A Family of CCCP Algorithms which minimize the TRW Free Energy

Yu NISHIYAMA

Laboratory for Integrated Theoretical Neuroscience,
RIKEN Brain Science Institute
Hirosawa 2-1, Wako, Saitama 351-0198, Japan

ynishiam@gmail.com

Xingyao YE and Alan L. YUILLE

Department of Statistics,
University of California, Los Angeles (UCLA)
Los Angeles, CA 90095-1554

{yexy,yuille}@stat.ucla.edu

Received 29 January 2011

Abstract We propose a family of convergent double-loop algorithms which minimize the TRW free energy. These algorithms are based on the concave convex procedure (CCCP) so we call them TRW-CCCP. Our formulation includes many free parameters which specify an infinite number of decompositions of the TRW free energy into convex and concave parts. TRW-CCCP is guaranteed to converge to the global minima for any settings of these free parameters, including adaptive settings if they satisfy conditions defined in this paper. We show that the values of these free parameters control the speed of convergence of the the inner and outer loops in TRW-CCCP. We performed experiments on a two-dimensional Ising model observing that TRW-CCCP converges to the global minimum of the TRW free energy and that the convergence rate depends on the parameter settings. We compare with the original message passing algorithm (TRW-BP) by varying the difficulty of the problem (by adjusting the energy function) and the number of iterations in the inner loop of TRW-CCCP. We show that on difficult problems TRW-CCCP converges faster than TRW-BP (in terms of total number of iterations) if few inner loop iterations are used.

Keywords TRW Free Energy, CCCP, Probabilistic Inference, Markov Random Fields, Marginals

§1 Introduction

Probabilistic inference – computing a set of marginals and/or the most probable assignment (MAP estimate) given a large-scale graphical model – has many applications in computer vision [1, 2], protein folding [3], genetic analysis [4], and neural science [5]. If the graphical models are defined over singly-connected graphs (i.e. trees), then the marginals can be efficiently and exactly computed by belief propagation (BP) algorithms. But exact and efficient algorithms do not exist in general when the models are defined over graphs with cycles. This motivated researchers to explore approximate algorithms for graphs with cycles, favoring algorithms which are as accurate as possible and are guaranteed to converge.

Variational inference algorithms for marginals are obtained via two steps; (1) selecting a free energy which is a function of pseudo-marginals, and (2) designing an optimization algorithms to minimize it. The pseudo-marginals which minimize the free energy yield the approximate marginals. Several convergent algorithms have been obtained for minimizing the Bethe free energy [6, 7]. But the Bethe free energy, and the Kikuchi generalization, typically have multiple local minima and so even convergent algorithms are not guaranteed to find the global minimum. By contrast, the Tree-reweighted (TRW) free energy [8], is constructed as a convex upper bounds of the log partition function. This implies that there is a single minimum and so any convergent algorithm is guaranteed to find it. But the original message passing algorithm (TRW-BP) [8] is not guaranteed to be convergent although it often converges efficiently in practice. This has lead to the recent development of convergent algorithms for free-energies. Recent examples include Globerson and Jaakkola's TRW-GP and Meltzer *et al.*'s sum-TRW-S which we describe in the discussion section (6).

In this paper, we propose a family of convergent double-loop algorithms to minimize the TRW free energy. They are designed following the convex concave procedure (CCCP) and include many free parameters which affect the updates in the inner and outer loops, and hence control the speed of convergence. These TRW-CCCP algorithms are guaranteed to monotonically decrease the TRW free energy and hence are guaranteed to converge to the global minimum. We implement TRW-CCCP for the Ising spin model, explore the convergence rates as a function of the free parameters and give comparisons to alternative methods. Another motivation for exploring CCCP algorithms is because of their recent use for learning latent Support Vector Machine

(SVM) models [9].

This paper is organized as follows. We briefly introduce the TRW free energy and our notation in section (2) and then present a family of CCCP in section (3). TRW-CCCP algorithms are proposed in section (4) and experiments and discussion follow in sections (5) and (6), respectively.

§2 TRW Free Energy

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph consisting of the vertex set $\mathcal{V}$ associated with discrete-valued random variables $\mathbf{x} = \{x_1, \ldots, x_n\}$ and an edge set $\mathcal{E}$. We consider pairwise Markov Random Fields (MRFs) defined over graph $\mathcal{G}$, given by

$$p(\mathbf{x};\theta) = \exp\left\{\sum_{i\in\mathcal{V}}\theta_i(x_i) + \sum_{ij\in\mathcal{E}}\theta_{ij}(x_i,x_j) - \Phi(\theta)\right\},$$

where we use θ to denote the set of all parameters $\{\{\theta_i\},\{\theta_{ij}\}\}$ and $\Phi(\theta)$ is the normalizing constant (referred to as the log partition function or negative free energy). The marginal distribution over a subset α of the vertex set $\mathcal{V}$ is specified by $p_\alpha(\mathbf{x}_\alpha) = \sum_{\mathbf{x}\backslash\mathbf{x}_\alpha} p(\mathbf{x};\theta)$ (but this computation is impractical for loopy graphs). Throughout this paper, we will denote the set of true marginals by $\mathbf{p} = \{p_\alpha(\mathbf{x}_\alpha)\}$ and will focus on the singleton and pairwise marginals $\mathbf{p} = \{\{p_i(x_i)\},\{p_{ij}(x_i,x_j)\}\}$. We address the inference task which is to compute, or approximate, the true marginals $\mathbf{p}$ for an MRF defined over a graph $\mathcal{G}$ with cycles.

The task of computing the marginals $\mathbf{p}$ can be re-expressed as a search problem within the local polytope [10], which is the collection of all candidates for the true marginals – namely the set of beliefs $\mathbf{b} = \{b_\alpha(\mathbf{x}_\alpha)\}$ defined over cliques which satisfy the consistency constraints between marginals. In the pairwise setting we consider in this paper, the local polytope $\mathcal{L}(\mathcal{G})$ is defined by

$$\mathcal{L}(\mathcal{G}) = \left\{\mathbf{b}\geq\mathbf{0}\;\middle|\;\begin{array}{ll}\sum_{x_i} b_i(x_i) = 1, & {}^\forall i\in\mathcal{V}\\ \sum_{x_j} b_{ij}(x_i,x_j) = b_i(x_i),\quad \sum_{x_i} b_{ij}(x_i,x_j) = b_j(x_j), & {}^\forall x_i, {}^\forall x_j, {}^\forall ij\in\mathcal{E}.\end{array}\right\}$$

We search over the polytope by defining a free energy $\mathcal{L}(\mathcal{G})$. The beliefs which minimize the free energy yield approximations to the marginals $\mathbf{b}^* \simeq \mathbf{p}$. The free energies have the following form;

$$F(\mathbf{b}) = -\mathbf{b}\cdot\theta - S(\mathbf{b}),$$

where

$$-\mathbf{b}\cdot\theta = \sum_{i\in\mathcal{V}} \langle \theta_i(x_i)\rangle_{b_i} + \sum_{ij\in\mathcal{E}} \langle \theta_{ij}(x_i,x_j)\rangle_{b_{ij}}.$$

Here $-\mathbf{b}\cdot\theta$ is the linear term which is the expected energy of the MRF and $S(\mathbf{b})$ is the entropy term. The choice of the entropy term determines the approximation used in the free energy and, in particular, whether the free energy is convex. The entropy $S(\mathbf{b})$ is usually chosen so that it is computational tractable.

We introduce the TRW entropy following the original derivation [8]. Let T be a spanning tree on the graph $\mathcal{G}$ and $\mathcal{T}$ be the set of all spanning trees. Then the entropy of each spanning tree $T\in\mathcal{T}$ is given by

$$S_T(\mathbf{b}) = \sum_{i\in\mathcal{V}} S_i(b_i) - \sum_{ij\in T} I_{ij}(b_{ij})$$

where $I_{ij}(b_{ij})$ is the mutual information measured using the pairwise beliefs $b_{ij}(x_i,x_j)$. The tree entropy $S_T(\mathbf{b})$ is convex in $\mathbf{b}$ over the local polytope $\mathcal{L}(\mathcal{G})$. Now define a probability distribution $\rho(T)$ over the spanning trees satisfying $\rho(T)\geq 0$ and $\sum\limits_{T\in\mathcal{T}}\rho(T) = 1$. The TRW entropy is given by the weighted linear combination of such tree entropies;

$$S_{TRW}(\mathbf{b}) = \sum_{T\in\mathcal{T}} \rho(T) S_T(\mathbf{b}) = \sum_{i\in\mathcal{V}} S_i(b_i) - \sum_{ij\in\mathcal{E}} \rho_{ij} I_{ij}(b_{ij})$$

where ρ_{ij} is the edge appearance probability of $ij\in\mathcal{E}$, meaning the probability that the edge $ij\in\mathcal{E}$ appears in the spanning trees (obtained from the distribution $\rho(T)$ over spanning trees). The TRW entropy $S_{TRW}(\mathbf{b})$ is convex over the local polytope $\mathcal{L}(\mathcal{G})$, since it is a linear combination of convex terms.

More precisely, let $\rho_{\mathbf{e}} = \{\rho_{ij}\}$ denote the set of edge appearance probabilities and $v(T)\in\{0,1\}^{|\mathcal{E}|}$ be an indicator variable such that $[v(T)]_{ij} = 1$, $ij\in T$, and otherwise 0. Then, the edge appearance vector $\rho_{\mathbf{e}}$ is obtained from the distribution $\rho(T)$ over the set of spanning tree polytope;

$$\mathbb{T}(\mathcal{G}) = \left\{ \rho_{\mathbf{e}} \in \mathbb{R}^{|\mathcal{E}|} | \rho_{\mathbf{e}} = \sum_{T\in\mathcal{T}} \rho(T) v(T) \right\}.$$

We choose $\rho(T)$ to ensure that all elements of the edge appearance vector $\rho_{\mathbf{e}}$ are strictly positive [8].

Then the TRW entropy can be rewritten over the local polytope $\mathcal{L}(\mathcal{G})$ as

$$\bar{S}_{TRW}(\mathbf{b}) = \sum_{i \in \mathcal{V}} c_i S_i(b_i) + \sum_{ij \in \mathcal{E}} c_{ij} S_{ij}(b_{ij}) \tag{1}$$

where $c_{ij} = \rho_{ij}$ and $c_i = 1 - \sum_{j \in N_i} \rho_{ij}$ (N_i is the set of neighboring nodes of node i). This is of similar form to the Bethe-approximation entropy $S_{Bethe}(\mathbf{b})$, which can be obtained by setting $\rho_{\mathbf{e}} = \mathbf{1}$ (which is not in the spanning tree polytope $\mathbf{1} \notin \mathbb{T}(\mathcal{G})$ and hence is not guaranteed to be convex if the graph is loopy). In the subsequent sections, we give convergent algorithms to minimize the following TRW free energy;

$$F_{TRW}(\mathbf{b}) = -\mathbf{b} \cdot \theta - \bar{S}_{TRW}(\mathbf{b}). \tag{2}$$

§3 CCCP

We now introduce CCCP (the Concave-Convex Procedure) [6, 11], which is an optimization procedure that can monotonically decrease any objective function $F(\mathbf{b})$, provided it can be expressed as the difference between two convex functions $f^0_{vex}(\mathbf{b})$ and $g^0_{vex}(\mathbf{b})$. Once the function has been expressed as $F(\mathbf{b}) = f^0_{vex}(\mathbf{b}) - g^0_{vex}(\mathbf{b})$, the variables $\mathbf{b}$ are updated by satisfying $\partial f^0_{vex}(\mathbf{b}^{t+1}) = \partial g^0_{vex}(\mathbf{b}^t)$ where ∂ is the gradient with respect to $\mathbf{b}$. In other words, the variables $\mathbf{b}$ are updated so that the tangent vectors are the same for both convex functions. CCCP was applied to the Bethe and the more general Kikuchi free energies [6] yielding algorithms, Bethe-CCCP and Kikuchi-CCCP, which are guaranteed to converge to local minima of the free energies.

We point out that there are an infinite number of ways to express a function $F(\mathbf{b})$ into convex and concave functions. This can be seen by changing the decomposition $F(\mathbf{b}) = f^0_{vex}(\mathbf{b}) - g^0_{vex}(\mathbf{b})$ to a new decomposition by adding a convex function $h_{vex}(\mathbf{b})$ to $f^0_{vex}(\mathbf{b})$ and subtracting it from $g^0_{vex}(\mathbf{b})$ (i.e., $F(\mathbf{b}) = f^0_{vex}(\mathbf{b}) + h_{vex}(\mathbf{b}) - (g^0_{vex}(\mathbf{b}) + h_{vex}(\mathbf{b}))$).

Taking into account the infinite number of decompositions, we can restate the CCCP framework in reference [6, 11]. Let $\mathcal{F}_{vex}$ be the set of all convex functions and $\mathrm{Dec}(F)$ be the set of all pairs of two convex functions such that the difference of the two convex functions is equal to the objective $F(\mathbf{b})$, that is,

$$\mathrm{Dec}(F) = \left\{ (f_{vex}, g_{vex}) \in \mathcal{F}^2_{vex} \mid f_{vex}(\mathbf{b}) - g_{vex}(\mathbf{b}) = F(\mathbf{b}) \right\}.$$

Then, the following theorem holds for CCCP.

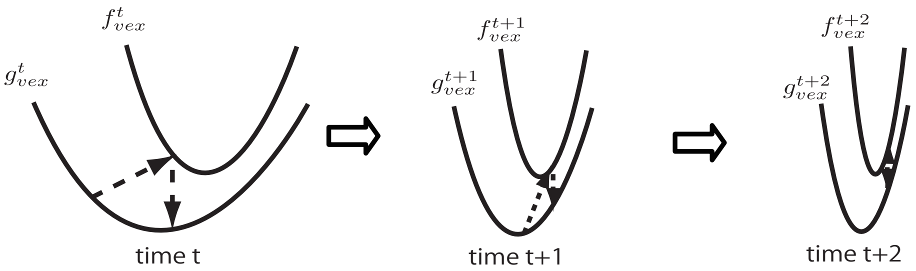

Fig. 1 Illustration of the geometrical meaning of CCCP for an infinite number of decompositions. An objective function $F(\mathbf{b})$ can be monotonically minimized by a series of decompositions $\{(f_{vex}^t, g_{vex}^t) \in \mathrm{Dec}(F)\}$. For each time step t, variables $\mathbf{b}$ are updated so that the two tangent vectors of the two convex functions are the same.

Theorem 3.1

Given a decomposition of an objective function $F(\mathbf{b})$ into convex and concave functions, $(f_{vex}, g_{vex}) \in \mathrm{Dec}(F)$, the iterative algorithm

$$\partial f_{vex}(\mathbf{b}^{t+1}) = \partial g_{vex}(\mathbf{b}^t) \tag{3}$$

is guaranteed to monotonically decrease the objective function $F(\mathbf{b})$.

More generally, the following theorem holds.

Theorem 3.2

Given a series of decompositions of an objective function $F(\mathbf{b})$ into convex and concave functions, $\{(f_{vex}^t, g_{vex}^t) \in \mathrm{Dec}(F)\}$, the iterative algorithm

$$\partial f_{vex}^t(\mathbf{b}^{t+1}) = \partial g_{vex}^t(\mathbf{b}^t) \tag{4}$$

is guaranteed to monotonically decrease the objective function $F(\mathbf{b})$.

The intuitive meaning of the CCCP framework is illustrated in Fig. 1. Iterative algorithms (3) and (4) give a family of CCCP algorithms induced by an objective function $F(\mathbf{b})$. If we denote the specific CCCP algorithm under a decomposition (f_{vex}^0, g_{vex}^0) by $\mathrm{CCCP}(F; f_{vex}^0, g_{vex}^0)$ then the family of CCCP algorithms is given by the collection:

$$\mathrm{CCCP}(F) = \{\mathrm{CCCP}(F; f_{vex}, g_{vex}) \mid (f_{vex}, g_{vex}) \in \mathrm{Dec}(F)\}\,. \tag{5}$$

This means that a CCCP algorithm derived under decomposition (f_{vex}^0, g_{vex}^0) can be modified to give an infinite number of convergent algorithms.

Many existing algorithms in machine learning and neural networks can be interpreted in terms of CCCP [11]. This includes Expectation-Maximization (EM), Legendre minimization, Generalized Iterative Scaling, and Sinkhorn's algorithm. Even steepest descent can be derived in this manner. We note that all of these algorithms can be modified by using alternatives CCCP decompositions.

§4 A family of CCCP to TRW Free Energy

We derive a family of CCCP algorithms to minimize the TRW free energy (2) by expressing it as the difference of two convex free energies.

Let f_{vex} be a convex free energy which has the form of

$$f_{vex}(\mathbf{b};\mathbf{u}) = -\mathbf{b}\cdot\theta - \sum_{i\in\mathcal{V}} u_i S_i(b_i) - \sum_{ij\in\mathcal{E}} u_{ij} S_{ij}(b_{ij}), \tag{6}$$

where $\mathbf{u} = (\{u_i\}.\{u_{ij}\})$ is a free vector to ensure that $f_{vex}(\mathbf{b};\mathbf{u})$ is convex. In what follows we use $\mathbf{u}\cdot\mathbf{S}$ to denote the linear combination of entropies in (6). Let $\tilde{\mathcal{F}}_{vex}$ be the set of all convex free energies given by

$$\tilde{\mathcal{F}}_{vex} = \{f_{vex} \mid f_{vex}(\mathbf{b};\mathbf{u}) \in \mathcal{F}_{vex}\}$$

and $\mathcal{U}$ be the corresponding set of all the free vectors $\mathbf{u}$. A sufficient condition for $\mathbf{u}$ is such that all parameters are strictly positive ($\mathbf{u} > \mathbf{0}$). We then decompose the TRW free energy as[*1]

$$F_{TRW}(\mathbf{b}) = -\mathbf{b}\cdot\theta + f_{vex}(\mathbf{b};\mathbf{u}) - g_{vex}(\mathbf{b};\mathbf{u}), \tag{7}$$

where the two convex free energies $f_{vex}(\mathbf{b};\mathbf{u})$, $g_{vex}(\mathbf{b};\mathbf{u}) \in \tilde{\mathcal{F}}_{vex}$ are parameterized by

$$\begin{aligned} f_{vex}(\mathbf{b};\mathbf{u}) &= -\mathbf{b}\cdot\theta - \tilde{\mathbf{u}}\cdot\mathbf{S} \\ g_{vex}(\mathbf{b};\mathbf{u}) &= -\mathbf{b}\cdot\theta - \mathbf{u}\cdot\mathbf{S}. \end{aligned}$$

Here $\tilde{\mathbf{u}}$ denotes $\tilde{\mathbf{u}} = \mathbf{c} + \mathbf{u}$ and $\mathbf{c} = \{\{c_i\},\{c_{ij}\}\}$ is the coefficient vector given in eq. (1). A set of free vectors $\mathbf{u}$ which give an infinite number of decompositions (7) are:

$$\mathcal{U}_{\text{CCCP}} = \left\{\mathbf{u}\in\mathbb{R}^{|\mathcal{V}|+|\mathcal{E}|} \mid \mathbf{u}\in\mathcal{U}, \tilde{\mathbf{u}}\in\mathcal{U}\right\}. \tag{8}$$

CCCP algorithms applied to the TRW free energy under eq. (7) are guaranteed to converge for any $\mathbf{u}\in\mathcal{U}_{\text{CCCP}}$. A sufficient condition is such that $\mathbf{u} > 0$ and $\tilde{\mathbf{u}} > 0$, that is, $\mathbf{u} > 0$ and $\mathbf{u} > -\mathbf{c}$.

The resulting TRW-CCCP algorithms are shown in Fig. 2. TRW-CCCP are comprised of double-loop (outer loop + inner loop). The updates in both the inner and

[*1] Here we restricted ourselves to an infinite number of decompositions in the set of all convex free energies $\tilde{\mathcal{F}}_{vex}$ and not in the set of all convex functions $\mathcal{F}_{vex}$ ($\tilde{\mathcal{F}}_{vex} \subset \mathcal{F}_{vex}$). Algorithms are easily derived under such the class of decompositions. Other family of decompositions (e.g. polynomials), following the new CCCP (Section 3), could be possible.

outer loops depend on a free vector $\mathbf{u}$ (i.e., there is a family of updates of inner and outer loops). In every outer loop, the inner loop needs to be iterated until convergence. The inner loop corresponds to solving for a set of beliefs that satisfy the local consistency constraints. Such beliefs can be computed by the unique fixed-point of the inner loop for any setting of the vector $\mathbf{u} \in \mathcal{U}_{\mathrm{CCCP}}$. The outer loop, in every iteration, corresponds to solving for a set of beliefs that satisfies the CCCP update rule (4). The outer loop is guaranteed to decrease the TRW free energy in any setting of the vector $\mathbf{u} \in \mathcal{U}_{\mathrm{CCCP}}$. The family of the algorithms can be controlled by the setting of the free vector, which yields the difference in the speed of convergence, how fast the inner and outer loops reach their fixed-points. Experiments are shown in Section 5.

Note that TRW-CCCP in figure 2 is specified by a set of parameters

$$\left(\left\{\frac{\tilde{u}_i}{\tilde{u}_i + \tilde{u}_{ij}}\right\}, \left\{\frac{\tilde{u}_{ij}}{\tilde{u}_i + \tilde{u}_{ij}}\right\}, \left\{\frac{\tilde{u}_i}{u_i}\right\}, \left\{\frac{\tilde{u}_{ij}}{u_{ij}}\right\}\right). \tag{9}$$

Interestingly, the algorithm can be defined even when $\mathbf{u} \to \infty$. For example, consider a partial-homogeneous case such that $u_i = u_1$ for $i \in \mathcal{V}$ and $u_{ij} = u_2$ for $ij \in \mathcal{E}$, i.e., all free parameters $\mathbf{u}$ can be specified by two parameters (u_1, u_2). Let u_1, u_2 satisfy $u_2 = \alpha u_1$ with a positive real value $\alpha > 0$. By letting parameter $u_1 \to \infty$, TRW-CCCP algorithm ”converges” to the specific algorithm specified by

$$\left(\frac{\tilde{u}_i}{\tilde{u}_i + \tilde{u}_{ij}}, \frac{\tilde{u}_{ij}}{\tilde{u}_i + \tilde{u}_{ij}}, \frac{\tilde{u}_i}{u_i}, \frac{\tilde{u}_{ij}}{u_{ij}}\right) \to \left(\frac{1}{1+\alpha}, \frac{\alpha}{1+\alpha}, 1, 1\right). \tag{10}$$

CCCP framework itself (see Theorem 3.1,3.2) is not defined when $\mathbf{u} \to \infty$. However, resulting algorithm TRW-CCCP is defined even in such the case. This results from parameterization of decompositions of an objective function.

§5 Experiments

We now show numerical results of TRW-CCCP. We experimented with a Ising spin model on a two dimensional grid [8, 12, 13]. MRFs are given by

$$p(\mathbf{x}; \theta) \propto \exp\left\{\sum_{i \in \mathcal{V}} \theta_i x_i + \sum_{ij \in \mathcal{E}} \theta_{ij} x_i x_j\right\} \tag{11}$$

where $\mathbf{x} \in \{+1, -1\}^n$. We considered a uniform distribution over two spanning forests (all horizontal and all vertical chains) for the distribution over spanning trees $\rho(T)$. We considered the partial homogeneous case for free parameters $\tilde{\mathbf{u}}$, $\mathbf{u}$ such that $(\tilde{u}_1, u_2) = (\tilde{u}, 0)$ using the notation in the last section, i.e., $\tilde{u}_i = \tilde{u}$ for $i \in \mathcal{V}$ and $u_{ij} = 0$ for

$ij \in \mathcal{E}$. In this case, a sufficient condition for $\tilde{u}$ is $\tilde{u} > \max\{0, \{c_i\}\} = 0$. We used the same free parameters $(\tilde{u}_1, u_2) = (\tilde{u}, 0)$ over the algorithms.

In our experiments, TRW-CCCP always converged and computed the same pseudo-marginals $\mathbf{b}^*$, independent of the setting of parameter $\tilde{u}$ in the range $\tilde{u} > 0$. The converged values were also the same as those of TRW-BP. This means that TRW-CCCP computed the global minimum of the TRW free energy (because if TRW-BP converges then it converges to the minimum of the TRW free energy). Fig. 3 shows numerical results of TRW-CCCP for parameter settings of $\tilde{u}$. The TRW free energies all monotonically decreased with each step of the outer loop. The effect of the parameter $\tilde{u}$ (thus u) on TRW-CCCP was as follows; when the parameter value u is small, ($\tilde{u} \approx 0$), convergence requires a large number of steps of inner loop within each outer loop, but only a small number of steps of the outer loop. Conversely, if the parameter value u is large ($\tilde{u} \gg 0$), then we need a smaller number of steps of the inner loop within each outer loop, but a large number of steps of outer loop. Hence there exists a trade-off for parameter u between the inner and outer loops. The effect of the free parameters $\mathbf{u}$ on TRW-CCCP can also be analytically explained. This is clearer if we look at the TRW-CCCP algorithms when represented in terms of Lagrange multipliers, see the Appendix. If the parameters $\mathbf{u}$ are large, then each step of the outer loop only causes small changes to the beliefs – i.e. $\mathbf{b}^{(t+1)} \simeq \mathbf{b}^{(t)}$. This means that TRW-CCCP works as a localized algorithm, which needs more steps of the outer loop to reach the minimum of TRW free energy but less steps of the inner loop within each outer loop (Fig. 3). Conversely, when the parameters $\mathbf{u}$ are small, the beliefs can change during one iteration of the outer loop which requires more steps of the inner loop to impose the constraints, see Fig. 3. Thus, the free parameters work like a set of step sizes in the TRW-CCCP and control to what extent TRW-CCCP is localized. But the TRW free energy is monotonically decreased for any settings of the step sizes chosen in $\mathcal{U}_{\mathrm{CCCP}}$.

Our experiments, see Fig. 3 (lower panel) showed that TRW-CCCP could converge faster than TRW-BP in terms of total number of iterations provided: (i) the energy function had large random weights (leading to greater frustration) and (ii) we used a small number of iterations of the inner loop. Theoretical convergence of TRW-CCCP requires that the inner loops converge for each step of the outer loop, but in practice this requirement seemed unnecessary. For easier problems with small weights, and presumably less frustration, TRW-BP generally converged faster than TRW-CCCP.

§6 Discussion & Conclusion

This paper presented a framework of a family of CCCP by considering an infinite number of decompositions and introducing the $\mathbf{u}$ variables. We illustrated CCCP in Fig. 1 and stress that many discrete iterative algorithms can be expressed in terms of CCCP for specific decompositions. It is interesting to consider what new algorithms we can obtain by exploring the space of decompositions and see how the convergence properties depend on the settings of the parameter $\mathbf{u}$.

In particular, we developed a new set of convergent CCCP algorithms which are guaranteed to find the global minimum of the TRW free energy. The algorithms include many free parameters ($\mathbf{u} \in \mathbb{R}^{|\mathcal{V}|+|\mathcal{E}|}$) which could be altered to change the convergence rate. We implemented the TRW-CCCP algorithms, showed that it converged to the global minimum of the TRW free energy and showed that different settings of $\mathbf{u}$ affected the number of iterations required of the inner and outer loops. In particular, smaller settings of $\mathbf{u}$ made TRW-CCCP highly non-local, while still reducing the TRW free energy monotonically.

We compared the speed of TRW-CCCP with alternative algorithms such as Wainwright's TRW-BP [8)] and Meltzer's convergent sum-TRW-S [13)]. Our experiments showed that TRW-CCCP was slower if the size of the weights was small and the inner loop was run with strict convergence conditions. But TRW-CCCP converged faster (in terms of numbers of iterations) than TRW-BP for problems with large weights and by relaxing the convergence requirement on the inner loops. This suggests that TRW-CCCP may be useful in this difficult energy function regime. We note that the convergence of sum-TRW-S [13)] requires performing the updates in a specific order which may be restrictive on large-scale graphs. By contrast, TRW-CCCP is very flexible and has a large range of parameter settings which can be used.

A convergent single loop algorithm (TRW-GP) was recently proposed by Globerson *et al* where the dynamics took place in dual space [12)]. The updates of TRW-GP are based on gradient descent with a small step size. In practice its convergence is reported to be slower than TRW-BP, but we did not implement it and compare with TRW-CCCP. We note that TRW-CCCP can be local or non-local, depending on $\mathbf{u}$, while TRW-GP is constrained to be local in the dual space.

Finally, we point out that we can obtain Yuille's Bethe-CCCP algorithm [6)] from TRW-CCCP by setting $\rho_{\mathbf{e}} = \mathbf{1}$ (in 2) and setting $\mathbf{u} = \mathbf{c}_{\max} - \mathbf{c}_B$ in $\mathcal{U}_{\mathrm{CCCP}}$, where $\mathbf{c}_B$ is the over-counting vector of the Bethe approximation and $\mathbf{c}_{\max}$ is the maximum vector. Heskes [7)] double-loop algorithm to minimize the general Kikuchi free energies can also be obtained in a similar manner. The algorithm exploits tighter upper bounds, expecting that tighter bounds yield faster algorithm. Our algorithm explicitly expresses a family of upper bounds with free parameters.

Acknowledgment

This research was partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for JSPS Fellows 19005165 and for Scientific Research 18079007.

References

1) Freeman, W., Pasztor, E. and Carmichael, O., ”Learning Low-Level Vision,” *International Journal of Computer Vision, 40, 1*, pp. 25-47, 2000.

2) Tappen, M. and Freeman, W., ”Comparison of Graph Cuts with Belief Propagation for Stereo, Using Identical MRF Parameters,” in *proc. of the 9th International Conference on Computer Vision (ICCV2003)*, pp. 900-907, 2003.

3) Yanover, C., Meltzer, T. and Weiss, Y., ”Linear programming relaxations and belief propagation -an empirical study,” *Journal of Machine Learning Research, 7*, pp. 1887-1907, 2006.

4) Lauritzen, S. and Sheehan, N, ”Graphical Models for Genetic Analyses,” *Statistical Science, 18, 4*, pp. 489-514, 2003.

5) Steimer, A., Maass, W. and Douglas, R., ”Belief propagation in networks of spiking neurons,” *Neural Comput., 21, 9*, pp. 2502-2523, 2009.

6) Yuille, A., ”CCCP Algorithms to Minimize the Bethe and Kikuchi Free Energies: Convergent Alternatives to Belief Propagation,” *Neural Comput., 14, 7*, pp. 1691-1722, 2002.

7) Heskes, T., ”Convexity Arguments for Efficient Minimization of the Bethe and Kikuchi Free Energies,” *Journal of Artificial Intelligence Research, 26*, pp. 153-190, 2006.

8) Wainwright, M., Jaakkola, T. and Willsky, A., ”A New Class of Upper Bounds on the Log Partition Function,” *IEEE trans. Information Theory, 51, 7*, pp. 2313-2335, 2005.

9) Yu, C. and Joachims, T., ”Learning structural svms with latent variables,” in *proc. of the 26th International Conference on Machine Learning (ICML2009)*, pp. 1169-1176, 2009.

10) Wainwright, M. and Jordan, M., ”Graphical models, exponential families, and variational inference,” *Technical Report, UC Berkeley, Dept. of Statistics*, 2003.

11) Yuille, A. and Rangarajan, A., ”The Concave-Convex Procedure,” *Neural Comput., 15, 4*, pp. 915-936, 2003.

12) Globerson, A. and Jaakkola, T., ”Convergent propagation algorithms via oriented trees,” in *proc. of The 23rd Conference on Uncertainty in Artificial Intelligence (UAI2007)*, 2007.

13) Meltzer, T., Globerson, A. and Weiss, Y., ”Convergent message passing algorithms -a unifying view,” in *proc. of The 25rd Conference on Uncertainty in Artificial Intelligence (UAI2009)*, 2009.

§7 Appendix

We show a family of CCCP algorithms applied to the TRW free energy (TRW-CCCP), where the outer and inner loops are expressed in terms of beliefs and Lagrange multipliers. The Lagrangian of the TRW free energy is given by[*2]

$$\begin{aligned} L_{TRW}(\mathbf{b};\eta,\nu) \;=\; & F_{TRW}(\mathbf{b}) + \sum_{ij\in\mathcal{E}}\sum_{x_i}\eta_{ij,i}(x_i)\left(\sum_{x_j} b_{ij}(x_i,x_j) - b_i(x_i)\right) \\ & + \sum_{ij\in\mathcal{E}}\sum_{x_j}\eta_{ij,j}(x_j)\left(\sum_{x_i} b_{ij}(x_i,x_j) - b_j(x_j)\right) \\ & + \sum_{i\in\mathcal{V}} v_i\left(\sum_{x_i} b_i(x_i) - 1\right) + \sum_{ij\in\mathcal{E}} v_{ij}\left(\sum_{x_i,x_j} b_{ij}(x_i,x_j) - 1\right). \end{aligned}$$

The following theorem then holds for the minimization of the Lagrangian.

Theorem 7.1
(Outer loop) For any parameters $\mathbf{u}$ taken from the set $\mathcal{U}_{CCCP}$ in eq. (8), the following belief updates are guaranteed to monotonically decrease the TRW free energy if the multipliers η and ν are chosen to ensure that beliefs satisfy the consistency constraints:

$$\begin{aligned} b_i^{(t+1)}(x_i) \quad &= \left[b_i^{(t)}(x_i)\right]^{\frac{u_i}{\tilde{u}_i}} \exp\left\{\frac{1}{\tilde{u}_i}\left(\sum_{k\in N_i}\eta_{ik,i}(x_i) - \varphi_i - v_i\right) - \frac{c_i}{\tilde{u}_i}\right\} \\ b_{ij}^{(t+1)}(x_i,x_j) &= \left[b_{ij}^{(t)}(x_i,x_j)\right]^{\frac{u_{ij}}{\tilde{u}_{ij}}} \exp\left\{-\frac{1}{\tilde{u}_{ij}}\left(\eta_{ij,i}(x_i) + \eta_{ij,j}(x_j) + \varphi_{ij} + v_{ij}\right) - \frac{c_{ij}}{\tilde{u}_{ij}}\right\}, \end{aligned}$$

where $\tilde{u}_i$ and $\tilde{u}_{ij}$ are given by $\tilde{u}_i = u_i + c_i$ and $\tilde{u}_{ij} = u_{ij} + c_{ij}$.Here c_i and c_{ij} are coefficients of the singleton and pairwise entropies $S_i(b_i)$ and $S_{ij}(b_{ij})$, respectively, and given by $c_{ij} = \rho_{ij}$ and $c_i = 1 - \sum_{j\in N_i}\rho_{ij}$.

Theorem 7.2
The multipliers η and ν, which ensure that beliefs satisfy the consistency constraints,

[*2] The setting of constraints here is redundant and either of multipliers ν_i or ν_{ij} can be removed.

are given by the unique fixed-point of the following inner loop;

$$\begin{aligned}
\eta_{ij,i}^{(\tau+1)}(x_i) &= \eta_{ij,i}^{(\tau)}(x_i) + \frac{\tilde{u}_i\tilde{u}_{ij}}{\tilde{u}_i + \tilde{u}_{ij}} \ln \frac{\sum_{x_j} b_{ij}^{(t+1)}(x_i, x_j)}{b_i^{(t+1)}(x_i)} \\
v_i^{(\tau+1)} &= v_i^{(\tau)} + \tilde{u}_i \ln \sum_{x_i} b_i^{(t+1)}(x_i) \\
v_{ij}^{(\tau+1)} &= v_{ij}^{(\tau)} + \tilde{u}_{ij} \ln \sum_{x_i, x_j} b_{ij}^{(t+1)}(x_i, x_j),
\end{aligned}$$

where free parameters $\mathbf{u}$ take the same values as the setting in outer loop. The fixed-point is guaranteed to be reached by serial updates (In each step, one multiplier is chosen and updated while the other multipliers are fixed.).

Algorithm 1 TRW-CCCP; A family of CCCP to minimize the TRW free energy.

```
1. Initialization: $b_\alpha^{(1)}(\mathbf{x}_\alpha) = b_\alpha^{(0)}(\mathbf{x}_\alpha) \propto \exp\left[\frac{\theta_\alpha(\mathbf{x}_\alpha)}{c_\alpha}\right]$    $\alpha \in \mathcal{V} \cup \mathcal{E}$
2. Choose a free vector $\mathbf{u}$ in the set $\mathcal{U}_{\mathrm{CCCP}}$.
3. Inner loop:
       $b_i(x_i) \propto \left[\frac{b_{ij}(x_i)}{b_i(x_i)}\right]^{\frac{\tilde{u}_{ij}}{\tilde{u}_i + \tilde{u}_{ij}}} b_i(x_i)$
       $b_{ij}(x_i, x_j) \propto \left[\frac{b_{ij}(x_i)}{b_i(x_i)}\right]^{-\frac{\tilde{u}_i}{\tilde{u}_i + \tilde{u}_{ij}}} b_{ij}(x_i, x_j)$    $ij \in \mathcal{E}$
4. Outer loop:
       $b_\alpha^{(t+2)}(\mathbf{x}_\alpha) \propto \left[\frac{b_\alpha^{(t+1)}(\mathbf{x}_\alpha)}{b_\alpha^{(t)}(\mathbf{x}_\alpha)}\right]^{\frac{u_\alpha}{\tilde{u}_\alpha}} b_\alpha^{(t+1)}(\mathbf{x}_\alpha)$    $\alpha \in \mathcal{V} \cup \mathcal{E}$
5. Output: A set of approximate marginals $\mathbf{b}^*(\simeq \mathbf{p})$ when outer loop converges.
       Otherwise, set beliefs to $\mathbf{b}^{(t)} = \mathbf{b}^{(t+1)}$, $\mathbf{b}^{(t+1)} = \mathbf{b}^{(t+2)}$ and go to 2.
```

Fig. 2 TRW-CCCP in which all updates are expressed in terms of beliefs. For another representation using Lagrange multipliers, see Appendix. In step **1.** two beliefs ($\mathbf{b}^{(0)}$ and $\mathbf{b}^{(1)}$) are initialized and all beliefs $b_\alpha(\mathbf{x}_\alpha)$ are set with the corresponding potentials $\theta_\alpha(\mathbf{x}_\alpha)$ and entropy coefficients c_α.**2.** A free vector $\mathbf{u}$, whose value affects the subsequent updates in outer and inner loops, is taken from the set $\mathcal{U}_{\mathrm{CCCP}}$ given in eq. (8). **3.** In the inner loop, beliefs $\mathbf{b}^{(t+1)}$ are iterated with a serial update schedule until convergence (Given $\mathbf{b}^{(0)}$ and $\mathbf{b}^{(1)}$, beliefs $\mathbf{b}^{(1)}$ are iterated). In the updates, beliefs $b_{ij}(x_i)$ are given by $b_{ij}(x_i) = \sum_{x_j} b_{ij}(x_i, x_j)$. Parameters $\tilde{u}_i$ and $\tilde{u}_{ij}$ denote $\tilde{u}_i = u_i + c_i$ and $\tilde{u}_{ij} = u_{ij} + c_{ij}$, respectively. **4.** When beliefs $\mathbf{b}^{(t+1)}$ are judged to converge in the inner loop, beliefs are incremented by one step using $\mathbf{b}^{(t+1)}$ and $\mathbf{b}^{(t)}$ in the outer loop. **5.** When the outer loop is judged to converge by a stopping criterion (e.g., $||\mathbf{b}^{(t+2)} - \mathbf{b}^{(t+1)}|| < \epsilon$), then $\mathbf{b}^{(t+2)}$ is the final result of the approximate marginals $\mathbf{b}^*$. Otherwise, set beliefs $\mathbf{b}^{(t)} = \mathbf{b}^{(t+1)}$, $\mathbf{b}^{(t+1)} = \mathbf{b}^{(t+2)}$ and go to **2**. In any setting of $\mathbf{u} \in \mathcal{U}_{\mathrm{CCCP}}$, including a series of free vectors $\mathbf{u}^1, \mathbf{u}^2, \cdots \in \mathcal{U}_{\mathrm{CCCP}}$, TRW-CCCP is guaranteed to converge.

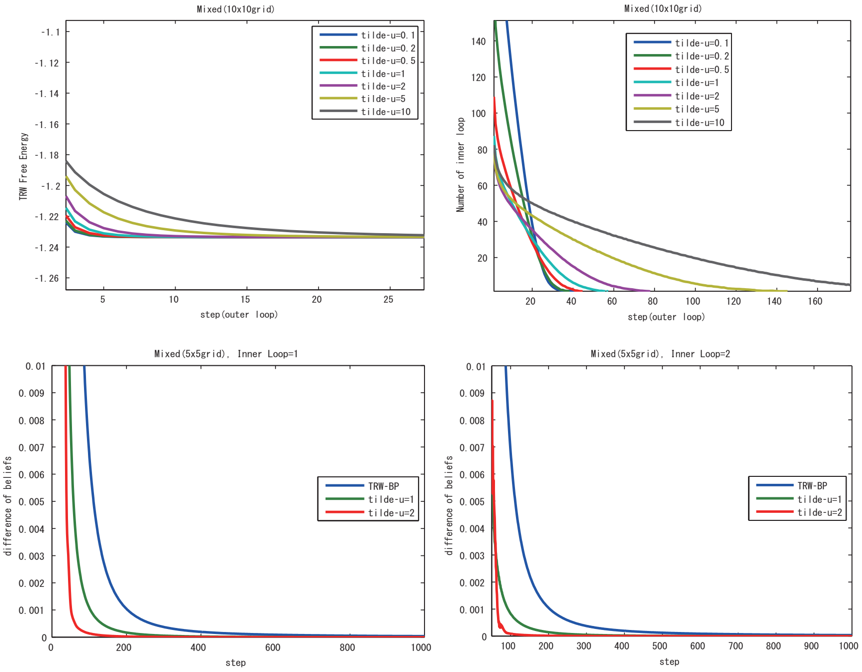

Fig. 3 Numerical results of TRW-CCCP on a two-dimensional Ising model for some settings of the parameter $\tilde{u}$ ($\tilde{u} = 0.1, 0.2, 0.5, 1, 2, 5, 10$). The upper two figures show a mixed case (10x10 grid) where the interaction parameters $\{\theta_{ij}\}$ and the field parameters $\{\theta_i\}$ are both uniformly sampled from $[-1, 1]$. The TRW free energies are all monotonically decreased with each step of the outer loop (upper left figure). The behavior was the same for the attractive cases. The upper right figure shows the number of steps of the inner loop within each outer loop. All results are averaged over 20 samples. In this experiments, we put a strict criterion on the convergence of the inner loop. We observed an interesting behavior of TRW-CCCP with respect to parameter $\tilde{u}$ (thus u); small values of the parameter u lead to a large number of steps of inner loop within each outer loop but only a small number of steps of outer loop are needed until convergence. Conversely, large values of parameter u lead to a small number of steps of the inner loop, within each outer loop, and more steps of the outer loop. The lower two figures show a mixed case (5x5 grid), where the parameters $\{\theta_{ij}\}$ and $\{\theta_i\}$ are sampled from $[-30, 30]$ and correspond to a more frustrated, and more difficult, system. The vertical axes show the difference between beliefs at neighboring iteration steps (given by $\sum_{i\in\mathcal{V}}\sum_{x_i}(b_i^{(t+1)}(x_i) - b_i^{(t)}(x_i))^2$). The number of steps of the inner loop was fixed to 1 and 2 in the lower left and right figures, respectively. Observe that TRW-CCCP converges faster than TRW-BP on these harder problems in terms of total number of iterations.