Belief Propagation, Mean-field, and Bethe approximations

Alan Yuille, Dept. Statistics, UCLA, yuille@stat.ucla.edu

0.1 Section Draft

This chapter describes methods for estimating the marginals and maximum a posteriori (MAP) estimates of *probability distributions defined over graphs* by approximate methods including Mean Field Theory (MFT), variational methods, and belief propagation. These methods typically formulate this problem in terms of minimizing a *free energy function* of *pseudomarginals*. They differ by the design of the free energy and the choice of algorithm to minimize it. These algorithms can often be interpreted in terms of *message passing*. In many cases, the free energy has a *dual formulation* and the algorithms are defined over the *dual variables* (e.g., the messages in belief propagation). The quality of performance depends on the types of free energies used – specifically how well they approximate the *log partition function* of the probability distribution – and whether there are suitable algorithms for finding their minima. We start in section (II) by introducing two types of Markov Field models that are often used in computer vision. We proceed to define MFT/variational methods in section (III), whose free energies are lower bounds of the log partition function, and describe how inference can be done by expectation-maximization, steepest descent, or discrete iterative algorithms. The following section (IV) describes message passing algorithms, such as belief propagation and its generalizations, which can be related to free energy functions (and dual variables). Finally in section (V) we describe how these methods relate to Markov Chain Monte Carlo (MCMC) approaches, which gives a different way to think of these methods and which can lead to novel algorithms.

0.2 Two Models

We start by presenting two important probabilistic vision models which will be used to motivate the algorithms described in the rest of the section.

The first type of model is formulated as a standard Markov Random Field (MRF) with input $\mathbf{z}$ and output $\mathbf{x}$. We will describe two

vision applications for this model. The first application is image labeling where $\mathbf{z} = \{z_i : i \in \mathcal{D}\}$ specifies the intensity values $z_i \in \{0, 255\}$ on the image lattice $\mathcal{D}$ and $\mathbf{x} = \{x_i : i \in \mathcal{D}\}$ is a set of image labels $x_i \in \mathcal{L}$, see figure (1). The nature of the labels will depend on the problem. For edge detection, $|\mathcal{L}| = 2$ and the labels l_1, l_2 will correspond to 'edge' and 'non-edge'. For labeling the MSRC dataset [36] $|\mathcal{L}| = 23$ and the labels $l_1, ..., l_{23}$ include 'sky', 'grass', and so on. A second application is binocular stereo, see figure (2), where the input is the input images to the left and right cameras, $\mathbf{z} = (\mathbf{z}^L, \mathbf{z}^R)$, and the output is a set of disparities $\mathbf{x}$ which specify the relative displacements between corresponding pixels in the two images and hence determine the depth, see figure (2) (!!cite: stereo chapter).

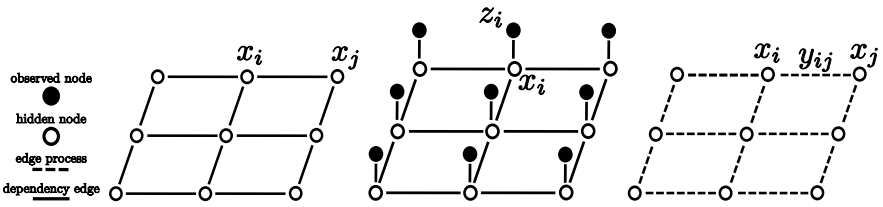

Figure 0.1 GRAPHS for different MRF's. Conventions (far left), basic MRF graph (middle left), MRF graph with inputs z_i (middle right), and graph with lines processors y_{ij} (far right).

We can model these two applications by a posterior probability distribution $P(\mathbf{x}|\mathbf{z})$ and hence is a conditional random field [24]. This distribution is defined on a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where the set of nodes $\mathcal{V}$ is the set of image pixels $\mathcal{D}$ and the edges $\mathcal{E}$ are between neighbouring pixels – see figure (1). The $\mathbf{x} = \{x_i : i \in \mathcal{V}\}$ are random variables specified at each node of the graph. $P(\mathbf{x}|\mathbf{z})$ is a Gibbs distribution specified by an energy function $E(\mathbf{x}, \mathbf{z})$ which contains unary potentials $U(\mathbf{x}, \mathbf{z}) = \sum_{i \in \mathcal{V}} \phi(x_i, \mathbf{z})$ and pairwise potentials $V(\mathbf{x}, \mathbf{x}) = \sum_{ij \in \mathcal{E}} \psi_{ij}(x_i, x_j)$. The unary potentials $\phi(x_i, \mathbf{z})$ depend only on the label/disparity at node/pixel i and the dependence on the input $\mathbf{z}$ will depend on the application: (I) For the labeling application $\phi(x_i, \mathbf{z}) = g(\mathbf{z})_i$, where $g(.)$ is a non-linear filter, which can be obtained by an algorithm like AdaBoost [41], and evaluated in a local image window surrounding pixel i. (II) For binocular stereo, we can set $\phi(x_i, \mathbf{z}^L, \mathbf{z}^R) = |f(\mathbf{z}^L)_i - f(\mathbf{z}^R)_{i+x_i}|$, where $f(.)$ is a vector-value filter and $|.|$ is the L1-norm, so that $\phi(.)$ takes small values at the disparities x_i for which the filter responses are similar on the two images.

The pairwise potentials impose prior assumptions about the local 'context' of the labels and disparities. These models typically assume that neighboring pixels will tend to have similar labels/disparities – see figure (2).

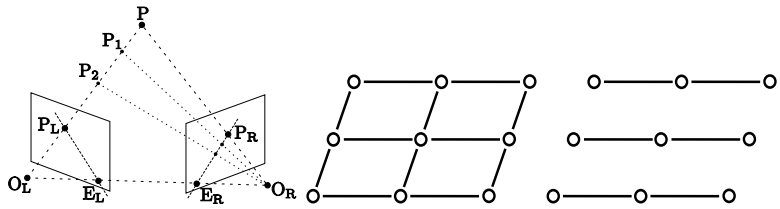

Figure 0.2 Stereo. The geometry of stereo (left). A point P in 3-D space is projected onto points P_L, P_R in the left and right images. The projection is specified by the focal points O_L, O_R and the directions of gaze of the cameras (the camera geometry). The geometry of stereo enforces that points in the plane specified by P, O_L, O_R must be projected onto corresponding lines E_L, E_R in the two images (the epipolar line constraint). If we can find the correspondence between the points on epipolar lines then we can use trigonometry to estimate their depth, which is (roughly) inversely proportional to the disparity, which is the relative displacement of the two images. Finding the correspondence is usually ill-posed unless and requires making assumptions about the spatial smoothness of the disparity (and hence of the depth). Current models impose weak smoothness priors on the disparity (center). Earlier models assumed that the disparity was independent across epipolar lines which lead to similar graphic models (right) where inference could be done by dynamic programming.

In summary, the first type of model is specified by a distribution $P(\mathbf{x}|\mathbf{z})$ defined over discrete-valued random variables $\mathbf{x} = \{x_i : i \in \mathcal{V}\}$ defined on a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$:

$$P(\mathbf{x}|\mathbf{z}) = \frac{1}{Z(\mathbf{z})} \exp\{-\sum_{i \in \mathcal{V}} \phi_i(x_i, \mathbf{z}) - \sum_{ij \in \mathcal{E}} \psi_{ij}(x_i, x_j)\}. \tag{0.1}$$

The goal will be to estimate properties of the distribution such as the MAP estimator and the marginals (which relate to each other, as discussed in subsection (III-E):

$$\begin{aligned} \mathbf{x}^* &= \arg\max_{\mathbf{x}} P(\mathbf{x}|\mathbf{z}), \text{ the MAP estimate,} \\ p_i(x_i) &= \sum_{\mathbf{x}/i} P(\mathbf{x}|\mathbf{z}), \ \forall i \in \mathcal{V} \text{ the marginals.} \end{aligned} \tag{0.2}$$

The *second type of model* has applications to image segmentation, image denoising, and depth smoothing. It is called the weak membrane model and it was proposed independently by Geman and Geman [16] and Blake and Zisserman [5]). This model has additional 'hidden variables' $\mathbf{y}$, which are used to explicitly label discontinuities. It is also a generative model which specifies a likelihood function and a prior probability (by contrast to conditional random fields which specify the posterior distribution only). This type of model can be extended by using more sophisticated hidden variables to perform tasks such as long range motion correspondence [46], object alignment [7], and the detection of particle tracks in high energy physics experiments [28].

The input to the weak membrane model is the set of intensity (or depth) values $\mathbf{z} = \{z_i : i \in \mathcal{D}\}$ and the output is $\mathbf{x} = \{x_i : i \in \mathcal{D}\}$ defined on a corresponding output lattice (formally we should specify two different lattices, say $\mathcal{D}_1$ and $\mathcal{D}_2$, but this makes the notation too cumbersome). We define a set of edges $\mathcal{E}$ which connect neighbouring pixels on the output lattice and define the set of line processes $\mathbf{y} = \{y_j : j \in \mathcal{D}_e\}$ with $y_{ij} \in \{0, 1\}$ over these edges, see figure (1). The weak membrane is a generative model so it is specified by two probability distributions: (i) the likelihood function $P(\mathbf{z}|\mathbf{x})$, which specifies how the observed image $\mathbf{z}$ is a corrupted version of the image $\mathbf{x}$, and (ii) the prior distribution $P(\mathbf{x}, \mathbf{y})$ which imposes a *weak membrane* by requiring that neighbouring pixels take similar values except at places where the line process is activated.

The simplest version of the weak membrane model is specified by the distributions:

$$P(\mathbf{z}|\mathbf{x}) = \prod_{i \in \mathcal{D}} \sqrt{\frac{\tau}{\pi}} \exp\{-\tau(z_i - x_i)^2\}, \quad P(\mathbf{x}, \mathbf{y}) \propto \exp\{-E(\mathbf{x}, \mathbf{y})\},$$
$$\text{with } E(\mathbf{x}, \mathbf{y}) = A \sum_{(i,j) \in \mathcal{E}} (x_i - x_j)^2 (1 - y_{ij}) + B \sum_{(i,j) \in \mathcal{E}} y_{ij}. \quad (0.3)$$

In this model the intensity variables x_i, z_i are continuous-valued while the line processor variables $y_{ij} \in \{0, 1\}$, where $y_{ij} = 1$ means that there is an (image) edge at $ij \in \mathcal{E}_x$. The likelihood function $P(\mathbf{z}|\mathbf{x})$ assume independent zero-mean Gaussian noise (for other noise models, like shot noise, see Geiger and Yuille [14] and Black and Rangarajan [3]). The prior $P(\mathbf{x}, \mathbf{y})$ encourages neighboring pixels i, j to have similar intensity values $x_i \approx x_j$ except if there is an edge $y_{ij} = 1$. This prior imposes piecewise smoothness, or weak smoothness, which is justified by statistical studies of intensities and depth measurements (see Zhu

and Mumford [51], Black and Roth [4]). More advanced variants of this model will introduce higher order coupling terms of form $y_{ij}y_{kl}$ into the energy $E(\mathbf{x}, \mathbf{y})$ to encourage edges to group into longer segments which may form closed boundaries.

The weak membrane model leads to a particularly hard inference problem since it requires estimating continuous and discrete variables, $\mathbf{x}$ and $\mathbf{y}$, from $P(\mathbf{x}, \mathbf{y}|\mathbf{z}) \propto P(\mathbf{z}|\mathbf{x})P(\mathbf{x}, \mathbf{y})$.

0.3 Mean Field Theory and Variational Methods

Mean field theory (MRT), also known as variational methods, offers a strategy to design inference algorithms for MRF models. The approach has several advantages: (I) It takes optimization problems defined over discrete variables and converts them into problems defined in terms of continuous variables. This enables us to compute gradients of the energy and use optimization techniques that depend on them such as steepest descent. In particular, we can take hybrid problems defined in terms of both discrete and continuous variables, like the weak membrane, and convert them into continuous optimization problems. (II) We can use 'deterministic annealing' methods to develop 'continuation methods' where we define a one-parameter family of optimization problems indexed by a temperature parameter T. We can solve the problems for large values of T (for which the optimization is simple) and track the solutions to low values of T (where the optimization is hard), see section (III-E). (III) We can show that MFT gives a fast deterministic approximation to Markov Chain Monte Carlo (MCMC) stochastic sampling methods, as described in section (V), and hence can be more efficient that stochastic sampling. (IV) MFT methods can give bounds for quantities such as the partition function $\log Z$ which are useful for model selection problems, as described in [2].

0.3.1 Mean Field Free Energies

The basic idea of MFT is to approximate a distribution $P(\mathbf{x}|\mathbf{z})$ by a simpler distribution $B^*(\mathbf{x}|\mathbf{z})$ which is chosen so that it is easy to estimate the MAP estimate of $P(.)$, and any other estimator, from the approximate distribution $B^*(.)$. This requires specifying a class of approximating distributions $\{B(.)\}$, a measure of similarity between distributions $B(.)$ and $P(.)$, and an algorithm for finding the $B^*(.)$ that minimizes the similarity measure.

In this chapter, the class of approximating distributions are chosen to be factorizable so that $B(\mathbf{x}) = \prod_{i \in \mathcal{V}} b_i(x_i)$, where the $\mathbf{b} = \{b_i(x_i)\}$ are *pseudo-marginals* which obey $b_i(x_i) \geq 0, \ \forall i, x_i$ and $\sum_{x_i} b_i(x_i) = 1, \ \forall i$. This means that the MAP estimate of $\mathbf{x} = (x_1, ..., x_N)$ can be approximated by $\overline{x}_i = \arg\max_{x_i} b^*(x_i)$ once we have determined $B^*(\mathbf{x})$. But note that MFT can be extended to 'structured mean field theory, which allows more structure to the $\{B(.)\}$, see [2]. The similarity measure is specified by the Kullback-Leibler divergence $KL(B, P) = \sum_{\mathbf{x}} B(\mathbf{x}) \log \frac{B(\mathbf{x})}{P(\mathbf{x})}$ which has the properties that $KL(B, P) \geq 0$ with equality only if $B(.) = P(.)$. It can be shown, see section (III-B), that this is equivalent to a mean field free energy $\mathcal{F}(B)$ which is a variational approximation to the free energy $F = \sum_{\mathbf{x}} P(\mathbf{x}) E(\mathbf{x}) - \sum_{\mathbf{x}} P(\mathbf{x}) \log P(\mathbf{x})$ of a physical system described by $P(\mathbf{x}) = \frac{1}{Z} \exp\{-E(\mathbf{x})\}$ [29]. The mean field approximation is obtained by substituting replacing $B(.)$ with $P(.)$ to obtain $\mathcal{F} = \sum_{\mathbf{x}} B(\mathbf{x}) E(\mathbf{x}) - \sum_{\mathbf{x}} B(\mathbf{x}) \log B(\mathbf{x})$.

For the first type of model we define the mean field free energy $\mathcal{F}_{\text{MFT}}(\mathbf{b})$ by:

$$\mathcal{F}_{\text{MFT}}(\mathbf{b}) = \sum_{ij \in \mathcal{E}} \sum_{x_i, x_j} b_i(x_i) b_j(x_j) \psi_{ij}(x_i, x_j)$$
$$+ \sum_{i \in \mathcal{V}} \sum_{x_i} b_i(x_i) \phi_i(x_i, \mathbf{z}) + \sum_{i \in \mathcal{V}} \sum_{x_i} b_i(x_i) \log b_i(x_i). \tag{0.4}$$

The first two terms are the expectation of the energy $E(\mathbf{x}, \mathbf{z})$ with respect to the distribution $\mathbf{b}(\mathbf{x})$ and the third term is the negative entropy of $\mathbf{b}(\mathbf{x})$. If the labels can take only two values – i.e. $x_i \in \{0, 1\}$ – then the entropy can be written as $\sum_{i \in \mathcal{V}} \{b_i \log b_i + (1 - b_i) \log(1 - b_i)\}$ where $b_i = b_i(x_i = 1)$. If the labels take a set of values $l = 1, .., N$, then we can express the entropy as $\sum_{i \in \mathcal{V}} \sum_{l=1}^{M} b_{il} \log b_{il}$ where $b_{il} = b_i(x_i = l)$ and hence the $\{b_{il}\}$ satisfy the constraint $\sum_{l=1}^{M} b_{il} = 1, \ \forall i$.

For the second (weak membrane) model we use pseudo-marginals $\mathbf{b}(\mathbf{y})$ for the line processes $\mathbf{y}$ only. This leads to a free energy $\mathcal{F}_{\text{MFT}}(\mathbf{b}, \mathbf{x})$ specified by:

$$\mathcal{F}_{\text{MFT}}(\mathbf{b}, \mathbf{x}) = \tau \sum_{i \in \mathcal{V}} (x_i - z_i)^2 + A \sum_{ij \in \mathcal{E}} (1 - b_{ij})(x_i - x_j)^2$$
$$+ B \sum_{ij \in \mathcal{E}} b_{ij} + \sum_{ij \in \mathcal{E}} \{b_{ij} \log b_{ij} + (1 - b_{ij}) \log(1 - b_{ij})\}, \tag{0.5}$$

where $b_{ij} = b_{ij}(y_{ij} = 1)$ (the derivation uses the fact that $\sum_{y_{ij}=0}^{1} b_{ij}(y_{ij})y_{ij} = b_{ij}$). As described below, this free energy is exact and involves no approximations.

0.3.2 Mean Field Free Energy and Variational Bounds

We now describe in more detail the justifications for the mean field free energies. For the first type of models the simplest derivations are based on the Kullback-Leibler divergence which was introduced into the machine learning literature by Saul and Jordan [35]. But the mean field free energies can also be derived by related statistics physics techniques [29] and there were early applications to neural networks [18], vision [23] and machine learning [31].

Substituting $P(\mathbf{x}) = \frac{1}{Z}\exp\{-E(\mathbf{x})\}$ and $B(\mathbf{x}) = \prod_{i\in\mathcal{V}} b_i(x_i)$ into the Kullback-Leibler divergence $KL(B, P)$ gives:

$$KL(B,P) = \sum_{\mathbf{x}} B(\mathbf{x})E(\mathbf{x}) + \sum_{\mathbf{x}} B(\mathbf{x})\log B(\mathbf{x}) + \log Z = \mathcal{F}_{\mathrm{MFT}}(B) + \log Z. \tag{0.6}$$

Hence minimizing $\mathcal{F}_{\mathrm{MFT}}(B)$ with respect to B gives: (i) the best factorized approximation to $P(\mathbf{x})$, and (ii) a lower bound to the partition function $\log Z \geq \min_B \mathcal{F}_{\mathrm{MFT}}(B)$ which can be useful to assess model evidence [2].

For the weak membrane model the free energy follows from Neal and Hinton's variational formulation of the expectation maximization EM algorithm [27]. The goal of EM is to estimate $\mathbf{x}$ from $P(\mathbf{x}|\mathbf{z}) = \sum_{\mathbf{y}} P(\mathbf{x}, \mathbf{y}|\mathbf{z})$ after treating the $\mathbf{y}$ as 'nuisance variables' which should be summed out [2]. This can be expressed [27] in terms of minimizing the free energy function:

$$\mathcal{F}_{\mathrm{EM}}(B, \mathbf{x}) = -\sum_{\mathbf{y}} B(\mathbf{y})\log P(\mathbf{x}, \mathbf{y}|\mathbf{z}) + \sum_{\mathbf{y}} B(\mathbf{y})\log B(\mathbf{y}). \tag{0.7}$$

The equivalence of minimizing $\mathcal{F}_{\mathrm{EM}}[B, \mathbf{x}]$ and estimating $\mathbf{x}^* = \arg\max_{\mathbf{x}} P(\mathbf{x}|\mathbf{z})$ can be verified by re-expressing $\mathcal{F}_{\mathrm{EM}}[B, \mathbf{x}]$ as $-\log P(\mathbf{x}|\mathbf{z}) + \sum_{\mathbf{y}} B(\mathbf{y})\log\frac{B(\mathbf{y})}{P(\mathbf{y}|\mathbf{x},\mathbf{z})}$, from which it follows that the global minimum occurs at $\mathbf{x}^* = \arg\min_{\mathbf{x}}\{-\log P(\mathbf{x}|\mathbf{z})\}$ and $B(\mathbf{y}) =$

$P(\mathbf{y}|\mathbf{x}^*, \mathbf{z})$ (because the second term is the Kullback-Leibler divergence which is minimized by setting $B(\mathbf{y}) = P(\mathbf{y}|\mathbf{x}, \mathbf{z})$.

The EM algorithm minimizes $\mathcal{F}_{\rm EM}[B, \mathbf{x}]$ with respect to B and $\mathbf{x}$ alternatively, which gives the E-step and the M-step respectively. For the basic weak membrane model both steps of the algorithm can be performed simply. The E-step requires minimizing a quadratic function, which can be performed by linear algebra, while the M-step can be computed analytically:

$$\text{Minimize wrt } \mathbf{x} \ \{\sum_i \tau(x_i - z_i)^2 + A \sum_{(i,j)\in E} b_{ij}(x_i - x_j)^2, \tag{0.8}$$

$$B(\mathbf{y}) = \prod_{(i,j)\in E} b_{ij}(y_{ij}) \ \ b_{ij} = \frac{1}{1 + \exp\{-A(x_i - x_j)^2 + B\}}. \tag{0.9}$$

The EM algorithm is only guaranteed to converge to a local minimum of the free energy and so good choices of initial conditions are needed. A natural initialization for the weak membrane model is to set $\mathbf{x} = \mathbf{z}$, perform the E-step, then the M-step, and so on. Observe that the M-step corresponds to performing a weighted smoothing of the data $\mathbf{z}$ where the smoothing weights are determined by the current probabilities $B(\mathbf{y})$ for the edges. The E-step estimates the probabilities $B(\mathbf{y})$ for the edges given the current estimates for the $\mathbf{x}$.

Notice that the EM free energy does not put any constraints of the form of the distribution B and yet the algorithm results in a factorized distribution, see equation (9). This results naturally because the variables that are being summed out – the $\mathbf{y}$ variables – are conditionally independent (i.e. there are no terms in the energy $E(\mathbf{x}, \mathbf{z})$ which couple y_{ij} with its neighbors). In addition we can compute $P(\mathbf{x}|\mathbf{z}) = \sum_{\mathbf{y}} P(\mathbf{x}, \mathbf{y}|\mathbf{z})$ analytically to obtain $\frac{1}{Z}\exp\{-\tau \sum_{i\in mD}(x_i - z_i)^2 - \sum_{ij\in mE} g(x_i - x_j)\}$, where $g(x_i - x_j) = -\log\{\exp\{-A(x_i - x_j)^2\} + \exp\{B\}\}$. The function $g(x_i - x_j)$ penalizes $x_i - x_j$ quadratically for small $x_i - x_j$ but tends to a finite value asymptotically for large $|x_i - x_j|$.

Suppose, however, that we consider a modified weak membrane model which includes interactions between the line processes – terms in the energy like $C \sum_{(ij)\times(kl)\in\mathcal{E}_y} y_{ij} y_{kl}$ which encourage lines to be continuous. It is now impossible either to: (a) solve for $B(\mathbf{y})$ in closed form for the E-step of EM, or (b) to compute $P(\mathbf{x}|\mathbf{y})$ analytically. Instead

we use the mean field approximation by requiring that B is factorizable – $B(\mathbf{y}) = \prod_{ij\in\mathcal{E}} b_{ij}(y_{ij})$. This gives a free energy:

$$\mathcal{F}_{\mathrm{MFT}}(\mathbf{b},\mathbf{x}) = \tau\sum_{i\in\mathcal{V}}(x_i - z_i)^2 + A\sum_{ij\in\mathcal{E}}(1-b_{ij})(x_i-x_j)^2$$
$$+B\sum_{ij\in\mathcal{E}} b_{ij} + C\sum_{(ij)\times(kl)\in\mathcal{E}_y} b_{ij}b_{kl} + \sum_{ij\in\mathcal{E}}\{b_{ij}\log b_{ij} + (1-b_{ij})\log(1-b_{ij})\} \quad (0.10)$$

.

0.3.3 Minimizing the Free Energy by Steepest Descent

The mean field free energies are functions of continuous variables (since discrete variables have been replaced by continuous probability distributions) which enables us to compute gradients of the free energy. This allows us to use steepest descent algorithms, or variants like Newton-Raphson. Suppose we take the MFT free energy from equation (4), restrict $x_i \in \{0,1\}$, set $b_i = b_i(x_i = 1)$, then basic steepest descent can be written as:

$$\begin{aligned}\frac{db_i}{dt} &= -\frac{\partial\mathcal{F}_{\mathrm{MFT}}}{\partial b_i}, \\ &= 2\sum_j\sum_{x_j}\psi_{ij}(x_i,x_j)b_j + \phi_i(x_i) - \{b_i\log b_i + (1-b_i)\log(1-b_i)\}\end{aligned} \quad (0.11)$$

The MFT free energy decreases monotonically because $\frac{d\mathcal{F}_{\mathrm{MFT}}}{dt} = \sum_i \frac{\partial\mathcal{F}_{\mathrm{MFT}}}{\partial b_i}\frac{db_i}{dt} = -\sum_i\{\frac{\partial\mathcal{F}_{\mathrm{MFT}}}{\partial b_i}\}^2$ (note that the energy decreases very slowly for small gradients – because the square of a small number is very small). The negative entropy term $\{b_i\log b_i + (1-b_i)\log(1-b_i)\}$ is guaranteed to keep the values of b_i within the range $[0,1]$ (since the gradient of the negative entropy equals $\log b_1/(1-b_i)$ which becomes infinitely large as $b_i \mapsto 0$ and $b_i \mapsto 1$).

There are many variants to steepest descent because we can multiply the gradient by any positive function and still ensure that the MFT free energy decreases. These variants can be useful because they can improve numerical stability. For example, we can set $\frac{db_i}{dt} = -b_i(1-b_i)\frac{\partial\mathcal{F}_{\mathrm{MFT}}}{\partial b_i}$ and obtain $\frac{d\mathcal{F}_{\mathrm{MFT}}}{dt} = -\sum_i b_i(1-b_i)\frac{\partial\mathcal{F}_{\mathrm{MFT}}}{\partial b_i}^2$. This example is identical to the Hopfield analog network models [18] [45] formulated by $du_i/dt = \frac{\partial F}{\partial b_i}$ where $u_i = \log b_i/(1-b_i)$ or, equivalently,

$b_i = 1/(1+\exp\{-u_i\})$. This relates to a simplified model of neuroscience where each neuron receives a set of inputs $\{b_i\}$ at its dendrites (from other neurons), weights these inputs by ψ (the strength of the synapse), sums the weighted inputs and pass them through a non-linear threshold (at the soma of the neuron), and outputs the response as input to other neurons. In practice, real neurons are considerably more complicated but Hopfield's model remains the mean field approximation to the simplest "artificial neuron model".

Similarly we can perform steepest descent on the MFT free energies for the second class of model yielding equations:

$$\begin{aligned} \frac{dx_i}{dt} &= -\frac{\partial \mathcal{F}_{\mathrm{MFT}}(\mathbf{b}, \mathbf{x})}{\partial x_i}, \\ \frac{db_{ij}(y_{ij})}{dt} &= -\frac{\partial \mathcal{F}_{\mathrm{MFT}}(\mathbf{b}, \mathbf{x})}{\partial b_{ij}(y_{ij})}. \end{aligned} \tag{0.12}$$

Again we can modify these equations – for example, inserting a $b_{ij}(1-b_{ij})$ term on the right hand side of the equation for $\frac{db_{ij}(y_{ij})}{dt}$ and using the weak smoothness model (with line process interactions) gives the Koch, Marroquin, Yuille (KMY) model [23].

Although steepest descent is an extremely popular technique it has several practical problems. When implemented on a digital compute it requires approximating the derivative db_i/dt by $\frac{b_i(t+\Delta)-b_i(t)}{\Delta}$ where Δ is a time step. But the choice of Δ is not easy – if it is too large then the algorithm will be unstable and fail to converge, but if it is too small then convergence will be extremely slow. In addition, the stability will depend on the largest gradient magnitude $|\frac{\partial \mathcal{F}_{\mathrm{MFT}}}{\partial b_i}|$ of all nodes i, so Δ may need to be kept small just because of the size of the gradient at one node. This suggests modifying the steepest descent rule so that none of the gradients get too large – for example, the gradients of the MFT free energy become very large as $b_i \mapsto 0$ and $b_i \mapsto 1$ because of the entropy term and multiplying the gradient by $b_i(1-b_i)$ helps prevent these changes from being too large. We refer to [32] for more details on how to implement steepest descent efficiently.

0.3.4 Discrete Iterative Algorithms

Discrete iterative algorithms are designed to decrease the energy for each iteration without needing a time-step parameter Δ. These algorithms can also give large changes in the states at each iteration

rather than "hugging the energy surface" as local methods like steepest descent tends to do. Historically they were first introduced by writing down the fixed point conditions for the variational models and then writing algorithms whose fixed points occurred at extrema of the free energy. Such methods did not always converge. It was realized that discrete iterative methods could be designed which always provably decrease the energy at each iteration.

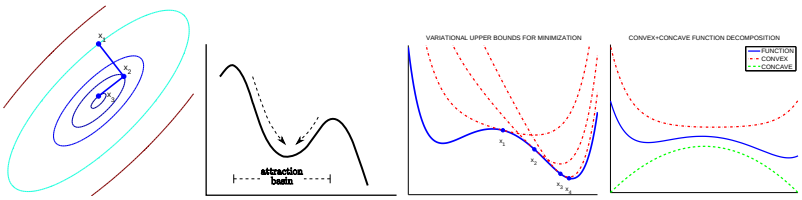

Figure 0.3 The steepest descent algorithm moves downhill in the direction of the gradient (far left) but "hugs the energy surface" and can get trapped (middle left) in local minima of the energy function. Variational bounding requires finding a bounding energy function at each iteration step and minimizing this bound – some bounds are tighter than others (middle right). CCCP is a special case of variational bounding which decomposes the energy function into a sum of a convex and a concave part (far right) and uses this to construct a bound. Variational bounding and CCCP perform large moves and can avoid some local minima.

We describe two strategies for obtaining discrete iterative algorithms (DIA) to minimize any cost function $E(\mathbf{x})$ (e.g., a free energy). The first is variational bounding [34],[21], also known as majorization [9], and the second is CCCP [50] CCCP is a special case but nevertheless seems to include most DIA's obtained by variational bounding and existing algorithms (e.g., EM, generalized iterative scaling, Sinhkorn's algorithm) [50].

We define variational bounding as follows. Suppose we want to minimize $E(\mathbf{x})$. Let us be at $\mathbf{x}^t$ at iteration step t. We construct a bounding function $E_b(\mathbf{x} : \mathbf{x}^t)$, so that $E_b(\mathbf{x}^t : \mathbf{x}^t) = E(\mathbf{x}^t)$ and $E(\mathbf{x}) \leq E_b(\mathbf{x}, \mathbf{x}^t)$. Then choose the next state $\mathbf{x}^{t+1}$ so that $E_b(\mathbf{x}^{t+1} : \mathbf{x}^t) \leq E_b(\mathbf{x}^t : \mathbf{x}^t)$ which implies that $E(\mathbf{x}^{t+1}) \leq E(\mathbf{x}^t)$. Variational bounding is useful because it is often practical to find bounding functions $E_b(\mathbf{x} : \mathbf{x}^t)$ which can be minimized so that $\mathbf{x}^{t+1} = \arg\min E(\mathbf{x} : \mathbf{x}^t)$ [34],[21],[9].

CCCP is a special case of variational bounding. It can be shown that almost all functions $E(\mathbf{x})$ can be decomposed as a sum of a convex $E_{\text{vex}}(\mathbf{x})$ and concave $E_{\text{cave}}(\mathbf{x})$ function [50]. It follows, from properties of convexity, that $E_b(\mathbf{x} : \mathbf{x}^t) = E_{\text{vex}}(\mathbf{x}) + E_{\text{cave}}(\mathbf{x}^t) + (\mathbf{x} - \mathbf{x}^t) \cdot \frac{\partial E_{\text{cave}}}{\partial \mathbf{x}}(\mathbf{x}^t)$

is a bounding function (as for variational bounding). We can minimize $E_b(\mathbf{x}, \mathbf{x}^t)$ by the CCCP procedure by choosing $\mathbf{x}^{t+1}$ so that $\frac{\partial E_{\text{vex}}}{\partial \mathbf{x}}(\mathbf{x}^{t+1}) = -\frac{\partial E_{\text{cave}}}{\partial \mathbf{x}}(\mathbf{x}^t)$.

For free energies, the entropy term is convex and the remaining term will be concave provided $\psi_{ij}(x_i, x_j)$ is positive definite. This can often be imposed by rewriting the original energy function to include 'diagonal' pairwise terms $\psi_{ii}(x_i, x_i)$ which are then 'subtracted' by changing the unary potentials $\phi_i(x_i)$. It is always possible to pick diagonal terms to be sufficiently large so that $\log \psi_{ij}(x_i, x_j)$ (see [47]). For example, consider the Ising model with $E(\mathbf{x}) = \sum_{ij} T_{ij} x_i x_j + \sum_i \theta_i x_i$. We can write this as $E(\mathbf{x}) = \sum_{ij} T_{ij} x_i x_j - \alpha \sum_i x_i^2 + \sum_i \theta_i x_i + \alpha \sum_i x_i$, where α is chosen to be large enough so that the first two terms are negative definite (e.g. make α bigger than the largest positive eigenvalue of the matrix $\mathbf{T} = \{T_{ij}\}$). This does not alter the distribution but will alter the mean field approximation.

This gives a DIA update equation:

$$b_i^{t+1}(x_i) = \frac{\exp\{-\sum_j \sum_{x_j} \psi_{ij}(x_i, x_j) b_j^t(x_t) - \phi_i(x_i)\}}{\sum_{z_i} \exp\{-\sum_j \sum_{z_j} \psi_{ij}(z_i, z_j) b_j^t(z_j^t) - \phi_i(z_i)\}}. \tag{0.13}$$

where the denominator is used to impose the constraint that $\sum_{x_i} b_i(x_i) = 1, \ \forall i$.

We can also apply DIA's in combination with other optimization methods. For example, for the weak membrane free energy we can define an two step algorithms where the first step applies a DIA to update the $b_{ij}(y_{ij})$ and the second step solves the linear equations (8) for $\mathbf{x}$. More generally, we can alternate DIA on $\mathbf{b}$ with any algorithm on $\mathbf{x}$ that is guaranteed to decrease the energy at each iteration.

0.3.5 Temperature and Deterministic annealing

So far we have concentrated on using MFT to estimate the marginal distributions. We now describe how MFT can attempt to estimate the most probable states of the probability distribution $\mathbf{x}^* = \arg\max_{\mathbf{x}} P(\mathbf{x})$. The strategy is to introduce a temperature parameter T and a family of probability distributions related to $P(\mathbf{x})$. (Refer to chapter by Weiss!!).

More precisely, we define a one-parameters family of distributions $\propto \{P(\mathbf{x})\}^{1/T}$ where T is a temperature parameter (the constant of proportionality is the normalization constant). This is equivalent to

specifying Gibbs distributions $P(\mathbf{x};T) = \frac{1}{Z(T)}\exp\{-E(\mathbf{x})/T\}$, where the default distribution $P(\mathbf{x})$ occurs at $T = 1$. The key observation is that as $T \mapsto 0$, the distribution gets strongly peaked about the state $\mathbf{x}^* = \arg\min_{\mathbf{x}} E(\mathbf{x})$ with lowest energy (or states if there are two or more global minima). Conversely, at $T \mapsto \infty$ all states will become equally likely and $P(\mathbf{x};T)$ will tend to the uniform distribution.

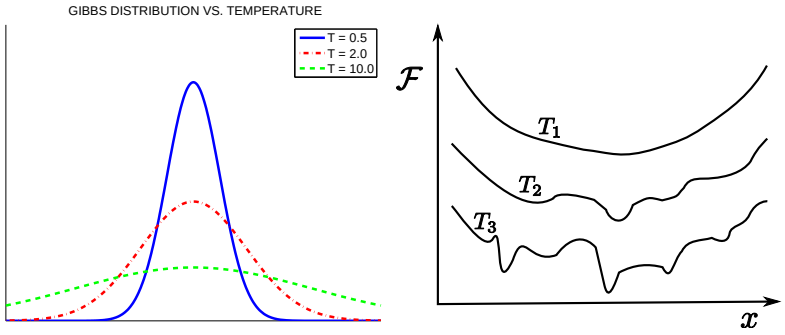

Figure 0.4 The probability distribution $\{P(\mathbf{x})\}^{1/T}$ gets sharply peaked as $T \mapsto 0$ and tends to a uniform distribution for large T (left). The mean field free energy $\mathcal{F}$ is convex for large T and becomes less smooth as T decreases (right). This motivates simulated annealing and deterministic annealing, which is related to graduated non-convexity. For some models, there are phase transitions where the minima of the free energy change drastically at a critical temperature T_c.

Introducing this temperature parameter modifies the free energies by multiplying the entropy term by T. For example, we modify equation (4) to be

$$\mathcal{F}_{\mathrm{MFT}}(\underline{}) = \sum_{ij\in\mathcal{E}}\sum_{x_i,x_j} b_i(x_i)b_j(x_j)\psi_{ij}(x_i,x_j)$$
$$+\sum_{i\in\mathcal{V}}\sum_{x_i} b_i(x_i)\phi_i(x_i,\mathbf{z}) + T\sum_{i\in\mathcal{V}}\sum_{x_i} b_i(x_i)\log b_i(x_i). \qquad (0.14)$$

Observe that for large T, the convex entropy term will dominate the free energy causing it to become convex. But for small T, the remaining terms dominate. In general, we expect that the landscape of the free energy will become smoothed as T increases and in some cases it is possible to compute a temperature T_c above which the free energy has an obvious solution [12]. This motivates a continuation approach

known as *deterministic annealing* which involves minimizing the free energy at large temperatures and using this to provide initial conditions for minimizing the free energies at smaller temperatures. In practice, the best results often require introducing temperature dependence into the parameters [12]. At sufficiently small temperatures the global minima of the free energy can approach the MAP estimates but technical conditions need to be enforced, see [47].

Deterministic annealing was motivated by *simulated annealing* [22] performs stochastic sampling, see section (V) from the distribution $P(\mathbf{x}; T)$ gradually reducing T, so that eventually the samples come form $P(\mathbf{x} : T = 0)$ and hence correspond to the global minimum $\mathbf{x} = \arg\min_{\mathbf{x}} E(\mathbf{x})$. This approach is guaranteed to converge [16] but the theoretically guaranteed rate of convergence is impractically slow and so, in practice, rates are chosen heuristically. Deterministic annealing is also related to the continuation techniques described in Blake and Zisserman [5] to obtain solutions to the weak membrane model.

0.4 Bethe Free Energy and Belief Propagation

We now present a different approach to estimating (approximate) marginals and MAPs of an MRF. This is called belief propagation BP. It was originally proposed as a method for doing inference on trees (e.g. graphs without closed loops) [30] for which it is guaranteed to converge to the correct solution (and is related to dynamic programming). But empirical studies showed that belief propagation will often yield good approximate results on graphs which do have closed loops [26].

To illustrate the advantages of belief propagation, consider the binocular stereo problem which can be addressed by using the first type of model. For binocular stereo there is the epipolar line constraint which means that, provided we know the camera geometry, we can reduce the problem to one-dimensional matching, see figure (2). We impose weak smoothness in this dimension only and then use dynamic programming to solve the problem [15]. But a better approach is to impose weak smoothness in both directions which can be solved (approximately) using belief propagation [38], see figure (2).

Belief propagation is related to the Bethe Free energy [11]. This free energy, see equation (20), appears better than the mean field theory free energy because it includes pairwise pseudo-marginal distributions and reduces to the MFT free energy if these are replaced by the product of unary marginals. But, except for graphs without closed loops (or a single closed loop), there are no theoretical results showing that the

Bethe free energy yields a better approximation than mean field theory. There is also no guarantee that BP will converge for general graphs.

0.4.1 Message Passing

BP is defined in terms of messages $m_{ij}(x_j)$ from i to j, and is specified by the sum-product update rule:

$$m_{ij}^{t+1}(x_j) = \sum_{x_i} \exp\{-\psi_{ij}(x_i, x_j) - \phi_i(x_i)\} \prod_{k \neq j} m_{ki}^t(x_i). \tag{0.15}$$

The unary and binary pseudomarginals are related to the messages by:

$$b_i^t(x_i) \propto \exp\{-\phi_i(x_i)\} \prod_k m_{kj}^t(x_j), \tag{0.16}$$

$$\begin{aligned} b_{kj}^t(x_k, x_j) &\propto \exp\{-\psi_{kj}(x_k, x_j) - \phi_k(x_k) - \phi_j(x_j)\} \\ &\times \prod_{\tau \neq j} m_{\tau k}^t(x_k) \prod_{l \neq k} m_{lj}^t(x_j). \end{aligned} \tag{0.17}$$

The update rule for BP is not guaranteed to converge to a fixed point for general graphs and can sometimes oscillate wildly. It can be partially stabilized by adding a damping term to equation (15). For example, by multiplying the right hand side by $(1 - \epsilon)$ and adding a term $\epsilon m_{ij}^t(x_j)$.

To understand the converge of BP observe that the pseudo-marginals b satisfy the *admissibility constraint*:

$$\frac{\prod_{ij} b_{ij}(x_i, x_j)}{\prod_i b_i(x_i)^{n_i - 1}} \propto \exp\{-\sum_{ij} \psi_{ij}(x_i, x_j) - \sum_i \phi(x_i)\} \propto P(\mathbf{x}), \tag{0.18}$$

where n_i is the number of edges that connect to node i. This means that the algorithm re-parameterizes the distribution from an initial specification in terms of the ϕ, ψ to one in terms of the pseudo-marginals b. For a tree, this re-parameterization is exact (i.e. the pseudo-marginals become the true marginals of the distribution – e.g., we can represent a one-dimensional distribution by $P(\mathbf{x}) = \frac{1}{Z}\{-\sum_{i=1}^{N-1} \psi(x_i, x_{i+1}) - \sum_{i=1}^{N} \phi_i(x_i)\}$ or by $\prod_{i=1}^{N-1} p(x_i, x_{i+1}) / \prod_{i=2}^{N-1} p(x_i)$.

It follows from the message updating equations (15,17) that at convergence, the b's satisfy the *consistency constraints*:

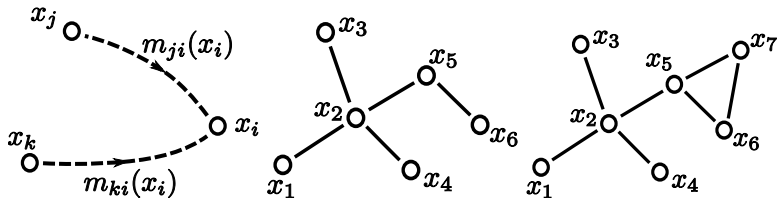

Figure 0.5 Message passing (left) is guaranteed to converge to the correct solution on graphs without closed loops (center) but only gives good approximations on graphs with a limited number of closed loops (right).

$$\sum_{x_j} b_{ij}(x_i, x_j) = b_i(x_i), \quad \sum_{x_i} b_{ij}(x_i, x_j) = b_j(x_j). \tag{0.19}$$

This follows from the fixed point conditions on the messages – $m_{kj}(x_j) = \sum_{x_k} \exp\{-\phi_k(x_k)\} \exp\{-\psi_{jk}(x_j, x_k)\} \prod_{l \neq j} m_{lk}(x_k)$ $\forall k, j, x_j$.

In general, the admissibility and consistency constraints characterize the fixed points of belief propagation. This has an elegant interpretation within the framework of information geometry [19].

0.4.2 The Bethe Free Energy

The Bethe free energy [11] differs from the MFT free energy by including pairwise pseudo-marginals $b_{ij}(x_i, x_j)$:

$$\mathcal{F}[b; \lambda] = \sum_{ij} \sum_{x_i, x_j} b_{ij}(x_i, x_j) \psi_{ij}(x_i, x_j) + \sum_i \sum_{x_i} b_i(x_i) \phi_i(x_i)$$
$$+ \sum_{ij} \sum_{x_i, x_j} b_{ij}(x_i, x_j) \log b_{ij}(x_i, x_j) - \sum_i (n_i - 1) \sum_{x_i} b_i(x_i) \log b_i(x_i). \tag{0.20}$$

But we must also impose consistency and normalization constraints which we impose by lagrange multipliers $\{\lambda_{ij}(x_j)\}$ and $\{\gamma_i\}$:

$$\sum_{i,j}\sum_{x_j}\lambda_{ij}(x_j)\{\sum_{x_i}b_{ij}(x_i,x_j)-b_j(x_j)\}$$

$$+\sum_{i,j}\sum_{x_i}\lambda_{ji}(x_i)\{\sum_{x_j}b_{ij}(x_i,x_j)-b_i(x_i)\}+\sum_i\gamma_i\{\sum_{x_i}b_i(x_i)-1\} \quad (0.21)$$

It can be shown [43] (differentiate with respect to b) that the extrema of the Bethe free energy also obey the admissibility and consistency constraints. Hence the fixed points of belief propagation must correspond to extrema of the Bethe free energy.

0.4.3 Where do the messages come from? The dual formulation.

Where do the messages in belief propagation come from? At first glance, they do not appear directly in the Bethe free energy. But observe that the consistency constraints are imposed by lagrange multipliers $\lambda_{ij}(x_j)$ which have the same dimensions as the messages.

We can think of the Bethe free energy as specifying a *primal problem* defined over *primal variables* b and *dual variables* λ. The goal is to minimize $\mathcal{F}[b;\lambda]$ with respect to the primal variables and maximize it with respect to the dual variables. There corresponds a *dual problem* which can be obtained by minimizing $\mathcal{F}[b;\lambda]$ with respect to b to get solutions $b(\lambda)$ and substituting them back to obtain $\hat{\mathcal{F}}_d[\lambda]=\mathcal{F}[b(\lambda);\lambda]$. Extrema of the dual problem correspond to extrema of the primal problem (and vice versa).

It is straightforward to show that minimizing $\mathcal{F}$ with respect to the b's give the equations:

$$b_i^t(x_i) \propto \exp\{-1/(n_i-1)\{\gamma_i-\sum_j\lambda_{ji}(x_i)-\phi_i(x_i)\}\} \quad (0.22)$$

$$b_{ij}^t(x_i,x_j) \propto \exp\{-\psi_{ij}(x_i,x_j)-\lambda_{ij}^t(x_j)-\lambda_{ji}^t(x_i)\}. \quad (0.23)$$

Observe the similarity between these equations and those specified by belief propagation, see equations (15). They become identical if we identify the messages with a function of the λ's:

$$\lambda_{ji}(x_i) = -\sum_{k\in N(i)/j}\log m_{ki}(x_i). \quad (0.24)$$

There are, however, two limitations of the Bethe free energy. Firstly it does not provide a bound of the partition function (unlike MFT) and so it is not possible to using bounding arguments to claim that Bethe is 'better' than MFT (i.e. it is not guaranteed to give a tighter bound). Secondly, Bethe is non-convex (except on trees) which has unfortunate consequences for the dual problem – the maximum of the dual is not guaranteed to correspond to the minimum of the primal. Both problems can be avoided by an alternative approach, described in Weiss's chapter!! which gives convex upper bounds on the partition function and specifies convergent (single-loop) algorithms.

0.4.4 Double Loop Minimization of the Bethe Free Energy

We can attempt to minimize the Bethe free energy directly by specifying an algorithm which acts directly on the b's. For example, we can apply steepest descent or CCCP/variational bounding. This requires working with variables that have higher dimensions than the messages (contrast $b_{ij}(x_i, x_j)$ with $m_{ij}(x_j)$). But it is easier to obtain convergence results guaranteeing that the algorithms will converge to, at least, a local minimum of the Bethe free energy. These theoretical results, however, come with caveats which must be addressed. Steepest descent will require specifying a time constant Δ and convergence is only guaranteed if Δ is sufficiently small.

It is straightforward to apply CCCP and decompose the free energy into a sum of convex and concave parts [49] because the entropy terms are convex or concave in the pseudomarginals (depending on their sign) while the energy terms are linear in the pseudomarginals and hence both convex and concave. This gives many possible decompositions from which we can construct a convex bounding energy for each time step [17]. But the consistency constraints make it impossible to minimize the convex energy function analytically and instead a convex minimization algorithm is required. This gives a double loop algorithm [49] where the inner loop performs this convex minimization for each step of the outer loop. By contrast, for the CCCP example given in equation (13) there is no need for an inner loop between we can obtain a closed form solution for the minimum of the energy bound. Empirically studies [49][17] show that double loop algorithms are stable and can give better solutions than belief propagation but may require more computation time.

0.5 Stochastic Inference

Stochastic sampling methods – markov chain monte carlo (MCMC) – can also be applied to obtain samples from an MRF which can be used to estimate states. For example, Geman and Geman [16] used simulated annealing – MCMC with changing temperature – to perform inference on the weak smoothness model. As we describe, stochastic sampling is closely related to MFT and BP. Indeed both can be derived as deterministic approximations to MCMC.

0.5.1 MCMC

MCMC is a stochastic method for obtaining samples from a probability distribution $P(\mathbf{x})$. It requires choosing a transition kernel $K(\mathbf{x}|\mathbf{x}')$ which obeys the fixed point condition $P(\mathbf{x}) = \sum_{\mathbf{x}'} K(\mathbf{x}|\mathbf{x}')P(\mathbf{x}')$. In practice, the kernel is usually chosen to satisfy the stronger *detailed balance* condition $P(\mathbf{x})K(\mathbf{x}'|\mathbf{x}) = K(\mathbf{x}|\mathbf{x}')P(\mathbf{x}')$ (the fixed point condition is recovered by taking $\sum_{\mathbf{x}'}$). In addition the kernel must satisfy additional conditions $K(\mathbf{x}|\mathbf{x}') \geq 0$, $\sum_{\mathbf{x}} K(\mathbf{x}|\mathbf{x}') = 1 \ \forall \mathbf{x}'$, and for any pair of states $\mathbf{x}, \mathbf{x}'$ it must be possible to find a trajectory $\{\mathbf{x}_i : i = 0, .., N\}$ such that $\mathbf{x} = \mathbf{x}_0$, $\mathbf{x}' = \mathbf{x}_N$, and $K(\mathbf{x}_{i+1}|\mathbf{x}_i) > 0$ (i.e. you have a non-zero probability of moving between any two states by a finite number of transitions).

This defines a random sequence $x_0, x_1, ..., x_n$ where x_0 is specified and x_{i+1} is sampled from $K(\mathbf{x}_{i+1}|\mathbf{x}_i)$. It can be shown that x_n will tend to a sample from $P(\mathbf{x})$ as $n \mapsto \infty$. (The convergence rate is exponential in the magnitude of the second largest eigenvalue of $K(.|.)$ – but this eigenvalue can almost never be calculated).

The Gibbs sampler is one of the most popular MCMCs, partly because it is so simple. It has transition kernel $K(\mathbf{x}|\mathbf{x}') = \sum_r \rho(r) K_r(\mathbf{x}|\mathbf{x}')$, where $\rho(r)$ is a distribution on the lattice site(s) r (usually $\rho(.)$ is the uniform distribution) and is formally specified by:
$K_r(\mathbf{x}|\mathbf{x}') = P(\mathbf{x}_r|\mathbf{x}'_{N(r)})\delta_{\mathbf{x}_{/r},\mathbf{x}'_{/r}}$.

The Gibbs sampler proceeds by first picking a lattice site(s) at random from $\rho(.)$ and then sampling the state $\mathbf{x}_r$ of the site from the conditional distribution $P(\mathbf{x}_r|\mathbf{x}'_{N(r)})$. As we will illustrate below, the conditional distribution will take a simple form for MRFs and so sampling from it is usually straightforward. It can easily be checked that the Gibbs sampler satisfies the detailed balance conditions.

For example, consider the binary-values case with $x_i \in \{0, 1\}$ and with potentials $\psi_{ij}(x_i, x_j) = \psi_{ij} x_i x_j$ and $\phi_i(x_i) = \phi_i x_i$. The MFT update (using DIA) and the Gibbs sampler are respectively given by:

$$b_i^{t+1} = \frac{1}{1+\exp\{2\sum_j \psi_{ij} b_j^t + \phi_i\}},$$

$$x_i^{t+1} \text{ is sampled from } P(x_i|x_{/i}) = \frac{1}{1+\exp\{x_i(\sum_j \psi_{ij} x_j + \phi_i)\}}. \quad (0.25)$$

Equation (25) shows that the updates for Gibbs sampling are very similar to the updates for MFT. A classic result, reviewed in [1], shows that MFT can be obtained by taking the expectation of the update for the Gibbs sampler. In the next section we will report a similar result for belief propagation.

The Metropolis-Hastings sampler is the most general transition kernel that satisfies the detailed balance conditions. It is of form:

$$K(\mathbf{x}|\mathbf{x}') = q(\mathbf{x}|\mathbf{x}') \min\{1, \frac{p(\mathbf{x})q(\mathbf{x}'|\mathbf{x})}{p(\mathbf{x}')q(\mathbf{x}|\mathbf{x}')}\}, \text{ for } \mathbf{x} \neq \mathbf{x}'. \quad (0.26)$$

Here $q(\mathbf{x}|\mathbf{x}')$ is a proposal probability (which only obeys relaxed conditions). The sampler proceeds by selecting a possible transition $\mathbf{x}' \mapsto \mathbf{x}$ from the proposal probability $q(\mathbf{x}|\mathbf{x}')$ and accepting this transitions with probability $\min\{1, \frac{p(\mathbf{x})q(\mathbf{x}'|\mathbf{x})}{p(\mathbf{x}')q(\mathbf{x}|\mathbf{x}')}\}$. A key advantage of this approach is that it only involves evaluating the ratios of the probabilities $P(\mathbf{x})$ and $P(\mathbf{x}')$ which are typically simple quantities to compute (see the examples below).

In many cases, the proposal probability is selected to be a uniform distribution over a set of possible states. For example, for the first type of model we let the proposal probability choose a site i at a new state value x_i' at random (from uniform distributions) which proposes a new state $\mathbf{x}'$. We always accept this proposal if $E(\mathbf{x}') \leq E(\mathbf{x})$ and we accept it with probability $\exp\{E(\mathbf{x}) - E(\mathbf{x}')\}$ if $E(\mathbf{x}') > E(\mathbf{x})$. Hence each iteration of the algorithm usually decreases the energy but there is also the possibility of going uphill in energy space, which means it can escape the local minima which can trap steepest descent methods. But it must be realized that an MCMC algorithm converges to samples from the distribution $P(\mathbf{x})$ and not to a fixed states, unless we perform annealing by sampling from the distribution $\frac{1}{Z[T]}P(\mathbf{x})^{1/T}$ and letting T tend to zero. As discussed in section (III-E), annealing rates must be determined by trial and error since the theoretical bounds are too slow.

In general, MCMC is usually slow unless problem specific knowledge is used. Gibbs sampling is popular because it very simple and easy to program but can only exploit a limited amount of knowledge. Most practical applications use Metropolis-Hastings with proposal probabilities which exploit knowledge of the problem. In computer vision, data driven Markov Chain Monte Carlo (DDMCMC) [39][40] shows how effective proposal probabilities can be, but this work is beyond the scope of this chapter. For a detailed introduction to advanced MCMC methods see [25].

0.5.2 Relationship between Gibbs sampling and Belief Propagation

We now show the relationship between Gibbs sampling and belief propagation. We define an update rule on the probability distribution $\mu^t(\mathbf{x})$ (analogous to MCMC) by:

$$\mu^{t+1}(\mathbf{x}) = \sum_{\mathbf{x}'} K(\vec{x}|\mathbf{x}')\mu^t(\mathbf{x}'). \tag{0.27}$$

Observe that the fixed points of this update rule are $\mu^t(\mathbf{x}) = P(\mathbf{x})$ and that MCMC is a way to implement this equation by sampling. Substituting the Gibbs sampler into these equations (27) and marginalizing yields the update equations:

$$\mu^{t+1}(\mathbf{x}_r) = \sum_{\mathbf{x}'_{N(r)}} P(\mathbf{x}_r|\mathbf{x}'_{N(r)})\mu^t(\vec{x}'_{N(r)}). \tag{0.28}$$

As described in [33], the pseudomarginals $b(\mathbf{x}_r)$ can be used to construct estimates of the local probability $B(\mathbf{x}_{N(r)})$ over larger subregions of the graph. Replacing $\mu(\mathbf{x}_r)$ by $b(\mathbf{x}_r)$ and $\mu(\mathbf{x}_{N(r)})$ by $B(\mathbf{x}_{N(r)})$ gives:

$$b^{t+1}(\mathbf{x}_r) = \sum_{\vec{x}'_{N(r)}} P(\mathbf{x}_r|\mathbf{x}'_{N(r)})B^t(\vec{x}'_{N(r)}) \ \forall r \in \Lambda_A. \tag{0.29}$$

It can be shown [33] that this corresponds to BP (by converting the update equation for the messages to an update equation on the beliefs). Hence both MFT and BP can be related to deterministic approximations to MCMC. This raises the issue about how best to combine MCMC with MFT/BP methods, which is an important topic for future research.

0.6 Discussion

This chapter described mean field theory and belief propagation techniques for performing inference "of marginals" on MRF models. We discussed how these method could be formulated in terms of minimizing free energies, such as mean field free energies and the Bethe free energies. See [43] for extensions to the Kikuchi free energy and the chapter by Weiss!! for convex free energies. We describe a range of algorithms that can be used to perform minimization. This includes steepest descent, discrete iterative algorithms, and message passing. We showed how belief propagation could be described as dynamics in the dual space of the primal problem specified by the Bethe free energy. We introduce a temperature parameter which enables inference methods to obtain MAP estimates and also motivates continuation methods, such as deterministic annealing. We briefly describe stochastic MCMC methods, such as Gibbs sampling and Metropolis-Hastings, and show that mean field algorithms and belief propagation can both be thought of as deterministic approximations to Gibbs sampling.

There have been many extensions to the basic methods described in this chapter. We refer to [2] for an entry into the literature on structured mean field methods, expectation maximization, and the trade-offs between these approaches. Other recent variants of mean field theory methods are described in [33]. Recently CCCP algorithms have been shown to be useful for learning latent structural SVMs with latent variables [44]. Work by Felzenszwalb and Huttenlocher [13] shows how belief propagation methods can be made extremely fast by taking advantage of properties of the potentials and the multi-scale properties of many vision problems. Researchers in the UAI community have discovered ways to derive generalizations of BP starting from the perspective of efficient exact inference [8]. Convex free energies introduced by Wainwright et al [42] have nicer theoretical properties that the Bethe free energy and have led to alternatives to BP, such as TRW and provably convergent algorithms– see Weiss chapter!! Stochastic sampling techniques such as MCMC remains a very active area of research, see [25] for an advanced introduction to techniques such as particle filtering which have had important applications to tracking [6]. The relationship between sampling techniques and deterministic methods is an interesting area of research and there are successful algorithms which combine both aspects. For example, there are recent nonparametric approaches which combine particle filters with belief propagation to do inference on graphical models where the variables are continuous valued [37][20]. It is unclear, however, whether the deterministic methods described in

this chapter can be extended to perform the types of inference that advanced techniques like data driven MCMC can perform [39][40].

Bibliography

[1] Y. Amit. "Modelling Brain Function: The World of Attrcator Neural Networks". Cambridge University Press. 1992.

[2] C.M. Bishop. Pattern Recognition and Machine Learning. Springer. Second edition. 2007.

[3] M. J. Black and A. Rangarajan, "On the unification of line process, outlier rejection, and robust statistics with applications in early vision", Int'l J. of Comp. Vis., Vol. 19, No. 1 pp 57-91. 1996.

[4] S. Roth and M. Black. Fields of Experts. International Journal of Computer Vision. Vol. 82. Issue 2. pp 205-229. 2009.

[5] A. Blake and A. Zisserman. Visual Reconstruction. MIT Press. 1987.

[6] A. Blake, and M. Isard: The CONDENSATION Algorithm - Conditional Density Propagation and Applications to Visual Tracking. NIPS 1996: 361-367. 1996.

[7] H. Chui and A. Rangarajan, A new point matching algorithm for non-rigid registration, Computer Vision and Image Understanding (CVIU), 89:114-141, 2003.

[8] A. Choi and A. Darwiche. A Variational Approach for Approximating Bayesian Networks by Edge Deletion. In Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence (UAI), pages 80-89, 2006.

[9] J. DeLeeuw. Applications of convex analysis to multidimensional scaling", in Barra; Brodeau, F.; Romie, G. et al., Recent developments in statistics, pp. 133145. 1977.

[10] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, B, 39, 1-38. 1977.

[11] C. Domb and M.S. Green. Eds. Phase Transitions and Critical Phenomena. Vol.2. Academic Press. London. 1972.

[12] R. Durbin, R. Szeliski and A.L. Yuille.“ An Analysis of an Elastic net Approach to the Travelling Salesman Problem”. *Neural Computation.* **1**, pp 348-358. 1989.

[13] P. Felzenszwalb and D. P. Huttenlocher. Efficient Belief Propagation for Early Vision. Proceedings of Computer Vision and Pattern Recognition. 2004.

[14] D. Geiger and A.L. Yuille. “A common framework for image segmentation”. *International Journal of Computer Vision*, Vol.6. 3:227-243. August. 1991.

[15] D. Geiger, B. Ladendorf and A.L. Yuille. “Occlusions and binocular stereo”.*International Journal of Computer Vision.* 14, pp 211-226. 1995.

[16] S. Geman and D. Geman. Stochastic relaxation, Gibbs distribution and Bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1984.

[17] T. Heskes, K. Albers and B. Kappen. Approximate Inference and Constrained Optimization. Proc. 19th Conference. Uncertainty in Artificial Intelligence. 2003.

[18] J.J. Hopfield and D.W. Tank. Neural computation of decisions in optimization problems. Biological Cybernetics. 52, pp 141-152. 1985.

[19] S. Ikeda, T. Tanaka, S. Amari. “Stochastic Reasoning, Free Energy, and Information Geometry”. Neural Computation. 2004.

[20] M. Isard, ”PAMPAS: Real-Valued Graphical Models for Computer Vision,” cvpr, vol. 1, pp.613, 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR ’03) - Volume 1, 2003.

[21] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, L. K. Saul. An Introduction to Variational Methods for Graphical Models, Machine Learning, v.37 n.2, p.183-233, Nov.1.1999

[22] S. Kirkpatrick, C. Gelatt, and M. Vecchi. Optimization by simulated annealing. Science. 220, 671-680. 1983.

[23] C.Koch, J.Marroquin and A.L.Yuille.“Analog “Neuronal” Networks in Early Vision”. *Proceedings of the National Academy of Science.* 83:pp 4263-4267. 1986.

[24] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proc. 18th International Conf. on Machine Learning, Morgan Kaufmann, San Francisco, CA. 282289. 2001.

[25] J.S. Liu. Monte Carlo Strategies in Scientific Computing. Springer. 2001.

[26] R.J. McEliece, D.J.C. MacKay, and J.F. Cheng. Turbo Decoding as an instance of Pearl's belief propagation algorithm. IEEE Journal on Selected Areas in Communication. 16(2), pp 140-152. 1998.

[27] R.M. Neal and G.E. Hinton. A view of the EM Algorithm that justifies incremental, sparse, and other variants.In M. I. Jordan (), Learning in Graphical Models (355-368). Cambridge, MA: MIT Press. 1999.

[28] M. Ohlsson, C. Peterson and A.L. Yuille. "Track Finding with Deformable Templates - The Elastic Arms Approach." *Computer Physics Communications*. 71, pp 77-98. October. 1992.

[29] G. Parisi. *Statistical Field Theory* Addison-Wesley. Reading. Ma. 1988.

[30] J. Pearl. "Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference." Morgan Kaufmann, San Mateo, CA. 1988.

[31] C. Peterson and J.R. Anderson. A mean field theory learning algorithm for neural networks. Complex Systems, 1(5), 995-1019. 1987.

[32] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. Numerical Recipes in C: The Art of Scientific Computing. Cambridge University Press. 1992.

[33] . M. Rosen-Zvi, M. I. Jordan, A. L. Yuille: The DLR Hierarchy of Approximate Inference. Uncertainty in Artificial Intelligence. 2005: 493-500.

[34] J. Rustagi. Variational Methods in Statistics. Academic Press. 1976.

[35] L. Saul and M. Jordan. Exploiting tractable substructures in intractable networks. NIPS 8, pp 486-492. 1996.

[36] J. Shotton, J. Winn, C. Rother, A. Criminisi. TextonBoost: Joint Appearance, Shape and Context Modeling for Multi-Class Object Recognition and Segmentation. In Proc. ECCV 2006.

[37] E. Sudderth, A.T. Ihler, W.T. Freeman, and A.S. Willsky. Nonparametric Belief Propagation. CVPR. pp 605-612. 2002.

[38] J. Sun, H-Y Shum, and N-N Zheng. Stereo Matching using Belief Propagation. Proc. 7th European Conference on Computer Vision. pp 510-524. 2005.

[39] Z. Tu and S-C. Zhu, Image Segmentation by Data-Driven Markov Chain Monte Carlo, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 24, No. 5, May, 2002.

[40] Z. Tu, X. Chen, A.L. Yuille and S.C. Zhu. "Image Parsing: Segmentation, Detection, and Recognition." *Int. Journal of Computer Vision.* (Marr Prize Special Edition.) (63) 2 pp 113-140. 2005.

[41] P. Viola and M. Jones. Robust Real-time Object Detection. International Journal of Computer Vision. 2001.

[42] M.J. Wainwright, T.S. Jaakkola, and A.S. Willsky. "Tree-Based Reparamterization Framework for Analysis of Sum-Product and Related Algorithms". *IEEE Transactions on Information Theory.* Vol. 49, pp 1120-1146. No. 5. 2003.

[43] J.S. Yedidia, W.T. Freeman, and Y. Weiss, "Generalized belief propagation". In *Advances in Neural Information Processing Systems 13*, pp 689-695. 2001.

[44] C-N Yu and T. Joachims. Learning Structural SVMs with Latent Variables, Proceedings of the International Conference on Machine Learning (ICML), 2009.

[45] A.L. Yuille. "Energy function for early vision and analog networks." *Biological Cybernetics.* 61, pp 115-123. June 1989.

[46] A.L. Yuille and N.M. Grzywacz. "A Mathematical Analysis of the Motion Coherence Theory." *International Journal of Computer Vision.* **3**, pp 155-175. 1989.

[47] A.L. Yuille and J.J. Kosowsky. "Statistical Physics Algorithms that Converge." Neural Computation. **6**, pp 341-356. 1994.

[48] A. L. Yuille, P. Stolorz and J. Utans. "Statistical Physics, Mixtures of Distributions and the EM Algorithm." *Neural Computation.* Vol. 6, No. 2. pp 334-340. 1994.

[49] A.L. Yuille. "CCCP Algorithms to Minimize the Bethe and Kikuchi Free Energies: Convergent Alternatives to Belief Propagation". *Neural Computation.* Vol. 14. No. 7. pp 1691-1722. 2002.

[50] A.L. Yuille and Anand Rangarajan. "The Concave-Convex Procedure (CCCP)". *Neural Computation.* 15:915-936. 2003.

[51] S. C. Zhu and D.B. Mumford. Prior Learning and Gibbs Reaction Diffusion. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.19, no.11, pp1236-1250, Nov. 1997