

# On the Complexity of Protein Folding (Extended Abstract)

Pierluigi Crescenzi \*   Deborah Goldman †   Christos Papadimitriou ‡   Antonio Piccolboni §  
Mihalis Yannakakis ¶

## Abstract

We show that the protein folding problem in the two-dimensional H-P model is NP-complete.

## 1 Introduction

### 1.1 Background

Proteins are polymer chains consisting of monomers of twenty different kinds—that is, strings over a 20-letter alphabet. Much of the genetic information in the DNA contains the sequence information of proteins, with three nucleotides (A-C-G-T symbols) encoding one monomer (member of the 20-letter alphabet). In turn, proteins in an organism fold, presumably by dint of the attraction or repulsion between monomers, to form a very specific geometric pattern, known as the protein's *native state*. It is this geometric pattern that determines the macroscopic properties, behavior, and function of a protein. It is in general reasonably stable and unique.

Understanding how the *genotype* (the genetic information of an organism) determines the *phenotype* (the macroscopic characteristics of the organism) is a problem at the

\*Dipartimento di Sistemi e Informatica, Università di Firenze, Via C. Lombroso 6/17, 50134 Firenze, Italy. E-mail: piluc@piluc.dsi.unifi.it, URL: <http://dsi2.dsi.unifi.it/piluc>. Research partially supported by the MURST project *Efficienza di Algoritmi e Progetto di Strutture Informative*.

†Department of Mathematics, University of California at Berkeley, Berkeley, CA 94720. Research supported by NDSEG Fellowship.

‡Computer Science Division, University of California at Berkeley, Berkeley, CA 94720. Research supported by NSF grant CCR96226361, and JSEP grant FDF 49620-97-1-0220-03-98.

§Università di Milano, Dipartimento di Scienze dell'Informazione, Via Comelico 39/41, 20135 Milano, Italy. Research partially supported by MURST project *Efficienza di Algoritmi e Progetto di Sistemi Informativi* and by CNR grant 97.02399.OT12.

¶Bell Laboratories, 700 Mountain Avenue, Murray Hill, NJ 07974.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

STOC '98 Dallas Texas USA

Copyright ACM 1998 0-89791-962-9/98/5...\$5.00

forefront of today's science (often referred to dramatically as "breaking the genetic code" or "the last phase of the Mendelian revolution"). This mapping can be roughly divided into three parts. Of these the first (the mapping from DNA sequences to monomer sequences) is simple and very well-understood. In contrast, the mapping from the sequence of a protein to the geometric configuration of its native state is much more intricate and complex, and less understood; it has been the subject of intense investigation for decades. (The third part, going from proteins to macroscopic characteristics, seems even more intractable.) It seems clear that the forces underlying protein folding are the interactions between their monomers; recently, the view that *non-local* interactions dominate this process has been gaining ground [5]. To test this and other hypotheses concerning protein folding, researchers resorted to *simplified models* of proteins, mathematical abstractions of proteins that hide many aspects and exaggerate the effect of others; analysis and computer simulation of such models can then be compared to experimental results with actual proteins, to determine whether the emphasized aspects are indeed the dominant ones.

Perhaps the most successful and best-studied such model, and the one with apparently the best match with experiments<sup>1</sup>, is the *two-dimensional hydrophilic-hydrophobic model*, or H-P model, proposed by Dill [4]. In this model it is assumed that the protein is a sequence of 0s and 1s, and folding entails embedding the sequence in the two-dimensional lattice (see Figure 1). Each such folding is evaluated with a *score*, equal to the number of pairs of 1s that are adjacent in the lattice without being adjacent in the sequence; for example, in Figure 1 the score is five, corresponding to the five pairs of 1s connected by dotted lines. The score captures a simple model of energy minimization, in which the "hydrophobic" 1s tend to be close to each other and thus avoid exposure, while 0s are neutral. It is assumed in this model that the native folded state is the one that maximizes score. It is therefore an interesting problem, given a sequence of 0s and 1s, to find the embedding on the lattice that maximizes score. *In this paper we prove that this problem is NP-complete* (Theorem 3).

There have been several NP-completeness results related to protein folding in the literature. A few years ago, several authors pointed out that certain general restatements of the

<sup>1</sup>Chan and Dill [3] state that "for chain lengths for which exhaustive enumeration is possible (up to about 30 monomers), two-dimensional models more accurately represent the physically important surface-interior ratios of proteins than do three-dimensional models."

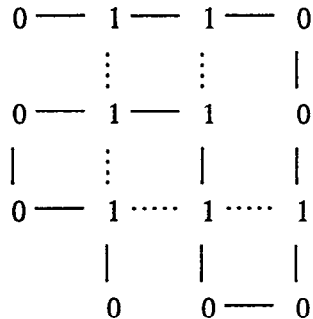


Figure 1: Embedding in the two-dimensional lattice.

problem, in which monomers attract or repel each other in ways that are general and can be used in encoding, are NP-complete [6, 11, 15]. More interestingly, it was proved in [12] that a combinatorial generalization of the H-P model to an infinite alphabet, of which one symbol is neutral like H-P’s 0 symbol, and the score counts the number of adjacencies of elements with the same symbol, is NP-complete. More recently, [10] improved this to a finite, albeit very large alphabet. The present result is the first to settle the complexity of the simple two-dimensional H-P model actually proposed in the literature as the ultimate simplification of the protein folding problem. The H-P problem has been attacked from the point of view of approximation algorithms [7]; the present result sheds little light on this aspect of the problem, as our reduction is not in any interesting way approximation-preserving. An interesting variant of the H-P model was introduced in [1], where proteins are embedded in a *hexagonal* lattice, devoid of the somewhat excessive dependence of the minimum-energy solution on *parity* properties of the string one sees in the ordinary H-P model. We do not at present know how to extend our NP-completeness result to this case.

## 1.2 The main result

Our reduction is from the Hamilton cycle problem. As is common in previous proofs of weaker results, we start by showing that the folding problem for *sets of sequences* (that is, when many sequences are to be optimally folded) is NP-complete (Theorem 1 in Section 2). We then proceed to establish the result for a single sequence, by resorting to certain interesting variants of the planar Hamilton cycle problem (Theorem 3 in Section 3). In our proof we utilize an idea of Trevisan [14], whereby graphs can be embedded in the hypercube so that adjacency is captured by Hamming distance.

Our proof captures one of the basic intuitions of the H-P model, namely that hydrophobic monomers will tend to form a large “sphere” (in the two-dimensional lattice, a large hydrophobic square). Impurities in this sphere then must be aligned optimally to maximize score, and it is the complexity of this alignment that our proof captures. Finally, a version of our proof (in fact, without the planarity complication) can be used to prove the NP-completeness of the three-dimensional version of the the protein folding problem in the H-P model—and in fact, the MAXSNP-completeness of the related problem of *minimizing losses* in three dimensions. Independently, and a few months prior to us, Berger and Leighton [2] proved that the three-dimensional protein folding problem in the H-P model is NP-complete; in fact, the approximability implications of their result for minimiz-

ing losses are stronger than ours.

## 1.3 On Levinthal’s paradox

That proteins fold so as to minimize energy has been accepted for decades. This view quickly leads to a puzzling aspect of the problem, known as *Levinthal’s paradox*, which can be paraphrased as follows “How can a folding protein choose so quickly among so many possible foldings the one with minimum energy?” [5]. Our result can thus be seen as a more compelling restatement of that paradox, since it implies that finding the optimum folding in the two-dimensional HP-model—the simplest abstraction of the protein folding problem one finds in the literature, and presumably a vast simplification of the true detailed 3-dimensional energy minimization problem in actual proteins—is NP-complete, that is to say, among the provably hardest problems of the sort alluded to by the paradox, in which we must optimize among an astronomical population of states.

We do not necessarily see this as a paradox. It is well-known in our field that NP-completeness is best used not as a terminal indicator of intractability, but as a tool for better defining the right problem. In particular, it seems more likely that Nature is not solving an NP-complete problem in its generality, but is focusing on a tractable special case. Furthermore, this special case may be the result not of a fortunate accident, but of deliberate (or, at any rate, systematic) development of specialized instances over millions of years. The robustness of the native state is particularly suggestive in this regard. It could be that the energy-minimization process involved in protein folding is a rather unsophisticated randomized hill-climbing, except that the underlying landscape of local optima is *very flat*, with a single local minimum with a very large region of attraction. This local minimum need not be the global optimum, just an overwhelmingly attractive local optimum.

The question remaining is, why should evolution favor such landscapes? The answer is that, in organisms, proteins function by collaborating with other proteins, with geometrically compatible (“snapping”) native states. Variability of local optima would imply frequent dysfunctionality, and is therefore evolutionarily undesirable. In recent computational experiments [13], it has been demonstrated that such evolutionary pressure does help develop flat landscapes in a variety of optimization settings. For example, evolving instances of the traveling salesman problem by rewarding uniformity of local optima quickly yields instances (such as cities along the perimeter of an approximately convex polygon) which indeed have very flat landscapes.

## 2 The multistring folding problem

The *two-dimensional lattice* is the graph,  $(Z^2, L)$ , with node set  $Z^2$  (all points in the Euclidean plane with integer coordinates), and edges all pairs in  $L = \{(x, y), (x', y')\} : |x - x'| + |y - y'| = 1\}$ . Consider a set of strings  $S = \{s_1, \dots, s_m\}$  from the alphabet  $\{0, 1\}$ . A *folding* of these strings is an embedding of  $S$  into the lattice, that is to say, a one-to-one mapping  $f$  from the set  $\{(i, j) : 1 \leq i \leq m, 1 \leq j \leq |s_i|\}$  to  $Z^2$  such that for all  $1 \leq i \leq m, 1 \leq j \leq |s_i| - 1$  we have  $(f(i, j), f(i, j + 1)) \in L$ . Fix a folding  $f$ ; the points  $f(i, j)$  and  $f(i, j + 1)$  are called *f-neighbors*. An edge of the lattice  $\{(x, y), (x', y')\} \in L$  is said to be a *loss* if (a) these points are not *f-neighbors*, and (b) exactly one of these two points is the image under  $f$  of a pair  $(i, j)$  such that the  $j$ th symbol of  $s_i$  is a 1. Each position in a string containing a one, and

which is not the first or the last, can participate in zero, one, or two losses.

The MULTISTRING FOLDING PROBLEM is the following: Given a set of strings  $s_1, \dots, s_m \in \{0, 1\}^*$  and an integer  $E$ , is there a folding with  $E$  or fewer losses? If, as is the case in the strings we construct, no string starts or ends in a 1, then it is easy to see that the total score of a folding is equal to twice the number of 1's, minus the losses, divided by two; hence, minimizing losses is the same as maximizing score; the traditional way of stating the protein folding problem.

In this section we prove the following theorem:

**Theorem 1** *The MULTISTRING FOLDING PROBLEM is NP-complete.*

In the next section we shall show that the problem remains NP-complete even if there is only one string.

## 2.1 Description of the reduction

We start from the following NP-complete problem: Given a graph  $G = (V, E)$  with nodes of degree four or less, and two nodes  $v_1, v_n \in V$ , is there a Hamilton path from  $v_1$  to  $v_n$ ?

As a preliminary step in our reduction, we first map the nodes in  $G$  to the hypercube according to a map used by Trevisan in [14]. Using Hadamard codes, he showed that there exists a function, which we call  $T$ , mapping the  $n$  nodes of the graph to codewords in  $\{0, 1\}^{8n}$  such that the images of two unconnected nodes have Hamming distance strictly greater than two nodes connected by an edge; in particular, applying  $T$  to the nodes of  $G$  we find, for  $i \neq j$  in  $\{1, 2, \dots, n\}$ :

1. If  $\{v_i, v_j\}$  is an edge in  $G$ , then  $d_H(T(v_i), T(v_j)) = \frac{7}{2}n$ .

2. If  $\{v_i, v_j\}$  is not an edge, then  $d_H(T(v_i), T(v_j)) = 4n$ .

(Here we assume that  $n$  is even.) Notice that if there is a Hamilton path from  $v_1$  to  $v_n$  in  $G$ , then there is a Hamilton path from  $T(v_1)$  to  $T(v_n)$  in the Hamming space of length  $\frac{7}{2}(n-1)n$ ; otherwise, if there is no Hamilton path from  $v_1$  to  $v_n$  in  $G$ , then any Hamilton path from  $T(v_1)$  to  $T(v_n)$  must have length strictly greater than  $\frac{7}{2}(n-1)n$ . We note, finally, that the function  $T$  may be chosen so that  $T(v_1)$  and  $T(v_n)$  contain at most as many zeros as  $T(v_i)$ , for any  $i \in \{1, \dots, n\}$ .

We now construct a set of strings  $S$  and an integer  $E$ , such that there is a Hamilton path from  $v_1$  to  $v_n$  in  $G$  if and only if there is a folding of the strings in  $S$  with at most  $E$  losses. The allowed number of losses is

$$E = 7(n-1)n.$$

As for the set of strings  $S$ , let  $L = 180n^{14}$ .  $S$  will contain  $L$  strings,  $s_1, \dots, s_L$ , such that the first  $L/n$  strings correspond to node  $v_1$ , the second  $L/n$  to node  $v_2$ , and so on. All strings, with the exception of  $s_1$  and  $s_L$ , will be constructed similarly and so that strings corresponding to the same city are identical. Define  $q = \lceil \sqrt{LE} \rceil$  and let  $c$  be an even positive integer to be specified later. Let  $d$  (suggesting *dense*) denote the string  $d = \prod_{i=1}^{L/90} 1^8 0$ , let  $m$  (suggesting *middle*) denote the string  $m = \prod_{i=1}^{2E-16n} 1^{cq} 0$ , and, for  $i \in [n]$ , let  $t(i)$  (suggesting the *Trevisan code*) denote the string  $t(i) = \prod_{i=1}^{8n} (1^{cq} T(v_i))_i^2$ . The description of the

strings  $s_2$  through  $s_{L-1}$  follows: for  $k \in \{2, \dots, L-1\}$  and  $i = \lceil \frac{nk}{L} \rceil$ ,

$$s_k = 0^{L^4} d_1^{L/10} d_1^{L/5-2E(cq+1)} m t(i) 1^{2L/5} d_0^{L^4}.$$

Notice that each string contains a prefix and suffix of  $L^4$  zeros. There are three dense substrings of ones and zeros in each string. Finally, toward the middle of the string, there is the substring  $mt(i)$  of length  $2E(c\lceil \sqrt{LE} \rceil + 1)$ , which we call the *sparse substring*, which consists of the sparse string  $m$  and the sparse string  $t(i)$  containing two copies of the Trevisan code (interspersed between strings of ones). The substring lying between the prefix and suffix, called the *internal substring*, has total length  $L$ .

The strings  $s_1$  and  $s_L$ , called the *flanks*, are identical, respectively, to strings  $s_2$  and  $s_{L-1}$  except for the fact that 4 zeros are inserted between every other pair of two adjacent ones in  $s_2$  and  $s_{L-1}$ , beginning with the first pair of ones in each maximal substring of adjacent ones. Notice that, by placing two copies of each bit of the Trevisan code next to each other (separated by ones), we have arranged for all maximal substrings of adjacent ones to have even length. Formally, setting  $d' = \prod_{i=1}^{L/90} (10^4 1)^4 0$ ,  $m' = \prod_{i=1}^{2E-16n} (10^4 1)^{cq/2} 0$ , and, for  $i \in [n]$ ,

$$t(i)' = \prod_{i=1}^{8n} \begin{cases} ((10^4 1)^{cq/2} T(v_i))_i^2 & T(v_i)_i = 0 \\ (10^4 1)^{cq/2} T(v_i)_i 0^4 (10^4 1)^{cq/2-1} 10^4 T(v_i)_i & T(v_i)_i = 1 \end{cases}$$

for each  $k \in \{1, L\}$  and  $i = \lceil \frac{nk}{L} \rceil$ ,

$$s_k = 0^{L^4} d' (10^4 1)^{L/20} d' (10^4 1)^{L/10-E(cq+1)} m' t(i)' (10^4 1)^{L/5} d' 0^{L^4}.$$

This completes the description of the reduction.

## 2.2 The intended folding

In this subsection we show that if there is a Hamilton path from  $v_1$  to  $v_n$  in  $G$ , then there is a folding with at most  $E$  losses. This is rather easy; the hard direction is the opposite, which is sketched in the next subsection.

Let  $(v_{i_1}, v_{i_2}, \dots, v_{i_n})$  be the order of the nodes contained in a Hamilton path, where  $v_{i_1} = v_1$  and  $v_{i_n} = v_n$ . We construct a folding, called the *intended folding*, by arranging the non-flank strings,  $s_2, \dots, s_{L-1}$ , vertically to form a  $(2L^4 + L) \times (L-2)$  rectangle as follows: all the strings corresponding to node  $v_{i_1}$  are placed adjacent with their first bits in the same horizontal line at the left side of the rectangle, then the strings corresponding to node  $v_{i_2}$  are placed next, and so on until the strings corresponding to node  $v_{i_n}$  complete the rectangle. The flank strings  $s_1$  and  $s_L$ , which differ only from  $s_2$  and  $s_{L-1}$  through the addition of groups of 4 zeros, also have vertical orientation except for the fact that the 4-zero groups are bent to the left and right, respectively. In this way, the flanks  $s_1$  and  $s_L$  can be placed, respectively, adjacent to the left and right sides of the rectangle so that their first bits are in line with the first bits of the other strings and so that, since the 4-zero groups have been excluded, the resulting patterns of bits adjacent to the rectangle are exactly the strings corresponding to nodes  $v_1$  and  $v_n$ . A schematic drawing of the intended folding appears in Figure 2.

Note that in the intended folding the central  $L \times L$  square is composed primarily of ones with some horizontal lines of zeros and horizontal lines containing code bits running through it. It should be clear that the only place where a

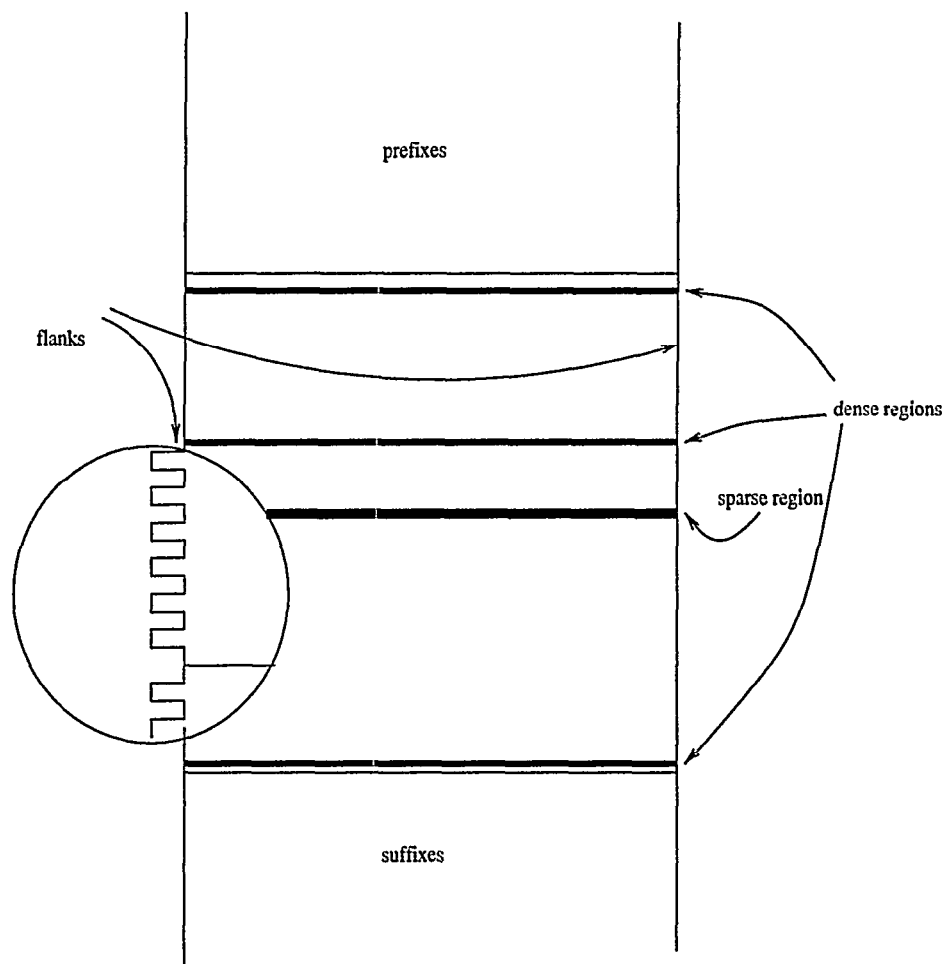


Figure 2: The intended folding.

loss may occur is where two code bits  $T(v_{i_j})_l$  and  $T(v_{i_{j+1}})_l$  from the Trevisan code corresponding to two different nodes  $v_{i_j}$  and  $v_{i_{j+1}}$  are adjacent. However, by the properties of the Trevisan code and because we have arranged the order of the identical groups of strings to be the same as the order  $(v_{i_1}, v_{i_2}, \dots, v_{i_n})$  of the Hamilton path, we are guaranteed that the two copies of the Trevisan code result in at most  $2\frac{1}{2}(n-1)n = E$  adjacent but not neighboring zero-one pairs. Therefore, the folding has at most  $E$  losses, and one direction has been proved.

### 2.3 The converse

In this section we summarize the (quite long and involved) proof of the converse; the full proof can be found at <http://http.cs.berkeley.edu/~christos/papers>. We consider a folding of the strings with at most  $E$  losses; we have to show that it is the intended folding corresponding to a Hamilton path of  $G$ .

We define the region  $R$  to consist of all points within the  $L \times L$  square of the intended folding, as well as all points surrounded by such points. We first prove that the largest component of this region has area at least  $L^2 - O(E)$  and perimeter at most  $4L + O(E)$ . Next, we show the smallest rectangle surrounding this component has sides of length  $L \pm O(\sqrt{LE})$ , and there is a square of sides  $L - O(\sqrt{LE})$  contained in  $R$ . We then consider a string passing through the center of the rectangle, and prove that it is "relatively straight," proceeding without too many bendings, from one end of the square to the opposite. We then prove that any string that passes through a narrow horizontal strip traverses the square from its top to the bottom side, and that in fact that almost all strings so traverse the square. It follows that the folding is the intended one, and corresponds to a Hamilton path in  $G$ .

### 3 The string folding problem

In this section we show that the STRING FOLDING PROBLEM (the special case of the multistring problem with  $|S| = 1$ , which captures the protein folding problem in the 2-dimensional H-P model) is also NP-complete.

Let us call a planar graph *special* if it consists of disjoint faces with nodes of degree three, connected together by paths of length two, and becomes triply connected if all nodes of degree two are collapsed. See Figure 3 for an example.

**Theorem 2** *The Hamilton cycle problem remains NP-complete even if restricted to special planar graphs.*

**Proof:** The reduction from exact cover to planar Hamilton cycle in [8] produces a special planar graph, if the 2-input and 3-input "or" gadgets are replaced by the ones shown in Figure 4. ■

Fix a planar graph  $G$ . Two Hamilton cycles are called *orthogonal* if they have the following property: Their disjoint union (where we duplicate edges in their intersection) is a degree-4 planar graph with multiple edges which can be embedded in the plane in such a way that the edges around each node alternate between the two cycles. Figure 5 depicts the two Hamilton cycles of the *diamond* graph (plus another node  $G$ ); they are orthogonal because, by duplicating the three paths of length two, one obtains a degree-four graph around each node of which edges of the two Hamilton cycles alternate.

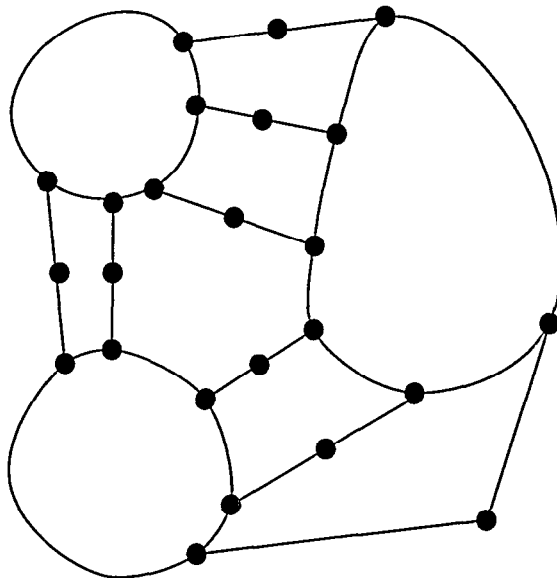


Figure 3: A special planar graph.

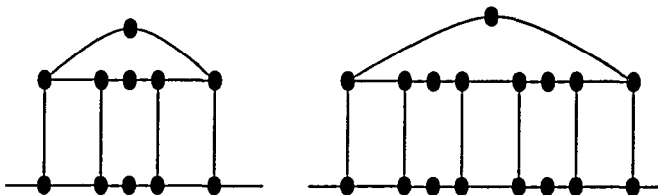


Figure 4: New 2-input and 3-input gadgets.

Suppose that a graph contains the *diamond* graph depicted in Figure 5 (ignore the node  $G$  standing for the rest of the graph). The diamond graph has four endpoints, denoted N, S, E, W, whereby it is connected to the rest of the graph. Any Hamilton cycle of the overall graph must traverse the diamond either from N to S, or from E to W (but not, e.g., from E to N).

**Theorem 3** *The STRING FOLDING PROBLEM is NP-complete.*

**Proof:** We start from the Hamilton cycle problem for special planar graphs. Given any special planar graph  $G$ , we shall modify the graph so that it contains a "standard" Hamilton cycle  $H_0$ , such that any Hamilton cycle of the original graph corresponds to a cycle of the modified graph that is orthogonal to  $H_0$ . Starting from  $G$  and its embedding, take only the degree-2 nodes of  $G$ , and consider two such nodes adjacent if they are on the same face of the embedding. Since the original graph is special, the resulting graph  $G'$  is connected.

Consider thus a cycle  $C$  of  $G'$  (allowing repeated nodes but no self-loops) that visits all nodes of  $G'$  at least once. If the two nodes adjacent to an occurrence of  $v$  are on the same face, repeat that occurrence twice. Now for each node  $v$ , count the occurrences of  $v$  on  $C$  and let  $a(v)$  be the resulting number.

Replace now each degree-2 node  $v$  of  $G$ , and its adjacent edges, by  $a(v)$  copies of the diamond; the copies are disjoint, and the N and S nodes of the two outermost ones (or the unique one, if  $a(v) = 1$ ) coincide with the nodes of  $G$  adjacent to  $v$ , see Figure 6. Let  $C = (v_0, v_1, \dots, v_k = v_0)$ . For

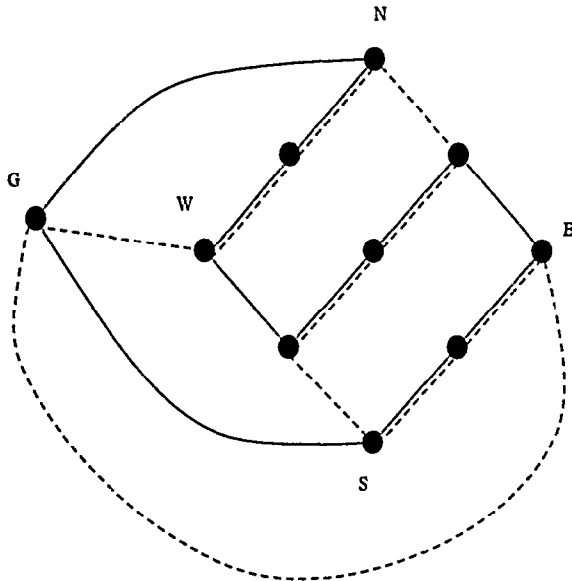


Figure 5: The diamond graph.

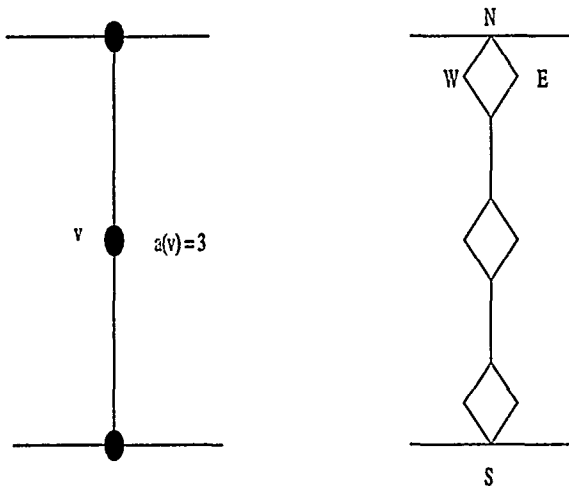


Figure 6: Replacing a degree-2 node by diamonds.

$i = 0, \dots, k - 1$ , suppose the  $i$ th element of  $G$  is the  $b_i$ th occurrence of node  $v_i$  (let  $b_k = 1$ ); for each  $i = 1, \dots, k$ , join the E or W node of the  $b_{i-1}$ th copy of the diamond replacing node  $v_{i-1}$ , whichever has not been considered up to now, with the E or W node of the  $b_i$ th copy of the diamond replacing node  $v_i$ , whichever is in the same face with the previous node (for  $v_0$ , if  $v_1 \neq v_0$ , we start with the endpoint, E or W, that is on the same face as  $v_1$ , and if  $v_1 = v_0$ , we start with the endpoint which is not on the same face as  $v_2$ ). Notice that these new edges do not harm the planarity of the graph, and they, together with the E-W traversal of the diamonds, form the standard Hamilton cycle,  $H_0$ , of the resulting graph  $G''$ .

$H_0$  is the only Hamilton cycle of  $G$  utilizing a E-W traversal of the diamonds. Any Hamilton cycle utilizing a N-S traversal must correspond to a Hamilton cycle of the original graph  $G$ . It is easy to see that any such cycle will be orthogonal to  $H_0$  —because the E-W and the N-S traversals of the diamond are orthogonal.

We shall now construct the instance of the string folding problem. We take any degree-2 node and replace it with two degree-1 nodes, and make these nodes the endpoints of the Hamilton path sought.  $H_0$  becomes a Hamilton path as well. We now perform Trevisan's transformation *having deleted the E-W edges* of the graph (that is, the endpoints of these edges have large Hamming distance in the Trevisan code). We then perform the multistring reduction, with the following modifications:

- The number of strings corresponding to each city,  $L/n$ , is odd. This is trivial to accomplish by adding one string to each set.
- All strings corresponding to the same city are connected together in one string, by ordering them arbitrarily, and connecting the end of the suffix of string  $2i - 1$  to the end of the suffix of the string  $2i$ , and the beginning of the prefix of string  $2i$  to the beginning of the prefix of string  $2i + 1$ ,  $i = 1, \dots, \frac{k-1}{2}$ .
- Finally, all of these  $n$  long strings are connected together in the order suggested by the Hamilton path  $H_0$  by long (of length  $2L^4 + 2L^2$ ) strings of zeros.

We claim that there is a folding with  $E$  losses if and only if the original special graph had a Hamilton cycle. Suppose that indeed there is a folding with  $E$  or fewer losses. By the precise same argument as in the proof of Theorem 1, there is a Hamilton path in the graph  $G''$  that does not utilize the E-W edges, and hence there is a Hamilton cycle in the original graph.

Conversely, suppose that  $G$  did have a Hamilton cycle. Then  $G''$  has a Hamilton cycle  $H$  other than  $H_0$ , and in fact one that is orthogonal to  $H_0$ . But this means that we can arrange the  $n$  strings corresponding to the cities as in the intended folding in the proof of Theorem 1, joined together as they are via their prefixes and suffixes in the order suggested by  $H_0$ , because  $H_0$  is orthogonal to  $H$ . ■

## References

- [1] R. Agarwala, S. Batzoglou, V. Dančík, S. E. Decatur, M. Farach, S. Hannenhalli, S. Muthukrishnan, S. Skiena. Local rules for protein folding on a triangular lattice and generalized hydrophobicity in the HP model. *J. Comp. Biol.* 4 (1997), pp. 275-296.
- [2] B. Berger and T. Leighton. Protein Folding in the Hydrophobic-Hydrophilic (HP) Model is NP-Complete. *J. of Computational Biology*, Spring (1998), vol. 5, no. 1, pp. 27-40.
- [3] H. S. Chan, K. A. Dill. The protein folding problem. *Physics Today* (1993), pp. 24-32.
- [4] K. A. Dill. Dominant forces in protein folding. *Biochemistry* 29 (1990), pp. 7133-7155.
- [5] K. A. Dill, S. Bromberg, K. Yue, K. Fiebig, D. P. Yee, P. D. Thomas, H. S. Chan. Principles of protein folding - A perspective from simple exact models. *Protein Science* 4 (1995), pp. 561-602.
- [6] A. S. Fraenkel. Complexity of protein folding. *Bulletin of Mathematical Biology* 55 (1993), pp. 1199-1210.
- [7] W. E. Hart, S. Istrail. Fast protein folding in the hydrophobic-hydrophilic model within three-eighths of optimal. *Proceedings of the 27th Annual ACM Symposium on the Theory of Computing*, 1995, pp. 157-168.

- [8] D. S. Johnson, C. H. Papadimitriou. "Computational Complexity," in *The Traveling Salesman Problem*, edited by E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, D. B. Shmoys, Wiley Interscience, 1985.
- [9] J. King. Deciphering the rules of protein folding. *Chemical Engineering News* 67 (1989), pp. 32-54.
- [10] A. Nayak, A. Sinclair, U. Zwick. Spatial Codes and the Hardness of String Folding. *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, to appear.
- [11] J. T. Ngo, J. Marks, M. Karplus. Computational complexity, protein structure prediction, and the Levinthal paradox. In *The protein folding problem and tertiary structure prediction*, edited by K. M. Merz and S. M. Le Grand, Birkhauser, Boston, 1994.
- [12] M. Paterson, T. Przytycka. On the complexity of string folding. *Discrete Applied Mathematics* 71 (1996), pp. 217-230.
- [13] C. H. Papadimitriou, M. Sideri The Evolution of Flat Landscapes. Manuscript, 1998.
- [14] L. Trevisan. When Hamming Meets Euclid: The Approximability of Geometric TSP and MST [Extended Abstract]. *Proceedings of the 29th Annual ACM Symposium on the Theory of Computing*, 1997, pp.21-29.
- [15] R. Unger, J. Moult. Finding the lowest free energy conformation of a protein is an NP-hard problem: proof and implications. *Bulletin of Mathematical Biology* 55 (1993), pp. 1183-1198.