# Local Rules for Protein Folding on a Triangular Lattice and Generalized Hydrophobicity in the HP Model


Richa Agarwala* NCHGR/NIH

Serafim Batzoglou† MIT

Vlado Dančík‡ USC

Scott E. Decatur§ MIT

Sridhar Hannenhalli¶ USC

Martin Farach‖ Rutgers University

S. Muthukrishnan** Bell Labs

Steven Skiena†† SUNY Stony Brook



**Abstract**

We consider the problem of determining the three-dimensional folding of a protein given its one-dimensional amino acid sequence. We use the HP model for protein folding proposed by Dill [7], which models protein as a chain of amino acid residues that are either hydrophobic or polar, and hydrophobic interactions are the dominant initial driving force for the protein folding. Hart and Istrail [9] gave approximation algorithms for folding proteins on the cubic lattice under HP model. In this paper, we examine the choice of a lattice by considering its algorithmic and geometric implications and argue that triangular lattice is a more reasonable choice. We present a set of folding rules for a triangular lattice and analyze the approximation ratio they achieve. In addition, we introduce a generalization of the HP model to account for residues having different levels of hydrophobicity. After describing the biological foundation for this generalization, we show that in the new model we are able to achieve similar constant factor approximation guarantees on the triangular lattice as were achieved in the standard HP model.



*National Center for Human Genome Research/National Institutes of Health, Bethesda, MD 20892, (richa@helix.nih.gov) under a contract with R.O.W. Sciences, Inc. Most of this work was done while this author was at DIMACS center, Rutgers University and was supported by Special Year National Science Foundation grant BIR-9412594.

†MIT Laboratory for Computer Science and Department of Mathematics, 545 Technology Square, Room 342, Cambridge, MA 02139. (serafim@theory.lcs.mit.edu) Supported by a DOE Contract.

‡Department of Mathematics, University of Southern California, Los Angeles, CA 90089-1113, USA. (dancik@hto.usc.edu, http://www-hto.usc.edu/people/Dancik.html) Supported by NIH grant GM36230.

§MIT Laboratory for Computer Science and Department of Mathematics, 545 Technology Square, Room 313, Cambridge, MA 02139. (sed@theory.lcs.mit.edu, http://theory.lcs.mit.edu/~sed) Supported by a grant from the Reed Foundation through the MIT School of Science.

¶Department of Mathematics, University of Southern California, Los Angeles, CA-90089-1113, USA. (hannenha@hto.usc.edu, http://www-hto.usc.edu/people/Hannenhalli.html) Supported by NSF Young Investigator Award, NIH grant 1R01 HG00987 and DOE grant DE-FG02-94ER61919.

‖Department of Computer Science, Rutgers University, Piscataway, NJ 08855, USA. (farach@cs.rutgers.edu, http://www.cs.rutgers.edu/∼farach) Supported by NSF Career Development Award CCR-9501942, an Alfred P. Sloan Research Fellowship, and NATO Grant CRG 960215.

**Information Sciences Center, Bell Labs, Lucent Technologies, 700 Mountain Avenue, Murray Hill, NJ 07974. (muthu@research.bell-labs.com).

††Department of Computer Science, State University of New York, Stony Brook, NY 11794-4400, USA. (skiena@cs.sunysb.edu, http://www.cs.sunysb.edu/∼skiena) Supported by ONR award 400x116yip01 and NSF Grant CCR-9625669.

While the structures derived from our folding rules are probably still far from biological reality, we hope that having a set of folding rules with different properties will yield more interesting folds when combined.

# 1 Introduction

A long standing problem in molecular biology is to determine the three-dimensional structure of a protein when only given the sequence of amino acid residues which compose the protein chain. Due to the complexity of the protein folding problem, scientists have proposed a variety of models which attempt to simplify the problem by abstracting only the "essential physical properties" of real proteins. In these models, the three dimensional space is often represented by a *lattice*. Residues which are adjacent in the primary sequence (*i.e.* covalently linked) must be placed at adjacent points in the lattice. A *conformation* of a protein is simply a self-avoiding walk along the lattice. The protein folding problem *STRING-FOLD* is that of finding a conformation of the protein sequence on the lattice such that the overall *energy* is minimized, for some reasonable definition of energy. The lattice formulation described so far leaves open two important questions: what type of lattice should one use; and what energy function is appropriate. Once these two questions have been answered, one may then address the algorithmic complexity of optimizing the energy function for the lattice. For a variety of such simple models, this minimization problem is in fact NP-hard [16, 15, 18].

In this paper, we consider the *Hydrophobic-Polar (HP) Model* introduced by Dill [7]. The HP model abstracts the problem by grouping the 20 amino acids which compose proteins into two classes: hydrophobic (or non-polar) residues and hydrophilic (or polar) residues. For concreteness, we will take our input to be a finite string over the alphabet $\{\mathtt{H},\mathtt{P}\}^+$, where P represents polar residues, and H represents hydrophobic residues. Dill *et.al.* [6] survey the literature analyzing this model.

## 1.1 Selecting an energy function.

Given a conformation, we say that a pair of residues form a *topological contact* (or simply contact) if the residues are not covalently linked (i.e. they are not adjacent along the sequence) and are placed on neighboring lattice points. A *bond* refers to a topological contact between a pair of H's. Define the *free energy* of a conformation as $(-1)\times(\#\text{ of bonds})$. The optimal conformation for the protein is the one which has the lowest free energy.

The biological foundation of this energy function is the belief that the first-order driving force of protein folding is due to a "hydrophobic collapse" in which those residues which prefer to be shielded from water (hydrophobic residues) are driven to the core of the protein, while those which interact more favorably with water (polar residues) remain on the outside of the protein. The protein is hypothesized to fold in such a way as to minimize the surface area of hydrophobic residues exposed to water or polar residues.

## 1.2 Selecting a lattice.

HP simulations have typically followed Dill's original choice of square lattices. Unfortunately, one rather severe consequence of the structure of the square lattice is that no two amino acids can be in adjacent lattice points if the string between them is of odd length. We call this the *parity constraint* of square lattices. Thus, the string $(\mathtt{PH})^n$ has no bonds in a square lattice, despite the fact that such a protein string has many potential bonds in "real space".

The bizarreness of the parity constraint illustrates another possible pitfall of algorithmic analysis of this problem. An approximation ratio for a maximization algorithm is the ratio of a lower bound on the performance of the algorithm and an upper bound on the optimal solution. Thus, it is desirable to raise the lower bound of the algorithmic solution, or to lower the upper bound of the optimal solution. But in the case of the *STRING-FOLD* problem on a square lattice, the upper bound is artificially low, due to the parity constraint. To illustrate, consider once again the string $(\texttt{PH})^n$, which has a trivial upper bound of 0 bonds, so any algorithm will achieve the optimum. Thus, any approximation ratio on a square lattice will have little meaning for moving towards a "realistic" solution of the problem.

If the square lattice is so seriously flawed, what is a more interesting lattice choice? We propose using triangular lattices as a folding model. The two- and three-dimensional triangular lattices are shown in Figure 1 and Figure 12.[1] The triangular lattice does not exhibit the parity problem. That is, for every pair of sites $x$, $y$ on any string, there exists a confirmation of the string on the triangular lattice such that $x$ and $y$ are neighboring sites on the lattice.

The triangular lattice represents the dual of the (thought to be) densest packing of unit spheres in both two and three dimensions [4]. This provides an additional geometric argument to support the triangular lattice model under the assumption of amino acids of roughly similar size. The higher "kissing" numbers of triangular lattice packings (6 in $R^2$ and 12 in $R^3$) over square lattice packings (4 in $R^2$ and 6 in $R^3$) provide greater freedom to construct conformations, as evidenced by the absence of a parity constraint. Finally, using a triangular lattice enables the core of the protein to take on more natural shapes. Consider the string `HHPHPHPHPHPH`. On a two dimensional triangular lattice, the first `H` can be the centroid and rest of the `H`'s form a hexagon around it. This makes a better fit for the sphere as against the ones we can get with square lattice.

### 1.3 Past results.

From a computational point of view, it is not known whether or not the problem of finding an optimal conformation using the above specified energy function (on either square or triangular lattices) is NP-hard. The problem is known to be NP-complete when the alphabet size is unbounded and the lattice is a two- or three-dimensional square lattice [17]. Hart and Istrail [9] presented algorithms with approximation factor of 1/4 on the two-dimensional square lattice and 3/8 on the three-dimensional square lattice. Also Bornberg-Bauer [3] implemented chain growth algorithms and Hart [11] investigates the inverse protein folding problem for the square lattice.

Unfortunately optimal conformations on square lattices may be arbitrarily worse than the optimal conformations on lattices without the parity problem. It is therefore interesting to examine the HP folding problem on triangular lattices, and to strive for conformations approaching the more natural optimal score found there. The utility of the triangular lattice was also observed independently by Kleinberg [13]. Based on our star folding Hart and Istrail [12] are able to achieve 31/36-approximation for sequences with side chains on the face-centered cubic lattice.

### 1.4 Our results.

We present a collection of local folding rules for the HP model on triangular lattices, in both two and three dimensions, and we prove approximation ratios for each of these rules. For all of our rules, these ratios are better than those achieved in by Hart and Istrail [9] for the square lattice. As

[1] This three-dimensional lattice is based on the topology of the $\beta$ form of Silicon Carbide. The nodes in this lattice correspond to the silicon atoms in the Silicon Carbide crystal. Two nodes in this lattice are connected if there exists a carbon atom that is bonded to both of the two silicon atoms corresponding to the nodes.

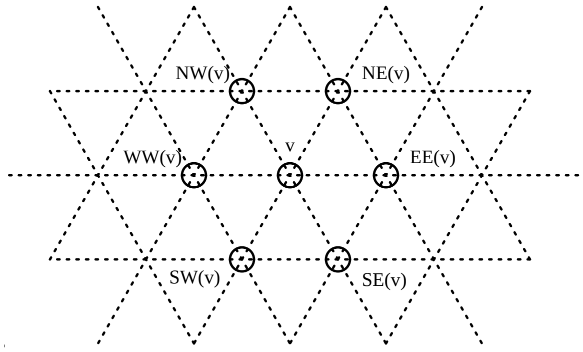


Figure 1: The two-dimensional triangular lattice

pointed out above, an approximation ratio can often be a misleading. We provide these numbers as a rough guideline, since no more appropriate measure of goodness for a rule is available. Yet, since optimal solutions for triangular lattices are so much more densely packed than are optimal solutions for square lattices, we achieve our approximations by local rules which yield very dense structures. That is, since the upper bound of the optimal solution is much higher, the performance of the folding algorithms must increase correspondingly.

In Sections 2 and Sections 3, we describe the folding rules for the two- and three-dimensional lattices, respectively. All of our folding rules are implementable in linear time. The following table summarizes the proposed rules for protein folding under the HP model in the two- and three-dimensional triangular lattices.

| | backbone folding | improved backbone folding | arrow folding | improved arrow folding | star folding | combined folding | improved star folding |
|---|---|---|---|---|---|---|---|
| 2D | 1/4 | 1/2 | 1/2 | 6/11 | n/a | n/a | n/a |
| 3D | 3/10 | 2/5 | 14/30 | n/a | 16/30 | 44/75 | 3/5 |

In Section 4, we extend the HP model by considering a more general representation of hydrophobic residues. The new model is motivated by the fact that certain hydrophobic residues are more hydrophobic than others. While in the standard HP model all hydrophobic residues have identical hydrophobicities, our new model allows different residues to have different hydrophobicities and contacts between hydrophobic residues contribute to the energy function proportional to their combined hydrophobic strength. After describing the biological foundation for this generalization, we show that in the new model we are able to achieve similar constant factor approximation guarantees on the triangular lattice as were achieved in the standard HP model.

## 2 Rules for Two Dimensions

The two dimensional triangular lattice is shown in Fig. 1. Every node in this lattice has 6 neighbors. For a node $v$, denote the six neighbors of $v$ as shown in Fig. 1.

For a binary sequence $S$, let $s_1$ be the number of `H`'s in $S$. For simplicity, assume that $S$ begins and ends with a `P`. For a conformation $f$ of $S$, let $C(S, f)$ denote the number of bonds.

**Lemma 1** *For every conformation $f$ of binary sequence $S$ in the two dimensional triangular lattice*

$$C(S, f) \leq 2 \cdot s_1.$$

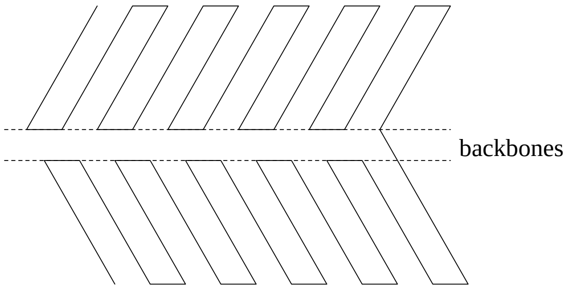


Figure 2: The backbone-folding

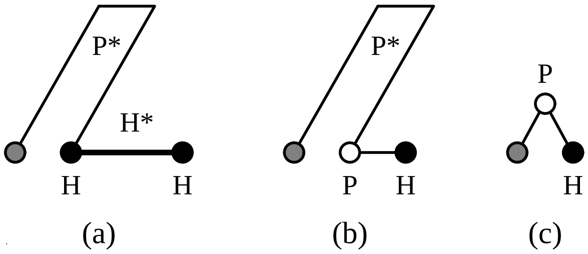


Figure 3: The conformation of $\mathtt{P}^i\mathtt{H}^j$, $i, j > 0$ when (a) $j > 1$, (b) $j = 1$, $i > 1$, (c) $j = 1$, $i = 1$. White circles represent P, black circles represent H, and gray circles represent H preceding $\mathtt{P}^i$. Thick line represents sequence of $j - 2$ H's.

**Proof** Since any internal node (not corresponding to the beginning or the end of the sequence) can have at most 4 topological neighbors in the two dimensional triangular lattice (out of six neighbors, two are connected along the sequence), $C(S, f) \leq 2 \cdot s_1$. ■

Notice that a H-node at the beginning or at the end of the sequence can have 5 topological neighbors leading to $C(S, f) \leq 2 \cdot s_1 + 1$. This situation is avoided due to the simplifying assumption.

## 2.1 Backbone-folding: 1/4-Approximation

We describe a very simple folding strategy based on laying out two *backbones* anti-parallel to each other such that most of the bonds are achieved among the H's along a backbone and the H's across the backbones. The general scheme of backbone folding is shown in Fig. 2. We refer to this conformation as *backbone-folding*. This is the folding rule used in [10].

Fig. 3 describes the rules for backbone folding of sequence $\mathtt{P}^+\mathtt{H}^+$. It is easy to see that these rules create a conformation satisfying the following two conditions: (1) every H is placed on the backbone, (2) no two P's are consecutive along the backbone .

**Lemma 2** *Let $f^b$ be the backbone-folding of binary sequence $S$. Then*

$$C(S, f^b) \geq \frac{1}{2} s_1.$$

**Proof** Each node of the backbone has two topological neighbors on the other backbone and one of those has to be H. All H's are on the backbones, each creates at least one bond and therefore $C(S, f^b) \geq \frac{1}{2} s_1$. ■

Lemmas 1 and 2 together imply

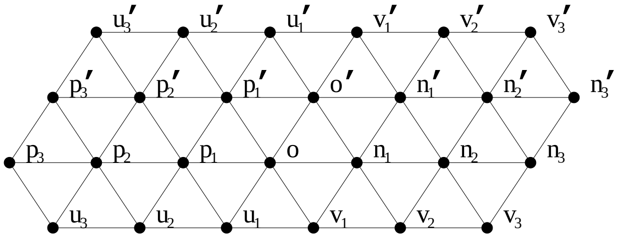


Figure 4: Position labels for the 2D construction.

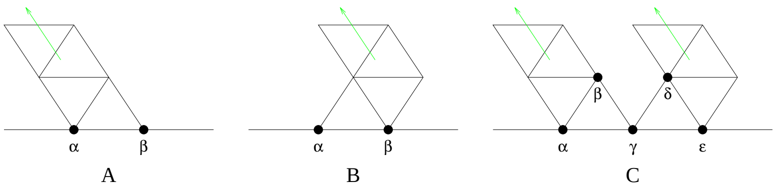


Figure 5: Shape types for Full regions.

**Corollary 1** *Backbone-folding yields a 1/4-approximation for the STRING-FOLD problem in the HP model for two-dimensional triangular lattices for large values of* $s_1$.

### 2.2 Improved Backbone-folding: 1/2-Approximation

We give a folding algorithm for the two-dimensional triangular lattice and prove that its performance is guaranteed to be at least 1/2 of optimal. Recall from Lemma 1 that the optimal conformation of a protein on the 2d lattice has 2 bonds per H.

The algorithm starts folding the protein by bringing the first and last H-residues together. It then folds it up to the middle forming a backbone of two parallel lines. The backbone is divided into two types of regions, namely full (F) and transition (T) regions. Both lines of the backbone enter and exit each region together. In full regions, only H-residues are placed on the backbone and ladders around the two lines forming the backbone are used to accommodate any intervening P-residues. When in the process of building a full region all ladders for P-residues are exhausted, the full region stops there and a transition region starts. When ladders for P-residues are available again, a new full region starts. Full can be arbitrarily long and each guarantees 1 bond per H-residue, plus a few additional bonds. These additional bonds are counted together with the bonds of the next transition region,[2] so as every region to guarantee at least 1 bond per H-residue.

The algorithm starts at the two ends of the protein, brings the first and last H-residues together, and incrementally fixes the positions of the residues up to the middle using as a skeleton two adjacent parallel lines in the lattice, $\ell_1$ and $\ell_2$. The rules below treat lines $\ell_1$ and $\ell_2$ identically. From now on we refer to them as $\ell, \ell'$ where we do not specify which is which. Figure 4 defines a reference system around a point $o$ on line $\ell$. The convention is that $o'$ is the point on $\ell'$ opposite to $o$, and to the direction opposite to the endpoints of the protein.

Some notation: We say that $o = x$ where $x \in \{\mathtt{H}, \mathtt{P}, \varepsilon\}$ to specify the contents of $o$ with $\varepsilon$ standing for "empty". We say $o \equiv x$ where $x$ is one of $\alpha, \beta$, etc. when referring to folds in Figure 5, to specify that the $x$ point in the fold coincides with the point $o$ we refer to.

The $F$-regions start by placing two H-residues one on each line, and touching. For each $\ell$ (=

[2] Transition regions can contribute fewer bonds than 1 per H-residue.

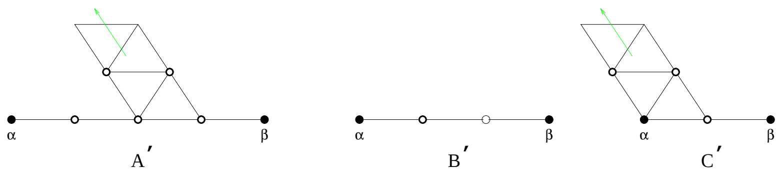


Figure 6: Shape types for Transition regions.

$\ell_1$ or $\ell_2$) if we call $o$ the starting point of the $F$-region on $\ell$, the $T$-rules make sure that $u_1 = \varepsilon$. The $F$-rules apply under the following conditions: (1) The last H-residue placed on $\ell$ was placed on $o \in F$. (2) $o' \in F$ (3) $v_1 = \varepsilon$. Whenever $o \in F$ and either (2) or (3) fails, then $n_1$ will be the first residue of $\ell$ in the new $T$-region and the rules for $T$-regions apply. Denote by $R_0, R_1, R_2, \ldots$ the subsequence of residues starting from $o$ and towards $o'$. Denote by $R'_0, R'_1, R'_2, \ldots$ the subsequence of residues starting from $o'$ and towards $o$. Notice that the ladders in folds A,B,C contain only H-residues.

**Rules for $F$-regions:**

1. **If** $u_1 = \varepsilon$ **then**
   **If** $R_0 \cdots R_k =$ HP*HHH and $n'_1 \in F$
   **then** fold according to C letting $o \equiv \alpha$
   **Else** fold according to A letting $o \equiv \alpha$.

2. **If** $u_1 \neq \varepsilon$ **then**
   **If** $R_0 \cdots R_k =$ HHP*H
   **then** fold according to C letting $o \equiv \gamma$
   **Else** fold according to B letting $o \equiv \alpha$

The $T$-region consists of one application of the rules for $T$-regions. That is to say, one of the folds A′, B′, C′ in Figure 6 is chosen for $\ell_1, \ell_2$, whereby the $T$-region lies between, and not including, the points $\alpha, \beta$ on these folds. (The empty circles in the folds of Figure 6 represent positions where H-residues may be placed.) The two points $\alpha$ (one for each line) are the end of the preceding $F$-region, and the two points $\beta$ are the beginning of the succeeding $F$-region. The idea is for $T$ to be kept short, and in particular not to contain "too many" H-residues. Towards that aim, the following restriction applies: in all rules, we put as few residues in the ladders while still assuming that the $\beta$ position of the fold is an H-residue. That is to say, when using A′ the first H-residue among $R_3, R_4, \ldots$ is placed on $\beta$. When using C′ the first among $R_2, R_3, \ldots$ is placed on $\beta$.[3] As a consequence, the only residue that can be H off the line $\ell$, when using A′ is $R_3$, and when using C′ is $R_1$. In the rules below, let $o, o'$ be the last two residues of the preceding $F$-region, and let $R_0, R_1, \ldots$ and $R'_0, R'_1, \ldots$ be the subsequences starting at $o, o'$, respectively.

**Rules for $T$-regions:**

1. **If** $u'_1 = \varepsilon$ **then**
   **If** $R_2 =$ H **then** use C′ on $\ell$ and $\ell'$
   **Else If** $R_3 =$ H **then** use C′ on $\ell'$ and B′ on $\ell$
   **Else If** $R'_3 =$ H **then** use B′ on $\ell'$ and A′ on $\ell$

[3]This implies that an H-residue *can* be put on ladders of A′ or C′, which was prohibited for ladders in the $F$-region.

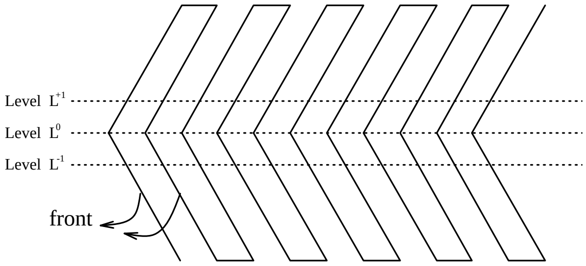


Figure 7: The arrow-folding

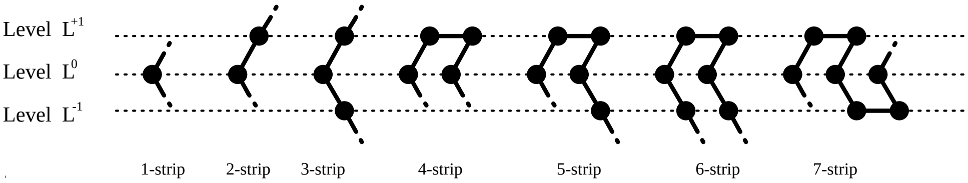


Figure 8: Placement of strips in the arrow-folding

**Else** use A$'$ on $\ell$ and $\ell'$

2. **If** $u'_1 \neq \varepsilon$ **then**
   **If** possible, **then** choose the shortest two flat folds for both $\ell$ and $\ell'$ s.t. the $\beta$ positions of both folds touch.
   **Else If** $R'_3 = \mathtt{H}$ **then** use B$'$ on $\ell'$ and A$'$ on $\ell$.
   **Else** use A$'$ on $\ell$ and $\ell'$.

**Lemma 3** *Let $f^B$ be the improved backbone-folding of binary sequence $S$. Then*

$$C(S, f^B) \geq s_1.$$

Lemmas 1 and 3 together imply

**Corollary 2** *Improved backbone-folding yields a 1/2-approximation for the STRING-FOLD problem in the HP model for two-dimensional triangular lattices for large values of $s_1$.*

### 2.3 Arrow-folding: 1/2 Approximation

Define a *strip* as a maximal substring of $S$ containing only `H`'s. A *k-strip* is a strip of size $k$. For example, `HHHH` is a 4-strip starting at position 2 of $S =$ `PHHHHPHPPPPHHPHHHHHH`. Define a *separator* to be a maximal substring of $S$ containing only `P`'s and a *k-separator* likewise. For example, `PPPP` is a 4-separator between 1-strip and 2-strip in the previous example. A $k$-strip ($k$-separator) is *long* if $k > 1$. In the following we describe a certain conformation which we call *arrow-folding* and show that for this conformation, we get at least $s_1$ bonds, leading to a 1/2-approximation.

Fig. 7 shows the general scheme of arrow-folding. Define levels $L^0$, $L^{+1}$ and $L^{-1}$ as shown in Fig. 7. We fold a sequence along this pathway such that (i) all the `H`'s are distributed only in $L^0$, $L^{+1}$ and $L^{-1}$ and (ii) all the nodes on level $L^0$ are `H`'s. This is achieved by placing the strips as shown in Fig. 8. In the figure we show the case of $k-$strips, $1 \leq k \leq 7$, but it follows a general pattern as is evident in the figure. For $k \equiv 0(mod 3)$, both the first and the last nodes of the strip

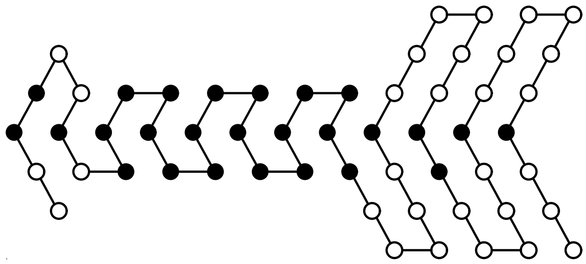

Figure 9: Example of arrow-folding for the sequence (PPHHPPHPHHHHHHHHHHHHHHHHHHHPPPPPHPPP-PPPHHPPPPPHPPPPPPHPPP).

are on level $L^{+1}$ or $L^{-1}$. For $k \equiv 1(mod3)$, both the first and the last nodes of the strip are on level $L^0$. For $k \equiv 2(mod3)$, the first node of the strip is on level $L^0$ and the last node of the strip is on level $L^{+1}$ or $L^{-1}$. Notice that a mirror image with respect to the $L^0$ line does not affect the general schema of folding. We connect two consecutive strips using the separators such that the entire sequence still follows the general schema dictated by arrow-folding. The precise description of arrow-folding is cumbersome and unnecessary. Fig. 9 shows an example of arrow-folding.

**Lemma 4** *Let $f^a$ be the arrow-folding of binary sequence $S$. Then*

$$C(S, f^a) \geq s_1 - 3.$$

**Proof** To compute the number of bonds, notice that every H node $v$ on level $L^0$ pairs exclusively to the H on $L^0$ to it's right, *i.e.* $EE(v)$ (except for the terminal case). Also, every H node $v$ at level $L^{+1}$ ($L^{-1}$) pairs exclusively to the H node $SE(v)$ ($NE(v)$) at level $L^0$. Simple counting gives the number of bonds as $s_1 - 3$. ∎

Lemmas 1 and 4 together imply

**Corollary 3** *Arrow-folding yields a 1/2-approximation for the STRING-FOLD problem in the HP model for two-dimensional triangular lattices for large values of $s_1$.*

## 2.4 Improved Arrow-folding: a 6/11-approximation

Even when folding sequence $S$ optimally, it can happen that some internal node H has less than 4 bonds, i.e its topological neighbor is an internal node P or no internal node at all. We call such pair of topological neighbors a *conflict*. Let $c_f$ be the number of conflicts in conformation $f$. Since each conflict excludes a bond, the number of bonds formed by conformation $f$ is

$$C(S, f) = (4s_1 - c_f)/2.$$

Conflicts in conformations are tightly related to *breakpoints* in the folded sequence $S$. A breakpoint is a position in $S$ at which a P is followed by a H or a H is followed by a P. Fig.10 shows all possibilities for two triangles adjacent to a breakpoint. Not all breakpoints in $S$ force conflicts in conformations, *weak* breakpoints HP-HP (or PH-PH) can be folded without creating conflicts, *Strong* breakpoints PP-HP (or HH-PH) in any conformation always create at least one conflict and *very strong* breakpoints PP-HH (or HH-PP) create two conflicts in any conformation.

Breakpoints P-HP..., ...PH-P (P-HH..., ...HH-P) at the beginning and at the and of $S$ are strong (very strong). Let $b_S$ be the number of strong breakpoints plus twice the number of very strong breakpoints. The same conflict can be forced by at most two different non-weak breakpoints, therefore $c_f \geq b_S/2$. Thus we have improved upper bound on $C(S, f)$.

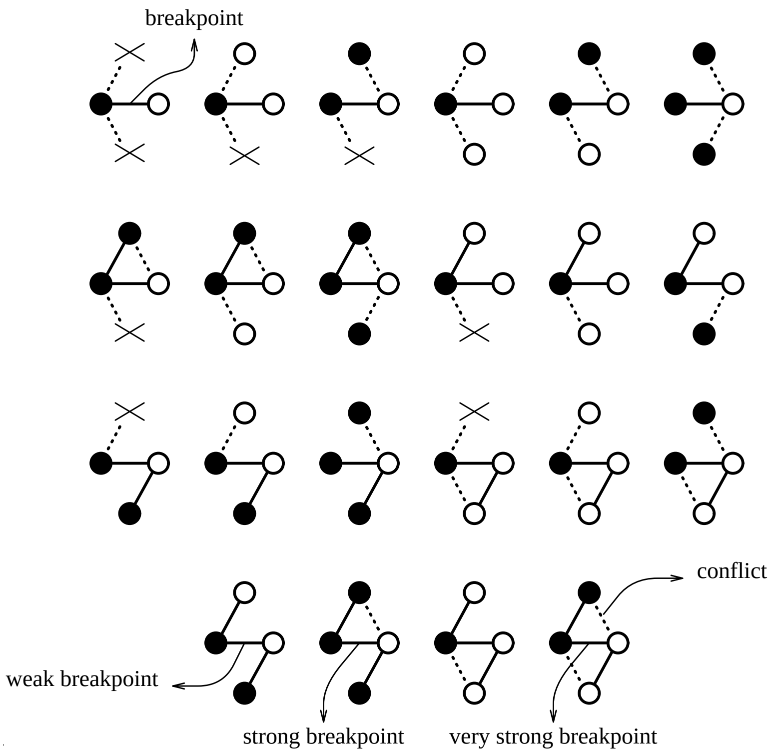


Figure 10: Conformation with breakpoints and conflicts; Solid lines represent the folding pathway; Dashed lines represent conflicts; Crosses represent absence of a node of $S$.

**Lemma 5** *For every conformation $f$ of binary sequence $S$ in the two dimensional triangular lattice we have*

$$C(S,f) \leq 2s_1 - b_S/4.$$

Note that each long strip(or long separator) contributes 2 to the value of $b_S$ (the first and the last strip of P's contribute only 1).

For certain parts of the sequence, a better analysis of arrow-folding leads to an approximation ratio better than 1/2. For other parts of the sequences we need to modify the "regular" arrow-folding to achieve a better approximation.

Suppose $S$ is arbitrarily divided into $k$ segments $S_1, \ldots, S_k$, $S = S_1 \cdots S_k$. Let $C_1, \ldots, C_k$ be the corresponding division of bonds ($C_i$ is the number of bonds contributed exclusively by the H nodes in $S_i$), $C_1 + \cdots + C_k = C$. We denote the number of H's in $S_i$ by $t_i$ and the number of breakpoints in $S_i$ exclusively due to $S_i$ by $b_i$. Let $r_i = \frac{C_i}{2t_i - b_i/4}$ be the ratio achieved by conformation $f$ for sequence $S_i$. From the fact that $(a+c)/(b+d) \geq \min\{a/b, c/d\}$ we have

$$\frac{C(S,f)}{2s_1 - b_S/4} = \frac{C_1 + \cdots + C_k}{(2t_1 - b_1/4) + \cdots + (2t_k - b_k/4)} \geq \min\{r_1, r_2 \ldots, r_k\}.$$

In the following we demonstrate a partitioning scheme for the input sequence and show that in the worst case we achieve 6/11 approximation. We refer to the conformation thus obtained as *improved-arrow-folding.* We assume that the input binary sequence starts with a P. We scan the sequence left to right, cut the sequence at certain place and analyze the cut prefix. In any particular iteration we examine at most 4 strips and 4 separators ahead to decide on the position of the next cut. Recall that any `H` node in the sequence contributes one bond in the arrow-folding. Any exclusive bond contributed by a `H` node in addition to the "regular" is an *extra bond*. For a certain prefix let $\delta$ denote the total number of extra bonds contributed by the `H` nodes in the prefix. Also let $\beta$ denote the number of breakpoints contributed by the prefix. Let $s$ denote the number of `H` nodes in the prefix. So on a per `H` node basis the extra bonds contributed exclusively by the prefix is $\delta/s$ and the number of breakpoints is $\beta/s$. Hence, for this prefix, the ratio $r$ of the number of bonds achieved to the number of bonds achievable is $\frac{1+\delta/s}{2-\beta/(4\cdot s)}$.

Notice that in the arrow-folding, whenever we have a strip of size greater than 5, we achieve 1 extra bond and the number of extra bonds increases by 1 for every increase in the size of the strip by 3 (for $k-$strips, $k \equiv 1 \pmod 3$, $k \geq 7$, we need to slightly deviate from the normal arrow-folding to achieve the claimed extra bonds). A little inspection of the above expression reveals that an extra bond outweighs an increase of 3 in $s$. From this perspective, the worst case for the ratio is when the length of any strip is 5. That is when $s$ is maximum without giving us the advantage of extra bond. In the following analysis, often we will take the size of strip as 5 without favoring our case, supported by this argument.

Also notice that 2 consecutive long strips with a separator between them gives us an extra bond between the last front of the first strip and first front of the second strip.

The approach to the improvement is the following. We study certain prefix and propose an alternative conformation for the prefix such that the segment before it and the segment after it are oblivious of this alternative conformation. As far as the neighboring segments are considered it is an arrow-folding, *i.e.* the alternative conformation can assume a normal arrow-folding before it and always ends like a normal arrow-folding. For this we need to ensure that the alternative conformation is contained inside the pattern space (Fig. 7) imposed by the normal arrow-folding. We also need to ensure that the first `H` node of any segment starts at current level `P` (so that the `H` nodes from last segment can count on gaining a bond via this node). Notice that a previous long strip may end at level `P` or at level $1^+$ (or $1^-$). So to start a new segment we must take into account both of these cases for the previous segment whenever it makes a difference. We exercise the liberty to change the level for a segment with respect to the same level for the previous segment. In other words we may shift the levels up or down for consecutive segments to gain extra bonds. It will become clear from the case analysis in the following.

Let the prefix of $S$ containing 4 strips and 4 separators be $\mathtt{P}^A\mathtt{H}^B\mathtt{P}^C\mathtt{H}^D\mathtt{P}^E\mathtt{H}^F\mathtt{P}^G\mathtt{H}^H$. We will decide on the position to cut depending on the values of $A$ through $H$. In the following we do a case analysis on different values of $A$ through $H$ and let $P$ be the cut prefix.

**Case 1:** $A > 1$ ; $P = \mathtt{P}^A\mathtt{H}^B$.
If $B = 1$ then $\delta = 0$, $\beta = 2$, $s = 1$, hence, $r = \frac{1}{2-2/4} = 2/3$.
Otherwise, $\delta = 0$, $\beta = 4$, $s = 5$, hence, $r = \frac{1}{2-1/5} = 5/9$.

**Case 2:** $A = 1$ and $B \geq 6$ ; $P = \mathtt{P}\mathtt{H}^B$.
The worst case for $r$ is when $B = 8$, $\delta = 1$, $\beta = 2$, hence $r = \frac{1+1/8}{2-1/16} = 18/31$.

**Case 3:** $A = 1$ and $1 < B \leq 3$; $P = \mathtt{P}\mathtt{H}^B$.
The worst case for $r$ is when $B = 3$, $\delta = 0$, $\beta = 2$, hence $r = \frac{1}{2-1/6} = 6/11$.

**Case 4:** $A = 1$ and ($B = 4$ or $B = 5$)

**Case 4a:** $D > 1$ ; $P = \mathtt{P}\mathtt{H}^B\mathtt{P}^C\mathtt{H}^D$.
We gain an extra bond due to consecutive long strips. Worst case for $r$ is when $B = D = 5$, $\delta = 1$, $\beta \geq 4$, $s = 10$, hence $r \geq \frac{1+1/10}{2-1/10} = 11/19$.

**Case 4b:** $C > 1$ and $D = 1$ ; $P = \mathtt{P}\mathtt{H}^B\mathtt{P}^C\mathtt{H}$.
Worst case for $r$ is when $B = 5$, $\delta = 0$, $\beta = 4$, $s = 6$, hence $r = \frac{1}{2-1/6} = 6/11$.

**Case 4c:** $C = 1$ and $D = 1$ ; $P = \texttt{PH}^4\texttt{PH}$ or $P = \texttt{PH}^5\texttt{PH}$.
From Fig. 11(a)(b), it's evident that we gain two extra bonds when $B = 5$ and 1 extra bond when $B = 4$, hence the worse of the cases is when $B = 4$, $\delta = 1$, $\beta = 2$, $s = 5$, hence $r = \frac{1+1/5}{2-1/10} = 12/19$.
**Case 5:** $A = 1$ and $B = 1$
**Case 5a:** $C > 1$ and $D > 1$ ; $P = \texttt{PHP}^C\texttt{H}^D$.
$D = 5$ (worst case), $\delta = 0$, $\beta = 4$, $s = 6$, hence $r = \frac{1}{2-1/6} = 6/11$.
**Case 5b:** $C = 1$ and $D = 2$ ; $P = \texttt{PHPH}^2$.
$\delta = 0$, $\beta = 2$, $s = 3$, hence $r = \frac{1}{2-1/6} = 6/11$.
**Case 5c:** $C = 1$ and ($D = 3$ or $D = 4$ or $D = 5$) ; $P = \texttt{PHPH}^3$ or $P = \texttt{PHPH}^4$ or $P = \texttt{PHPH}^5$.
From Fig. 11(c)(d)(e), it's evident that we gain two extra bonds when $D = 5$ and 1 extra bond when $D = 4$ or $D = 3$, hence the worst of the cases is when $D = 4$, $\delta = 1$, $\beta = 2$, $s = 5$, hence $r = \frac{1+1/5}{2-1/10} = 12/19$.
**Case 5d:** $C = 1$ and $D > 5$ ; $P = \texttt{PHPH}^D$.
$D = 8$ (worst case), $\delta = 1$, $\beta = 2$, $s = 9$, hence $r = \frac{1+1/9}{2-1/18} = 4/7$.
**Case 5e:** $C > 1$ and $D = 1$ ; $P = \texttt{PHP}^C\texttt{H}$.
$\delta = 0$, $\beta = 2$, $s = 2$, hence $r = \frac{1}{2-1/4} = 4/7$.
**Case 6:** $A = B = C = D = 1$
**Case 6a:** $F > 1$ ; $P = \texttt{PHPHP}^E\texttt{H}^F$.
From Fig. 11(f), it's evident that we gain 1 extra bond when $F > 1$, hence worst of the cases is when $F = 5$, $\delta = 1$, $\beta \geq 2$, $s = 7$, hence $r \geq \frac{1+1/7}{2-1/14} = 16/27$.
**Case 6b:** $E > 1$ and $F = 1$ ; $P = \texttt{PHPHP}^E\texttt{H}$.
$\delta = 0$, $\beta = 2$, $s = 3$, hence $r = \frac{1}{2-1/6} = 6/11$.
*reminder:* $A = B = C = D = E = F = 1$
**Case 7:** $H > 1$ ; $P = \texttt{PHPHPHP}^G\texttt{H}^H$.
From Fig. 11(g), it's evident that we gain 1 extra bond when $H > 1$, hence worst of the cases is when $H = 5$, $\delta = 1$, $\beta \geq 2$, $s = 8$, hence $r \geq \frac{1+1/8}{2-1/16} = 18/31$.
**Case 8:** $H = 1$ ; $P = \texttt{PHPHPHP}^G\texttt{H}$.
From Fig. 11(h), it's evident that we gain 1 extra bond. $\delta = 1$, $\beta \geq 0$, $s = 4$, hence $r \geq \frac{1+1/4}{2} = 5/8$.

The worst ratio among all the cases is 6/11 which implies that

**Corollary 4** *Improved-arrow-folding yields a 6/11-approximation for the STRING-FOLD in the HP model on a two dimensional triangular lattice for large values of $s_1$.*

# 3 Rules for Three Dimensions

Analogous to the two dimensional case, the three dimensional triangular lattice mimics the configuration of equal sized solid spherical objects packed in three dimensional space. It can be described as a stack of two dimensional triangular lattices where every individual lattice is slightly offset with respect to the lattices above and below it (Figure 12). As in the two dimensional case, we assume that the sequence starts and ends with a P.

**Lemma 6** *For every conformation $f$ of $S$ on a three dimensional triangular lattice*

$$C(S, f) \leq 5 \cdot s_1.$$

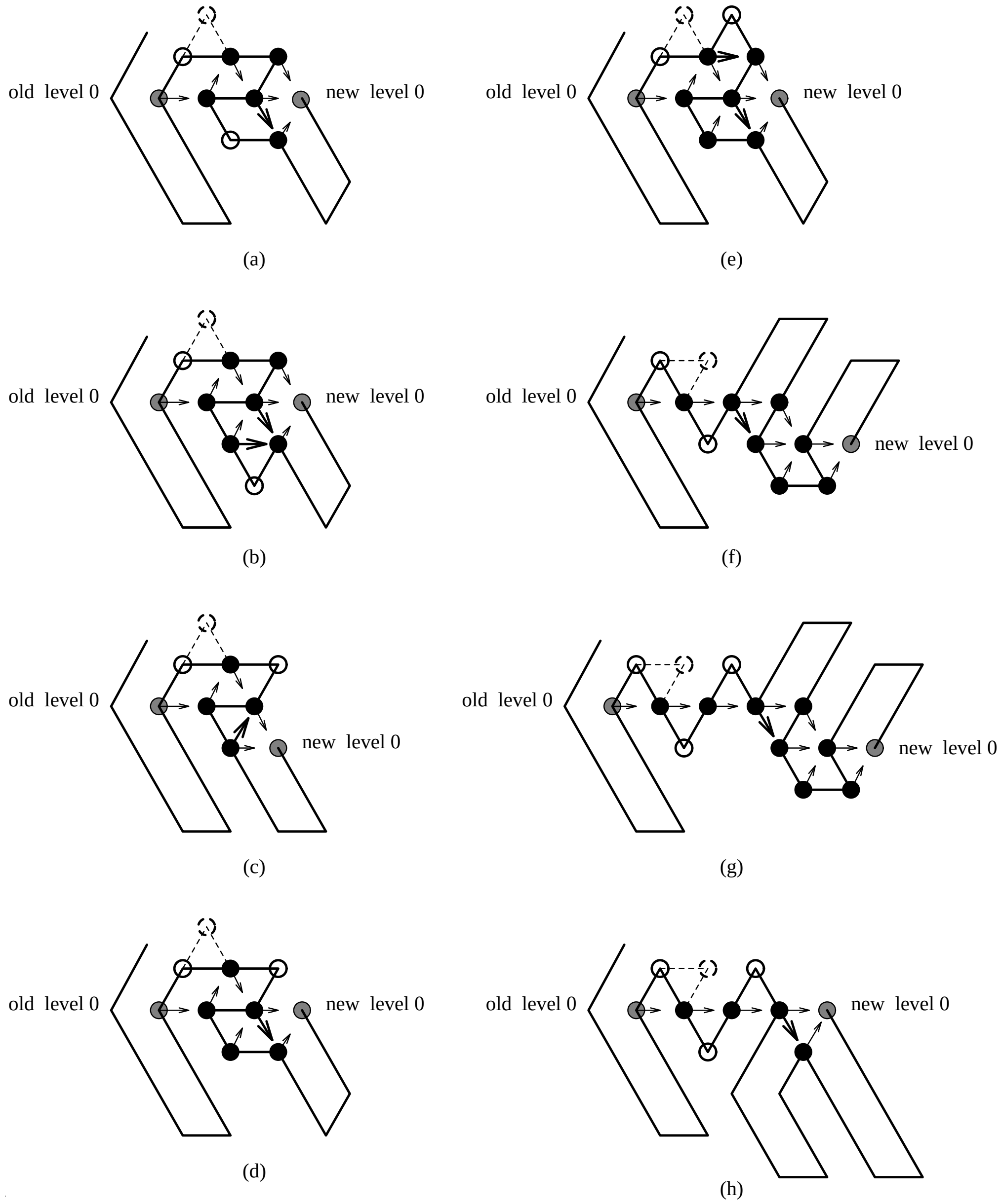


Figure 11: The gray nodes represent the H nodes from the previous or the next segment. Solid dark line represent the folding pathway. Dashed line shows a shift in the conformation when the previous strip ends at level H. Thin arrows represent the bonds charged to the source node. Thick arrow represents an extra bond. Long separators are not shown explicitly.
(a) case 4c, $P = \mathtt{PH}^4\mathtt{PH}$ (b) case 4c, $P = \mathtt{PH}^5\mathtt{PH}$ (c) case 5c, $P = \mathtt{PHPH}^3$ (d) case 5c, $P = \mathtt{PHPH}^4$ (e) case 5c, $P = \mathtt{PHPH}^5$ (f) case 6a, $P = \mathtt{PHPHP}^E\mathtt{H}^F$ (g) case 7, $P = \mathtt{PHPHPHP}^G\mathtt{H}^H$ (h) case 8, $P = \mathtt{PHPHPHP}^G\mathtt{H}$

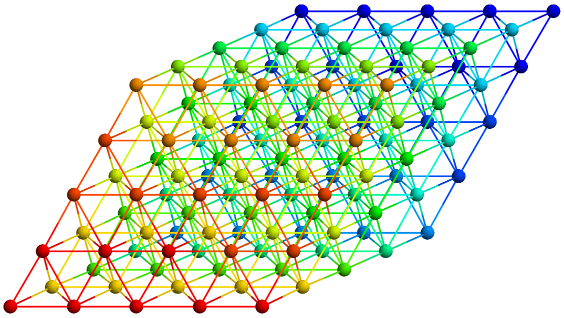

Figure 12: The Three-Dimensional Triangular Lattice.

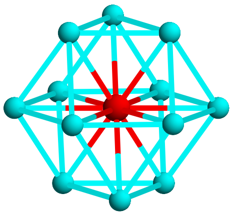

Figure 13: A single node and its twelve neighbors.

**Proof** Every node has 12 neighbors in this lattice, as shown in Figure 13. Since any internal residue can have at most 10 topological contacts in the three dimensional triangular lattice (out of 12 neighbors, two are connected along the sequence), clearly, $C(S, f) \leq 5 \cdot s_1$. ∎

### 3.1 Backbone-folding$^{3D}$: a 3/10-approximation

Since the three dimensional triangular lattice is a stack of two dimensional triangular lattices, an obvious extension of the backbone-folding for the three dimensional triangular lattice would look like a stack of backbone-foldings (*backbone-folding*$^{3D}$). We could fold the entire sequence into the two dimensional lattice, divide it into a few segments and stack up the conformations of individual segments on consecutive parallel planes. Instead of two parallel lines we now have two parallel backbone planes containing all `H`'s of the sequence.

**Lemma 7** *Let $f^b$ be the 3D backbone folding of sequence $S$ on the three dimensional triangular lattice. Ignoring the boundary conditions*

$$C(S, f^b) \geq \frac{3}{2} \cdot s_1.$$

**Proof** The two parallel backbone planes consist of the parallel backbones of the segments. Each `H` on the $i$-th backbone creates at least 3 bonds, two with $(i-1)$-th and $(i+1)$-th backbones in the same plane and one with the $i$-th backbone in the opposite plane. ∎

Lemmas 6 and 7 together imply

**Corollary 5** *Backbone-folding$^{3D}$ yields a 3/10-approximation for the STRING-FOLD problem in the HP model on a three dimensional triangular lattice for large values of $s_1$.*

### 3.2 Improved Backbone-folding$^{3D}$: a 2/5-approximation

Instead of stacking up a number of 2D backbone foldings, we can construct a triangular "tube" of three backbones such that one strand runs anti-parallel to the other two strands.

**Lemma 8** *Let $f^b$ denote improved backbone-folding of $S$ with tube $T$ on the three dimensional triangular lattice. Ignoring the boundary conditions*

$$C(S, f^b) \geq 2 \cdot s_1.$$

**Proof** In $f^b$, if $x, y, z$ are neighbors and form a triangle along a cross section of $T$, then all the three pairs $xy, yz, xz$ form bonds. Ignoring the boundary condition where we are folding the backbone to form $T$, each `H` on a triangular lattice $i$ forms 2 bonds with backbone strands in both lattice $(i-1)$ and lattice $(i+1)$. Hence, each `H` forms at least 4 bonds. ∎

**Corollary 6** *Improved backbone-folding$^{3D}$ yields a 2/5-approximation for the STRING-FOLD problem in the HP model on a three dimensional triangular lattice for large values of $s_1$.*

### 3.3 Arrow-folding$^{3D}$: a 7/15-approximation

We extend two dimensional arrow-folding into three dimensions in the same manner as we did with backbone folding. We divide the entire sequence into a few segments and fold each segment according to the arrow-folding rule for the two dimensional lattice and stack up the conformations of individual segments on consecutive parallel planes.

**Lemma 9** *Let $f^b$ be the 3D arrow folding of sequence $S$ on the three dimensional triangular lattice. Ignoring the boundary conditions*

$$C(S, f^b) \geq \frac{7}{3} \cdot s_1.$$

**Proof** All `H`'s of sequence $S$ are on one of level planes -1,0 or +1 and there are only `H`'s on the plane level 0. Each `H` on level 0 forms three exclusive bonds, one with a `H` in the same segment and two with `H`'s at level 0 of the next segment. Each `H` on levels $\pm 1$ forms two exclusive bonds, one with a `H` at level 0 of the same segment and one with a `H` at level 0 of the next segment. Each front of arrow folding can contain one, two or three `H`'s. In the first case we have three exclusive bonds per one `H`, in the second case we have five exclusive bonds per two `H`'s and in the third case we have seven exclusive bonds per three `H`'s. ∎

Lemmas 6 and 9 together imply

**Corollary 7** *Arrow-folding$^{3D}$ yields a 7/15-approximation for the STRING-FOLD problem in the HP model on a three dimensional triangular lattice for large values of $s_1$.*

### 3.4 Star-folding: a 16/30-approximation

In the following we describe *star-folding* and show that for this conformation, we get at least $\frac{8}{3}s_1$ bonds, leading to a 16/30-approximation. We place all `H`'s in a single, tightly-packed, hydrophobic core. The core is composed of six `H`'s at each level and contains as many levels as is necessary to accommodate all `H`'s in the sequence. The pattern for one level, containing six `H`'s in the core and possibly ladders of `P`'s is shown in Figure 14. Then we place these individual stars on consecutive planes in a slightly offset manner as shown in Figure 15 by connecting one end of a star with an end of the star at the level above and the other end to an end of the star at the level below.

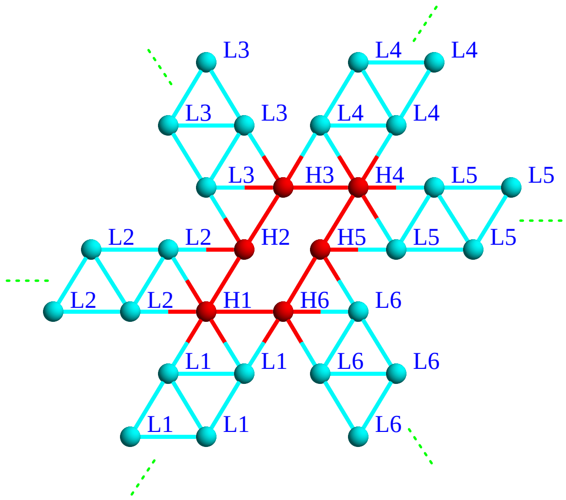


Figure 14: Individual plane in star-folding.

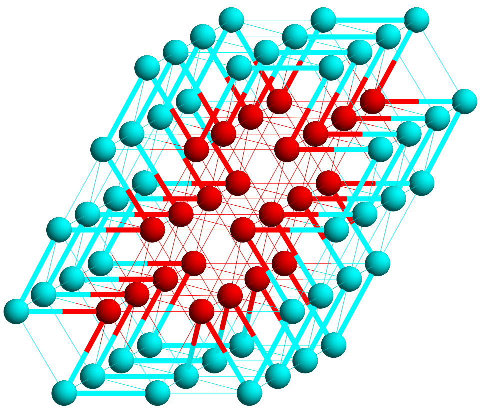

Figure 15: Four levels of the 16/30 factor layout.

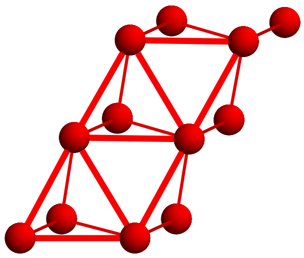

Figure 16: Hydrophobic contacts within one plane and to the next plane.

**Lemma 10** *Let $f^s$ denote star-folding of $S$ on the three dimensional triangular lattice. Ignoring the boundary conditions*

$$C(S, f^s) \geq \frac{16}{6} \cdot s_1.$$

**Proof** To compute the number of bonds, we will look at the six H's on a certain plane and count the number of bonds exclusive to these six H's. As shown in Figure 16, each group of six H's in the constructed core has 9 bonds within the group (thick lines) and 13 bonds to the next group (thin lines). At most 6 of these bonds could be covalent bonds. Thus, each group of six H's contributes at least $9 + 13 - 6 = 16$ exclusive bonds and $C(S, f^s) \geq \frac{16}{6} s_1$. ■

Lemmas 6 and 10 together imply

**Corollary 8** *Star-folding yields a 16/30-approximation algorithm for the STRING-FOLD problem in the HP model on a three dimensional triangular lattice for large values of $s_1$.*

### 3.5 Combined (Backbone-Star) folding: a 44/75-approximation

To improve approximation ratio we derive better upper bound. Weak, strong and very strong breakpoints are defined in the same way as in section 2.4. Let $b_S$ be twice the number of weak breakpoints plus three times the number of strong breakpoints plus four times the number of very strong breakpoints. Now we can bound the maximal number of bonds in the same manner as we did in Lemma 5.

**Lemma 11** *For every conformation $f$ of binary sequence $S$ in the three dimensional triangular lattice we have*

$$C(S, f) \leq 5s_1 - b_S/8.$$

**Proof** Let $c_f$ be the number of conflicts in conformation $f$. Since each conflict excludes a bond, the number of bonds formed by conformation $f$ is $C(S, f) = (10s_1 - c_f)/2$. Each edge of the lattice is adjacent to four triangles (and two squares). Two of the triangles always create conflicts and the situation in the other two triangles is as described by Fig. 10. This means that every very strong breakpoint forces four conflicts, every strong breakpoint forces three conflicts and every weak breakpoint forces two conflicts. Moreover, one conflict can be created by at most four different breakpoints, therefore $c_f \geq b_S/4$ and $C(S, f) \leq 5s_1 - b_S/8$. ■

Closer look on the backbone-folding reveals that a long separator of P's followed by a single H is the most unfavorable case. On the other side, single H's form extra bonds in the star-folding. We can decide the type of folding depending on the number of single H's. This yields 44/75 approximation.

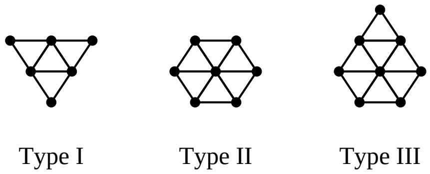


Figure 17: Cluster types.

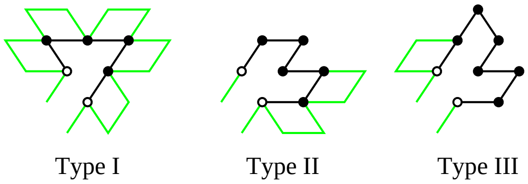


Figure 18: Ladders and pivots for the clusters.

Suppose the number of H's in the sequence $S$ is $s_1$ and $t_1$ of them are single H's preceded by long separator of P's. Each such long separator contributes at least 6 to the $b_S$ and for every conformation of $S$ we have $C(S, f) \leq 5s_1 - 6t_1/8$.

First we shall investigate the backbone folding for $S$. When both backbone planes are fully covered by H's, each H forms seven bonds (4 in its plane and 3 between the planes) and $t_1$ single H's form even nine bonds. Backbone folding puts $t_1$ P's on the backbone, each of them cancels at most nine bonds. So the backbone folding for $S$ forms $(7(s_1 - t_1) + 9t_1 - 9t_1)/2 = (7s_1 - 7t_1)/2$ bonds. In the star-folding $f^s$ each single H forms two extra bonds, thus we have $C(S, f^s) = \frac{16}{6}s_1 + t_1$. For $t_1 < \frac{5}{27}s_1$ the backbone folding is more favorable, while for $t_1 > \frac{5}{27}s_1$ the star folding produces better approximation ratio. The worst case is when $t_1 = \frac{5}{27}s_1$ and we get

**Corollary 9** *Combined (backbone-star) folding yields a 44/75-approximation algorithm for the STRING-FOLD problem in the HP model on a three dimensional triangular lattice for large values of* $s_1$*.*

### 3.6 Improved Star folding: a 3/5 approximation.

Note that the algorithm of Section 3.4 constructs a fold such that the position of the $i$-th hydrophobic residue in the sequence depends *only* on $i$. Specifically, the positions of the hydrophobic residues do not depend at all on the polar residues in the sequence. The algorithm presented in this section instead takes into account the polar residues in determining the fold, and is able to achieve an improved approximation factor. We first describe the algorithm and then analyze its performance.

As before, we construct a leveled hydrophobic core, but in the new algorithm the levels may differ as a function of the local makeup of hydrophobic versus polar residues. The three types of levels, or *clusters*, used are shown in Figure 17.

In Figure 18, we show the locations of the polar ladders for each of the types as well as the entry and exit point for each type. The ladders are shown in light lines while the dark lines represent the sequence of H's in the cluster. Note that of the H's shown might not be covalently linked. The entry and exit H's (*pivot points*) are shown with empty circles. As is evident from Figure 18, any subsequence starting and ending with an H and containing 6 H's can always be folded into a Type

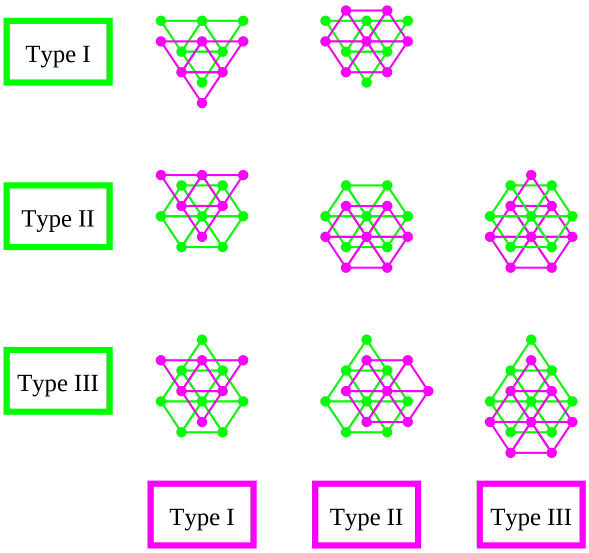

Figure 19: Arrangements of clusters from one level to the next.

I cluster. Consider a subsequence containing 7 `H`'s, starting and ending with an `H`, and containing either one or two groups of consecutive `P`'s between the `H`'s. Such a subsequence must contain 3 consecutive `H`'s, and can always be folded into a Type II cluster. Finally, a subsequence containing 8 `H`'s and starting with 7 consecutive `H`-residues can always be folded into a Type III cluster. The following algorithm contains the rules for folding an arbitrary protein sequence using the cluster types described above.

1. Let $p$ be the number of intervening `P` groups in the subsequence that contains the next 7 `H`'s.
2. If $p \geq 3$ fold the next 6 `H`'s into a Type I cluster[4].
3. If $p = 1$ or $p = 2$ fold the next 7 `H`'s into a Type II cluster.
4. If $p = 0$ (and therefore the next 7 `H`'s are consecutive):
   - If the previous cluster is of Type I, fold the next 7 `H`-residues into a Type II cluster.
   - If the previous cluster is of Type II or III, fold the next 8 `H`'s into a Type III cluster.
5. Continue again with Step 1.

Figure 19 displays all of the possible arrangements of one cluster type following another. The light shapes denote clusters at the previous level and the dark shapes represent clusters at the current level. A cluster of one level connects to a cluster of the next level using the pivot points shown in Figure 18. More specifically, a cluster is folded clockwise or counterclockwise depending on whether it started in the upper, or lower pivot point of Figure 18, respectively. Thus it reaches the other pivot point, which is then used to connect to the next level. The next level starts at the lower pivot point if the previous level ended at the lower pivot point, and similarly for the upper pivot point. A ladder, defined by the two pivot points where the two clusters connect, and the direction shown in Figure 18, is used to accommodate an arbitrary number of `P`'s between the boundary `H`'s of the two clusters.

Let *inner* contacts of a cluster be the non-covalent bonds among the `H`'s of the cluster, plus possibly the non-covalent bond between the exit `H` of this cluster and the entry `H` of the next cluster.

[4]The 7th `H` is left for the next level.

Let *outer* contacts of a cluster be the contacts that the cluster achieves with the previous cluster, minus one (between the entry and exit `H`-residue). A Type I cluster contains 9 contacts within the cluster plus one contact between the pivot `H`'s of this and the next level. Since there are at least 3 intervening polar groups in the next 7 `H`'s subsequence starting with the entry `H` of this cluster, at most 3 of these 10 contacts may be covalent. Thus, a Type I cluster contributes at least 7 inner contacts. Similarly a Type II cluster achieves at least $7(= 12 + 1 - 6)$ inner contacts except when placed on top of a Type I cluster, when it can achieve as few as $6(= 12 + 1 - 7)$ inner contacts.[5] A Type III cluster achieves at least $7(= 14+1-8)$ inner contacts. From the diagrams in Figure 19 we can enumerate the outer contacts each additional cluster is guaranteed to contribute. For example, placing a Type I cluster on top of a Type II cluster achieves 13 contacts, of which one may be a covalent bond between the two endpoint `H`-residues. Thus, it achieves at least 12 outer contacts. The enumeration of inner and outer contacts is summarized in Table 1.

| | Next Cluster Type | | |
|---|---|---|---|
| | Type I | Type II | Type III |
| New Residues | 6 | 7 | 8 |
| On Type I | **18** = 11+7 | **20** = 14+6 | — |
| On Type II | **19** = 12+7 | **21** = 14+7 | **23** = 16+7 |
| On Type III | **20** = 13+7 | **22** = 15+7 | **24** = 17+7 |

Table 1: Number of contacts (outer+inner) achieved by each combination of types.

Notice that any cluster on top of a cluster of the same type achieves at least 3 contacts per `H`. Say that we *ascend* a level whenever we move to a cluster of higher type (*i.e.* Type I to Type II, or Type II to Type III). Similarly say we *descend* one or two levels, whenever we move to a cluster of lower type. Notice in Table 1, that each time we ascend a level, we are one contact short of achieving 3 contacts per residue, whereas each time we descend one or two levels, we achieve 1 or 2 contacts more, respectively, than 3 contacts per residue. For the fold of one sequence, the total number of levels we ascend, has to equal the total number of levels we descend ($\pm$ at most 2). Thus this algorithms achieves asymptotically 3 contacts per residue.

**Corollary 10** *Improved Star-folding yields a 3/5-approximation algorithm for the STRING-FOLD problem in the HP model on a three dimensional triangular lattice for large values of* $s_1$.

# 4 Generalized Hydrophobicity

The standard HP model makes the simplifying assumption that all hydrophobic residues have the same energy contribution to the hydrophobic collapse. Yet it is well known that various residues vary in their *hydrophobicity* (see for example Kyte and Doolittle [14] or Engelman, Steitz and Goldman [8].) We therefore propose an extension of the HP model which accounts for the varying hydrophobicities [5].[6]

[5] This exception is due to the rule in which we put a Type II cluster on top of a Type I cluster even if the next 7 `H`-residues contain no intervening polar groups.

[6] Note that although surface area of the actual residue may contribute to the strength of hydrophobicity attributed to a residue, we do not model the actual size differences in the spatial layout of the protein. Furthermore, we do not address the fact that polar residues also have diversity when compared to one another as well as when compared to water. These extension not considered here would still further improve the modeling of the protein folding problem.

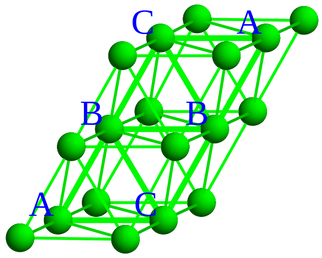


Figure 20: Contacts for positions A, B and C.

In the new model, we allow each of the 20 amino acids to have a value from the set $\{0, 1, 2, \ldots\}$. Zero represents polar residues and non-zero values represent proportional levels of hydrophobicity. We then consider the value of a contact to be the amount of hydrophobicity buried from polar residues and solvent. Thus, a contact between any residue and solvent shields no hydrophobicity and is given a value of 0. A contact between a hydrophobic residue (with value from $\{1, 2, \ldots.\}$) and a polar residue (value 0) also shields no hydrophobicity and is given a value of 0. Finally, a contact between two hydrophobic residues (each with value from $\{1, 2, \ldots.\}$) shields both and is given value equal to their sum, the combined amount of hydrophobicity that is shielded.

If the ratio between the largest and smallest hydrophobic values assigned to the 20 amino acids is $\rho$, then blindly using an algorithm for the binary HP model could in the worst case result in a factor of $\rho$ loss in the approximation factor. In the case of the algorithm of Section 3.4, as $\rho$ grows arbitrarily the 16/30 approximation factor only falls by a constant to 2/5. These losses occur since some positions in a fold are involved in fewer contacts than others. If the strongly hydrophobic residues are placed in these positions, then the approximation suffers. Therefore, we must adapt our algorithms to adjust the construction based on the strength of the hydrophobic residues in the protein. Below, we adapt the algorithm of Section 3.4 for the generalized HP model and analyzed its performance. In addition, other folding algorithms such as the one in Section 3.6 should also be examined in this respect and one would hope that their performance could also be kept comparable to their performance in the binary HP model.

In adapting the algorithm of Section 3.4 we are able to achieve the same 16/30 asymptotic approximation that was possible in the binary HP model. The new algorithm will once again place all hydrophobic (non-zero) residues in the core using the layout shown in Figure 15. In this layout, the number of contacts that a hydrophobic residue participates in depends on which of three different types of positions within the core it occupies (See Figure 20). If we number the hydrophobic residues starting at 0, then the number of hydrophobic contacts (2 of which may be covalent) in which the $r$-th hydrophobic residue participates is

$$\begin{array}{llll} 6 & \Rightarrow & \text{if } r \equiv 0 \bmod 3 & \text{(Position A)} \\ 9 & \Rightarrow & \text{if } r \equiv 1 \bmod 3 & \text{(Position B)} \\ 7 & \Rightarrow & \text{if } r \equiv 2 \bmod 3 & \text{(Position C)} \end{array}$$

out of a possible 12 contacts. Thus, there are three equivalence classes of residues, depending on their hydrophobic position in the primary sequence modulo 3. For $i \in \{0, 1, 2\}$, let $H_i$ be the set of residues at position $r$ such that $r \equiv i \bmod 3$, let $W_i$ be the sum of the hydrophobic values of all of the residues in set $H_i$ and let indices $i_1, i_2, i_3$ be such that $W_{i_1} \leq W_{i_2} \leq W_{i_3}$. If we define $W = W_{i_1} + W_{i_2} + W_{i_3}$, then the optimal score for a protein is at most $(12 - 2) * W$. Note that a single scan of the residues is sufficient to determine the values $W_0$, $W_1$ and $W_2$.

By placing the first hydrophobic residue at a different starting position in the construction, we effectively replace $r$ in the above equations with $r+1$ or $r+2$. Furthermore, by wrapping the core in the opposite direction, we effectively replace $r$ by $-r$. Therefore, we adapt the construction to place the residues of $H_{i_3}$ in position B, the residues of $H_{i_2}$ in position C, and the residues of $H_{i_1}$ in position A. If we let $x = W_{i_1}$, $y = W_{i_2} - W_{i_1}$ and $z = W_{i_3} - W_{i_2}$, then we achieve:

$$\begin{aligned}
\text{Score} &= (9-2) * W_{i_3} + (7-2) * W_{i_2} + (6-2) * W_{i_1} \\
&= 7 * (x+y+z) + 5 * (x+y) + 4 * (x) \\
&= 16x + 12y + 7z \\
&\geq 16x + \frac{32}{3}y + \frac{16}{3}z \qquad (1) \\
&= \frac{16}{3} \cdot (3x + 2y + z) \\
&= \frac{16}{3} \cdot W
\end{aligned}$$

The approximation factor is therefore at least 16/30. Inequality (1) is equality (yielding an approximation factor lower bound of 16/30) when $y = z = 0$, *i.e.* $W_0 = W_1 = W_2$. When the weights of the three classes differ, the approximation factor bound increases and asymptotically approaches 7/10 as all of the weight becomes concentrated in one of the three classes.

In some proteins, the weight of residues in one class (*i.e.* $W_0$) may be largest in the first half of the protein sequence, while the weight of residues in another class (*i.e.* $W_2$) may be largest in the second half of the protein sequence. In such cases, we would prefer one wrapping of the core in the first half and another wrapping in the second half. In order to change the wrapping in the middle of the sequence, there is a loss in the score due to the break in the core. Yet, in some cases this loss would be more than made up for by placing the heavier weighted residues at the B position in both halves of the sequence. Furthermore, this change in wrapping can be done as many times as is productive. Using dynamic programming, one could construct the best score possible by this strategy in time $O(nb)$ where $n$ is the number of residues in the protein and $b$ is the number of breaks permitted.

Note that these arguments for improving the approximation do not yield better *worst case* asymptotic approximations since here the worst case is when all classes have equal weight. But, since real proteins undoubtably have diverse hydrophobic makeup, the ability of an algorithm to leverage this diversity when present would be preferable. As this diversity is not guaranteed to exist in a worst-case analysis, it would also be of interest to determine if there are properties in biological protein sequences which HP folding algorithm *could* make use of to improve their performance guarantees, possibly using an average case analysis.

## 5 Acknowledgement

Thanks to Bonnie Berger, Gunnar Hoest, Jonathan King, Jon Kleinberg, and Lior Pachter for discussion on protein folding and lattices. Images generated with the help of RasMol Molecular Renderer, Version 2.5.1 (Roger Sayle, October 1994). An abstract [2] of this paper apeared at RECOMB'97 and an extended abstract [1] appeared at SODA'97.

## References

[1] R. Agarwala, S. Batzoglou, V. Dančík, S. Decatur, S. Hannenhalli, M. Farach, , and S. Skiena. Local rules for protein folding on a triangular lattice and generalized hydrophobicity in HP

model. In *8th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'97)*, pages 390–399, 1997.

[2] R. Agarwala, S. Batzoglou, V. Dančík, S. Decatur, S. Hannenhalli, M. Farach, S. Muthukrishnan, and S. Skiena. Local rules for protein folding on a triangular lattice and generalized hydrophobicity in HP model. In *First Annual International Conference on Computational Molecular Biology (RECOMB'97)*, pages 1–2, 1997.

[3] Erich Bornberg-Bauer. Chain growth algorithms for hp-type lattice proteins. In *First Annual International Conference on Computational Molecular Biology (RECOMB'97)*, pages 47–55, 1997.

[4] J.H. Conway and N.J.A. Sloane. *Sphere Packings, Lattices, and Groups.* Springer-Verlag, New York, second edition edition, 1993.

[5] S. Decatur. Protein folding in the generalized hydrophobic-polar model on the triangular lattice. Technical Memo MIT-LCS-TM-559, Massachusetts Institute of Technology, May 1996.

[6] K. A. Dill, Sarina Bromberg, Kaizi Yue, Klaus M. Fiebig, David B. Yee, Paul D. Thomas, and hue Sun Chan. Principles of protein folding: A perspective from simple exact models. *Prot. Sci*, 4:561–602, 1995.

[7] K.A. Dill. Theory for the folding and stability of globular-proteins. *Biochemistry*, 24(6):1501–1509, 1985.

[8] D. M. Engelman, T. A. Steitz, and A. Goldman. Identifying nonpolar transbilayer helices in amino acid sequences of membrane proteins. *Annu. Rev. Biophys. Chem.*, 1986.

[9] W. Hart and S. Istrail. Fast protein folding in the hydrophobic-hydrophilic model within three-eighths of optimal. *Journal of Computational Biology*, 3(1):53–96, 1996.

[10] W. Hart and S. Istrail. Invariant patterns in crystal lattices: Implications for protein folding algorithm. In *Combinatorial Pattern Matching, Proc. 7th Annual Symposium (CPM'96)*, 1996. 288-303.

[11] William Hart. On the computational complexity of sequence design problems. In *First Annual International Conference on Computational Molecular Biology (RECOMB'97)*, pages 128–136, 1997.

[12] William Hart and Sorin Istrail. Lattice and off-lattice side chain models of protein folding: Linear time structure prediction better than 86 In *First Annual International Conference on Computational Molecular Biology (RECOMB'97)*, pages 137–146, 1997.

[13] J. Kleinberg. Some computational problems in protein structure prediction. MIT EECS Area Exam (Cmte: R. Davis, T. Lozano-Perez, D.K. Gifford), December 1995.

[14] J. Kyte and R. F. Doolittle. A simple method for displaying the hydropathic character of protein. *J. Mol. Biol.*, 1982.

[15] R. Lathrop. The protein folding threading problem with sequence amino acid interaction preferences is NP-complete. *Protein Engineering*, 7(9):1059–1068, 1994.

[16] J. T. Ngo and J. Marks. Computational complexity of a problem in molecular structure prediction. *Protein Engineering*, 5(4):313–321, 1992.

[17] M. Paterson and T. Przytycka. On the complexity of string folding. In *ICALP'96*, volume 1099 of *LNCS*, pages 658–669, 1996.

[18] R. Unger and J. Moult. Finding the lowest free energy conformation of a protein is a NP-hard problem: Proof and implications. *Bulletin of Mathematical Biology*, 55(6):1183–1198, 1993.