Novel Approaches to the Protein Design Problem

Adi Mukund Jennifer Zou

April 20, 2015

1 Introduction

The protein design problem can be described as follows. Given a set of backbone coordinates $\mathbf{c} = (\vec{c_1}, \vec{c_2}, \ldots, \vec{c_n})$ and rotamer library R of length r, identify the optimal rotamer assignment sequence $\vec{r} = (r_1, r_2, \ldots, r_n)$, $r_i \in R$, $1 \leq i \leq n$ according to an energy function $E(\vec{r})$. This assignment sequence is known as the global minimum energy conformation (GMEC). This problem has been proven to be NP-hard [1], and algorithms such as DEE [2] have been created to allow for combinatorial pruning of the residue search space and make the design problem computationally tractable. However, such algorithms cannot guarantee a time complexity less than the worst-case $O(nr^n)$.

Recent graph based algorithms such as BWM* [3] use sparse residue interaction graphs in order to more efficiently compute functions over the residue space and identify optimal assignments more rapidly. Such graph-based algorithms have been able to achieve combinatorial speedups while maintaining provable accuracy and returning ensemble of minimum energy conformations.

Probabilistic models of protein design assign a probability distribution for rotamers in each position of the protein sequence. In algorithms such as belief propagation, the beliefs or the approximate marginal probabilities of each rotamer are computed iteratively. These approximations are computationally less intense, but they are not always provably accurate [4]. Furthermore, such algorithms are only guaranteed to converge on tree graphs, which are not commonly observed in natural proteins [5].

2 Graph Cuts and the GMEC

2.1 The Graph Labeling Problem

The goal of this project was to apply graph-based algorithms to the protein design problem. We began by attempting to characterize the protein design problem as a graph cut problem and identifying relevant algorithms that might allow efficient approximations of the GMEC.

The protein design problem is most accurately represented not as solely a graph cut problem, but as a graph labeling problem, where each rotamer is a label. The Graph Labeling (GL) problem can be stated as follows: classify a set $\mathcal{V}$ of n objects by assigning to each object a label from a given set $\mathcal{L}$ of labels, given a weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$. For each $p \in \mathcal{V}$ there is a label cost $\mathbf{c}_p(a) \geq 0$ for assigning the label $a = f_p$ to p, and for every edge pq there is a pairwise cost $\mathbf{c}_{pq}(a,b) = w_{pq}d_{pq}(ab)$ where $d_{pq}(ab)$ is the distance between (or cost of) label a on vertex p and b on vertex q. Thus, the cost of a labeling f is as follows:

$$\mathrm{COST}(f) = \sum_{p \in \mathcal{V}} \mathbf{c}_p(f_p) + \sum_{(p,q) \in \mathcal{E}} w_{pq} d_{pq}(f_p, f_q) \tag{1}$$

The most promising algorithm for an efficient solution was [8], which provides an approximation algorithm based on graph cuts for the non-metric labeling problem, which requires a distance function $d(a,b)$ such that $d(a,b) = 0 \iff a = b$ and $d(a,b) \geq 0$. The algorithm provides a labeling with a cost that is an f-approximation to the minimum-cost labeling, where $f = \frac{d_{max}}{d_{min}}$, where d_{max} is the maximum distance between any two rotamers and d_{min} is the minimum distance.

2.2 Application to the Protein Design Problem

The internal energy of a protein can easily be modeled by equation 1, where the function $\mathbf{c}_p(f_p)$ is held to represent the internal energy of a rotamer f_p at position p and the function $w_{pq}d_{pq}(f_p, f_q)$ is held to represent the pairwise interaction of a rotamer f_p at position p and a rotamer f_q at position q. In addition, each residue is modeled by a single node in the vertex set $\mathcal{V}$, and interactions between residues are represented by edges between nodes in the edge set $\mathcal{E}$.

In order to maintain the distance constraint $\mathbf{c}_{pq}(a,b) = w_{pq}d_{pq}(ab)$, the label set L was set as the Cartesian product $R \times \mathcal{V}$ of the set of all rotamers with the set of all positions. Thus, a given label $l \in \mathcal{L}$ represents a specific rotamer at a particular position. The pairwise interaction between a label and itself can then be set as zero. In order to ensure that a rotamer for position 1 was not assigned to position 2, the cost of labeling a rotamer to the wrong position was set to be prohibitively high.

3 Methods

3.1 Selection of proteins

3.2 Generation of pairwise interaction matrices

OSPREY ([6], [7]) was used to generate pairwise interaction matrices and provide a baseline against which to compare the results generated by the graph -based algorithm.

3.3 Optimizing the metric labeling problem

The FastPD Markov Random Field optimization library ([8] and [9]) was used in order to test the graph algorithm. The FastPD algorithm transforms the initial GL problem into the minimization of a linear program according to [10].

$$
\begin{aligned}
min \sum_{p \in V} \sum_{a \in L} c_p(a) x_p(a) + \sum_{(p,q) \in E} w_{pq} \sum_{a,b \in L} d(a,b) x_{pq}(a,b) \\
s.t. \sum_{a} x_p(a) = 1 \qquad \forall p \in V \\
\sum_{a} x_{pq}(a,b) = x_q(b) \qquad \forall b \in L, (p,q) \in E \\
\sum_{b} x_{pq}(a,b) = x_p(a) \qquad \forall a \in L, (p,q) \in E
\end{aligned}
\tag{2}
$$

Equation 2 shows a formulation of an integer linear program, where $x_p(a)$ is a binary variable that indicates whether vertex p is assigned to label a, and $x_{pq}(a,b)$ indicates that vertices p and q are assigned the labels a and b. The first constraint requires each vertex to be assigned to one label, and the other constraints maintains consistency between the labeled vertices and edges.

Optimizing the cost of the metric labeling problem is NP-hard, and many approximation algorithms (α-expansion, α-β-swap) either assume metric distances that satisfy the triangle inequality or provide no guarantees about optimality. Since pairwise interaction energies between protein residues are inherently nonmetric distances, these algorithms are not adequate for the protein design problem.

FastPD only requires a nonmetric distance function that satisfies $d(a, b)$ such that $d(a, b) = 0 \iff a = b$ and $d(a, b) \geq 0$. Since the pairwise interaction energy between a residue and itself is zero, and all energies are greater than zero, this algorithm can be applied to the protein design problem.

Relaxation of the integer linear program constraints transforms the NP-hard optimization problem into one that can be solvable in polynomial time. The FastPD algorithm modifies the constraints to $s_p(\cdot) \geq 0$ and $x_{pq}(\cdot, \cdot) \geq 0$.

3.4 The Primal-Dual Schema

The FastPD algorithm utilizes the primal-dual schema in order to rapidly obtain an f-approximation to the optimal solution. Given a primal program

$$\begin{gathered} \min \mathbf{c}^T \mathbf{x} \\ \text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq 0 \end{gathered}$$

the dual program can be formulated as follows:

$$\begin{gathered} \max \mathbf{b}^T \mathbf{y} \\ \text{s.t. } \mathbf{A}^T \mathbf{y} \leq \mathbf{c} \end{gathered}$$

The primal-dual principle states that if $\mathbf{x}$ and $\mathbf{y}$ are solutions satisfying $\mathbf{c}^T \mathbf{x} \leq f \cdot \mathbf{b}^T \mathbf{y}$, then $\mathbf{x}$ is an f-approximation to the optimal solution $\mathbf{x}*$. FastPD starts with initial guesses for $\mathbf{x}$ and $\mathbf{y}$ and repeatedly improves them by making every variable in the program a node in the graph and using a max-flow/min-cut algorithm to alter the assigned values to each variable. The algorithm converges to an approximation factor $f_{app} = 2\frac{d_{max}}{d_{min}}$ where d_{max} is the largest distance between two labels and d_{min} is the smallest distance between two labels.

4 Results

1. DHFR
2. 1CC8
3. 1FSV

References

[1] Pierce, Niles A. and Winfree, Erik. Protein Design is NP-hard Protein Eng. (2002) 15 (10): 779-782 doi:10.1093/protein/15.10.779

[2] Dahiyat, B. I. De Novo Protein Design: Fully Automated Sequence Selection. Science 278, 82-87 (1997)

[3] Jou, J.D. Jain, S. Georgiev, I. Donald, B.R. BWM*: A Novel, Provable, Ensemble-based Dynamic Programming Algorithm for Sparse Approximations of Computational Protein Design. RECOMB (2015) Warsaw, Poland. April 12, 2015 (In Press)

[4] Kamisetty H1, Xing EP, Langmead CJ. Free energy estimates of all-atom protein structures using generalized belief propagation. J Comput Biol. 2008 Sep;15(7):755-66. doi: 10.1089/cmb.2007.0131.

[5] M. Fromer, C. Yanover, and M. Linial. Design of multispecific protein sequences using probabilistic graphical modeling. Proteins 75;3(2009 May 15):682-705.

[6] C. Chen, I. Georgiev, A. C. Anderson, and B. R. Donald. Computational structure-based redesign of enzyme activity. PNAS USA, 106(10): 37643769, 2009.

[7] P. Gainza, K. E. Roberts, I. Georgiev, R. H. Lilien, D. A. Keedy, C. Chen, F. Reza, A. C. Anderson, D. C. Richardson, J. S. Richardson, and B. R. Donald. OSPREY: Protein design with ensembles, flexibility, and provable algorithms. Methods in Enzymology, 523:87-107, 2013.

[8] Komodakis, N.; Tziritas, G., "Approximate Labeling via Graph Cuts Based on Linear Programming," Pattern Analysis and Machine Intelligence, IEEE Transactions on , vol.29, no.8, pp.1436,1453, Aug. 2007 doi: 10.1109/TPAMI.2007.1061

[9] N. Komodakis, G. Tziritas and N. Paragios, "Performance vs Computational Efficiency for Optimizing Single and Dynamic MRFs: Setting the State of the Art with Primal Dual Strategies". Computer Vision and Image Understanding, 2008 (Special Issue on Discrete Optimization in Computer Vision).

[10] C. Chekuri, S. Khanna, J. Naor, and L. Zosin, Approximation Algorithms for the Metric Labeling Problem via a New Linear Programming

Formulation, Proc. 12th Ann. ACM-SIAM Symp. Discrete Algorithms, pp. 109-118, 2001.