Invariant Patterns in Crystal Lattices: Implications for Protein Folding Algorithms (Extended Abstract)*†

William E. Hart‡ Sorin Istrail§

December 11, 1995

Abstract

Crystal lattices are infinite periodic graphs that occur naturally in a variety of geometries and which are of fundamental importance in polymer science. Discrete models of protein folding use crystal lattices to define the space of protein conformations. Because various crystal lattices provide discretizations of the same physical phenomenon, it is reasonable to expect that there will exist "invariants" across lattices that define fundamental properties of protein folding process; an invariant defines a property that transcends particular lattice formulations. This paper identifies two classes of invariants, defined in terms of sublattices that are related to the design of algorithms for the structure prediction problem. The first class of invariants is used to define a master approximation algorithm for which provable performance guarantees exist. This algorithm can be applied to generalizations of the hydrophobic-hydrophilic model that have lattices other than the cubic lattice, including most of the crystal lattices commonly used in protein folding lattice models. The second class of invariants applies to a related lattice model. Using these invariants, we show that for this model the structure prediction problem is intractable across a variety of three-dimensional lattices. It turns out that these two classes of invariants are respectively sublattices of the two- and three-dimensional square lattice. As the square lattices are the standard lattices used in empirical protein folding studies, our results provide a rigorous confirmation of the ability of these lattices to provide insight into biological phenomenon. Our results are the first in the literature that identify algorithmic paradigms for the protein structure prediction problem which transcend particular lattice formulations.

1 Introduction

Crystal lattice models are vehicles for reasoning about the protein folding phenomenon through analogy. Crystal lattices are infinite periodic graphs that are generated by translations of a "unit cell" that fill a two or three-dimensional space. In polymer science many important results have been obtained through the use of lattice models [4, 8]. In the context of protein folding, lattices provide a natural discretization of the space of protein conformations. The sequence of amino acids

*An appendix containing proofs and definitions is included as a separate document for review at the discretion of the program committee.

†This work was supported by the Applied Mathematical Sciences program, U.S. Department of Energy, Office of Energy Research, and was performed at Sandia National Laboratories, operated for the U.S. Department of Energy under contract No. DE-AC04-94AL85000.

‡wehart@cs.sandia.gov; Sandia National Labs, Massively Parallel Computing Research Laboratory, P. O. Box 5800 , Albuquerque, NM 87185-1110, http://www.cs.sandia.gov/~wehart/main.html

§scistra@cs.sandia.gov, Sandia National Labs, Massively Parallel Computing Research Laboratory, P. O. Box 5800 , Albuquerque, NM 87185-1110, http://www.cs.sandia.gov/~scistra/main.html

DISCLAIMER

that defines a protein can be viewed as a path with the amino acids. The path is labeled with amino acids on vertices. A conformation of a protein is a self-avoiding embedding of this path into a lattice, where each vertex of the path is mapped to a vertex of the lattice and edges of the path are mapped to edges of the lattice. With every conformation we can associate an energy value using rules defined by the model, which take into account the neighborhood relationship of the amino acids. The central focus of this paper is the design of algorithms that construct a conformation of minimal or near-minimal energy for a given sequence.

Of particular interest here is the design of algorithms that can be applied to a variety of lattice models. Results that transcend particular lattice frameworks are of significant interest because they can say something about the general biological problem with a higher degree of confidence. In fact, it is reasonable to expect that there will exist invariants across lattices that fundamentally relate to the protein folding problem, because lattice models provide discretizations of the same physical phenomenon. However, the identification of such invariants has not been previously addressed.

This paper identifies invariants across lattice models that can be described in terms of sublattices. These invariants give the ability to address the following question. Given an energy formula for crystal lattices, does there exist an algorithm that takes a sequence and a lattice and produces a conformation with minimal energy? If such an algorithm exists, it may provide valuable insight into the protein folding process because it captures essential features of protein folding.

We address this question in two ways. First we design performance guaranteed approximation algorithms for protein folding in the hydrophobic-hydrophilic model. This model categorizes amino acids as hydrophobic (nonpolar) or hydrophilic (polar), and the energy of a conformation is equal to the number of hydrophobic-hydrophobic contacts. The invariant we use to design a "master" approximation algorithm for crystal lattices employs special sublattices which we call latticoids. Latticoids impose a structure in which a skeleton of hydrophobic contacts can be constructed, thereby leading to folding algorithms whose performance can be analyzed. In the particular case of the square two-dimensional lattice, the latticoid describes the structure used in the approximation algorithms described by Hart and Istrail [7].

We prove that our master approximation algorithm has performance guarantees for a class of lattices that includes most of the lattices commonly used in simple exact protein folding models, e.g. two- and three-dimensional square lattice [4, 6, 11], the diamond (carbon) lattice [12] and the face-centered-cubic lattice [2]. Furthermore, this class encompasses a large number of other lattices studied in crystallography. Our main theorems state that latticoids of the two-dimensional square lattice can be embedded into all of these lattices, and therefore, every lattice in the class is approximable in linear time.

Second, we prove that lattice models related to those considered by Unger and Moult [13] are NP-complete. The lattice model considered by Unger and Moult uses a distance-related energy formula between an unbounded number of amino acid types. Our results extend their NP-completeness argument to any three-dimensional lattice into which a certain type of sublattice can be embedded. All of the three-dimensional lattices mentioned above fall into this class.

2 Preliminaries

2.1 Lattice Models for Protein Folding

A lattice model is composed of four components: (1) an alphabet of types of amino acids that the model considers, (2) the set of protein instances represented as sequences from this alphabet, (3) an energy table specifying the contact energies between different types of amino acids and (4) a crystal lattice that provides a discretization of the conformation space. The protein folding models

analyzed in this paper are hydrophilic-hydrophobic models (HP models). The alphabet used in an HP model is $A = \{0, 1\}$, and the set of protein instances is the set of binary sequences $\sigma = \{0, 1\}^+$. Each sequence $s \in \sigma$ is the (hypothesized) hydrophobic-hydrophilic pattern of a protein sequence, where 1 represents a hydrophobic amino acid, and 0 represents a hydrophilic amino acid. We will refer to s as a protein instance. The energy table $\mathcal{E}$ is indexed by the alphabet symbols, $\mathcal{E} = (e(a, b))_{a,b \in A}$. For HP models, $e(a, b) = -1$ if $a = b = 1$, and $e(a, b) = 0$ otherwise.

HP models abstract the hydrophobic interaction process in protein folding by reducing a protein to a heteropolymer that represents a predetermined pattern of hydrophobicity in the protein. The HP model on square and cubic lattices was proposed by Dill [3]. This is one of the most studied lattice models for protein folding, and despite its simplicity, the model is powerful enough to capture a variety of properties of actual proteins [4].

We consider HP models on a large class of crystal lattices, including the square lattice. Crystal lattices are infinite periodic graphs that are generated by translations of a "unit cell" that fill a two- or three-dimensional space. A unit cell contains a finite graph that is connected to neighboring unit cells. Examples of crystal lattices are shown in Figure 1.

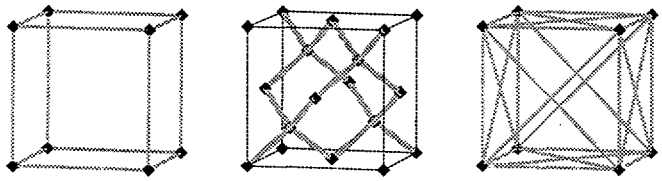


Figure 1: Examples of crystal lattices: (a) cubic, (b) diamond, and (c) cubic with planar diagonals.

One can interpret a protein sequence $s = s_1 ... s_m$ as an m-vertex node-labeled path, where for $1 \leq i \leq m$, node i is labeled with s_i. The path has $m - 1$ edges that are called *bonds*. A *conformation* C of a protein sequence s in a lattice L is a path in the lattice in which the protein sequence is embedded, i.e., the protein vertices are mapped one-to-one to lattice points, and protein bonds are mapped to the corresponding lattice edges. The *energy* of a conformation of the protein sequence s in L is defined as the sum of the energies of the "contact edges". A contact edge is a lattice edge that is not a protein bond (in the embedding) but has both endpoints labeled. In HP models, contact edges with 1s at their endpoints have weight -1 while all other contact edges have weight 0.

The *native* conformation of a protein is the conformation that has biological function. According to the Thermodynamic Hypothesis the native conformation of a protein is the conformation with the minimum energy among the set of all conformations. Consequently, given a lattice model, $PF = (A, \sigma, \mathcal{E}, L)$, the protein folding structure prediction problem is to find a native conformation of s in L. It is unknown whether this problem is NP-complete for HP models, but a few related models have been shown to be NP-complete [13, 9, 10]. Furthermore, Hart and Istrail [7] have demonstrated that performance guaranteed approximation algorithms exist for HP models on square and cubic lattices.

Let $\mathcal{Z}_L(s)$ be the energy of the conformation generated for protein instance s on lattice L with by algorithm $\mathcal{Z}_L$, and let $OPT_L(s)$ be the energy of the optimal conformation of s on L. A standard performance guarantee used for approximation algorithms is the asymptotic performance ratio $R^\infty(\mathcal{Z}_L)$ [5]. If $R^\infty(\mathcal{Z}_L) = \tau$, then as $\mathcal{Z}_L$ is applied to larger protein instances, the value of

solutions generated by $\mathcal{Z}_L$ approaches a factor of τ of the optimum. Here, "large" protein instances have low conformational energy at their native state, which may be independent of their length. Since $\mathcal{Z}_L(s) \leq 0$ and $OPT_L(s) \leq 0$, both of these ratios are scaled between 0 and 1 such that a ratio closer to 1 indicates better performance.

2.2 Protein Sequence Structure

This section summarizes key definitions concerning the structure of protein instances from Hart and Istrail [7]. Let $s = s_1, \ldots, s_m$ be a protein instance, $s_i \in \{0, 1\}$. Let $l(s)$ equal the length of the sequence s. Let $M_{max}(s)$ equal the length of the longest sequence of zeros in s, and let $M_{min}(s)$ equal the length of the shortest sequence of zeros in s. Finally, let $E(s)$ equal the number of adjacent elements in the sequence, s_j and s_{j+1} for which $s_j = 1$ and $s_{j+1} = 1$.

An instance s can be decomposed into a sequence of *blocks*. A block b_i has the form $b_i = 1$ or $b_i = 1Z_{i_1}1 \ldots Z_{i_h}1$, where the Z_{i_j} are odd-length sequences of 0's and $h \geq 1$. A *block separator* z_i is a sequence of 0's that separates two consecutive blocks, where $l(z_i) \geq 0$ and $l(z_i)$ is even for $i = 1, \ldots, h-1$. Thus s is decomposed into $z_0 b_1 z_1 \ldots b_h z_h$. Since $l(z_i) \geq 0$, this decomposition treats consecutive 1's as a sequence of blocks separated by zero-length block separators. Let $N(b_i)$ equal the number 1's in b_i. Thus the sequence

$$0\underbrace{10101}_{b_1}\underbrace{1}_{b_2}\underbrace{1}_{b_3}\underbrace{10101}_{b_4}0000\underbrace{1010101}_{b_5}$$

can be represented as $l(z) = (1, 0, 0, 0, 4, 0)$ and $N(b) = (3, 1, 1, 3, 4)$.

Note that two 1's can be endpoints of a contact edge only if there is an even number of elements between them [7]. It follows from our definition of blocks that two 1's within a block cannot be in contact. Further, any pair of 1's take from blocks b_k and b_j may be in contact only when $|k - j|$ is odd.

Since 1's from a block can only be in contact of 1's from every other block, it is useful to divide blocks into two categories: x-blocks and y-blocks. For example, let $x_i = b_{2i}$ and let $y_i = b_{2i-1}$. This makes it clear that 1's from an x-block can only be in contact with 1's from an y-block. Let B_x and B_y be the number of x-blocks and y-blocks respectively. Further, let $X = X(s) = \sum_{i=1}^{B_x} N(x_i)$ and $Y = Y(s) = \sum_{i=1}^{B_y} N(y_i)$.

Let $T_x(s)$ equal the number of endpoints of s that are 1's in x-blocks, and let $T_y(s)$ equal the number of endpoints of s that are 1's in y-blocks. We assume that the division into x- and y-blocks is such that $X \leq Y$ and if $X = Y$ then $T_x(s) \geq T_y(s)$. For example, the sequence

$$0\underbrace{10101}_{y_0}\underbrace{1}_{x_0}\underbrace{1}_{y_1}\underbrace{10101}_{x_1}0000\underbrace{1010101}_{y_2}$$

can be represented as $z_0 y_0 z_1 x_0 z_2 y_1 z_3 x_1 z_4 y_2 z_5$, where $l(z) = (1, 0, 0, 0, 4, 0)$, $N(x) = (1, 3)$, and $N(y) = (3, 1, 4)$.

A *superblock* B_i is comprised of sequences of blocks as follows: $B_i = b_{i_1} z_{i_1} \ldots z_{i_{h-1}} b_{i_h}$. Let $N_x(B_i)$ equal the sum of $N(b_j)$, where b_j are x-blocks in B_i. Let $N_y(B_i)$ equal the sum of $N(b_j)$, where b_j are y-blocks in B_i. Finally, let $N(B_i) = N_x(B_i) + N_y(B_i)$.

3 A Paradigm for a Master Approximation Algorithm

This section describes the a paradigm for a master approximation algorithm. This paradigm captures two aspects of the protein folding algorithms described by Hart and Istrail [7]: (1) the

selection of a folding point that balances hydrophobicity and (2) the skeleton of contact edges that forms the hydrophobic core. We use the following to define the algorithm.

Definition 1 Given a path p in a lattice L from a to b, let $d_p(a,b)$ be the length of p. A path p from a to b is *polynomial extensible* if there exist paths p_k for every $k \in \mathbf{Z}^{>0}$ such that $d_{p_k}(a,b) = d_p(a,b) + 2k$ and there exists an polynomial time algorithm that given p and k constructs p_k. The collection of the paths of an polynomial extensible path p is called the *extension* of p in L.

Given polynomial extensible paths p from a to b and q from c to d, we say that p and q are *extensibly disjoint* if their extensions are vertex disjoint.

A *latticoid*, $\hat{L}$, of L is an infinite graph that contains an infinite sequence of contact edges (a_i, b_i) with the following properties: (1) There is an polynomial extensible path p_i^a from a_i to a_{i+1} and polynomial extensible path p_i^b from b_i to b_{i+1}, (2) There is a constant $\kappa > 0$ such that for every i and j, $d_{p_i^a}(a_i, a_{i+1}) = d_{p_j^b}(b_j, b_{j+1}) = 2\kappa$, and (3) The set of paths $\{p_i^a, p_i^b \mid i = 1, \ldots\}$ are mutually extensibly disjoint. The *dilation* of the latticoid is $\Delta_{\hat{L}} = \kappa$.

Figure 2 illustrates two latticoids of the two-dimensional square lattice, $\mathbf{L}_0$.

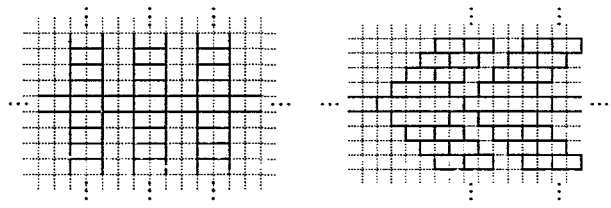


Figure 2: Two possible latticoids used by $\mathcal{A}_{\mathbf{L}_0}$: (a) $\hat{\mathbf{L}}_0^i$, $i = 2$, and (b) $\hat{\mathbf{L}}_0^H$. Dark lines indicate edges that are used for some protein instance. Dotted lines indicate the remaining edges in $\mathbf{L}_0$. The contact edges are the vertical edges of the centered bolded horizontal row.

The *master approximation algorithm* takes a latticoid $\hat{L}$. The master approximation algorithm selects a single folding point (turning point) that divides a protein instance into a y-superblock B' and an x-superblock B''. The folding point is selected using "Subroutine 1" from Hart and Istrail [7]. Subroutine 1 selects a folding point that balances the hydrophobicity between the x-blocks and y-blocks on each half of the folding point. The following lemma describes the key property of the folding point that is selected.

Lemma 1 ([7], Lemma 1) The folding point selected by Subroutine 1 partitions a protein instance s into two superblocks B' and B'' such that either

$$N_y(B') \geq \lceil (Y+1)/2 \rceil \quad \text{and} \quad N_x(B'') \geq \lceil X/2 \rceil, \quad \text{or} \quad N_y(B') \geq \lceil Y/2 \rceil \quad \text{and} \quad N_x(B'') \geq \lceil (X+1)/2 \rceil.$$

After selecting the folding point, the conformation of the two superblocks is dictated by the latticoid $\hat{L}$. The latticoid specifies the placement of the contact edges between the superblocks, as well as the conformation of the loops within each superblock. This generalizes the notion of "normal form" that was used to describe the approximation algorithms in Hart and Istrail [7].

Decomposition into x- and y-blocks requires a single pass through the protein instance. Subroutine 1 requires a single pass through the sequence of blocks, which is no longer than the length of

the protein instance. The construction of the final conformation requires polynomial time to create the paths for the zero-loops. Thus the computation required by Algorithm $\mathcal{A}_{\hat{L}}$ is polynomial.

Let $\mathcal{A}_{\hat{L}}(s)$ represent the energy of the final conformation generated by Algorithm $\mathcal{A}_{\hat{L}}$. The performance of Algorithm $\mathcal{A}_{\hat{L}}$ can be bounded as follows.

Lemma 2

$$\mathcal{A}_{\hat{L}}(s) \leq -\left\lceil \frac{X}{2\Delta_{\hat{L}}} \right\rceil + 1.$$

Let $\delta(L)$ be the maximum degree of all vertices in L. Since L is a crystal lattice generated by a unit cell, $\delta(L)$ is finite. It follows that $OPT_L(s) \leq -(\delta(L) - 2)X(s) - 2$. Proposition 1 presents the asymptotic ratio for Algorithm $\mathcal{A}_{\hat{L}}$.

Proposition 1 $R^{\infty}(\mathcal{A}_{\hat{L}}) \geq 1/(2\Delta_{\hat{L}}(\delta(L) - 2))$.

To illustrate the application of the master approximation algorithm, consider its application to the diamond lattice, which has previously been used in lattice models for protein folding [12]. Figure 3 shows the embedding of a "dilated" square lattice into a plane of unit cells for the diamond lattice. Dashed and solid lines between vertices in each unit cell indicate the edges of the diamond lattice that are used to embed a square lattice for which one dimension is dilated to length two. Edges not used for this embedding are omitted. The solid lines illustrate a conformation of a protein on this lattice that the master approximation algorithm would generate. Note that this conformation can be embedded in the latticoid $\hat{\mathrm{L}}_0^2$ (see Figure 3). Now $\delta(L) = 4$ for the diamond lattice L. It follows from Proposition 1 that $R^{\infty}(\mathcal{A}_L) = 1/8$.

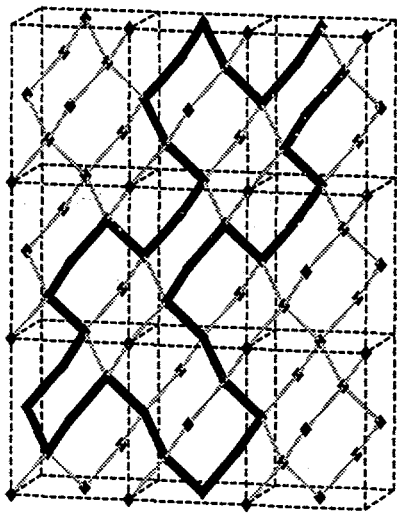


Figure 3: Illustration of the embedding of the $\hat{\mathrm{L}}_0^2$ latticoid into a diamond lattice. Dashed and solid lines show the embeding. The solid lines illustrate a conformation generated by the master approximation algorithm.

4 A Complexity Theory for Protein Folding on Crystal Lattices

In this section we describe a framework for analyzing the design of efficient approximation algorithms with provable performance guarantees. The unifying theme is polynomial approximability asymptotic within a constant of optimal. This theory defines polynomial embedding reductions from one lattice to another, and relates the approximability on the first lattice to the approximability on the second. Further, this theory includes a notion of *completeness*, which defines the "hardest" members in the class.

Definitions A lattice L is *polynomial kernel-approximable* if there is a polynomial algorithm $\mathcal{A}$ and constants $\alpha_L, \beta_L \in \mathbf{Z}^{>0}$ such that for all protein instances s, $A(s) = -\alpha_L X(s) + \beta_L$. A class of lattices $\mathcal{L}$ is *polynomial kernel-approximable* if for every $L \in \mathcal{L}$, L is polynomial kernel-approximable. Let **PKAL** be the class of polynomial kernel-approximable lattices. A lattice L is *polynomial approximable* if there is a polynomial algorithm $\mathcal{A}$ and a constant $\tau_L \in \mathbf{R}^{>0}$ such $R^{\infty}(\mathcal{A}) \geq \tau_L$. A class of lattices $\mathcal{L}$ is *polynomial approximable* if for every $L \in \mathcal{L}$, L is polynomial approximable. Let **PAL** be the class of polynomial approximable lattices. A *sublattice* $\hat{L}$ of L is a subgraph of L that is obtained by removing edges and vertices from L. A particular sublattice is the latticoid.

While we aspire to a framework for general approximability, our current framework applies to kernel-approximability.

Lemma 3 If L is polynomial kernel-approximable, then there exists an polynomial algorithm $\mathcal{A}$ and constant C_L such that $R^{\infty}(\mathcal{A}) \geq C_L$.

Corollary 1 If $\hat{L}$ is a sublattice of a lattice L and $\hat{L}$ is polynomial kernel-approximable, then L is polynomial kernel-approximable.

Definition 2 A *core* of a lattice L is a set of sublattices $D(L) = \{\hat{L}^1, \hat{L}^2, \ldots\}$, where $D(L)$ is finite or countably infinite.

Folding algorithms in a lattice L_1 can be transferred to folding algorithms in another lattice L_2, a folding "reduction", if the sublattice used in L_1 by the approximation algorithm can be embedded in L_2. This reduction can be polynomial in the sense that each unit cell is given by a finite description, and the symmetries in the crystal lattice are with respect to the neighboring cells (and thus also of finite description). This notion of reduction is formalized in the following definition.

Definition 3 A *polynomial embedding reduction* of L_1 to L_2 via core $D(L_1)$ is a polynomial time function $\psi : \hat{L}_1 \rightarrow \hat{L}_2$ such that: (1) $\hat{L}_1$ is a sublattice in $D(L_1)$, (2) $\hat{L}_2$ is a sublattice of L_2, and (3) $\psi(\hat{L}_1)$ is lattice isomorphic to $\hat{L}_2$ (i.e. graph isomorphic). We say that $\hat{L}_1$ is embedded into $\hat{L}_2$. If there is a polynomial embedding reduction from L_1 to L_2 via core $D(L_1)$, we write $L_1 \propto_{D(L_1)} L_2$.

Definition 4 A lattice L with core $D(L)$ is *polynomial core kernel-approximable* if $D(L) \subseteq$ **PKAL**.

Lemma 4 If a lattice L_1 with core $D(L_1)$ is polynomial core kernel-approximable and $L_1 \propto_{D(L_1)} L_2$, then L_2 is polynomial kernel-approximable.

The central concept of this theory is the notion of completness defined as follows.

Definition 5 Let $\mathcal{L}$ be a class of lattices. A lattice L is called *$\mathcal{L}$-complete via core $D(L)$* if (1) $L \in \mathcal{L}$ and (2) $\forall L' \in \mathcal{L}$, $L \propto_{D(L)} L'$.

Similar to the theory of NP-completness, if any member of the complete set is core-approximable then we can design polynomial approximation algorithms for all lattices in the class.

Theorem 1 Let L be a lattice with core $D(L)$. If L is $\mathcal{L}$-complete and polynomial core kernel-approximable then $\mathcal{L} \subseteq \mathbf{PKAL} \subset \mathbf{PAL}$.

5 Approximable Lattices

In this section we describe a class of lattices $\mathcal{L}$ that is polynomial kernel-approximable. $\mathcal{L}$ is a broad class of lattices includes many of the lattices previously used in lattice models for protein folding. Further, it includes many other important crystallographic lattices. The proof that $\mathcal{L}$ is polynomial kernal-approximable confirms that performance guaranteed approximability is not an artifact of the square and cubic lattices. Further, this lattice independence results suggests that the algorithmic mechanisms used to generate these approximate conformations may play a role in biological systems.

To prove kernel-approximability for this class, we describe a core of $\mathbf{L_0}$ for which $\mathbf{L_0}$ is $\mathcal{L}$ complete. Let $D(\mathbf{L_0})$ be a core of $\mathbf{L_0}$ composed of the union of all possible latticoids of $\mathbf{L_0}$. This core clearly includes the latticoids describe in Figure 2. Let $\mathcal{L}$ be the set of lattices for which $\mathbf{L_0} \propto_{D(\mathbf{L_0})} L$, for all $L \in \mathcal{L}$. The following lemma follows direction from the definition of $\mathcal{L}$.

Lemma 5 The lattice $\mathbf{L_0}$ is $\mathcal{L}$-complete.

The following theorem follows directly from the Lemma 5 and Proposition 1.

Theorem 2 $\mathcal{L} \subseteq \mathbf{PKAL}$.

We will now describe a subset of the lattices in $\mathcal{L}$. Our description of $\mathcal{L}$ is split into three parts.

1. Embeddings of the planar square lattice:
   $\mathcal{L}$ clearly contains all lattices into which $\mathbf{L_0}$ can be embedded. These include: (a) Bravias lattices, which contain all points R of the form $R = n_1 a_1 + n_2 a_2 + n_3 a_3$, where n_i are integers and a_i are linearly independent vectors in $\mathbf{R}^n$ [1], (b) the planar triangular lattice, which tiles the plane with equilateral triangles, (c) the hexagonal close packed crystal structure, and (d) the flourite structure.

2. Embeddings of $\hat{\mathbf{L}}_0^1$:
   The diamond lattice is an example of an embedding of $\hat{\mathbf{L}}_0^2$.

3. Embeddings of the planar hexagonal lattice:
   The latticoid $\hat{\mathbf{L}}_0^H$ can be reduced to the hexagonal lattice, which shows that the hexagonal lattice is in $\mathcal{L}$. It can be shown that if the hexagonal lattice can be completely embedded into other lattices, then they are in $\mathcal{L}$. This is significant since there are a large number of crystal lattices for which the hexagonal lattice can be embedded. The catalog of lattices in Wells [14] contains many three-dimensional lattices into which the hexagonal lattice can be embedded.

Although $\mathcal{L} \subseteq \mathbf{PKAL}$, it is unclear whether this relation is strict. $\mathcal{L}$ certainly spans a broad class of crystal lattices. Furthermore, we believe that it contains many biologically relevant crystal lattices. For example, it contains most of the lattices previously used in previous protein folding lattice models [2, 4, 6, 11, 12].

6 Hardness Results

In this section, we generalize the NP-hardness proof by Unger and Moult [13] to show that it is applicable for a variety of lattices. Let L be a three-dimensional crystal lattice and let $\mathbf{Z}$ be the set of integers. Suppose that S is a protein instance represented by a sequence of amino acids $s_1, \ldots, s_n$. For a conformation of S, suppose the coordinate of s_i is (x_i, y_i, z_i). Then $d_{ij}^x = |x_i - x_j|$, $d_{ij}^y = |y_i - y_j|$, and $d_{ij}^z = |z_i - z_j|$. We can define a lattice-specific protein folding problem as follows.

L-**PF**

Instance: A sequence $S = (s_1, \ldots, s_n)$, $s_i \in A \subset \mathbf{Z}$; a positive function $g : [0, n]^3 \to \mathbf{R}^+$; a matrix $C \in \mathbf{Z}^{m \times m}$, $m = |A|$; $B \in \mathbf{Z}$.

Question: Is there an embedding of S in L such that $\sum_{i=1}^{n} \sum_{j \neq i} C_{s_i,s_j} g(d_{ij}^x, d_{ij}^y, d_{ij}^x) \leq B$?

Unger and Moult [13] demonstrate that L-PF is NP-complete for the lattice L defined by the unit cell in Figure 1c. The NP-completeness of L-PF problems can be generalized to a variety of other lattices by noting a key property of the conformations used to construct their proof. The reduction from OLA used by Unger and Moult requires that certain residues be placed along a line parallel to the x-axis in the optimal conformation. Further, it must be possible to construct vertex-independent paths between these residues for any permutation of their ordering along this line.

A second class of invarant patterns in lattices occurs in the context of this type of NP-completeness argument. We can abstract the type of structure needed for the reduction as a sub-lattice. Using the ideas similar to the previous invariants, we can then construct NP-completeness reductions for a variety of crystal lattices. Figure 4 illustrates the concept of this class of invariants on two lattices: the cubic and diamond lattice.

Theorem 3 Let L be a Bravais, diamond, flourite or hexagonal close packed lattice. Then L-PF is NP-complete.

Acknowledgements

Our thanks to Ken Dill for suggesting the extension of our previous results to other lattice models and for discussions that inspired this work. We also thank Martin Karplus for his interest in our work and for his insight into the importance of performance guaranteed approximation algorithms for protein folding.

References

[1] N. W. Ashcroft and N. D. Mermin. *Solid State Physics.* Holt, Rinehart and Winston, 1976.

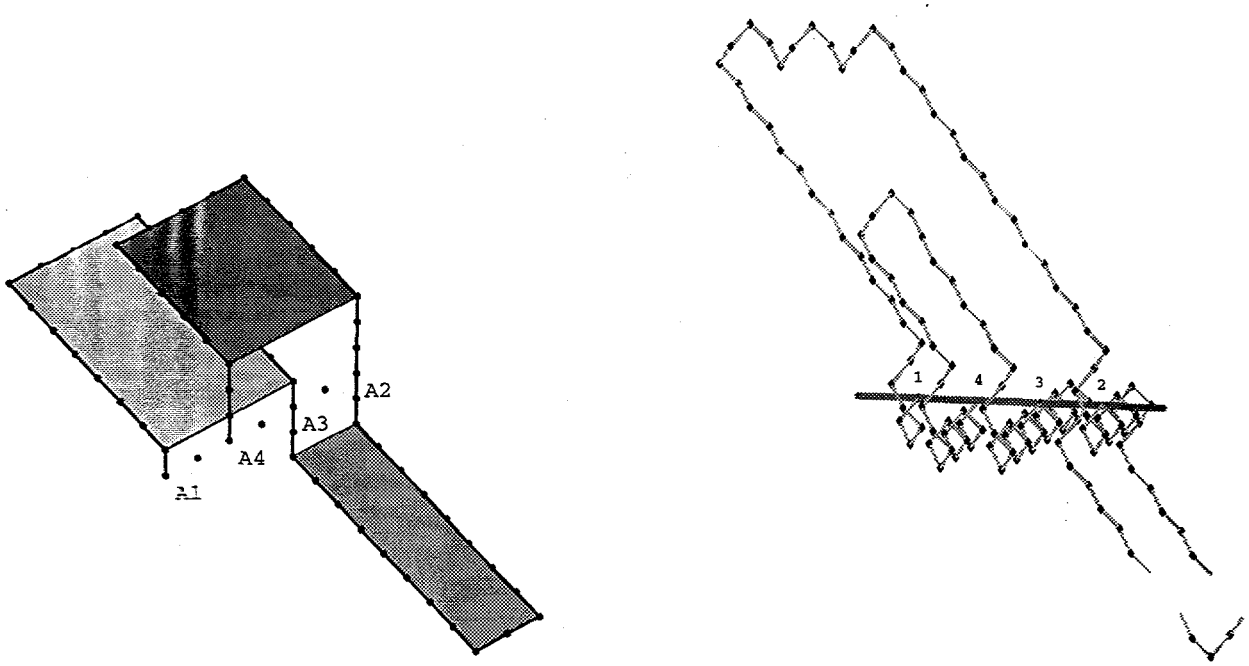
[2] D. G. Covell and R. L. Jernigan. *Biochemistry*, 29:3287, 1990.

[3] K. A. Dill. *Biochemistry*, 24:1501, 1985.

[4] K. A. Dill, S. Bromberg, K. Yue, K. M. Fiebig, D. P. Yee, P. D. Thomas, and H. S. Chan. Principles of protein folding: A perspective from simple exact models. *Prot. Sci.*, 4:561–602, 1995.

[5] M. R. Garey and D. S. Johnson. *Computers and Intractability - A guide to the theory of NP-completeness.* W.H. Freeman and Co., 1979.

[6] A. M. Gutin and E. I. Shakhnovich. Ground state of random copolymers and the discrete random energy model. *J. Chem. Phys.*, 98:8174–8177, 1993.



(a) (b)

Figure 4: Illustration of the conformational invariants neede for (a) the cubic lattice and (b) the diamond crystal lattice. The numbers indicate the amino acids that are placed collinear parallel to the x-axis. The break in the chain in (b) shortens the diagonally oriented loop.

[7] W. Hart and S. Istrail. Fast protein folding in the hydrophobic-hydrophilic model within three-eighths of optimal. To appear in *Journal of Computational Biology*, Spring 1996. Extended abstract in *Proc. of 27th Annual ACM Symposium on Theory of Computation*, May 1995.

[8] M. Karplus and E. Shakhnovich. *Protein folding: Theoretical studies of thermodynamics and dynamics*, chapter 4, pages 127–195. W. H. Freeman and Company, 1993.

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

[10] M. Paterson, March 1995. Personal communication.

[11] E. I. Shakhnovich and A. M. Gutin. Engineering of stable and fast-folding sequences of model proteins. *Proc. Natl. Acad. Sci.*, 90:7195–7199, 1993.

[12] A. Sikorski and J. Skolnick. Dynamice Monte Carlo simulations of globular protein folding/unfolding pathways. II. α-helical motifs. *J. Molecular Biology*, 212:819–836, July 1990.

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

[14] A. F. Wells. *Three-dimensional nets and polyhedra.* American Crystallographic Association, 1979.

Appendix to "Invariant Patterns in Crystal Lattices: Implications for Protein Folding Algorithms (Extended Abstract)"

William E. Hart and Sorin Istrail
{wehart,scistra}@cs.sandia.gov

1. Proofs of Lemmas, Propositions and Theorems
   (a) PROOF OF LEMMA 2
   (b) PROOF OF PROPOSITION 1
   (c) PROOF OF LEMMA 3
   (d) PROOF OF COROLLARY 1
   (e) PROOF OF LEMMA 4
   (f) PROOF OF THEOREM 1
   (g) PROOF OF THEOREM 3
2. Table of Symbols and Definitions
3. The Diamond Lattice

A Proofs of Lemmas, Propositions and Theorem

A.1 PROOF OF LEMMA 2

Proof. From Lemma 1 we know that $N_y(B') \geq \lceil X(s)/2 \rceil$ and $N_x(B'') \geq \lceil X(s)/2 \rceil$. Consequently the number of contact edges is at least

$$\left\lceil \frac{\lceil X(s)/2 \rceil}{\Delta_{\hat{L}}} \right\rceil \geq \left\lceil \frac{X(s)}{2\Delta_{\hat{L}}} \right\rceil .$$

If step 2b is applied, then one of the possible contact edges is eliminated. Consequently, the final energy is at least $-\left\lceil \frac{X(s)}{2\Delta_{\hat{L}}} \right\rceil + 1$. ∎

A.2 PROOF OF PROPOSITION 1

Proof. We know from Lemma 2 that

$$\mathcal{A}_{\hat{L}}(s) \leq -\left\lceil \frac{X(s)}{2\Delta_{\hat{L}}} \right\rceil + 1.$$

Now $OPT_L(s) \leq -(\delta(L)-2)X(s) - 2$, so

$$R_{\mathcal{A}_{\hat{L}}}(s) = \frac{\mathcal{A}_{\hat{L}}(s)}{OPT_{\hat{L}}(s)} \geq \frac{-\left\lceil \frac{X(s)}{2\Delta_{\hat{L}}} \right\rceil + 1}{-(\delta(L)-2)X(s)-2} \geq \frac{-\frac{X(s)}{2\Delta_{\hat{L}}} + 1}{-(\delta(L)-2)X(s)-2} = \frac{X(s) - 2\Delta_{\hat{L}}}{2\Delta_{\hat{L}}(\delta(L)-2)X(s) + 4\Delta_{\hat{L}}}. \tag{1}$$

For $s \in S_N^L$, $-(\delta(L)-2)X(s) - 2 \leq N$, so $X(s) \geq -(N-2)/(\delta(L)-2)$. Since Equation (1) is monotonically increasing for $X(s) \geq 0$, we have

$$R_{\mathcal{A}_{\hat{L}}}(s) \geq \frac{-(N-2)/(\delta(L)-2) - 2\Delta_{\hat{L}}}{-2\Delta_{\hat{L}}(N-2)/(\delta(L)-2) + 4\Delta_{\hat{L}}} = \frac{N - 2 - 4\Delta_{\hat{L}} + 2\Delta_{\hat{L}}\delta(L)}{2\Delta_{\hat{L}}(\delta(L)-2)N}$$

so

$$R^N(\mathcal{A}_{\hat{L}}) \geq \frac{N - 2 - 4\Delta_{\hat{L}} + 2\Delta_{\hat{L}}\delta(L)}{2\Delta_{\hat{L}}(\delta(L)-2)N}$$

and

$$R^\infty(\mathcal{A}_{\hat{L}}) = \sup\{r \mid R^N(\mathcal{A}_{\hat{L}}) \geq r, N \in \mathbf{Z}\} \geq \lim_{N\to\infty} \frac{N - 2 - 4\Delta_{\hat{L}} + 2\Delta_{\hat{L}}\delta(L)}{2\Delta_{\hat{L}}(\delta(L)-2)N} = 1/(2\Delta_{\hat{L}}(\delta(L)-2)).$$

∎

A.3 PROOF OF LEMMA 3

Proof. From the definition of polynomial kernel-approximability, we know that there exists constants α_L and β_L such that for all protein instances s, $\mathcal{Z}(s) \leq -\alpha_L X(s) + \beta_L$. Now $OPT_L(s) \leq -(\delta(L)-2)X(s) - 2$. Thus

$$R_{\mathcal{Z}}(s) = \frac{\mathcal{Z}(s)}{OPT_L(s)} \geq \frac{-\alpha_L X(s) + \beta_L}{OPT_L(s)} \geq \frac{-\alpha_L X(s) + \beta_L}{-(\delta(L)-2)X(s) - 2}.$$

For $s \in S_N$, $-(\delta(L)-2)X(s)-2 \leq OPT(s) \leq N$, so $X(s) \geq -(N+2)/(\delta(L)-2)$. Since $\frac{-\alpha_L X(s)+\beta_L}{-(\delta(L)-2)X(s)-2}$ is monotonically increasing for $X(s) \geq 0$, we have

$$R_{\mathcal{Z}}(s) \geq \frac{\alpha_L(N+2)/(\delta(L)-2)+\beta_L}{(\delta(L)-2)(N+2)/(\delta(L)-2)+2} = \frac{\alpha_L(N+2)+\beta_L(\delta(L)-2)}{(\delta(L)-2)(N+2)+2(\delta(L)-2)},$$

so

$$R_{\mathcal{Z}}^N \geq \frac{\alpha_L(N+2)+\beta_L(\delta(L)-2)}{(\delta(L)-2)(N+2)+2(\delta(L)-2)}$$

and

$$R_{\mathcal{Z}}^{\infty} = \sup\{r \mid R_{\mathcal{Z}}^N \geq r, N \in \mathbf{Z}\} \geq \lim_{N\to\infty} \frac{\alpha_L(N+2)+\beta_L(\delta(L)-2)}{(\delta(L)-2)(N+2)+2(\delta(L)-2)} = \frac{\alpha_L}{(\delta(L)-2)}.$$

Letting $C_L = \frac{\alpha_L}{(\delta(L)-2)}$, we are done. ∎

A.4 PROOF OF COROLLARY 1

Proof. Since $\hat{L}$ is polynomial kernel-approximable there exists an algorithm $\mathcal{A}$ that generates conformations on $\hat{L}$ such that $A(s) \leq -\alpha_L X(s) + \beta_L$ for constants α_L and β_L. Each of these conformations is trivially in L, so the proof for Lemma 3 is applicable to L. ∎

A.5 PROOF OF LEMMA 4

Proof. If $L_1 \propto_{D(L_1)} L_2$ then there exists a latticoid $\hat{L}_2$ that is graph isomorphic to a latticoid $\hat{L}_1 \in D(L_1)$. Since L_1 is core approximable, there exists an approximation algorithm $\mathcal{Z}$ for $\hat{L}_1$ such that $\mathcal{Z}(s) \leq -\alpha_L X(s) + \beta_L$ for constants α_L and β_L.

Now consider an approximation algorithm $\mathcal{Y}$ that applies algorithm $\mathcal{Z}$ to an instance s, and then applies the reduction to map the conformation in $\hat{L}_1$ to a conformation in $\hat{L}_2$. Clearly, $\mathcal{Y}(s) = \mathcal{Z}(s)$. Since algorithm $\mathcal{Y}$ is generates conformations in L_2 such that $\mathcal{Y}(s) \leq -\alpha_L X(s) + \beta_L$. L_2 is polynomial approximable. ∎

A.6 PROOF OF THEOREM 1

Proof. Consider $L' \in \mathcal{L}$. Since L is $\mathcal{L}$-complete via $D(L)$, $L \propto_{D(L)} L'$. Let $\hat{L} \in D(L)$ and $\hat{L}'$ be the latticoids that are isomorphic under this reduction. Since L is polynomial core kernel-approximable, $\hat{L}$ is polynomial kernel-approximable. It follows from Lemma 3 that $\hat{L}'$ is polynomial core kernel-approximable. From Corollary 1 this implies that L' is polynomial kernel-approximable. This argument applies for any $L' \in \mathcal{L}$, so $\mathcal{L}$ is polynomial kernel-approximable. ∎

A.7 PROOF OF THEOREM 3

Proof. We show that if $\mathcal{L}$ is the cubic lattice then L-PF is NP-complete. The proof follows similarly for the other crystal lattices.

To transform an instance of OLA to L-PF, we construct a protein instance as follows. Let $\bar{A} \subset A$ be a set of amino acids that correspond to the vertices in V, and let $\bar{f}(a_i) = f(v_i)$, for $a_i \in S$ and $v_i \in V$. Consider

$$S = \underbrace{xxx\ldots xx}_{4n+3} a_1 \underbrace{xxx\ldots xx}_{4n+3} a_2 \underbrace{xxx\ldots xx}_{4n+3} \ldots \underbrace{xxx\ldots xx}_{4n+3} a_n.$$

The costs are

$$C_{s_i,s_j} = \begin{cases} |\bar{f}(s_i) - \bar{f}(s_j)| & \text{if } s_i, s_j \in \bar{A} \\ 0 & \text{otherwise} \end{cases},$$

We use the same parameter B to bound the energy as in the OLA instance. The distance function g is given by

$$g(d^x ij, d^y ij, d^z ij) = \begin{cases} d^x ij/2 & \text{if } d^y ij, d^z ij = 0 \text{ ; and} \\ & d^x ij \text{ is even} \\ (B+1)/C_{\min} & \text{otherwise} \end{cases},$$

where $C_{\min}$ is the smallest nonzero cost in C.

As in Unger and Moult's formulation, small energies are only possible if the a_i lie along a line in the three dimensional lattice. The changes made to their reduction further restrict the optimal conformation to have the a_i lie at an even distance along a line. Figure 4a illustrates the structure of conformations that can assume low energy.

It follows that each of the a_i so configured can be connected by an even-length path of xs. Unger and Moult's arguments suffice to demonstrate that the optimal conformation is found if and only if OLA is solved, with the observation that the additional xs added to the sequence S guarantee that the a_i can be connected when spaced apart in this fashion. ∎

B Table of Symbols and Definitions

Here is a summary of the definitions used throughout this paper.

$\mathbf{Z}$	the set of integer numbers
$\mathbf{R}$	the set of real numbers
s	a protein instance
s_i	the i-th monomer of a protein instance; $s_i \in \{0,1\}$
$l(s)$	the length of s
$M_{max}(s)$	the length of the longest subsequence of zeros in s
$M_{min}(s)$	the length of the shortest subsequence of zeros in s
$E(s)$	the number of connected neighbors in s, s_j and s_{j+1}, for whi ch $s_j = 1$ and $s_{j+1} = 1$
$N(s)$	the number of ones in s
b_i	the i-th block
z_i	the i-th block-separator, between b_i and b_{i+1}
x_i	the i-th x-block
y_i	the i-th y-block
$N_x(s)$	the number of ones in s that are in x-blocks
$N_y(s)$	the number of ones in s that are in y-blocks
B_x	the total number of x-blocks
B_y	the total number of y-blocks
$X(s)$	the number of ones in the x-blocks; $X(s) = N_x(s)$
$Y(s)$	the number of ones in the y-blocks; $Y(s) = N_y(s)$
s_i^x	the i-th one in s that is in an x-block
s_i^y	the i-th one in s that is in an y-block
$T_x(s)$	the number of endpoints that are ones in an x-block
$T_y(s)$	the number of endpoints that are ones in an y-block
$OPT_L(s)$	the energy of the optimal conformation of s on the lattice L
S_N^L	the set of protein instances whose optimal energy is less than equal to N on the lattice L