Exploring the HP Model for Protein Folding

A Major Qualifying Project Report
submitted to the Faculty of
WORCESTER POLYTECHNIC INSTITUTE
in partial fulfillment of the requirements for the
Degree of Bachelor of Science

By:

Matthew Gilzinger

Approved:

Prof. Brigitte Servatius

April 24, 2012

Contents

List of Tables

List of Figures

1 Motivation

1.1 What is a Protein?

An amino acid consists of a carboxyl group, an amine group, and a side chain molecule bonded together. Amino acids are classified based on their side chain and these side chains can be hydrophobic or they can be hydrophillic. The carboxyl group in an amino acid can chemically bond to the amine group in another amino acid. Because of this, strings of amino acids can be made, and these strings are called *proteins*. Proteins are used for tasks in cells, and their structure is essential to their function.

Primary Structure

The *primary structure* of a protein is the order of the amino acids comprising the protein. The secondary and tertiary structures of a protein are determined by the nature of the environment of the protein and the primary structure of the protein.[8]

Secondary Structure

The *secondary structure* of a protein is local three dimensional structure in a protein. The secondary structure could specify that a certain substring of a protein folds into a helix, while a different substring folds into a sheet, but the secondary structure says nothing about the positions of the sheet and helix relative to each other. The possible local structures in the secondary structure are helices, sheets, and random coil. Random coil is a substring of a protein that has no clear or distinct shape, and this is due to weak or possibly non-existant forces between amino acids in the substring.[8]

Tertiary Structure

The *tertiary structure* of a protein is the three-dimensional shape of the entire protein.[24] While secondary structure focuses on local shape, the tertiary structure is the global shape of the protein, and is harder to evaluate. Because of the link between the three-dimensional shape of the protein and its function, the tertiary structure of a protein is of significant scientific interest.[8]

Quartenary Structure

While *tertiary structure* is the shape of *a* protein, quartenary structure is how multiple proteins interact or bond with each other. Hemoglobin for example, is comprised of four subunits. Quartenary structure is not in the scope of this paper, so it will not be discussed much further.[8]

1.2 What is Protein Folding?

Protein folding is the process through which a protein reaches its final shape. A protein is under many inter-molecular forces, such as hydrogen bonds and interactions between hydrophobic aminon acids and the solution the protein is contained in, and these forces caues the chain to bend and twist, while reducing it's energy until it reaches a minimal energy state. Almost 25% of all Nobel prizes in chemistry since 1956 have been related to protein folding.[11] Because a proteins function is dependent on its shape, if a protein misfolds, it becomes inert, or worse, dangerous to the cell that it resides in. Parkinsons disease is caused by misfolded proteins aggregating.[7] By the "thermodynamic hypothesis", the final shape a protein takes is the shape that has the lowest possible energy for the protein. Determining the final shape a protein takes, or protein folding, is a major problem in the field of computational biology. [24]

1.3 Using Differential Equations to Model Protein Folding

There are many intermolecular forces in a protein, such as dipole-dipole interactions, hydrogen bonding, and hydrophobic-hydrophobic interactions. To find the minimal energy state of a protein, we could create a system of differential equations to solve, where we are solving for the positions of all of the atoms in the protein, with the equations being the forces between all pairs of atoms in the protein. If we take myoglobin for example, we have a protein of approximately 150 amino acids, each of which contain dozens of atoms. As a result, myoglobin consists of thousands of atoms, so the number of atom-atom pairs in myoglobin is in the millions, if not tens of millions. Solving such an enormous system of differential equations is computationally prohibitive, and as a result, cruder approximations need to be used.

1.4 A Discrete Model

The number of ways a protein of a given length can bend are limited, and as a result it is possible to find the final conformation of a protein with brute force by calculating every possible conformation of a protein and finding the one with the least energy. The number of possible conformations a protein can take is vast; it increases at an exponential rate with respect to the number of amino acids.[13, 25] Because of this, using the brute force method of finding the exact confomation of a protein with more than a short length is infeasible. Because of this, we have to simplify the search space at the cost of accuracy.

In 1989, Ken Dill proposed a simplified model for exploring the process of protein folding by using points on a regular lattice and an energy function. This model is called the HP Model.[17] The HP model is based on the assumption that hydrophobic-hydrophobic amino acid interaction is a primary factor in protein folding, and quite a bit of evidence has accrued to support this assumption.[11]. Variants of the HP Model have been proposed and studied.[1, 16, 18, 22] The HP Model will be discussed further in Chapter 2.

1.5 Kinetics of Protein Folding

Levinthal's paradox is the difficulty in explaining how the difference between the staggeringly huge number of possible conformations a protein can take is, and how quickly a protein folds to a global minimum, can be bridged.[5] For a crude example, we can use myoglobin, a protein found in muscle tissue in humans, which consists of approximately 150 amino acids. If we assume that every amino acid can take one of four possible states, there is $4^{150} = 2^{300} \approx 10^{90}$ possible states for myoglobin to conform to, which is underestimating the number of ways the amino acids can bend. If the protein randomly samples one million possible conformations a nanosecond, it would take 3×10^{64} *millenia* for the protein to test all possible conformations!

Clearly myoglobin folds much faster than that. Clearly a protein uses a method other than randomly or sequentially testing all possible conformations, so how does a protein fold into a minimal energy shape? The HP model is useful because it greatly simplifies the search space for a protein, but allows enough structure to study how the process of protein folding works. Recent experiments have shown that proteins fold by quickly optimizing energy on a local scale before optimizing it on a global scale.[11]

1.6 Nuclear Magnetic Resonance (NMR) Spectroscopy

Nuclear Magnetic Resonance (NMR) Spectroscopy is used in experimentally determining the final structure of a protein. By analyzing the *cross peaks* in the output of a NMR spectroscopy, one can determine which amino acids are 'close' (less than 5 angstroms apart) which provides very useful data when computing the tertiary or secondary structure of a protein. However, NMR spectroscopy has limitations. The protein sample must be crystalized to be able to be used in NMR spectroscopy, and the molecular weight of the protein must be less than approximately 30,000 dals. By combining computational models with experimental data, scientists can determine the final shape of many proteins. [26]

1.7 Inverse Protein Folding

Where in protein folding, the goal is to predict the final structure of the protein given an amino acid sequence, the goal of inverse protein folding is to find an amino acid sequence that generates a protein structure.[19] This paper will not consider this problem. There is also literature on algorithms for protein folding, [3, 9, 15] but we won't discuss them either.

2 Mathematical Background

2.1 Essential Graph Theory Background

This paper, especially this section, will use a significant amount of graph theory, a branch of discrete mathematics.[21] Using "Graphs & Digraphs" by Chartrand et al., [6] essential basic definitions from the book are listed:

Graph — A *graph* G is a finite nonempty set V of objects called *vertecies* together with a possibly empty set E of 2-element subsets of V called *edges* To indicate that a graph G has *vertex set* V and *edge set* E, we write $G = (V, E)$.

Subgraph — A graph $G' = (V', E')$ is a *subgraph* of another graph $G = (V, E)$ if $V' \subseteq V$ and $E' \subseteq E$.

Isomorphic — Two graphs G and H are *isomorphic* if there exists a bijective function $\phi : V(G) \to V(H)$ such that two vertecies u and v are adjacent in G if and only if $\phi(u)$ and $\phi(v)$ are adjacent in H. The function ϕ is called an *isomorphism* from G to H. If there is no such function ϕ as described, then G and H are *non-isomorphic graphs*

Incident — A vertex v and an edge e in a graph are incident with each other if one of the vertecies comprising e is v.

Adjacent — Two vertecies u and v in a graph (V, E) are adjacent if there is an edge in E incident to both u and v.

Walk — For two (not neccessarily distinct) vertecies u and v in a graph G, a $u - v$ *walk* W in G is a sequence of vertecies in G, beginning with u and ending w v such that consecutive vertecies in W are adjacent in G.

Path — A path is a walk in a graph G in which no vertex is repeated.

Cycle — A walk in a graph G where the starting vertex is the same as the ending vertex and no other vertex is repeated.

Degree — In a graph G the *degree* of a vertex v, denoted $deg(v)$, is the number of edges in G that are incident to the vertex v.

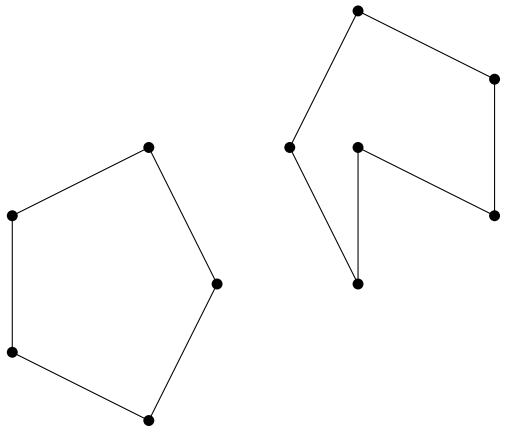


Figure 1: Two simple polygons, P_1 and P_2

2.2 The Handshake Principle

Imagine a conference attended by many people, some of them shaking hands. If we want to know how many handshakes took place, we could watch the entire conference and count the handshakes as they occur. Another way to count the number of handshakes is to hold a poll at the exit, and ask everyone who attended the conference how many hands they shook. By adding the all of the answers, and dividing the sum by two, we get the same result: how many handshakes took place. Each indidivdual's answer is a "local" property, while the observer's answer is a "global" property.

A more formal statement of the relation between the local and global methods of handshake counting can be given by representing all the attendees as vertecies of a graph, and each representing each handshake as an edge. If $G = (V, E)$ is a graph, then

$$\frac{1}{2} \sum_{v \in V} deg(v) = |E|$$

Since every edge is a 2-subset of V, the preceding equation holds for every graph $G = (V, E)$. We will refer to this as the *Handshake Principle* because the name represents the basic truth of this theorem; it takes two 'hands' to make a 'handshake'.

2.3 Pick's Theorem

Another example of how global properties are influenced by local properties is Pick's Theorem. Pick's Theorem applies to 'simple polygons', and by *simple* polygons, such as the ones shown in Figure 1, we mean shapes on a euclidean

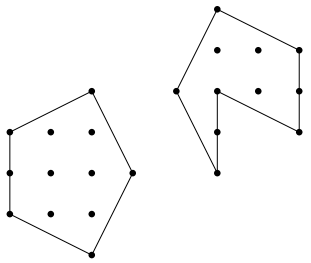


Figure 2: P_1 and P_2 with all interior and boundary points shown

plane that can be defined by a finite sequence of integer points, which we will call the vertecies of the polygon, ending at the starting point. We will call line segments between consecutive vertecies in a polygon edges, and we note that these edges include their endpoints. Furthermore, these simple polygons need not be convex or without 'holes' in them, but no two edges in the polygon can intersect. For Pick's Theorem, we need to be able to tell which points are 'inside' the polygon, and which ones are on the boundary. We say a point is on the boundary of a simple polygon if it is on an edge of the simple polygon. For the 'interior' and 'exterior', we will use a parity definition. For any point in the plane of the polygon, if we take a ray in any direction *in the plane* of the polygon, it will intersect with the edges of the polygon a finite number of times unless the ray is colinear with one of the edges. Because there are only a finite number of edges in the polygon, we can find a ray from the point that is not colinear with an edge in the polygon. We will denote N to be the number of times this ray intersects with an edge of the polygon. If N is odd, we say the point is in the *interior* of the polygon, and if N is even, we say the point is in the *exterior of the polygon*. Finally, the area of a polygon is a measure of the size of the interior of the polygon.

Pick's Theorem states that for any simple polygon where the vertecies lie on integer points, then where i is the number of integer points inside the polygon not on the boundary and where b is the number of integer points on the boundary of the polygon, then the area A of the polygon is

$$A = i + \frac{b}{2} - 1$$

We submit an incomplete proof to help convince the reader of the truth of this theorem.

Supose we have a simple polygon P with area A that can be divided into two smaller polygons P_1 and P_2 with areas A_1 and A_2 respectively, such as in Figure 3. Clearly $A = A_1 + A_2$. Let's say we know that Pick's theorem is true for P_1 and P_2, so we want to show that Pick's theorem applies to P as well.

Let the number of interior integer points of P_1 and P_2 be i_1 and i_2 respectively and b_1 and b_2 be the number of boundary points respectively. Assuming

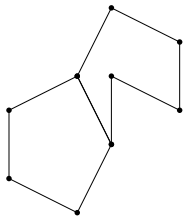


Figure 3: A simple polygon P divided into two smaller simple polygons, P_1 and P_2

that the theorem holds for these smaller examples, we have $A_1 = i_1 + \frac{b_1}{2} - 1$ and $A_2 = i_2 + \frac{b_2}{2} - 1$ for the areas for polygon 1 and polygon 2. Suppose we were to attach one edge of polygon 1 to one edge of polygon 2 in such a manner that only the boundaries of the two polygons overlap and there is only one connected overlapping region, a line segment, and there are o overlapping integer points on the mutual border. Because the two objects are polygons with vertexes being integer points, the line segment begins and ends on integer points. In this process the overlap of the two boundaries consists of two endpoints, which happen to remain boundary points, and other points in between, which become interior points. So, the number of boundary points in the new polygon becomes $b = b_1 + b_2 - 2o + 2$ and the number of interior points becomes $i = i_1 + i_2 + o - 2$. So, the area of the new polygon becomes $i + \frac{b}{2} - 1 = i_1 + i_2 + o - 2 + \frac{b_1 + b_2 - 2o + 2}{2} - 1 = i_1 + i_2 + o - 2 + \frac{b_1}{2} + \frac{b_2}{2} - o + 1 - 1 = i_1 + \frac{b_1}{2} - 1 + i_2 + \frac{b_2}{2} - 1 = A_1 + A_2$ which shows that the area of the two adjoined polygons still satisfies the theorem, $A_1 + A_2 = A = i + \frac{b}{2} - 1$.

For further reading, a full proof is found at [14], a generalization is found at [12], and an extension to higher dimensions is found at [4]

2.4 Lattices

A *lattice* is a set of points with special properties. We can define a lattice to be the span of a set of vectors in $\mathbb{R}^n$ using integer coefficients. Note the difference

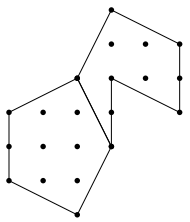


Figure 4: Pick's Theorem also applies to polygon P

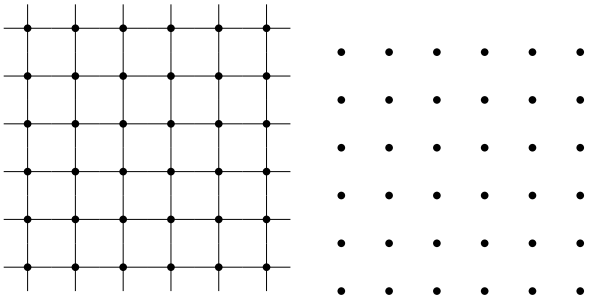


Figure 5: The square *lattice graph* compared to the square *lattice*

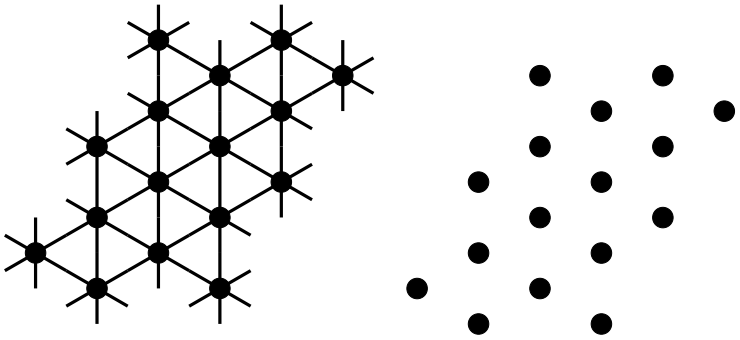


Figure 6: The triangular *lattice graph* compared to the triangular *lattice*

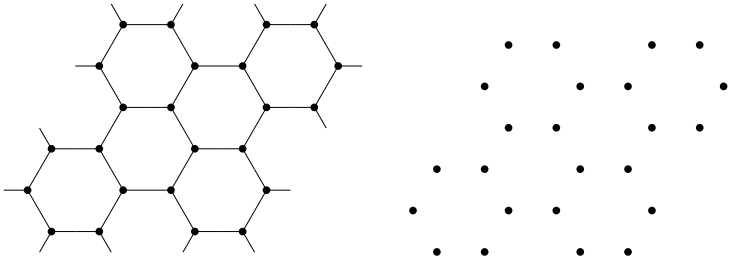


Figure 7: The honeycomb *lattice graph* compared to the honeycomb *lattice*

that specifying integer coefficents as opposed to real coefficents makes. For example,

$$S = \left\{ \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right\}$$

and

$$T = \left\{ \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right\}$$

both span to $\mathbb{R}^2$ when using real coefficients, but when using integer coefficients, S spans to $\mathbb{Z}^2$ yet T spans to $\mathbb{Z}^2 \cup \left\{ \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix} : x, y \in \mathbb{Z} \right\}$

However, to be able to use a lattice like a grid, we need to define edges in some fashion. To do this, we will define a *Lattice Graph* to be a graph with a vertex set that is a lattice in $\mathbb{R}^n$ and a edge set that is all 2-tuples of points in the vertex set which, for some metric function $d : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$. Unless specified otherwise, we will assume that for a lattice in $\mathbb{R}^n$, its lattice graph uses the euclidean metric in $\mathbb{R}^n$ To define a lattice graph, we will use the following notation: $G = (L, d)$ where L is a lattice in $\mathbb{R}^n$ and d is a metric function on the lattice, and if d is the euclidean metric in $\mathbb{R}^n$, the shorthand $G = (L)$ will suffice.

2.5 Common Lattices

Integer Lattice

We will define the integer lattice to be the lattice graph $L = (\mathbb{Z})$. It isn't useful for modeling, but it can be used for defining other latticies using the cartesian product of graphs.

Square Lattice

We can define the square lattice graph to be the graph $L = \left(\mathbb{Z}^2\right)$ We would like to note that taking the cartesian product of two integer latticies yields the square lattice.

Triangular Lattice

We can define the triangular lattice in two ways.

$L = \left(\left\{ (x - \frac{y}{2}, \frac{\sqrt{3}}{2} y) : \forall x, y \in \mathbb{Z} \right\} \right)$

$L' = \left(\mathbb{Z}^2, d\right)$

$$d((x_1, x_2), (y_1, y_2)) = \begin{cases} |x_1 - y_1| + |x_2 - y_2| & : (x_1 - y_1)(x_1 - y_2) < 0 \\ \max\{|x_1 - y_1|, |x_2 - y_2|\} & : (x_1 - y_1)(x_1 - y_2) \geq 0 \end{cases}$$

The former of the two methods of expressing the triangular lattice yields the traditionial lattice where the lattice points are the verticies of a equilateral triangle tiling of the plane, while the latter of the two uses integer lattice points, which is easier for computation.

Honeycomb Lattice

The honeycomb lattice isn't actually a lattice, because the span of the points in the honeycomb lattice is the triangular lattice, but its properties make it worth mentioning. First, the triangular lattice graph and the hexagonal lattice graphs are duals of each other, which means if we take the vertex set to be the midponts of all of the triangular regions formed by the triangular lattice graph, and we let the edge set be all pairs of vertecies where their cooresponding triangles share an edge, then we get the honeycomb lattice graph. If we do the same with the honeycomb lattice, by taking the vertex set to be the centers of all of the hexagonal regions formed by the honeycomb lattice graph, and let the edge set be all pairs of vertecies whose cooresponding hexagons share an edge, we get the triangular lattice graph again.

Using our definition of a lattice, latticies are self similar under translation. This means that if we take an arbitrary point in a lattice, and move the lattice so that the point we picked takes the spot of some other point in the lattice, the lattice stays the same. However, if we translate the honeycomb lattice, it may not be the same, but what we get is *always* a rotation of the original lattice by half a turn if it's not already identical.

Because of these properties, we will call the honeycomb lattice a *semi-lattice* because it is not a lattice, but it has similar properties.

Cubic Lattice

The cubic lattice graph is: $L = \left(\mathbb{Z}^3\right)$ We would also like to note that we can get the cubic lattice by taking the cartesian product of the square lattice and the integer lattice.

Triangular Prism Lattice

If we take the cartesian product of the integer lattice and the triangular lattice, we end up with the triangular prism lattice. The triangular prism lattice is also the verticies of the tiling of 3-space with triangular prisms where the tops and bottoms are equilateral triangles and the 3 sides are squares.

Hexagonal Prism Lattice

If we take the cartesian product of the integer lattice and the honeycomb lattice, we end up with the hexagonal prism lattice.

Self-Avoiding Walks

A self-avoiding walk on the square lattice can be explained as follows:

We can let the square lattice be a graph where the vertex set is $\mathbb{Z}^2$ and the edge set is all pairs of points in $\mathbb{Z}^2$ which have a euclidean distance of 1. We can then create a walk starting from some point on the lattice graph, without loss of generality we can say it is $(0, 0)$, and ending at some other point on the lattice

graph. If no vertex in the walk is repeated, we can call the walk a *self-avoiding walk*. In the square lattice graph a path is a self-avoiding walk.

2.6 A Formal Definition of the HP Model

While many variations and extensions of the HP model have been proposed and explored, we will focus on the original HP model Dill proposed and how the choice of lattice affects it.[16, 18] The HP model as proposed by Ken Dill, simplifies the amino acid sequence of a protein to sequence of hydrophobic and hydrophillic (non-polar and polar) 'beads'. Then, the sequence of 'beads' then is associated with a self-avoiding walk in a lattice-graph with a length of one less than the number of 'beads' by associating each vertex in the self-avoiding walk with the cooresponding bead in the chain of beads. We will call this bead sequence self-avoiding walk pair a 'modeled protein'.

The 'energy' of this modeled protein is calculated by taking every pair of hydrophobic 'beads' that are not adjacent in the primary structure of the protein and adding -1 to the total energy if there is an edge in the lattice graph cooresponding to the two 'beads' (vertecies in the lattice graph). Dill used the square lattice, but different lattices can be used, even ones in higher or lower dimensions. However, we would like to note that modeling the conformation of a protein in one dimension is not very insightful.[10, 20] If we use ($\mathbb{Z}$) as a lattice graph, we only get two possible conformations, one going to the left and one going to the right.

3 Conformations of a HP String

A *conformation* is the shape a modeled protein takes in a lattice in the HP model. Because there is an isomorphism between any two points on a lattice, we can fix a starting point for all paths without any loss of generality. We will use $(0,0)$ or $(0,0,0)$ for latticies defined on $\mathbb{Z}^2$ or $\mathbb{Z}^3$. So, we will say that two conformations are the same if one can be transformed into the other by an isometry. Many latticies have rotational symmetry. The square lattice has rotational symmetry group C_4 while the triangular lattice has rotational symmetry group C_6. So, when we try to find a compact and unambiguous way to express a conformation, we will assume that the first two lattice points are $(0,0),(1,0)$ for lattice graphs on $\mathbb{Z}^2$ and $(0,0,0),(1,0,0)$ for lattice graphs on $\mathbb{Z}^3$ unless the conformation is a single point, then in that case the conformation will be $(0,0)$ or $(0,0,0)$ in the two dimensional or three dimensional cases respectively. We already can fix the starting point due to isomorphism under translation, We note that for any turn any self-avoiding walk makes, the turn cannot take the 'path' 'backwards' on itself. As a result, we can reduce the number of possible directions to go to next in a self-avoiding walk by 1, except for the starting step which remains the same. In the square lattice, it is possible to make a reduction of 4^n possible conformations to $4 \times 3^{n-1}$ possible conformations of length n and in the triangular lattice a reduction from 6^n to $6 \times 5^{n-1}$

We will use a 'relative' system, instead of 'absolute', so for the square lattice for example, instead of using 4 directions, say North, South, East, and West, we will use the 3 directions Straight, Left, and Right. For the triangular lattice graph, instead of using 6 directions, we will use the five: Sharp Left, Left, Straight, Right, Sharp Right. As a result, we can express a self-avoiding walk with n points by a string with $n-2$ symbols, because the first two points are assumed, and every symbol specifies a new point in the walk. However, there is no way to express a single point walk with our method, so we choose to reject all single point self-avoiding walks as trivial and not worth studying.

However, it becomes necessary to distinguish between what is expressable by a conformation code, and what is a self-avoiding walk. The set of all self-avoiding walks on a given lattice graph G is a proper subset of the set of all walks on the same lattice graph G expressable by conformation codes, so we will call walks expressable by conformation codes *self-avoiding walk candidates*. If we choose to express self-avoiding walk candidates as a series of characters, we can encode them with shorthand. For example on the square lattice graph, we can use the symbols

```
- x +
```

to represent Left, Straight, and Right respectively For example, to encode the conformation of the protein in figure 8, we would use

```
x++x-++-x+xx
```

For the triangular lattice, we propose the inclusion of the following symbols for Sharp Left and Sharp Right respectively:

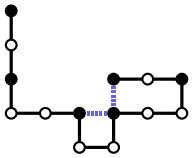


Figure 8: A protein has a conformation code of x++x-++-x+xx Using the bijective relation, this conformation maps to 122,102,201,211

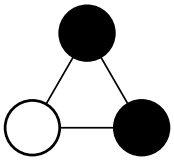


Figure 9: In the triangular lattice, the conformation codes << and >> form cycles

< >

For a simple and elegant way to iterate through all of the self-avoiding walk candidates with n nodes, we note that there is a bijective relation between all positive numbers in base three with up to n digits and all possible conformation codes of length n on the square lattice graph, and similarly between base 5 numbers up to length n and self-avoiding walk candidates on the triangular lattice graph. We can prove this by mapping each symbol in a shorthand notation for a square lattice graph conformation to three digit numbers with up to n digits. We use the following: `-` $\rightarrow 0$, `x` $\rightarrow 1$, `+` $\rightarrow 2$, `<` $\rightarrow 3$, `>` $\rightarrow 4$ because it works for both the square and triangular lattice graphs simultaneously. We would like to note that in the case of the square lattice graph, the sequences `---` and `+++` can never appear because these sequences repesent the smallest cycle possible in the square lattice graph, C_4. Similarly, `<<` and `>>` can never appear in valid conformations in the triangular lattice because these sequences map to C_3 in the triangular lattice. In fact, we can test a conformation for 'validity' in a similar method, a conformation code of length l maps to a self-avoiding walk if and only if it contains no substring that maps to a cycle in the appropiate lattice graph. We could use this fact to find a bound on the number of self-avoiding walks of length n by creating an exhaustive list of conformations that map to cycles, but such a search would be exhaustive and exhausting. We can find a lower bound for the number of self-avoiding walks of length n in the square lattice by only letting the walk go up or right. At every step, the walk moves away from the origin, so it cannot make a cycle, and for every step, the walk can go either left or up, so we get a lower bound of 2^n. Bounds with higher bases for the exponent can be made, but they require significantly more

effort to create, and we already have exponential upper and lower bounds for the number of self-avoiding walks of length n, so we can say that the number of self-avoiding walks of length n increases exponentially with respect to the length of the walk.

4 Lattice Properties

4.1 Square Lattice

The square lattice can be imagined as the set of points $\mathbb{Z}^2$ where two points, $x = (x_1, x_2)$ and $y = (y_1, y_2)$ are adjacent if $|x_1 - y_1| + |x_2 - y_2| = 1$ We would like to note that x and y are adjacent if and only if the sum of the differences between the first and second coordinates of x and y is 1 For convienence, will we define P_S to be the set of all points in the square lattice.

Parity

As Agarwala et. al. mentioned, the square lattice has the *parity property* where for a walk in a square lattice graph, it is impossible for vertecies visited on an even step to be adjacent with other vertecies visted on an even step and similarly for odd aminos.[1, 2] We provide a proof.

Let us divide the points of the square lattice into two sets, E and O:

$$E = \{(x, y) | x + y \ is \ even\}$$

$$O = \{(x, y) | x + y \ is \ odd\}$$

$E \cup O = P_S$ and $E \cap O = \emptyset$. So, if $x \in P_S$ then either $x \in E$ or $x \in O$. Let us assume that $x = (x_1, x_2) \in P_S$. Because a point $y \in P_S$ is adjacent to x in the square lattice iff $|x_1 - y_1| + |x_2 - y_2| = 1$, the only points that are adjacent to x are $(x_1 + 1, x_2), (x_1 - 1, x_2), (x_1, x_2 + 1), (x_1, x_2 - 1)$. For all four of those points, the sum of the coordinates comes out to either $x_1 + x_2 + 1$ or $x_1 + x_2 - 1$. If $x \in E$, then $x_1 + x_2$ is even so all of the points adjacant to x are in O and vice versa. Because a conformation of a modeled protein forces adjacent 'beads' in the string to be adjacent in their embedding in the lattice, 'beads' in odd positions in the string (assuming we start enumeration of the 'beads' at 1) go to even positions in the lattice and 'beads' in even positions in the string go to odd positions in the lattice. We have already shown that only pairs of points not from the same set can be adjacent so 'beads' in odd positions in the string cannot be adjacent to other 'beads' in odd positions in the string and mutatis mutandis.

We note that

4.2 Triangular Lattice

The triangular lattice is traditionally represented as the verticies of an infinite tesselation of equilateral triangles, but for our convienence, we will represent it as $\mathbb{Z}^2$ with a different distance function than the square lattice graph. We will define $d(x, y)$ to be the following:

$$d(x, y) = \begin{cases} |x_1 - y_1| + |x_2 - y_2| & : \ (x_1 - y_1)(x_1 - y_2) < 0 \\ \max\{|x_1 - y_1|, |x_2 - y_2|\} & : \ (x_1 - y_1)(x_1 - y_2) \geq 0 \end{cases}$$

If we say that x and y are adjacent in the triangular lattice if $d(x, y) = 1$), then $(0, 0)$ is adjacent to $(0, 1)$,$(0, -1)$,$(1, 0)$,$(-1, 0)$,$(1, 1)$, and $(-1, -1)$ We can

easily transform these coordinates to the ones in the traditional triangular lattice as such:
$(x, y) \rightarrow (x - \frac{1}{2}y, \frac{\sqrt{3}}{2}y)$

Parity

Unlike the square lattice graph, the triangular lattice doesn't have the parity property. As shown in the sequence in Figure 10, we can force the endpoints of any 'bead sequence' of arbitrary length to be adjacent in the triangular lattice graph. We can do this by forming a ribbon-like structure. A modeled protein embedded in the triangular lattice can achieve a lower energy than a modeled protein embedded in the square lattice.

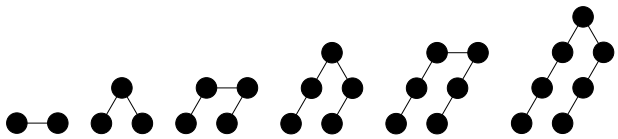


Figure 10: Visual Proof by Handwaving

4.3 Honeycomb Lattice

The honeycomb lattice is the triangular lattice with $\frac{1}{3}$ of its vertices removed and $\frac{2}{3}$ of its edges removed. If we start with the triangular lattice graph defined on $\mathbb{Z}^2$, and remove all vertices (x, y) such that $x + y \ mod \ 3 == 0$ and remove all edges incident to the removed verteces, we get a lattice graph isomorphic to the honeycomb lattice graph. By removing every third point in the triangular lattice, we are left with two different classes of vertices, those where $x + y = 1 \ mod \ 3$ and those where $x + y = 2 \ mod \ 3$, and furthermore, no two vertices in the same class are adjacant! The honeycomb lattice has the parity property. Furthermore, we note that the shortest path in the honeycomb lattice graph that has the two endpoints sharing an edge is 5 edges long. This can be seen easily when we note that the first three edges brings the endpoints of the path further apart no matter which path we choose!

4.4 Cubic Lattice

The cubic lattice is the extension of the square lattice into three dimensions. We can express the set of points of the cubic lattice as $\mathbb{Z}^3$ with $(0, 0, 0)$ as the origin and once again with two points x and y being adjacent if and only if the sum of the differences of their coordinates is 1 or $|x_1 - y_1| + |x_2 - y_2| + |x_3 - y_3| = 1$ [23]

Parity

We can show, with minor modification to the proof for the square lattice, that the cubic lattice also has the parity property.

Let us divide the points of the square lattice into two sets, E and O:

$$E = \{(x, y, z) | x + y + z \ is \ even\}$$

$$O = \{(x, y, z) | x + y + z \ is \ odd\}$$

$E \cup O = P_S$ and $E \cap O = \emptyset$. So, if $x \in P_S$ then either $x \in E$ or $x \in O$. Let us assume that $x = (x_1, x_2, x_3) \in P_S$. Because a point $y \in P_S$ is adjacent to x in the cubic lattice iff $|x_1 - y_1| + |x_2 - y_2| + |x_3 - y_3| = 1$, the only points that are adjacent to x are $(x_1 + 1, x_2, x_3), (x_1 - 1, x_2, x_3), (x_1, x_2 + 1, x_3), (x_1, x_2 - 1, x_3), (x_1, x_2, x_3 - 1), (x_1, x_2, x_3 + 1)$ For all six of those points, the sum of the coordinates comes out to either $x_1 + x_2 + x_3 + 1$ or $x_1 + x_2 + x_3 - 1$. If $x \in E$, then $x_1 + x_2 + x_3$ is even so all of the points adjacant to x are in O and vice versa. Because a conformation of a modeled protein forces adjacent 'beads' in the string to be adjacent in their embedding in the lattice, 'beads' in odd positions in the string (assuming we start enumeration of the 'beads' at 1) go to even positions in the lattice and 'beads' in even positions in the string go to odd positions in the lattice. We have already shown that only pairs of points not from the same set can be adjacent so 'beads' in odd positions in the string cannot be adjacent to other 'beads' in odd positions in the string and mutatis mutandis.

5 Bounds

The energy of a modeled protein is determined by the following formula where C is the number of contacts of the modeled protein in the lattice and E is the energy of the protein:
$E = -C$

The number of contacts that a protein can have is at least zero, so we immediately have an upper bound for the possible energy a protein can have: zero. Furthermore, this bound is exact and cannot be lowered because if we take the modeled protein where the bead chain makes a straight line with no bends, it is impossible for any contacts to form, regardless of the bead chain, so this protein conformation has a energy of 0. As a result, we will only consider lower bounds for the energy of a modeled protein as the upper bound is trivial.

5.1 Lower Bounds on Minimal Energy

5.1.1 Square Lattice

Naïve Bound To start off this subsection, we will introduce a simple and naïve lower bound for the lowest energy that a protein can take. With the exception of the endpoints of the bead string, every black bead can come into contact with at most two other beads, but if the bead is an endpoint of the bead chain, it can come into contact with at most three other beads. However, because of the handshake rule, we are double counting each contact so we have to treat endpoints as contributing $\frac{3}{2}$ contacts and midpoints as contributing 1 contact to the total number of possible contacts. So, if we assume a bead chain consisting solely of hydrophobic beads, we get a lower bound for the energy of the protein of: $E \geq n+$ where n is the number of beads in the bead chain and E is the energy of a modeled protein with length n. Because replacing a hydrophobic bead with a polar bead can never add more contacts, this bound applies to *all* modeled proteins of a given length n If we can count the number of hydrophobic beads in the bead chain, we can get a slightly better bound. If we take H to be the number of hydrophobic beads in the bead chain, we can use $E \geq H + 1$ instead of $E \geq N + 1$ as a lower bound, because polar beads can't form any contacts under the Dill HP model.

Parity Bound As discussed in Chapter 4, the square lattice has the parity property where if you divide the beads in the modeled protein by their position in the modeled protein's bead chain into two sets by whether or not their position in the sequence is even and odd, contacts between two beads can only happen if the two beads are in different groups. We can create a bound on the number of contacts a protein can have and as a direct result, we can bound the energy a protein can take.

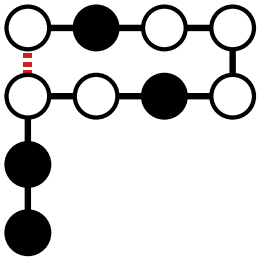


Figure 11: A modeled protein

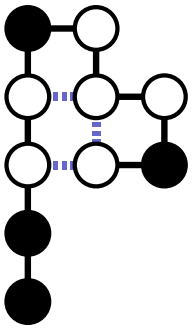


Figure 12: A minimal energy conformation of the protein in Figure 11

In Figure 11, hydrophobic amino acids are white beads and polar amino acids are black beads, and bonds are represented by solid lines and contacts are represented by dashed lines. Starting at the top left corner, and using the abbreviations N for hydrophobic (non-polar) and P for polar amino acids, we get the following as the bead chain:
$NPNNNPNNPP$
Using the enhanced naïve lower bound, we get $E \geq -7$ which is not a very good lower bound considering the lowest energy this bead chain can take is -3, as seen in the Figure 12.

We can use the parity of the square lattice to our advantage. We can seperate the beads into two groups, based off whether each bead is in an even or odd position in the HP string. By splitting NPNNNPNNPP into two groups this way, we get $N_P N_N N_P N_N P_P$ so the two groups are:
$N\_N\_N\_N\_P$ (odd) and $\_P\_N\_P\_N\_P$ (even)
By the handshake rule, a bead may come into contact with at most 2 other beads, unless the bead is at the end of a bead chain, then it can come into contact with 3 other beads. We see that the first group has four non-polar beads with one corresponding to an endpoint while the second group has two hydrophobic beads neither of which correspond to an endpoint. As a result,

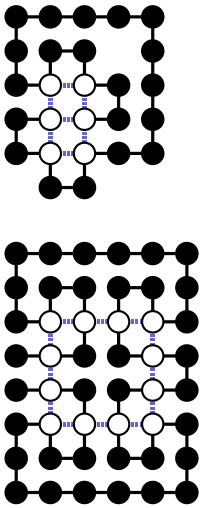


Figure 13: Two modeled proteins showing that the lower bounds on energy given by the naïve bound can be achieved.

the beads in the first group can have at most $3 + 2 + 2 + 2 = 9$ contacts and the beads in the second group can have at most $2 + 2 = 4$ contacts. Because contacts between beads are mutual, it follows that both groups should share the same number of contacts in a protein. We get the minimum of 9 and 4 for an upper bound on the number of contacts the bead sequence can make leaving us with -4 for a lower bound for the optimal energy of the bead chain. The lower bound of -4 is much closer to the actual minimum energy of the example modeled protein which is -3, than the naïve bound of -7

5.1.2 Triangular Lattice

Naïve Bound We will once again discuss the naïve bound, but this time we will explore the cubic lattice. In the infinite triangular lattice, every point is adjacent to exactly six other points. When a bead is on a endpoint of a bead chain, it has exactly one bond, which leaves five adjacent points to form contacts, and similarly for beads in the middle of a bead chain, they have exactly two

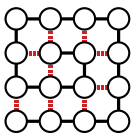


Figure 14: For chains consisting solely of hydrophobic amino acids, squares like this one have the lowest energy possible for the chain in the square lattice

bonds, which leave four adjacent points to form contacts. With this knowledge, we can form a very naïve lower bound for the possible energy of a modeled protein in the triangular lattice, $E \geq 2N + 1$ where N is the length of the HP string. We can improve the naïve lower bound by replacing the total number of beads with the total number of hydrophobic beads, H, to get $E \geq 2H + 1$

Parity Bound Because the triangular lattice doesn't have the parity property, we cannot create a better lower bound on the lowest energy a protein can take in the triangular lattice by splitting the HP string.

5.1.3 Cubic Lattice

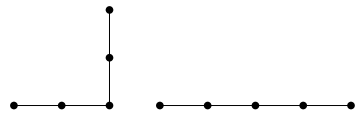
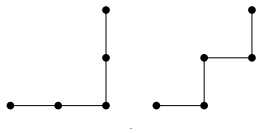
Naïve Bound When we consider that any point in the cubic lattice is surrounded by 6 other points, we can easily get the same results for naïve bounds for the cubic lattice: $E \geq 2N + 1$ where N is the length of the HP string. Similarly where H is the number of hydrophobic beads in the bead chain, $E \geq 2H + 1$

Parity Bound Like the square lattice, the cubic lattice also has the parity property. The procedure for calculating the lower bound for energy using the parity principle is exactly the same as calculating the lower bound for energy using the parity principle on the square lattice except we acknowledge that endpoint beads can have up to 5 contacts instead of 3 contacts and that midpoint beads can have up to 4 contacts instead of 2. We will use the example HP string from earlier to demonstrate this.

We can use the parity of the cubic lattice to our advantage. We can seperate the beads into two groups, based off whether each bead is in an even or odd position in the HP string. By splitting NPNNNPNNPP into two groups this way, we get $N_P N_N N_P N_N P_P$ so the two groups are:
$N\_N\_N\_N\_P$ (odd) and $\_P\_N\_P\_N\_P$ (even)
By the handshake rule, a bead may come into contact with at most 4 other beads, unless the bead is at the end of a bead chain, then it can come into contact with 5 other beads. We see that the first group has four non-polar beads with one corresponding to an endpoint while the second group has two hydrophobic beads neither of which correspond to an endpoint. As a result, the beads in the first group can have at most $5 + 4 + 4 + 4 = 17$ contacts and the beads in the second group can have at most $4 + 4 = 8$ contacts. Because contacts between beads are mutual, it follows that both groups should share the same number of contacts in a protein. We get the minimum of 17 and 8 for an upper bound on the number of contacts the bead sequence can make leaving us with -8 for a lower bound for the optimal energy of the bead chain. The lower bound of -8 higher than the naïve bound of $-\frac{5+4+4+4+4+4}{2} = -\frac{25}{2} = -12.5$.

6 Energy Landscapes

To model protein *folding*, we need a way to *fold* a modeled protein. We propose three operations on modeled proteins in the square lattice. If the walk that is generated by an operation is a self-avoiding walk, then the operation is valid, otherwise the operation is invalid for the conformation operated on.

- *Identity*: We propose the inclusion of an identity operation. The identity operation leaves the modeled protein unchanged.
- *Bend*: If we can represent the conformation as a sequence of relative steps, we can define the bend operation to be where we change one of the directions in the sequence to another direction. 
- *Flip*: If we represent the conformation of as a sequence of absolute steps, such as up, down, left, or right, we can define the flip operation to be the interchange of two consecutive steps. If the two steps are the same, then the flip operation is the same as the identity operation. 

We will refer to these operations as the *basic operations*.

We would like to call two conformations *equivalent* if one can be transformed into the other using a sequence of the basic operations. Furthermore, we call two conformations *equipotent* if one can be transformed into the other using a sequence of the basic operations, and the initial, final, and all of the intermediate conformations have the same energy. We can extend the three operations to the triangular lattice without changing their definitions.

Note that in the rest of this chapter, polar amino acids are white beads and hydrophobic amino acids are black beads

Proteins can fold rapidly, in some cases proteins can reach their optimal configuration in *microseconds* despite the immense number of conformations a protein can take.[11] A theoretical structure that has been used to explain this and other puzzling properties of protein folding is the *energy funnel*.[11] The energy funnel is based off the concept that there is more than one way for an unfolded protein to reach its native state, and that the *energy landscape*, roughly takes the shape of a 'funnel' as one approaches the optimal state while folding.

Given the HP string NpNppNppNppNpN, we would like to find a sequence of operations that will reduce the energy of our modeled protein to the minimal energy without increasing the energy at any point.

Figure 15: The unfolded state of the protein "NpNppNppNppNpN" in the square lattice

Before we proceed any further, we will use some of the techniques discussed in Chapter 5 to get a lower bound on the minimal energy of the HP string NpNppNppNppNpN in the square lattice. To use the parity bound from Chapter 5, we will need to split the string into two groups. The first group will contain the first bead and every other bead after that, so the first group is N_N_p_p_N_p_p_. The second group will contain the second bead and every other bead after that, so the second group is _p_p_N_p_p_N_N. Using the parity bound, we get $min\{-|3+2+2|, -|2+2+3|\} = -7$ as a lower bound on the minimal energy of the HP string NpNppNppNppNpN in the square lattice. For this modeled protein to achieve an energy of -7, the 6 black beads must fully fill a 2×3 block in the square lattice. If we let this happen, then the endpoints must reside in the centers of the long edges of the rectangle. Otherwise, the endpoints would be unable to come into contact with three other black beads.

Without loss of generality, we can assume that the black beads forms a rectangle that is two beads tall and three beads wide. Furthermore, we can study the endpoint that resides in the upper of the two midpoints of the rectangle without any loss of generality. The next black bead in the HP sequence is separated from the endpoint by a single white bead. Because the endpoint is at the middle of the top of the rectangle, the first step must be upwards. As a result, the next bead, which is a black bead, cannot reside in the rectangle, proving that the modeled protein cannot reach a minimal energy of -7 in the square lattice. Therefore, the minimal energy of the modeled protein is -6 or higher in the square lattice.

Can we improve this bound further? Because there are only a finite number of possible conformations, we can prove that the minimal energy of the HP string NpNppNppNppNpN in the square lattice is -5 by examining all of the possible conformations. We claim without proof that the conformation in Figure 16 is optimal and now proceed with the folding sequence in Figure 17:

As you can see, the two ends of the protein folded together to form a 'ribbon'. There are no basic operations that the modeled protein can use without increasing its energy. We will try a different set of moves to try to reach a state of minimal energy, in Figure 18

The sequence in Figure 18 leads to a final state with a lower energy than the one in Figure 17. Note how the two black beads on the corners of this conformation are isolated. Furthermore, none of the basic operations can be applied to this conformation without increasing the energy of the modeled protein.

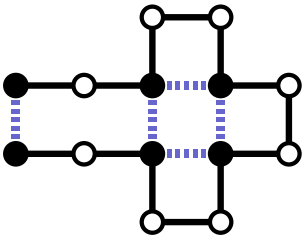


Figure 16: A minimal energy conformation of the protein

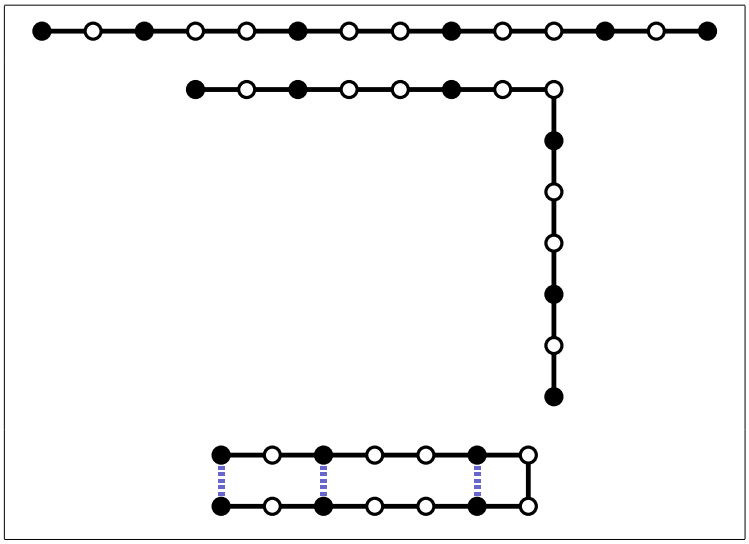


Figure 17: A sub-optimal folding sequence of NpNppNppNppNpN

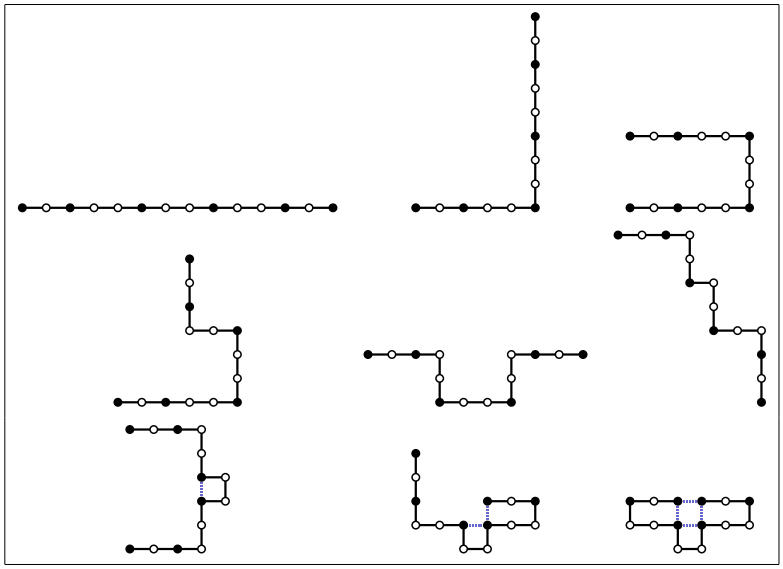


Figure 18: A folding sequence that results in a final energy of -4

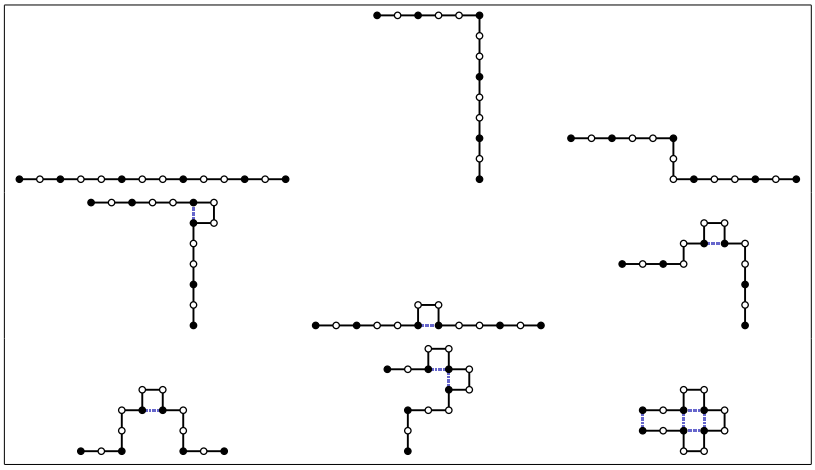


Figure 19: A folding sequence that results in a minimal energy conformation.

The folding sequence in Figure 19 starts from the unfolded state and ends in a minimal energy state. Furthermore, the modeled protein goes between these states without increasing its energy at any point.

What we have shown in the previous three examples is that the energy of a modeled protein as it folds to a minimal energy state is not necessarily non-increasing. Furthermore, we have shown that it is possible for a protein to end up in a sub-optimal conformation where it cannot fold to a lower energy state without first increasing its energy. In fact, the final state in all three examples are ones where there is no valid operation that can be applied without increasing the energy of the modeled protein. Based off of this fact, we will introduce some terminology.

- An *energy well* is the set of all conformations that are equipotent to a given conformation, including itself.
- A *proper* energy well is where no conformation in the energy well can be transformed into a conformation of less energy without increasing the energy at any point.
- A *minimal* energy well is an energy well comprising of native states.

Assuming that our conjecture that all conformations of a modeled protein are equivalent is true, for a given conformation in a proper energy well, there is at least one folding sequence from the given conformation to a conformation with minimal energy. Every folding sequence is a finite sequence of conformations, so there is a maximum energy attained in the sequence. Because there is a finite number of sequences, we can take the minimum of all of these maximum energies. We take this minimum energy, and we call the difference between this minimum energy and the energy of our given conformation to be the *depth* of the energy well. It doesn't matter which conformation in the energy well we use to get the depth because the result will always be the same.

We would like to point out, without proof, that the depth of the energy well of the final state in Figure 18 is 2, and the depth of the energy well of the final state in Figure 17 is also 2. If we visualize the energy landscape as a funnel shaped sheet, energy wells would be local minima in the sheet.

This leads us to a question, is it possible that a protein that has deep energy wells is more likely to misfold than one with shallow energy wells?

7 Case Studies

For the purposes of this section, we will denote hydrophobic beads as N and polar beads as p. So, a HP string with three hydrophobic beads followed by four polar beads would be NNNpppp.

7.1 pNNpNpNpNNN

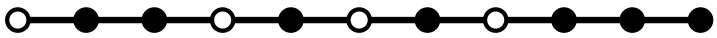


Figure 20: The unfolded conformation of pNNpNpNpNNN

The HP string pNNpNpNpNNN has only one minimal energy conformation in the square lattice, unique up to isometry, but it has 6 minimal energy conformations in the triangular lattice, unique up to isometry.

Figure fig:bob is the unique, up to isometry, minimal energy conformation of the HP string pNNpNpNpNNN

In the triangular lattice graph, the conformations in Figure 22 are equipotent. There are only two differences, which way the hydrophobic tail bends in the center and which direction the loose polar end faces. In fact, the six conformations in the triangular lattice forms a single minimal energy well. Through the flip operation, we have $\epsilon \leftrightarrow \phi$, $\alpha \leftrightarrow \beta$, and $\gamma \leftrightarrow \delta$. And by bending the polar end cap through all three possible conformations, we get $\alpha \leftrightarrow \gamma \leftrightarrow \epsilon$, and $\beta \leftrightarrow \delta \leftrightarrow \phi$. As a result, we find that through our basic operations, we get that $\{\alpha, \beta, \gamma, \delta, \epsilon, \phi\}$ is the minimal energy well. The 6 different conformations are essentially same shape. Also, not too surprisingly, the hydrophobic core forms a ball surrounded by non-polar residues. In this manner, a is a similar shape to the triangular lattice minimal energy conformations because it is a 'glob' of non-polar residues surrounded by polar residues.

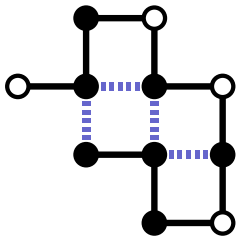


Figure 21: Conformation a

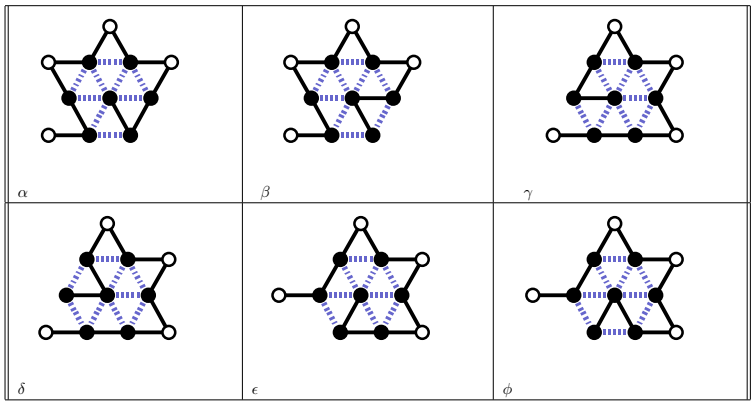


Figure 22: Conformations α,β, γ, δ, ϵ, and ϕ of pNNpNpNpNNN

Table 1: Number of conformations with a given energy in the square lattice

0:	3,962
-1:	1,307
-2:	225
-3:	18
-4:	1
SUM:	5,513

Table 2: Number of conformations with a given energy in the triangular lattice

0:	50,546
-1:	111,202
-2:	100,919
-3:	53,223
-4:	19,737
-5:	6,774
-6:	1,420
-7:	453
-8:	151
-9:	6
SUM:	344,431

Figure 23: Scatterplot of the Energy, Diameter, and Area of the conformations of pNNpNpNpNNN in the Square Lattice

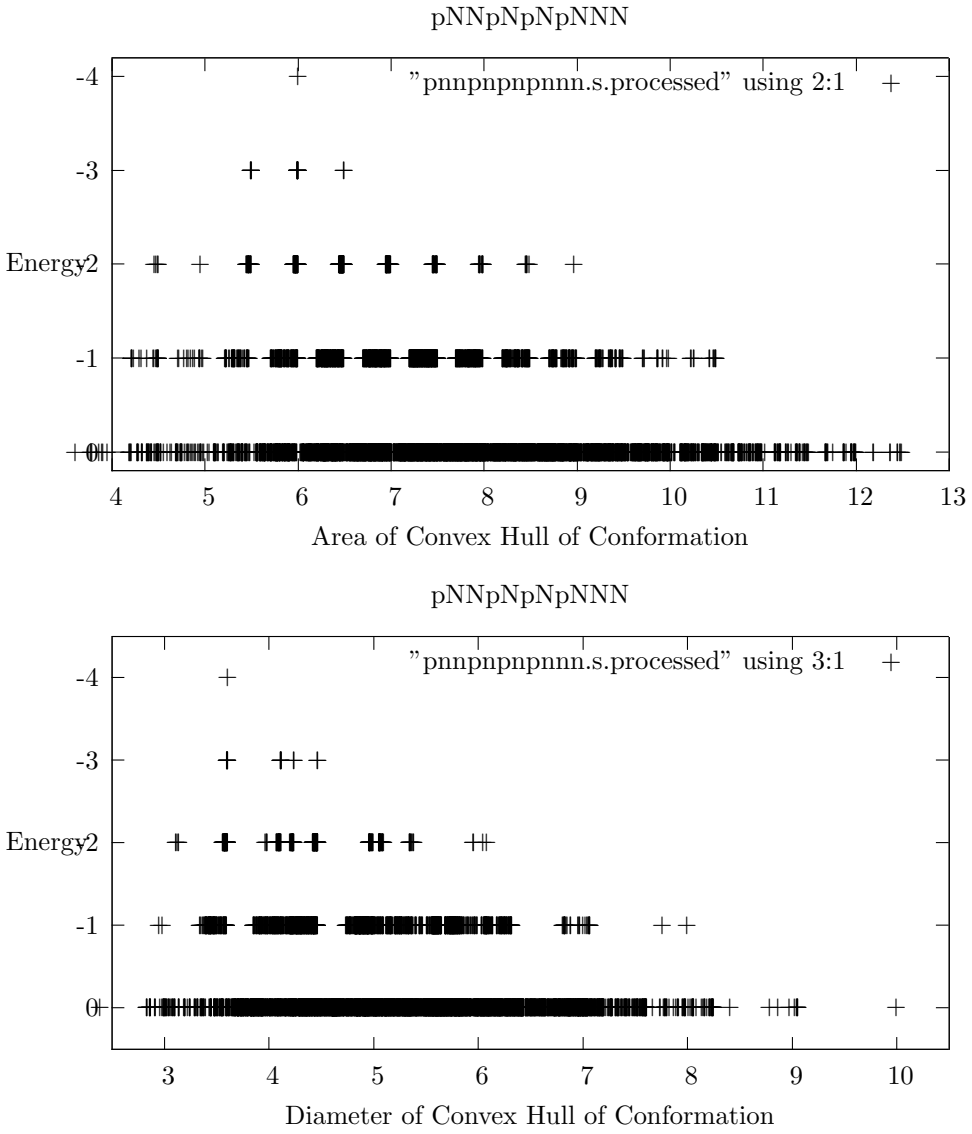
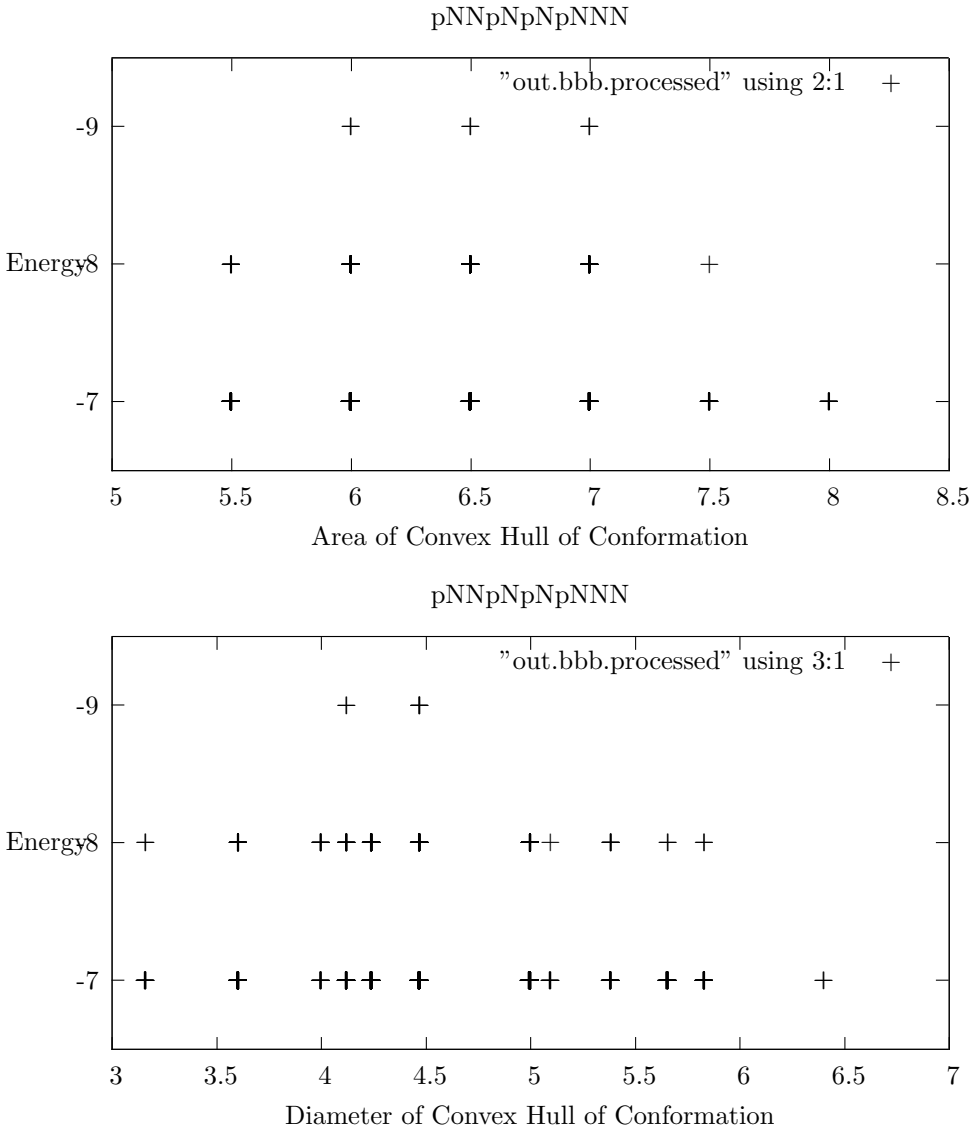


Figure 24: Scatterplot of the Energy, Diameter, and Area of the conformations of pNNpNpNpNNN in the Triangular Lattice with energy of at most -7



For the purpose of these case studies, we will define the term *diameter*. For a given point p in a set of points P, we will call the *eccentricity* of p to be the maximum value of the set $E = \{d(p, x)|x \in P\}$ where $d(a, b)$ is the euclidean distance between two points, a and b. We will represent the eccentricity of a point p in a set of points P as a function, $ecc(p, P)$. Now, we will define the *diameter* of a set of points, P, as the maximum of the set $F = \{ecc(p, P)|p \in P\}$.

Figures 7.1 and 7.1 show that there are more conformations than possible diameters or areas of the convex hulls of the conformations. This is because it is possible to have two or more conformations map to the same set of points, but in a different order. For an example of this, see Figure 22. In Figure 22, conformations α and β have the same set of points, but they are in a different order.

In Figure 7.1 and Figure 7.1, we also see something else. It appears that for each energy level in the scatter plots, the points are centered around the minimal energy conformation's points. It also appears that the conformation which corresponds to a straight line is an outlier, in 7.1 we see that it lays significantly further away from the other conformations with energy 0. Perhaps the distance from the center can be used as a measure of entropy? In the straight line conformation, the points are as stretched out as far as possible from each other.

We see in tables 1 and 2 that the energy state of 0 is the most common in the square lattice, but in the triangular lattice, the most common energy state is -1, closely followed by -2. Furthermore, in the triangular lattice the energy state of 0 is the 4^{th} most common energy state. This can be explained when we take into account that in the square lattice, we can get 2^{n-1} conformations with n beads where there cannot be any contacts by simply only going up or to the right. This is somewhat close to the observed limit near e^n total conformations with n amino acids, whereas in the triangular lattice, the limit is closer to 5^n so the 2^{n-1} in the triangular lattice is a significantly smaller proportion of the total number of conformations.

7.2 NNNpNpNpppNppNpN

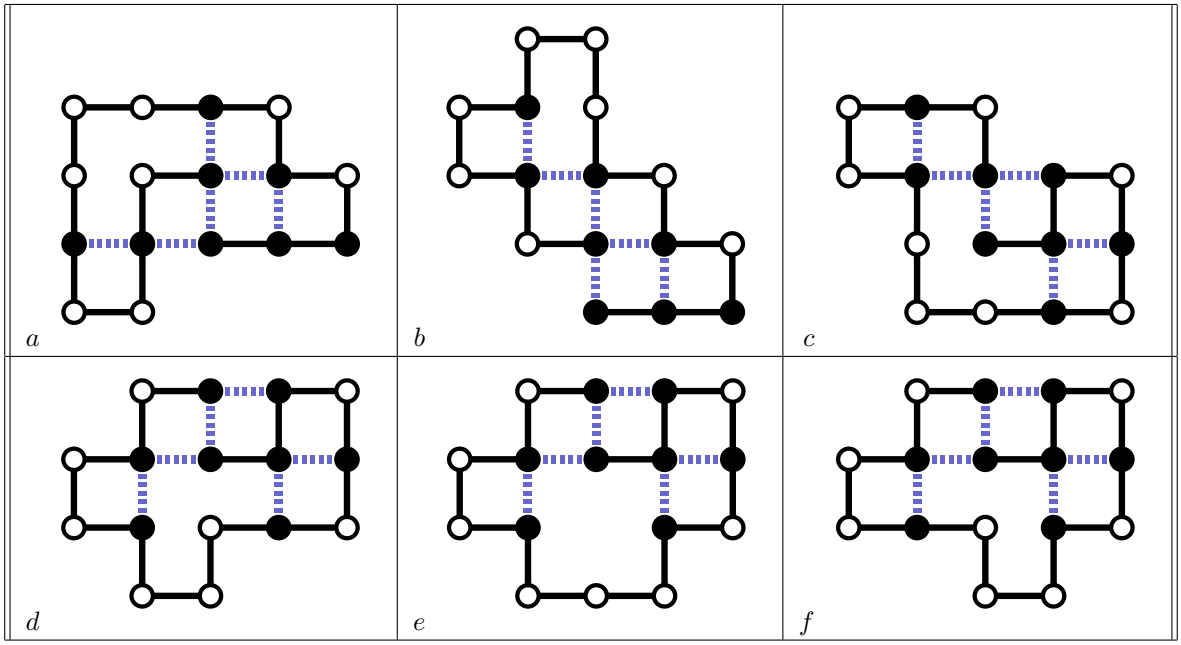


Figure 25: Conformations a, b, c, d, e, and f of NNNpNpNpppNppNpN

Studying the tables for the number of conformations of NNNpNpNpppNpp-NpN, we see that the minimum in the triangular lattice is once again roughly twice as low as that in the square lattice. We also note that the number of possible conformations in the triangular lattice is 3 orders of magnitude greater than the number of conformations in the square lattice. Noting how there are roughly half a billion conformations in the triangular lattice, even taking as little as a whole second to analyze each of the conformations would become a non-stop endeavor that lasts over 15 years, which is clearly out of the scope of this paper.

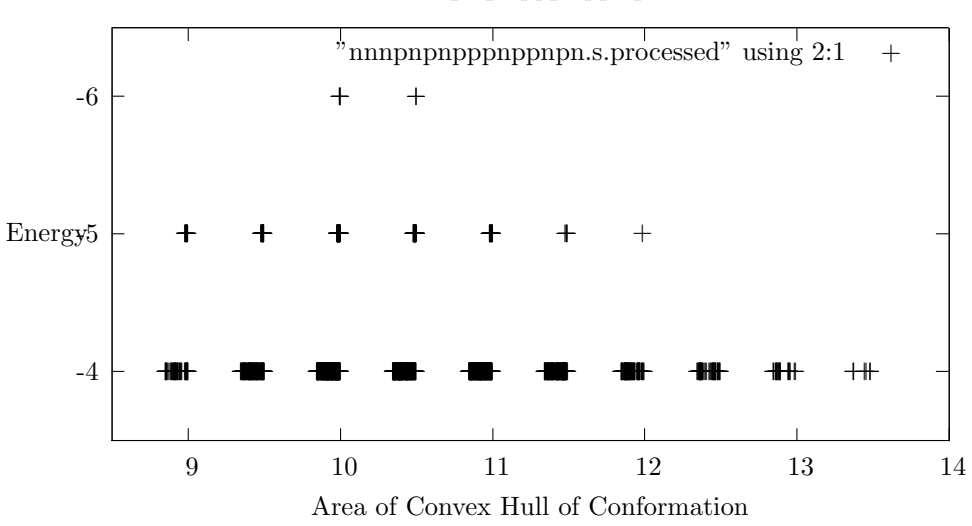
Table 3: Number of conformations with a given energy in the square lattice (Unique up to 8-fold symmetry)

0:	538,120	67.091%
-1:	211,279	26.342%
-2:	44,355	5.530%
-3:	7,093	.884%
-4:	1,092	.136%
-5:	130	.017%
-6:	6	.000%
SUM:	802,075	100.000%

Table 4: Number of conformations with a given energy in the triangular lattice (Unique up to 12-fold symmetry)

0:	58,310,910	12.102%
-1:	144,385,164	29.967%
-2:	148,055,634	30.729%
-3:	84,743,452	17.588%
-4:	32,860,115	6.820%
-5:	10,245,377	2.126%
-6:	2,579,916	.534%
-7:	503,102	.104%
-8:	106,898	.022%
-9:	19,718	.004%
-10:	2,884	.000%
-11:	596	.000%
-12:	33	.000%
SUM:	481,813,799	100.000%

Figure 26:



Also it is clear that computing the conformation web for a protein in the triangular lattice would be prohibitive. Assuming that a conformation of length 16 can, at any point, transform into roughly 100 other conformations, out of roughly 1 billion, we see that the conformation web would be a sparse graph and a list of nodes and their connections would be the ideal way to represent the graph. We get roughly a TB of storage required to store the web if we assume that a conformation of length 16 takes somewhere on the order of 10 bytes to represent. We can expect that a conformation of length 20 would take roughly a peta-byte of memory to represent - which is an absurd amount of storage. Clearly, we have to limit our explorations and representations of conformation webs to small subsets of conformation webs.

For example, it would not be unreasonable to explore the structure of the conformation web consisting of very low energy conformations to look for energy wells. The number of low energy conformations generally represent a miniscule proportion of the total number of possible conformations, which means we can cut many orders of magnitude off of the computation time or storage space needed to explore this subset of the conformation web. In our example, the proportion of conformations of energy less than or equal to -10 to the total number of conformations is 6.2×10^{-6} Exploring this subset of the energy web of this HP string or a similar HP string is tractable and a possible topic for further research.

Looking at the chart of the number of conformations in the triangular lattice, we can see that the bulk of the conformations are of low energy, which is not

surprising, but the expected value of the energy of a conformation randomly chosen from all of the possible valid conformations with equal probability is -1.9 in the case of the triangular lattice and -.4 in the case of the square lattice. We conjecture that the distribution of the energy of the conformations of the square lattice follows an exponential distribution while the triangular lattice follows a poisson distribution with a moderate value for λ

Unlike pNNpNpNpNNN, NNNpNpNpppNppNpN has 6 different minimal energy states (unique up to isometry) which are much more varied in structure. Conformation e is interesting because it has an empty point in the interior of the structure, however, $\{d, e, f\}$ forms a minimal energy well, because $d \leftrightarrow e$ and $e \leftrightarrow f$ by flipping an amino in or out of the hole. If the protein was surrounded by a sufficiently water-like solution to repel the hydrophobic aminos, e would be an undesirable conformation because the hole inside it exposes 3 hydrophobic aminos to solution.

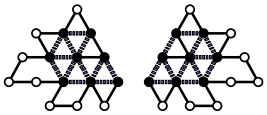


Figure 27: First minimal energy conformation well. Conformations 1 and 12

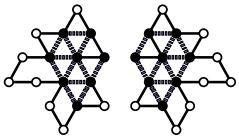


Figure 28: Second minimal energy conformation well. Conformations 2 and 11

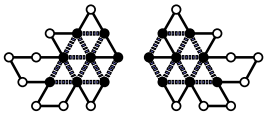


Figure 29: Third minimal energy conformation well. Conformations 3 and 10

When we compare the minimal energy square and triangular lattice conformations, we find that the triangular conformations have a 'maximally-dense' hydrophobic core, while the square lattice conformations do not. However, this is more of a property of the triangular lattice than a property of the protein sequence itself.

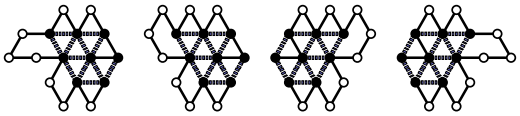


Figure 30: Fourth minimal energy conformation well. Conformations 4, 5, 8, and 9

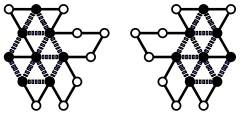


Figure 31: Fifth minimal energy conformation well. Conformations 6 and 7

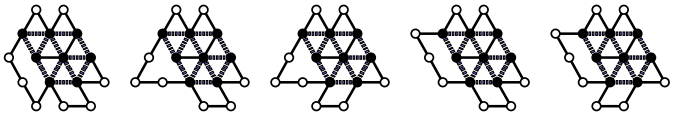


Figure 32: Sixth minimal energy conformation well. Conformations 13, 14, 15, 16, and 17

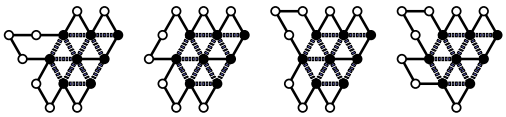


Figure 33: Seventh minimal energy conformation well. Conformations 18, 19, 20, and 21

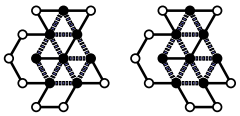


Figure 34: Eigth minimal energy conformation well. Conformations 22 and 23

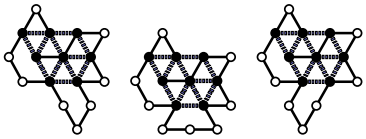


Figure 35: Ninth minimal energy conformation well. Conformations 24, 25, and 26

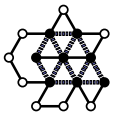


Figure 36: Tenth minimal energy conformation well. Conformation 27

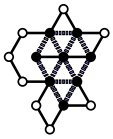


Figure 37: Eleventh minimal energy conformation well. Conformation 28

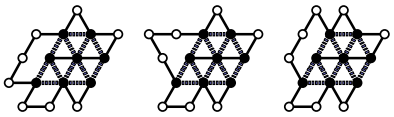


Figure 38: Twelvth minimal energy conformation well. Conformations 29, 30, and 31

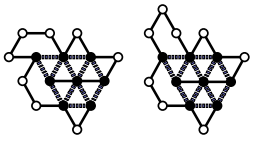


Figure 39: Thirteenth minimal energy conformation well. Conformations 32 and 33

8 Conclusions

We explored the HP model not only on the square lattice as originally proposed by Ken Dill, but we also used the triangular lattice.

We found upper and lower bounds on the number of self-avoiding walks in the square and triangular lattices ranging from trivial ones to more complicated bounds. In the square lattice, we got $O(b^n)$ for some b in $[1+\sqrt{2}, 3]$. We found lower bounds for the minimal energy of a bead sequence. We also examined different lattices and their properties. We counted the number of all self-avoiding walks of length up to 16 in the square and triangular lattices by exhaustively listing the valid self-avoiding walks for each length. We used these comprehensive lists of self-avoiding walks to thoroughly study two HP sequences, one of length 11, and the other of length 16. We studied these sequences in the square and triangular lattices.

The diameter of the convex hull of a conformation can be used as an estimate of the energy of the conformation. Further research in this matter could prove useful. Our examples demonstrated that the same holds true for the area of the convex hull. Both of these measures can be easily computed for a given conformation.

Our model and examples are in the Euclidean plane, but extensions to 3-space are easy and tractable. However, in 3-space self-avoiding walks can be knotted and new problems arise.

References

[1] Richa Agarwala, Serafim Batzoglou, Vlado Dančík, Scott E. Decatur, Martin Farach, Sridhar Hannenhalli, and Steven Skiena. Local rules for protein folding on a triangular lattice and generalized hydrophobicity in the hp model. *Journal of Computational Biology*, 4:275 – 296, 1997.

[2] Richa Agarwala, Serafim Batzoglou, Vlado Dančík, Scott E. Decatur, Martin Farach, Sridhar Hannenhalli, and Steven Skiena. Local rules for protein folding on a triangular lattice and generalized hydrophobicity in the HP model. In *Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms (New Orleans, LA, 1997)*, pages 390–399, New York, 1997. ACM.

[3] A. A. Albrecht, A. Skaliotis, and K. Steinhöfel. Stochastic protein folding simulation in the three-dimensional HP-model. *Comput. Biol. Chem.*, 32(4):248–255, 2008.

[4] Tom M. Apostol. Lattice points. *Cubo Mat. Educ.*, 2:157–173, 2000.

[5] Hue Sun Chan and Ken A. Dill. Protein folding in the landscape perspective: Chevron plots and non-arrhenius kinetics. *Proteins: Structure, Function, and Bioinformatics*, 30(1):2–33, 1998.

[6] Gary Chartrand, Linda Lesniak, and Ping Zhang. *Graphs & Digraphs Fifth Edition*. CRC Press, Florida, USA, 2011.

[7] Fabrizio Chiti and Christopher M. Dobson. Protein misfolding, functional amyloid, and human disease. *Annual Review of Biochemistry*, 75:333–366, 2006.

[8] Kuo-Chen Chou and Yu-Dong Cai. Predicting protein quaternary structure by pseudo amino acid composition. *Proteins: Structure, Function, and Bioinformatics*, 53(2):282–289, 2003.

[9] Nathan Clisby, Richard Liang, and Gordon Slade. Self-avoiding walk enumeration via the lace expansion. *Journal of Physics A: Mathematical and Theoretical*, 40(36):10973, 2007.

[10] Ken A. Dill, Sarina Bromberg, Kaizhi Yue, Hue Sun Chan, Klaus M. Ftebig, David P. Yee, and Paul D. Thomas. Principles of protein folding a perspective from simple exact models. *Protein Science*, 4(4):561–602, 1995.

[11] Ken A Dill, S Banu Ozkan, Thomas R Weikl, John D Chodera, and Vincent A Voelz. The protein folding problem: when will it be solved? *Current Opinion in Structural Biology*, 17(3):342 – 346, 2007. Nucleic acids / Sequences and topology.

[12] François Dubeau and Sébastien Labbé. A general form of Pick's theorem. *Int. J. Pure Appl. Math.*, 18(3):285–306, 2005.

[13] Bin Fu and Wei Wang. A $2^{O(n^{1-\frac{1}{d}}\log n)}$ time algorithm for d-dimensional protein folding in the HP-model. In *Automata, languages and programming*, volume 3142 of *Lecture Notes in Comput. Sci.*, pages 630–644. Springer, Berlin, 2004.

[14] John Harrison. A formal proof of pick's theorem. *Mathematical Structures in Computer Science*, 21(Special Issue 04):715–729, 2011.

[15] William E. Hart and Sorin C. Istrail. Fast protein folding in the hydrophobichydrophilic model within three-eighths of optimal. *Journal of Computational Biology*, 3:53–96, 1996.

[16] Tamjidul Hoque and Abdul Sattar. Extended hp model for protein structure prediction. *Journal of Computational Biology*, 16:85–103, Jan 2009.

[17] Sorin Istrail and Fumei Lam. Combinatorial algorithms for protein folding in lattice models: a survey of mathematical results. *Commun. Inf. Syst.*, 9(4):303–345, 2009.

[18] Hüseyin Kaya and Hue Sun Chan. Energetic components of cooperative protein folding. *Phys. Rev. Lett.*, 85:4823–4826, Nov 2000.

[19] Alireza Hadj Khodabakhshi, Ján Maňuch, Arash Rafiey, and Arvind Gupta. Inverse protein folding in 3D hexagonal prism lattice under HPC model. *J. Comput. Biol.*, 16(6):769–802, 2009.

[20] Kit Fun Lau and Ken A. Dill. A lattice statistical mechanics model of the conformational and sequence spaces of proteins. *Macromolecules*, 22(10):3986–3997, 1989.

[21] Milan Randic, Jure Zupan, Alexandru T. Balaban, Drazen Vikic-Topic, and Dejan Plavsic. Graphical representation of proteins. *Chemical Reviews*, 111(2):790–862, 2011.

[22] Rolf and Backofen. A polynomial time upper bound for the number of contacts in the hp-model on the face-centered-cubic lattice (fcc). *Journal of Discrete Algorithms*, 2(2):161–206, 2004. Combinatiorial Pattern Matching.

[23] Reinhard Schiemann, Michael Bachmann, and Wolfhard Janke. Exact enumeration of three-dimensional lattice proteins. *Computer Physics Communications*, 166(1):8–16, 2005.

[24] Diogo Stelle, Maria C. Barioni, and Luis P. Scott. Using data mining to identify structural rules in proteins. *Applied Mathematics and Computation*, 218(5):1997 – 2004, 2011.

[25] T. Wst and D.P. Landau. The hp model of protein folding: A challenging testing ground for wanglandau sampling. *Computer Physics Communications*, 179(1-3):124–127, 2008. Special issue based on the Conference on Computational Physics 2007.

[26] K Wthrich. Protein structure determination in solution by nmr spectroscopy. *Journal of Biological Chemistry*, 265(36):22059–62, 1990.