Stochastic Reasoning, Free Energy, and Information Geometry

Shiro Ikeda *shiro@ism.ac.jp*
Institute of Statistical Mathematics, Tokyo 106-8569, Japan, and
Gatsby Computational Neuroscience Unit, University College London,
London WC1N 3AR, U.K.

Toshiyuki Tanaka *tanaka@eei.metro-u.ac.jp*
Department of Electronics and Information Engineering,
Tokyo Metropolitan University, Tokyo 192-0397, Japan

Shun-ichi Amari *amari@brain.riken.jp*
RIKEN Brain Science Institute, Saitama 351-0198, Japan

Abstract

Belief propagation (BP) is a universal method of stochastic reasoning. It gives exact inference for stochastic models with tree interactions, and works surprisingly well even if the models have loopy interactions. Its performance has been analyzed separately in many fields, such as, AI, statistical physics, information theory, and information geometry. The present paper gives a unified framework to understand BP and related methods, and to summarize the results obtained in many fields. In particular, BP and its variants including tree reparameterization (TRP) and concave-convex procedure (CCCP) are reformulated with information geometrical terms, and their relations to the free energy function are elucidated from information geometrical viewpoint. We then propose a family of new algorithms. The stabilities of the algorithms are analyzed, and methods to accelerate them are investigated.

1 Introduction

Stochastic reasoning is a technique used in wide areas of AI, statistical physics, information theory, and others, to estimate the values of random variables based on partial observation of them (Pearl (1988)). Here, a large number of mutually

interacting random variables are represented in the form of joint probability. However, the interactions often have specific structures such that some variables are independent of others when a set of variables are fixed. In other words, they are conditionally independent, and their interactions take place only through these conditioning variables. When such a structure is represented by a graph, it is called a graphical model (Lauritzen & Spiegelhalter (1988); Jordan (1999)). The problem is to infer the values of unobserved variables based on observed ones by reducing the conditional joint probability distribution to the marginal probability distributions.

When the random variables are binary, their marginal probabilities are determined by the conditional expectation, and the problem is to calculate them. However, when the number of binary random variables is large, the calculation is computationally intractable from the definition. Apart from sampling methods, one way to overcome this problem is to use belief propagation (BP) proposed in AI (Pearl (1988)). It is known that BP gives exact inference when the underlying causal graphical structure does not include any loop, but it is also applied to loopy graphical models (loopy BP), and gives amazingly good approximate inference.

The idea of loopy BP is successfully applied to the decoding algorithms of turbo codes and low-density parity-check (LDPC) codes as well as spin-glass models and Boltzmann machines. It should be also noted that some variants have been proposed to improve the convergence property of loopy BP. Tree reparameterization (TRP) (Wainwright et al. (2002)) is one of them, and convex concave computational procedure (CCCP) (Yuille (2002); Yuille & Rangarajan (2003)) is another algorithm which is reported to have better convergence property.

The reason why loopy BP works so well is not fully understood, and there are a number of theoretical approaches which attempt to analyze its performance. The statistical physical framework utilizes the Bethe free energy (Yedidia et al. (2001a)) or the like (Kabashima & Saad (1999, 2001)), and a geometrical theory was initiated by Richardson (2000) to understand the turbo decoding. Information geometry (Amari & Nagaoka (2000)), which has been successfully used in the study of the mean field approximation (Tanaka (2000, 2001); Amari et al. (2001)), gives a framework to elucidate the mathematical structure of BP (Ikeda et al. (2002, 2004)). A similar framework is also given to describe TRP (Wainwright et al. (2002)).

The problem is interdisciplinary where various concepts and frameworks originate from AI, statistics, statistical physics, information theory, and information geometry. In the present paper, we focus on undirected graphs which is a general representation of graphical models, and give a unified framework to understand BP, CCCP, their variants, and the role of the free energy, based on information geometry. To this end, we propose a new function of the free-energy type to which the Bethe free energy (Yedidia et al. (2001a)) and that of (Kabashima & Saad (2001)) are closely related. By constraining the search space in proper ways, we obtain a family of algorithms including BP, CCCP, and a variant of CCCP without double loops. We also give their stability analysis. The error analysis was given in another paper (Ikeda et al. (2004)).

The paper is organized as follows. In section 2, the problem is stated compactly followed by preliminary of information geometry. Section 3 introduces information geometrical view of BP, the characteristics of its equilibrium, and related algorithms, TRP and CCCP. We discuss the free energy which is related to BP in section 4, and new algorithms are proposed with stability analysis in section 5. Section 6 gives some extensions of BP from information geometrical viewpoint, and finally section 7 concludes the paper.

2 Problem and Geometrical Framework

2.1 Basic Problem and Strategy

Let $\boldsymbol{x} = (x_1, \cdots, x_n)^T$ be hidden and $\boldsymbol{y} = (y_1, \cdots, y_m)^T$ be observed random variables. We start with the case where each x_i is binary i.e., $x_i \in \{-1, +1\}$ for simplicity. An extension to wider class of distributions will be given in subsection 6.1.

The conditional distribution of $\boldsymbol{x}$ given $\boldsymbol{y}$ is written as $q(\boldsymbol{x}|\boldsymbol{y})$, and our task is to give a good inference of $\boldsymbol{x}$ from the observations. We hereafter simply write $q(\boldsymbol{x})$ for $q(\boldsymbol{x}|\boldsymbol{y})$ and omit $\boldsymbol{y}$.

One natural inference of $\boldsymbol{x}$ is the maximum a posteriori (MAP), that is

$$\hat{\boldsymbol{x}}_{map} = \operatorname*{argmax}_{\boldsymbol{x}} q(\boldsymbol{x}).$$

This minimizes the error probability that $\hat{\boldsymbol{x}}_{map}$ does not coincide with the true one. However, this calculation is not tractable when n is large because the number of candidates of $\boldsymbol{x}$ increases exponentially with respect to n. The maximization of the posterior marginals (MPM) is another inference that minimizes the number of component errors. If each marginal distribution $q(x_i)$, $i = 1, \cdots, n$, is known, the MPM inference decides $\hat{x}_i = +1$ when $q(x_i = +1) \geq q(x_i = -1)$ and $\hat{x}_i = -1$ otherwise. Let η_i be the expectation of x_i with respect to $q(\boldsymbol{x})$, that is

$$\eta_i = E_q[x_i] = \sum_{x_i} x_i q(x_i).$$

The MPM inference gives $\hat{x}_i = \mathrm{sgn}\, \eta_i$, which is directly calculated if we know the marginal distributions $q(x_i)$, or the expectation

$$\boldsymbol{\eta} = E_q[\boldsymbol{x}].$$

The present paper focuses on the method to obtain a good approximation to $\boldsymbol{\eta}$, which is equivalent to the inference of $\prod_{i=1}^{n} q(x_i)$.

For any $q(\boldsymbol{x})$, $\ln q(\boldsymbol{x})$ can be expanded as a polynomial of $\boldsymbol{x}$ up to degree n, because every x_i is binary. However, in many problems, mutual interactions of random variables exist only in specific manners. We represent $\ln q(\boldsymbol{x})$ in the form

$$\ln q(\boldsymbol{x}) = \boldsymbol{h} \cdot \boldsymbol{x} + \sum_{r=1}^{L} c_r(\boldsymbol{x}) - \psi_q,$$

where $\boldsymbol{h} \cdot \boldsymbol{x} = \sum_i h_i x_i$ is the linear term, $c_r(\boldsymbol{x}), r = 1, \cdots, L$, is a simple polynomial representing the r-th clique among related variables, and ψ_q is logarithm of the normalizing factor or the partition function, which is called the (Helmholtz) free energy,

$$\psi_q = \ln \sum_{\boldsymbol{x}} \exp\left[\boldsymbol{h} \cdot \boldsymbol{x} + \sum_r c_r(\boldsymbol{x})\right]. \tag{2.1}$$

In the case of Boltzmann machines (Figure 1) and conventional spin-glass models, $c_r(\boldsymbol{x})$ is a quadratic function of x_i, that is,

$$c_r(\boldsymbol{x}) = w_{ij} x_i x_j,$$

where r is the index of the edge which corresponds to the mutual coupling between x_i and x_j.

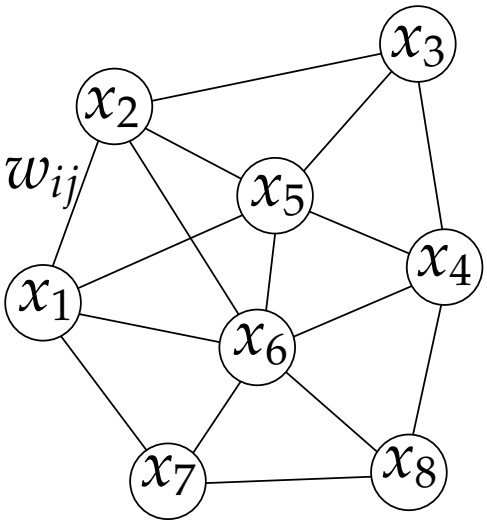

Figure 1: Boltzmann machine.

It is more common to define the true distribution $q(\boldsymbol{x})$ of an undirected graph as a product of clique functions as

$$q(\boldsymbol{x}) = \frac{1}{Z_q} \prod_{i=1}^{n} \phi_i(x_i) \prod_{r \in \mathcal{C}} \phi_r(\boldsymbol{x}_r),$$

where $\mathcal{C}$ is the set of cliques. In our notation, $\phi_i(x_i)$ and $\phi_r(\boldsymbol{x}_r)$ are denoted as follows

$$h_i = \frac{1}{2} \ln \frac{\phi_i(x_i = +1)}{\phi_i(x_i = -1)}, \quad c_r(\boldsymbol{x}) = \ln \phi_r(\boldsymbol{x}_r), \quad \psi_q = \ln Z_q.$$

When there are only pairwise interactions, $\phi_r(\boldsymbol{x}_r)$ has a form of $\phi_r(x_i, x_j)$.

2.2 Important Family of Distributions

Let us consider the set of probability distributions

$$p(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{v}) = \exp[\boldsymbol{\theta} \cdot \boldsymbol{x} + \boldsymbol{v} \cdot \boldsymbol{c}(\boldsymbol{x}) - \psi(\boldsymbol{\theta}, \boldsymbol{v})] \tag{2.2}$$

parameterized by $\boldsymbol{\theta}$ and $\boldsymbol{v}$, where $\boldsymbol{v} = (v_1, \cdots, v_L)^T$, $\boldsymbol{c}(\boldsymbol{x}) = (c_1(\boldsymbol{x}), \cdots, c_L(\boldsymbol{x}))^T$, and $\boldsymbol{v} \cdot \boldsymbol{c}(\boldsymbol{x}) = \sum_{r=1}^{L} v_r c_r(\boldsymbol{x})$. We name the family of the probability distributions S, which is an exponential family

$$S = \left\{ p(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{v}) \;\middle|\; \boldsymbol{\theta} \in \Re^n, \boldsymbol{v} \in \Re^L \right\}, \tag{2.3}$$

where its canonical coordinate system is $(\boldsymbol{\theta}, \boldsymbol{v})$. The joint distribution $q(\boldsymbol{x})$ is included in S, which is easily proved by setting $\boldsymbol{\theta} = \boldsymbol{h}$ and $\boldsymbol{v} = \mathbf{1}_L = (1, \cdots, 1)^T$,

$$q(\boldsymbol{x}) = p(\boldsymbol{x}; \boldsymbol{h}, \mathbf{1}_L).$$

We define M_0 as a submanifold of S specified by $\boldsymbol{v} = \mathbf{0}$,

$$M_0 = \left\{ p_0(\boldsymbol{x}; \boldsymbol{\theta}) = \exp[\boldsymbol{h} \cdot \boldsymbol{x} + \boldsymbol{\theta} \cdot \boldsymbol{x} - \psi_0(\boldsymbol{\theta})] \mid \boldsymbol{\theta} \in \Re^n \right\}.$$

Every distribution of M_0 is an independent distribution which includes no mutual interaction between x_i and x_j, $(i \neq j)$, and canonical coordinate system of M_0 is $\boldsymbol{\theta}$. The product of marginal distributions of $q(\boldsymbol{x})$, that is, $\prod_{i=1}^n q(x_i)$, is included in M_0. The ultimate goal is to derive $\prod_{i=1}^n q(x_i)$ or corresponding coordinate $\boldsymbol{\theta}$ of M_0.

2.3 Preliminary of Information Geometry

In this subsection we give preliminaries of information geometry (Amari & Nagaoka (2000); Amari (2001)). First we define e–flat and m–flat submanifolds of S.

e–flat submanifold: Submanifold $M \subset S$ is said to be e–flat, when, for all $t \in [0, 1], q(\boldsymbol{x}), p(\boldsymbol{x}) \in M$, the following $r(\boldsymbol{x}; t)$ belongs to M.

$$\ln r(\boldsymbol{x}; t) = (1 - t) \ln q(\boldsymbol{x}) + t \ln p(\boldsymbol{x}) + c(t),$$

where $c(t)$ is the normalization factor. Obviously, $\{r(\boldsymbol{x}; t) \mid t \in [0, 1]\}$ is an exponential family connecting two distributions, $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$. When an e–flat submanifold is a one–dimensional curve, it is called an e–geodesic. In terms of the e–affine coordinates $\boldsymbol{\theta}$, a submanifold M is e–flat when it is linear in $\boldsymbol{\theta}$.

m–flat submanifold: Submanifold $M \subset S$ is said to be m–flat when, for all $t \in [0, 1], q(\boldsymbol{x}), p(\boldsymbol{x}) \in M$, the following mixture $r(\boldsymbol{x}; t)$ belongs to M.

$$r(\boldsymbol{x}; t) = (1 - t) q(\boldsymbol{x}) + t p(\boldsymbol{x}).$$

When an m–flat submanifold is a one–dimensional curve, it is called an m–geodesic. Hence, the above mixture family is the m–geodesic connecting them.

From the definition, any exponential family is an *e*–flat manifold. Therefore S and M_0 are *e*–flat. Next we define the *m*–projection (Amari & Nagaoka (2000)).

Definition 1. *Let M be an e–flat submanifold in S, and let $q(\boldsymbol{x})\in S$. The point in M that minimizes the KL–divergence from $q(\boldsymbol{x})$ to M, denoted by*

$$\Pi_M \circ q(\boldsymbol{x}) = \underset{p(\boldsymbol{x})\in M}{\operatorname{argmin}} D[q(\boldsymbol{x}); p(\boldsymbol{x})] \tag{2.4}$$

is called the m–projection of $q(\boldsymbol{x})$ to M.

Here, $D[\cdot;\cdot]$ is the KL (Kullback-Leibler)-divergence defined as

$$D[q(\boldsymbol{x}); p(\boldsymbol{x})] = \sum_{\boldsymbol{x}} q(\boldsymbol{x}) \ln \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})}.$$

The KL–divergence satisfies $D[q(\boldsymbol{x}); p(\boldsymbol{x})]\geq 0$, and $D[q(\boldsymbol{x}); p(\boldsymbol{x})] = 0$ when and only when $q(\boldsymbol{x}) = p(\boldsymbol{x})$ holds for every $\boldsymbol{x}$. Although symmetry $D[q; p] = D[p; q]$ does not hold in general, it is regarded as an asymmetric squared distance. Finally, the *m*–projection theorem follows.

Theorem 1. *Let M be an e–flat submanifold in S, and let $q(\boldsymbol{x})\in S$. The m–projection of $q(\boldsymbol{x})$ to M is unique and given by a point in M such that the m–geodesic connecting $q(\boldsymbol{x})$ and $\Pi_M \circ q$ is orthogonal to M at this point in the sense of the Riemannian metric due to the Fisher information matrix.*

Proof. A detailed proof is found in Amari & Nagaoka (2000) and the following is a sketch of it. First, we define the inner product and prove the orthogonality. A rigorous definition concerning the tangent space of manifold is found in Amari & Nagaoka (2000).

Let us consider a curve $p(\boldsymbol{x};\alpha) \in S$, which is parameterized by a real-valued parameter α. Its tangent vector is represented by a random vector $\partial_\alpha \ln p(\boldsymbol{x};\alpha)$, where $\partial_\alpha = \partial/\partial\alpha$. For two curves $p_1(\boldsymbol{x};\alpha)$ and $p_2(\boldsymbol{x};\beta)$ which intersect at $\alpha = \beta = 0$, $p(\boldsymbol{x}) = p_1(\boldsymbol{x};0) = p_2(\boldsymbol{x};0)$, we define the inner product of the two tangent vectors by

$$E_{p(\boldsymbol{x})}[\partial_\alpha \ln p_1(\boldsymbol{x};\alpha)\partial_\beta \ln p_2(\boldsymbol{x};\beta)]_{\alpha=\beta=0}.$$

Note that this definition is consistent with the Riemannian metric defined by the Fisher information matrix.

Let $p^*(\boldsymbol{x})$ be an m–projection of $q(\boldsymbol{x})$ to M, and the m–geodesic connecting $q(\boldsymbol{x})$ and $p^*(\boldsymbol{x})$ be $r_m(\boldsymbol{x};\alpha)$, which is defined as

$$r_m(\boldsymbol{x};\alpha) = \alpha q(\boldsymbol{x}) + (1-\alpha)p^*(\boldsymbol{x}), \quad \alpha \in [0,1].$$

The derivative of $\ln r_m(\boldsymbol{x};\alpha)$ along the m–geodesic at $p^*(\boldsymbol{x})$ is

$$\partial_\alpha \ln r_m(\boldsymbol{x};\alpha)|_{\alpha=0} = \frac{q(\boldsymbol{x}) - p^*(\boldsymbol{x})}{r_m(\boldsymbol{x};\alpha)}\Big|_{\alpha=0} = \frac{q(\boldsymbol{x}) - p^*(\boldsymbol{x})}{p^*(\boldsymbol{x})}.$$

Let an e–geodesic included in M be $r_e(\boldsymbol{x};\beta)$, which is defined as

$$\ln r_e(\boldsymbol{x};\beta) = \beta \ln p'(\boldsymbol{x}) + (1-\beta)\ln p^*(\boldsymbol{x}) + c(\beta), \quad p'(\boldsymbol{x}) \in M, \quad \beta \in [0,1].$$

The derivative of $\ln r_e(\boldsymbol{x};\beta)$ along the e–geodesic at $p^*(\boldsymbol{x})$ is

$$\partial_\beta \ln r_e(\boldsymbol{x};\beta)|_{\beta=0} = \ln p'(\boldsymbol{x}) - \ln p^*(\boldsymbol{x}) + c'(0).$$

The inner product becomes

$$E_{p^*(\boldsymbol{x})}[\partial_\alpha \ln p^*(\boldsymbol{x}) \partial_\beta \ln p^*(\boldsymbol{x})] = \sum_{\boldsymbol{x}} [q(\boldsymbol{x}) - p^*(\boldsymbol{x})][\ln p'(\boldsymbol{x}) - \ln p^*(\boldsymbol{x})]. \tag{2.5}$$

The fact that $p^*(\boldsymbol{x})$ is an m–projection from $q(\boldsymbol{x})$ to M gives $\partial_\beta D[q; r_e(\beta)]|_{\beta=0} = 0$, that is,

$$\partial_\beta D[q(\boldsymbol{x}); r_e(\boldsymbol{x};\beta)]\big|_{\beta=0} = \sum_{\boldsymbol{x}} q(\boldsymbol{x})[\ln p^*(\boldsymbol{x}) - \ln p'(\boldsymbol{x})] - c'(0) = 0. \tag{2.6}$$

Moreover, since $D[p^*; r_e(\beta)]$ is minimized to 0 at $\beta = 0$, we have

$$\partial_\beta D[p^*(\boldsymbol{x}); r_e(\boldsymbol{x};\beta)]\big|_{\beta=0} = \sum_{\boldsymbol{x}} p^*(\boldsymbol{x})[\ln p^*(\boldsymbol{x}) - \ln p'(\boldsymbol{x})] - c'(0) = 0. \tag{2.7}$$

Equation 2.5 is proved to be zero by combining equations 2.6 and 2.7. Furthermore, it immediately proves the Pythagorean theorem

$$D[q(\boldsymbol{x}); p'(\boldsymbol{x})] = D[q(\boldsymbol{x}); p^*(\boldsymbol{x})] + D[p^*(\boldsymbol{x}); p'(\boldsymbol{x})].$$

This holds for every $p'(\boldsymbol{x}) \in M$. Suppose the m–projection is not unique, and let another point be $p^{**}(\boldsymbol{x}) \in M$ which satisfies $D[q; p^{**}] = D[q; p^*]$. Then the following equation holds

$$D[q(\boldsymbol{x}); p^{**}(\boldsymbol{x})] = D[q(\boldsymbol{x}); p^*(\boldsymbol{x})] + D[p^*(\boldsymbol{x}); p^{**}(\boldsymbol{x})] = D[q(\boldsymbol{x}); p^*(\boldsymbol{x})].$$

This is true only if $p^*(\boldsymbol{x}) = p^{**}(\boldsymbol{x})$ which proves the uniqueness of the m–projection. □

2.4 MPM Inference

We show the MPM inference is immediately given if the m–projection from $q(\boldsymbol{x})$ to M_0 is given. From the definition in equation 2.4, the m–projection of $q(\boldsymbol{x})$ to M_0 is characterized by $\boldsymbol{\theta}^*$, that satisfies

$$p_0(\boldsymbol{x}; \boldsymbol{\theta}^*) = \Pi_{M_0} \circ q(\boldsymbol{x}).$$

Hereafter, we denote the m–projection to M_0 in terms of the parameter $\boldsymbol{\theta}$ as,

$$\boldsymbol{\theta}^* = \pi_{M_0} \circ q(\boldsymbol{x}) = \operatorname*{argmin}_{\boldsymbol{\theta}} D[q(\boldsymbol{x}); p_0(\boldsymbol{x}; \boldsymbol{\theta})].$$

By taking the derivative of $D[q(\boldsymbol{x}); p_0(\boldsymbol{x}; \boldsymbol{\theta})]$ with respect to $\boldsymbol{\theta}$, we have

$$\sum_{\boldsymbol{x}} \boldsymbol{x} q(\boldsymbol{x}) - \partial_{\boldsymbol{\theta}} \psi_0(\boldsymbol{\theta}^*) = \mathbf{0}, \tag{2.8}$$

where $\partial_{\boldsymbol{\theta}}$ shows the derivative with respect to $\boldsymbol{\theta}$. From the definition of exponential family,

$$\partial_{\boldsymbol{\theta}} \psi_0(\boldsymbol{\theta}) = \partial_{\boldsymbol{\theta}} \ln \sum_{\boldsymbol{x}} \exp(\boldsymbol{h} \cdot \boldsymbol{x} + \boldsymbol{\theta} \cdot \boldsymbol{x}) = \sum_{\boldsymbol{x}} \boldsymbol{x} p_0(\boldsymbol{x}; \boldsymbol{\theta}). \tag{2.9}$$

We define the new parameter $\boldsymbol{\eta}_0(\boldsymbol{\theta})$ in M_0 as

$$\boldsymbol{\eta}_0(\boldsymbol{\theta}) = \sum_{\boldsymbol{x}} \boldsymbol{x} p_0(\boldsymbol{x}; \boldsymbol{\theta}) = \partial_{\boldsymbol{\theta}} \psi_0(\boldsymbol{\theta}). \tag{2.10}$$

This is called the expectation parameter (Amari & Nagaoka (2000)). From equations 2.8, 2.9, and 2.10, the m–projection is equivalent to marginalizing $q(\boldsymbol{x})$. Since translation between $\boldsymbol{\theta}$ and $\boldsymbol{\eta}_0$ is straightforward for M_0, once the m–projection or equivalently the product of marginals of $q(\boldsymbol{x})$ is obtained, the MPM inference is given immediately.

3 BP and Variants: Information Geometrical View

3.1 BP

Information Geometrical View of BP

In this subsection, we give the information geometrical view of BP. The well-known definition of BP is found somewhere else (Pearl (1988); Lauritzen & Spiegelhalter (1988); Weiss (2000)), and the detail is not given in this paper. We note that our

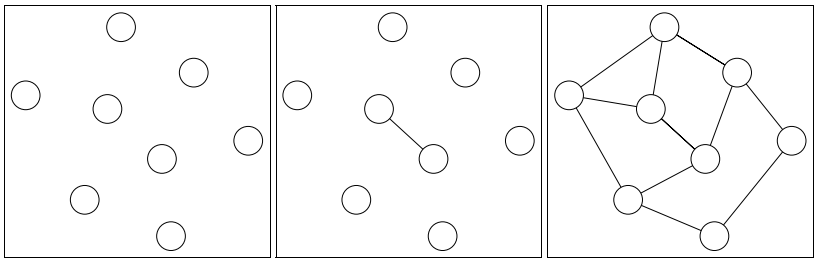

Figure 2: A: Belief graph, B: Graph with a single edge, C: Graph with all edges.

derivation is based on BP for undirected graphs. For loopy graphs, it is well-known that BP does not necessarily converge, and even if it does, the result is not equal to the true marginals.

Figure 2 shows three important graphs for BP. The belief graph in Figure 2.A corresponds to $p_0(\boldsymbol{x};\boldsymbol{\theta})$, and that in Figure 2.C corresponds to the true distribution $q(\boldsymbol{x})$. Figure 2.B shows an important distribution which includes only a single edge. This distribution is defined as $p_r(\boldsymbol{x};\boldsymbol{\zeta}_r)$ where

$$p_r(\boldsymbol{x};\boldsymbol{\zeta}_r) = \exp\Big[\boldsymbol{h}\cdot\boldsymbol{x} + c_r(\boldsymbol{x}) + \boldsymbol{\zeta}_r\cdot\boldsymbol{x} - \psi_r(\boldsymbol{\zeta}_r)\Big], \quad r = 1,\cdots,L.$$

This can be generalized without any change to the case when $c_r(\boldsymbol{x})$ is a polynomial. The set of the distributions $p_r(\boldsymbol{x};\boldsymbol{\zeta}_r)$ parameterized by $\boldsymbol{\zeta}_r$ is an e–flat manifold defined as

$$M_r = \Big\{p_r(\boldsymbol{x};\boldsymbol{\zeta}_r) \mid \boldsymbol{\zeta}_r \in \Re^n\Big\}, \quad r = 1,\cdots,L.$$

Its canonical coordinate system is $\boldsymbol{\zeta}_r$. We also define the expectation parameter $\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r)$ of M_r as follows

$$\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r) = \partial_{\zeta_r}\psi_r(\boldsymbol{\zeta}_r) = \sum_{\boldsymbol{x}} \boldsymbol{x}\, p_r(\boldsymbol{x};\boldsymbol{\zeta}_r), \quad r = 1,\cdots,L. \tag{3.1}$$

In M_r, only the r-th edge is taken into account but all the other edges are replaced by a linear term $\boldsymbol{\zeta}_r\cdot\boldsymbol{x}$ and $p_0(\boldsymbol{x};\boldsymbol{\theta}) \in M_0$ is used to integrate all the information from $p_r(\boldsymbol{x};\boldsymbol{\zeta}_r), r = 1,\cdots,L$, giving $\boldsymbol{\theta}$, which is the parameter of $p_0(\boldsymbol{x};\boldsymbol{\theta})$, to infer $\prod_i q(x_i)$. In the iterative process of BP $\boldsymbol{\zeta}_r$ of $p_r(\boldsymbol{x};\boldsymbol{\zeta}_r), r = 1,\cdots,L$ are modified by using the information of $\boldsymbol{\theta}$, which in turn is renewed by integrating local information $\{\boldsymbol{\zeta}_r\}$. Information geometry has elucidated its geometrical meaning for special graphs for

error correcting codes (Ikeda et al. (2004), see also Richardson (2000)), and we give the framework for general graphs in the following.

BP is stated as follows: Let $p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^t)$ be the approximation to $q(\boldsymbol{x})$ at time t, which each M_r, $r = 1, \cdots, L$ specifies.

Information geometrical view of BP

1. *Set* $t = 0$, $\boldsymbol{\xi}_r^t = \boldsymbol{0}$, $\boldsymbol{\zeta}_r^t = \boldsymbol{0}$, $r = 1, \cdots, L$.

2. *Increment* t *by one and set* $\boldsymbol{\xi}_r^{t+1}$, $r = 1, \cdots, L$ *as follows,*

$$\boldsymbol{\xi}_r^{t+1} = \pi_{M_0} \circ p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^t) - \boldsymbol{\zeta}_r^t. \tag{3.2}$$

3. *Update* $\boldsymbol{\theta}^{t+1}$ *and* $\boldsymbol{\zeta}_r^{t+1}$ *as follows,*

$$\boldsymbol{\zeta}_r^{t+1} = \sum_{r' \neq r} \boldsymbol{\xi}_{r'}^{t+1}, \quad \boldsymbol{\theta}^{t+1} = \sum_r \boldsymbol{\xi}_r^{t+1} = \frac{1}{L-1} \sum_r \boldsymbol{\zeta}_r^{t+1}.$$

4. *Repeat steps 2 and 3 until convergence.*

The algorithm is summarized as follows: Calculate iteratively

$$\begin{aligned} \boldsymbol{\theta}^{t+1} &= \sum_r [\pi_{M_0} \circ p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^t) - \boldsymbol{\zeta}_r^t], \\ \boldsymbol{\zeta}_r^{t+1} &= \boldsymbol{\theta}^{t+1} - [\pi_{M_0} \circ p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^t) - \boldsymbol{\zeta}_r^t], \quad r = 1, \cdots, L. \end{aligned}$$

We have introduced two sets of parameters $\{\boldsymbol{\xi}_r\}$ and $\{\boldsymbol{\zeta}_r\}$. Let the converged point of BP be $\{\boldsymbol{\xi}_r^*\}$, $\{\boldsymbol{\zeta}_r^*\}$, and $\boldsymbol{\theta}^*$, where $\boldsymbol{\theta}^* = \sum_r \boldsymbol{\xi}_r^* = \sum_r \boldsymbol{\zeta}_r^* / (L-1)$ and $\boldsymbol{\theta}^* = \boldsymbol{\xi}_r^* + \boldsymbol{\zeta}_r^*$. With these relations, the probability distribution of $q(\boldsymbol{x})$, its final approximations $p_0(\boldsymbol{x}; \boldsymbol{\theta}^*) \in M_0$, and $p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^*) \in M_r$ are described as

$$\begin{aligned} q(\boldsymbol{x}) &= \exp[\boldsymbol{h} \cdot \boldsymbol{x} + c_1(\boldsymbol{x}) + \cdots + c_r(\boldsymbol{x}) + \cdots + c_L(\boldsymbol{x}) - \psi_q], \\ p_0(\boldsymbol{x}; \boldsymbol{\theta}^*) &= \exp[\boldsymbol{h} \cdot \boldsymbol{x} + \boldsymbol{\xi}_1^* \cdot \boldsymbol{x} + \cdots + \boldsymbol{\xi}_r^* \cdot \boldsymbol{x} + \cdots + \boldsymbol{\xi}_L^* \cdot \boldsymbol{x} - \psi_0(\boldsymbol{\theta}^*)], \\ p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^*) &= \exp[\boldsymbol{h} \cdot \boldsymbol{x} + \boldsymbol{\xi}_1^* \cdot \boldsymbol{x} + \cdots + c_r(\boldsymbol{x}) + \cdots + \boldsymbol{\xi}_L^* \cdot \boldsymbol{x} - \psi_r(\boldsymbol{\zeta}_r^*)]. \end{aligned}$$

The idea of BP is to approximate $c_r(\boldsymbol{x})$ by $\boldsymbol{\xi}_r^* \cdot \boldsymbol{x}$ in M_r, taking the information from $M_{r'}$ $(r' \neq r)$ into account. The independent distribution $p_0(\boldsymbol{x}; \boldsymbol{\theta})$ integrates all the information.

Common BP Formulation and Information Geometrical One

BP is generally described as a set of message updating rules. Here, we describe the correspondence between common formulation and information geometrical one. In the graphs with pairwise interactions, messages and believes are updated as

$$\begin{aligned} m_{ij}^{t+1}(x_j) &= \frac{1}{Z}\sum_{x_i}\phi_i(x_i)\phi_{ij}(x_i,x_j)\prod_{k\in\mathcal{N}(i)\setminus j} m_{ki}^{t}(x_i), \\ b_i(x_i) &= \frac{1}{Z}\phi_i(x_i)\prod_{k\in\mathcal{N}(i)} m_{ki}^{t+1}(x_i), \end{aligned}$$

where Z is the normalization factor and $\mathcal{N}(i)$ is the set of vertices connected to vertex i. The vector $\boldsymbol{\xi}_r$ corresponds to $m_{ij}(x_j)$. More precisely, when r is a edge connecting i and j,

$$\xi_{r,j} = \frac{1}{2}\ln\frac{m_{ij}(x_j=+1)}{m_{ij}(x_j=-1)}, \quad \xi_{r,i} = \frac{1}{2}\ln\frac{m_{ji}(x_i=+1)}{m_{ji}(x_i=-1)}, \quad \xi_{r,k} = 0 \text{ for } k\neq i,j,$$

where $\xi_{r,i}$ denotes the i-th component of $\boldsymbol{\xi}_r$. Note that i-th component of $\boldsymbol{\xi}_r$ is not generally 0 if r-th edge includes vertex i and that equation 3.2 updates $m_{ij}(x_j)$ and $m_{ji}(x_i)$ simultaneously. Now, it is not difficult to understand the following correspondences

$$\begin{aligned} \theta_i &= \sum_{r'}\xi_{r',i} = \frac{1}{2}\ln\prod_{k\in\mathcal{N}(i)}\frac{m_{ki}(x_i=+1)}{m_{ki}(x_i=-1)}, \\ \zeta_{r,i} &= \theta_i - \xi_{r,i} = \frac{1}{2}\ln\prod_{k\in\mathcal{N}(i)\setminus j}\frac{m_{ki}(x_i=+1)}{m_{ki}(x_i=-1)}, \end{aligned}$$

where θ_i and $\zeta_{r,i}$ are the i-th component of $\boldsymbol{\theta}$ and $\boldsymbol{\zeta}_r$ respectively and r corresponds to the edge connecting i and j. Note that $\zeta_{r,k} = \theta_k$ holds for $k \neq i,j$.

Equilibrium of BP

The following theorem proved in Ikeda et al. (2004) characterizes the equilibrium of BP.

Theorem 2. *The equilibrium* $(\boldsymbol{\theta}^*, \{\boldsymbol{\zeta}_r^*\})$ *satisfies*

1) *m–condition:* $\boldsymbol{\theta}^* = \pi_{M_0}\circ p_r(\boldsymbol{x};\boldsymbol{\zeta}_r^*)$.

2) *e–condition:* $\boldsymbol{\theta}^* = \dfrac{1}{L-1}\displaystyle\sum_{r=1}^{L}\boldsymbol{\zeta}_r^*$.

It is easy to check that the m–condition is satisfied at the equilibrium of the BP algorithm from equation 3.2 and $\boldsymbol{\theta}^* = \boldsymbol{\zeta}_r^* + \boldsymbol{\xi}_r^*$. In order to check the e–condition, we have to note that $\boldsymbol{\xi}_r^*$ corresponds to a message. If the same set of messages is used to calculate the belief of each vertex, the e–condition is automatically satisfied. Therefore, at each iteration of BP, the e–condition is satisfied. Although, in some algorithms, multiple sets of messages are defined, and a different set is used to calculate each belief. In such cases, the e–condition plays an important role.

In order to have an information geometrical view, we define two submanifolds M^* and E^* of S (see equation 2.3) as follows,

$$\begin{aligned} M^* &= \left\{ p(\boldsymbol{x}) \;\middle|\; p(\boldsymbol{x}) \in S,\ \sum_{\boldsymbol{x}} \boldsymbol{x} p(\boldsymbol{x}) = \sum_{\boldsymbol{x}} \boldsymbol{x} p_0(\boldsymbol{x}; \boldsymbol{\theta}^*) = \boldsymbol{\eta}_0(\boldsymbol{\theta}^*) \right\}, \\ E^* &= \left\{ p(\boldsymbol{x}) = C p_0(\boldsymbol{x}; \boldsymbol{\theta}^*)^{t_0} \prod_{r=1}^{L} p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^*)^{t_r} \;\middle|\; \sum_{r=0}^{L} t_r = 1,\ t_r \in \Re \right\}, \qquad (3.3) \\ C &: \text{normalization factor.} \end{aligned}$$

Note that M^* and E^* are an m–flat and an e–flat submanifold, respectively.

The geometrical implications of these conditions are as follows:

m–condition: The m–flat submanifold M^* which includes $p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^*)$, $r = 1, \cdots, L$, and $p_0(\boldsymbol{x}; \boldsymbol{\theta}^*)$ is orthogonal to M_r, $r = 1, \cdots, L$ and M_0, that is, they are the m–projections to each other.

e–condition: The e–flat submanifold E^* includes $p_0(\boldsymbol{x}; \boldsymbol{\theta}_0^*)$, $p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^*)$, $r = 1, \cdots, L$, and $q(\boldsymbol{x})$.

The equivalence between the e–condition of theorem 2 and the geometrical one stated above is proved straightforwardly by setting $t_0 = -(L-1)$ and $t_1 = \cdots = t_L = 1$ in equation 3.3.

From the m–condition, $\sum_{\boldsymbol{x}} \boldsymbol{x} p_0(\boldsymbol{x}; \boldsymbol{\theta}^*) = \sum_{\boldsymbol{x}} \boldsymbol{x} p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^*)$ holds, and from the definitions in equations 2.10 and 3.1, we have,

$$\boldsymbol{\eta}_0(\boldsymbol{\theta}^*) = \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r^*), \quad r = 1, \cdots, L. \qquad (3.4)$$

It is not difficult to show that equation 3.4 is the necessary and sufficient condition for the m–condition, and it implies not only that the m–projection of $p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^*)$ to M_0 is $p_0(\boldsymbol{x}; \boldsymbol{\theta}^*)$, but also that the m–projection of $p_0(\boldsymbol{x}; \boldsymbol{\theta}^*)$ to M_r is $p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^*)$, that is,

$$\boldsymbol{\zeta}_r^* = \pi_{M_r} \circ p_0(\boldsymbol{x}; \boldsymbol{\theta}^*), \quad r = 1, \cdots, L.$$

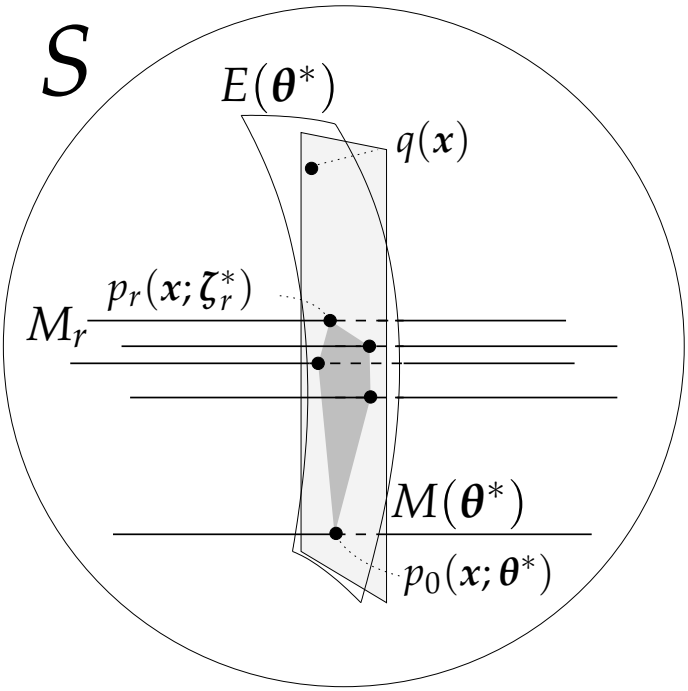

Figure 3: Structure of equilibrium.

where π_{M_r} denotes the m–projection to M_r. When BP converges, the e–condition and the m–condition are satisfied, but it does not necessarily imply $q(\boldsymbol{x}) \in M^*$, in other words $p_0(\boldsymbol{x};\boldsymbol{\theta}^*) = \prod_{i=1}^n q(x_i)$, because there is a discrepancy between M^* and E^*. This is shown schematically in Figure 3.

It is well-known that in the graphs with tree structures, BP gives the true marginals, that is, $q(\boldsymbol{x}) \in M^*$ holds. In this case, we have the following relation

$$q(\boldsymbol{x}) = \frac{\prod_{r=1}^L p_r(\boldsymbol{x};\boldsymbol{\zeta}_r^*)}{p_0(\boldsymbol{x};\boldsymbol{\theta}^*)^{L-1}}. \tag{3.5}$$

This relationship gives the following proposition.

Proposition 1. *When $q(\boldsymbol{x})$ is represented with a tree graph, $q(\boldsymbol{x})$, $p_0(\boldsymbol{x};\boldsymbol{\theta}^*)$, and $p_r(\boldsymbol{x};\boldsymbol{\zeta}_r^*)$, $r = 1, \cdots, L$ are included in M^* and E^* simultaneously.*

This proposition shows that when a graph is a tree, $q(\boldsymbol{x})$ and $p_0(\boldsymbol{x};\boldsymbol{\theta}^*)$ are included in M^* and the fixed point of BP is the correct solution. In the case of a loopy graph, $q(\boldsymbol{x}) \notin M^*$ and the correct solution is not generally a fixed point of BP.

However, we still hope that BP gives a good approximation to the correct marginals. The difference between the correct marginals and the BP solution

is regarded as the discrepancy between E^* and M^*, and if we can qualitatively evaluate it, the error of the BP solution is estimated. We have given a preliminary analysis in Ikeda et al. (2002, 2004), which showed that the principal term of the error is directly related to the e–curvature (see Amari & Nagaoka (2000) for the definition) of M^*, which mainly reflects the influence of the possible shortest loops in the graph.

3.2 TRP

There have been proposed some variants of BP, and information geometry gives a general framework to understand them. We begin with TRP (Wainwright et al. (2002)). TRP selects the set of trees $\{\mathcal{T}_i\}$, where each tree $\mathcal{T}_i$ consists of a set of edges, and renew related parameters in the process of inference. Let the set of edges be $\mathcal{L}$ and $\mathcal{T}_i \subset \mathcal{L}$, $i = 1, \cdots, K$ be its subsets where each graph with the edges $\mathcal{T}_i$ does not have any loop. The choice of the sets $\{\mathcal{T}_i\}$ is arbitrary, but every edge must be included at least in one of the trees.

In order to give the information geometrical view, we use the parameters $\boldsymbol{\zeta}_r$, $\boldsymbol{\theta}_r$, $r = 1, \cdots, L$, and $\boldsymbol{\theta}$. The information geometrical view of TRP is given as follows,

Information geometrical view of TRP

1. *Set* $t = 0$, $\boldsymbol{\zeta}_r^t = \boldsymbol{\theta}_r^t = \mathbf{0}$, $r = 1, \cdots, L$, *and* $\boldsymbol{\theta}^t = \mathbf{0}$.

2. *For a tree* $\mathcal{T}_i$, *construct a tree distribution* $p_{\mathcal{T}_i}^t(\boldsymbol{x})$ *as follows*

$$\begin{aligned} p_{\mathcal{T}_i}^t(\boldsymbol{x}) &= C p_0(\boldsymbol{x}; \boldsymbol{\theta}^t) \prod_{r \in \mathcal{T}_i} \frac{p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^t)}{p_0(\boldsymbol{x}; \boldsymbol{\theta}_r^t)} \\ &= C' \exp\Big\{ \boldsymbol{h} \cdot \boldsymbol{x} + \sum_{r \in \mathcal{T}_i} c_r(\boldsymbol{x}) + \big[\sum_{r \in \mathcal{T}_i} (\boldsymbol{\zeta}_r^t - \boldsymbol{\theta}_r^t) + \boldsymbol{\theta}^t \big] \cdot \boldsymbol{x} \Big\}. \end{aligned} \tag{3.6}$$

   By applying BP, calculate the marginal distribution of $p_{\mathcal{T}_i}^t(\boldsymbol{x})$, *and let* $\boldsymbol{\theta}^{t+1} = \pi_{M_0} \circ p_{\mathcal{T}_i}^t(\boldsymbol{x})$. *Then update* $\boldsymbol{\theta}_r^{t+1}$ *and* $\boldsymbol{\zeta}_r^{t+1}$ *as follows,*

   For $r \in \mathcal{T}_i$,

$$\boldsymbol{\theta}_r^{t+1} = \boldsymbol{\theta}^{t+1}, \quad \boldsymbol{\zeta}_r^{t+1} = \pi_{M_r} \circ p_{\mathcal{T}_i}^t(\boldsymbol{x}).$$

   For $r \notin \mathcal{T}_i$,

$$\boldsymbol{\theta}_r^{t+1} = \boldsymbol{\theta}_r^t, \quad \boldsymbol{\zeta}_r^{t+1} = \boldsymbol{\zeta}_r^t.$$

3. *Repeat step 2 for trees* $\mathcal{T}_j \in \{\mathcal{T}_i\}$.

4. *Repeat steps 2 and 3 until* $\boldsymbol{\theta}_r^{t+1} = \boldsymbol{\theta}^{t+1}$*, holds for every* r*, and* $\{\boldsymbol{\zeta}_r^{t+1}\}$ *converges.*

Let us show that the e– and the m–conditions are satisfied at the equilibrium of TRP. Since $p_{\mathcal{T}_i}^t(\boldsymbol{x})$ is a tree graph, BP in step 2 gives the exact inference of marginal distributions. Moreover, from equations 3.5 and 3.6, we have

$$p_{\mathcal{T}_i}^t(\boldsymbol{x}) = C p_0(\boldsymbol{x};\boldsymbol{\theta}^t)\frac{\prod_{r\in\mathcal{T}_i} p_r(\boldsymbol{x};\boldsymbol{\zeta}_r^t)}{\prod_{r\in\mathcal{T}_i} p_0(\boldsymbol{x};\boldsymbol{\theta}_r^t)} = \frac{\prod_{r\in\mathcal{T}_i} p_r(\boldsymbol{x};\boldsymbol{\zeta}_r^{t+1})}{p_0(\boldsymbol{x};\boldsymbol{\theta}^{t+1})^{|\mathcal{T}_i|-1}}.$$

where $|\mathcal{T}_i|$ is the cardinality of $\mathcal{T}_i$. By comparing 2nd and 3rd terms, and using $\boldsymbol{\theta}_r^{t+1} = \boldsymbol{\theta}^{t+1}$, $r \in \mathcal{T}_i$,

$$\sum_{r\in\mathcal{T}_i}(\boldsymbol{\zeta}_r^t - \boldsymbol{\theta}_r^t) + \boldsymbol{\theta}^t = \sum_{r\in\mathcal{T}_i}\boldsymbol{\zeta}_r^{t+1} - (|\mathcal{T}_i|-1)\boldsymbol{\theta}^{t+1} = \sum_{r\in\mathcal{T}_i}(\boldsymbol{\zeta}_r^{t+1} - \boldsymbol{\theta}_r^{t+1}) + \boldsymbol{\theta}^{t+1}.$$

Since $\sum_{r\notin\mathcal{T}_i}(\boldsymbol{\zeta}_r^t - \boldsymbol{\theta}_r^t)$ does not change through step 2, we have the following relation, which shows the e–condition holds for the convergent point of TRP,

$$\sum_r \boldsymbol{\zeta}_r^* - (L-1)\boldsymbol{\theta}^* = \sum_r(\boldsymbol{\zeta}_r^* - \boldsymbol{\theta}_r^*) + \boldsymbol{\theta}^* = \sum_r(\boldsymbol{\zeta}_r^t - \boldsymbol{\theta}_r^t) + \boldsymbol{\theta}^t = \mathbf{0}.$$

When TRP converges, the operation of step 2 shows that each tree distribution has the same marginal distribution, which shows $p_{\mathcal{T}_i}^*(\boldsymbol{x}) \in M^*$, where $p_{\mathcal{T}_i}^*(\boldsymbol{x})$ is the tree distribution constructed with the converged parameters. Since $\boldsymbol{\zeta}_r^* = \pi_{M_r} \circ p_{\mathcal{T}_i}^*(\boldsymbol{x})$, $r \in \mathcal{T}_i$ holds, $p_r(\boldsymbol{x};\boldsymbol{\zeta}_r^*) \in M^*$ also holds for $r = 1, \cdots, L$, which shows the m–condition is satisfied at the convergent point.

3.3 CCCP

CCCP is an iterative procedure to obtain the minimum of a function, which is represented by the difference of two convex functions (Yuille & Rangarajan (2003)). The idea of CCCP was applied to solve the inference problem of loopy graphs, where the Bethe free energy, which we will discuss in section 4, is the energy function (Yuille (2002)) (therefore, it is CCCP-Bethe, but in the following, we refer it as CCCP). The detail of the derivation will be given in Appendix, and CCCP is defined as follows in information geometrical framework.

Information geometrical view of CCCP

inner loop: *Given $\boldsymbol{\theta}^t$, calculate $\{\boldsymbol{\zeta}_r^{t+1}\}$ by solving*

$$\pi_{M_0} \circ p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^{t+1}) = L\boldsymbol{\theta}^t - \sum_r \boldsymbol{\zeta}_r^{t+1}, \quad r = 1, \cdots, L. \tag{3.7}$$

outer loop: *Given a set of $\{\boldsymbol{\zeta}_r^{t+1}\}$ as the result of the inner loop, calculate*

$$\boldsymbol{\theta}^{t+1} = L\boldsymbol{\theta}^t - \sum_r \boldsymbol{\zeta}_r^{t+1}. \tag{3.8}$$

From equations 3.7 and 3.8, one obtains

$$\boldsymbol{\theta}^{t+1} = \pi_{M_0} \circ p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r^{t+1}), \quad r = 1, \cdots, L,$$

which means that CCCP enforces the m–condition at each iteration. On the other hand, the e–condition is satisfied only at the convergent point, which can be easily verified by letting $\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t = \boldsymbol{\theta}^*$ in equation 3.8 to yield the e–condition $(L-1)\boldsymbol{\theta}^* = \sum_r \boldsymbol{\zeta}_r^*$. One can therefore regard that the inner and outer loops of CCCP solve the m–condition and the e–condition, respectively.

4 Free Energy Function

4.1 Bethe Free Energy

We have described the information geometrical view of BP and related algorithms. It gives the characteristics of the equilibrium points, but it is not enough to describe the approximation accuracy, and the dynamics of the algorithm.

An energy function helps us to clarify them, and there are some functions proposed for this purpose. Most popular one is the Bethe free energy. The Bethe free energy itself has been well known in the literature of statistical mechanics, being used in formulating the so-called Bethe approximation (Itzykson & Drouffe (1989)). As far as we know, Kabashima & Saad (2001) were the first to point out that BP is derived by considering a variational extremization of a free energy. It was Yedidia et al. (2001a) who introduced to the machine-learning community the formulation of BP based on the Bethe free energy. Following Yedidia et al. (2001a) and using their terminology, the definition of the free energy is given as follows,

$$\mathcal{F}_\beta = \sum_r \sum_{\boldsymbol{x}_r} b_r(\boldsymbol{x}_r) \ln \frac{b_r(\boldsymbol{x}_r)}{\exp[h_i x_i + h_j x_j + c_r(\boldsymbol{x})]} - \sum_i (l_i - 1) \sum_{x_i} b_i(x_i) \ln \frac{b_i(x_i)}{\exp(h_i x_i)}.$$

Here, $\boldsymbol{x}_r$ denotes the pair of vertices which is included in the edge r, $b_i(x_i)$ and $b_r(\boldsymbol{x}_r)$ are a belief and a pairwise belief respectively, and l_i is the number of neighbors of vertex i. From its definition, $\sum_{x_i} b_i(x_i) = 1$, and $\sum_{\boldsymbol{x}_r} b_r(\boldsymbol{x}_r) = 1$ is satisfied. In information geometrical formulation,

$$b_r(\boldsymbol{x}_r) = p_r(\boldsymbol{x}_r; \boldsymbol{\zeta}_r).$$

And by setting

$$p_r(x_k; \boldsymbol{\zeta}_r) = p_0(x_k; \boldsymbol{\theta}), \quad k \notin r\text{-th edge},$$

the Bethe free energy becomes

$$\mathcal{F}_\beta = \sum_r [\boldsymbol{\zeta}_r \cdot \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r) - \psi_r(\boldsymbol{\zeta}_r)] - (L-1)[\boldsymbol{\theta} \cdot \boldsymbol{\eta}_0(\boldsymbol{\theta}) - \psi_0(\boldsymbol{\theta})]. \tag{4.1}$$

In Yedidia et al. (2001a,b), the following reducibility conditions (also called the marginalization conditions) are further imposed,

$$b_i(x_i) = \sum_{x_j} b_{ij}(x_i, x_j), \quad b_j(x_j) = \sum_{x_i} b_{ij}(x_i, x_j). \tag{4.2}$$

These conditions are equivalent to the m–condition in equation 3.1, that is, $\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r) = \boldsymbol{\eta}_0(\boldsymbol{\theta})$, $r = 1, \cdots, L$, so that every $\boldsymbol{\zeta}_r$ is no more an independent variable but is dependent on $\boldsymbol{\theta}$. With these constraints, the Bethe free energy is simplified as follows,

$$\mathcal{F}_{\beta m}(\boldsymbol{\theta}) = (L-1)\psi_0(\boldsymbol{\theta}) - \sum_r \psi_r(\boldsymbol{\zeta}_r(\boldsymbol{\theta})) + \Big[\sum_r \boldsymbol{\zeta}_r(\boldsymbol{\theta}) - (L-1)\boldsymbol{\theta}\Big] \cdot \boldsymbol{\eta}_0(\boldsymbol{\theta}). \tag{4.3}$$

We have to note that at each step of the BP algorithm, equation 4.2 is not satisfied, but the e–condition is satisfied. Therefore assuming equation 4.2 for original BP immediately gives the equilibrium, and no free parameter is left. Without any free parameter, it is not possible to take the derivative, which does not allow us to give any further analysis in terms of the Bethe free energy. Thus, it is important to specify, in any analysis based on the free energy, what are the independent variables and what are not, in order for a proper argument.

Finally, we mention the relation between the Bethe free energy and the conventional (Helmholtz) free energy ψ_q, logarithm of the partition function of $q(\boldsymbol{x})$ defined in equation 2.1. When the e–condition is satisfied, $\mathcal{F}_{\beta m}(\boldsymbol{\theta})$ becomes

$$\mathcal{F}_{\beta m}(\boldsymbol{\theta}) = (L-1)\psi_0(\boldsymbol{\theta}) - \sum_r \psi_r(\boldsymbol{\zeta}_r(\boldsymbol{\theta})) = -\Big\{\psi_0(\boldsymbol{\theta}) + \sum_r [\psi_r(\boldsymbol{\zeta}_r(\boldsymbol{\theta})) - \psi_0(\boldsymbol{\theta})]\Big\}.$$

This formula shows that the Bethe free energy can be regarded as an approximation to the conventional free energy by a linear combination of ψ_0 and $\{\psi_r\}$. Moreover, if the graph is tree, the result of Proposition 1 shows that the Bethe free energy is equivalent to $-\psi_q$.

4.2 A New View on Free Energy

Instead of assuming equation 4.2, let us start from the free energy defined in equation 4.3 without any constraint on the parameters, that is, all of $\boldsymbol{\theta}, \boldsymbol{\zeta}_1, \cdots, \boldsymbol{\zeta}_L$ are the free parameters,

$$\mathcal{F}(\boldsymbol{\theta}, \boldsymbol{\zeta}_1, \cdots, \boldsymbol{\zeta}_L) = (L-1)\psi_0(\boldsymbol{\theta}) - \sum_r \psi_r(\boldsymbol{\zeta}_r) + \left[\sum_r \boldsymbol{\zeta}_r - (L-1)\boldsymbol{\theta}\right] \cdot \boldsymbol{\eta}_0(\boldsymbol{\theta}). \quad (4.4)$$

The above function is rewritten in terms of the KL–divergence as,

$$\mathcal{F}(\boldsymbol{\theta}, \boldsymbol{\zeta}_1, \cdots, \boldsymbol{\zeta}_L) = D[p_0(\boldsymbol{x}; \boldsymbol{\theta}); q(\boldsymbol{x})] - \sum_r D[p_0(\boldsymbol{x}; \boldsymbol{\theta}); p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r)] + C,$$

where C is a constant. The following theorem is easily derived.

Theorem 3. *The equilibrium $(\boldsymbol{\theta}^*, \boldsymbol{\zeta}_r^*)$ of BP is a critical point of $\mathcal{F}(\boldsymbol{\theta}, \boldsymbol{\zeta}_1, \cdots, \boldsymbol{\zeta}_r)$.*

Proof. By calculating

$$\frac{\partial \mathcal{F}}{\partial \boldsymbol{\zeta}_r} = \mathbf{0},$$

we easily have

$$\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r) = \boldsymbol{\eta}_0(\boldsymbol{\theta}),$$

which is the *m*–condition. By calculating

$$\frac{\partial \mathcal{F}}{\partial \boldsymbol{\theta}} = \mathbf{0}, \quad (4.5)$$

we are led to the *e*–condition $(L-1)\boldsymbol{\theta} = \sum_r \boldsymbol{\zeta}_r$. □

The theorem shows that equation 4.4 works as the free energy function without giving any constraint.

4.3 Relation to Other Free Energies

The function $\mathcal{F}(\boldsymbol{\theta}, \boldsymbol{\zeta}_1, \cdots, \boldsymbol{\zeta}_L)$ works as a free energy, but it is also important to compare it with other "free energies." First, we compare it with the one proposed by Kabashima & Saad (2001). It is a function of $(\boldsymbol{\zeta}_1, \cdots, \boldsymbol{\zeta}_L)$ and $(\boldsymbol{\xi}_1, \cdots, \boldsymbol{\xi}_L)$, given by

$$\mathcal{F}_{KS}(\boldsymbol{\zeta}_1, \cdots, \boldsymbol{\zeta}_L; \boldsymbol{\xi}_1, \cdots, \boldsymbol{\xi}_L) = \mathcal{F}(\boldsymbol{\theta}, \boldsymbol{\zeta}_1, \cdots, \boldsymbol{\zeta}_L) + \sum_r D[p_0(\boldsymbol{x}; \boldsymbol{\theta}); p_0(\boldsymbol{x}; \boldsymbol{\zeta}_r + \boldsymbol{\xi}_r)],$$

where $\boldsymbol{\theta} = \sum_r \boldsymbol{\xi}_r$. It is clear from the definition that the choice of $\boldsymbol{\xi}_r$ which makes $\mathcal{F}_{KS}$ minimum is $\boldsymbol{\xi}_r = \boldsymbol{\theta} - \boldsymbol{\zeta}_r$, for all r, and $\mathcal{F}_{KS}$ becomes equivalent to $\mathcal{F}$.

Next, we consider the dual form of the free energy $\mathcal{F}_\beta$ in equation 4.1. The dual form is defined by introducing the Lagrange multipliers (Yedidia et al. (2001a)), and redefining the free energy as a function of them. The multipliers are defined on the reducibility conditions, $b_i(x_i) = \sum_{x_j} b_{ij}(x_i, x_j)$ and $b_j(x_j) = \sum_{x_i} b_{ij}(x_i, x_j)$. They are equivalent to $\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r) = \boldsymbol{\eta}_0(\boldsymbol{\theta})$, which is the m–condition in information geometrical formulation. Let $\boldsymbol{\lambda}_r \in \Re^n$, $r = 1, \cdots, L$ be, the Lagrange multipliers, and the free energy becomes

$$\mathcal{G}(\boldsymbol{\theta}, \{\boldsymbol{\zeta}_r\}, \{\boldsymbol{\lambda}_r\}) = \mathcal{F}_\beta(\boldsymbol{\theta}, \{\boldsymbol{\zeta}_r\}) - \sum_r \boldsymbol{\lambda}_r \cdot [\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r) - \boldsymbol{\eta}_0(\boldsymbol{\theta})], \quad \boldsymbol{\lambda}_r \in \Re^n.$$

The original extremal problem is equivalent to the extremal problem of $\mathcal{G}$ with respect to $\boldsymbol{\theta}$, $\{\boldsymbol{\zeta}_r\}$, and $\{\boldsymbol{\lambda}_r\}$. The dual form $\mathcal{G}_\beta$ is derived by redefining $\mathcal{G}$ as a function of $\{\boldsymbol{\lambda}_r\}$, where the extremal problem of $\boldsymbol{\theta}$ and $\{\boldsymbol{\zeta}_r\}$ are solved. By solving $\partial_{\boldsymbol{\theta}} \mathcal{G} = \mathbf{0}$, we have

$$\boldsymbol{\theta}(\{\boldsymbol{\lambda}_r\}) = \frac{1}{L-1} \sum_r \boldsymbol{\lambda}_r,$$

while $\partial_{\boldsymbol{\zeta}_r} \mathcal{G} = \mathbf{0}$ gives

$$\boldsymbol{\zeta}_r(\boldsymbol{\lambda}_r) = \boldsymbol{\lambda}_r.$$

Finally the dual form $\mathcal{G}_\beta$ becomes

$$\mathcal{G}_\beta(\{\boldsymbol{\lambda}_r\}) = (L-1)\psi_0(\boldsymbol{\theta}(\{\boldsymbol{\lambda}_r\})) - \sum_r \psi_r(\boldsymbol{\zeta}_r(\boldsymbol{\lambda}_r)). \tag{4.6}$$

Although $\mathcal{F}$ in equation 4.4 becomes equivalent to $\mathcal{G}_\beta$ by assuming the e–condition, $\mathcal{F}$ is free from the e– and the m–conditions, and is different from $\mathcal{G}_\beta$.

From the definition of the Lagrange multipliers, $\mathcal{G}_\beta$ is introduced to analyze the extremal problem of $\mathcal{F}_\beta$ under the m–condition, where the e–condition is not satisfied. The m–constraint free energy $\mathcal{F}_{\beta m}$ in equation 4.3 shows $\mathcal{F}$ is equivalent to $\mathcal{F}_\beta$ under the m–condition.

Finally we summarize as follows: Under the m–condition, $\mathcal{F}$ is equivalent to $\mathcal{F}_\beta$ and under the e–condition $\mathcal{F}$ is equivalent to the dual form $\mathcal{G}_\beta$.

4.4 Property of Fixed Points

Let us study the stability of the fixed point of $\mathcal{F}_\beta$ or equivalently $\mathcal{F}$ under the m–condition. Since the m–condition is satisfied, every $\boldsymbol{\zeta}_r$ is a dependent variable of $\boldsymbol{\theta}$, and we consider the derivative with respect to $\boldsymbol{\theta}$. From the m–condition, we have

$$\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r) = \boldsymbol{\eta}_0(\boldsymbol{\theta}), \quad \frac{\partial \boldsymbol{\zeta}_r}{\partial \boldsymbol{\theta}} = I_r^{-1}(\boldsymbol{\zeta}_r) I_0(\boldsymbol{\theta}), \quad r = 1, \cdots, L. \tag{4.7}$$

Here, $I_0(\boldsymbol{\theta})$ and $I_r(\boldsymbol{\zeta}_r)$ are the Fisher information matrices of $p_0(\boldsymbol{x}; \boldsymbol{\theta})$ and $p_r(\boldsymbol{x}; \boldsymbol{\zeta}_r)$, respectively, which are defined as

$$I_0(\boldsymbol{\zeta}_r) = \partial_{\boldsymbol{\theta}} \boldsymbol{\eta}_0(\boldsymbol{\theta}) = \partial_{\boldsymbol{\theta}}^2 \psi_0(\boldsymbol{\theta}), \quad I_r(\boldsymbol{\zeta}_r) = \partial_{\boldsymbol{\zeta}_r} \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r) = \partial_{\boldsymbol{\zeta}_r}^2 \psi_r(\boldsymbol{\zeta}_r), \quad r = 1, \cdots, L.$$

Equation 4.7 is proved as follows,

$$\begin{aligned} \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r) + I_r(\boldsymbol{\zeta}_r)\delta\boldsymbol{\zeta}_r \simeq \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r + \delta\boldsymbol{\zeta}_r) &= \boldsymbol{\eta}_0(\boldsymbol{\theta} + \delta\boldsymbol{\theta}) \simeq \boldsymbol{\eta}_0(\boldsymbol{\theta}) + I_0(\boldsymbol{\theta})\delta\boldsymbol{\theta} \\ \delta\boldsymbol{\zeta}_r &= I_r(\boldsymbol{\zeta}_r)^{-1} I_0(\boldsymbol{\theta})\delta\boldsymbol{\theta}. \end{aligned} \tag{4.8}$$

The condition of the equilibrium is equation 4.5 which yields the e–condition, and the second derivative gives the property around the stationary point, that is

$$\frac{\partial^2 \mathcal{F}}{\partial \boldsymbol{\theta}^2} = I_0(\boldsymbol{\theta}) + I_0(\boldsymbol{\theta}) \sum_r \Big[I_r(\boldsymbol{\zeta}_r)^{-1} - I_0(\boldsymbol{\theta})^{-1} \Big] I_0(\boldsymbol{\theta}) + \Delta, \tag{4.9}$$

where, Δ is the term related to the derivative of the Fisher information matrix, which vanishes when the e–condition is satisfied.

If equation 4.9 is positive definite at the stationary point, the Bethe free energy is at least locally minimized at the equilibrium. But it is not always positive definite. Therefore, the conventional gradient descent method of $\mathcal{F}_\beta$ or $\mathcal{F}$ may fail.

5 Algorithms and Their Convergences

5.1 e–constraint Algorithm

Since the equilibrium of BP is characterized with the e– and the m–conditions, there are two possible algorithms for finding the equilibrium. One is to constrain the parameters always to satisfy the e–condition, and search for the parameters which satisfy the m–condition (e–constraint algorithm), the other is to constrain the parameters to satisfy the m–condition, and search for the parameters which satisfy the e–condition (m–constraint algorithm).

In this section, we discuss e–constraint algorithms. BP is an e–constraint algorithm since the e–condition is satisfied at each step, but its convergence is not necessarily guaranteed. We give an alternate of the e–constraint algorithm which has a better convergence property. Let us begin with proposing a new cost function as

$$\mathcal{F}_e(\{\boldsymbol{\zeta}_r\}) = \sum_r ||\boldsymbol{\eta}_0(\boldsymbol{\theta}) - \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r)||^2, \tag{5.1}$$

under the e–constraint $\boldsymbol{\theta} = \sum_r \boldsymbol{\zeta}_r/(L-1)$. If the cost function is minimized to 0, the m–condition is satisfied, and it is an equilibrium. A naive method to minimize $\mathcal{F}_e$ is the gradient descent algorithm. The gradient is

$$\frac{\partial \mathcal{F}_e}{\partial \boldsymbol{\zeta}_r} = -2I_r(\boldsymbol{\zeta}_r)[\boldsymbol{\eta}_0(\boldsymbol{\theta}) - \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r)] + \frac{2}{L-1} I_0(\boldsymbol{\theta}) \sum_r [\boldsymbol{\eta}_0(\boldsymbol{\theta}) - \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r)]. \tag{5.2}$$

If the derivative is available, $\boldsymbol{\zeta}_r$ and $\boldsymbol{\theta}$ are updated as,

$$\boldsymbol{\zeta}_r^{t+1} = \boldsymbol{\zeta}_r^t - \delta \frac{\partial \mathcal{F}_e}{\partial \boldsymbol{\zeta}_r^t}, \quad \boldsymbol{\theta}^{t+1} = \frac{1}{L} \sum_r \boldsymbol{\zeta}_r^{t+1},$$

where δ is a small positive learning rate. It is not difficult to calculate $\boldsymbol{\eta}_0(\boldsymbol{\theta})$, $\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r)$, and $I_0(\boldsymbol{\theta})$, and the rest of the problem is to calculate the first term of equation 5.2. Fortunately, we have the relation,

$$I_r(\boldsymbol{\zeta}_r)\boldsymbol{h} = \lim_{\alpha \to 0} \frac{\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r + \alpha \boldsymbol{h}) - \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r)}{\alpha}.$$

If $(\boldsymbol{\eta}_0(\boldsymbol{\theta}) - \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r))$ is substituted for $\boldsymbol{h}$, this becomes the first term of equation 5.2. Now, we propose a new algorithm.

A new e–constraint algorithm

1. *Set* $t = 0$, $\boldsymbol{\theta}^t = \mathbf{0}$, $\boldsymbol{\zeta}_r^t = \mathbf{0}$, $r = 1, \cdots, L$.

2. *Calculate* $\boldsymbol{\eta}_0(\boldsymbol{\theta}^t)$, $I_0(\boldsymbol{\theta}^t)$, *and* $\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r^t)$, $r = 1, \cdots, L$.

3. *Let* $\boldsymbol{h}_r = \boldsymbol{\eta}_0(\boldsymbol{\theta}^t) - \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r^t)$ *and calculate* $\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r^t + \alpha \boldsymbol{h}_r)$ *for* $r = 1, \cdots, L$, *where* $\alpha > 0$ *is small. Then calculate*

$$\boldsymbol{g}_r = \frac{\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r^t + \alpha \boldsymbol{h}) - \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r^t)}{\alpha}.$$

4. *For* $t = 1, 2, \cdots$, *update* $\boldsymbol{\zeta}_r^{t+1}$ *as follows,*

$$\begin{aligned} \boldsymbol{\zeta}_r^{t+1} &= \boldsymbol{\zeta}_r^t - \delta\Big[-2\boldsymbol{g}_r + \frac{2}{L-1} I_0(\boldsymbol{\theta}^t) \sum_r \boldsymbol{h}_r\Big], \\ \boldsymbol{\theta}^{t+1} &= \frac{1}{L-1} \sum_r \boldsymbol{\zeta}_r^{t+1}. \end{aligned}$$

5. *If* $\mathcal{F}_e(\{\boldsymbol{\zeta}_r\}) = \sum_r ||\boldsymbol{\eta}_0(\boldsymbol{\theta}) - \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r)||^2 > \epsilon$ *(*ϵ *is a threshold) holds,* $t+1 \to t$ *and go to 2.*

This algorithm is an *e*–constraint algorithm, and does not include double loops, which is similar to the BP algorithm, but we have introduced a new parameter α which can affect the convergence. We have checked, with small-sized numerical simulations, that if α is sufficiently small, this problem can be avoided, but further theoretical analysis is needed. Another problem is that this algorithm converges to any fixed point of BP, even if it is not a stable fixed point of BP. For example, when $\boldsymbol{\zeta}_r$ and $\boldsymbol{\theta}$ are extremely large, eventually every component of $\boldsymbol{\eta}_r$ and $\boldsymbol{\eta}_0$ becomes close to 1, which is a trivial useless fixed point of this algorithm. In order to avoid this, it is natural to use the Riemannian metric for the norm, instead of the square norm defined in equation 5.1. The local metric modifies the cost function to

$$\mathcal{F}_{eR}(\{\boldsymbol{\zeta}_r\}) = \sum_r [\boldsymbol{\eta}_0(\boldsymbol{\theta}) - \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r)]^T I_0(\boldsymbol{\theta}_0)^{-1} [\boldsymbol{\eta}_0(\boldsymbol{\theta}) - \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r)],$$

where $\boldsymbol{\theta}_0$ is the convergent point. Since $I_0(\boldsymbol{\theta}_0)^{-1}$ diverges at trivial fixed points mentioned above, we expect $\mathcal{F}_{eR}(\{\boldsymbol{\zeta}_r\})$ to be a better cost function. The gradient can be calculated similarly by fixing $\boldsymbol{\theta}_0$, which is unknown. Hence, we replace it by $\boldsymbol{\theta}^t$. The calculation of $\boldsymbol{g}_r$ should also be modified to

$$\tilde{\boldsymbol{g}}_r = \frac{\boldsymbol{\eta}_r(\boldsymbol{\zeta}_r^t + \alpha I_0(\boldsymbol{\theta}^t)^{-1} \sum_r \boldsymbol{h}_r)}{\alpha}$$

from the point of view of the natural gradient method (Amari (1998)). We finally have

$$\zeta_r^{t+1} = \zeta_r^t - 2\delta I_0(\theta^t)^{-1}\Big[-\tilde{g}_r + \frac{1}{L-1}\sum_r h_r\Big].$$

Since $I_0(\theta)$ is a diagonal matrix, computation is simple.

5.2 m–constraint Algorithm

The other possibility is to constrain the parameters always to satisfy the m–condition, and modify the parameters to satisfy the e–condition. Since the m–condition is satisfied, $\{\zeta_r\}$ are dependent on θ.

A naive idea is to repeat the following two steps,

Naive m–constraint algorithm

1. *For $r = 1, \cdots, L$,*
$$\zeta_r^t = \pi_{M_r} \circ p_0(x; \theta^t). \tag{5.3}$$
2. *Update the parameters as*
$$\theta^{t+1} = L\theta^t - \sum_r \zeta_r^t.$$

Starting from θ^t, the algorithm finds $\{\zeta_r^{t+1}\}$ that satisfies the m–condition by equation 5.3, and θ^{t+1} is adjusted to satisfy the e–condition.

This is a simple recursive algorithm without double loops. We call it the naive m–constraint algorithm. One may use an advanced iteration method that uses, instead of ζ_r^t, new ζ_r^{t+1}. In this case, the algorithm is

$$\zeta_r^{t+1} = \pi_{M_r} \circ p_0(x; \theta^{t+1}), \quad \text{where} \quad \theta^{t+1} = L\theta^t - \sum_r \zeta_r^{t+1}.$$

In this algorithm, starting from θ^t, one should solve a non-linear equation in θ^{t+1}, because $\{\zeta_r^{t+1}\}$ are functions of θ^{t+1}. This algorithm therefore uses double loops, the inner loop and the outer loop. This is the idea of CCCP, and it is also an m–constraint algorithm.

Stability of the algorithms

Although the naive m–constraint algorithm and CCCP share the same equilibrium $\boldsymbol{\theta}^*$ and $\{\boldsymbol{\zeta}_r^*\}$, their local stabilities at the equilibrium are different. It is reported that CCCP has superior properties in this respect. The local stability of BP was analyzed by Richardson (2000) and also by Ikeda et al. (2004) in geometrical terms. The stability condition of BP is given by the conditions of the eigen values of a matrix defined by the Fisher information matrices. In this paper, we give the local stability of the other algorithms.

If we eliminate the intermediate variables $\{\boldsymbol{\zeta}_r\}$ in the inner loop, the naive m–constraint algorithm is

$$\boldsymbol{\theta}^{t+1} = L\boldsymbol{\theta}^t - \sum_r \pi_{M_r} \circ p_0(\boldsymbol{x}; \boldsymbol{\theta}^t), \tag{5.4}$$

and CCCP is represented as

$$\boldsymbol{\theta}^{t+1} = L\boldsymbol{\theta}^t - \sum_r \pi_{M_r} \circ p_0(\boldsymbol{x}; \boldsymbol{\theta}^{t+1}). \tag{5.5}$$

In order to derive the variational equation at the equilibrium, we note that, for the m–projection

$$\boldsymbol{\zeta}_r = \pi_{M_r} \circ p_0(\boldsymbol{x}; \boldsymbol{\theta}),$$

a small perturbation $\delta\boldsymbol{\theta}$ in $\boldsymbol{\theta}$ is updated as

$$\delta\boldsymbol{\zeta}_r = I_r(\boldsymbol{\zeta}_r)^{-1} I_0(\boldsymbol{\theta})\delta\boldsymbol{\theta},$$

(see equation 4.8). The variational equations are hence for equation 5.4,

$$\delta\boldsymbol{\theta}^{t+1} = \Big[LE - \sum_r I_r(\boldsymbol{\zeta}_r)^{-1} I_0(\boldsymbol{\theta})\Big]\delta\boldsymbol{\theta}^t,$$

where E is the identity matrix, and for equation 5.5,

$$\delta\boldsymbol{\theta}^{t+1} = L\Big[E + \sum_r I_r(\boldsymbol{\zeta}_r)^{-1} I_0(\boldsymbol{\theta})\Big]^{-1}\delta\boldsymbol{\theta}^t,$$

respectively. Let K be a matrix defined by

$$K = \frac{1}{L}\sum_r \sqrt{I_0(\boldsymbol{\theta})} I_r(\boldsymbol{\zeta}_r)^{-1} \sqrt{I_0(\boldsymbol{\theta})},$$

and $\delta\tilde{\boldsymbol{\theta}}^t$ be a new variable defined as

$$\delta\tilde{\boldsymbol{\theta}}^t = \sqrt{I_0(\boldsymbol{\theta})}\delta\boldsymbol{\theta}^t.$$

The variational equations for equations 5.4 and 5.5 are then

$$\begin{aligned}\delta\tilde{\boldsymbol{\theta}}^{t+1} &= L(E - LK)\delta\tilde{\boldsymbol{\theta}}^{t}, \\ \delta\tilde{\boldsymbol{\theta}}^{t+1} &= L(E + LK)^{-1}\delta\tilde{\boldsymbol{\theta}}^{t},\end{aligned}$$

respectively.

The equilibrium is stable when the absolute values of the eigenvalues of the respective coefficient matrices are smaller than 1. Let $\lambda_1, \cdots, \lambda_n$ be the eigenvalues of K. They are all real and positive, since K is a symmetric positive-definite matrix. We note that λ_i are close to 1, when $I_r(\boldsymbol{\zeta}_r) \approx I_0(\boldsymbol{\theta})$ or M_r is close to M_0. The following theorem shows CCCP has a good convergent property.

Theorem 4. *The equilibrium of the naive m–constraint algorithm in equation 5.4 is stable when*

$$1 + \frac{1}{L} > \lambda_i > 1 - \frac{1}{L}, \quad i = 1, \cdots, n.$$

The equilibrium of CCCP is stable when the eigen values of K satisfies

$$\lambda_i > 1 - \frac{1}{L}, \quad i = 1, \cdots, n. \tag{5.6}$$

Under the m–constraint, the Hessian of $\mathcal{F}(\boldsymbol{\theta})$ at an equilibrium point is equal to (cf. equation 4.9)

$$\sqrt{I_0(\boldsymbol{\theta})}\big[LK - (L-1)E\big]\sqrt{I_0(\boldsymbol{\theta})},$$

so that the stability condition (equation 5.6) for CCCP is equivalent to the condition that the equilibrium is a local minimum of $\mathcal{F}$ under the m–constraint which is equivalent to the m–constraint Bethe free energy $\mathcal{F}_{\beta m}(\boldsymbol{\theta})$. The theorem therefore states that CCCP is locally stable around an equilibrium if and only if the equilibrium is a local minimum of $\mathcal{F}_{\beta m}(\boldsymbol{\theta})$, whereas the naive m–constraint algorithm is not necessarily stable even if the equilibrium is a local minimum. A similar result is obtained in Heskes (2003).

It should be noted that the above local stability result for CCCP does not follow from the global convergence result given by Yuille (2002). Yuille has shown that CCCP decreases the cost function and converges to an extremal point of $\mathcal{F}_{\beta m}(\boldsymbol{\theta})$ which means the fixed point is not necessarily a local minimum, but can be a saddle point. Our local linear analysis shows a stable fixed point of CCCP is a local minimum of $\mathcal{F}_{\beta m}(\boldsymbol{\theta})$.

Natural gradient and discretization

Let us consider a gradient rule for updating $\boldsymbol{\theta}$ to find a minimum of $\mathcal{F}$ under the m–condition,

$$\dot{\boldsymbol{\theta}} = -\frac{\partial \mathcal{F}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}.$$

When we have a metric to measure the distance in the space of $\boldsymbol{\theta}$, it is natural to use the metric for gradient (natural gradient, see Amari (1998)). For statistical models, the Riemannian metric given by the Fisher information matrix is a natural choice, since it is derived from KL-divergence. The natural gradient version of the update rule is

$$\dot{\boldsymbol{\theta}} = -I_0^{-1}(\boldsymbol{\theta})\frac{\partial \mathcal{F}}{\partial \boldsymbol{\theta}} = (L-1)\boldsymbol{\theta} - \sum_r \pi_{M_r} \circ p_0(\boldsymbol{x};\boldsymbol{\theta}). \tag{5.7}$$

For the implementation, it is necessary to discretize the continuous-time update rule. The "fully explicit" scheme of discretization (Euler's method) reads

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t + \Delta t\Big[(L-1)\boldsymbol{\theta}^t - \sum_r \pi_{M_r} \circ p_0(\boldsymbol{x};\boldsymbol{\theta}^t)\Big]. \tag{5.8}$$

When $\Delta t = 1$, this is equivalent to the naive m–constraint algorithm (equation 5.4). However, we do not necessarily have to let $\Delta t = 1$: Instead, we may use arbitrary positive value for Δt. We will show how the convergence rate will be affected by the change of Δt later.

The "fully implicit" scheme yields

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t + \Delta t\big[(L-1)\boldsymbol{\theta}^{t+1} - \sum_r \pi_{M_r} \circ p_0(\boldsymbol{x};\boldsymbol{\theta}^{t+1})\big], \tag{5.9}$$

which, after rearrangement of terms, becomes

$$\big[1 - \Delta t(L-1)\big]\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t - \Delta t \sum_r \pi_{M_r} \circ p_0(\boldsymbol{x};\boldsymbol{\theta}^{t+1}).$$

When $\Delta t = 1/L$, this equation is equivalent to CCCP in equation 5.5. Again, we do not have to be bound to the choice $\Delta t = 1/L$. We will also show the relation between Δt and the convergence rate later.

We have just shown that the naive m–constraint algorithm and CCCP can be viewed as first-order methods of discretization applied to the continuous-time natural gradient system shown in equation 5.7. The local stability result for CCCP proved in theorem 4 can also be understood as an example of the well-known

absolute stability property of the fully-implicit scheme applied to linear systems. It should also be noted that other more sophisticated methods for solving ordinary differential equations, such as Runge-Kutta methods (possibly with adaptive step-size control), the Bulirsch-Stoer method, and so on (Press et al. (1992)), are applicable to formulate m–constraint algorithms with better properties, for example, better stability. In this paper, however, we do not discuss possible extension along this line any further.

Acceleration of m–constraint algorithms

We give the analysis of equations 5.8 and 5.9 in this section.

The variational equation for equation 5.8 is

$$\delta\tilde{\boldsymbol{\theta}}^{t+1} = \big\{E - [LK - (L-1)E]\Delta t\big\}\delta\tilde{\boldsymbol{\theta}}^{t}.$$

Let

$$\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n \tag{5.10}$$

be the eigenvalues of K. Then, the convergence rate is improved by choosing an adequate Δt. The convergence rate is governed by the largest absolute values of the eigenvalues of $E - \big[LK - (L-1)E\big]\Delta t$, which are given by

$$\mu_i = 1 - \big[L\lambda_i - (L-1)\big]\Delta t.$$

From equation 5.10 we have $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_n$. The stability condition is $|\mu_i| < 1$ for all i. At a locally stable equilibrium point, $\mu_1 < 1$ always holds, so that the algorithm is stable if $\mu_n > -1$ holds. The convergence to a locally stable equilibrium point is most accelerated when $\mu_1 + \mu_n = 0$, which holds by taking

$$\Delta t_{\text{opt}} = \frac{2}{L(\lambda_1 + \lambda_n - 1) + 2}.$$

The variational equation for equation 5.9 is

$$\delta\tilde{\boldsymbol{\theta}}^{t+1} = \big\{E + [LK - (L-1)E]\Delta t\big\}^{-1}\delta\tilde{\boldsymbol{\theta}}^{t},$$

and the convergence rate is governed by the largest of the absolute values of the eigenvalues of $\{E + [LK - (L-1)E]\Delta t\}^{-1}$, which should be smaller than 1 for

convergence. The eigenvalues are

$$\mu_i = \frac{1}{1 + \left[L\lambda_i - (L-1)\right]\Delta t}.$$

We again have $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_n$. At a locally stable equilibrium point, $0 < \mu_n$ and $\mu_1 < 1$ always hold, so that the algorithm is always stable. In principle, the smaller μ_1 becomes, the faster the algorithm converges, so that taking $\Delta t \to +\infty$ yields the fastest convergence. However, the algorithm in this limit reduces to the direct evaluation of the e–condition under the m–constraint with one update step of the parameters. This is the fastest if it is possible, but this is usually infeasible for loopy graphs.

6 Extension

6.1 Extend the Framework to Wider Class of Distributions

In this section, two important extensions of BP is given in the information geometrical framework. First, we extend the model to the case where the marginal distribution of each vertex is an exponential family. A similar extension is given in Wainwright et al. (2003).

Let $\boldsymbol{t}_i$ be the sufficient statistics of the marginal distribution of x_i, that is, $q(x_i)$. The marginal distribution is in the family of distributions defined as follows

$$p(x_i; \boldsymbol{\theta}_i) = \exp[\boldsymbol{\theta}_i \cdot \boldsymbol{t}_i - \varphi_i(\boldsymbol{\theta}_i)].$$

This includes many important distributions. For example, multinomial distribution and Gaussian distribution are included in this family.

Let us define $\boldsymbol{t} = (\boldsymbol{t}_1^T, \cdots, \boldsymbol{t}_n^T)^T$ and $\boldsymbol{\theta} = (\boldsymbol{\theta}_1^T, \cdots, \boldsymbol{\theta}_n^T)^T$, and let the true distribution be

$$q(\boldsymbol{x}) = \exp[\boldsymbol{h} \cdot \boldsymbol{t} + \boldsymbol{c}(\boldsymbol{x}) - \psi_q].$$

We can now redefine equation 2.2 as follows,

$$p(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{v}) = \exp[\boldsymbol{\theta} \cdot \boldsymbol{t} + \boldsymbol{v} \cdot \boldsymbol{c}(\boldsymbol{x}) - \psi(\boldsymbol{\theta}, \boldsymbol{v})],$$

and S in equation 2.3 as

$$S = \left\{ p(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{v}) \;\middle|\; \boldsymbol{\theta} \in \Theta, \boldsymbol{v} \in \mathcal{V} \right\}.$$

When the problem is to infer the marginal distribution $q(x_i)$ of $q(\boldsymbol{x})$, we can redefine the BP algorithm in this new S, by redefining M_0 and M_r. This extension based on the new definition is simple, and we do not give further details in this article.

6.2 Generalized Belief Propagation

In this section, we show the information geometrical framework for the general belief propagation (GBP) (Yedidia et al. (2001b)), which is an important extension of BP.

A naive explanation of GBP is that the cliques are reformulated by subsets of $\mathcal{L}$, which is the set of all the edges. This brings us a new implementation of the algorithm and different inference. In information geometrical formulation, we define $c'_s(\boldsymbol{x})$ as a new clique function, which summarizes the interactions of the edges in $\mathcal{L}_s$, $s = 1, \cdots, L'$, that is,

$$c'_s(\boldsymbol{x}) = \sum_{r \in \mathcal{L}_s} c_r(\boldsymbol{x}),$$

where $\mathcal{L}_s \subseteq \mathcal{L}$. Those $\mathcal{L}_s$ may have overlaps, and $\mathcal{L}_s$ must be chosen to satisfy $\cup_s \mathcal{L}_s = \mathcal{L}$.

GBP is a general framework, which includes a lot of possible cases. We categorize them into three important classes, and give an information geometrical framework for them

Case 1

In the simplest case, each $\mathcal{L}_s$ does not have any loop. This is equivalent to TRP. As we have seen in section 3.2, the algorithm is explained in information geometrical framework.

Case 2

In the next case, each $\mathcal{L}_s$ can have loops, but there is no overlap, that is, $\mathcal{L}_s \cap \mathcal{L}_{s'} = \emptyset$ for $s \neq s'$. The extension to this case is also simple. We can apply information geometry by redefining M_r as M_s, where its definition is given as follow

$$M_s = \left\{ p_s(\boldsymbol{x}; \boldsymbol{\zeta}_s) = \exp[\boldsymbol{h} \cdot \boldsymbol{x} + c'_s(\boldsymbol{x}) + \boldsymbol{\zeta}_s \cdot \boldsymbol{x} - \psi_s(\boldsymbol{\zeta}_s)] \;\middle|\; \boldsymbol{\zeta}_s \in \Re^n \right\}.$$

Since some loops are treated in a different way, the result might be different from BP.

Case 3

Finally, we describe the case where each $\mathcal{L}_s$ can have loops and overlaps with the other sets. In this case we have to extend the framework. Suppose $\mathcal{L}_s$ and $\mathcal{L}_{s'}$ have an overlap, and both have loops. We explain the case with an example in Figure 4.

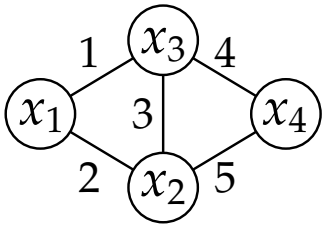

Figure 4: Case 3.

Let us first define the following distributions,

$$\begin{aligned} q(\boldsymbol{x}) &= \exp\Big[\boldsymbol{h}\cdot\boldsymbol{x} + \sum_{i=1}^{5} c_i(\boldsymbol{x}) - \psi_q\Big], \\ p_0(\boldsymbol{x};\boldsymbol{\theta}) &= \exp[\boldsymbol{h}\cdot\boldsymbol{x} + \boldsymbol{\theta}\cdot\boldsymbol{x} - \psi_0(\boldsymbol{\theta})], \qquad (6.1) \\ p_1(\boldsymbol{x};\boldsymbol{\zeta}_1) &= \exp\Big[\boldsymbol{h}\cdot\boldsymbol{x} + \sum_{i=1}^{3} c_i(\boldsymbol{x}) + \boldsymbol{\zeta}_1\cdot\boldsymbol{x} - \psi_1(\boldsymbol{\zeta}_1)\Big], \qquad (6.2) \\ p_2(\boldsymbol{x};\boldsymbol{\zeta}_2) &= \exp\Big[\boldsymbol{h}\cdot\boldsymbol{x} + \sum_{i=3}^{5} c_i(\boldsymbol{x}) + \boldsymbol{\zeta}_2\cdot\boldsymbol{x} - \psi_2(\boldsymbol{\zeta}_2)\Big]. \qquad (6.3) \end{aligned}$$

Even if $\boldsymbol{\zeta}_1$, $\boldsymbol{\zeta}_2$, and $\boldsymbol{\theta}$ satisfy the e–condition as $\boldsymbol{\theta} = \boldsymbol{\zeta}_1 + \boldsymbol{\zeta}_2$, this does not imply

$$C\frac{p_1(\boldsymbol{x};\boldsymbol{\zeta}_1)p_2(\boldsymbol{x};\boldsymbol{\zeta}_2)}{p_0(\boldsymbol{x};\boldsymbol{\theta})}$$

is equivalent to $q(\boldsymbol{x})$, since $c_3(\boldsymbol{x})$ is counted twice. Therefore, we introduce another model $p_3(\boldsymbol{x};\boldsymbol{\zeta}_3)$, which has the following form.

$$p_3(\boldsymbol{x};\boldsymbol{\zeta}_3) = \exp\Big[\boldsymbol{h}\cdot\boldsymbol{x} + c_3(\boldsymbol{x}) + \boldsymbol{\zeta}_3\cdot\boldsymbol{x} - \psi_3(\boldsymbol{\zeta}_3)\Big]. \qquad (6.4)$$

Now,

$$C\frac{p_1(\boldsymbol{x};\boldsymbol{\zeta}_1)p_2(\boldsymbol{x};\boldsymbol{\zeta}_2)}{p_3(\boldsymbol{x};\boldsymbol{\zeta}_3)}$$

becomes equal to $q(\boldsymbol{x})$ where $\boldsymbol{\zeta}_3 = \boldsymbol{\zeta}_1 + \boldsymbol{\zeta}_2$ is the e–condition.

Next we look at the m–condition. The original form of the m–condition is

$$\sum_{\boldsymbol{x}} \boldsymbol{x} p_0(\boldsymbol{x};\boldsymbol{\theta}) = \sum_{\boldsymbol{x}} \boldsymbol{x} p_s(\boldsymbol{x};\boldsymbol{\zeta}_s),$$

but, in this case, this form is not enough. We need a further condition, that is,

$$p_s(x_2, x_3;\boldsymbol{\zeta}_s) = \sum_{x_1,x_4} p_s(\boldsymbol{x};\boldsymbol{\zeta}_s)$$

should be the same for $s = \{1,2,3\}$. The models in equations 6.1, 6.2, 6.3, and 6.4 are not sufficient, since we do not have enough parameters to specify a joint distribution of (x_2, x_3), and the model must be extended. In the binary case, we can extend the models by adding one variable as follows,

$$\begin{aligned}
p_1(\boldsymbol{x};\boldsymbol{\zeta}_1, v_1) &= \exp\Big[\boldsymbol{h}\cdot\boldsymbol{x} + \sum_{i=1}^{3} c_i(\boldsymbol{x}) + \boldsymbol{\zeta}_1\cdot\boldsymbol{x} + v_1 x_2 x_3 - \psi_1(\boldsymbol{\zeta}_1, v_1)\Big],\\
p_2(\boldsymbol{x};\boldsymbol{\zeta}_2, v_2) &= \exp\Big[\boldsymbol{h}\cdot\boldsymbol{x} + \sum_{i=3}^{5} c_i(\boldsymbol{x}) + \boldsymbol{\zeta}_2\cdot\boldsymbol{x} + v_2 x_2 x_3 - \psi_2(\boldsymbol{\zeta}_2, v_2)\Big],\\
p_3(\boldsymbol{x};\boldsymbol{\zeta}_3, v_3) &= \exp[\boldsymbol{h}\cdot\boldsymbol{x} + c_3(\boldsymbol{x}) + \boldsymbol{\zeta}_3\cdot\boldsymbol{x} + v_3 x_2 x_3 - \psi_3(\boldsymbol{\zeta}_3, v_3)],
\end{aligned}$$

and the m–condition becomes,

$$\begin{aligned}
\sum_{\boldsymbol{x}} \boldsymbol{x} p_0(\boldsymbol{x};\boldsymbol{\theta}) &= \sum_{\boldsymbol{x}} \boldsymbol{x} p_s(\boldsymbol{x};\boldsymbol{\zeta}_s, v_s), \quad s = 1,2,3,\\
\sum_{\boldsymbol{x}} x_2 x_3 p_1(\boldsymbol{x};\boldsymbol{\zeta}_1, v_1) &= \sum_{\boldsymbol{x}} x_2 x_3 p_2(\boldsymbol{x};\boldsymbol{\zeta}_2, v_2) = \sum_{\boldsymbol{x}} x_2 x_3 p_3(\boldsymbol{x};\boldsymbol{\zeta}_3, v_3).
\end{aligned}$$

We revisit the e–condition, which is now extended as,

$$\boldsymbol{\zeta}_3 = \boldsymbol{\zeta}_1 + \boldsymbol{\zeta}_2, \quad v_3 = v_1 + v_2.$$

This is a simple example, but we can describe any GBP problem in the information geometrical framework in a similar way.

7 Conclusion

Stochastic reasoning is an important technique widely used for graphical models including many interesting applications. BP is a useful method to solve it, and in order to analyze its behavior and to give a theoretical foundation, a variety of approaches have been proposed from AI, statistical physics, information theory, and information geometry. We have shown a unified framework to understand various

interdisciplinary concepts and algorithms from the point of view of information geometry. Since information geometry captures the essential structure of the manifold of probability distributions, we are successful in clarifying the intrinsic geometrical structures and their difference of various algorithms proposed so far.

The BP solution is characterized with the *e*– and the *m*–conditions. We have shown that BP and TRP explore the solution in the subspace where the *e*–condition is satisfied, while CCCP does in the subspace where the *m*–condition is satisfied. This analysis makes us possible to obtain new efficient variants of these algorithms. We have proposed new *e*– and *m*–constraint algorithms. The possible acceleration methods for the *m*–constraint algorithm and CCCP are shown with local stability and convergence rate analysis. We have clarified the relation among the free-energy-like functions and have proposed a new one. Finally we have shown possible extensions of BP from information geometrical viewpoint.

This work is a first step toward information geometrical understanding of BP. By using the framework, we expect further understanding and a new improvement of the methods will emerge.

Appendix: Information Geometrical View of CCCP

In this section, we derive the information geometrical view of CCCP. The following two theorems play important roles in CCCP.

Theorem 5. *(Yuille & Rangarajan (2003) section 2) Let $E(\boldsymbol{x})$ be an energy function with bounded Hessian $\partial E(\boldsymbol{x})/\partial \boldsymbol{x} \partial \boldsymbol{x}$. Then we can always decompose it into the sum of a convex function and a concave function.*

Theorem 6. *(Yuille & Rangarajan (2003) section 2) Consider an energy function $E(\boldsymbol{x})$ (bounded below) of form $E(\boldsymbol{x}) = E_{vex}(\boldsymbol{x}) + E_{cave}(\boldsymbol{x})$ where $E_{vex}(\boldsymbol{x})$, $E_{cave}(\boldsymbol{x})$ are convex and concave functions of $\boldsymbol{x}$ respectively. Then the discrete iterative CCCP algorithm $\boldsymbol{x}^t \mapsto \boldsymbol{x}^{t+1}$ given by*

$$\nabla E_{vex}(\boldsymbol{x}^{t+1}) = -\nabla E_{cave}(\boldsymbol{x}^t)$$

is guaranteed to monotonically decrease the energy $E(\boldsymbol{x})$ as a function of time and hence to converge to a minimum or saddle point of $E(\boldsymbol{x})$ (or even a local maximum if it starts at one).

The idea of CCCP was applied to solve the inference problem of loopy graphs, where the Bethe free energy $\mathcal{F}_\beta$ in equation 4.1 is the energy function (Yuille (2002)). The concave and convex functions are defined as follows

$$\begin{aligned}
\mathcal{F}_\beta(\boldsymbol{\theta}, \{\boldsymbol{\zeta}_r\}) &= \sum_r [\boldsymbol{\zeta}_r \cdot \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r) - \psi_r(\boldsymbol{\zeta}_r)] - (L-1)[\boldsymbol{\theta} \cdot \boldsymbol{\eta}_0(\boldsymbol{\theta}) - \psi_0(\boldsymbol{\theta})] \\
&= \mathcal{F}_{vex}(\boldsymbol{\theta}, \{\boldsymbol{\zeta}_r\}) + \mathcal{F}_{cave}(\boldsymbol{\theta}, \{\boldsymbol{\zeta}_r\}), \\
\mathcal{F}_{vex}(\boldsymbol{\theta}, \{\boldsymbol{\zeta}_r\}) &= \sum_r [\boldsymbol{\zeta}_r \cdot \boldsymbol{\eta}_r(\boldsymbol{\zeta}_r) - \psi_r(\boldsymbol{\zeta}_r)] + [\boldsymbol{\theta} \cdot \boldsymbol{\eta}_0(\boldsymbol{\theta}) - \psi_0(\boldsymbol{\theta})], \\
\mathcal{F}_{cave}(\boldsymbol{\theta}) &= -L[\boldsymbol{\theta} \cdot \boldsymbol{\eta}_0(\boldsymbol{\theta}) - \psi_0(\boldsymbol{\theta})].
\end{aligned}$$

Let the m–condition be satisfied, and $\mathcal{F}_{vex}$ is a function of $\boldsymbol{\theta}$. Next, since $\boldsymbol{\eta}_0$ and $\boldsymbol{\theta}$ has a one-to-one relation, let $\boldsymbol{\eta}_0$ be the coordinate system. The gradient of $\mathcal{F}_{vex}$ and $\mathcal{F}_{cave}$ is given as follows,

$$\nabla_{\boldsymbol{\eta}_0} \mathcal{F}_{vex}(\boldsymbol{\eta}_0) = \boldsymbol{\theta} + \sum_r \boldsymbol{\zeta}_r, \quad -\nabla_{\boldsymbol{\eta}_0} \mathcal{F}_{cave}(\boldsymbol{\eta}_0) = L\boldsymbol{\theta}.$$

Finally, the CCCP algorithm is written as

$$\begin{aligned}
\nabla_{\boldsymbol{\eta}_0} \mathcal{F}_{vex}(\boldsymbol{\eta}_0^{t+1}) &= -\nabla_{\boldsymbol{\eta}_0} \mathcal{F}_{cave}(\boldsymbol{\eta}_0^t), \\
\boldsymbol{\theta}^{t+1} + \sum_r \boldsymbol{\zeta}_r^{t+1} &= L\boldsymbol{\theta}^t.
\end{aligned} \tag{A.1}$$

Since the m–condition is not satisfied in general, the inner loop solves the condition, while the outer loop updates the parameters as equation A.1.

Acknowledgment

We thank the anonymous reviewers for valuable feedback. This work was supported by the Grant-in-Aid for Scientific Research Nos. 14084208 and 14084209, MEXT, Japan and No. 14654017, JSPS, Japan.

References

Amari, S. (1998). Natural gradient works efficiently in learning. *Neural Computation*, *10*, 251–276.

Amari, S. (2001). Information geometry on hierarchy of probability distributions. *IEEE Trans. Information Theory*, *47*, 1701–1711.

Amari, S., Ikeda, S., & Shimokawa, H. (2001). Information geometry and mean field approximation: The α-projection approach. In M. Opper & D. Saad (Eds.), *Advanced Mean Field Methods – Theory and Practice* (pp. 241–257). Cambridge, MA: MIT Press.

Amari, S. & Nagaoka, H. (2000). *Methods of Information Geometry*. Providence, RI: AMS and Oxford University Press.

Heskes, T. (2003). Stable fixed points of loopy belief propagation are minima of the Bethe free energy. In S. Becker, S. Thrun, & K. Obermayer (Eds.), *Advances in Neural Information Processing Systems*, *15* (pp. 359–366). Cambridge, MA: MIT Press.

Ikeda, S., Tanaka, T., & Amari, S. (2002). Information geometrical framework for analyzing belief propagation decoder. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in Neural Information Processing Systems*, *14* (pp. 407–414). Cambridge, MA: MIT Press.

Ikeda, S., Tanaka, T., & Amari, S. (2004). Information geometry of turbo codes and low-density parity-check codes. to appear in IEEE Trans. Information Theory.

Itzykson, C. & Drouffe, J.-M. (1989). *Statistical Field Theory*, volume 1. New York: Cambridge University Press.

Jordan, M. I. (1999). *Learning in Graphical Models*. Cambridge, MA: MIT Press.

Kabashima, Y. & Saad, D. (1999). Statistical mechanics of error-correcting codes. *Europhysics Letters*, *45*, 97–103.

Kabashima, Y. & Saad, D. (2001). The TAP approach to intensive and extensive connectivity systems. In M. Opper & D. Saad (Eds.), *Advanced Mean Field Methods – Theory and Practice* (pp. 65–84). Cambridge, MA: MIT Press.

Lauritzen, S. L. & Spiegelhalter, D. J. (1988). Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society B*, *50*, 157–224.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo. CA: Morgan Kaufmann.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1992). *Numerical Recipes in C: The art of scientific computing*. Cambridge: Cambridge University Press.

Richardson, T. J. (2000). The geometry of turbo-decoding dynamics. *IEEE Trans. Information Theory*, *46*, 9–23.

Tanaka, T. (2000). Information geometry of mean-field approximation. *Neural Computation*, *12*, 1951–1968.

Tanaka, T. (2001). Information geometry of mean-field approximation. In M. Opper & D. Saad (Eds.), *Advanced Mean Field Methods – Theory and Practice* (pp. 259–273). Cambridge, MA: MIT Press.

Wainwright, M., Jaakkola, T., & Willsky, A. (2002). Tree-based reparameterization for approximate inference on loopy graphs. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in Neural Information Processing Systems*, *14* (pp. 1001–1008). Cambridge, MA: MIT Press.

Wainwright, M., Jaakkola, T., & Willsky, A. (2003). Tree-reweighted belief propagation algorithms and approximate ML estimate by pseudo-moment matching. In C. M. Bishop & B. J. Frey (Eds.), *Proceeding of Ninth International Workshop on Artificial Intelligence and Statistics*.

Weiss, Y. (2000). Correctness of local probability propagation in graphical models with loops. *Neural Computation*, *12*, 1–41.

Yedidia, J. S., Freeman, W. T., & Weiss, Y. (2001a). Bethe free energy, Kikuchi approximations, and belief propagation algorithms. Technical Report TR2001–16, Mitsubishi Electric Research Laboratories.

Yedidia, J. S., Freeman, W. T., & Weiss, Y. (2001b). Generalized belief propagation. In T. K. Leen, T. G. Dietterich, & V. Tresp (Eds.), *Advances in Neural Information Processing Systems*, *13* (pp. 689–695). Cambridge, MA: MIT Press.

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

Yuille, A. L. & Rangarajan, A. (2003). The concave-convex procedure. *Neural Computation*, *15*, 915–936.