Model-Independent Mean Field Theory as a Local Method for Approximate Propagation of Information

M. Haft, R. Hofmann and V. Tresp*
Corporate Technology, Department: Information and Communications
Siemens AG, 81730 München, Germany

January 21, 2002

Abstract

We present a systematic approach to mean £eld theory (MFT) in a general probabilistic setting without assuming a particular model. The mean £eld equations derived here may serve as a *local* and thus very simple method for approximate inference in probabilistic models such as Boltzmann machines or Bayesian networks. "Model-independent" means that we do not assume a particular type of dependencies; in a Bayesian network, for example, we allow arbitrary tables to specify conditional dependencies. In general, there are multiple solutions to the mean £eld equations. We show that improved estimates can be obtained by forming a weighted mixture of the multiple mean £eld solutions. Simple approximate expressions for the mixture weights are given. The general formalism derived so far is evaluated for the special case of Bayesian networks. The bene£ts of taking into account multiple solutions are demonstrated by using MFT for inference in a small and in a very large Bayesian network. The results are compared to the exact results.

*E-mail: [Michael.Haft,Reimar.Hofmann,Volker.Tresp]@mchp.siemens.de

1 Introduction

The bene£ts of using a probabilistic setting in many applied £elds where uncertainty plays a prominent role –such as image processing, neural networks and arti£cial intelligence– have become increasingly apparent [1]. Unfortunately, probabilistic solutions often require involved computation [2] and further progress is closely related to the development of methods for the ef£cient handling of probability distributions. The goal of this paper is to extend the concept of using mean £eld theory (MFT) as a systematic approach for approximating probability distributions. MFT is widely used in physics, in particular, in statistical mechanics [3, 4] and has found a number of applications in other areas as well [5, 6, 7, 8]. We present MFT in a generic way in the context of graphical models, which are a general framework for dealing with uncertainty in dependency models [1, 9, 10, 11]. The use of MFT in the context of graphical models was pioneered by Jordan, Saul and Jaakola [12, 13]. In our paper we develop this approach in two new directions. First, in contrast to previous work we develop a systematic approach to MFT *without* reference to a particular model but instead work in a general probabilistic setting*. The mean £eld equations based on our rigorous formalism are new in their general form. They can be applied for example to arbitrary graphical models, which include Markov random £elds, Boltzmann machines and Bayesian networks as a special case. The main advantage of our mean £eld equations is that they provide *local* inference rules. No global operations are needed when using MFT for propagating information in large systems of interacting modules.

The second new contribution of this paper is to address the problem of multiple solutions of the mean £eld equations. Coping with multiple solutions has been originally discussed in [14] and simultaneously in [15, 16, 17]. We show that in the case of multiple solutions, a weighted mixture of these solutions leads to reasonable estimates of expected values. Approximate and very plausible mixing parameters are derived. The general formalism is applied to the special case of Bayesian networks. In this case the mixing parameters can be obtained in a consistent framework, that is, by means of only *local* computations. The bene£ts of taking into account multiple solutions of the mean £eld equations are demonstrated by using MFT for inference in a small illustration network representing a medical domain. It turns out that every solution of the mean £eld equations in this network

*In [12, 13] Jordan et al. use 'sigmoid belief nets', a network of binary variables with a particular kind of dependencies. The Boltzmann machines used in [6] are completely connected networks of binary variables with 'two-way interactions'. Here, we do not assume any particular kind of variables or a particular type of dependencies. As a consequence we may run mean £eld inference in *any* Bayesian network. At the moment we have implemented an interface to the Hugin net-£le format.

can be interpreted as a 'disease scenario'. Furthermore we run MFT in a very large network where exact methods are at their limit and we discuss drawbacks of MFT from a practical point of view.

Finally, we comment on the relevance of MFT for human reasoning. Consistent propagation of information in large networks of interacting modules is in general a demanding task and requires global operations [1]. MFT, on the other hand, suggests itself as a local and very simple prescription for communication of autonomous processors.

2 Mean Field Theory in a Probabilistic Setting

2.1 The Cross Entropy as a Measure of Distance

In the following, a set of N variables $\mathbf{X} = \{X_1, \ldots, X_N\}$ with a £nite number of discrete states $x_i \in \mathcal{H}_i$ is assumed. $P(\mathbf{X})$ denotes a probability distribution on the domain $\mathcal{H} = \mathcal{H}_1 \otimes \cdots \otimes \mathcal{H}_N$. We further assume that any distribution is strictly positive. $P(\mathbf{x})$ resp. $P(x_i)$ is the probability of the event $\mathbf{X} = \mathbf{x}$ resp. $X_i = x_i$. That is, $P(x_i)$ is a real number, $P(x_i) \in]0, 1[$. In many interesting domains, $P(\mathbf{X})$ is computationally intractable. For this reason we introduce a distribution $Q(\mathbf{X})$ which is de£ned on the same domain of variables and which incorporates some simplifying constraints. The goal is to determine $Q(\mathbf{X})$ such that –obeying these constraints– it is 'as close as possible' to the given untractable distribution $P(\mathbf{X})$. As a measure of distance between $P(\mathbf{X})$ and $Q(\mathbf{X})$ we use the cross entropy (Kullback-Leibler distance) [18]

$$\mathcal{D}(Q\|P) = \sum_{\mathbf{x}\in\mathcal{H}} Q(\mathbf{x}) \log \frac{Q(\mathbf{x})}{P(\mathbf{x})} \equiv \left\langle \log \frac{Q(\mathbf{X})}{P(\mathbf{X})} \right\rangle_{Q(\mathbf{X})} . \tag{1}$$

Note, that this distance is not symmetric in P and Q and that, with even more justi£cation, we might have used

$$\mathcal{D}(P\|Q) = \sum_{\mathbf{x}\in\mathcal{H}} P(\mathbf{x}) \log \frac{P(\mathbf{x})}{Q(\mathbf{x})} \equiv \left\langle \log \frac{P(\mathbf{X})}{Q(\mathbf{X})} \right\rangle_{P(\mathbf{X})} . \tag{2}$$

as a distance measure. There is at least one strong reasons to prefer (2) as a measure of distance. This distance measure is convex in $Q(\mathbf{X})$ in contrast to the measure (1). That means there is only one optimum with respect to $Q(\mathbf{X})$. For this optimum we have $Q(\mathbf{C}_i) = P(\mathbf{C}_i)$ for all cliques $\mathbf{C}_i$ of Q (assuming no further constraints except the cliques structure [11]). In particular, $Q(\mathbf{X}_i) = P(\mathbf{X}_i)$ for all variables of the domain. I.e., optimizing Q with respect to (2) means calculating exact marginals or doing exact inference.

The reason to use the former measure (1) nevertheless is that the expectation is with respect to the less complex, approximate distribution $Q(\mathbf{X})$ in contrast to (2), where the expectation is with respect to $P(\mathbf{X})$. As a consequence, in algorithms derived from (1) we typically £nd expectations with respect to the less complex distribution $Q(\mathbf{X})$. On the one hand, this will £nally lead us to the very simple and *local* concept of MFT. On the other hand, however, we will have to face the problem of multiple minima. Note, that though (1) has in general local minima with respect to $Q(\mathbf{X})$ any of the above two measures of distance is zero only if $Q(\mathbf{X}) = P(\mathbf{X})$.

2.2 The Mean Field Assumption

MFT is a concept from theoretical physics and is used to describe systems of many interacting particles. Many different facets of MFT can be found in £elds as different as relativistic nuclear physics [19, 20], statistical physics [3, 4, 21], image processing [7] and neural networks [22, 23, 24, 8]. As a consequence, there exist a number of ways to derive mean £eld equations. Following the above discussion we de£ne as mean £eld approximation the distribution $Q(\mathbf{X})$ which is closest to $P(\mathbf{X})$ using distance measure $\mathcal{D}(Q\|P)$. Furthermore –and this is really the heart of the mean £eld approximation [3]– we assume that the variables in the Q-distribution are *independent* variables X_i. In this case we can write

$$Q(X_1, \ldots, X_N) = \prod_{i=1}^{N} Q(X_i). \tag{3}$$

At £rst sight this ansatz seems to be much too simple. Nevertheless, one can take advantage of this approach for approximate propagation of information (evidence), as we will see later. To the best of our knowledge, Jordan et al. [12, 13] were the £rst ones to de£ne MFT in a general way as the ansatz (3) *together* with $\mathcal{D}(Q\|P)$ as a measure of distance.

2.3 General Mean Field Equations

Minimization of $\mathcal{D}(Q\|P)$ can be done in an iterative way. Suppose $Q(X_k)$, $k = 1, \ldots, N$, are our current estimates of the Q-marginals. Our goal is to obtain an improved approximation to $P(\mathbf{X})$ by minimizing $\mathcal{D}(Q\|P)$ with respect to $Q(X_i)$ thereby assuming *£xed* marginals $Q(X_j)$, $j \neq i$. Let us denote the complement of X_i by $\overline{\mathbf{X}}_i$, that is $\overline{\mathbf{X}}_i \equiv \{X_j, j \neq i\} \equiv \mathbf{X} \backslash X_i$. When minimizing $\mathcal{D}(Q\|P)$ with respect to $Q(X_i)$ we have to take into account the normalization constraint $\sum_{x_i \in \mathcal{H}_i} Q(x_i) = 1$. This can be done by using a Lagrange parameter λ, i.e., we

have to solve the equations

$$\frac{\partial}{\partial Q(x_i)} \left[\mathcal{D}(Q\|P) - \lambda \left(\sum_{x_i \in \mathcal{H}_i} Q(x_i) - 1 \right) \right] = 0 \tag{4}$$

with respect to the probabilities $Q(x_i)$, $x_i \in \mathcal{H}_i$. First, we split up $\mathcal{D}(Q\|P)$ using the relations $P(\mathbf{X}) = P(\overline{\mathbf{X}}_i)P(X_i|\overline{\mathbf{X}}_i)$ and $Q(\mathbf{X}) = Q(\overline{\mathbf{X}}_i)Q(X_i)$. Inserting these relations in (1) we obtain

$$\begin{aligned} \mathcal{D}(Q\|P) &= \left\langle \log Q(\overline{\mathbf{X}}_i) \right\rangle_{Q(\overline{\mathbf{X}}_i)} - \left\langle \log P(\overline{\mathbf{X}}_i) \right\rangle_{Q(\overline{\mathbf{X}}_i)} + & (5) \\ & \left\langle \log Q(X_i) \right\rangle_{Q(X_i)} - \left\langle \log P(X_i|\overline{\mathbf{X}}_i) \right\rangle_{Q(\mathbf{X})} . & (6) \end{aligned}$$

Only the terms in the last line (6) depend on $Q(x_i)$, those in the £rst line (5) do not. Differentiating the term $\langle \log Q(X_i) \rangle_{Q(X_i)}$ we £nd

$$\begin{aligned} \frac{\partial}{\partial Q(x_i)} \left\langle \log Q(X_i) \right\rangle_{Q(X_i)} &= \frac{\partial}{\partial Q(x_i)} \sum_{x_i \in \mathcal{H}_i} Q(x_i) \log Q(x_i) \\ &= \log Q(x_i) + 1. \end{aligned}$$

After differentiating both the second term $\left\langle \log P(X_i|\overline{\mathbf{X}}_i) \right\rangle_{Q(\mathbf{X})}$ of line (6) and the constraint of Eq. (4) we obtain

$$Q(x_i) = \frac{1}{\exp(1-\lambda)} \exp \left\langle \log P\left(x_i|\overline{\mathbf{X}}_i\right) \right\rangle_{Q(\overline{\mathbf{X}}_i)} . \tag{7}$$

The Lagrange parameter λ or normalizing constant $\exp(1-\lambda)$ can be calculated easily,

$$\exp(1-\lambda) = \sum_{x_i \in \mathcal{H}_i} \exp \left\langle \log P\left(x_i|\overline{\mathbf{X}}_i\right) \right\rangle_{Q(\overline{\mathbf{X}}_i)} . \tag{8}$$

This sum involves only $|\mathcal{H}_i|$ terms, i.e., for binary variables only two terms.

The result (7) is the unique solution to Eq. (4). It corresponds to a global minimum of $\mathcal{D}(Q\|P)$ with respect to $Q(X_i)$ given our current estimates of $Q(X_j)$, $j \neq i$. That means, updating $Q(X_i)$ according to (7) decreases $\mathcal{D}(Q\|P)$. Subsequently, we choose another variable out of $\overline{\mathbf{X}}_i$ and solve the mean £eld equations for this variable. Thus iterating repeatedly over all variables X_i we stepwise descend in $\mathcal{D}(Q\|P)$. The cross entropy $\mathcal{D}(Q\|P)$ is always positive, and, hence, this iteration ends up in a local minimum of $\mathcal{D}(Q\|P)$.

The equations (7) may be viewed as mean £eld equations in their most general form since no model assumptions were made. As a special case we now assume

that $P(\mathbf{X})$ is the Boltzmann distribution of a system of spins $x_i \in \{\pm 1\}$ de£ned by the Hamiltonian $H(\mathbf{x}) = -(1/2)\mathbf{x}^T J\mathbf{x}$ with a symmetric interaction matrix J and diagonal elements $J_{ii} = 0$. For this system the conditional distribution $P(x_i|\overline{\mathbf{x}}_i)$ reads $P(x_i|\overline{\mathbf{x}}_i) \propto \exp\left(\beta x_i J_i \cdot \overline{\mathbf{x}}_i\right)$, where J_i is the ith row of the interaction matrix J and β is the inverse temperature. Hence, for the mean £eld equations (7) we obtain

$$Q(x_i) \propto \exp\left(\beta x_i J_i \cdot \left\langle \overline{\mathbf{X}}_i \right\rangle_{Q(\overline{\mathbf{X}}_i)}\right). \tag{9}$$

For binary variables the mean values $\langle X_i \rangle$ completely determine the marginals $Q(X_i)$. In our case $x_i \in \{\pm 1\}$ we have $Q(x_i) = (1/2)(1 + x_i \langle X_i \rangle)$. Using this fact it can be shown easily that Eq. (9) leads to

$$\langle X_i \rangle_{Q(X_i)} = \tanh\left(\beta J_i \cdot \left\langle \overline{\mathbf{X}}_i \right\rangle_{Q(\overline{\mathbf{X}}_i)}\right), \tag{10}$$

which is the well-known mean £eld equation for a system of interacting spins [3], whereby the expected values $\langle X_i \rangle_{Q(X_i)}$ are usually denoted as magnetizations m_i.

2.4 Locality of Mean Field Theory

The most appealing point of MFT is that only local operations are needed for iteration of the mean £eld equation (7). Given the Markov boundary[†] $\mathbf{M}_i$ of the variable X_i the mean £eld equation (7) may be simpli£ed to

$$Q(x_i) \propto \exp\left\langle \log P\left(x_i|\mathbf{M}_i\right)\right\rangle_{Q(\mathbf{M}_i)}. \tag{11}$$

Iterating these mean £eld equations means recursively estimating marginals $Q(X_i)$ based on the current marginals $Q(X_j)$ of *only the 'neighboring'* variables $X_j \in \mathbf{M}_i$ until the system relaxes into a consistent state. For updating $Q(X_i)$ we only need the conditional distribution $P\left(X_i|\mathbf{M}_i\right)$, which can be stored 'locally at node i', and the current estimates of the marginals $Q(X_j)$, $X_j \in \mathbf{M}_i$, which can be stored at the corresponding 'neighboring nodes' of node i. All information which is needed for the renewed estimation of $Q(X_i)$ in equation (11) is thus available from node i and the neighboring nodes of node i (the Markov boundary $\mathbf{M}_i$ of node i).

[†]The Markov boundary $\mathbf{M}_i$ of a variable X_i is the minimal set of variables $\mathbf{M}_i \subset \mathbf{X}$ which makes X_i independent of the 'rest' given $\mathbf{M}_i$, i.e., $P(X_i|\mathbf{M}_i, rest) = P(X_i|\mathbf{M}_i)$. In the above physical example the Markov boundary of X_i is the set of variables X_j with $J_{ij} \neq 0$.

3 Mixing Mean Field Solutions

The iteration of the mean £eld equations (11) converges to one of typically many local minima of $\mathcal{D}(Q\|P)$. In many physical model systems, these local solutions are of particular interest since they explain phase transitions and the phenomenon of spontaneous symmetry breaking [3]. The mean £eld dynamics in a Hop£eld network converges to a local minimum of the 'free energy landscape' and thus restores *one* of many stored patterns. However, if we want to have a good approximation of a global distribution $P(\mathbf{X})$ and in particular if we are interested in expected values with respect to $P(\mathbf{X})$ we have to care about all solutions of the mean £eld equations (7). In the following we pursue the idea that instead of selecting *one particular* mean £eld solution, it might be more advantageous to form a weighted average (a mixture) of several mean £eld solutions. The mixture weights are derived in a principled way and are shown to be optimal under certain assumption. An additional bene£t is that we can relax the assumption of independent units since a mixture distribution can approximate a much larger class of distributions than the components of the mixture.

We enumerate the different mean £eld solutions by a 'hidden variable' a. That is, $Q(\mathbf{X}|a)$ now denotes a different mean £eld solutions for a different a. By assigning mixture weights $Q(a)$ to every solution we form the mixture distribution

$$Q(\mathbf{X}) = \sum_a Q(\mathbf{X}|a)Q(a). \tag{12}$$

Again, the goal now is to determine the $Q(a)$ under the constraint $\sum_a Q(a) = 1$, such that $\mathcal{D}(Q\|P)$ is minimized. It is an easy exercise to perform this optimization via a Lagrange parameter λ analogous to the previous derivation. In a few lines we obtain for all a

$$\langle \log Q(\mathbf{X}) \rangle_{Q(\mathbf{X}|a)} = \langle \log P(\mathbf{X}) \rangle_{Q(\mathbf{X}|a)} - 1 + \lambda. \tag{13}$$

We have to solve Eq. (13) for $Q(a)$, which implicitly enters the above expression via $Q(\mathbf{X})$ and Eq. (12). However, the above Eq. (13) cannot be solved in a straightforward way for $Q(a)$. With the aim of a simple expression we therefore use an additional approximation. The left hand side of (13) may be expressed as

$$\begin{aligned} \langle \log Q(\mathbf{X}) \rangle_{Q(\mathbf{X}|a)} &= \left\langle \log \left[Q(a)Q(\mathbf{X}|a) + \sum_{a' \neq a} Q(a')Q(\mathbf{X}|a') \right] \right\rangle_{Q(\mathbf{X}|a)} \\ &\approx \log Q(a) + \langle \log Q(\mathbf{X}|a) \rangle_{Q(\mathbf{X}|a)}, \end{aligned} \tag{14}$$

where we have neglected the terms $Q(a')Q(\mathbf{X}|a')$ for $a' \neq a$ in the argument of the logarithm. We may do so if $Q(\mathbf{x}|a)Q(\mathbf{x}|a') \approx 0$ for all $a \neq a'$, that is, if there is

no or suf£ciently small overlap between different mean £eld solutions. By means of this 'small-overlap' approximation in (13) we obtain for the mixture weights

$$\begin{aligned} Q(a) &\propto \exp\left[-\left\langle \log \frac{Q(\mathbf{X}|a)}{P(\mathbf{X})} \right\rangle_{Q(\mathbf{X}|a)} \right] \\ &\propto \exp\Big[-\mathcal{D}(Q\,(\mathbf{X}|a)\,\|P\,(\mathbf{X})\,)\Big]. \end{aligned} \tag{15}$$

This means, different mean £eld solutions $Q(\mathbf{X}|a)$ contribute to the global distribution $Q(\mathbf{X})$ according to their distance $\mathcal{D}\,(Q(\mathbf{X}|a)\|P(\mathbf{X}))$ to $P(\mathbf{X})$. That is a plausible result which we might have guessed. Note, however, that this nice result relies on the small-overlap approximation, i.e., on the assumption that different minima of $\mathcal{D}(Q\|P)$ are not 'close' to one another.

4 Mean Field Theory for Bayesian Networks

So far we did not make any assumptions about $P(\mathbf{X})$, and, hence, our results (the mean £eld equations (11) and the mixture weights (15)) are very general. We will now focus on a particular parameterization of a probability distribution, namely, on Bayesian networks [1, 25]. A Bayesian network has an expansion of the form

$$P(\mathbf{X}) = \prod_i P(X_i|X_1,\ldots X_{i-1}) = \prod_i P(X_i|\mathbf{\Pi}_i), \tag{16}$$

where in a typical Bayesian network every variable X_i has only a small set of 'parents' $\mathbf{\Pi}_i \subseteq \{X_1,\ldots X_{i-1}\}$. The £rst equality is valid in general; it is just the chain rule of probability. For $\mathbf{\Pi}_i \subset \{X_1,\ldots X_{i-1}\}$ in Eq. (16) the second equality corresponds to the assertion of some conditional independencies. Usually the structure of a Bayesian network is depicted as an acyclic graph where arcs point from all parent $\mathbf{\Pi}_i$ to their corresponding children X_i (see Fig. 1 later in the text as an example). The 'tables' $P(X_i|\mathbf{\Pi}_i)$ associated with the nodes X_i are the parameters of a Bayesian network.

For updating node X_i according to Eq. (11) we need to know the Markov boundary $\mathbf{M}_i$ of X_i and the conditional distribution $P(X_i|\mathbf{M}_i)$. For a Bayesian network the Markov boundary of a node is given by its parents, its children and all 'coparents', that is, all parents of all children [1]. Let $\mathcal{C}_i$ be the index set of all children of node X_i. For the conditional distribution $P(X_i|\mathbf{M}_i)$ we have

$$P(X_i|\mathbf{M}_i) \propto P(X_i|\mathbf{\Pi}_i) \prod_{k\in\mathcal{C}_i} P(X_k|\mathbf{\Pi}_k) \tag{17}$$

which can be easily derived from (16). Using this result in (11) we obtain

$$Q(x_i) \propto \exp\left[\left\langle \log P\left(x_i|\mathbf{\Pi}_i\right)\right\rangle_Q + \sum_{k\in\mathcal{C}_i}\left\langle \log P\left(X_k|\mathbf{\Pi}_k\right)\right\rangle_Q\right]. \qquad (18)$$

On the right hand side any instantiation of X_i is £xed to $X_i = x_i$, and the expected values are evaluated over the remaining variables. If compared to Eq. (11) this result greatly economizes the mean £eld updating rule. For evaluation of the expectation in (11) we have to perform a sum over the state space of the Markov boundary $\mathbf{M}_i$. In (18) we have to calculate different expectations which, however, are less expensive to evaluate for they only involve the table $P(x_i|\mathbf{\Pi}_i)$ and the tables $P(X_k|\mathbf{\Pi}_k)$, $k \in \mathcal{C}_i$.

Furthermore, note that given *any* table $P(x_i|\mathbf{\Pi}_i)$ we can exactly evaluate the expectation $\left\langle \log P\left(x_i|\mathbf{\Pi}_i\right)\right\rangle_Q$ by just performing the corresponding sum over the state space of $\mathbf{\Pi}_i$. Thus we may run mean £eld inference in *any* Bayesian network without further approximations. For nodes X_i with a large number of parents $\mathbf{\Pi}_i$, however, the evaluation of the expectation $\left\langle \log P\left(x_i|\mathbf{\Pi}_i\right)\right\rangle_Q$ is expensive. In practice large tables very often have a simple structure, e.g., by assuming a noisy-OR gate. Only rarely all degrees of freedom of a large table are needed. One should of course try to exploit the structure of a large table to calculate the expectation $\left\langle \log P\left(x_i|\mathbf{\Pi}_i\right)\right\rangle_Q$ more ef£ciently. As an example see the second illustration of mean £eld inference in the next section, where we exploit a tree-like structure of the tables. In [12] Saul et al. use an additional approximation to evaluate corresponding terms in their case of a sigmoid belief network.

It remains to be shown that in the case of a Bayesian network even the mixture weights (15) can be calculated in an ef£cient way by means of *only local computations*. If we use the expansion (16) we obtain

$$\mathcal{D}\left(Q(\mathbf{X}|a)\|P(\mathbf{X})\right) = \sum_i \left\langle \log \frac{Q(X_i|a)}{P(X_i|\mathbf{\Pi}_i)} \right\rangle_{Q(X_i,\mathbf{\Pi}_i|a)}. \qquad (19)$$

Every term in the sum on the right hand side requires only local information, i.e., only the conditional distribution $P(X_i|\mathbf{\Pi}_i)$ and the distribution $Q(X_i,\mathbf{\Pi}_i|a) = Q(X_i|a)Q(\mathbf{\Pi}_i|a)$. $P(X_i|\mathbf{\Pi}_i)$ and $Q(X_i|a)$ are properties of X_i, i.e. , they can be stored locally at node i. $Q(X_j|a)$, $X_j \in \mathbf{\Pi}_i$, describes neighboring nodes of node X_i.

Thus, for Bayesian networks, for instance, we £nd a very simple computational scheme. In many other cases it might be computationally more expensive to perform the expectation in the updating rule (11) and to compute the distance $\mathcal{D}\left(Q(\mathbf{X}|a)\|P(\mathbf{X})\right)$ in (15) to obtain the mixture weights $Q(a)$. Mean £eld inference as formulated in this section directly refers to the parameters of a Bayesian

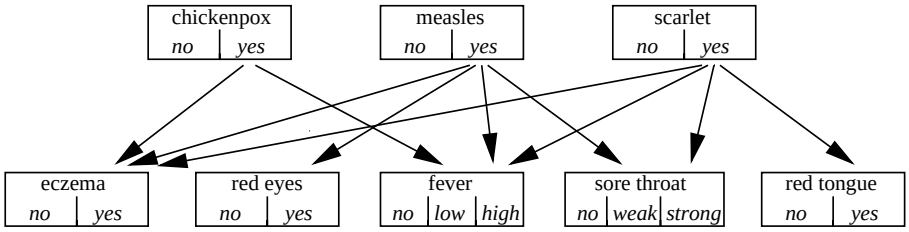

Figure 1: Our Bayesian network example for illustration of mean £eld inference. The network is modeling three children's diseases (chickenpox, measles and scarlet). The arcs are pointing from diseases to symptoms (eczema, red eyes, fever, sore throat and red tongue), that is, from cause to effect. Arces point from the positiv £nding of a desease to that state of a symptom which is typically present given the desease [26]. Note, that the variables are not just binary. Plausible values for the conditional probabilities of that network have been estimated by consulting a text book on children's diseases.

network (namely to the tables $P(X_i|\mathbf{\Pi}_i)$; see the update equation (18) and the distance (19)). There is no intermediate redundant representation of the Bayesian network, such as a junction tree [25].

5 Illustration of Mean Field Inference

5.1 A Small Network Example

Quite a bit of theory has been presented so far. It is now time to show how things work in practice. In particular, we want to demonstrate the bene£ts of mixing multiple mean £eld solutions. A simple Bayesian network for illustration purposes is depicted in Fig. 1. The goal of this network is to support medical diagnosis. In our simple example we just want to discern between measles, chickenpox and scarlet fever.

Suppose a patient complains about an eczema and a weakly sore throat. We enter that piece of knowledge into the corresponding nodes. Our goal is to obtain probabilities for the remaining nodes, in particular, for the disease nodes. For that reason we use the discussed mean £eld ansatz for the remaining nodes, i.e. , we iterate the mean £eld equations for the remaining nodes. For our illustration network we £nd two different solutions of the mean £eld equations. These two solutions, the corresponding mixture distribution and the exact probabilities are compared in table 1.

The £rst solution may be termed as 'measles scenario' the other solution is the 'scarlet scenario'[‡]. Thus, mean £eld mixture model supplies us not only with beliefs for the unknown nodes; we obtain additional information about the character of the exact distribution $P(\mathbf{X}|evidence)$ as well, namely that the joint distribution is approximately a composition of two modes. Based on these two modes we may easily calculate approximate joint probabilities for any set of nodes; see for example table 2. The two modes mainly differ in the belief for the node 'red eyes'. To obtain a unique diagnosis a natural question therefore is: 'Does the patient have red eyes?' Suppose his eyes are red. Propagating that evidence by iterating the mean £eld equations for all still unknown nodes we £nd that there is only one solution left, the measles scenario. Our £nal belief for measles is 0.99, that for chickenpox is 0.03.

	£rst MF-solution 'measles scenario' $Q(a) = 0.68$	second MF-solution 'scarlet scenario' $Q(a) = 0.32$	marginals of the MF-mixture distribution	marginals of the exact distribution
measles	0.996	0.008	0.679	0.641
scarlet	0.008	0.985	0.322	0.301
chickenpox	0.030	0.031	0.030	0.054
red eyes	0.903	0.052	0.630	0.598
red tongue	0.031	0.695	0.244	0.240
low fever	0.257	0.257	0.257	0.258
high fever	0.551	0.547	0.550	0.527

Table 1: Marginal probabilities of MFT as compared to the exact results. The £rst two columns show that any single mean £eld solution on its own results in a very poor approximation of the exact marginals.

5.2 A Large Network Example

One of the main motivations for using mean £eld inference instead of exact inference is that mean £eld inference can handle networks which are too large for exact inference. For this second experiment we chose a network size which can still be handled by the exact inference algorithms of the commercial inference engine HUGIN, but which is close to the limits. This way we can still compare the results.

Our network (which also is for illustration purposes rather than a real world application) is depicted in Fig. 2. The network is a simple model for the temporal

[‡]You can compare these two solutions with the two solutions 'all spins up' and 'all spins down' in a ferro magnet below the Curie temperature.

		scarlet	
		no	yes
measles	no	P: 0.067 Q: 0.008	P: 0.292 Q: 0.313
	yes	P: 0.623 Q: 0.671	P: 0.017 Q: 0.008

Table 2: Joint probability table of the mean £eld mixture distribution (Q) as compared to the exact results (P). Plain MFT is based on the assumption of independent variables (3) and, hence, cannot easily explain joint tables. This example shows, however, that the mixture distribution Q may give reasonable approximations to joint tables as well.

development of the state of, let's say, a telecomunication network. We assume that there are N switches connected to each other to route the calls between different users. Our Bayesian network model consists of time slices, one slice for every time step $t = 1, 2, \ldots$. In every time slice there is a node for every switch describing the state of that switch at time t. At every time step any switch may be *ok*, *slow* or *down*. In principal, the devices of the network should work in a deterministic way. However, this is at best true at a very detailed level of description. Our coarse description of the state of every switch makes modelling much easier, however necessarily involves the concept of probability. On this level of description (and in our simple demo network) the dynamics of the network is governed by a sequence of some 'probabilistic rules'. Verbally they may be formulated as follows:

- If switch i is *ok*
  - and all other switches are also working properly switch i may spontaneously become *slow* or even go *down* in the next time step with a small probability.
  - and any of the other switches is already *down* or *slow* there is an increased probability that switch i will also become *slow* or even go *down* in the next time step.
- If switch i is *slow* there is a good chance that it will recover or, on the other hand, completely go down in the next time step.
- If switch i is *down* it will be *down* in the next time step with a probability close to one.

Based on these rules the conditional probability table for every switch given the state of the system one time step ago can be most ef£ciently represented as a tree with only a small number of leaves.

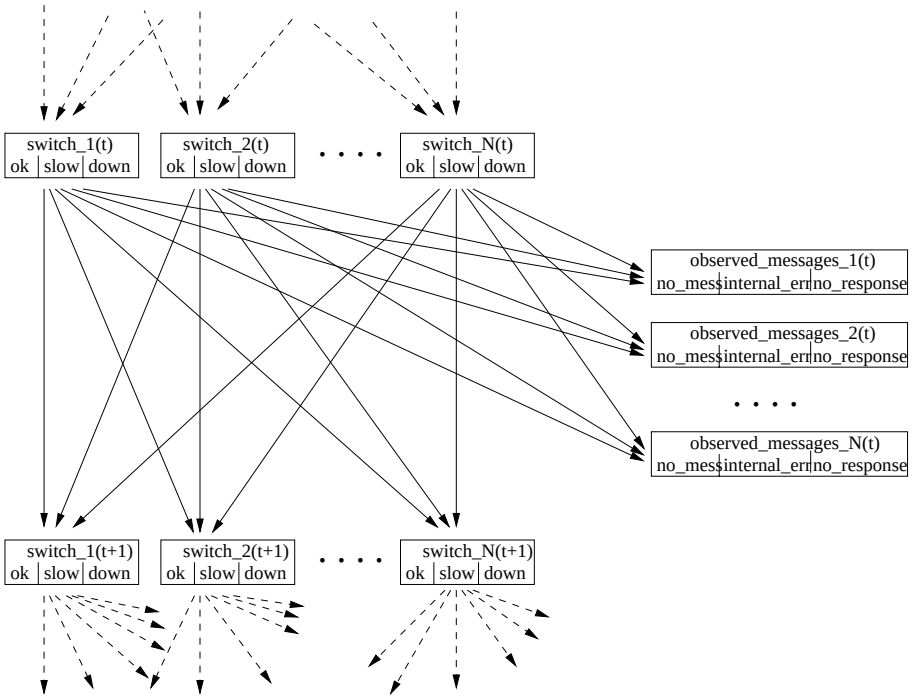

Figure 2: A Bayesian network time slice model describing the temporal development of a system of N communicating switches. The state of switch i at time $t+1$ (switch_$i(t+1)$) dependes on the state of all other switches at time t. The nodes to the right are the messages which the operator observes from every switch. See text for a detailed description.

Due to the dynamics of the network error messages typically occur in form of 'bursts'. If the network operator observes such a ¤ood of messages his task is to localize the 'root cause', that is, the switch which failed £rst. The evidence which the network operator has for every switch is:

- *no message* from that switch,
- the switch is reporting *internal errors*
- or it is reporting that there is *no response* from another switch.

A similar set of probabilistic rules describes the process of the emission of messages from the system.

Building the junction tree for the described network is possible up to $N = 8$ switches. In this case the largest cliques (containing all the switches of two successive time slices) have 16 nodes resulting in $4.3 * 10^7$ numbers for the potentials of these cliques. Running the junction tree algorithm for $N = 9$ fails on a $512MByte$ machine. Running MFT is in principal possible up to $N = 15$. However, in this case the tables associated with each node are also of the size $4.3 * 10^7$, and calculating the expected values in (18) is much too expensive to be of any practical relevance. However, using a table representation of the conditional dependencies means dealing with an enormous amount of redundant parameters. As mentioned above an adequate representation is, e.g., a tree. This structure can be easily exploited to very ef£ciently evaluate the expected values in (18). Running MFT is then possible even in networks with $N > 15$.

To get an impression of the results we compare MFT with exact methods in a network of size $N = 7$ and four succesive time slices. We draw random samples form the joint distribution and accordingly selected the state of the 'observable' nodes (the nodes 'observed_messages_xxx' in Fig. 2). Then searching for mean £eld solutions by starting the iteration at 30 different points typically resulted in about 10 different solutions whereby according to the mixture weights typically only two or three of all solutions are dominant.

Table 3 shows a typical result, which is quite uesful for root cause analysis. Occasionally we found less accurate results like that in table 4. The reason for the the difference is that in this cases we have failed to £nd all relevant solutions of the mean £eld equations. More extensive search for solutions (e.g. by starting the iteration process for 100 times) makes results like that of table 4 a very rare case. However, this is at the expense of an increased computational load. In general, it is very helpful to have a good heuristic idear for a particular domain where to start the mean £eld iteration so that the most relevant mean £eld solutions can be found very fast. For example, in diagnostic domains it is in general good strategy to start the iteration in all states where only one cause or desease has a very high probability.

Concerning the 'small overlap' asumption our experience is that only in very rare case we £nd two solutions which are close together. In particular in very large networks the mean £eld solutions typically have nearly zero overlap. In case there are two nearly identical solutions the mixture distribution will not depend crucially on the relative weighting of these two solutions. Hence, when testing wether the most recently computed solution is (within numeric bounds) identical to a solution found previously (in which case it is considered only once in the mixing procedure) we reject the new solution even if it is not identical but close to a previous solution.

		mean £eld results	exact results
switch 3, $t=0$	*slow*	0.135	0.104
	down	0.840	0.866
switch 5, $t=0$	*slow*	0.135	0.104
	down	0.840	0.866
switch 3, $t=1$	*slow*	0.020	0.019
	down	0.979	0.980
switch 5, $t=1$	*slow*	0.020	0.019
	down	0.979	0.980
switch 6, $t=2$	*slow*	0.935	0.878
	down	0.000013	0.000001
switch 6, $t=3$	*slow*	0.375	0.360
	down	0.285	0.272

Table 3: Results of a typical run of MFT with a random sample of evidence for the observable nodes in our demonstration network of Fig. 2. We have listed some switches with high failure probability. The mean £eld results are compared to exact results obtained by using HUGIN (right column). The switches 3 and 5 are probably *down* already at $t = 0$ or at least at $t = 1$. Switch 6 is probably *slow* at $t = 2$ and may be even dowm at $t = 3$.

6 Discussion

In this article we have discussed MFT in a model-independent way as a method to approximate a given probability distribution. Furthermore, we have extended the standard mean £eld approach by the idea of mixing different mean £eld solutions. Our approach is model-independent in so far as you may use the resulting mean £eld equation (7) in arbitrary probabilistic domains. The only restriction is that the variables have to be descrete. (See Frey and Hinton [27] for an example of mean £eld theory in the case of continuous hidden variables.)

As illustrated in our experiments, our approach can be used for approximate propagation of evidence (inference). Thereby, £rst, evidence is entered into the model, *then* the mean £eld approximation $P(\mathbf{X}|\text{evidence}) \approx Q(\mathbf{X}|\text{evidence}) = \prod_i Q(X_i|\text{evidence})$ is calculated. The results clearly demonstrated that reasonable probabilistic approximations can only be achieved if we take into account multiple solutions of the mean £eld equations. In doing so, we may even obtain easy interpretable information about the joint distribution of several variables.

The presented procedure (£nding solutions of the mean £eld equations (7) and

		mean £eld results	exact results
switch 6, $t=0$	*slow*	0.504	0.373
	down	0.000030	0.000068
switch 4, $t=1$	*slow*	0.517	0.362
	down	0.223	0.359
switch 5, $t=1$	*slow*	0.134	0.302
	down	0.0	0.0
switch 6, $t=1$	*slow*	0.818	0.744
	down	0.000001	0.000007
switch 7, $t=1$	*slow*	0.087	0.298
	down	0.0008	0.0067

Table 4: The 'worst' result which we found when running MFT in the network of Fig. 2. Results like this may be found in particular if one misses to £nd all relevant solutions of the mean £eld equations.

mixing them) does not optimize the parameters $Q(a)$ and $Q(\mathbf{X}|a)$ of the approximating distribution $Q(\mathbf{X})$ simultaneously since the different solutions $Q(\mathbf{X}|a)$ of the mean £eld equations (7) for different a are determined independently and prior to determining the mixture weights $Q(a)$. It might be possible to derive a more re£ned simultaneous optimization of the parameters $Q(a)$ and $Q(\mathbf{X}|a)$. However, the resulting equations will not be as simple as (11) and (15). Their simplicity and *locality* (!) justi£es the above step by step procedure and the introduced small-overlap approximation. When used for inference in graphical models, MFT exploits the structure of a graphical model even in non tree-like graphs since, as discussed previously, only 'neighboring nodes' have to communicate. This locality is the appealing point of MFT. There is no necessity to compile the original graph to a tree-like cover model as it is done by the junction tree algorithm by means of moralization and triangulation [9, 10]. Loops in the original graph may lead to an exponential complexity for exact inference methods (as, e.g., in our illustration in section 5.2), however, are of minor relevance for MFT. In particular in the case of Bayesian networks, mean £eld inference exhibits further simpli£cations. An additional advantage is that in many cases the existence of several mean £eld solutions sheds a light on the structure of the exact distribution In our example the exact distribution could be interpreted as being composed of two 'scenarios'.

Thus, in many domains MFT represents an interesting complement to other inference methods. However, as the last example may suggests, in some other domains mean £eld theory may not be appropriate. The weak point of mean £eld

inference is that a good result depends in particular on £nding all relevant solutions of the mean £eld equations or the best minima of the cross entropy (1). Though the mean £eld iteration descendes to the next minimum of the cross entropy very fast there may in some domains exist many solutions. Our experience is that mean £eld theory can be used in domains where

- there are not too many deterministic or nearly deterministic dependencies among the variables of a domain, which makes the cross entropy (1) a very rough landscape with many local minima.
- there is much evidence available which rules out many of the local minima.
- you have a good heuristic idea where to start iterating the mean £eld equations.

Finally, a few words on human reasoning are appropriate. In his book "Probabilistic Reasoning in Intelligent Systems" Pearl argues that '... *any viable model of human reasoning should be able to perform this task* (consistent propagation of information) *with a self-activated propagation mechanism, i.e., with an array of simple autonomous processors, communication locally via the links provided by the network itself. The impact of each new piece of evidence is viewed as a perturbation that propagates through the network via message-passing between neighbouring variables, with a minimal external supervision.*' Mean £eld inference exactly meets these demands. As a consequence mean £eld inference permits a signi£cant amount unsupervized parallelism, which is ascribed to the human way of information processing. Furthermore, arguing in terms of 'scenarios' is much closer to the human way of reasoning then global probabilistic calculations. Mean £eld inference even re¤ects this way of arguing.

References

[1] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, San Francisco, 2 edition, 1988.

[2] G. Cooper. Computational complexity of probabilistic inference using bayesian belief networks. *Arti£cial Intelligence*, 42:393–405, 1990.

[3] J. J. Binney, N. J. Dowrick, A. J. Fisher, and M. E. J. Newman. *The Theory of Critical Phenomena*. Oxford University Press, New York, 3 edition, 1995.

[4] G. Parisi. *Statistical Field Theory*. Addison-Wesley, Reading, MA, 1988.

[5] C. Peterson and J. R. Anderson. A mean £eld theory learning algorithm for neural networks. *Complex Systems*, 1:995–1019, 1987.

[6] C. Peterson and E. Hartman. Explorations of the mean £eld theory learning algorithm. *Neural Networks*, 2:475–494, 1989.

[7] T. Hofmann and J. Buhmann. Pairwise data clustering by deterministic annealing. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(1):1–14, 1997.

[8] M. Haft. Robust 'topological' codes by keeping control of internal redundancy. *Phys. Rev. Letters*, 81(18):4016–4019, 1998.

[9] S. L. Lauritzen and D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their applicatin to expert systems. *J. Roy. Statist. Soc. B*, 50:154–227, 1988.

[10] F. V. Jensen, S. L. Lauritzen, and K. G. Olsen. Bayesian updating in causal probabilistic networks by local computations. *Computational Statistics Quaterly*, 4:296–282, 1990.

[11] J. Wittaker. *Graphical Models in Applied Multivariate Statistics*. Wiley, New York, 1990.

[12] L. K. Saul and M. I. Jordan. Exploiting tractrable substructures in untractable networks. In D. Touretzky, M. Moser, and M. Hasselmo, editors, *Advances in Neural Information Processing Systems: Proceedings of the 1995 Conference*, pages 486–492, Cambridge MA, 1995. MIT Press.

[13] K. L. Saul, T. Jaakkola, and M. I. Jordan. Mean £eld theory for signoide belief networks. *Journal of Arti£cial Intelligence Research*, 4:61–76, 1996.

[14] M. Haft, R. Hofmann, and V. Tresp. Model-independent mean £eld theory as a local method for approximate propagation of information. *Technical Report, http://www7.informatik.tu-muenchen.de/ hofmannr/mf_abstr.html*, 1997.

[15] C. M. Bishop, T. Lawrence, N. D.and Jaakkola, and M. I. Jordan. Approximating posterior distributions in belief networks using mixtures. In M. I. Jordan, M. J. Kearns, and S. A. Solla, editors, *Advances in Neural Information Processing Systems 10, Proceedings of the 1997 Conference*, pages 416–422. MIT Press, Cambridge MA, 1998.

[16] N. D. Lawrence, C. M. Bishop, and M. I. Jordan. Mixture representations for inference and learning in boltzmann machines. In G. F. Cooper and S. Moral,

editors, *Uncertainty in Arti£cial Intelligence: Proceedings of the Fourteenth Conference*, pages 320–327. Morgan Kaufmann, San Francisco, CA, 1998.

[17] T. S. Jaakkola and M. I. Jordan. Improving the mean £eld approximation via the use of mixture distributions. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 163–173. Kulwer, 1998.

[18] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, New York, 1991.

[19] B. D. Serot and J. D. Walecka. *Advances in Nuclear Physics Vol. 16: The Relativistic Nuclear Many Body Problem*. Plenum Press, New York, 1986.

[20] N. K. Glendenning, D. von-Eiff, M. Haft, H. Lenske, and M. K. Weigel. Relativistic mean £eld calculations of Λ- and Σ-hypernuclei. *Phys. Rev. C*, 48:889–895, 1993.

[21] K. H. Fischer and J. A. Hertz. *Spin Glasses*. Cambridge University Press, 1991.

[22] J. L. van Hemmen and R. Kühn. Nonlinear neural networks. *Phys. Rev. Letters*, 57(7):913–916, 1986.

[23] J. L. van Hemmen and R. Kühn. Collective phenomena in neural networks. In E. Domany, J. L. van Hemmen, and K. Schulten, editors, *Models of Neural Networks I*, pages 1–113. Springer, New York, 1991.

[24] J. A. Hertz, A. Krogh, and R. G. Palmer. *Introduction to the theory of neural computation*. Addison-Wesley, 1991.

[25] V. J. Jensen. *An Introduction to Bayesian Networks*. UCL Press, London, 1996.

[26] R. Hofmann and M. Haft. De£ning structure *and* conditional probabilities of Bayesian networks in graphical form. *IEEE Trans. Knowledge an Data Engineering*, 1999. under review.

[27] B. J. Frey and G. E. Hinton. Variational learning in non-linear Gaussian belief networks. *Neural Computation*, 11(1):193–213, 1999. available at http://www.cs.utoronto.ca/ frey.