Proteins, Particles, and Pseudo-Max-Marginals: A Submodular Approach

Jason L. Pacheco PACHECOJ@CS.BROWN.EDU
Erik B. Sudderth SUDDERTH@CS.BROWN.EDU
Department of Computer Science, Brown University, Providence, RI 02912, USA

Abstract

Variants of max-product (MP) belief propagation effectively find modes of many complex graphical models, but are limited to discrete distributions. Diverse particle max-product (D-PMP) robustly approximates max-product updates in continuous MRFs using stochastically sampled particles, but previous work was specialized to tree-structured models. Motivated by the challenging problem of protein side chain prediction, we extend D-PMP in several key ways to create a generic MAP inference algorithm for loopy models. We define a modified diverse particle selection objective that is provably submodular, leading to an efficient greedy algorithm with rigorous optimality guarantees, and corresponding max-marginal error bounds. We further incorporate tree-reweighted variants of the MP algorithm to allow provable verification of global MAP recovery in many models. Our general-purpose MATLAB library is applicable to a wide range of pairwise graphical models, and we validate our approach using optical flow benchmarks. We further demonstrate superior side chain prediction accuracy compared to baseline algorithms from the state-of-the-art Rosetta package.

1. Introduction

Continous random variables are often used to model complex interactions among objects in the world around us, leading to challenging multi-modal posterior distributions. The *maximum a posteriori* (MAP) inference objective for such models is typically non-convex, and optimization algorithms become trapped in local optima. Approaches that discretize the latent space and apply *max-product* (MP) belief propagation (Pearl, 1988; Wainwright et al., 2005; Wainwright & Jordan, 2008) can be effective in few dimensions, but for high-dimensional models only coarse discretizations are feasible. Continuous optimization can be performed via Monte Carlo sampling and simulated annealing (Geman & Geman, 1984; Andrieu et al., 2003), but these methods often require long computation times.

A number of stochastic local search methods have been developed (Trinh & McAllester, 2009; Peng et al., 2011; Besse et al., 2012; Kothapa et al., 2011) that combine the flexibility of sampling-based approaches with the efficiency of MP message passing. This family of *particle max-product* (PMP) methods share a general framework: at each iteration new hypotheses are sampled from stochastic proposals, evaluated via discrete max-product message updates, and accepted or rejected based on some criterion. PMP algorithms differ primarily in their choice of stochastic proposals and particle selection criteria.

The *diverse particle max-product* (D-PMP) (Pacheco et al., 2014) algorithm maintains hypotheses near multiple local optima via an optimization-based selection step that minimizes distortions in MP message values. D-PMP has excellent empirical performance on a human pose estimation task, but there is little theoretical justification for its particle selection integer program (IP), and the proposed greedy algorithm has no optimality guarantees. Previous D-PMP formulations also assumed a tree-structured *Markov random field* (MRF) where MP provides exact max-marginals, and several key assumptions would be violated by a naive generalization to loopy graphical models.

In this paper, we generalize D-PMP to arbitrary pairwise MRFs with cycles by adapting *tree-reweighted max-product* (RMP) belief propagation (Wainwright et al., 2005). We define an alternative message distortion metric which leads to a *submodular* particle selection IP. An efficient greedy algorithm is guaranteed to produce message errors within a fraction of $(1 - \frac{1}{e})$ of the best achievable, and thus provide provably accurate max-marginal estimates. Our MATLAB library implements the D-PMP algorithm for general pairwise MRFs. For the tasks of optical flow estimation and protein side chain prediction, we demonstrate substantial improvements over previous PMP algorithms, and performance levels that match or exceed state-of-the-art domain-specific inference algorithms.

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

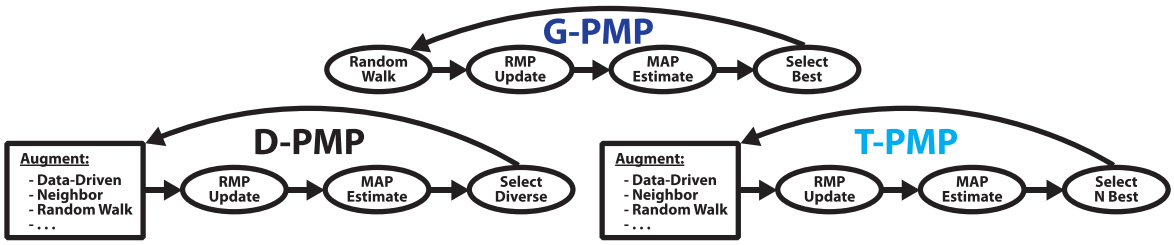


Figure 1. **Loopy PMP flowcharts.** The high-level structure of three variants of loopy particle max-product: the Greedy PMP (G-PMP) of Peng et al. (2011), the Top-N PMP (T-PMP) of Besse et al. (2012), and the Diverse PMP (D-PMP) of Pacheco et al. (2014).

2. Background

We begin with a brief introduction to max-product inference for discrete MRFs, which forms the basis for our particle-based approximations. To ground these concepts we introduce the protein side chain prediction task, which will be used to motivate and validate our approach.

2.1. Max-Product Belief Propagation

Max-product belief propagation (Pearl, 1988; Aji & McEliece, 2000; Wainwright & Jordan, 2008) performs MAP inference by passing messages along the edges of a graphical model. Consider a pairwise MRF, with edges $(s,t) \in \mathcal{E}$ and nodes $s \in \mathcal{V}$:

$$p(x) \propto \prod_{s \in \mathcal{V}} \psi_s(x_s) \prod_{(s,t) \in \mathcal{E}} \psi_{st}(x_s, x_t). \tag{1}$$

In MRFs with cycles, tree-reweighted max-product (RMP) (Wainwright et al., 2005) approximates MAP inference via a set of spanning trees, with edge appearance probabilities ρ_{st}. The RMP message from node t to s is:

$$\widetilde{m}_{ts}(x_s) = \max_{x_t} \psi_t(x_t)\psi_{st}(x_s,x_t)^{\frac{1}{\rho_{st}}} \frac{\prod_{u \in \Gamma(t) \setminus s} \widetilde{m}_{ut}(x_t)^{\rho_{st}}}{\widetilde{m}_{st}(x_t)^{1-\rho_{st}}}, \tag{2}$$

where $\Gamma(t)$ is the set of nodes neighboring t. Fixed points yield *pseudo-max-marginals*, which do not necessarily correspond to valid max-marginal distributions:

$$\widetilde{\nu}_s(x_s) \propto \psi_s(x_s) \prod_{u \in \Gamma(s)} \widetilde{m}_{us}(x_s)^{\rho_{us}} \approx \max_{\{x' \mid x'_s = x_s\}} p(x').$$

Via connections to linear programming relaxations, RMP provides a bound on the MAP probability at each iteration, and a certificate of optimality using Lagrange multipliers.

2.2. Particle Max-Product Belief Propagation

For continuous variables $x \in \mathcal{X}$, the message functions of Eq. (2) cannot be computed in general. Particle max-product (PMP) methods approximate messages by optimizing over a discrete set of *particles* found via stochastic search. Given a current set of N particles $\mathbb{X}_t \subset \mathcal{X}_t$, each PMP iteration has three stages, summarized in Figure 1.

Stochastic Proposals To allow higher-likelihood state configurations to be discovered, at each iteration PMP first creates an augmented particle set $\mathbb{X}^{\text{aug}} = \mathbb{X} \cup \mathbb{X}^{\text{prop}}$ of size αN, $\alpha > 1$. New particles are drawn from proposal distributions $\mathbb{X}^{\text{prop}} \sim q(\mathbb{X})$. In the simplest case, Gaussian random walk proposals $q^{\text{gauss}}(x_s) = N(x_s \mid \bar{x}_s, \Sigma)$ sample perturbations of current particle locations $\bar{x}_s$ (Trinh & McAllester, 2009; Peng et al., 2011). For some models, a more informative *neighbor-based* proposal is possible that samples from edge potentials $q^{\text{nbr}}(x_s \mid \bar{x}_t) \propto \psi_{st}(x_s, \bar{x}_t)$ conditioned on a particle $\bar{x}_t$ at neighboring node $t \in \Gamma(s)$ (Besse et al., 2012). Specialized "bottom-up" proposals based on approximations of observation potentials $\psi_s(x_s)$ can also be effective (Pacheco et al., 2014).

Max-Product Optimization Standard or reweighted MP message updates are used to approximate the max-marginal distribution of each proposed particle. The αN values of each discrete message vector satisfy $m_{ts}(x_s) =$

$$\max_{x_t \in \mathbb{X}_t} \psi_t(x_t)\psi_{st}(x_s,x_t)^{\frac{1}{\rho_{st}}} \frac{\prod_{u \in \Gamma(t) \setminus s} m_{ut}(x_t)^{\rho_{st}}}{m_{st}(x_t)^{1-\rho_{st}}}.$$

Message updates require $\mathcal{O}(\alpha^2 N^2)$ operations, and compute the pseudo-max-marginal $\nu_s(x_s)$ for each $x_s \in \mathbb{X}^{\text{aug}}$.

Particle Selection Particles are accepted or rejected to yield N new states $\mathbb{X}^{\text{new}} \subset \mathbb{X}^{\text{aug}}$. Particle selection makes subsequent iterations more computationally efficient.

The simple *greedy PMP* (G-PMP) method selects the single particle $x^*_s = \arg\max_{x_s \in \mathbb{X}^{\text{aug}}_s} \nu_s(x_s)$ with the highest max-marginal, and samples all other particles as Gaussian perturbations of this state (Trinh & McAllester, 2009; Peng et al., 2011). G-PMP updates are efficient, but they cannot preserve multiple modes, and random walk proposals do not effectively explore high-dimensional spaces.

A less greedy selection method retains the N particles with highest estimated max-marginal probability. This *top-N PMP* (T-PMP) (Pacheco et al., 2014) generalizes PatchMatch BP (Besse et al., 2012), a method specialized to low-level vision tasks which utilizes top-N particle selection and neighbor proposals. T-PMP finds high probability solutions quickly, but the top-N particles are often slight perturbations of the same solution, reducing the number of effective particles and causing sensitivity to initialization.

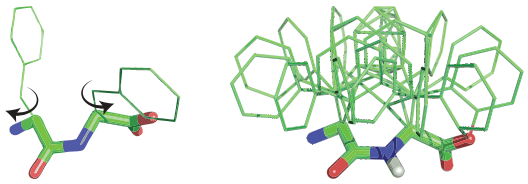


Figure 2. **Protein side chain.** *Left:* A simple protein with two amino acids forms a backbone (thick) and side chains (thin). *Right:* A regular discretization of the first dihedral angle.

To avoid the particle degeneracy common to G-PMP and T-PMP, the *diverse PMP* (D-PMP) (Pacheco et al., 2014) method selects particles via combinatorial optimization. An IP favors particles that minimally distort the current MP messages, and thus implicitly encourages diversity. By preserving solutions near multiple local optima, D-PMP reasons more globally and is less sensitive to initialization.

2.3. Protein Side Chain Prediction

Predicting protein structure, which is governed by pairwise energetic interactions, is a natural target for PMP algorithms (Peng et al., 2011; Soltan Ghoraie et al., 2013). Proteins are formed by chains of amino acids which consist of a *backbone* and a *side chain* unique to each amino acid type. Each protein assumes a 3D structure, or *conformation*, relating to its function. Given an amino acid sequence and a fixed backbone geometry, side chain prediction methods estimate the atomic coordinates of all side chain atoms.

We model side chain prediction as MAP inference in a pairwise MRF (Yanover & Weiss, 2002). The latent space is specified in terms of *dihedral* angles $x \in \mathbb{R}^d$, which describe the relative orientation between two planes (Fig. 2). The number of dihedral angles d varies by amino acid type. Energy is more easily modeled in terms of inter-atomic distance $r_{ij}(x)$ between pairs of atoms i, j. Pairwise terms encode interactions between nearby side chains via the *attractive* and *repulsive* components of the "6-12 Lennard-Jones" log-potential:

$$\log \psi_{st}(x_s, x_t) = \sum_{i=1}^{N_s} \sum_{j=1}^{N_t} 4\epsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^6 - \left(\frac{\sigma}{r_{ij}} \right)^{12} \right]. \quad (3)$$

Here N_s is the number of atoms in the s^{th} amino acid, ϵ controls the strength of attraction and σ the cutoff distance where atoms do not interact. Log-likelihoods $\log \psi_s(x_s)$ are given by a Gaussian mixture fit to the marginal statistics of observed dihedral angles. More details are in Sec. 4.2.

3. Loopy Diverse Particle Max-Product

The D-PMP message updates of Pacheco et al. (2014) can be directly applied to loopy MRFs, since each step decomposes into local computations on the graph. But, a naive extension may have convergence problems like those observed for loopy MP in many discrete models. Using RMP message passing, combined with our method for resolving ties, we can verify that global optimality is achieved and ensure that the MAP estimate is nondecreasing. We also introduce a new IP objective in the particle selection step which is a monotonic submodular maximization. This IP allows us to use a standard greedy algorithm for particle selection, and attain a multiplicative optimality bound.

3.1. Submodular Particle Selection

For each node $t \in \mathcal{V}$ we select particles to minimize the distortion between two message vectors. Specifically, we choose a subset of particles which minimizes the L_1 norm,

$$\begin{aligned} &\underset{z}{\text{minimize}} \quad \sum_{s \in \Gamma(t)} \|m_{ts} - \hat{m}_{ts}(z)\|_1 \quad (4) \\ &\text{subject to} \quad \|z\|_1 \leq N, \quad z \in \{0,1\}^{\alpha N}. \end{aligned}$$

The message vector m_{ts} is computed over the augmented particles $\mathbb{X}_t^{\text{aug}} = \{x_t^{(1)}, \ldots, x_t^{(\alpha N)}\}$, with $\alpha > 1$. The message vector $\hat{m}_{ts}(z)$ is computed over any subset of at most N particles $\mathbb{X}_t^{\text{new}} \subset \mathbb{X}_t^{\text{aug}}$ indexed by the indicator vector z,

$$m_{ts}(a) = \max_{b \in \{1,\ldots,\alpha N\}} M_{st}(a,b), \quad (5)$$

$$\hat{m}_{ts}(a; z) = \max_{b \in \{1,\ldots,\alpha N\}} z(b) M_{st}(a,b). \quad (6)$$

Here we have accumulated the terms needed for RMP message updates in a *message foundation* matrix $M_{st}(a,b) =$

$$\psi_t(x_t^{(b)}) \psi_{st}(x_s^{(a)}, x_t^{(b)})^{\frac{1}{\rho_{st}}} \frac{\prod_{u \in \Gamma(t) \setminus s} m_{ut}(b)^{\rho_{st}}}{m_{st}(b)^{1-\rho_{st}}}. \quad (7)$$

Pseudo-Max-Marginal Bounds Particles are chosen to minimize message distortions, but our primary goal is to maintain approximations of the pseudo-max-marginals:

$$\nu_s(x_s) \propto \psi_s(x_s) \prod_{u \in \Gamma(s)} m_{us}(x_s)^{\rho_{us}}, \quad (8)$$

and analogously for pseudo-max-marginals on the selected particles $\hat{\nu}$. If the potentials are bounded above and normalized so that $0 \preceq \psi \preceq 1$, then the sum of message distortions bounds the psuedo-max-marginal error.

Proposition 1. Let $0 \preceq \hat{m} \preceq m \preceq 1$ and edge appearance probabilities $\rho_{st} \in [0,1]$. For all nodes $s \in \mathcal{V}$ we have:

$$\|\nu_s - \hat{\nu}_s\|_1 \leq \sum_{t \in \Gamma(s)} \|m_{ts} - \hat{m}_{ts}\|_1^{\rho_{ts}} \quad (9)$$

We provide a proof in the Appendix. Note that we do not bound the difference between the D-PMP max-marginals $\nu_s(x_s)$ and the continuous max-marginals $\widetilde{\nu}_s(x_s)$; such results typically require strong and unrealistic assumptions (Peng et al., 2011). Instead, Eq. (9) shows that if we succeed in producing small message errors, the particle selection step will not significantly distort the pseudo-max-marginals, nor will it discard important hypotheses.

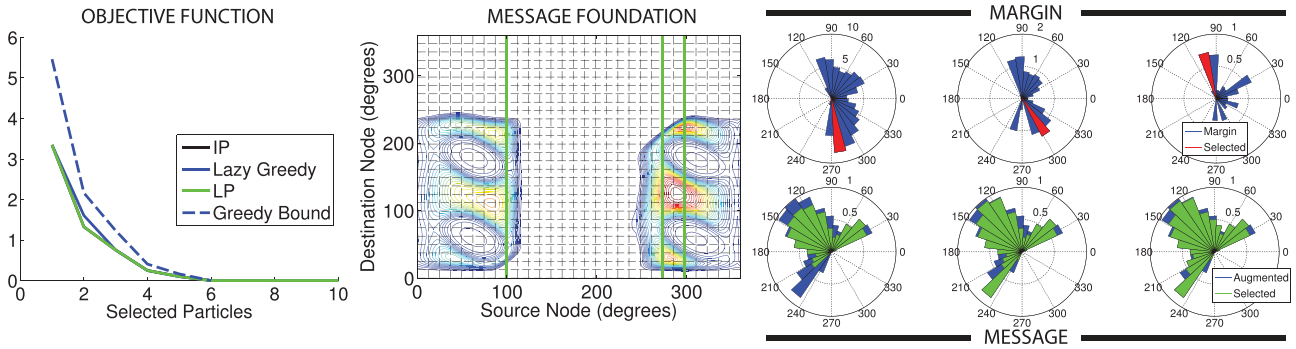


Figure 3. **LAZYGREEDY particle selection for side chain prediction** for the simple two-amino-acid protein of Fig. 2. We fix all but the first dihedral angle and select 3 particles out of a regular grid of 30 angles. *Left:* Objective function for optimal IP, an LP relaxation, and LAZYGREEDY . *Center:* Message foundation matrix showing locations of all particles (black) and the three selected particles of the source node (green). *Right:* Augmented message (lower, blue), message approximations for the first three particle selections (lower, green), and corresponding margins (upper); the selected particle at each step is the maximizer of the margin (red).

Submodularity The particle selection IP we propose minimizes the absolute sum of message differences (the L_1 norm). In contrast, the selection objective proposed by Pacheco et al. (2014) minimizes the maximum message difference (the L_∞ norm). With this modified error metric, minimizing Eq. (4) is equivalent to maximizing a monotonic submodular function.

Definition 1 (Submodularity). A set function $f : 2^Z \to \mathbb{R}$ defined over subsets of Z is submodular iff for any subsets $Y \subseteq X \subseteq Z$ and an element $e \notin X$ the function f satisfies,

$$f(Y \cup \{e\}) - f(Y) \geq f(X \cup \{e\}) - f(X).$$

The quantity $\Delta(Y, e) \triangleq f(Y \bigcup \{e\}) - f(Y)$ is the *margin*, and Def. (1) states that for any $e \notin X$ the margin is non-increasing in $|Y|$. This property of *diminishing marginal returns* allows us to use efficient greedy algorithms.

Proposition 2. The optimization of Equation (4) is equivalent to maximizing a monotonic submodular objective subject to cardinality constraints.

Proof. We focus on a single node and drop subscripts. Dropping constants, we can minimize Eq. (4) as follows:

$$\underset{z:\|z\|_1 \leq N}{\arg\max} \sum_a F_a(z) = \sum_a \left[\max_{1 \leq b \leq N} z(b) M(a, b) \right] \quad (10)$$

Let $y, z \in \{0, 1\}^{\alpha N}$ be particle selections and $y \subseteq z$ such that $y(b) = 1 \Rightarrow z(b) = 1$. For some candidate particle $\bar{b}$:

$$\overline{y}(b) = \begin{cases} 1, & \text{if } b = \bar{b} \\ y(b), & \text{o.w.} \end{cases} \qquad \overline{z}(b) = \begin{cases} 1, & \text{if } b = \bar{b} \\ z(b), & \text{o.w.} \end{cases}$$

The margins are given by direct calculation:

$$F_a(\overline{y}) - F_a(y) = \max(0, M(a, \bar{b}) - \hat{m}(a; y))$$
$$F_a(\overline{z}) - F_a(z) = \max(0, M(a, \bar{b}) - \hat{m}(a; z)).$$

Since $y \subseteq z$ we have that F_a is submodular,

$$F_a(\overline{y}) - F_a(y) \geq F_a(\overline{z}) - F_a(z).$$

A sum of submodular functions is submodular, and monotonicity holds since $\hat{m}(y) \leq \hat{m}(z)$. □

3.2. LAZYGREEDY Particle Selection

The LAZYGREEDY algorithm exploits diminishing marginal returns to avoid redundant computations (Minoux, 1978; Leskovec et al., 2007). Each iteration updates and sorts the largest margin until a stable maximizer is found. The algorithm terminates when the particle budget is exhausted, or the maximum margin is zero. Surprisingly, this greedy approach yields solutions within a factor $(1 - \frac{1}{e}) \approx 0.63$ of optimal (Nemhauser et al., 1978).

Initialize: For each node t let $M = \left[M_{s_1 t}^T, \ldots, M_{s_d t}^T\right]^T$ be the message foundations of neighbors $\Gamma(t) = \{s_1, \ldots, s_d\}$. Initialize the selection vector z and margins:

$$\Delta(b) = \sum_{a=1}^{d\alpha N} M(a, b), \quad z(b) = 0 \; \forall \; b \in \{1, \alpha N\}. \quad (11)$$

First Iteration: Ensure that the current MAP estimate x^* is never discarded by setting $z(b^*) = 1$, where b^* is the index of x_t^* in the augmented particle set $\mathbb{X}_t^{\text{aug}}$ (see Sec. 3.3). Update the message approximation $\hat{m}(a) = M(a, b^*)$.

Iterations 2 *to* N*:* Choose the largest margin to update,

$$\widetilde{b} = \underset{\{b | z(b) = 0\}}{\arg\max} \Delta(b).$$

If $\Delta(\widetilde{b}) = 0$ then terminate prematurely, the message can be perfectly reconstructed with a subset of particles. If $\Delta(\widetilde{b})$ has already been updated on the current iteration then set $z(\widetilde{b}) = 1$ and update the message approximation,

$$\hat{m}(a) = \max(\hat{m}(a), M_t(a, \widetilde{b})).$$

Otherwise, update the margin and repeat,

$$\Delta(\widetilde{b}) \triangleq \sum_a \left[\max(\hat{m}(a), M(a, \widetilde{b})) - \hat{m}(a) \right].$$

Selections are performed in parallel and updates at one node do not affect the selection at neighboring nodes. Figure 3 graphically demonstrates LAZYGREEDY selection on the small toy protein of Fig. 2.

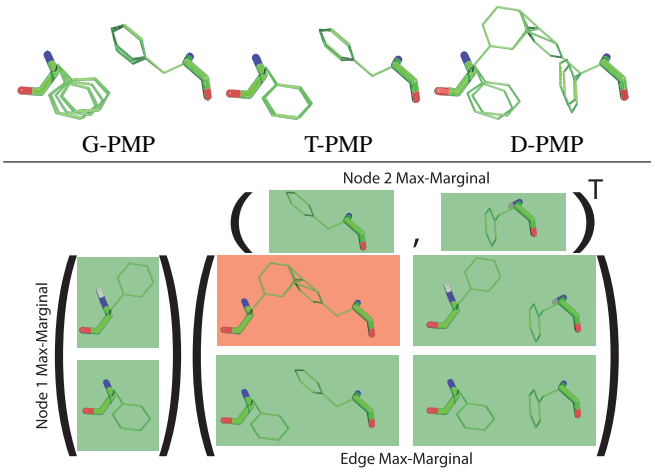


Figure 4. **Label Conflicts.** *Above:* Selected side chain particles of two amino acids (PDB: 1QOW). Diversity in the D-PMP particle set presents more opportunity for an inconsistent labeling. *Below:* Naively maximizing the node max-marginal over two tied states can produce a very unlikely joint configuration.

3.3. Resolving ties

In PMP we resolve ties using an approach similar to one proposed for discrete MRFs (Weiss et al., 2007). For discrete models the RMP pseudo-max-marginals ν admit a provably MAP solution x^* if a consistent labeling exists in the set of maxima (Wainwright et al., 2005):

$$x_s^* \in \arg\max_{x_s} \nu_s(x_s), \quad (x_s^*, x_t^*) \in \arg\max_{x_s, x_t} \nu_{st}(x_s, x_t).$$

For continuous distributions exact ties rarely exist, but small numerical errors in the estimated pseudo-max-marginals can perturb the particle that is inferred to be most likely, and lead to joint states with low probability due to "conflicted" edges. This problem is common in the side chain model, and as illustrated in Fig. 4, the diversity in the D-PMP particles makes conflicts more likely. To address this we relax the set of optima to be states with pseudo-max-marginal values within tolerance ϵ of the maximum:

$$\mathrm{OPT}(\nu_s) \triangleq \{x_s^* : |\nu_s(x_s^*) - \arg\max_{x_s} \nu_s(x_s)| \leq \epsilon\}.$$

Let $\mathcal{V}_T$ be the set of tied nodes with more than one near-maximal state, and $\mathcal{E}_T \triangleq \mathcal{E} \cap (\mathcal{V}_T \otimes \mathcal{V}_T)$ the edges joining them. Let x_{NT}^* be the unique assignments for non-tied nodes. Construct an MRF over the remaining tied nodes as

$$p_T(x_T) \propto \prod_{s \in \mathcal{V}_T} \widetilde{\psi}_s(x_s) \prod_{(s,t) \in \mathcal{E}_T} \psi_{st}(x_s, x_t), \qquad (12)$$

with the conditioned node potentials

$$\widetilde{\psi}_s(x_s) = \psi_s(x_s) \prod_{t \in \Gamma(s) \setminus \mathcal{V}_T} \psi_{st}(x_s, x_t^*). \qquad (13)$$

We label the remaining nodes $x_T^* = \arg\max_{x_T} p_T(x_T)$ using the junction tree algorithm. If the junction tree contains a unique maximizer, then $x^* = (x_T^*, x_{NT}^*)$ is the global MAP over the particles $\mathbb{X}$. This guarantee follows from the reparameterization property of pseudo-max-marginals and Theorem 2 of Weiss et al. (2007). Clique size is reduced by eliminating non-tied nodes, and by constraining labels to the set of tied states $x_T \in \mathrm{OPT}(\nu_s)$.

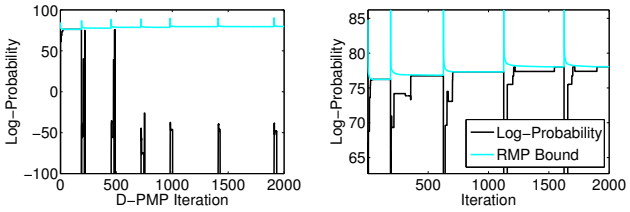


Figure 5. **Primal & Dual Trajectories** for a single protein (PDB: 1QOW) over all RMP iterations and 10 D-PMP steps; peaks indicate resampling. *Left:* Without resolving ties a MAP labeling is not obtained. *Right:* With tie resolution the duality gap vanishes.

4. Experimental Results

We consider two tasks that demonstrate the effectiveness and flexibility of D-PMP inference. We begin with optical flow estimation, a low-level vision task which recovers 2D pixel motion in an image sequence. Optical flow is a well-studied problem where specialized inference methods are thought to be near-optimal for the model and dataset we consider, and so provide a good comparison. We then revisit our running example of protein side chain prediction, which is more challenging due to increased dimensionality and complex potentials. Many methods for side chain prediction make coarse discrete approximations to speed up computation, and we show significant improvement using D-PMP to optimize the continuous energy model.

4.1. Optical Flow

Given a pair of (grayscale) images I_1 and I_2 in $\mathbb{R}^{M \times N}$, we estimate the motion of each pixel s from one image to the next. This *flow vector* x_s is decomposed into horizontal u and vertical v scalar components. The model presented below is based on the Classic-C method (Sun et al., 2014). To reduce the number of edges we model flow at the superpixel level, holding flow constant over each superpixel. Edges are given by the immediate neighbors in I_1.

The pairwise log-potential enforces a *smoothness* prior on flow vectors. We use the robust Charbonnier penalty, a differentiable approximation to L_1, which is approximately quadratic in the range $[-\sigma, \sigma]$ and smoothly transitions to a linear function outside this range. The potential decomposes additively as $\log \psi_{st} = \phi_{st}^{\mathrm{vert}} + \phi_{st}^{\mathrm{hor}}$ into vertical and horizontal components, defined as follows:

$$\phi_{st}^{\mathrm{hor}}(u_s, u_t) = -\lambda_s \sqrt{\sigma^2 + (u_s - u_t)^2}. \qquad (14)$$

The spatial smoothness depends on scaling parameter λ_s.

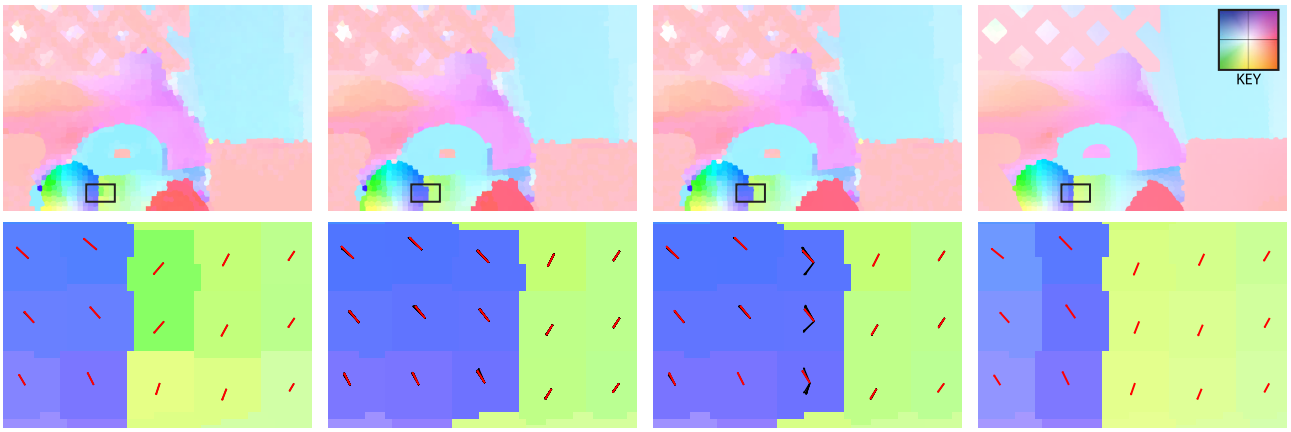
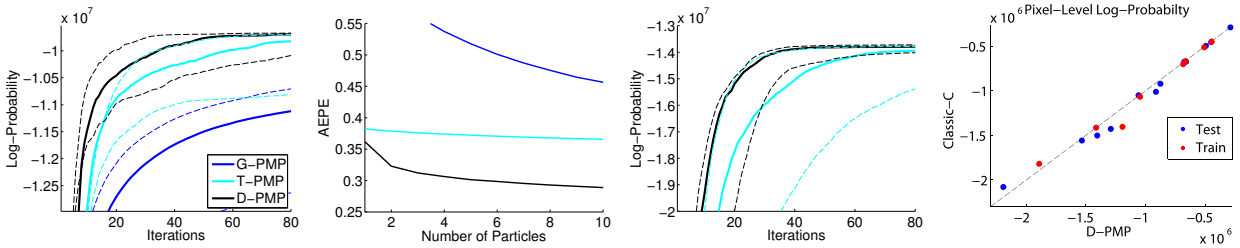


Figure 6. **Preserving multiple hypotheses.** *Top Row:* Final flow estimate of each method for the "Rubber Whale" sequence. The color key (top-right) encodes flow vector orientation, color saturation denotes magnitude. *Bottom Row:* Detail of highlighted region showing selected flow particles as vectors (*black*) and the MAP label (*red*). The MAP estimates of D-PMP and T-PMP have higher probability than ground truth, but D-PMP preserves the correct flow in the particle set.



Log-Probability | Oracle Avg. EPE | Log-Prob. (Test) | Pixel-Level Log-Prob.

Figure 7. **Optical flow results.** *Left:* Log-probability quantiles showing median (solid) and best/worst (dashed) MAP estimates versus PMP iteration for 11 random initializations on the Middlebury training set. *Left-Center:* Oracle AEPE over the training set. *Right-Center:* Log-probability quantiles on the test set (G-PMP omitted due to poor performance on training). *Right:* Log-probability of the MAP estimates at the pixel-level model obtained by initializing L-BFGS at the D-PMP solution.

Likelihood potentials $\log \psi_s(x_s) = \phi_s(x_s)$ assume brightness constancy: properly matched pixels should have similar intensities. Each superpixel s contains a number of pixels $\mathcal{I}_s = \{(i_1, j_1), \ldots, (i_k, j_k)\}$, and for each pixel (i, j) we compute the warped coordinates $(\widetilde{i}, \widetilde{j}) = (i+u_s, j+v_s)$. The likelihood penalizes the difference in image intensities, again using the Charbonnier penalty:

$$\phi_s(u_s, v_s) = -\lambda_d \sum_{(i,j)\in\mathcal{I}_s} \sqrt{\sigma^2 + (I_1(i,j) - I_2(\widetilde{i}, \widetilde{j}))^2} \quad (15)$$

In computing the warped coordinates we also constrain any pixels which flow outside the image boundary to be exactly on the boundary, $\widetilde{i} = \min(M, \max(0, i + u_s))$. We apply bicubic interpolation for non-integer coordinates.

Results We evaluate on the Middlebury optical flow benchmark (Baker et al., 2011) using 11 random initializations. D-PMP and T-PMP utilize the same set of proposals (75% neighbor, 25% random walk). We compute SLIC superpixels (Achanta et al., 2012) with region size 5 and regularizer 0.1; about 5,000 to 15,000 per image. We use the Charbonnier widths $\sigma = 0.001$ recommended for this model (Sun et al., 2014), but learn different scaling parameters ($\lambda_s = 16, \lambda_d = 1$) to compensate for our superpixel representation.

	Avg. Log-Prob. (p value)	Avg. EPE (p value)
RMP	-2.446E6 (0.008)	1.623 (0.008)
G-PMP	-1.408E6 (0.008)	0.699 (0.008)
T-PMP	-1.212E6 (0.008)	0.382 (0.727)
D-PMP	-1.209E6 (–)	0.362 (–)
Classic-C	–	0.349 (0.727)

Table 1. **Optical flow MAP estimates.** Average log-probability and AEPE over 11 random initializations on the Middlebury training set. Reported p values are compared to D-PMP using a Wilcoxon signed rank test, we consider $p < 0.05$ significant.

The Middlebury training set contains 8 images with ground truth flow, and we report log-probability quantiles over this set (Fig. 7 (left)). To demonstrate diversity in the particle sets we report average endpoint error (AEPE) of the *oracle solution*—we choose the flow particle closest to ground truth in the order given by the particle selection step

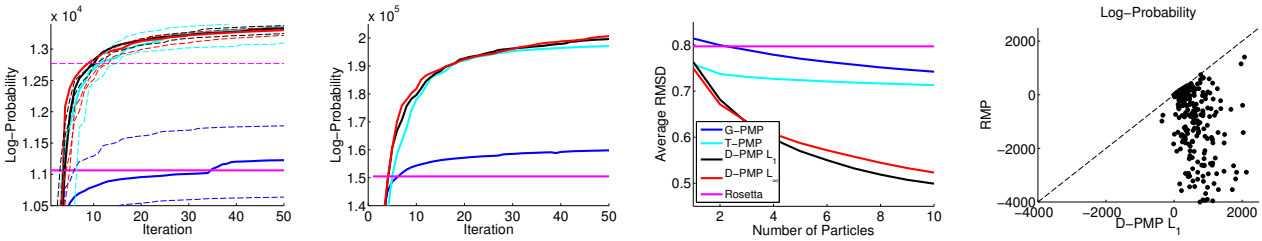


Figure 8. **Side chain prediction.** We compare each method and both L_1 and L_∞ diverse selection methods. *Left:* Total log-probability over 20 proteins. Median (solid) and best/worst (dashed) results on 11 random initializations. *Left-Center:* Total log-probability for 370 proteins. *Right-Center:* RMSD (in angstroms Å) of the oracle solution on larger set. *Right:* Log-probability of all 370 proteins versus the fixed rotamer discretization with RMP inference.

(Fig. 7 (left-center)). D-PMP shows a large reduction in AEPE after just a few particles. T-PMP remains nearly flat, suggesting little diversity. In just two dimensions the Gaussian spread of G-PMP particles naturally leads to an error reduction, although higher. The benefit of particle diversity is best visualized near object boundaries (see Fig. 6).

We compare to a specialized coarse-to-fine, multiscale inference algorithm for Classic-C[1], using default settings and with the median filter disabled. We also compare to RMP on a fixed regular discretization of 200 flow vectors. As shown in Table 1, D-PMP yields significantly higher probability solutions, but is equivalent to T-PMP in AEPE. D-PMP also achieves equivalent results to Classic-C optimization, which is highly tuned to the Middlebury dataset.

We cannot directly compare probability of the Classic-C and D-PMP solutions, because the former models flow at the pixel level. Instead, using L-BFGS initialized from the D-PMP solution, we optimize the pixel level model and compare log-probability of the result with Classic-C for both training and test sequences (Fig. 7 (right)). Again, even compared to a highly-tuned specialized optimization method, D-PMP achieves statistically equivalent results.

4.2. Protein Side Chain Prediction

Most computational approaches optimize side chain placement over a standard discretization, known as a *rotamer library* (Bower et al., 1997; Fromer et al., 2010). Rotamer configurations are learned from the marginal statistics of experimentally validated side chains and generally allocate three states $\{60°, 180°, 300°\}$ for each dihedral angle, resulting in up to 81 possible states per node. This is a coarse discretization which can fail to capture important details of the side chain placement. Applying D-PMP we optimize the continuous energy function, allowing estimation of *non-rotameric* side chains which do not obey the standard discretization (see Fig. 9). Log-likelihoods are the so-called *Dunbrack probabilities*—Gaussian mixtures with means centered on rotamer configurations.

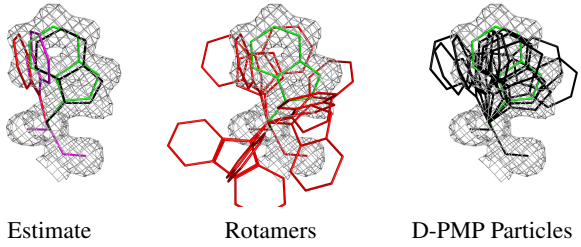


Figure 9. **Non-rotameric side chains.** *Left:* X-ray (green), RMP (red), Rosetta (magenta) and D-PMP (black) estimates. *Center:* Standard rotamers are all poor approximations. *Right:* Final D-PMP particles all overlapping the level set of the electron density (mesh). (PDB: 1GK9, Trp154) (Shapovalov & Dunbrack, 2007)

Results We evaluate the energy function with Rosetta (Rohl et al., 2004), a state-of-the-art molecular modeling package. We configure the Rosetta energy using three terms: the Lennard-Jones attractive and repulsive (`fa_atr, fa_rep`) terms and the Dunbrack probabilities (`fa_dun`), each with unit weight. We run PMP with 50 particles for 50 iterations. D-PMP and T-PMP proposals are 50% random walks from Gaussians wrapped to account for angular discontinuities, and 50% samples from the rotamer marginals. Neighbor-based proposals are not used, due to the complex transformation between dihedral angles and atom locations. We compare to Rosetta's implementation of simulated annealing using Metropolis-Hastings proposals from the discrete rotamers, followed by local continuous optimization.

We experiment on two sets of proteins selected from the Protein Data Bank[2], resolved using X-ray diffraction below 1.5-Å resolution, and less than 1000 amino acids. We run each method from 11 random initialization on a small set (20 proteins) and report quantiles of total log-probability (Fig. 8 (left)). Both D-PMP and T-PMP outperform G-PMP, due to their ability to exploit the model likelihood through rotamer proposals, with D-PMP showing the tightest confidence intervals. The second set is larger (370 proteins) and we report the total log-probability of a single run

[1] `http://people.seas.harvard.edu/~dqsun`
Experiments use code accessed on 06 February 2015.

[2] http://www.pdb.org

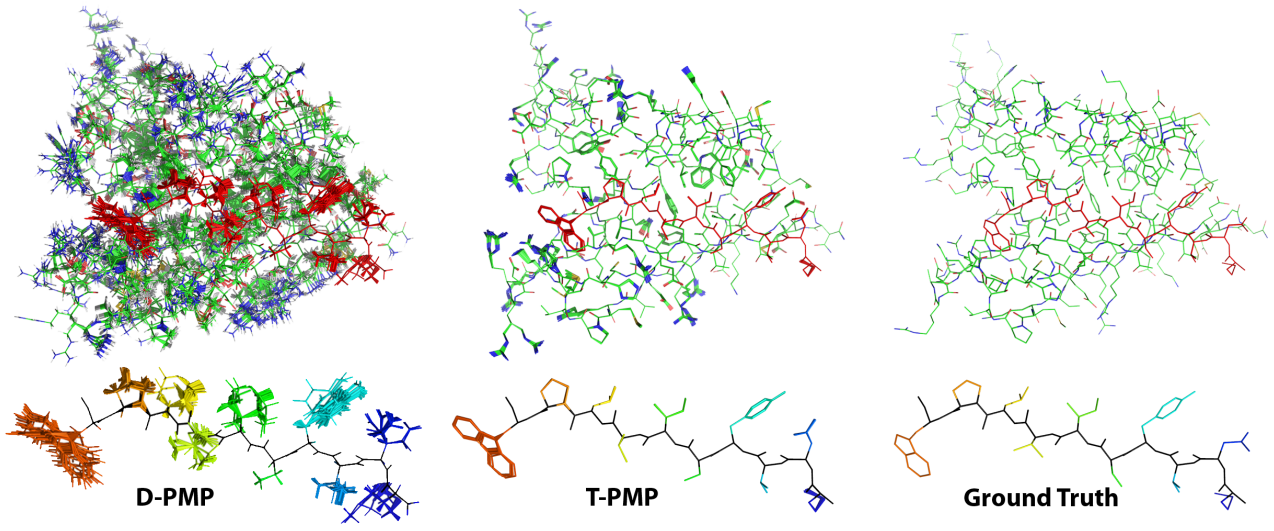


Figure 10. **Side chain particles.** *Top Row:* Final particles for T-PMP and D-PMP, and the ground truth conformation of a single protein (PDB ID: 2H8E). Region marked in red is detailed below. *Bottom Row:* Closeup of first ten amino acids, showing the fixed backbone (black) and final particles colored by backbone location. D-PMP preserves more diverse particles in areas of uncertainty.

for each method (Fig. 8 (left-center)).

Diversity is important in structure prediction, since proteins are known to alternate between many stable configurations (Ma et al., 1999). Fig. 10 shows a qualitative comparison of diversity between D-PMP and T-PMP for a single protein. To measure diversity we report RMSD of the oracle solution (Fig. 8); D-PMP shows a substantial improvement in accuracy after only a few particles. We also compare the submodular particle selection (L_1) with the *minimax* formulation (L_∞); both preserve diversity similarly, but the former offers stronger theoretical justification.

5. Discussion

We have generalized previous PMP algorithms in several substantial ways. Our proposed extensions to D-PMP not only allow inference in loopy MRFs, but our reformulation of the particle selection IP allows for greedy optimization within a guaranteed optimality bound. We demonstrate effectiveness in protein structure prediction, where we are substantially more accurate than the G-PMP algorithm that Peng et al. (2011) applied to a broader structure prediction task, and the state-of-the-art Rosetta package. The same general-purpose D-PMP algorithm is also competitive with standard inference algorithms for a very loopy optical flow model. A MATLAB library, built on UGM (Schmidt, 2007), implementing these methods is available[3].

Acknowledgements We thank Silvia Zuffi for her advice about connections between submodularity and the particle selection problem. This research supported in part by ONR Award No. N00014-13-1-0644. J. Pacheco supported in part by funding from the Naval Undersea Warfare Center, Division Newport, Rhode Island.

[3] http://www.cs.brown.edu/~pachecoj

Appendix. Proof of Proposition 1

To simplify we ignore normalization terms and drop dependence on z so $\hat{m}(z) = \hat{m}$. The proof is by induction on the number of neighbors, for the base case let $\Gamma(s) = \{i, j\}$:

$$\begin{aligned}\|\nu_s - \hat{\nu}_s\|_1 &\leq \sum_{x_s} \Big[(m_{is}(x_s)^{\rho_{is}} - \hat{m}_{is}(x_s)^{\rho_{is}})m_{js}(x_s)^{\rho_{js}} \\ &\qquad + (m_{js}(x_s)^{\rho_{js}} - \hat{m}_{js}(x_s)^{\rho_{js}})\hat{m}_{is}(x_s)^{\rho_{is}}\Big] \\ &\leq \sum_{x_s} \Big[(m_{is}(x_s)^{\rho_{is}} - \hat{m}_{is}(x_s)^{\rho_{is}}) \\ &\qquad + (m_{js}(x_s)^{\rho_{js}} - \hat{m}_{js}(x_s)^{\rho_{js}})\Big] \\ &\leq \|m_{is} - \hat{m}_{is}\|_1^{\rho_{is}} + \|m_{js} - \hat{m}_{js}\|_1^{\rho_{js}}\end{aligned}$$

The first inequality drops $\psi_s \in [0, 1]$, and $|\cdot|$ since $\hat{m}_{ts} \preceq m_{ts}$. The second inequality holds since $m, \hat{m} \in [0, 1]$. The last follows from the triangle inequality since, $|x - y|^\rho$ is a metric (though not a norm for $\rho \in (0, 1)$). For the inductive step let $\Gamma(s) = \{t_1, \ldots, t_n\}$ and assume:

$$\|\nu_s^{\setminus n} - \hat{\nu}_s^{\setminus n}\|_1 \leq \sum_{i \neq n} \|m_{t_i s} - \hat{m}_{t_i s}\|_1^{\rho_{t_i s}}$$

where $\nu_s^{\setminus n}(x_s)$ is the product of all messages except $m_{t_n s}$:

$$\begin{aligned}\|\nu_s - \hat{\nu}_s\|_1 &\leq \|m_{t_n s} - \hat{m}_{t_n s}\|_1^{\rho_{t_n s}} + \|\nu_s^{\setminus n} - \hat{\nu}_s^{\setminus n}\|_1 \\ &\leq \sum_{i=1}^{n} \|m_{t_i s} - \hat{m}_{t_i s}\|_1^{\rho_{t_i s}}.\end{aligned}$$

References

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Susstrunk, S. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE PAMI*, 34(11):2274–2282, 2012.

Aji, S. M. and McEliece, R. J. The generalized distributive law. *IEEE Info. Theory*, 46(2):325–343, 2000.

Andrieu, C., De Freitas, N., Doucet, A., and Jordan, M. I. An introduction to MCMC for machine learning. *JMLR*, 50(1-2):5–43, 2003.

Baker, S., Scharstein, D., Lewis, J. P., Roth, S., Black, M. J., and Szeliski, R. A database and evaluation methodology for optical flow. *IJCV*, 92(1):1–31, 2011.

Besse, F., Rother, C., Fitzgibbon, A., and Kautz, J. PMBP: Patchmatch belief propagation for correspondence field estimation. In *BMVC*, 2012.

Bower, M. J., Cohen, F. E., and Dunbrack Jr, R. L. Prediction of protein side-chain rotamers from a backbone-dependent rotamer library: a new homology modeling tool. *Journal of molecular biology*, 267(5):1268–1282, 1997.

Fromer, M., Yanover, C., Harel, A., Shachar, O., Weiss, Y., and Linial, M. Sprint: side-chain prediction inference toolbox for multistate protein design. *Bioinformatics*, 26 (19):2466–2467, 2010.

Geman, S. and Geman, D. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE PAMI*, 6(6):721–741, November 1984.

Kothapa, R., Pacheco, J., and Sudderth, E. Max-product particle belief propagation. Master's project report, Brown University Dept. of Computer Science, 2011.

Leskovec, J., Krause, A., Guestrin, C., Faloutsos, C., VanBriesen, J., and Glance, N. Cost-effective outbreak detection in networks. In *KDD*, pp. 420–429. ACM, 2007.

Ma, B., Kumar, S., Tsai, C.-J., and Nussinov, R. Folding funnels and binding mechanisms. *Protein Engineering*, 12(9):713–720, 1999.

Minoux, M. Accelerated greedy algorithms for maximizing submodular set functions. In *Optimization Techniques*, pp. 234–243. Springer, 1978.

Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. An analysis of approximations for maximizing submodular set functions. *Math. Prog.*, 14(1):265–294, 1978.

Pacheco, J., Zuffi, S., Black, M., and Sudderth, E. Preserving modes and messages via diverse particle selection. In *ICML*, pp. 1152–1160, 2014.

Pearl, J. *Probabilistic Reasoning in Intelligent systems: Networks of Plausible Inference*. Morgan Kaufmann, San Francisco, CA, 1988.

Peng, J., Hazan, T., McAllester, D., and Urtasun, R. Convex max-product algorithms for continuous MRFs with applications to protein folding. In *ICML*, 2011.

Rohl, C. A., Strauss, C. E. M., Misura, K. M. S., and Baker, D. Protein structure prediction using rosetta. *Methods in enzymology*, 383:66–93, 2004.

Schmidt, M. UGM: A matlab toolbox for probabilistic undirected graphical models. http://www.cs.ubc.ca/~schmidtm/Software/UGM.html, 2007.

Shapovalov, M. V. and Dunbrack, R. L. Statistical and conformational analysis of the electron density of protein side chains. *Proteins: Struct., Func., and Bioinf.*, 66(2): 279–303, 2007.

Soltan Ghoraie, L., Burkowski, F., Li, S. C., and Zhu, M. Residue-specific side-chain polymorphisms via particle belief propagation. *IEEE/ACM Trans. on Comp. Bio. and Bioinf.*, 2013.

Sun, D., Roth, S., and Black, M. J. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *IJCV*, 106(2):115–137, 2014.

Trinh, H. and McAllester, D. Unsupervised learning of stereo vision with monocular cues. In *BMVC*, 2009.

Wainwright, M. J. and Jordan, M. I. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1:1–305, 2008.

Wainwright, M. J., Jaakkola, T. S., and Willsky, A. S. Map estimation via agreement on trees: message-passing and linear programming. *Information Theory, IEEE Transactions on*, 51(11):3697–3717, 2005.

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

Yanover, C. and Weiss, Y. Approximate inference and protein-folding. In *Advances in neural information processing systems*, pp. 1457–1464, 2002.