of (44), then  $\theta_f$  must be a stationnary points of  $G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i))$  i.e.,

$$\nabla_{\Theta} G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta_f, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)) = 0. \quad (45)$$

<sup>9</sup>Although already considered in different scientific papers, the first general definition of IPLFM algorithm was given in [8] in terms local node update rules. In fact, (44) can be understood as a so-called "Hybrid-EM" algorithm when none of the nodes follows the standard "E-log" rule.

**Proof:** If  $\theta_f$  is a fixed point of (44), then we must have

$$\nabla_{\Theta} G_{\Theta, \Theta'}^{\Omega}(\theta_f, \theta_f) = 0. \quad (46)$$

Now, since  $\nabla_{\Theta} G_{\Theta, \Theta'}^{\Omega}(\theta, \theta) = -\nabla_{\Theta} G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i))$  from (28), we also have

$$\nabla_{\Theta} G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta_f, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)) = 0. \quad (47)$$

and  $\theta_f$  is therefore a stationary point of the Bethe free energy.

This property gives a nice interpretation of the fixed point of the IPLFM algorithm in terms of stationary point of the Bethe free energy. It basically states that any fixed point of the IPLFM algorithm must be stationary point of the Bethe free energy. This feature is of course highly desirable since any solution of (13) must also cancel the first derivative of the Bethe free energy. Interestingly, this result generalize the "cycle-free" theorem proved in [7], [8]: when the FGs is cycle-free, the Bethe free energy is equal to  $-\log p_{\mathbf{Y}, \Theta}(\mathbf{y}, \theta)$  and we the fixed points of the IPLFM algorithm are stationnary point of the true LF.

The next property relates the fixed points of the IPLM algorithm to those of the EM algorithm:

**Property 2:** Let  $\Gamma_G$  denotes the set of fixed points of (44) and let  $\Gamma_{EM}$  denote the set of fixed points of the (extended) EM algorithm (see Appendix III). Then, we have

$$\Gamma_G \subseteq \Gamma_{EM}. \quad (48)$$

**Proof:** We must show that if  $\theta_f$  is a fixed point of (44) then it is also a fixed point of the extended EM algorithm. Now, any  $\theta_f$  which satisfies the following two sufficient conditions

$$\sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta_f) \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta_f) = 0, \quad (49)$$

$$\sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta_f) \nabla_{\Theta}^2 \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta_f) \preceq 0. \quad (50)$$

is a fixed point of the extended EM algorithm. From our previous result, we know that the first condition is fullfilled for any fixed point of (44). Let us show that any fixed point of (44) also satisfies the second one. If  $\theta_f$  is a fixed point of (44), then

$$\nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta, \theta) \preceq 0. \quad (51)$$

Now using (37), we have

$$\nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta, \theta) = \sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta_f) \nabla_{\Theta}^2 \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta_f) + \mathbf{D}, \quad (52)$$

where  $\mathbf{D}$  is a definite positive matrix. As a consequence we have

$$\nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta, \theta) \succeq \sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta_f) \nabla_{\Theta}^2 \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta_f), \quad (53)$$

and

$$\nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta, \theta) \preceq 0 \quad \Rightarrow \quad \sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta_f) \nabla_{\Theta}^2 \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta_f) \preceq 0. \quad (54)$$

□

**Property 3:** The IPLM algorithm never *locally* converges to minima of  $L_{\Theta}(\theta)$ . Moreover, it *locally* converges to a maximum of  $L_{\Theta}(\theta)$ , say  $\theta_m$ , if and only if:

$$\nabla_{\Theta, \Theta'} G_{\Theta, \Theta'}(\theta_m, \theta_m) \succ \nabla_{\Theta}^2 G_{\Theta, \Theta'}(\theta_m, \theta_m). \quad (55)$$

**Proof:** Let  $\theta_f$  be a fixed point of (44) and let us consider the following condition of local convergence:

$$-\mathbf{I} \prec \mathbf{R}_G(\theta_f) \prec \mathbf{I}, \quad (56)$$

where  $\mathbf{I}$  is the unitary matrix and (see Appendix IV)

$$\mathbf{R}_G(\theta_f) = (-\nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta_f, \theta_f))^{-1} \nabla_{\Theta, \Theta'} G_{\Theta, \Theta'}^{\Omega}(\theta_f, \theta_f), \quad (57)$$

is the (local) rate of convergence of (44) around  $\theta_f$ . We will show that (56) is never satisfied for minima whereas it is satisfied for maxima if and only if (55) is satisfied.

Then, the algorithm converges if and only if (56) is satisfied. Using (57) and taking into account that  $\nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta_f, \theta_f) \preceq 0$  for any fixed point, condition (56) may also be rewritten as

$$\nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta_f, \theta_f) \prec \nabla_{\Theta, \Theta'} G_{\Theta, \Theta'}^{\Omega}(\theta_f, \theta_f) \prec -\nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta_f, \theta_f). \quad (58)$$

Adding  $\nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta_f, \theta_f)$  and taking (36) into account, we have the following equivalent condition of convergence:

$$2\nabla_{\Theta}^2 G_{\Theta, \Theta'}^{\Omega}(\theta_f, \theta_f) \prec \nabla_{\Theta}^2 L_{\Theta}(\theta_f) \prec 0. \quad (59)$$

Based on this expression we can draw the two following conclusions. First, if  $\theta_f$  is a minima of  $L_{\Theta}(\theta_f)$ , then  $\theta_f$  is not a stable fixed point of (44). Indeed, if  $\theta_f$  corresponds to a minimum, it implies

$$\nabla_{\Theta}^2 L_{\Theta}(\theta_f) \succ 0. \quad (60)$$

Therefore, the second inequality in (58) is violated and the algorithm does not converge to  $\theta_f$ . On the other hand, if  $\theta_f$  corresponds to a maximum of  $L_\Theta(\theta_f)$ , the second inequality in (58) is always satisfied and the (local) convergence to  $\theta_f$  is therefore ensured if and only if

$$2\nabla_\Theta^2 G_{\Theta,\Theta'}(\theta_f, \theta_f) \prec \nabla_\Theta^2 L_\Theta(\theta_f), \quad (61)$$

which is equivalent to

$$\nabla_{\Theta,\Theta'} G_{\Theta,\Theta'}^\Omega(\theta_f, \theta_f) \succ \nabla_\Theta^2 G_{\Theta,\Theta'}^\Omega(\theta_f, \theta_f), \quad (62)$$

by using (36).  $\square$

In property 1, we saw that some fixed points of the IPLM algorithm can possibly correspond to maxima of the the Bethe free energy. From property 3, we see that even if a maximum of the Bethe free energy is a fixed point of (44), the algorithm will not converge to it. Moreover, property 3 provides necessary and sufficient conditions (55) for convergence to the minima of the Bethe free energy. At the best of our knowledge, it is usually not possible to prove that the IPFLM algorithm will *always* converge to the minima of the Bethe free energy (even it does in a lot of examples we have tested). Based on properties 1,2 and 3, we can therefore draw the Venn diagram in Fig. 2 of the dependence between the fixed points of the EM and IPLFM algorithm and the stationary points of  $L_\Theta(\theta)$ .

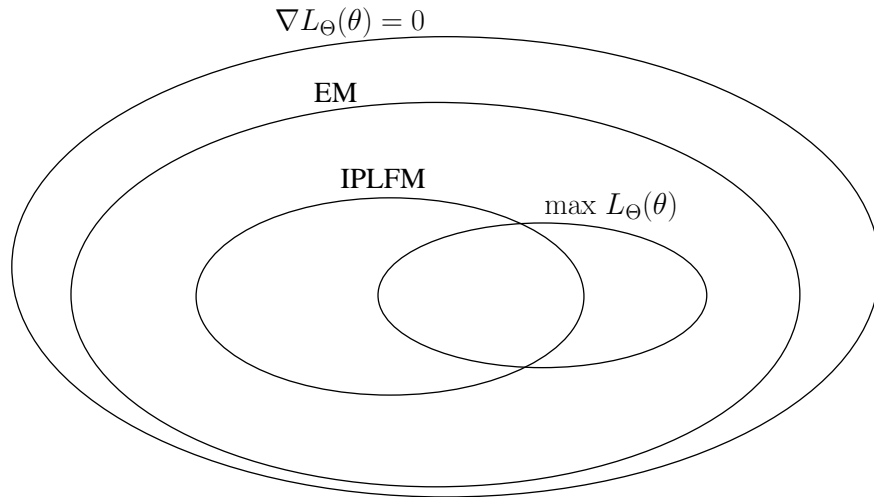


Fig. 2. Venn diagram of the dependence between the fixed points of the EM and IPLFM algorithms and the stationary points and maxima of  $L_\Theta(\theta)$ .

## V. ENSURING THE CONVERGENCE: A CONSTRAINED VERSION OF THE IPLFM ALGORITHM

Although the fixed points of (44) have been shown to stationary points of  $G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i))$ , the convergence to these fixed points may not be ensured (unlike gradient-based or EM algorithms). In some situations, ensuring the algorithm convergence may however be of major importance. From the Global Convergence Theorem [11], it follows that the convergence of (44) may be guaranteed by adding the following additional constraint:

$$L_{\Theta}(\theta^{(n+1)}) > L_{\Theta}(\theta^{(n)}) \quad \text{if } \theta^{(n)} \notin \Gamma_G \quad (63)$$

i.e. by ensuring a strict increase of  $L_{\Theta}(\theta)$  as long as  $\theta^{(n)}$  is not a fixed point. In this section, we propose a modification of (44) which ensures (63) to be satisfied at each iteration. The idea is to combine a result from the EM-algorithm theory with the proposed iterative procedure. Indeed, it is a well-known result (see e.g. [12]) that

$$\mathcal{Q}_{\Theta, \Theta'}(\theta, \theta^{(n)}) > \mathcal{Q}_{\Theta, \Theta'}(\theta^{(n)}, \theta^{(n)}) \Rightarrow L_{\Theta}(\theta) > L_{\Theta}(\theta^{(n)}).$$

Therefore, defining

$$\mathcal{T}^{(n)} = \{\theta \mid \mathcal{Q}_{\Theta, \Theta'}(\theta, \theta^{(n)}) > \mathcal{Q}_{\Theta, \Theta'}(\theta^{(n)}, \theta^{(n)})\} \quad (64)$$

we have that the following procedure is ensured to converge:

$$\theta^{(n+1)} = \arg \max_{\theta \in \mathcal{T}^{(n)}} G_{\Theta, \Theta'}^{\Omega}(\theta, \theta^{(n)}). \quad (65)$$

In the sequel, we will refer to the algorithm defined in (65) as the constrained IPLFM (CIPLFM) algorithm. In fact, the CIPLM algorithm can be understood as a particular GEM algorithm (see Appendix III) since at each iteration, it satisfies  $\mathcal{Q}_{\Theta, \Theta'}(\theta^{(n+1)}, \theta^{(n)}) > \mathcal{Q}_{\Theta, \Theta'}(\theta^{(n)}, \theta^{(n)})$ . As we will see in the next example, the algorithm (65) can however exhibit a much faster speed of convergence than the standard EM algorithm.

**Example:** Let's consider the following model

$$\mathbf{Y} = \mathbf{X} e^{j\theta} + \mathbf{W}, \quad (66)$$

where  $\mathbf{W}$  is a zero-mean white Gaussian noise. This problem corresponds to the estimation of the carrier phase offset in a digital communication system.

Let

$$\theta_{EM}^{(n+1)} = \arg \max_{\theta} \mathcal{Q}_{\Theta, \Theta'}(\theta, \theta_{EM}^{(n)}), \quad (67)$$

and

$$\Delta_{EM} = \theta_{EM}^{(n+1)} - \theta_{EM}^{(n)}. \quad (68)$$

It is easy to show that for the particular case of (66),  $\mathcal{Q}_{\Theta, \Theta'}(\theta, \theta_{EM}^{(n)})$  is symmetric around  $\theta_{EM}^{(n+1)}$  and therefore

$$\mathcal{T}^{(n)} = [\theta_{EM}^{(n)}, \theta_{EM}^{(n)} + 2\Delta_{EM}]. \quad (69)$$

In Fig. 3, we have illustrated the mean speed of convergence of the EM algorithm and (44), (65), i.e., it represents  $E_{\mathbf{Y}}\|\hat{\theta}^{(n)} - \hat{\theta}_{MAP}\|$  versus the number of iterations. We have considered the following setup. We assume that the variable  $X_k$  is a quaternary Phase-Shift Keying (QPSK) symbol, i.e.  $X_k \in \{\pm 1 \pm j\}$  resulting from the convolutional encoding of uniformly-distributed information bits (This simply characterizes the a priori distribution  $p_{\mathbf{X}}(\mathbf{x})$ ).

The (mean) speed of convergence of the EM, IPLFM and CIPLFM algorithms is represented in Fig. (3). In this example, we see that the IPLFM algorithm enables to greatly increase the speed of convergence of the algorithm: one needs 10 EM iterations to achieve an accuracy of  $10^{-4}$  whereas only 3 IPLFM iterations are sufficient. Note also the good performance of the CIPLFM algorithm. Its performance is very close to the one of the IPLFM algorithm. In fact, during the first iteration, one can notice that its speed of convergence is twice the speed of convergence of the EM algorithm. In this region, the IPLM algorithm is slightly faster than the CIPLFM. After a few iterations, when  $\theta^{(n)}$  is in a neighborhood of  $\theta^*$ , the speed of convergence of the IPLFM and CIPLFM algorithms are similar: the two curves are then parallel.

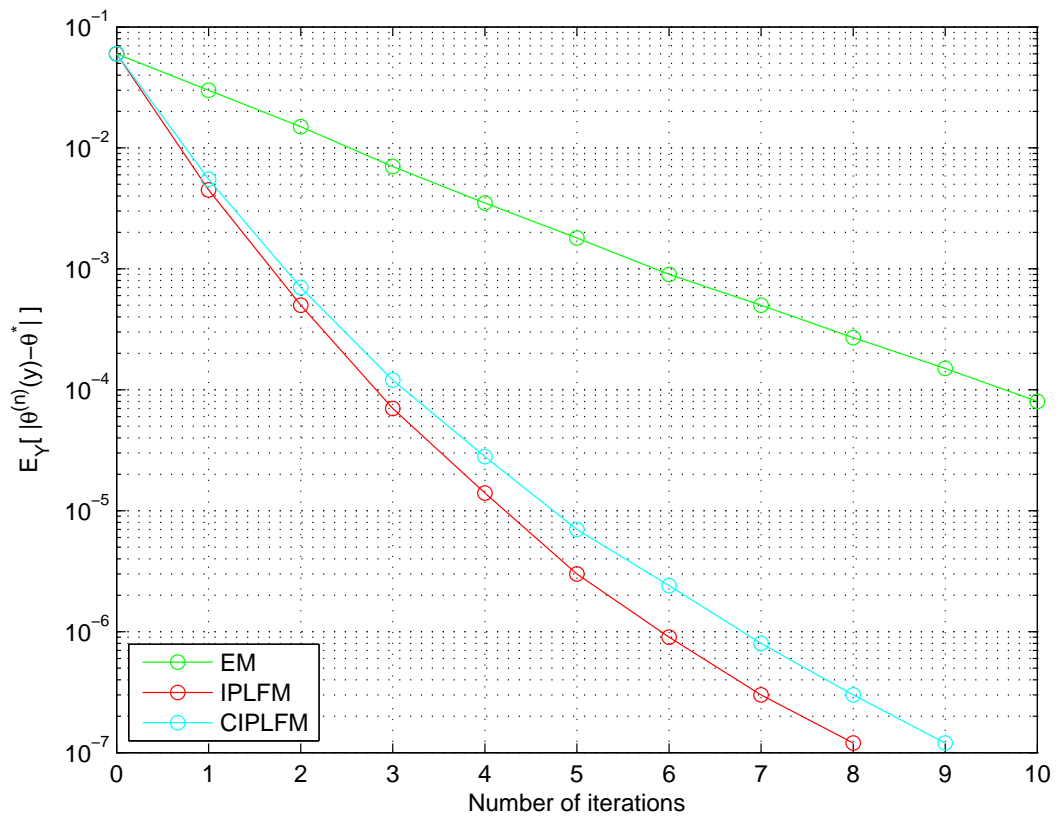


Fig. 3. (Mean) speed of convergence of the EM, IPLFM and CIPLFM algorithms in for carrier phase estimation.

## APPENDIX I

## FACTOR-GRAPH REPRESENTATION AND BELIEF-PROPAGATION ALGORITHM

Let  $f_{\mathbf{X}}(\mathbf{x})$  be a function of  $\mathbf{X} = [X_1, X_2, \dots, X_N]$ . Assume  $f_{\mathbf{X}}(\mathbf{x})$  factorizes as

$$f_{\mathbf{X}}(\mathbf{x}) = \prod_{a=1}^M \Psi_{\mathbf{X}_{V_a}}(\mathbf{x}_{V_a}), \quad (70)$$

where  $V_a \subset \{1, 2, \dots, N\}$ . The factor graph (FG) associated to (70) is a graphical representation of the factorization of  $f_{\mathbf{X}}(\mathbf{x})$  as  $\prod_{a=1}^M \Psi_{\mathbf{X}_{V_a}}(\mathbf{x}_{V_a})$  and is defined as follows. The FG contains one *factor node* for each factor  $\Psi_{\mathbf{X}_{V_a}}(\mathbf{x}_{V_a})$  (we have therefore  $M$  factor nodes in the FG) and one *variable node* for each element of  $\mathbf{x}$  (hence, there are  $N$  variable nodes in the FG). We draw an *edge* between a factor node  $\Psi_{\mathbf{X}_{V_a}}(\mathbf{x}_{V_a})$  and a variable node  $x_i$  if and only if  $i \in V_a$ . We give an example of factor graph in Fig. 4.

The belief propagation (BP) algorithm is an algorithm which applies on FG's and whose primary purpose is the evaluation of the marginals of the function that the FG represents (i.e.  $\sum_{\mathbf{x} \in \mathcal{X}^{x_i}} f_{\mathbf{X}}(\mathbf{x})$ ,  $\forall x_i$ ). The BP algorithm works as follows. For each edge in the graph, it computes two vectors of values, also called *messages*. Let  $\mathbf{m}_{a \rightarrow i}$  and  $\mathbf{m}_{i \rightarrow a}$  denote the two vectors of messages computed by the BP algorithm on the edge connecting  $\Psi_{\mathbf{X}_{V_a}}(\mathbf{x}_{V_a})$  and  $x_i$ . Each vector contains exactly  $|\mathcal{X}_i|$  elements and we will refer to the elements of  $\mathbf{m}_{a \rightarrow i}$  (resp.  $\mathbf{m}_{i \rightarrow a}$ ) as  $\mathbf{m}_{a \rightarrow i}(x_i)$  (resp.  $\mathbf{m}_{i \rightarrow a}(x_i)$ ) for  $x_i \in \mathcal{X}_i$ . The BP algorithm computes these elements as follows:

$$\mathbf{m}_{i \rightarrow a}(x_i) = \prod_{a' \in P_i \setminus \{a\}} \mathbf{m}_{a' \rightarrow i}(x_i), \quad (71)$$

$$\mathbf{m}_{a \rightarrow i}(x_i) = \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}^{x_i}} \Psi_{\mathbf{X}_{V_a}}(\mathbf{x}_{V_a}) \prod_{j \in V_a \setminus \{i\}} \mathbf{m}_{j \rightarrow a}(x_j), \quad (72)$$

where  $P_i \subset \{1, 2, \dots, M\}$  is such that  $a \in P_i \Leftrightarrow i \in V_a$  (i.e.  $P_i$  defines the set of factor nodes to which variable node  $x_i$  is connected in the FG). Equations (71) and (72) define the so-called message-update rules of the BP algorithm. We see from these update rules that the messages on a particular edge only depend on the messages on the *adjacent* edges.

We can consider two different cases:

- 1) **The FG of  $f_{\mathbf{X}}(\mathbf{x})$  is cycle-free:** In this case, it has been shown in [3] that the marginals of  $f_{\mathbf{X}}(\mathbf{x})$  can be computed as follows:

$$\sum_{\mathbf{x} \in \mathcal{X}^{x_i}} f_{\mathbf{X}}(\mathbf{x}) = \prod_{a \in P_i} \mathbf{m}_{a \rightarrow i}(x_i) \quad (73)$$

- 2) **The FG of  $f(\mathbf{x})$  contains cycles:** In this case,  $\prod_{a \in P_i} \mathbf{m}_{a \rightarrow i}(x_i)$  is only an approximation of the exact marginals  $\sum_{\mathbf{x} \in \mathcal{X}^{x_i}} f_{\mathbf{X}}(\mathbf{x})$ . Moreover, applying update rules (71)-(72) on the FG leads to an iterative algorithm. A nice characterization of the fixed points of the BP algorithm has recently been proposed by Yedidia [9] in terms of stationary points of the Bethe free energy.

We now prove the following result that we will use in the paper:

**Result:** Let

$$\gamma_{\Theta}^i(\theta) \triangleq \sum_{x_i \in \mathcal{X}_i} \prod_{a \in P_i} \mathbf{m}_{a \rightarrow i}(x_i), \quad (74)$$

$$\gamma_{\Theta}^a(\theta) \triangleq \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}} \Psi_{\mathbf{X}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta) \prod_{i \in V_a} \mathbf{m}_{i \rightarrow a}(x_i), \quad (75)$$

then

$$\gamma_{\Theta}^a(\theta) = \gamma_{\Theta}^i(\theta) \quad \text{for } 1 \leq a \leq M \text{ and } 1 \leq i \leq N. \quad (76)$$

**Note:**  $\gamma_{\Theta}^a(\theta)$  is actually equal to the sum of all the messages entering factor node  $a$  times the factor itself.

**Proof:** Using the BP message propagation rule (71), we can rewrite  $\gamma_{\Theta}^i$  as

$$\gamma_{\Theta}^i = \sum_{x_i \in \mathcal{X}_i} \mathbf{m}_{a' \rightarrow i}(x_i) \prod_{a' \in P_i \setminus \{a\}} \mathbf{m}_{a' \rightarrow i}(x_i), \quad (77)$$

$$= \sum_{x_i \in \mathcal{X}_i} \mathbf{m}_{a \rightarrow i}(x_i) \mathbf{m}_{i \rightarrow a}(x_i), \quad (78)$$

Similarly, using (72) we can rewrite (75) as

$$\gamma_{\Theta}^a = \sum_{x_i \in \mathcal{X}_i} \mathbf{m}_{i \rightarrow a}(x_i) \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}^{x_i}} \Psi_{\mathbf{X}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta) \prod_{j \in V_a \setminus \{i\}} \mathbf{m}_{j \rightarrow a}(x_j), \quad (79)$$

$$= \sum_{x_i \in \mathcal{X}_i} \mathbf{m}_{a \rightarrow i}(x_i) \mathbf{m}_{i \rightarrow a}(x_i), \quad (80)$$

We see therefore from (78)-(80) that for a given  $i$ ,  $\gamma_{\Theta}^i(\theta) = \gamma_{\Theta}^a(\theta) \forall a \in P_i$ . Since this result is valid for any  $i$ , we can conclude (76).

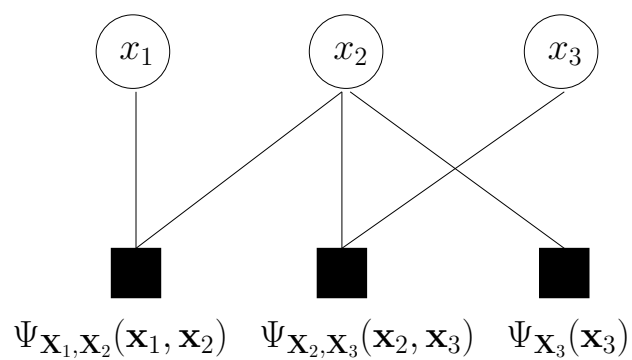


Fig. 4. FG representation of  $f_{\mathbf{X}}(\mathbf{x}) = \Psi_{\mathbf{X}_1, \mathbf{X}_2}(\mathbf{x}_1, \mathbf{x}_2) \Psi_{\mathbf{X}_2, \mathbf{X}_3}(\mathbf{x}_2, \mathbf{x}_3) \Psi_{\mathbf{X}_3}(\mathbf{x}_3)$ .

## APPENDIX II

## VARIATIONAL FORMULATIONS OF THE LF AS FREE ENERGY MAXIMIZATION PROBLEMS

The term "variational" usually refers to techniques which express a quantity or a function as the solution of an optimization problem. In this section, we will show that the LF can be expressed as the solution of different constrained optimization problems. The objective function of these optimization problems is usually referred to as "free energy" by reference to the statistical physics literature where they first appeared.

## A. The LF as the maximum of the Gibbs free energy of the system

**Definition:** the Gibbs free energy associated with probability  $p_{\mathbf{Y},\mathbf{X},\theta}(\mathbf{y}, \mathbf{x}, \theta)$  is defined as follows:

$$F_{\Theta,B(\mathbf{x})}(\theta, b(\mathbf{x})) = - \sum_{\mathbf{x} \in \mathcal{X}} b(\mathbf{x}) \log p_{\mathbf{Y},\mathbf{X},\theta}(\mathbf{y}, \mathbf{x}, \theta) + \sum_{\mathbf{x} \in \mathcal{X}} b(\mathbf{x}) \log b(\mathbf{x}), \quad (81)$$

where  $b(\mathbf{x})$  is a trial probability of  $\mathbf{X}$ .

**Result:** for any  $\theta$ , we have

$$\log p_{\Theta,\mathbf{Y}}(\theta, \mathbf{y}) = \min_{b(\mathbf{x})} F_{\Theta,B(\mathbf{x})}(\theta, b(\mathbf{x})) \quad (82)$$

$$\text{subject to } \sum_{\mathbf{x} \in \mathcal{X}} b(\mathbf{x}) - 1 = 0, \quad b(\mathbf{x}) \geq 0 \quad \forall \mathbf{x} \in \mathcal{X},$$

$$= - F_{\Theta,B(\mathbf{x})}(\theta, p_{\mathbf{X}|\mathbf{Y},\theta}(\mathbf{x}|\mathbf{y}, \theta)). \quad (83)$$

In other words, for any  $\theta$ , the minimum of the Gibbs free energy with respect to  $b(\mathbf{x})$  is the LF.

**Proof:** In order to show (82), it is convenient to rewrite  $\log p_{\Theta,\mathbf{Y}}(\theta, \mathbf{y})$  as

$$\log p_{\Theta,\mathbf{Y}}(\theta, \mathbf{y}) = \log p_{\Theta,\mathbf{Y},\mathbf{X}}(\theta, \mathbf{y}, \mathbf{x}) - \log p_{\mathbf{X}|\mathbf{Y},\theta}(\mathbf{x}|\mathbf{y}, \theta), \quad (84)$$

$$= \sum_{\mathbf{x}} b(\mathbf{x}) \log p_{\Theta,\mathbf{Y},\mathbf{X}}(\theta, \mathbf{y}, \mathbf{x}) - \sum_{\mathbf{x}} b(\mathbf{x}) \log p_{\mathbf{X}|\mathbf{Y},\theta}(\mathbf{x}|\mathbf{y}, \theta), \quad (85)$$

where (85) follows from the fact that the left-hand side does not depend on  $\mathbf{x}$ . Using this expression, it is then easy to rewrite the Gibbs free energy as follows

$$F(\theta, b(\mathbf{x})) = - \log p_{\mathbf{Y},\theta}(\mathbf{y}, \theta) + \sum_{\mathbf{x} \in \mathcal{X}} b(\mathbf{x}) \log \frac{b(\mathbf{x})}{p_{\mathbf{X}|\mathbf{Y},\theta}(\mathbf{x}|\mathbf{y}, \theta)}. \quad (86)$$

Since, from the constraints in (82),  $b(\mathbf{x})$  is a probability mass function, the second term in (86) is nothing else but the Kullback-Leibler distance between  $b(\mathbf{x})$  and  $p_{\mathbf{X}|\mathbf{Y},\Theta}(\mathbf{x}|\mathbf{y},\theta)$ . Therefore, we have

$$F(\theta, b(\mathbf{x})) = -\log p_{\mathbf{Y},\Theta}(\mathbf{y},\theta) + D_{KL}(b(\mathbf{x}), p_{\mathbf{X}|\mathbf{Y},\Theta}(\mathbf{x}|\mathbf{y},\theta)), \quad (87)$$

$$\geq -\log p_{\mathbf{Y},\Theta}(\mathbf{y},\theta), \quad (88)$$

where the last line follows from the non-negativity of the Kullback-Leibler distance. Since  $D_{KL}(f, g) = 0$  iff  $f = g$ , we get (82).  $\square$

### B. The LF as the minimum of the Bethe free energy of the system

In the previous section, we showed that the LF can be rewritten as the solution of an optimization problem where the goal function is the Gibbs free energy of the system. In this section, we will show that in some cases, the LF can also be rewritten as the solution of an optimization problem involving a different goal function: the *Bethe free energy of the system*.

**Definition:** The *Bethe free energy* associated to a particular factorization of  $p_{\Theta, \mathbf{X}, \mathbf{Y}}(\theta, \mathbf{x}, \mathbf{y})$ , say

$$p_{\Theta, \mathbf{X}, \mathbf{Y}}(\theta, \mathbf{x}, \mathbf{y}) = \prod_{a=1}^M \Psi_{\mathbf{X}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta), \quad (89)$$

where  $\{\mathbf{X}_{V_a}\}_{a=1}^M$  denotes  $M$  subsets of elements of  $\mathbf{X}$ , is given by

$$\begin{aligned} G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a(\mathbf{x}_{V_a}), b_i(x_i)) &= - \sum_{a=1}^M \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}} b_a(\mathbf{x}_{V_a}) \log \Psi_{\mathbf{X}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta) \\ &\quad + \sum_{a=1}^M \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}} b_a(\mathbf{x}_{V_a}) \log b_a(\mathbf{x}_{V_a}) \\ &\quad - (d_i - 1) \sum_{i=1}^N \sum_{x_i \in \mathcal{X}_i} b_i(x_i) \log b_i(x_i). \end{aligned} \quad (90)$$

$d_i$  is the number of occurrences of  $X_i$  in the  $\mathbf{X}_{V_a}$ 's and  $N$  is the number of elements of  $\mathbf{X}$ .  $b_a(\mathbf{x}_{V_a})$  (resp.  $b_i(x_i)$ ) is a trial probability mass function of  $\mathbf{X}_{V_a}$  (resp.  $X_i$ ).

**Result:** The LF can be characterized as

$$\log p_{\Theta, \mathbf{Y}}(\theta, \mathbf{y}) = \min_{(b_a(\mathbf{x}_{V_a}), b_i(x_i))} G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a(\mathbf{x}_{V_a}), b_i(x_i)) \quad (91)$$

subject to

- 1)  $\sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}} b_a(\mathbf{x}_{V_a}) - 1 = 0$ , for  $1 \leq a \leq M$ ,
- 2)  $b_a(\mathbf{x}_{V_a}) \geq 0$ , for  $1 \leq a \leq M$ ,
- 3)  $\sum_{x_i \in \mathcal{X}_i} b_i(x_i) - 1 = 0$ , for  $1 \leq i \leq N$ ,
- 4)  $b_i(x_i) \geq 0$  for  $1 \leq i \leq N$ ,
- 5)  $\sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}^{x_i}} b_a(\mathbf{x}_{V_a}) - b_i(x_i) = 0$  for  $1 \leq a \leq M$ ,  $\forall i \in V_a$ ,  $\forall x_i \in \mathcal{X}_i$

if and only if the FG representation of (6) is cycle free.

**Proof:** By the *Hammersley-Clifford theorem* [13], we have that, if the FG representation of (6) is cycle free then we must have

$$p_{\mathbf{X}|\mathbf{Y}, \Theta}(\mathbf{x}|\mathbf{y}, \theta) = \frac{\prod_{a=1}^M p_{\mathbf{x}_{V_a}|\mathbf{Y}, \Theta}(\mathbf{x}_{V_a}|\mathbf{y}, \theta)}{\prod_{i=1}^N p_{X_i|\mathbf{Y}, \Theta}^{d_i-1}(x_i|\mathbf{y}, \theta)}, \quad (92)$$

and vice-versa.

Now, we showed in section II that the optimal value of  $b(\mathbf{x})$  in the Gibbs-based formulation of the LF (83) is  $b(\mathbf{x}) = p_{\mathbf{X}|\mathbf{Y}, \Theta}(\mathbf{x}|\mathbf{y}, \hat{\theta}_{MAP})$ . Therefore, (92) suggests that we can restrict our attention on beliefs  $b(\mathbf{x})$  having the following mathematical structure:

$$b(\mathbf{x}) = \frac{\prod_{a=1}^M b_a(\mathbf{x}_{V_a})}{\prod_{i=1}^N b_i^{d_i-1}(x_i)}, \quad (93)$$

where  $b_a(\mathbf{x}_{V_a})$  and  $b_i(x_i)$  must be constrained to be probability mass functions, i.e.

$$\sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}} b_a(\mathbf{x}_{V_a}) - 1 = 0, \quad b_a(\mathbf{x}_{V_a}) \geq 0 \quad 1 \leq a \leq M, \quad (94)$$

$$\sum_{x_i \in \mathcal{X}_i} b_i(x_i) - 1 = 0, \quad b_i(x_i) \geq 0 \quad 1 \leq i \leq N, \quad (95)$$

and must be *consistent*<sup>10</sup>, i.e.,

$$\sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}^{x_i}} b_a(\mathbf{x}_{V_a}) - b_i(x_i) = 0, \quad 1 \leq a \leq M, \quad \forall i \in V_a, \quad \forall x_i \in \mathcal{X}_i, \quad (96)$$

<sup>10</sup>Intuitively, (96) imposes that if we take the marginal of  $b_a(\mathbf{x}_{V_a})$  with respect to  $X_i = x_i$ , we must recover  $b_i(x_i)$ .

where  $\mathcal{X}_{V_a}^{x_i}$  denotes the set of possible values of  $\mathcal{X}_{V_a}$  when  $X_i = x_i$ . It is therefore clear that adding the constraint (83) together with (94), (95), (96) to optimization problem (82) leads to an equivalent optimization problem. Another equivalent optimization problem can be found by plugging (93) in (83). Doing so, we obtain the following objective function:

$$\begin{aligned}
F_{\Theta, B(\mathbf{x})} \left( \theta, b(\mathbf{x}) = \frac{\prod_{a=1}^M b_a(\mathbf{x}_{V_a})}{\prod_{i=1}^N b_i^{d_i-1}(x_i)} \right) &= - \sum_{a=1}^M \sum_{\mathbf{x} \in \mathcal{X}} \frac{\prod_{a=1}^M b_a(\mathbf{x}_{V_a})}{\prod_{i=1}^N b_i^{d_i-1}(x_i)} \log \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta) \\
&+ \sum_{a=1}^M \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}} b_a(\mathbf{x}_{V_a}) \log b_a(\mathbf{x}_{V_a}) \\
&- (d_i - 1) \sum_{i=1}^N \sum_{x_i \in \mathcal{X}_i} b_i(x_i) \log b_i(x_i). \tag{97}
\end{aligned}$$

Since,

$$\sum_{\mathbf{x} \in \mathcal{X}^{\mathbf{x}_{V_{a'}}}} \frac{\prod_{a=1}^M b_a(\mathbf{x}_{V_a})}{\prod_{i=1}^N b_i^{d_i-1}(x_i)} = b_{a'}(\mathbf{x}_{V_{a'}}), \tag{98}$$

where  $\mathcal{X}^{\mathbf{x}_{V_{a'}}$  denotes the set of possible values of  $\mathcal{X}$  when  $\mathbf{X}_{V_{a'}} = \mathbf{x}_{V_{a'}}$ , we finally have

$$F_{\Theta, B(\mathbf{x})} \left( \theta, b(\mathbf{x}) = \frac{\prod_{a=1}^M b_a(\mathbf{x}_{V_a})}{\prod_{i=1}^N b_i^{d_i-1}(x_i)} \right) = G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a(\mathbf{x}_{V_a}), b_i(x_i)). \tag{99}$$

□

**Result [9]:** the beliefs  $b_a^*(\mathbf{x}_{V_a})$  and  $b_i^*(x_i)$  maximizing the Bethe free energy (91) may be expressed as

$$b_a^*(\mathbf{x}_{V_a}) = \gamma_a^{-1} \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta) \prod_{i \in V_a} \mathbf{m}_{i \rightarrow a}(x_i), \tag{100}$$

$$b_i^*(x_i) = \gamma_i^{-1} \prod_{a \in P_i} \mathbf{m}_{a \rightarrow i}(x_i). \tag{101}$$

where

$$\gamma_a = \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}} \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta) \prod_{i \in V_a} \mathbf{m}_{i \rightarrow a}(x_i), \tag{102}$$

$$\gamma_i = \sum_{x_i \in \mathcal{X}_i} \prod_{a \in P_i} \mathbf{m}_{a \rightarrow i}(x_i), \tag{103}$$

$$\mathbf{m}_{i \rightarrow a}(x_i) = \prod_{a' \in P_i \setminus \{a\}} \mathbf{m}_{a' \rightarrow i}(x_i), \tag{104}$$

$$\mathbf{m}_{a \rightarrow i}(x_i) = \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}^{x_i}} \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta) \prod_{j \in V_a \setminus \{i\}} \mathbf{m}_{j \rightarrow a}(x_j). \tag{105}$$

**Proof:** see [9].

**Discussion:** (100) to (105) define *necessary* conditions on  $b_a(\mathbf{x}_a)$  and  $b_i(\mathbf{x}_i)$  to maximize the Bethe free

energy. In particular, Yedidia *et al.* showed that any  $b_a(\mathbf{x}_{V_a})$  and  $b_i(x_i)$  satisfying (100) to (105) are stationary points (maximum, minimum or saddle point) of the constrained<sup>11</sup> Bethe free energy, and vice versa. Therefore, (100) to (105) also define *sufficient* conditions of optimality when the Bethe free energy is a convex function of  $b_a(\mathbf{x}_{V_a})$  and  $b_i(x_i)$  (since the Bethe free energy has then only one stationary point). This is for example the case when the FG associated to the Bethe free energy is cycle free (see [14] and references therein for a discussion about convexity of the Bethe free energy).

Let us now have a look at variables  $\gamma_a, \gamma_i, \mathbf{m}_{i \rightarrow a}(x_i), \mathbf{m}_{a \rightarrow i}(x_i)$  defined in (102)-(105). First, notice that  $b_a(\mathbf{x}_{V_a})$  and  $b_i(x_i)$  are *univoquely* defined by these variables via (100)-(101). Moreover, note that equations (104)-(105), relating variables  $\mathbf{m}_{i \rightarrow a}(x_i)$  and  $\mathbf{m}_{a \rightarrow i}(x_i)$ , are strictly equivalent to the BP-algorithm "message-update rules" defined in (71)-(72). This has as an important consequence that the BP algorithm can be used to compute the solution of (91). Indeed, after convergence the messages on the edges of the FG must satisfy (104)-(105). Therefore, the beliefs  $b_a(\mathbf{x}_{V_a})$  and  $b_i(x_i)$  computed according to (104)-(105) and using BP messages are ensured to be stationary points of the Bethe free energy. Hence, as long as long one considers a convex Bethe free energy, we get the solution of (91).

**Result:** The derivative with respect to  $\Theta$  of the minimum of the Bethe free energy can be expressed as

$$\nabla_{\Theta} G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)) = - \sum_{a=1}^M \sum_{\mathbf{x}_{V_a}} b_a^*(\mathbf{x}_{V_a}) \nabla_{\Theta} \log \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta). \quad (106)$$

**Proof:** By the derivation chainrule , we have

$$\begin{aligned} & \nabla_{\Theta} G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)) \\ &= - \sum_{a=1}^M \sum_{\mathbf{x}_{V_a}} b_a^*(\mathbf{x}_{V_a}) \nabla_{\Theta} \log \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta) \\ & \quad - \sum_{a=1}^M \sum_{\mathbf{x}_{V_a}} \left( \nabla_{\Theta} b_a^*(\mathbf{x}_{V_a}) \log \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta) - \nabla_{\Theta} b_a^*(\mathbf{x}_{V_a}) \log b_a^*(\mathbf{x}_{V_a}) - \nabla_{\Theta} b_a^*(\mathbf{x}_{V_a}) \right) \\ & \quad - \sum_{i=1}^N (d_i - 1) \sum_{x_i} \left( \nabla_{\Theta} b_i^*(x_i) \log b_i^*(x_i) - \nabla_{\Theta} b_i^*(x_i) \right). \end{aligned} \quad (107)$$

We will now show that all the terms in (107) but the first one cancel out. Using the definitions (100)-(101)

<sup>11</sup>Notice that the beliefs defined by (100)-(105) necessarily satisfy the constraints in (91).

of  $b_a^*(\mathbf{x}_{V_a})$  and  $b_i^*(x_i)$ , we have

$$\begin{aligned}
& \nabla_{\Theta} G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)) \\
&= - \sum_{a=1}^M \sum_{\mathbf{x}_{V_a}} b_a^*(\mathbf{x}_{V_a}) \nabla_{\Theta} \log \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta) \\
&\quad - \sum_{a=1}^M \left( - \sum_{\mathbf{x}_{V_a}} \nabla_{\Theta} b_a^*(\mathbf{x}_{V_a}) \log \prod_{i \in V_a} \mathbf{m}_{i \rightarrow a}(x_i) + (\log \gamma_a - 1) \sum_{\mathbf{x}_{V_a}} \nabla_{\Theta} b_a^*(\mathbf{x}_{V_a}) \right) \\
&\quad - \sum_{i=1}^N (d_i - 1) \left( \sum_{x_i} \nabla_{\Theta} b_i^*(x_i) \log \prod_{a \in P_i} \mathbf{m}_{a \rightarrow i}(x_i) - (1 - \log \gamma_i) \sum_{x_i} \nabla_{\Theta} b_i^*(x_i) \right). \tag{108}
\end{aligned}$$

Taking into account that  $\sum_{\mathbf{x}_{V_a}} \nabla_{\Theta} b_a^*(\mathbf{x}_{V_a}) = 0$  and  $\sum_{x_i} \nabla_{\Theta} b_i^*(x_i) = 0$ , we have

$$\begin{aligned}
& \nabla_{\Theta} G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)) \\
&= - \sum_{a=1}^M \sum_{\mathbf{x}_{V_a}} b_a^*(\mathbf{x}_{V_a}) \nabla_{\Theta} \log \Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta) + \sum_{a=1}^M \sum_{\mathbf{x}_{V_a}} \nabla_{\Theta} b_a^*(\mathbf{x}_{V_a}) \log \prod_{i \in V_a} \mathbf{m}_{i \rightarrow a}(x_i) \\
&\quad - \sum_{i=1}^N (d_i - 1) \sum_{x_i} \nabla_{\Theta} b_i^*(x_i) \log \prod_{a \in P_i} \mathbf{m}_{a \rightarrow i}(x_i). \tag{109}
\end{aligned}$$

Let us finally show that the second term in (109) is equal to minus the third one. First note that, by definition of  $b_a^*(\mathbf{x}_{V_a})$  and  $b_i^*(x_i)$ , we have  $\sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}^{x_i}} b_a^*(\mathbf{x}_{V_a}) = b_i^*(x_i)$  and therefore

$$\sum_{a=1}^M \sum_{\mathbf{x}_{V_a}} \nabla_{\Theta} b_a^*(\mathbf{x}_{V_a}) \log \prod_{i \in V_a} \mathbf{m}_{i \rightarrow a}(x_i) = \sum_{a=1}^M \sum_{i \in V_a} \sum_{x_i} \nabla_{\Theta} b_i^*(x_i) \log \mathbf{m}_{i \rightarrow a}(x_i). \tag{110}$$

Note that  $\sum_{a=1}^M \sum_{i \in V_a}$  is equal to  $\sum_{i=1}^N \sum_{a \in P_i}$  and is equivalent to counting all the edges in the FG. Moreover, taking into account that  $\mathbf{m}_{i \rightarrow a}(x_i) = \prod_{a' \in P_i \setminus a} \mathbf{m}_{a' \rightarrow i}(x_i)$ , we can see that each message  $\mathbf{m}_{a' \rightarrow i}(x_i)$  is counted exactly  $d_i - 1$  times. Therefore, we get

$$\begin{aligned}
\sum_{a=1}^M \sum_{i \in V_a} \sum_{a' \in P_i \setminus a} \sum_{x_i} \nabla_{\Theta} b_i^*(x_i) \log \mathbf{m}_{a' \rightarrow i}(x_i) &= \sum_{i=1}^N \sum_{a \in P_i} \sum_{a' \in P_i \setminus a} \sum_{x_i} \nabla_{\Theta} b_i^*(x_i) \log \mathbf{m}_{a' \rightarrow i}(x_i) \\
&= \sum_{i=1}^N \sum_{a \in P_i} (d_i - 1) \sum_{x_i} \nabla_{\Theta} b_i^*(x_i) \log \mathbf{m}_{a \rightarrow i}(x_i). \tag{111}
\end{aligned}$$

□

## APPENDIX III

## THE EXPECTATION-MAXIMIZATION ALGORITHM

## A. Classical Formulation of the EM Algorithm

The expectation-maximization (EM) algorithm, first defined by Dempster, Laird and Rubin in 1977 [1], is an powerful iterative method for solving MAP (or ML) problems. This algorithm proceeds in two steps: the expectation step (E-step) and the maximization step (M-step). At iteration  $(n + 1)$  we have

$$\text{E - step : } \mathcal{Q}_{\Theta, \Theta'}(\theta, \hat{\theta}^{(n)}) = \int p_{\mathbf{Z}|\mathbf{Y}, \Theta'}(\mathbf{z}|\mathbf{y}, \hat{\theta}^{(n)}) \log p_{\mathbf{Z}, \mathbf{Y}, \Theta}(\mathbf{z}, \mathbf{y}, \theta) d\mathbf{z} \quad (112)$$

$$\text{M - step : } \hat{\theta}^{(n+1)} = \arg \max_{\theta} \mathcal{Q}_{\Theta, \Theta'}(\theta, \hat{\theta}^{(n)}) \quad (113)$$

where  $\mathbf{z}$  is the so-called *complete data set* and is related to  $\mathbf{y}$  by  $\mathbf{y} = f(\mathbf{z})$ , where  $f(\cdot)$  denotes a many-to-one mapping<sup>12</sup>.

**Note:** The EM algorithm can be seen as a particular case of a more general iterative procedure defined as:

$$\mathcal{Q}_{\Theta, \Theta'}(\theta, \hat{\theta}^{(n)}) = \int p_{\mathbf{Z}|\mathbf{Y}, \Theta'}(\mathbf{z}|\mathbf{y}, \hat{\theta}^{(n)}) \log p_{\mathbf{Z}, \mathbf{Y}, \Theta}(\mathbf{z}, \mathbf{y}, \theta) d\mathbf{z} \quad (114)$$

$$\hat{\theta}^{(n+1)} \text{ such that } \mathcal{Q}_{\Theta, \Theta'}(\hat{\theta}^{(n+1)}, \hat{\theta}^{(n)}) > \mathcal{Q}_{\Theta, \Theta'}(\hat{\theta}^{(n)}, \hat{\theta}^{(n)}), \quad (115)$$

i.e. instead of maximizing the  $\mathcal{Q}$ -function, it is sufficient to find a value  $\hat{\theta}^{(n+1)}$  which increases it. The procedure defined by (114)-(115) is usually referred to *generalized EM (GEM) algorithm*. The GEM has basically the same properties as the EM algorithm: it never decreases the LF and its fixed points must be stationary points of the LF [12].

## B. The EM Algorithm as a Gibbs Free Energy Maximization Procedure

**Result [15]:** if the complete data set is defined as  $\mathbf{z} \triangleq [\mathbf{y}, \mathbf{x}]$ , the EM algorithm is equivalent to the following sequence of conditional maximizations:

$$b^{(n)}(\mathbf{x}) = \arg \max_{b(\mathbf{x})} -F_{\Theta, B(\mathbf{x})}(\hat{\theta}^{(n)}, b(\mathbf{x})) \quad \text{subject to } \sum_{\mathbf{x}} b(\mathbf{x}) = 1, \quad (116)$$

$$\hat{\theta}^{(n+1)} = \arg \max_{\theta} -F_{\Theta, B(\mathbf{x})}(\theta, b^{(n)}(\mathbf{x})), \quad (117)$$

<sup>12</sup>This means that there is only one  $\mathbf{y}$  associated to a given  $\mathbf{z}$  but that there may be several values of  $\mathbf{z}$  associated to the same value of  $\mathbf{y}$ . In practice, the choice of "good" complete data set is left to the user. The choice of the complete data set impacts the speed of convergence and the complexity of the EM algorithm. An example of complete data that we will be considering in the sequel is  $\mathbf{z} \triangleq [\mathbf{y}, \mathbf{x}]$ .

**Proof:** First, maximizing  $-F_{\Theta, B(\mathbf{x})}(\theta, b(\mathbf{x}))$  with respect to  $B(\mathbf{x})$  for  $\Theta = \hat{\theta}^{(n)}$  leads to  $b(\mathbf{x}) = p_{\mathbf{X}|\mathbf{Y}, \Theta}(\mathbf{x}|\mathbf{y}, \hat{\theta}^{(n)})$  (see Appendix II). On the other hand, maximizing  $F_{\Theta, B(\mathbf{x})}(\theta, b(\mathbf{x}))$  with respect to  $\Theta$  for  $B(\mathbf{x}) = b^{(n)}(\mathbf{x})$  leads to

$$\hat{\theta}^{(n+1)} = \arg \max_{\theta} -F_{\Theta, B(\mathbf{x})}(\theta, b^{(n)}(\mathbf{x})), \quad (118)$$

$$= \arg \max_{\theta} \sum_{\mathbf{x} \in \mathcal{X}} b^{(n)}(\mathbf{x}) \log p_{\mathbf{Y}, \mathbf{x}, \Theta}(\mathbf{y}, \mathbf{x}, \theta) - \sum_{\mathbf{x} \in \mathcal{X}} b^{(n)}(\mathbf{x}) \log b^{(n)}(\mathbf{x}), \quad (119)$$

$$= \arg \max_{\theta} \sum_{\mathbf{x} \in \mathcal{X}} b^{(n)}(\mathbf{x}) \log p_{\mathbf{Y}, \mathbf{x}, \Theta}(\mathbf{y}, \mathbf{x}, \theta), \quad (120)$$

where the last equality follows from the fact that the second term in (119) does not depend on  $\theta$ . Therefore, plugging  $b^{(n)}(\mathbf{x}) = p_{\mathbf{X}|\mathbf{Y}, \Theta}(\mathbf{x}|\mathbf{y}, \hat{\theta}^{(n)})$  in (120), we have

$$\hat{\theta}^{(n+1)} = \arg \max_{\theta} \sum_{\mathbf{x} \in \mathcal{X}} p_{\mathbf{X}|\mathbf{Y}, \Theta}(\mathbf{x}|\mathbf{y}, \hat{\theta}^{(n)}) \log p_{\mathbf{Y}, \mathbf{x}, \Theta}(\mathbf{y}, \mathbf{x}, \theta). \quad (121)$$

It is easy to show that we end up to the same recursive equation starting from the standard EM equations (112)-(113) and setting  $\mathbf{z} \triangleq [\mathbf{y}, \mathbf{x}]$ .  $\square$

### C. The extended EM algorithm as a Bethe free energy maximization procedure

**Result :** if the complete data set is defined as  $\mathbf{z} \triangleq [\mathbf{y}, \mathbf{x}]$  and the FG associated to (89) is cycle free, the EM algorithm is equivalent to the following procedure:

$$(b_a^{(n)}(\mathbf{x}_{V_a}), b_i^{(n)}(x_i)) = \arg \max_{(b_a(\mathbf{x}_{V_a}), b_i(x_i))} -G_{\Theta, B_a(\mathbf{x}_a), B_i(x_i)}(\hat{\theta}^{(n)}, b_a(\mathbf{x}_{V_a}), b_i(x_i)) \quad (122)$$

subject to

$$1) \quad \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}} b_a(\mathbf{x}_{V_a}) - 1 = 0, \quad \text{for } 1 \leq a \leq M,$$

$$2) \quad b_a(\mathbf{x}_{V_a}) \geq 0, \quad \text{for } 1 \leq a \leq M,$$

$$3) \quad \sum_{x_i \in \mathcal{X}_i} b_i(x_i) - 1 = 0, \quad \text{for } 1 \leq i \leq N,$$

$$4) \quad b_i(x_i) \geq 0 \quad \text{for } 1 \leq i \leq N,$$

$$5) \quad \sum_{\mathbf{x}_{V_a} \in \mathcal{X}_{V_a}^{x_i}} b_a(\mathbf{x}_{V_a}) - b_i(x_i) = 0 \quad \text{for } 1 \leq a \leq M, \forall i \in V_a, \forall x_i \in \mathcal{X}_i$$

$$\hat{\theta}^{(n+1)} = \arg \max_{\theta} -G_{\Theta, B_a(\mathbf{x}_a), B_i(x_i)}(\theta, b_a^{(n)}(\mathbf{x}_{V_a}), b_i^{(n)}(x_i)), \quad (123)$$

$$= \arg \max_{\theta} \sum_{a=1}^M \sum_{\mathbf{x}_a} b_a^{(n)}(\mathbf{x}_{V_a}) \log \Psi_{\mathbf{X}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta). \quad (124)$$

**Details :** On the one hand, if  $p_{\mathbf{X},\mathbf{Y},\Theta}(\mathbf{x},\mathbf{y},\theta) = \prod_{a=1}^M \Psi_{\mathbf{X}_{V_a},\Theta}(\mathbf{x}_{V_a},\theta)$ , we have from (121) that the  $\mathcal{Q}$ -function appearing in the EM algorithm definition can be rewritten as

$$\begin{aligned} \mathcal{Q}_{\Theta,\Theta'}(\theta,\hat{\theta}^{(n)}) &= \sum_{a=1}^M \sum_{\mathbf{x}} p_{\mathbf{X}|\mathbf{Y},\Theta}(\mathbf{x}|\mathbf{y},\hat{\theta}^{(n)}) \log \Psi_{\mathbf{X}_{V_a},\Theta}(\mathbf{x}_{V_a},\theta), \\ &= \sum_{a=1}^M \sum_{\mathbf{x}_{V_a}} p_{\mathbf{X}_{V_a}|\mathbf{Y},\Theta}(\mathbf{x}_{V_a}|\mathbf{y},\hat{\theta}^{(n)}) \log \Psi_{\mathbf{X}_{V_a},\Theta}(\mathbf{x}_{V_a},\theta). \end{aligned} \quad (125)$$

On the other hand, we showed in Appendix II-B that, if the FG associated to  $p_{\mathbf{X},\mathbf{Y},\Theta}(\mathbf{x},\mathbf{y},\theta) = \prod_{a=1}^M \Psi_{\mathbf{X}_{V_a},\Theta}(\mathbf{x}_{V_a},\theta)$  is cycle-free, the beliefs  $b_a^{(n)}(\mathbf{x}_{V_a})$  maximizing  $-G_{\Theta,B_a(\mathbf{x}_{V_a}),B_i(x_i)}(\theta^{(n)},b_a(\mathbf{x}_{V_a}),b_i(x_i))$  are equal to

$$b_a^{(n)}(\mathbf{x}_{V_a}) = p_{\mathbf{X}_{V_a}|\mathbf{Y},\Theta}(\mathbf{x}_{V_a}|\mathbf{y},\theta^{(n)}). \quad (126)$$

Therefore, plugging (126) in (124) we get (125). This proves the result.  $\square$

**Note:** Although the procedure described in (122)-(124) is equivalent to the EM algorithm only if the FG is cycle free, we can consider (122)-(124) for any kind of FGs, see e.g. [16], [8], [17]. If the FG has cycles, then (122)-(124) can be understood as an iterative algorithm enabling to look for the maximum of<sup>13</sup>  $G_{\Theta,B_a(\mathbf{x}_{V_a}),B_i(x_i)}(\theta,b_a^*(\mathbf{x}_{V_a}),b_i^*(x_i))$ . In this paper, we will therefore refer to the procedure defined in (122)-(124) as *extended EM algorithm*.

#### D. Another formulation of the extended EM algorithm

Let  $\Omega$  be a covering set of regions (see section IV) of the FG associated to  $p_{\mathbf{X},\mathbf{Y},\Theta}(\mathbf{x},\mathbf{y},\theta) = \prod_{a=1}^M \Psi_{\mathbf{X}_{V_a},\Theta}(\mathbf{x}_{V_a},\theta)$ . Then, the following procedure is equivalent to the extended EM algorithm:

$$\theta^{(n+1)} = \arg \max_{\theta} \mathcal{Q}_{\Theta,\Theta'}^{\Omega}(\theta,\theta^{(n)}) \quad (127)$$

where

$$\mathcal{Q}_{\Theta,\Theta'}^{\Omega}(\theta,\theta') = \sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{X}_{\mathcal{R}},\Theta'}(\mathbf{x}_{\mathcal{R}},\theta') \log \Psi_{\mathbf{X}_{\mathcal{R}},\Theta}(\mathbf{x}_{\mathcal{R}},\theta), \quad (128)$$

$$b_{\mathbf{X}_{\mathcal{R}},\Theta'}(\mathbf{x}_{\mathcal{R}},\theta') = \frac{\Psi_{\mathbf{X}_{\mathcal{R}},\Theta'}(\mathbf{x}_{\mathcal{R}},\theta') \Phi_{\mathbf{X}_{\mathcal{R}},\Theta'}(\mathbf{x}_{\mathcal{R}},\theta')}{\sum_{\mathbf{x}_{\mathcal{R}}} \Psi_{\mathbf{X}_{\mathcal{R}},\Theta'}(\mathbf{x}_{\mathcal{R}},\theta') \Phi_{\mathbf{X}_{\mathcal{R}},\Theta'}(\mathbf{x}_{\mathcal{R}},\theta')}, \quad (129)$$

$$\Psi_{\mathbf{X}_{\mathcal{R}},\Theta}(\mathbf{x}_{\mathcal{R}},\theta) \triangleq \prod_{a \in P_{\mathcal{R}}} \Psi_{\mathbf{X}_{V_a},\Theta}(\mathbf{x}_{V_a},\theta), \quad (130)$$

$$\Phi_{\mathbf{X}_{\mathcal{R}},\Theta'}(\mathbf{x}_{\mathcal{R}},\theta') \triangleq \prod_{i \in V_{\mathcal{R}}^B} \prod_{a \in P_i \setminus P_{\mathcal{R}}} \mathbf{m}_{ai}(x_i,\theta'). \quad (131)$$

<sup>13</sup>As showed in Appendix II-B,  $-G_{\Theta,B_a(\mathbf{x}_{V_a}),B_i(x_i)}(\theta,b_a^*(\mathbf{x}_{V_a}),b_i^*(x_i))$  is equal to  $\log p_{\mathbf{Y},\Theta}(\mathbf{y},\theta)$  when the FG is cycle free.

**Proof:** This follows from the fact that in a *cycle-free* region of FG, we have

$$\sum_{\mathbf{x}_{\mathcal{R}} \in \mathcal{X}_{\mathcal{R}}^{\times V_a}} b_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') = \sum_{\mathbf{x}_{\mathcal{R}} \in \mathcal{X}_{\mathcal{R}}^{\times V_a}} \frac{\Psi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta')}{\sum_{\mathbf{x}_{\mathcal{R}}} \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta')} \quad (132)$$

$$= \Psi_{\mathbf{x}_{V_a}, \Theta'}(\mathbf{x}_{V_a}, \theta') \prod_{i \in V_a} m_{ia}(x_i, \theta'), \quad (133)$$

$$= b_a^*(\mathbf{x}_a, \theta'), \quad (134)$$

where  $b_a^*(\mathbf{x}_a, \theta')$  is the belief maximizing the Bethe free energy when the value of  $\Theta$  in all the factor nodes  $\Psi_{\mathbf{x}_{V_a}, \Theta}(\mathbf{x}_{V_a}, \theta)$  is set to  $\theta'$ .  $\square$

**Result:** Based on the definition of  $\mathcal{Q}_{\Theta, \Theta'}^{\Omega}(\theta, \theta')$ , we have

$$\nabla_{\Theta} \mathcal{Q}_{\Theta, \Theta'}^{\Omega}(\theta, \theta') = \sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta), \quad (135)$$

$$\nabla_{\Theta}^2 \mathcal{Q}_{\Theta, \Theta'}^{\Omega}(\theta, \theta') = \sum_{\mathcal{R} \in \Omega} \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \nabla_{\Theta}^2 \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta), \quad (136)$$

$$\begin{aligned} \nabla_{\Theta, \Theta'} \mathcal{Q}_{\Theta, \Theta'}^{\Omega}(\theta, \theta') &= \sum_{\mathcal{R} \in \Omega} \left( \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \nabla_{\Theta'} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \right. \\ &\quad - \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \nabla_{\Theta'} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \Big) \\ &\quad + \sum_{\mathcal{R} \in \Omega} \left( \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \nabla_{\Theta'} \log \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \right. \\ &\quad \left. - \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \nabla_{\Theta} \log \Psi_{\mathbf{x}_{\mathcal{R}}, \Theta}(\mathbf{x}_{\mathcal{R}}, \theta) \sum_{\mathbf{x}_{\mathcal{R}}} b_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \nabla_{\Theta'} \log \Phi_{\mathbf{x}_{\mathcal{R}}, \Theta'}(\mathbf{x}_{\mathcal{R}}, \theta') \right). \end{aligned} \quad (137)$$

**Proof:** We skip the proof. (The derivation of (135)-(137) is very similar to (28), (37) and (38).)

Let us now express some properties of the EM algorithm with this new region-based formulation:

**Property 1:**

$$\nabla_{\Theta} \mathcal{Q}_{\Theta, \Theta'}^{\Omega}(\theta, \theta) = -\nabla_{\Theta} G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)). \quad (138)$$

As a direct consequence, the fixed points of the extended EM algorithm are stationary points of  $G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i))$ .

**Property 2:**

$$-\nabla_{\Theta}^2 G_{\Theta, B_a(\mathbf{x}_{V_a}), B_i(x_i)}(\theta, b_a^*(\mathbf{x}_{V_a}), b_i^*(x_i)) = \nabla_{\Theta}^2 \mathcal{Q}_{\Theta, \Theta'}^{\Omega}(\theta, \theta') + \nabla_{\Theta, \Theta'} \mathcal{Q}_{\Theta, \Theta'}^{\Omega}(\theta, \theta') \quad (139)$$

**Property 3:** If the FG is cycle free, we have

$$\nabla_{\Theta, \Theta'} \mathcal{Q}_{\Theta, \Theta'}^{\Omega}(\theta_s, \theta_s) \succeq 0 \quad (140)$$

**Proof (to do: extend to the cyclic case):** Since  $\mathcal{Q}_{\Theta, \Theta'}^{\Omega}(\theta, \theta')$  and  $\mathcal{Q}_{\Theta, \Theta'}(\theta, \theta')$  in (112) are equivalent, we can simply show that

$$\nabla_{\Theta, \Theta'} \mathcal{Q}_{\Theta, \Theta'}(\theta, \theta') \succeq 0. \quad (141)$$

Using the definition of  $\mathcal{Q}_{\Theta, \Theta'}(\theta, \theta')$  and the Bayes rule, we have

$$\nabla_{\Theta, \Theta'} \mathcal{Q}_{\Theta, \Theta'}(\theta, \theta') = \int \nabla_{\Theta'} p_{\mathbf{Z}|\mathbf{Y}, \Theta'}(\mathbf{z}|\mathbf{y}, \theta') \nabla_{\Theta} \log p_{\mathbf{Z}, \mathbf{Y}, \Theta}(\mathbf{z}, \mathbf{y}, \theta) d\mathbf{z}, \quad (142)$$

$$\begin{aligned} &= \int p_{\mathbf{Z}|\mathbf{Y}, \Theta'}(\mathbf{z}|\mathbf{y}, \theta') \nabla_{\Theta'} \log p_{\mathbf{Z}|\mathbf{Y}, \Theta'}(\mathbf{z}|\mathbf{y}, \theta') \\ &\quad \times \left( \nabla_{\Theta} \log p_{\mathbf{Z}|\mathbf{Y}, \Theta}(\mathbf{z}|\mathbf{y}, \theta) + \nabla_{\Theta} \log p_{\mathbf{Y}, \Theta}(\mathbf{y}, \theta) \right) d\mathbf{z}, \end{aligned} \quad (143)$$

$$\begin{aligned} &= \nabla_{\Theta} \log p_{\mathbf{Y}, \Theta}(\mathbf{y}, \theta) \int \nabla_{\Theta'} p_{\mathbf{Z}|\mathbf{Y}, \Theta'}(\mathbf{z}|\mathbf{y}, \theta') d\mathbf{z} \\ &\quad + \int p_{\mathbf{Z}|\mathbf{Y}, \Theta'}(\mathbf{z}|\mathbf{y}, \theta) \left( \nabla_{\Theta'} \log p_{\mathbf{Z}|\mathbf{Y}, \Theta'}(\mathbf{z}|\mathbf{y}, \theta') \right)^2 d\mathbf{z} \end{aligned} \quad (144)$$

Since the first term is always equal to zero, we finally have

$$\nabla_{\Theta, \Theta'} \mathcal{Q}_{\Theta, \Theta'}(\theta, \theta) = \int p_{\mathbf{Z}|\mathbf{Y}, \Theta'}(\mathbf{z}|\mathbf{y}, \theta) \left( \nabla_{\Theta'} \log p_{\mathbf{Z}|\mathbf{Y}, \Theta'}(\mathbf{z}|\mathbf{y}, \theta) \right)^2 d\mathbf{z} \succeq 0. \quad (145)$$

□

## APPENDIX IV

### LOCAL SPEED OF CONVERGENCE OF ALGORITHMS BASED ON ITERATIVE MAXIMIZATION

In this appendix, we give a general expression of the local speed of convergence of algorithms computing a new value  $\theta^{(n+1)}$  of the parameters by maximizing a function  $G_{\Theta, \Theta'}(\theta, \theta^{(n)})$ , i.e.

$$\theta^{(n+1)} = \arg \max_{\theta} G_{\Theta, \Theta'}(\theta, \theta^{(n)}). \quad (146)$$

Let  $\theta_f$  be a fixed point of (146). We define the *rate of convergence*  $\mathbf{R}$  of the algorithm as

$$(\theta^{(n+1)} - \theta_f) = \mathbf{R}(\theta^{(n)} - \theta_f) \quad \text{when } \theta^{(n)} \text{ is in a neighborhood of } \theta_f. \quad (147)$$

**Result [18]:** The rate of convergence of (146) in a neighborhood of a fixed point  $\theta_f$  may be characterized as follows

$$\mathbf{R} = - \left( \nabla_{\Theta}^2 G_{\Theta, \Theta'}(\theta_f, \theta_f) \right)^{-1} \nabla_{\Theta, \Theta'} G_{\Theta, \Theta'}(\theta_f, \theta_f) \quad (148)$$

**Proof:** Expanding  $\nabla_{\Theta} G_{\Theta, \Theta'}(\theta, \theta')$  around  $(\theta_f, \theta_f)$ , we have

$$\nabla_{\Theta} G_{\Theta, \Theta'}(\theta, \theta') = \nabla_{\Theta} G_{\Theta, \Theta'}(\theta_f, \theta_f) + \nabla_{\Theta}^2 G_{\Theta, \Theta'}(\theta_f, \theta_f)(\theta - \theta_f) + \nabla_{\Theta, \Theta'} G_{\Theta, \Theta'}(\theta_f, \theta_f)(\theta' - \theta_f). \quad (149)$$

Evaluating  $\nabla_{\Theta} G_{\Theta, \Theta'}(\theta, \theta')$  at  $(\theta^{(n+1)}, \theta^{(n)})$  and taking into account that

$$\nabla_{\Theta} G_{\Theta, \Theta'}(\theta^{(n+1)}, \theta^{(n)}) = 0, \quad (150)$$

$$\nabla_{\Theta} G_{\Theta, \Theta'}(\theta_f, \theta_f) = 0, \quad (151)$$

we get

$$-\nabla_{\Theta}^2 G_{\Theta, \Theta'}(\theta_f, \theta_f)(\theta^{(n+1)} - \theta_f) = \nabla_{\Theta, \Theta'} G_{\Theta, \Theta'}(\theta_f, \theta_f)(\theta^{(n)} - \theta_f). \quad (152)$$

If  $\nabla_{\Theta}^2 G_{\Theta, \Theta'}(\theta_f, \theta_f)$  is not singular, we therefore have (147).  $\square$

## REFERENCES

- [1] A. P. Dempster, N. M. Laird, and D. B. Rubin. "Maximum-likelihood from incomplete data via the EM algorithm". *J. Roy. Stat. Soc.*, 39(1):pp. 1–38, January 1977.
- [2] D.P. Bertsekas. *Nonlinear Programming*. Athena Scientific, USA, 2003.
- [3] F. R. Kschischang, B. J. Frey and H.-A. Loeliger. "Factor graphs and the sum-product algorithm". *IEEE Trans. on Inform. Theory*, 47:pp. 498–519, February 2001.
- [4] L. Zhang and A. Burr. "APPA Symbol Timing Recovery Scheme for Turbo-codes". In *IEEE International Symposium on Personal Indoor and Mobile Radio Communications, PIMRC'*, Lisbonne, Portugal, Nov. 2002.
- [5] J. Dauwels and H.-A. Loeliger. "Phase Estimation by Message Passing". In *IEEE International Conference on Communications, ICC'*, pages 523–527, Paris, France, June 2004.
- [6] C. Herzet, V. Ramon, and L. Vandendorpe. "Turbo-synchronization: a Combined Sum-product and Expectation-Maximization Algorithm Approach". In *IEEE Workshop on Signal Processing Advances in Wireless Communications, SPAWC'*, pages 191– 195, New-York, USA, June 2005.
- [7] C. Herzet. "Code-aided Synchronization for Digital Burst Communications". PhD thesis, available at <http://www.tele.ucl.ac.be/digicom/herzet/index.php>.
- [8] J. Dauwels. "On Graphical Models for Communications and Machine Learning: Algorithms, Bounds, and Analog Implementation". PhD. thesis, Swiss Federal Institute of Technology Zurich, 2005.
- [9] J.S. Yedidia and W.T. Freeman and Y. Weiss. "Constructing Free-Energy Approximations and Generalized Belief Propagation Algorithms". *IEEE Trans. on Inform. Theory*, 51(7), 2005.
- [10] N. Noels, H. Steendam and M. Moeneclaey. "The Cramer-Rao Bound for Phase Estimation from Coded Linearly Modulated Signals". *IEEE Communication Letters*, 7(5):pp. 207–209, May 2003.
- [11] W.I. Zangwill. *Nonlinear Programming: A Unified Approach*. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1969.
- [12] G.J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley Series in Probability and Statistics, USA, 1997.
- [13] R. Cowell. *Advanced inference in Bayesian networks*. MIT Press, Cambridge, MA, USA, 1998.
- [14] T. Heskes. On the uniqueness of loopy belief propagation fixed points. *Neural Computation*, 16:2379–2413, 2004.
- [15] Neal, R. M. and Hinton, G. E. "A view of the EM algorithm that justifies incremental, sparse, and other variants". *Learning in Graphical Models*, pages pp. 355–368, 1998.
- [16] N. Noels, C. Herzet, A. Dejonghe, V. Lottici, H. Steendam, M. Moeneclaey, M Luise and L. Vandendorpe. "Turbo-synchronization: an EM algorithm approach". In *IEEE International Conference on Communications, ICC'*, pages 2933–2937, Anchorage, May 2003.
- [17] Tom Heskes, Onno Zoeter, and Wim Wiegierinck. Approximate expectation maximization. In Sebastian Thrun, Lawrence Saul, and Bernhard Schölkopf, editors, *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA, 2004.
- [18] R. Salakhutdinov, S. Roweis, and Z. Ghahramani. On the convergence of bound optimization algorithms. URL: [citeseer.ist.psu.edu/584732.html](http://citeseer.ist.psu.edu/584732.html).