This document contains proofs and other supplemental information for the UAI 2012 submission, "Belief Propagation for Structured Decision Making".

A Randomized v.s. Deterministic Strategies

It is a well-known fact in decision theory that no randomized strategy can improve on the utility of the best deterministic strategy, so that:

Lemma 2.1. *For any ID,* $\max_{\boldsymbol{\delta}\in\Delta} EU(\boldsymbol{\delta}) = \max_{\boldsymbol{\delta}\in\Delta^o} EU(\boldsymbol{\delta})$.

Proof. Since $\Delta^o \subset \Delta$, we need to show that for any randomized strategy $\boldsymbol{\delta} \in \Delta$, there exists a deterministic strategy $\boldsymbol{\delta}' \in \Delta^o$ such that $\mathrm{EU}(\boldsymbol{\delta}) \le \mathrm{EU}(\boldsymbol{\delta}')$. Note that

$$\mathrm{EU}(\boldsymbol{\delta}) = \sum_{\boldsymbol{x}} q(\boldsymbol{x}) \prod_{i\in D} p_i^{\delta}(x_i|x_{\mathrm{pa}(i)}),$$

Thus, $\mathrm{EU}(\boldsymbol{\delta})$ is linear on $p_i^{\delta}(x_i|x_{\mathrm{pa}(i)})$ for any $i \in D$ (with all the other policies fixed); therefore, one can always replace $p_i^{\delta}(x_i|x_{\mathrm{pa}(i)})$ with some deterministic $p_i^{\delta'}(x_i|x_{\mathrm{pa}(i)})$ without decreasing $\mathrm{EU}(\boldsymbol{\delta})$. Doing so sequentially for all $i \in D$ yields to a deterministic rule $\boldsymbol{\delta}'$ with $\mathrm{EU}(\boldsymbol{\delta}) \le \mathrm{EU}(\boldsymbol{\delta}')$. □

One can further show that any (globally) optimal randomized strategy can be represented as a convex combination of a set of optimal deterministic strategies.

B Duality form of MEU

Here we give a proof of our main result.

Theorem 3.1. *(a). For an influence diagram with augmented distribution* $q(\boldsymbol{x}) \propto \exp(\theta(\boldsymbol{x}))$*, its log maximum expected utility* $\log \mathrm{MEU}(\boldsymbol{\theta})$ *equals*

$$\max_{\boldsymbol{\tau}\in\mathbb{M}} \{\langle \boldsymbol{\theta}, \boldsymbol{\tau}\rangle + H(\boldsymbol{x};\boldsymbol{\tau}) - \sum_{i\in D} H(x_i|x_{\mathrm{pa}(i)};\boldsymbol{\tau})\}. \quad (1)$$

Suppose $\boldsymbol{\tau}^*$ *is a maximum of* (1)*, then* $\boldsymbol{\delta}^* = \{\boldsymbol{\tau}^*(x_i|x_{\mathrm{pa}(i)})|i \in D\}$ *is an optimal strategy.*

(b). Under the perfect recall assumption, (1) *reduces to*

$$\max_{\boldsymbol{\tau}\in\mathbb{M}} \{\langle \boldsymbol{\theta}, \boldsymbol{\tau}\rangle + \sum_{o_i\in C} H(x_{o_i}|x_{o_{1:i-1}};\boldsymbol{\tau})\} \quad (2)$$

where $o_{1:i-1} = \{o_j|j = 1,\ldots,i-1\}$.

Proof. (a). Let $q^{\delta}(\boldsymbol{x}) = q(\boldsymbol{x}) \prod_{i\in D} p_i^{\delta}(x_i|x_{\mathrm{pa}(i)})$. We apply the standard duality result (1) of partition function on $q^{\delta}(\boldsymbol{x}) \propto \exp(\theta^{\delta}(\boldsymbol{x}))$,

$$\begin{aligned}\log \mathrm{MEU} &= \max_{\boldsymbol{\delta}} \log \sum_x \exp(\theta^{\delta}(\boldsymbol{x})) \\ &= \max_{\boldsymbol{\delta}} \{ \max_{\boldsymbol{\tau}\in\mathbb{M}} [\langle \boldsymbol{\theta}^{\delta}, \boldsymbol{\tau}\rangle + H(\boldsymbol{x};\boldsymbol{\tau})]\} \\ &= \max_{\boldsymbol{\tau}\in\mathbb{M}} \{ \max_{\boldsymbol{\delta}} [\langle \boldsymbol{\theta}^{\delta}, \boldsymbol{\tau}\rangle] + H(\boldsymbol{x};\boldsymbol{\tau})\}, \quad (3)\end{aligned}$$

and we have

$$\begin{aligned}&\max_{\boldsymbol{\delta}} \{\langle \boldsymbol{\theta}^{\delta}, \boldsymbol{\tau}\rangle\} \\ &= \max_{\boldsymbol{\delta}} \{\langle \boldsymbol{\theta} + \sum_{i\in D} \log p_i^{\delta}(x_i|x_{\mathrm{pa}(i)}), \boldsymbol{\tau}\rangle\} \\ &= \langle \boldsymbol{\theta}, \boldsymbol{\tau}\rangle + \sum_{i\in D} \max_{p_i^{\delta}} \{\sum_x \tau(\boldsymbol{x}) \log p_i^{\delta}(x_i|x_{\mathrm{pa}(i)})\} \\ &\overset{*}{=} \langle \boldsymbol{\theta}, \boldsymbol{\tau}\rangle + \sum_{i\in D} \{\sum_x \tau(\boldsymbol{x}) \log \tau(x_i|x_{\mathrm{pa}(i)})\} \\ &= \langle \boldsymbol{\theta}, \boldsymbol{\tau}\rangle - \sum_{i\in D} H(x_i|x_{\mathrm{pa}(i)};\boldsymbol{\tau}), \quad (4)\end{aligned}$$

where the equality "$\overset{*}{=}$" holds because the solution of $\max_{p_i^{\delta}} \{\sum_x \tau(\boldsymbol{x}) \log p_i^{\delta}(x_i|x_{\mathrm{pa}(i)})\}$ subject to the normalization constraint $\sum_{x_i} p_i^{\delta}(x_i|x_{\mathrm{pa}(i)}) = 1$ is $p_i^{\delta} = \tau(x_i|x_{\mathrm{pa}(i)})$. We obtain (1) by plugging (4) into (3).

(b). By the chain rule of entropy, we have

$$H(\boldsymbol{x};\boldsymbol{\tau}) = \sum_i H(x_{o_i}|x_{o_{1:i-1}};\boldsymbol{\tau}). \quad (5)$$

Note that we have $\mathrm{pa}(i) = o_{1:i-1}$ for $i \in D$ for ID with perfect recall. The result follows by substituting (5) into (1). □

The following lemma is the "dual" version of Lemma 2.1; it will be helpful for proving Corollary 3.2 and Corollary 3.3.

Lemma B.1. *Let* $\mathbb{M}^o$ *be the set of distributions* $\tau(\boldsymbol{x})$ *in which* $\tau(x_i|x_{\mathrm{pa}(i)}), i \in D$ *are deterministic. Then the optimization domain* $\mathbb{M}$ *in* (1) *of Theorem 3.1 can be replaced by* $\mathbb{M}^o$ *without changing the result, that is,* $\log \mathrm{MEU}(\boldsymbol{\theta})$ *equals*

$$\max_{\boldsymbol{\tau}\in\mathbb{M}^o} \{\langle \boldsymbol{\theta}, \boldsymbol{\tau}\rangle + H(\boldsymbol{x};\boldsymbol{\tau}) - \sum_{i\in D} H(x_i|x_{\mathrm{pa}(i)};\boldsymbol{\tau})\}. \quad (6)$$

Proof. Note that $\mathbb{M}^o$ is equivalent to the set of deterministic strategies Δ^o. As shown in the proof of Lemma 2.1, there always exists optimal deterministic strategies, that is, at least one optimal solutions of (1) falls in $\mathbb{M}^o$. Therefore, the result follows. □

Corollary 3.2. *For an ID with parameter* $\boldsymbol{\theta}$*, we have*

$$\log \mathrm{MEU} = \max_{\boldsymbol{\tau}\in\mathbb{I}} \{\langle \boldsymbol{\theta}, \boldsymbol{\tau}\rangle + \sum_{i\in C} H(x_i|x_{o_{1:i-1}};\boldsymbol{\tau})\} \quad (7)$$

where $\mathbb{I} = \{\boldsymbol{\tau} \in \mathbb{M}\colon x_{o_i} \perp x_{o_{1:i-1}\setminus \mathrm{pa}(o_i)} | x_{\mathrm{pa}(o_i)}\}$*, corresponding to the distributions respecting the imperfect recall structures; "*$x \perp y \,|\, z$*" means that* x *and* y *are conditionally independent given* z*.*

Proof. For any $\boldsymbol{\tau} \in \mathbb{I}$, we have $H(x_i|x_{\mathrm{pa}(i)};\boldsymbol{\tau}) = H(x_i|x_{o_{1:i-1}};\boldsymbol{\tau})$, hence by the entropic chain rule, the objective function in (7) is the same as that in (1).

Then, for any $\boldsymbol{\tau} \in \mathbb{M}^o$ and $o_i \in D$, since $\tau(x_{o_i}|x_{\mathrm{pa}(o_i)})$ is deterministic, we have $0 \le H(x_{o_i}|x_{o_{1:i-1}}) \le H(x_{o_i}|x_{\mathrm{pa}(o_i)}) = 0$, which implies $I(x_{o_i}; x_{o_{1:i}\setminus\mathrm{pa}(o_i)}|x_{\mathrm{pa}(o_i)}) = 0$, and hence $\mathbb{M}^o \subseteq \mathbb{I} \subseteq \mathbb{M}$. We thus have that the LHS of (7) is no larger than (1), while no smaller than (6). The result follows since (1) and (6) equal by Lemma B.1.

□

Corollary 3.3. *For any* ϵ*, let* $\boldsymbol{\tau}^*$ *be an optimum of*

$$\max_{\boldsymbol{\tau}\in\mathbb{M}} \{\langle \boldsymbol{\theta}, \boldsymbol{\tau}\rangle + H(\boldsymbol{x}) - (1-\epsilon)\sum_{i\in D} H(x_i|x_{\mathrm{pa}(i)})\}. \quad (8)$$

If $\boldsymbol{\delta}^* = \{\tau^*(x_i|x_{\mathrm{pa}(i)})|i \in D\}$ *is an deterministic strategy, then it is an optimal strategy of the MEU.*

Proof. First, we have $H(x_i|x_{\mathrm{pa}(i)};\boldsymbol{\tau}) = 0$ for $\boldsymbol{\tau} \in \mathbb{M}^o$ and $i \in D$, since such $\tau(x_i|x_{\mathrm{pa}(i)})$ are deterministic. Therefore, the objective functions in (8) and (6) are equivalent when the maximization domains are restricted on $\mathbb{M}^o$. The result follows by applying Lemma B.1. □

B.1 Derivation of Belief Propagation for MEU

Eq. (11) is similar to the objective of sum-product junction graph BP, except the entropy terms of the decision clusters are replaced by $\hat{H}_\epsilon(x_{c_k})$, which can be thought of as corresponding to some local MEU problem. In the sequel, we derive a similar belief propagation algorithm for (11), which requires that the decision clusters receive some special consideration. To demonstrate how this can be done, we feel it is helpful to first consider a local optimization problem associated with a single decision cluster.

Lemma B.2. *Consider a local optimization problem on decision cluster* c_k*,*

$$\max_{\tau_{c_k}} \{\langle \vartheta_{c_k}, \tau_{c_k}\rangle + H_\epsilon(x_{c_k};\tau_{c_k})\}.$$

Its solution is,

$$\tau_{c_k}(x_{c_k}) \propto \sigma_k[b_{c_k}(x_{c_k}), \epsilon] \overset{def}{=} b(x_{c_k}) b_\epsilon(x_{d_k}|x_{\mathrm{pa}(d_k)})^{1-\epsilon}$$

where $b_{c_k}(x_{c_k}) \propto \exp(\vartheta_{c_k}(x_{c_k}))$ *and* $b_\epsilon(x_{d_k}|x_{\mathrm{pa}(d_k)})$ *is the "annealed" conditional probability of* b_{c_k}*,*

$$b_\epsilon(x_{d_k}|x_{\mathrm{pa}(d_k)}) = \frac{b(x_{d_k}, x_{\mathrm{pa}(d_k)})^{1/\epsilon}}{\sum_{x_{d_k}} b(x_{d_k}, x_{\mathrm{pa}(d_k)})^{1/\epsilon}},$$

$$b(x_{d_k}, x_{\mathrm{pa}(d_k)}) = \sum_{x_{z_k}} b(x_{c_k}), \qquad z_k = c_k \setminus \{d_k, \mathrm{pa}(d_k)\}.$$

Proof. The Lagrangian function is

$$\langle \vartheta_{c_k}, \tau_{c_k}\rangle + H_\epsilon(x_{c_k};\tau_{c_k}) + \lambda \sum_{x_{c_k}} [\tau_{c_k}(x_{c_k}) - 1].$$

Its stationary point satisfies

$$\vartheta_{c_k}(x_{c_k}) - \log \tau_{c_k}(x_{c_k}) + (\epsilon - 1)\log \tau_{c_k}(x_{d_k}|x_{\mathrm{pa}(d_k)}) + \lambda,$$

or equivalently,

$$\tau_{c_k}(x_{c_k})[\tau_{c_k}(x_{d_k}|x_{\mathrm{pa}(d_k)})]^{\epsilon-1} = b_{c_k}(x_{c_k}). \quad (9)$$

Summing over x_{z_k} on both side of (9), we have

$$\tau_{c_k}(x_{\mathrm{pa}(d_k)})[\tau_{c_k}(x_{d_k}|x_{\mathrm{pa}(d_k)})]^{\epsilon} = b_{c_k}(x_{d_k}, x_{\mathrm{pa}(d_k)}), \quad (10)$$

Raising both sides of (9) to the power $1/\epsilon$ and summing over x_{d_k}, we have

$$[\tau_{c_k}(x_{x_{\mathrm{pa}(d_k)}})]^{1/\epsilon} = \sum_{x_{d_k}} [b_{c_k}(x_{d_k}, x_{\mathrm{pa}(d_k)})]^{1/\epsilon}. \quad (11)$$

Combining (11) with (10), we have

$$\tau_{c_k}(x_{d_k}|x_{\mathrm{pa}(d_k)}) = b_\epsilon(x_{d_k}|x_{\mathrm{pa}(d_k)}). \quad (12)$$

Finally, combining (12) with (9) gives

$$\tau_{c_k}(x_{c_k}) = b_{c_k}(x_{c_k}) b_\epsilon(x_{d_k}|x_{\mathrm{pa}(d_k)})^{1-\epsilon}. \quad (13)$$

□

The operator $\sigma_k[b(x_c);\epsilon]$ can be treated as imputing $b(x_c)$ with an "annealed" policy defined as $b_\epsilon(x_{d_k}|x_{\mathrm{pa}(d_k)})$; this can be seen more clearly in the limit as $\epsilon \to 0^+$.

Lemma B.3. *Consider a local MEU problem of a single decision node* d_k *with parent nodes* $\mathrm{pa}(d_k)$ *and an augmented probability* $b_{c_k}(x_{c_k})$*; let*

$$b^*(x_{d_k}|x_{\mathrm{pa}(d_k)}) = \lim_{\epsilon\to 0^+} b_\epsilon(x_{d_k}|x_{\mathrm{pa}(d_k)}), \quad \forall d_k \in D,$$

then $\boldsymbol{\delta}^* = \{b^*(x_{d_k}|x_{\mathrm{pa}(d_k)})\colon d_k \in D\}$ *is an optimal strategy.*

Proof. Let

$$\delta^*_{d_k}(x_{\mathrm{pa}(d_k)}) = \arg\max_{x_{d_k}} \{b_\epsilon(x_{d_k}|x_{\mathrm{pa}(d_k)})\},$$

One can show that as $\epsilon \to 0^+$,

$$b^*(x_{d_k}|x_{\mathrm{pa}(d_k)}) = \begin{cases} 1/|\delta^*_{d_k}| & \text{if } x_{d_k} \in \delta^*_{d_k} \\ 0 & \text{if otherwise,} \end{cases} \quad (14)$$

thus, $b^*(x_{d_k}|x_{\mathrm{pa}(d_k)})$ acts as a "maximum operator" of $b(x_{d_k}|x_{\mathrm{pa}(d_k)})$, that is,

$$\sum_{x_{d_k}} b(x_{d_k}|x_{\mathrm{pa}(d_k)}) b^*(x_{d_k}|x_{\mathrm{pa}(d_k)}) = \max_{x_{d_k}} b(x_{d_k}|x_{\mathrm{pa}(d_k)}).$$

Therefore, for any $\boldsymbol{\delta} \in \Delta$, we have

$$\begin{aligned} \mathrm{EU}(\boldsymbol{\delta}) &= \sum_{x_{c_k}} b_{c_k}(x_{c_k}) b^{\delta}(x_{d_k}|x_{\mathrm{pa}(d_k)}) \\ &= \sum_{x_{\mathrm{pa}(d_k)}} b(x_{\mathrm{pa}(d_k)}) \sum_{x_{d_k}} b(x_{d_k}|x_{\mathrm{pa}(d_k)}) b^{\delta}(x_{d_k}|x_{\mathrm{pa}(d_k)}) \\ &\le \sum_{x_{\mathrm{pa}(d_k)}} b(x_{\mathrm{pa}(d_k)}) \max_{x_{d_k}} b(x_{d_k}|x_{\mathrm{pa}(d_k)}) \\ &= \sum_{x_{\mathrm{pa}(d_k)}} b(x_{\mathrm{pa}(d_k)}) \sum_{x_{d_k}} b(x_{d_k}|x_{\mathrm{pa}(d_k)}) b^*(x_{d_k}|x_{\mathrm{pa}(d_k)}) \\ &= \mathrm{EU}(\boldsymbol{\delta}^*). \end{aligned}$$

This concludes the proof. □

Therefore, at zero temperature limit, the $\sigma_k[\cdot]$ operator in MEU-BP (12)-(13) can be directly calculated via (14), avoiding the necessity for power operations.

We now derive the MEU-BP in (12)-(13) for solving (11) using a Lagrange multiplier method similar to Yedidia et al. [2005]. Consider the Lagrange multiplier of (8),

$$\begin{aligned} &\langle \boldsymbol{\theta}, \boldsymbol{\tau} \rangle + \sum_{k \in \mathcal{R}} H_{c_k} + \sum_{k \in \mathcal{D}} H^{\epsilon}_{c_k} - \sum_{(kl) \in \mathcal{E}} H_{s_{kl}} + \\ &\sum_{(kl) \in \mathcal{E}} \sum_{x_{c_k \setminus s_{kl}}} \lambda_{s_{k\to l}}(x_{s_{kl}}) [\sum_{x_{s_{kl}}} \tau_{c_k}(x_{c_k}) - \tau_{s_{kl}}(x_{s_{kl}})], \end{aligned}$$

where the nonnegative and normalization constraints are not included and are dealt with implicitly. Taking its gradient w.r.t. τ_{c_k} and $\tau_{s_{kl}}$, one has

$$\tau_{c_k} \propto \psi_{c_k} m_{\sim k} \quad \textit{for normal clusters,} \quad (15)$$

$$\tau_{c_k} \propto \sigma_k[\psi_{c_k} m_{\sim k}; \epsilon] \quad \textit{for decision clusters,} \quad (16)$$

$$\tau_{s_{kl}} \propto m_{k\to l} m_{l\to k} \quad \textit{for separators,} \quad (17)$$

where $\psi_{c_k} = \exp(\theta_{c_k})$, $m_{k\to l} = \exp(\lambda_{k\to l})$ and $m_{\sim k} = \prod_{l \in \mathcal{N}(k)} m_{l\to k}$ is the product of messages sending from the set of neighboring clusters $\mathcal{N}(k)$ to c_k. The derivation of Eq. 16 used the results in Lemma B.2.

Finally, substituting the consistency constraints

$$\sum_{x_{c_k \setminus s_{kl}}} \tau_{c_k} = \tau_{s_{kl}}$$

into (15)-(17) leads the fixed point updates in (12)-(13).

B.2 Reparameterization Interpretation

We can give a reparameterization interpretation for the MEU-BP update in (12)-(13) similar to that of the sum-, max- and hybrid- BP algorithms [e.g., Wainwright et al., 2003a, Weiss et al., 2007, Liu and Ihler, 2011]. We start by defining a set of "MEU-beliefs" $\boldsymbol{b} = \{b(x_{c_k}), b(x_{s_{kl}})\}$ by $b(x_{c_k}) \propto \psi_{c_k} m_k$ for all $c_k \in \mathcal{C}$, and $b(x_{s_{kl}}) \propto m_{k\to l} m_{l\to k}$. Note that we distinguish between the "beliefs" $\boldsymbol{b}$ and the "marginals" $\boldsymbol{\tau}$ in (15)-(17). We have:

Lemma B.4. *(a). At each iteration of MEU-BP in (12)-(13), the MEU-beliefs $\boldsymbol{b}$ satisfy*

$$q(\boldsymbol{x}) \propto \frac{\prod_{k \in \mathcal{V}} b(x_{c_k})}{\prod_{(kl) \in \mathcal{E}} b(x_{s_{kl}})} \quad (18)$$

where $q(\boldsymbol{x})$ is the augmented distribution of the ID.

(b). At a fixed point of MEU-BP, we have

Sum-consistency: *(normal clusters)*
$$\sum_{c_k \setminus s_{ij}} b(x_{c_k}) = b(x_{s_{kl}}),$$

MEU-consistency: *(decision clusters)*
$$\sum_{c_k \setminus s_{ij}} \sigma_k[b(x_{c_k}); \epsilon] = b(x_{s_{kl}}).$$

Proof. (a). By simple algebraic substitution, one can show

$$\frac{\prod_{k \in \mathcal{V}} b(x_{c_k})}{\prod_{(kl) \in \mathcal{E}} b(x_{s_{kl}})} \propto \prod_{c_k \in \mathcal{C}} \psi_{c_k}(x_{c_k}).$$

Since $p(\boldsymbol{x}) \propto \prod_{c_k \in \mathcal{C}} \psi_{c_k}(x_{c_k})$, the result follows.

(b). Simply substitute the definition of $\boldsymbol{b}$ into the message passing scheme (12)-(13). □

B.3 Correctness Guarantees

Theorem 4.1. *Let $(\mathcal{G}, \mathcal{C}, \mathcal{S})$ be a consistent junction tree for a subset of decision nodes D', and $\boldsymbol{b}$ be a set of MEU-beliefs satisfying the reparameterization and the consistency conditions in Lemma B.4 with $\epsilon \to 0^+$. Let $\boldsymbol{\delta}^* = \{b_\epsilon(x_{d_k}|x_{\mathrm{pa}(d_k)}) : d_k \in D\}$, then $\boldsymbol{\delta}^*$ is a locally optimal strategy in that sense that $\mathrm{EU}(\{\boldsymbol{\delta}^*_{D'}, \boldsymbol{\delta}_{D\setminus D'}\}) \le \mathrm{EU}(\boldsymbol{\delta}^*)$ for any $\boldsymbol{\delta}_{D\setminus D'}$.*

Proof. On a junction tree, the reparameterization in (18) can be rewritten as

$$q(\boldsymbol{x}) = b_0 \prod_{k \in \mathcal{V}} \frac{b(x_{c_k})}{b(x_{s_k})},$$

where $s_k = s_{k,\pi(k)}$ ($s_k = \emptyset$ for the root node) and b_0 is the normalization constant.

For notational convenience, we only prove the case when $D' = D$, i.e., the junction tree is globally consistent . More general cases follow similarly, by noting

that any decision node imputed with a fixed decision rule can be simply treated as a chance node.

First, we can rewrite $\mathrm{EU}(\boldsymbol{\delta}^*)$ as

$$\begin{aligned}\mathrm{EU}(\boldsymbol{\delta}^*) &= \sum_x q(\boldsymbol{x}) \prod_{i\in D} b_\epsilon(x_i|x_{\mathrm{pa}(i)}) \\ &= b_0 \sum_x \prod_{k\in\mathcal{V}} \frac{b(x_{c_k})}{b(x_{s_k})} \prod_{i\in D} b_\epsilon(x_i|x_{\mathrm{pa}(i)}) \\ &= b_0 \sum_x \left\{ \prod_{k\in\mathcal{C}} \frac{b(x_{c_k})}{b(x_{s_k})} \right\} \cdot \left\{ \prod_{k\in\mathcal{D}} \frac{b(x_{c_k}) b_\epsilon(x_{d_k}|x_{\mathrm{pa}(d_k)})}{b(x_{s_k})} \right\} \\ &= b_0,\end{aligned}$$

where the last equality follows by the sum- and MEU-consistency condition (with $\epsilon \to 0^+$). To complete the proof, we just need to show that $\mathrm{EU}(\boldsymbol{\delta}) \le b_0$ for any $\boldsymbol{\delta} \in \Delta$. Again, note that $\mathrm{EU}(\boldsymbol{\delta})/b_0$ equals

$$\sum_x \left\{ \prod_{k\in\mathcal{C}} \frac{b(x_{c_k})}{b(x_{s_k})} \right\} \cdot \left\{ \prod_{k\in\mathcal{D}} \frac{b(x_{c_k}) p^\delta(x_{d_k}|x_{\mathrm{pa}(d_k)})}{b(x_{s_k})} \right\}.$$

Let $z_k = c_k \setminus s_k$; since $\mathcal{G}$ is a junction tree, the z_k form a partition of V, i.e., $\cup_k z_k = V$ and $z_k \cap z_l = 1$ for any $k \neq l$. We have

See insert (*)

where the equality (*) holds because $\{z_k\}$ forms a partition of V, and equality (19) holds due to the sum-consistency condition. The last inequality follows the proof in Lemma B.3. This completes the proof. □

Based to Theorem 4.1, we can easily establish person-by-person optimality of BP on an arbitrary junction tree.

Theorem 4.2. *Let $(\mathcal{G}, \mathcal{C}, \mathcal{S})$ be an arbitrary junction tree, and $\boldsymbol{b}$ and $\boldsymbol{\delta}^*$ defined in Theorem 4.1. Then $\boldsymbol{\delta}^*$ is a locally optimal strategy in Nash's sense: $\mathrm{EU}(\{\boldsymbol{\delta}_i^*, \boldsymbol{\delta}_{D\setminus i}\}) \le \mathrm{EU}(\boldsymbol{\delta}^*)$ for any $i \in D$ and $\boldsymbol{\delta}_{D\setminus i}$.*

Proof. Following Theorem 4.1, one need only show that any junction tree is consistent for any single decision node $i \in D$; this is easily done by choosing a tree-ordering rooted at i's decision cluster. □

C About the Proximal Algorithm

The proximal method can be equivalently interpreted as a marjorize-minimize (MM) algorithm [Hunter and Lange, 2004], or a convex concave procedure [Yuille, 2002]. The MM and CCCP algorithms have been widely applied to standard inference problems to obtain convergence guarantees or better solutions, see e.g., Yuille [2002], Liu and Ihler [2011].

The MM algorithm is an generalization of the EM algorithm, which solves $\min_{\boldsymbol{\tau}\in\mathbb{M}} f(\boldsymbol{\tau})$ by a sequence of surrogate optimization problems

$$\boldsymbol{\tau}^{t+1} = \arg\min_{\boldsymbol{\tau}\in\mathbb{M}} f^t(\boldsymbol{\tau}),$$

where $f^t(\boldsymbol{\tau})$, known as a *majorizing function*, should satisfy $f^t(\boldsymbol{\tau}) \ge f(\boldsymbol{\tau})$ for all $\boldsymbol{\tau} \in \mathbb{M}$ and $f^t(\boldsymbol{\tau}^t) = f(\boldsymbol{\tau}^t)$. It is straightforward to check that the objective in the proximal update (20) is a majorizing function. Therefore, the proximal algorithm can be treated as a special MM algorithm.

The convex concave procedure (CCCP) [Yuille and Rangarajan, 2003] is a special MM algorithm which decomposes the objective into a difference of two convex functions, that is,

$$f(\boldsymbol{\tau}) = f^+(\boldsymbol{\tau}) - f^-(\boldsymbol{\tau}),$$

where $f^+(\boldsymbol{\tau})$ and $f^-(\boldsymbol{\tau})$ are both convex, and constructs a majorizing function by linearizing the negative part, that is,

$$f^t(\boldsymbol{\tau}) = f^+(\boldsymbol{\tau}) - \nabla f^-(\boldsymbol{\tau}^t)^T(\boldsymbol{\tau} - \boldsymbol{\tau}^t).$$

One can easily show that $f^t(\boldsymbol{\tau})$ is a majorizing function via Jensen's inequality. To apply CCCP on the MEU dual (1), it is natural to set

$$f^+(\boldsymbol{\tau}) = -[\langle \boldsymbol{\theta}, \boldsymbol{\tau}\rangle + H(\boldsymbol{x}; \boldsymbol{\tau})]$$

and

$$f^-(\boldsymbol{\tau}) = -\sum_{i\in D} H(x_i|x_{\mathrm{pa}(i)}; \boldsymbol{\tau}).$$

Such a CCCP algorithm is recognizable as equivalent to the proximal algorithm in Section 4.2 with $w^t = 1$.

The convergence results for MM algorithms and CCCP are also well established; see Vaida [2005], Lange et al. [2000], Schifano et al. [2010] for the MM algorithm and Sriperumbudur and Lanckriet [2009], Yuille and Rangarajan [2003] for CCCP.

D Additively Decomposable Utilities

The algorithms we describe require the augmented distribution $q(\boldsymbol{x})$ to be factored, or have low (constrained) tree-width. However, this can easily not be the case for a direct representation of additively decomposable utilities. To explain, recall that the augmented distribution is $q(\boldsymbol{x}) \propto q^0(\boldsymbol{x})u(\boldsymbol{x})$, where $q^0(\boldsymbol{x}) = \prod_{i\in C} p(x_i|x_{\mathrm{pa}(i)})$, and $u(\boldsymbol{x}) = \sum_{j\in U} u_j(x_{\beta_j})$. In this case, the utility $u(\boldsymbol{x})$ creates a large factor with variable domain $\cup_j \beta_j$, and can easily destroy the factored structure of $q(\boldsymbol{x})$. Unfortunately, the naïve method of calculating the expectation node by node, or

$$\begin{aligned}
\mathrm{EU}(\boldsymbol{\delta})/b_0 &\le \sum_{\boldsymbol{x}} \Big\{ \prod_{k\in\mathcal{C}} \max_{x_{s_k}} \frac{b(x_{c_k})}{b(x_{s_k})} \Big\} \cdot \{ \prod_{k\in\mathcal{D}} \max_{x_{s_k}} \frac{b(x_{c_k}) p^{\delta}(x_{d_k}|x_{\mathrm{pa}(d_k)})}{b(x_{s_k})} \} && (*) \\
&= \Big\{ \prod_{k\in\mathcal{C}} \max_{x_{s_k}} \sum_{x_{z_k}} \frac{b(x_{c_k})}{b(x_{s_k})} \Big\} \cdot \Big\{ \prod_{k\in\mathcal{D}} \max_{x_{s_k}} \sum_{x_{z_k}} \frac{b(x_{c_k}) p^{\delta}(x_{d_k}|x_{\mathrm{pa}(d_k)})}{b(x_{s_k})} \Big\} && (19) \\
&= \prod_{k\in\mathcal{D}} \max_{x_{s_k}} \sum_{x_{z_k}} \frac{b(x_{c_k}) p^{\delta}(x_{d_k}|x_{\mathrm{pa}(d_k)})}{b(x_{s_k})} && (20) \\
&= \prod_{k\in\mathcal{D}} \max_{x_{s_k}} \sum_{x_{d_k}} \frac{b(x_{d_k}, x_{\mathrm{pa}(d_k)}) p^{\delta}(x_{d_k}|x_{\mathrm{pa}(d_k)})}{b(x_{s_k})} \\
&\le \prod_{k\in\mathcal{D}} \max_{x_{s_k}} \sum_{x_{d_k}} \frac{b(x_{d_k}, x_{\mathrm{pa}(d_k)}) b_{\epsilon}(x_{d_k}|x_{\mathrm{pa}(d_k)})}{b(x_{s_k})} \\
&= 1,
\end{aligned}$$

the commonly used general variable elimination procedures [e.g., Jensen et al., 1994] do not appear suitable for our variational framework.

Instead, we introduce an artificial product structure into the utility function by augmenting the model with a latent "selector" variable, similar to that used for the "complete likelihood" in mixture models. Let y_0 be an auxiliary random variable taking values in the utility index set U, so that

$$\tilde{q}(\boldsymbol{x}, y_0) = q^0(\boldsymbol{x}) \prod_j \tilde{u}_j(x_{\beta_j}, y_0),$$

where $\tilde{u}_j(x_{\beta_j}, y_0)$ is defined by $\tilde{u}_j(x_{\beta_j}, j) = \tilde{u}_j(x_{\beta_j})$ and $\tilde{u}_j(x_{\beta_j}, k) = 1$ for $j \neq k$. It is easy to verify that the marginal distribution of $\tilde{q}(\boldsymbol{x}, y_0)$ over y_0 is $q(\boldsymbol{x})$, that is, $\sum_{y_0} \tilde{q}(\boldsymbol{x}, y_0) = q(\boldsymbol{x})$. The treewidth of $\tilde{q}(\boldsymbol{x}, y_0)$ is no larger than one plus the treewidth of the graph (with utility nodes included) of the ID, which is typically much smaller than that of $q(\boldsymbol{x})$ when the complete utility $u(\boldsymbol{x})$ is included directly. A derivation similar to that in Theorem 3.1 shows that we can replace $\theta(\boldsymbol{x}) = \log q(\boldsymbol{x})$ in (1) with $\tilde{\theta}(\boldsymbol{x}, y_0) = \log \tilde{q}(\boldsymbol{x}, y_0)$, where y_0 is treated as a regular chance node, without changing the results. The complexity of this method may be further improved by exploiting the context-specific independence of $\tilde{q}(\boldsymbol{x}, y_0)$, i.e., that $\tilde{q}(\boldsymbol{x}|y_0)$ has a different dependency structure for different values of y_0, but we leave this for future work.

E Decentralized Sensor Network

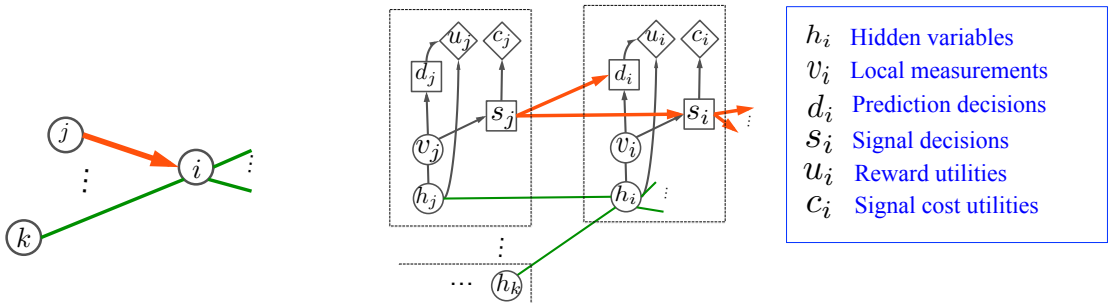
In this section, we provide detailed information about the influence diagram constructed for the decentralized sensor network detection problem in Section 5. Let $h_i, i = 1, \ldots, n_h$, be the hidden variables we want to detect using sensors. We assume the distribution $p(h)$ is an attractive pairwise MRF on a graph $G_h = (V_h, E_h)$,

$$p(h) = \frac{1}{Z} \exp[\sum_{(ij)\in G_h} \theta_{ij}(h_i, h_j))], \qquad (21)$$

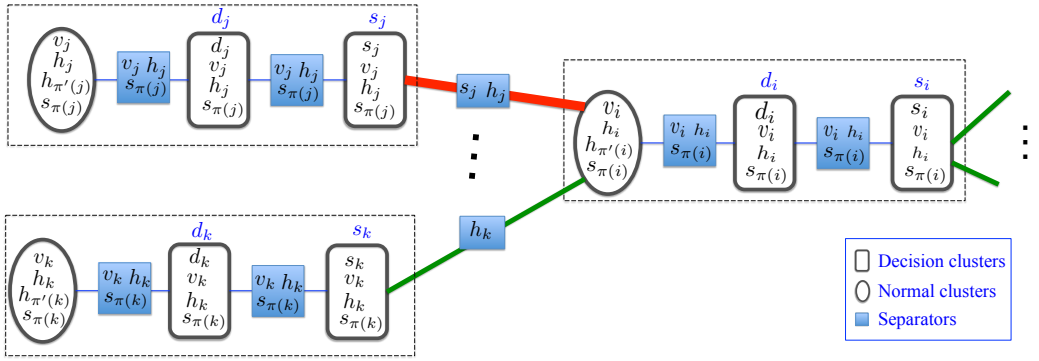
where h_i are discrete variables with p_h states (we take $p_h = 5$); we set $\theta_{ij}(k, k) = 0$ and randomly draw $\theta_{ij}(k, l)$ $(k \neq l)$ from the negative absolute values of a standard Gaussian variable $\mathcal{N}(0, 1)$. Each sensor gives a noisy measurement v_i of the local variable h_i with probability of error e_i, that is, $p(v_i|h_i) = 1 - e_i$ for $v_i = h_i$ and $p(v_i|h_i) = e_i/(p_h - 1)$ (uniformly) for $v_i \neq h_i$.

Let G_s be a DAG that defines the path on which the sensors are allowed to broadcast signals (all the downstream sensors receive the same signal); we assume the channels are noise free. Each sensor is associated with two decision variables: $s_i \in \{0, \pm 1\}$ represents the signal send from sensor i, where ± 1 represents a one-bit signal with cost λ and 0 represents "off" with no cost; and d_i represents the prediction of h_i based on v_i and the signals $s_{\mathrm{pa}_s(i)}$ received from i's upperstream sensors; a correct prediction $(d_i = h_i)$ yields a reward γ (we set $\gamma = \ln 2$). Hence, two types of utility functions are involved, the signal cost utilities u_λ, with $u_\lambda(s_i = \pm 1) = -\lambda$ and $u_\lambda(s_i = 0) = 0$; the prediction reward utilities u_γ with $u_\gamma(d_i, h_i) = \gamma$ if $d_i = h_i$ and $u_\gamma(d_i, h_i) = 0$ otherwise. The total utility function is constructed multiplicatively via $u = \exp[\sum_i u_\gamma(d_i, h_i) + u_\lambda(s_i)]$.

We also create two qualities of sensors: "good" sensors for which e_i are drawn from $\mathcal{U}([0, .1])$ and "bad" sensors $(e_i \sim \mathcal{U}([.7, .8]))$, where $\mathcal{U}$ is the uniform distribution. Generally speaking, the optimal strategies should pass signals from the good sensors to bad sen-



(a) The sensor network structure (b) The influence diagram for sensor network in (a)



(c) A loopy junction graph for the ID in (b)

Figure 1: (a) A node of the sensor network structure; the green lines denote the MRF edges, on some of which (red arrows) signals are allowed to path. (b) The influence diagram constructed for the sensor network in (a). (c) A junction graph for the ID in (b); $\pi(i)$ denotes the parent set of i in terms of the signal path G_s, and $\pi'(i)$ denotes the parent set in terms of the hidden process $p(h)$ (when $p(h)$ is transformed into a Bayesian network by triangularizing reversely along order o). The decision clusters (black rectangles) are labeled with their corresponding decision variables on their top.

sors to improve their predictive power. See Fig. 1 for the actual influence diagram.

The definition of the ID here is not a standard one, since $p(h)$ is not specified as a Bayesian network; but one could convert $p(h)$ to an equivalent Bayesian network by the standard triangulation procedure. The normalization constant Z in (21) only changes the expected utility function by a constant and so does not need to be calculated for the purpose of the MEU task.

Without loss of generality, for notation we assume the order $[1, \ldots, n_h]$ is consistent with the signal path G_h. Let $o = [h_1, v_1, d_1, s_1 \ ; \ \ldots \ ; \ h_{n_h}, v_{n_h}, d_{n_h}, s_{n_h}]$. The junction tree we used in the experiment is constructed by the standard triangulation procedure, backwards along the order o. A proper construction of a loopy junction graph is non-trivial; we show in Fig. 1(c) the one we used in Section 5. It is constructed such that the decision structure inside each sensor node is preserved to be exact, while at a higher level (among sensors), a standard loopy junction graph (similar to that introduced in Mateescu et al. [2010] that corresponds to Pearl's loopy BP) captures the correlation between the sensor nodes. One can shown that such a constructed junction graph reduces to a junction tree when the MRF G_h is a tree and the signal path G_s is an oriented tree.

F Additional Related Work

There exists a large body of work for solving influence diagrams, mostly on exact algorithms with prefect recall; see Koller and Friedman [2009] for a recent review. Our work is most closely connected to the early work of Jensen et al. [1994], who compile an ID to a junction tree structure on which a special message passing algorithm is performed; their notion of *strong junction trees* is related to our notion of *global consistency*. However, their framework requires the perfect recall assumption and it is unclear how to extend it to approximate inference. A somewhat different approach transforms the decision problem into a sequence of standard Bayesian network inference problems [Cooper, 1988, Shachter and Peot, 1992, Zhang, 1998], where each subroutine is a standard inference problem, and can be solved using standard algorithms, either exactly or approximately; again, their method only works within the perfect recall assumption. Other approximation algorithms for ID are also based on separately approximating individual components of exact algorithms, e.g., Sabbadin et al. [2011] and Sallans [2003] approximate the policy update methods by mean field methods; Nath and Domingos [2010] uses adaptive belief propagation to approximate the inner loop of greedy search algorithms. [Watthayu, 2008] proposed a loopy BP algorithm, but without theoretical justification. To the best of our knowledge, we know of no well-established "direct" approximation methods.

For ID without perfect recall (LIMID), backward-induction-like methods do not apply; most algorithms work by optimizing the decision rules node-by-node or group-by-group; see e.g., Lauritzen and Nilsson [2001], Madsen and Nilsson [2001], Koller and Milch [2003]; these methods reduce to the exact backward-reduction (hence guaranteeing global optimality) if applied on IDs with perfect recall and update backwards along the temporal ordering. However, they only guarantee local person-by-person optimality for general LIMIDs, which may be weaker than the optimality guaranteed by our BP-like methods. Other styles of approaches, such as Monte Carlo methods [e.g., Bielza et al., 1999, Cano et al., 2006, Charnes and Shenoy, 2004, Garcia-Sanchez and Druzdzel, 2004] and search-based methods [e.g., Luque et al., 2008, Qi and Poole, 1995, Yuan and Wu, 2010, Marinescu, 2010] have also been proposed. Recently, Maua and Campos [2011] proposed a method for finding the globally optimal strategies of LIMIDs by iteratively pruning non-maximal policies. However, these methods usually appear to have much greater computational complexity than SPU or our BP-like methods.

Finally, some variational inference ideas have been applied to the related problems of reinforcement learning or solving Markov decision processes [e,g., Sallans and Hinton, 2001, Furmston and Barber, 2010, Yoshimoto and Ishii, 2004].

References

C. Bielza, P. Muller, and D. R. Insua. Decision analysis by augmented probability simulation. *Management Science*, 45(7):995–1007, 1999.

A. Cano, M. Gómez, and S. Moral. A forward-backward Monte Carlo method for solving influence diagrams. *Int. J. Approx. Reason.*, 42(1-2):119–135, 2006.

J. Charnes and P. Shenoy. Multistage Monte Carlo method for solving influence diagrams using local computation. *Management Science*, pages 405–418, 2004.

G. F. Cooper. A method for using belief networks as influence diagrams. In *UAI*, pages 55–63, 1988.

A. Detwarasiti and R. D. Shachter. Influence diagrams for team decision analysis. *Decision Anal.*, 2, Dec 2005.

T. Furmston and D. Barber. Variational methods for reinforcement learning. In *AISTATS*, volume 9, pages 241–248, 2010.

D. Garcia-Sanchez and M. Druzdzel. An efficient sampling algorithm for influence diagrams. In *Proceedings of the Second European Workshop on Probabilistic Graphical Models (PGM–04), Leiden, Netherlands*, pages 97–104, 2004.

R. Howard and J. Matheson. Influence diagrams. In *Readings on Principles & Appl. Decision Analysis*, 1985.

R. Howard and J. Matheson. Influence diagrams. *Decision*

Anal., 2(3):127–143, 2005.

D. R. Hunter and K. Lange. A tutorial on MM algorithms. *The American Statistician*, 1(58), February 2004.

A. Iusem and M. Teboulle. On the convergence rate of entropic proximal optimization methods. *Computational and Applied Mathematics*, 12:153–168, 1993.

F. Jensen, F. V. Jensen, and S. L. Dittmer. From influence diagrams to junction trees. In *UAI*, pages 367–373. Morgan Kaufmann, 1994.

J. Jiang, P. Rai, and H. Daumé III. Message-passing for approximate MAP inference with latent variables. In *NIPS*, 2011.

D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

D. Koller and B. Milch. Multi-agent influence diagrams for representing and solving games. *Games and Economic Behavior*, 45(1):181–221, 2003.

O. P. Kreidl and A. S. Willsky. An efficient message-passing algorithm for optimizing decentralized detection networks. In *IEEE Conf. Decision Control*, Dec 2006.

K. Lange, D. Hunter, and I. Yang. Optimization transfer using surrogate objective functions. *Journal of Computational and Graphical Statistics*, pages 1–20, 2000.

S. Lauritzen and D. Nilsson. Representing and solving decision problems with limited information. *Managment Sci.*, pages 1235–1251, 2001.

Q. Liu and A. Ihler. Variational algorithms for marginal MAP. In *UAI*, Barcelona, Spain, July 2011.

M. Luque, T. Nielsen, and F. Jensen. An anytime algorithm for evaluating unconstrained influence diagrams. In *Proc. PGM*, pages 177–184, 2008.

A. Madsen and D. Nilsson. Solving influence diagrams using HUGIN, Shafer-Shenoy and lazy propagation. In *UAI*, volume 17, pages 337–345, 2001.

R. Marinescu. A new approach to influence diagrams evaluation. In *Research and Development in Intelligent Systems XXVI*, pages 107–120. Springer London, 2010.

B. Martinet. Régularisation d'inéquations variationnelles par approximations successives. *Revue Française dInformatique et de Recherche Opérationelle*, 4:154–158, 1970.

R. Mateescu, K. Kask, V. Gogate, and R. Dechter. Join-graph propagation algorithms. *Journal of Artificial Intelligence Research*, 37:279–328, 2010.

D. D. Maua and C. P. Campos. solving decision problems with limited information. In *NIPS*, 2011.

A. Nath and P. Domingos. Efficient belief propagation for utility maximization and repeated inference. In *AAAI Conference on Artificial Intelligence*, 2010.

J. Pearl. Influence diagrams - historical and personal perspectives. *Decision Anal.*, 2(4):232–234, dec 2005.

R. Qi and D. Poole. A new method for influence diagram evaluation. *Computational Intelligence*, 11(3):498–528, 1995.

P. Ravikumar, A. Agarwal, and M. J. Wainwright. Message-passing for graph-structured linear programs: Proximal projections, convergence, and rounding schemes. *Journal of Machine Learning Research*, 11: 1043–1080, Mar 2010.

R. T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM Journal on Control and Optimization*, 14(5):877, 1976.

K. Rose. Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. *Proc. IEEE*, 86(11):2210 –2239, Nov 1998.

R. Sabbadin, N. Peyrard, and N. Forsell. A framework and a mean-field algorithm for the local control of spatial processes. *International Journal of Approximate Reasoning*, 2011.

B. Sallans. Variational action selection for influence diagrams. Technical Report OEFAI-TR-2003-29, Austrian Research Institute for Artificial Intelligence, 2003.

B. Sallans and G. Hinton. Using free energies to represent q-values in a multiagent reinforcement learning task. In *NIPS*, pages 1075–1081, 2001.

E. Schifano, R. Strawderman, and M. Wells. Majorization-minimization algorithms for nonsmoothly penalized objective functions. *Electronic Journal of Statistics*, 4: 1258–1299, 2010.

R. Shachter. Model building with belief networks and influence diagrams. *Advances in decision analysis: from foundations to applications*, page 177, 2007.

R. Shachter and M. Peot. Decision making using probabilistic inference methods. In *UAI*, pages 276–283, 1992.

B. Sriperumbudur and G. Lanckriet. On the convergence of the concave-convex procedure. In *NIPS*, pages 1759–1767, 2009.

M. Teboulle. Entropic proximal mappings with applications to nonlinear programming. *Mathematics of Operations Research*, 17(3):pp. 670–690, 1992.

P. Tseng and D. Bertsekas. On the convergence of the exponential multiplier method for convex programming. *Mathematical Programming*, 60(1):1–19, 1993.

F. Vaida. Parameter convergence for EM and MM algorithms. *Statistica Sinica*, 15(3):831–840, 2005.

R. Viswanathan and P. Varshney. Distributed detection with multiple sensors: part I – fundamentals. *Proc. IEEE*, 85(1):54 –63, Jan 1997.

M. Wainwright and M. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1(1-2):1–305, 2008.

M. Wainwright, T. Jaakkola, and A. Willsky. A new class of upper bounds on the log partition function. *IEEE Trans. Info. Theory*, 51(7):2313–2335, July 2005.

M. J. Wainwright, T. Jaakkola, and A. S. Willsky. Tree-based reparameterization framework for analysis of sum-product and related algorithms. *IEEE Trans. Info. Theory*, 45:1120–1146, 2003a.

M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. MAP estimation via agreement on (hyper) trees: Message-passing and linear programming approaches. *IEEE Trans. Info. Theory*, 51(11):3697 – 3717, Nov 2003b.

W. Watthayu. Representing and solving influence diagram in multi-criteria decision making: A loopy belief propagation method. In *International Symposium on Computer Science and its Applications (CSA '08)*, pages 118 –125, Oct. 2008.

Y. Weiss and W. Freeman. On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs. *IEEE Trans. Info. Theory*, 47(2):736 –744, Feb 2001.

Y. Weiss, C. Yanover, and T. Meltzer. MAP estimation, linear programming and belief propagation with convex free energies. In *UAI*, 2007.

D. Wolpert. Information theory – the bridge connecting bounded rational game theory and statistical physics. *Complex Engineered Systems*, pages 262–290, 2006.

J. Yedidia, W. Freeman, and Y. Weiss. Constructing free-energy approximations and generalized BP algorithms. *IEEE Trans. Info. Theory*, 51, July 2005.

J. Yoshimoto and S. Ishii. A solving method for mdps by minimizing variational free energy. In *International Joint Conference on Neural Networks*, volume 3, pages

1817–1822. IEEE, 2004.

C. Yuan and X. Wu. Solving influence diagrams using heuristic search. In *Proc. Symp. AI& Math*, 2010.

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

A. L. Yuille and A. Rangarajan. The concave-convex procedure. *Neural Computation*, 15:915–936, April 2003.

N. L. Zhang. Probabilistic inference in influence diagrams. In *Computational Intelligence*, pages 514–522, 1998.

N. L. Zhang, R. Qi, and D. Poole. A computational theory of decision networks. *Int. J. Approx. Reason.*, 11:83–158, 1994.