Structure Prediction in an HP-type Lattice with an Extended Alphabet

Rolf Backofen and Sebastian Will
Institut für Informatik, LMU München
Oettingenstraße 67, D-80538 München

October 13, 1998

Abstract

The protein structure prediction problem is one of the most important problems in computational biology. This problem consists of finding the conformation of a protein (given by a sequence of amino-acids) with minimal energy. Because of the complexity of this problem, simplified models like Dill's HP-lattice model [15, 16] have become a major tool for investigating general properties of protein folding. Even for this simplified model, the structure prediction problem has been proven to be NP-complete [7, 5].

A disadvantage of the HP-problem is its high degeneracy. I.e., for every sequence there are a lot of conformations having the minimal energy. For this reason, extended alphabets have been used in the literature. One of these alphabets is the HPNX-alphabet [6], which considers hydrophobic amino acids as well as positive and negative charged ones.

In this paper, we describe an exact algorithm for solving the structure prediction problem for the HPNX-alphabet. To our knowledge, our algorithm is the first exact one for finding the minimal conformation of an lattice protein in a lattice model with an alphabet more complex than HP. We also compare our results with results as given for the HP-model.

1 Introduction

The protein structure prediction is one of the most important unsolved problems of computational biology. The problem can be specified as follows: Given a protein by its sequence of amino acids, what is its native structure? Many results in the past have shown the problem to be NP-hard. These results indicate that it is unlikely that one will find a general, efficient algorithm for solving this problem. But the situation is even worse, since one does not know the general principles why natural proteins fold into a native structure. E.g., these principles are interesting if one wants to design artificial proteins (for drug design). For the time being, one problem there is

that artificial proteins usually don't have a native structure (i.e., there is no stable structure that will be achieved by the protein).

To attack this problem, simplified models have been introduced, which became a major tool for investigating general properties of protein folding. An important class of simplified models are the so-called lattice models. The simplifications commonly used in this class of models are 1.) monomers (or residues) are represented using a unified size; 2.) bond length is unified; 3.) the positions of the monomers are restricted to positions ; and 4.) a simplified energy function.

There are different lattices. In principle, one can approximate real proteins arbitrarily close using sufficiently complex lattice models. The simplest used lattice is the cubic lattice, where every conformation of a lattice protein is a self-avoiding walk in $\mathbb{Z}^3$. A discussion of lattice proteins can be found in [8]. There is a bunch of groups working with lattice proteins. Examples of how lattice proteins can be used for predicting the native structure or for investigating principles of protein folding are [21, 1, 10, 20, 14, 11, 12, 2, 17].

An important representative of lattice models is the HP-model, which has been introduced by [15, 16]. In this model, the 20 letter alphabet of amino acids (and the corresponding manifoldness of forces between them) is reduced to a two letter alphabet, namely H and P. H represents *hydrophobic* amino acids, whereas P represent *polar* or hydrophilic amino acids. The energy function for the HP-model is given by the matrix as shown in Figure 1(a). It simply states that the energy contribution of a contact between two monomers is -1 if both are H-monomers, and 0 otherwise. Two monomers form a *contact* in some specific conformation if they are not connected via a bond, but occupy neighbouring positions in the conformation (i.e., the euclidian distance of the positions is 1). A conformation with *minimal energy* (in the following called *optimal conformation*) is just a conformation with the maximal number of contacts between H-monomers. Just recently, the structure prediction problem has been shown to be NP-complete even for the HP-model [5, 7].

(a)

	H	P
H	-1	0
P	0	0

(b)

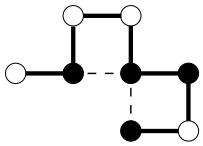


Figure 1: Energy matrix and sample conformation for the HP-model

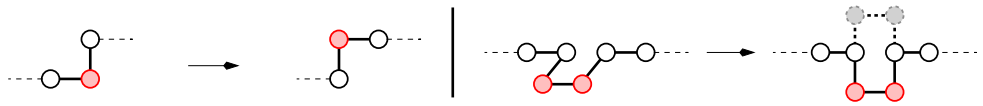
A sample conformation for the sequence PHPPHHPH in the two-dimensional lattice with energy -2 is shown in Figure 1(b). The white beads represent P, the black ones H monomers. The two contacts are indicated via dashed lines.

An example of the use of lattice models is the work by Šali, Shakhnovich and Karplus [21]. The same lattice model is used by several other people, e.g., [1, 20, 2, 12]. The authors investigate in [21] under which conditions a protein folds into its native structure. For this purpose, they have performed the following computer experiment on proteins in the cubic lattice:

1.) Generation of 200 random sequences of length 27.

2.) Determination of the minimal structures on the $3 \times 3 \times 3$-cube. The $3 \times 3 \times 3$-cube has exactly 27 position, which was the reason for using a sequence length of 27[1].

3.) simulation of protein folding on the lattice model using a Monte Carlo method with Metropolis criteria. The Monte Carlo method is as follows. Initially, a random conformation of the sequence is generated. Starting from this initial conformation, the algorithm performs so-called Monte Carlo steps in order to search for the minimal conformation. A single Monte Carlo step consists of the following operations: First, a local move is selected at random until a move is found that produces a valid conformation (i.e., a self-avoiding conformation). Two examples of allowed moves are



Here, the positions of the shaded monomers are changed, and the positions of the other monomers are kept unchanged.

Second, the resulting conformation is evaluated according to the Metropolis criterion. If the energy of the resulting conformation is lower than the energy of the previous one, then the conformation is always accepted. Otherwise, the conformation is accepted by random, where the probability of acception depends on the energy difference[2]. For every initial conformation, 50 000 000 Monte Carlo steps are performed.

Now a protein folds in that framework, if the Monte Carlo method finds its native conformation. The authors have found that a protein folds if there is a energy gap between the native structure and the energy of the next minimal structure.

In performing such experiments, it is clear that the quality of the predicted principle depends on several parameters. The first is the quality of the used lattice and energy function. The second, and even more crucial point, is the ability for finding the native structure as required by Step 2. For the energy function used by [21], there is no *exact* algorithm for finding the minimal structure. To be computational feasible, they have restricted in [21] the search for the native structure on the $3 \times 3 \times 3$-cube, as indicated in Step 2. But this approach has some drawbacks: 1.) The energy function had to be biased to a mean hydrophobicity in order to get proteins whose native structure is on the $3 \times 3 \times 3$-cube with high probability (see [21]); 2.) even then, it is not guaranteed that the minimal conformation is on this cube (for examples in the HP-model see [23]); 3.) the length of the proteins cannot be arbitrarily chosen.

Since there is an algorithm for finding the native structure on the HP-model, one could think of redoing the experiment within the HP-model. But the HP-model has

[1] In a later paper [10] the authors considered proteins of length 125.

[2] To be precise, the new conformation is accepted with a probability $e^{\frac{\triangle E}{k_b T}}$. k_b is the Boltzmann constant, and T is the folding temperature.

the problem that its degeneracy (i.e., the number of structures of a sequence that have minimal energy) is large [8, 23]. Hence, there is no dedicated native structure. But this implies that the HP-model is not suited for these experiments. For this reason, extended models such the HPNX-model [6] have been introduced.

2 Constraint Formulation

We start with the basic constraint formulation that underlies our search algorithm. Clearly, this basic formulation is not sufficient to yield an efficient search algorithm. But it shows how the constraint-based search can be used predicting a minimal energy structure of an HP (resp. HPNX) sequence. We then indicate which constraints have to be added and how to modify the search strategy in order to yield an efficient algorithm.

2.1 Basic Constraints and Search Algorithm

Our algorithm is based on constraint optimisation, which is the combination of two principles, namely generate-and-constraint with branch-and-bound. For using constraint optimisation, we have to transform the structure prediction problem into a constraint problem. A constraint problem consists of a set of variables together with some constraints (relations) on these variables.

For specifying the basic constraint problem, we need some definitions. We will describe the constraint formulation for the HP-model. The HPNX-model is an extension of the HP-model where the polar monomers are split into positively charged (P), negatively charged (N) and neutral (X) monomers. The energy function of the HPNX-model is given by the matrix

	H	P	N	X
H	-4	0	0	0
P	0	1	-1	0
N	0	-1	1	0
X	0	0	0	0

(1)

Since the basic constraint formulation is the same for the HP- and the HPNX-model, we will talk of polar monomers meaning P-monomers in the HP-model and PNX-monomers in the HPNX-model.

Let $s = s_1 \dots s_n$ be an HP- (or HPNX)-sequence of length n. A conformation c for this sequence is nothing else but a function $c : [1..n] \mapsto \mathbb{Z}^3$ assigning vectors to monomers such that

1. for all $1 \leq i < n$ we have $\|c(i) - c(i+1)\| = 1$ (i.e., every two successive monomers i and $i+1$ have distance 1)

2. and for all $i \neq j$ we have $c(i) \neq c(j)$ (the conformation c is self-avoiding).

Now we can encode the space of all possible conformations for a given sequence as a constraint problem as follows. We introduce for every monomer i new variables $\mathtt{X}_i$, $\mathtt{Y}_i$ and $\mathtt{Z}_i$, which denote the x-, y-, and z-coordinate of $c(i)$. Since we are using a cubic lattice, we know that this coordinates are all integers. But we can even restrict the possible values of these variables to the finite domain $[0..2n]$.[3] This is expressed by introducing the constraints

$$\mathtt{X}_i \in [1..(2 \cdot \text{length}(s))] \wedge \mathtt{Y}_i \in [1..(2 \cdot \text{length}(s))] \wedge \mathtt{Z}_i \in [1..(2 \cdot \text{length}(s))]$$

for every $1 \leq i \leq n$. The self-avoidingness is just $(\mathtt{X}_i, \mathtt{Y}_i, \mathtt{Z}_i) \neq (\mathtt{X}_j, \mathtt{Y}_j, \mathtt{Z}_j)$ for $i \neq j$.[4] Next we want to express that the distance between two successive monomers is 1, i.e.

$$\|(\mathtt{X}_i, \mathtt{Y}_i, \mathtt{Z}_i) - (\mathtt{X}_{i+1}, \mathtt{Y}_{i+1}, \mathtt{Z}_{i+1})\| = 1$$

Although this is some sort of constraint on the monomer position variables $\mathtt{X}_i, \mathtt{Y}_i, \mathtt{Z}_i$ and $\mathtt{X}_{i+1}, \mathtt{Y}_{i+1}, \mathtt{Z}_{i+1}$, this cannot be expressed directly in most constraint programming languages. Hence, we must introduce for every monomer i with $1 \leq i <$ length(s) three variables $\mathtt{Xdiff}_i$, $\mathtt{Ydiff}_i$ and $\mathtt{Zdiff}_i$. These variables have values 0 or 1. Then we can express the unit-vector distance constraint by

$$\begin{array}{lll} \mathtt{Xdiff}_i & = & |\mathtt{X}_i - \mathtt{X}_{i+1}| \qquad \mathtt{Zdiff}_i \;=\; |\mathtt{Z}_i - \mathtt{Z}_{i+1}| \\ \mathtt{Ydiff}_i & = & |\mathtt{Y}_i - \mathtt{Y}_{i+1}| \qquad 1 \;=\; \mathtt{Xdiff}_i + \mathtt{Ydiff}_i + \mathtt{Zdiff}_i. \end{array}$$

The constraints described above span the space of all possible conformations. I.e., every valuation of $\mathtt{X}_i, \mathtt{Y}_i, \mathtt{Z}_i$ satisfying the constraints introduced above is an *admissible* conformation for the sequence s, i.e. a self-avoiding walk of s. Given partial information about $\mathtt{X}_i, \mathtt{Y}_i, \mathtt{Z}_i$ (expressed by additional constraints as introduced by the search algorithm) we call a conformation c *compatible* with these constraints on $\mathtt{X}_i, \mathtt{Y}_i, \mathtt{Z}_i$ if c is admissible and c satisfies the additional constraints.

But in order to use constraint optimisation, we have to encode the energy function. For HP-type models, the energy function can be calculated if we know for every pair of monomers (i, j) whether i and j form a contact. i and j form a *contact* in a conformation c, if $j \notin \{i-1, i, i+1\}$ and

$$||c(i) - c(j)|| = 1.$$

For this purpose we introduce for every pair (i, j) of monomers with $i + 1 < j$ a variable $\mathtt{Contact}_{i,j}$. $\mathtt{Contact}_{i,j}$ is 1 if i and j have a contact in every conformation

[3] We even could have used $[1..n]$. But the domain $[0..2n]$ is more flexible since we can assign an arbitrary monomer the vector (n, n, n), and still have the possibility to represent all possible conformations.

[4] This cannot be directly encoded in Oz [18], but we reduce these constraints to difference constraints on integers.

which is compatible with the valuations of $\mathtt{X}_i, \mathtt{Y}_i, \mathtt{Z}_i$, and 0 otherwise. Then we can express this property in constraint programming as follows:

$$\begin{aligned} \mathtt{Xdiff}_{i,j} &= |\mathtt{X}_i - \mathtt{X}_j| \qquad \mathtt{Zdiff}_{i,j} = |\mathtt{Z}_i - \mathtt{Z}_j| \\ \mathtt{Ydiff}_{i,j} &= |\mathtt{Y}_i - \mathtt{Y}_j| \qquad \mathtt{Contact}_{i,j} \in \{0,1\} \\ (\mathtt{Contact}_{i,j} = 1) &\Leftrightarrow (\mathtt{Xdiff}_i + \mathtt{Ydiff}_i + \mathtt{Zdiff}_i = 1) \end{aligned} \tag{2}$$

where $\mathtt{Xdiff}_{i,j}$...$\mathtt{Zdiff}_{i,j}$ are new variables. The constraint (2) is called a reified constraint, and can be directly encode in Oz [18].

Using the variables $\mathtt{Contact}_{i,j}$, we can now easily encode the energy function for HP-type models. This means that we can now define a variable `Energy` which is subject to constraint optimisation. For the HP-model, we get the constraint

$$\mathtt{Energy} = \sum_{\mathtt{i+1<j} \wedge \mathtt{s(i)=H} \wedge \mathtt{s(j)=H}} -\mathtt{Contact}_{\mathtt{i,j}}.$$

For the HPNX-model, the corresponding constraint can be generated analogously using the energy matrix given in (1).

Thus, we have encoded self-avoiding walks together with a variable `Energy`. Now we can describe the search procedure, which is a combination of generate-and-constraint and branch-and-bound. In a generate step, a undetermined variable *var* out of the set of variables $\{\mathtt{X}_i, \mathtt{Y}_i, \mathtt{Z}_i \mid 1 \leq i \leq n\}$ is selected (according to some strategy). A variable is *determined* if its associated domain consists of only one value, and *undetermined* otherwise. Then, a value *val* out of the associated domain is selected and the variable is set to this value in the first branch (i.e., the constraint $var = val$ is inserted), and the search algorithm is called recursively. In the second branch, which is visited after the first branch is completed, the constraint $var \neq val$ is added.

Each insertion of a constraint leads through constraint propagation to narrowing of some (or many) domains of variables or even to failure, which both prune the search tree by removing inconsistent alternatives. Thus the search is done by alternating constraint propagation and branching with constraint insertion. The generate-and-constraint steps are iterated until all variables are determined (which implies, that a valid conformation is found). If we have found a valid conformation c, then the constraints will guarantee that `Energy` is determined. Let E_c be associated value of `Energy`. Then the additional constraint

$$\mathtt{Energy} < E_c \tag{3}$$

is added, and the search is continued in order to find the next best conformation, which must have a smaller energy than the previous ones due to the constraint (3). This implies that the algorithm finally finds a conformation with minimal energy.

At every node n of the search tree, we call the set of constraints introduced by the search algorithm so far the *configuration* at node n. Every conformation that is found below node n in the search tree must be compatible with the configuration

`Caveats`	Boolean; is 0 if the conformation contains no caveats
`Frx, Fry, Frz`	dimensions of the frame;
$\texttt{E}_j\texttt{.seh}$, $\texttt{E}_j\texttt{.soh}$	number of even and odd H-monomers of the j^{th} x-plane in the frame, respectively (where $1 \leq j \leq \texttt{Frx}$)
$\texttt{Elem}_j^i$	membership of monomer i in the j^{th} x-layer; the constraint $\texttt{Elem}_j^i$ will be defined only if i is an H-monomer
$\texttt{P}_k\texttt{.ctp}$	type of the k^{th} position of the frame (where $1 \leq k \leq \texttt{Frx} \cdot \texttt{Fry} \cdot \texttt{Frz}$); the core type $\texttt{P}_k\texttt{.ctp}$ of the k^{th} position is 1, if it is occupied by an H-monomer, and 0 otherwise
$\texttt{O}_i^k$	for every position k of the frame and every monomer i; $\texttt{O}_i^k$ has boolean value (i.e., 0 or 1), and is 1 iff monomer i occupies the k^{th} position of the frame

Figure 2: Some variables and their description.

at n, and vice versa. A *bounding function for* `Energy` is a function that takes a configuration of some node n, and yields some value E, where every conformation compatible with the configuration of n has an energy greater than E.

2.2 Auxiliary Variables and Search Strategy

Clearly, the above described constraint problem generated from a sequence s is not sufficient to yield an efficient implementation. For efficiency, one needs 1.) effective bounding functions; 2.) the ability for implementing a search strategy that tends to enumerate low energy conformations first.

We will illustrate this using the concept of the H-frame of an HP-model conformation. Given a conformation c, the H-frame is the minimal cube that contains all H-monomers of c. Now an optimal conformation has a maximal compact H-core. But a conformation with maximal compact H-core has a minimal H-core surface. Since the H-frame yields a lower bound on the surface of the H-core, we finally get a bounding function for the energy [22]. If one introduces variables $\texttt{Frx}, \texttt{Fry}, \texttt{Frz}$ for the dimensions of the frame, we are able to enumerate the $\texttt{Frx}, \texttt{Fry}, \texttt{Frz}$-variables before the variables $\texttt{X}_i, \texttt{Y}_i, \texttt{Z}_i$. This allows to apply the bounding function derived from the H-frame dimensions $\texttt{Frx}, \texttt{Fry}, \texttt{Frz}$ early in the search tree. Furthermore, this supports implementing a search strategy that prefers low energy conformation by enumerating H-frames with minimal dimensions first[5].

There are several further auxiliary variables that have been introduced for this reason. We have listed some of them together with the corresponding, more complex search strategy in Figures 2 and 3. A detailed description of the variables and search strategy, and improved bounding functions are given in [4].

[5]Clearly, the H-frame dimensions $\texttt{Frx}, \texttt{Fry}, \texttt{Frz}$ must satisfy that $\texttt{Frx} \cdot \texttt{Fry} \cdot \texttt{Frz} >$ # H-monomers in s

$$\texttt{Caveats} \;<\; \begin{matrix}\texttt{Frx}\\ \texttt{Fry}\\ \texttt{Frz}\end{matrix} \;<\; \begin{matrix}\texttt{E}_j\texttt{.seh}\\ \texttt{E}_j\texttt{.soh}\end{matrix} \;<\; \texttt{Elem}_j^i \;<\; \texttt{O}_i^k \;<\; \begin{matrix}\texttt{X}_i\\ \texttt{Y}_i\\ \texttt{Z}_i\end{matrix}$$

Figure 3: Search Strategy. Variables are selected according to the displayed order.

2.3 HPNX Extensions

As a benefit of our constraint programming approach it is possible to extend the HP algorithm to find the native structure of HPNX proteins.

Since one can see the HP model as embedded into HPNX, resp. HPNX as an extension to HP, a first naive approach to do such an extension is as follows. First, search all HP-optimal conformations of an HPNX sequence, i.e. the conformations that have maximal H-H-contacts. Then, second, find in the set of the HP-optimal conformations the ones with optimal HPNX-energy. This approach is certainly inefficient, since one has a lot of search steps because of the high degeneracy of the HP-model. But further it yields only those native HPNX-conformations that are also optimal in HP, but this is not necessarily true.

Our approach starts by updating the energy constraint. Now we get

$$\texttt{Energy} \;=\; -4 \cdot \mathit{HH\_Contacts} - \mathit{PN\_Contacts} + \mathit{PP\_Contacts} + \mathit{NN\_Contacts},$$

where HH_Contacts, PP_Contacts resp. NN_Contacts is the number of contacts between H, P resp. N monomers and PN_Contacts the number of contacts between P and N monomers.

To get an efficient implementation, we additionally need a good lower bound on the PN-energy, i.e., $-\mathit{PN\_Contacts}+\mathit{PP\_Contacts}+\mathit{NN\_Contacts}$. If the H-frame of the conformation is already fixed, we get a good lower bound by introducing the concept of compartments, which we will define now.

Given two points (x, y, z) and (x', y', z') in $\mathbb{Z}^3$, we define the distance function $D((x, y, z), (x', y', z'))$[6] by

$$D((x, y, z), (x', y', z')) = |x - x'| + |y - y'| + |z - z'|.$$

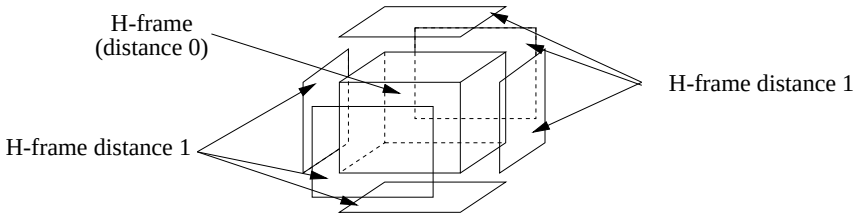
Let P_f denote the set of all points contained in a given H-frame f. Then we define the *H-frame distance* D_f of a point (x, y, z) by

$$D_f(x, y, z) = \min\{D((x, y, z), (x', y', z')) \mid (x', y', z') \in P_f\}.$$

Fix an H-frame f. We define a *compartment C with H-frame distance d* as a maximal, connected set of points, where all points have the same H-frame distance d. Note, that according to our definition there is a single compartment with H-frame distance

[6] D is the distance function corresponding to the 1-norm.

0, which is just the H-frame. Higher order compartments are placed around the H-frame as planes, lines and points. The compartments with H-frame distance 0 and 1 are as follows:



Now this concept helps us pruning the search tree in two ways. First, not every polar monomer i can be member of any compartment C. Instead, there is a restriction which depends on the H-frame distance of C and the position of i in the sequence. Second, we have found an appropriate lower bound on the PN-energy provided that membership of polar monomers to compartments is determined. We have to skip this part here due to space restrictions.

Lemma 2.1 *Let s be an HPNX-sequence of length n and i be a monomer of type P,N or X. We define* leftdist(i,s) *to be the minimal distance of i to an H-monomer j in s with $j < i$ if exists, and ∞ otherwise. Analogously we define* rightdist(i,s)*. Then i can be a member of compartment C with H-frame distance d, iff* $\text{leftdist}(i,s) \geq d \wedge \text{rightdist}(i,s) \geq d+1$, *or* $\text{leftdist}(i,s) \geq d+1 \wedge \text{rightdist}(i,s) \geq d$.

This can be directly encoded using constraints. The effect is that fixing the position of some H-monomers will determine the membership of PNX-monomers to compartments early in the search tree. This allows to apply our lower bound.

3 Results

S1hp	HPPPPHHHHPPHPHPHHHPHPPHHPPH	S3hp	HPHHPPHHPPHHHHPPPHPPPHHHPPH
S1	HXXNNHHHHXPHXHXHHHNHPPHHXPH	S3	HPHHNXHHPNHHHHXXXHXPXHHHPXH
S2hp	HPPPHHHHPHPHHPPPHPHHPHPPPHP	S4hp	HHPHHPHHPHHHHHHPPHHHHHPPHHHHHHH
S2	HXXXHHHHNHXHHXXNHPHHPHXNXHP	S4	HHXHHPHHXHHHHHHPPHHHHHXNHHHHHHH
S2.1	HXNNHHHHXHXHHNXNHXHHNHPPXHP	S5hp	PHPPHPPHPPHPPHPPHPPHPPHPPHPPHPPHPPHP
S2.2	HXXNHHHHXHPHHPXXHXHHXHXNNHX	S5	XHXNHXXHNPHXXHPXHXXHXNHPXHNXHPXHXPHX

Table 1: Test sequences

We investigate a set of test sequences as shown in Table 1. Here, we have grouped the sequences, such that for every group i there is a HP-sequence S$i$hp, from which the other sequences are generated by replacing P monomers by P, N and X monomers. We will call the S$i$hp the generating HP-sequence of the HPNX-sequences in i.

sequence	#steps find	# steps prove	sequence	#steps find	# steps prove
S1	3387	10237	S2	47	1879
S2.1	211	2089	S2.2	39	1963
S3	25	4879	S4	156	156
S5	507	18103			

Table 2: Search steps for sample sequences

The algorithm finds the native structure of all sequences listed in Table 1. Note that there is a difference between finding the native structure, and proving that the best found structure is really the optimal one (which requires that the complete search space has been investigated). Hence, we display in Table 2 the search steps needed to find the native conformation (# steps find), and the number of steps needed to show that the best found conformation is really optimal (# steps prove). In Table 3, we have compared the degeneracy of the HPNX-sequences with the corresponding HP-sequences. One can find that the degeneracy is strongly reduced in the HPNX-model.

sequence	HPNX degeneracy	HP degeneracy
S2	2	297
S2.1	4	297
S2.2	51	297
S4	51	1114
S5	16	3538

Table 3: Comparison of the degeneracy in the HPNX- and HP-model for some sample sequences as found by our algorithm. For the HP-sequences (second column), the same level of degeneracy was found in [22].

Acknowledgement We would like to thank Prof. Peter Clote, who got us interested in bioinformatics and inspired this research. The first author would like to thank Prof. Martin Karplus for helpful discussions on the topic of lattice models, and for motivating him to apply constraint programming techniques to lattice protein folding. The first author would also like to thank Dr. Erich Bornberg-Bauer, who initiated this research, too. He also explained the biological background, and gave many helpful hints.

References

[1] V. I. Abkevich, A. M. Gutin, and E. I. Shakhnovich. Impact of local and nonlocal interactions on thermodynamics and kinetics of protein folding. *Journal of Molecular Biology*, 252:460–471, 1995.

[2] V.I. Abkevich, A.M. Gutin, and E.I. Shakhnovich. Computer simulations of

prebiotic evolution. In Russ B. Altman, A. Keith Dunker, Lawrence Hunter, and Teri E. Klein, editors, *PSB'97*, pages 27–38, 1997.

[3] Rolf Backofen. Using constraint programming for lattice protein folding. In *Pacific Symposium on Biocomputing (PSB '98)*, volume 3, pages 387–398, 1998.

[4] Rolf Backofen. The Protein Structure Prediction Problem: A Constraint Optimisation Approach using a New Lower Bound. submitted.

[5] B. Berger and T. Leighton. Protein folding in the hydrophobic-hydrophilic (HP) modell is NP-complete. In *Proc. of the RECOMB'98*, pages 30–39, 1998.

[6] Erich Bornberg-Bauer. Chain growth algorithms for HP-type lattice proteins. In *Proc. of the* 1^{st} *Annual International Conference on Computational Molecular Biology (RECOMB)*, pages 47 – 55. ACM Press, 1997.

[7] P. Crescenzi, D. Goldman, C. Papadimitriou, A. Piccolboni, and M. Yannakakis. On the complexity of protein folding. In *Proc. of STOC*, 1998. To appear. Short version in *Proc. of RECOMB'98*, pages 61–62.

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

[9] Ken A. Dill, Klaus M. Fiebig, and Hue Sun Chan. Cooperativity in protein-folding kinetics. *Proc. Natl. Acad. Sci. USA*, 90:1942 – 1946, 1993.

[10] Aaron R. Dinner, Andreaj Šali, and Martin Karplus. The folding mechanism of larger model proteins: Role of native structure. *Proc. Natl. Acad. Sci. USA*, 93:8356–8361, 1996.

[11] S. Govindarajan and R. A. Goldstein. Evolution of model proteins on a foldability landscape. *Proteins*, 29(4):461–466, 1997.

[12] S. Govindarajan and R. A. Goldstein. The foldability landscape of model proteins. *Biopolymers*, 42(4):427–438, 1997.

[13] William E. Hart and Sorin C. Istrail. Fast protein folding in the hydrophobid-hydrophilic model within three-eighths of optimal. *Journal of Computational Biology*, 3(1):53 – 96, 1996.

[14] David A. Hinds and Michael Levitt. From structure to sequence and back again. *Journal of Molecular Biology*, 258:201–209, 1996.

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

[16] Kit Fun Lau and Ken A. Dill. Theory for protein mutability and biogenesis. *Proc. Natl. Acad. Sci. USA*, 87:638 – 642, 1990.

[17] Angel R. Ortiz, Andrzej Kolinski, and Jeffrey Skolnick. Combined multiple sequence reduced protein model approach to predict the tertiary structure of small proteins. In Russ B. Altman, A. Keith Dunker, Lawrence Hunter, and Teri E. Klein, editors, *PSB'98*, volume 3, pages 375–386.

[18] Gert Smolka. The Oz programming model. In Jan van Leeuwen, editor, *Computer Science Today*, Lecture Notes in Computer Science, vol. 1000, pages 324–343. Springer-Verlag, Berlin, 1995.

[19] R. Unger and J. Moult. Genetic algorithms for protein folding simulations.

Journal of Molecular Biology, 231:75–81, 1993.

[20] Ron Unger and John Moult. Local interactions dominate folding in a simple protein model. *Journal of Molecular Biology*, 259:988–994, 1996.

[21] A. Šali, E. Shakhnovich, and M. Karplus. Kinetics of protein folding. *Journal of Molecular Biology*, 235:1614–1636, 1994.

[22] Kaizhi Yue and Ken A. Dill. Sequence-structure relationships in proteins and copolymers. *Physical Review E*, 48(3):2267–2278, September 1993.

[23] Kaizhi Yue, M. Fiebig, Poaul D. Thomas, Hue Sun Chan, Eugene I. Shakhnovich, and Ken A. Dill. A test of lattice protein folding algorithms. *Proc. Natl. Acad. Sci. USA*, 92:325 – 329, 1995.