CONSTRAINT-BASED HYDROPHOBIC CORE CONSTRUCTION FOR PROTEIN STRUCTURE PREDICTION IN THE FACE-CENTERED-CUBIC LATTICE

SEBASTIAN WILL[a]
Institut für Informatik, Ludwig-Maximilians-Universität München,
Oettingenstraße 67, D-80538 München, Germany
wills@informatik.uni-muenchen.de

We present an algorithm for exact protein structure prediction in the FCC-HP-model. This model is a lattice protein model on the face-centered-cubic lattice that models the main force of protein folding, namely the hydrophobic force. The structure prediction for this model can be based on the construction of hydrophobic cores. The main focus of the paper is on an algorithm for constructing maximally and submaximally compact hydrophobic cores of a given size. This algorithm treats core construction as a constraint satisfaction problem (CSP), and the paper describes its constraint model. The algorithm employs symmetry excluding constraint-based search [6] and relies heavily on good upper bounds on the number of contacts. Here, we use and strengthen upper bounds presented earlier.[8] The resulting structure prediction algorithm (including previous work [8,7]) handles sequences of sizes in the range of real proteins fast, i.e. we predict a first structure often within a few minutes. The algorithm is the first exact one for the FCC, besides full enumeration which is impracticable for chain lengths greater than about 15. We tested the algorithm succesfully up to sequence length of 160, which is far beyond the capabilities even of previous heuristic approaches.

1 Introduction

Protein structure prediction is one of the most important unsolved problems of computational biology. It can be specified as follows: Given a protein by its sequence of amino acids (more generally monomers), what is its native structure? NP-completeness of the problem has been proven for many different models, among them lattice and off-lattice models.[10,12]

To tackle structure prediction and related problems, simplified models have been introduced. For this aim, they are used in hierarchical approaches for protein folding.[25] Here, see also the meeting review of CASP3,[17] where some groups have used lattice models. Furthermore, simplified models are a major tool for investigating general properties of protein folding.

Most important are the so-called lattice models, where protein structure is modeled as a self-avoiding walk on a lattice. In the literature, many different lattice models (each specified by a lattice and an energy function) have been

[a] Supported by the PhD programme GKLI of the "Deutsche Forschungsgemeinschaft".

used. It was shown how such models can be used for predicting the native structure or for investigating principles of protein folding.[24,1,15,23,16,2,19,25]

Of course, the question arises which lattice and energy functions should be preferred. There are two aspects that have to be evaluated when choosing a model: 1) the accuracy of the lattice in approximating real protein conformations (aka structures), and the ability of the energy function to discriminate native from non-native conformations, and 2) the availability and quality of search algorithms for finding minimal (or nearly minimal) energy conformations. Obviously, the two aspects are somewhat conflicting. While the first aspect is well-investigated in the literature [20,13] the second aspect is neglected.

In this paper, we follow the proposal of Agarwala et.al.[3] to use a lattice model with a simple energy function, namely the HP (hydrophobic-polar) model (which has been introduced by Lau and Dill [18] using cubic lattice), but on a better suited lattice (namely the face-centered cubic one). The resulting model is called the *FCC-HP-model*. In the HP-model, the 20 letter alphabet of amino acids is reduced to a two letter alphabet $\{H, P\}$. H represents *hydrophobic* amino acids, whereas P represent *polar* or hydrophilic amino acids. The energy function for the HP-model 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 occupy positions of minimal distance. A conformation with *minimal energy* (called *native conformation*) is just a conformation with the maximal number of contacts between H-monomers. Even for the HP-model, the structure prediction problem was shown as NP-complete.[10,12]

There are two reasons for using the FCC-HP-Model: First, the FCC can model real protein conformations with good quality (up to coordinate root mean square deviation below 2 Å).[20] Second, the HP-model models the aspect of hydrophobicity. Its energy function enforces compactification due to the hydrophobic force, while polar residues and solvent molecules are not explicitly regarded. Hydrophobicity is very important, since one assumes that the hydrophobic effect determines the overall configuration of a protein.[18,13]

Once a search algorithm for minimal energy conformations is established for the FCC-HP-model, one can employ it as a filter step in an hierarchical approach. This way, one can improve the energy function to achieve better biological relevance and go on to resemble amino acid positions more accurately.

Related Work In this paper, we describe a successful application of constraint-programming for finding native conformations in the FCC-HP-model. There, the situation as given in the literature was not very promising. Although the importance of the FCC-HP-model is widely known, exact algorithms for finding native conformations were known only for cubic lattice models. Even for the cubic lattice, there are only three exact algorithms known [26,4,9] that are able to enumerate minimal (or nearly minimal) energy conformations. However, the ability of this lattice to approximate real protein conformations is poor. Especially the parity problem was pointed out as drawback of the cubic lattice.[3] This problem is that every two monomers with chain positions of equal parity cannot form a contact.

So far, besides heuristic approaches (e.g., the hydrophobic zipper,[14] the genetic algorithm by Unger and Moult,[22] and the chain growth algorithm by Bornberg-Bauer [11]), there is only one approximation algorithm [3] for the FCC. It finds conformations whose number of contacts is guaranteed to be 60% of the number of contacts of the native conformation. The situation was even worse, since the main ingredient needed for an exact method was missing, namely bounds on the number of contacts between hydrophobic monomers given some partial information about the conformation. This changed with recent work,[5,8] where such a bound was introduced and applied for finding maximally compact hydrophobic cores. Given a conformation of an HP-sequence, the *hydrophobic core* of this sequence is the set of all points occupied by H-monomers. A hydrophobic core of n points is *maximally compact* if no packing of n points in the FCC has more contacts. Hydrophobic cores were used for structure prediction in the HP-model and HPNX-model on the cubic lattice before.[26,9]

Contribution and Use for Structure Prediction The goal of structure prediction in the FCC-HP-model can be achieved via the construction of hydrophobic cores (i.e. point sets) in the FCC. For predicting optimal structures of a sequence s, we will proceed as follows. First, search for the optimal number of contacts in any core of size $|s|$. Then, construct the set of all cores of size $|s|$ with optimal number of contacts. Try to thread the sequence s to all cores in this set. Now, possibly we cannot thread s to any of the cores. In this case, we iterate the process going on to suboptimal numbers of contacts.

The problem of threading a sequence s to a given core C (a set of lattice points) is finding a tuple $(\vec{p}_i)_{1\leq i\leq|s|}$ of lattice points, called a *structure* for s, subject to the constraints $\forall 1 \leq i < |s| : \vec{p}_i$ and $\vec{p}_{i+1}$ have minimal distance, $\forall 1 \leq i < j \leq |s| : \vec{p}_i \neq \vec{p}_j$, and $\forall 1 \leq i \leq |s| : s_i = H \rightarrow \vec{p}_i \in C$. As the problem is strongly constrained, we can solve it by constrained search.

The main contribution of this paper is an algorithm for constructing the maximally (and specified submaximally) compact hydrophobic cores of a given size in the FCC. A key idea of our method is to slice a core into layers orthogonal to the coordinate axis in every dimension. In previous work, upper bounds were given on the number of contacts for sequences of certain layer parameters.

As a result of this, only a very restricted number of layer parameter sequences has to be considered in a search for compact cores. Thus, the missing step is to search those candidate layer parameter sequences, here done by constraint-based search. We give a symmetric constraint model for the problem, which on the one hand permits to use the precomputed candidate layer parameter sequences for the layers in each of the three dimensions and on the other hand enables us to apply general symmetry exclusion.[6] A constraint-based algorithm is presented, suited for implementation in the constraint language Oz.[21]

2 Preliminaries and Basic Definitions

A *lattice* L is the minimal set of points that contains so called *generating vectors* $\vec{v}_1, \ldots, \vec{v}_n$ and where $\forall \vec{u}, \vec{v} \in L$, both $\vec{u} + \vec{v} \in L$ and $\vec{u} - \vec{v} \in L$ holds. The *face-centered cubic lattice* (FCC) is defined as the point set $D_3 = \{ \left(\begin{smallmatrix} x \\ y \\ z \end{smallmatrix}\right) \mid \left(\begin{smallmatrix} x \\ y \\ z \end{smallmatrix}\right) \in \mathbb{Z}^3 \text{and } x + y + z \text{ is even}\}$.

An (FCC-)core f is just a set of points in D_3. Define MV as the set of vectors $\vec{p} \in D_3$ with length $\sqrt{2}$ (which is the minimal euclidian distance of two lattice points). That is, $MV = \left\{ \left(\begin{smallmatrix} 0 \\ \pm 1 \\ \pm 1 \end{smallmatrix}\right), \left(\begin{smallmatrix} \pm 1 \\ 0 \\ \pm 1 \end{smallmatrix}\right), \left(\begin{smallmatrix} \pm 1 \\ \pm 1 \\ 0 \end{smallmatrix}\right) \right\}$. The *number of contacts* contacts(f) of a core f is defined as $\frac{1}{2} \left| \left\{ \ (p, q) \in f^2 \ \middle| \ \vec{p} - \vec{q} \in \mathrm{MV} \ \right\} \right|$.

A *caveat in a point set* f is a k-tuple of points $(\vec{p}_1, \ldots, \vec{p}_k)$ such that $\exists \vec{v} \in \mathrm{MV}\, \forall 1 \leq j < k : ((\vec{p}_{j+1} - \vec{p}_j) = \vec{v})$, $\{\vec{p}_1, \vec{p}_k\} \in f$ and $\forall 1 < j < k : \vec{p}_j \notin f$. Implicitly, we only handle cores without caveats. This restriction was also used in analogous cases by others[26] and is acceptable, since the presented ideas can be extended to handle caveats as well.

We define notations for certain point sets of $\mathbb{R}^3$, namely *lines* and *planes*. For vectors $\vec{a}, \vec{u} \in \mathbb{Z}^3$, let $\mathrm{LN}(\vec{a}, \vec{u})$ denote the set $\{\vec{p} \in \mathbb{R}^3 \mid \exists \lambda \in \mathbb{R} : \vec{p} = \vec{a} + \lambda \vec{u}\}$ and for $\xi \in \{x, y, z\}$ define $\mathrm{PL}(\xi, c)$ as the set $\{\vec{p} \in \mathbb{R}^3 \mid \xi = c\}$. For $a \in D_3$, we are interested in so-called *lattice lines* $\mathrm{LN}(\vec{a}, \vec{u})$, where $\vec{u} \in \mathrm{MV}$, and further so-called *non-lattice lines* $\mathrm{LN}(\vec{a}, \vec{u})$, where $\vec{u} \in \mathbb{Z}^3$ has length 2.

For a core f and $c \in \mathbb{Z}$, the set $f_{\xi=c} = f \cap \mathrm{PL}(\xi, c)$ is called the *ξ-layer of f in plane $\xi = c$*. For an x-layer f in plane $x = c$, define olines(f) as the tuple (a, b), where $a = \left| \left\{ \ l \ \middle| \ \exists \vec{a} \in D_3 : l = \mathrm{LN}(\vec{a}, \left(\begin{smallmatrix} 0 \\ 1 \\ 1 \end{smallmatrix}\right)) \wedge l \cap f \neq \emptyset \ \right\} \right|$ and analogously $b = \left| \left\{ \ l \ \middle| \ \exists \vec{a} \in D_3 : l = \mathrm{LN}(\vec{a}, \left(\begin{smallmatrix} 0 \\ -1 \\ 1 \end{smallmatrix}\right)) \wedge l \cap f \neq \emptyset \ \right\} \right|$. Define olines$(f)$ analogously for y-layers and z-layers f. Let min be the minimal number such that $\mathrm{PL}(\xi, \min) \cap f \neq \emptyset$ and m the maximal number such that $\mathrm{PL}(\xi, m + \min - 1) \cap f \neq \emptyset$. Define the *layer parameter sequence of a core f in dimension $\xi \in \{x, y, z\}$* as the finite sequence $\mathcal{L}_\xi(f) = (n_i, a_i, b_i)_{1 \leq i \leq m}$, such that $\forall 1 \leq i \leq m : |f_{\xi=\min + i - 1}| = n_i$ and $\mathrm{olines}(f_{\xi=\min + i - 1}) = (a_i, b_i)$.

Due to previous work,[8] we are able to compute for a layer parameter sequence $\mathcal{L}$ the upper bound on the number of contacts in the class of cores f with $\mathcal{L}_\xi(f) = \mathcal{L}$ for $\xi \in \{x, y, z\}$. This bound is denoted by $\mathrm{BMC}(\mathcal{L})$. Any core f with $\mathcal{L}_\xi(f) = \mathcal{L}$ has the same size determined by $\mathcal{L}$. Denote this size by $\mathrm{size}(\mathcal{L})$. Moreover, we are able to compute the sets of all layer parameter sequences, which have the same upper bound for a fixed core size. Define these sets by $S(n, \mathrm{bnd}) := \left\{ \mathcal{L} \;\middle|\; \mathrm{size}(\mathcal{L}) = n \wedge \mathrm{BMC}(\mathcal{L}) = \mathrm{bnd} \right\}$.

3 Problem Specification

For core construction, it remains to solve the following *core construction problem*. Given a core size n and the sets $S(n, \mathrm{bnd})$, construct a set $\mathrm{Cores}(n, \mathrm{con})$ of the cores of size n with at least con contacts modulo geometrical symmetry. First, note that it does not suffice to construct only maximally compact cores, if we want to use the cores for protein structure prediction, since there may be sequences which do not fit to all of the optimal cores. Second, by *modulo geometrical symmetry* we express that $\mathrm{Cores}(n, \mathrm{con})$ contains only one representative of every equivalence class due to translations, rotations and reflections. To abstract from at least translation symmetries is essential, since otherwise the set $\mathrm{Cores}(n, \mathrm{con})$ is trivially infinite.

Actually, we are going to solve the following (more general) problem. Given a set of layer parameter sequences S for cores of size n and a number of contacts con, compute the set of all cores f with at least con contacts which have layer parameter sequences from the set S in every dimension, i.e. compute

$$\mathrm{Cores}(n, S, \mathrm{con}) = \left\{ f \subset D_3 \;\middle|\; \begin{array}{c} \mathrm{contacts}(f) \geq \mathrm{con} \\ \wedge\ \forall \xi \in \{x, y, z\} : \mathcal{L}_\xi(f) \in S \end{array} \right\}$$

$$\text{modulo geometrical symmetry.}$$

As an abbreviation define $S_{con}(n, \mathrm{con}) = \bigcup_{\mathrm{bnd} \geq \mathrm{con}} S(n, \mathrm{bnd})$. Due to the equality $\mathrm{Cores}(n, \mathrm{con}) = \mathrm{Cores}(n, S_{con}(n, \mathrm{con}), \mathrm{con})$, the general problem solves the former problem of core construction. A difficulty remains with $\mathrm{Cores}(n, \mathrm{con})$, since $S_{con}(n, \mathrm{con})$ is not necessarily finite. However, in general there are finitely many cores $\mathrm{Cores}(n, S, \mathrm{con})$ only for finite input. Unfortunately, for sufficiently low numbers bnd, there may be layer parameter sequences $\mathcal{L} \in S(n, \mathrm{bnd})$, such that there is an $\min_j(n_j \neq 0) < i < max_j(n_j \neq 0)$ where $n_i = 0$. We say the sequence $\mathcal{L}$ has a *gap*. In this case, there are infinitely many layer parameter sequences expanding the gap in the sequence, which have the same bound bnd. Cores with gaps consist of separated sub-cores, instead of one connected set of points. Note that for structure prediction, this case occurs very rarely.

Nevertheless, we can cope with this problem, by a certain kind of symmetry exclusion. To generate cores in $\mathrm{Cores}(n, S_{con}(n, \mathrm{con}), \mathrm{con})$, we will first consider only the set S of layer parameter sequences in $S_{con}(n, \mathrm{con})$ without gaps. This guarantees that the set $\mathrm{Cores}(n, S, \mathrm{con})$ is finite. It is now possible to find the cores $f \in \mathrm{Cores}(n, S, \mathrm{con})$, which can be split into non-empty sub-cores $f' = f \cap \{\vec{p} \mid p_\xi \geq c\}$ and $f'' = f \cap \{\vec{p} \mid p_\xi < c\}$ along a plane $\xi = c$, such that $\mathrm{contacts}(f') + \mathrm{contacts}(f'') \geq \mathrm{con}$. Those cores can be used to generate an infinite number of elements of $\mathrm{Cores}(n, S_{con}(n, \mathrm{con}), \mathrm{con})$ by translation of one of the sub-cores. To be complete this has to be done recursively. Finally, note that for structure prediction we can even in this case restrict ourselves to finite sets of cores due to the restriction introduced by the chain length. However, in most cases, where we are interested in optimal or only slightly suboptimal cores, we can easily conclude that there are no such cores.

4 Description of the Algorithm

We solve the problem of constructing the cores in $\mathrm{Cores}(n, S, \mathrm{con})$ given a set of layer parameter sequences S without gaps for cores of size n and a number of contacts con. Our algorithm follows the constrain-and-generate principle. By large, this approach is to state constraints on solution variables and then enumerate values of the variables by branching to generate the solutions. At each branching, insert in the left branch a constraint c and in the right branch $\neg c$ to split the search space. In constraint programming, the branching is done concurrently to propagation of the stated constraint to prune the search tree.

We introduce variables together with data structures to organize them and constraints to express dependencies on the variables. It is useful to introduce auxiliary variables (instead of only the solution variables) and necessary to introduce redundant constraints for efficiency. Finally, we apply symmetry excluding constrained search [6] for enumerating the cores in $\mathrm{Cores}(n, S, \mathrm{con})$.

The main idea of our approach is getting as much knowledge as possible on the distribution of points from the layer parameter sequences. Therefore, we model layers and so called lattice lines of the layers to express the constraints by the a and b parameters. Further, it is crucial to employ the dependencies between layers of different dimensions. To express those dependencies we have to model non-lattice lines of the layers. The number of contacts con, yields further constraints, which are non-redundant to the former ones, since not every core satisfying the layer sequences has necessarily at least con contacts.

Variables All the variables are *finite domain variables (FD-variables)*, which means that their assigned values are restricted to values of finite integer do-

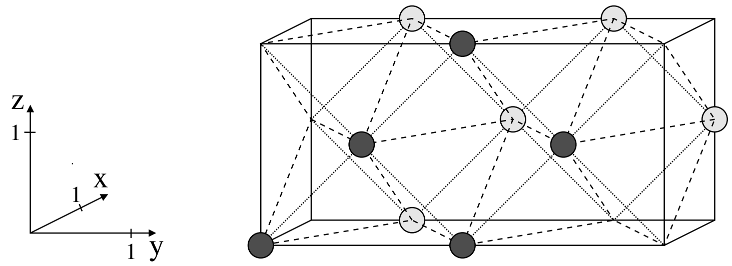

Figure 1: The cube for $m_x = 2$, $m_y = 5$, $m_z = 3$ and $\min_x + \min_y + \min_z$ even. The contacts within each x-layer are shown by dotted lines and the interlayer contacts between the two x-layers by dashed lines. The circles give an example core within the cube.

mains. Denote the number of non-empty layers in the dimension $\xi \in \{x, y, z\}$ by m_ξ. All points of the core will be placed in a $m_x \times m_y \times m_z$ *surrounding cube*. We can nearly fix the absolute coordinates of this cube to exclude translation symmetries. However, since D_3 contains only points of $\mathbb{Z}^3$ with even coordinate sum, the cube can only be fixed up to the minimal x, y, and z coordinate being one of $\{0, 1\}$. Store these coordinates in FD-variables $\min_x$, $\min_y$, and $\min_z$ respectively. Fix the surrounding cube to consist of the points $\mathrm{CB} = \{\min_x, \ldots, \min_x + m_x - 1\} \times \{\min_y, \ldots, \min_y + m_y - 1\} \times \{\min_z, \ldots, \min_z + m_z - 1\} \cap D_3$. Please see Figure 1 for an illustration.

For every point $\vec{p} \in \mathrm{CB}$, maintain a boolean FD-variable $\mathbf{pnt}(\vec{p}) \in \{0, 1\}$ that has value 1 iff the point $\vec{p}$ is element of the core. Let $\xi \in \{x, y, z\}$. For every layer $\xi = c$, where $\min_\xi \le c \le \min_x i + m_\xi - 1$, we have FD-variables $\mathbf{lay}(\xi, c).n$, $\mathbf{lay}(\xi, c).a$, and $\mathbf{lay}(\xi, c).b$ for the layer parameters.

Further, we have variables for all lattice and non-lattice lines within layers that intersect with the cube. For $\vec{v}$ in $\mathrm{LV} = \mathrm{MV} \cup \left\{ \begin{pmatrix} \pm 2 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \pm 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ \pm 2 \end{pmatrix} \right\}$, there are FD-variables $\mathbf{ln}(\vec{a}, \vec{v})$, for every set $\mathrm{LN}(\vec{a}, \vec{v})$ which has a non-empty intersection with CB. We identify variables $\mathbf{ln}(\vec{a}, \vec{v})$ and $\mathbf{ln}(\vec{a}', \vec{v})$ if $\mathrm{LN}(\vec{a}, \vec{v}) = \mathrm{LN}(\vec{a}', \vec{v})$. $\mathbf{ln}(\vec{a}, \vec{v})$ is the number of occupied points in $\mathrm{LN}(\vec{a}, \vec{v}) \cap D_3$. Finally, we introduce variables $\mathbf{con}(\vec{p}, \vec{q}) \in \{0, 1\}$ for $\vec{p}, \vec{q} \in \mathrm{CB}$, such that $\vec{p} - \vec{q} \in \mathrm{MV}$.

Basic Constraints Before giving the constraints, we introduce a notation to express *reified constraints*. Let c be a constraint, fix a mapping δ, $\delta(c) \in \{0, 1\}$, such that $\delta(c) = 1$ iff c holds. The FD-variables are subject to the following constraints. First of all, we get

$$\sum_{\vec{p} \in \mathrm{CB}} \mathbf{pnt}(\vec{p}) = n \quad \text{and} \quad \sum_{\vec{p}, \vec{q} \in \mathrm{CB}\ \vec{p} - \vec{q} \in \mathrm{MV}} \mathbf{con}(\vec{p}, \vec{q}) \le \mathrm{con}\,.$$

Any core must have one of the parameter layer sequences in each dimension $\xi \in \{x, y, z\}$. This is expressed by the (constructive) disjunction over all layer parameter sequences $\mathcal{L} = (n_i, a_i, b_i)_{1 \leq i \leq |\mathcal{L}|}$ in S of

$$\forall 1 \leq i \leq |\mathcal{L}| : \begin{array}{l} \quad \mathbf{lay}(\xi, min_\xi + i - 1).n = n_i \\ \wedge\, \mathbf{lay}(\xi, min_\xi + i - 1).a = a_i \wedge \mathbf{lay}(\xi, min_\xi + i - 1).b = b_i. \end{array}$$

Whereas, in general constructive disjunction is inefficient, here one can easily propagate information, e.g. the domains of the layer variables.

It remains to constrain relations between the variables to get the basic constraint formulation for our problem. First, we relate lines to points by $\mathbf{ln}(\vec{a}, \vec{v}) = \sum_{\vec{p} \in \mathrm{LN}(\vec{a}, \vec{v}) \cap \mathrm{CB}} \mathbf{pnt}(\vec{p})$ for all line variables. Then, we relate layer parameters to their layers by $\sum_{\vec{p} \in \mathrm{CB} \cap \mathrm{PL}(\xi, c)} \mathbf{pnt}(\vec{p}) = \mathbf{lay}(\xi, c).n$ for all $\xi \in \{x, y, z\}$ and $\min_\xi \leq c < \min_\xi + m_\xi$ and further, for x-layers and lattice lines in direction $\begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}$ introduce the constraints

$$\sum_{r \in \mathbb{Z},\ \mathrm{LN}\left(\begin{pmatrix} c \\ r \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}\right) \cap \mathrm{CB} \neq \emptyset} \delta\left(\mathbf{ln}\left(\begin{pmatrix} c \\ r \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}\right) > 0\right) = \mathbf{lay}(x, c).a$$

and the analogous constraints for $\mathbf{lay}(x, c).b$, the y-layers, and z-layers.

Now, relate contacts to points and to the total number of contacts. For any contact variable $\mathbf{con}(\vec{p}, \vec{q})$, introduce $\mathbf{con}(\vec{p}, \vec{q}) = \delta(\mathbf{pnt}(\vec{p}) = 1 \wedge \mathbf{pnt}(\vec{q}) = 1)$. Finally, state $\sum \mathbf{con}(\vec{p}, \vec{q}) = \mathrm{con}$. The previous constraints define the problem non-redundantly. For sufficient constraint propagation, we need to introduce redundant constraints like the following ones.

For example, the surrounding cube has to be large enough to include the core. Therefore, we introduce the constraints $\lceil \frac{m_x m_y m_z}{2} \rceil \geq n$ if the sum $\min_x + \min_y + \min_z$ is even and $\lfloor \frac{m_x m_y m_z}{2} \rfloor \geq n$ otherwise. Further, the line variables are connected to the layer parameter n by constraints

$$\sum_{r \in \mathbb{Z},\ \mathrm{LN}\left(\begin{pmatrix} c \\ r \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}\right) \cap \mathrm{CB} \neq \emptyset} \mathbf{ln}\left(\begin{pmatrix} c \\ r \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}\right) = \mathbf{lay}(x, c).n \quad \text{and analogous ones.}$$

Using Local Upper Bounds on the Number of Contacts The number of contacts within each layer is determined by the layer parameters, since we exclude caveats.[8] Thus, we can constrain the number of these (intra)layer contacts. We use also a constraint to forbid caveats directly. It constrains the core points along each lattice line to be connected.

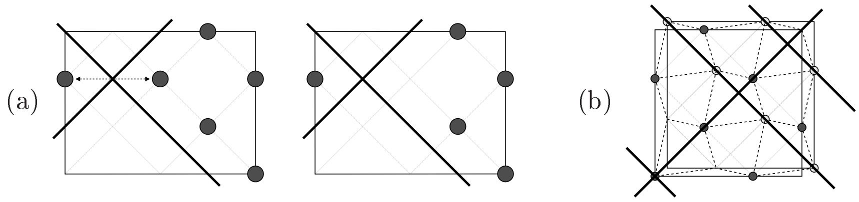

Figure 2: (a) The thick lines are drawn between non-overlapping pairs of lines. In both layers, we count one non-overlap and in the right layer one non-connect, since there is no connection (as shown in left layer by the arrow). (b) represents an example situation in the search. The thick lines are already known to intersect the core. Assume in each layer there are 5 core points, the beads mark remaining potential positions. The line constraints restrict the number of contacts, hence this additional knowledge is exploitable for the contacts bound.

Furthermore, we introduce redundant constraints that employ the upper bounds on the number of contacts between successive layers, called interlayer contacts. From earlier work,[8] we know non trivial upper bounds on the number of layer and interlayer contacts given parameters of the layers, namely the layer size, the previously defined olines(f) and the number of *non-connects* and *non-overlaps*. For an illustration of the latter terms, please see Figure 2(a).

Now, for $\min_\xi \leq c_1, c_2 < \min_\xi + m_\xi$ and $c_2 = c_1 \pm 1$, introduce FD-variables $\mathbf{ilay}(\xi, c_1, c_2).con$ to hold the number of interlayer contacts between layers $\xi = c_1$ and $\xi = c_2$. This variable is constrained to the sum of the corresponding contact variables and the total number of contacts is constrained to the sum of the layer contacts and the variables for interlayer contacts. The bound is strengthened and recomputed during the enumeration as more and more information, e.g. which lines intersect the core (see Figure 2(b)), becomes known. Therefore, variables to hold the additional parameters, the number of non-overlaps and non-connects, are introduced for each layer and corresponding constraints are stated. Furthermore, we introduce FD-variables $\mathbf{ilay}(\xi, c_1, c_2).i$ to hold the number of core points in $\xi = c_2$ with at least $i = 1, 2, 3$, or 4 contacts to core points in $\xi = c_1$. Such points were called *i-points*. Finally we can bind $\mathbf{ilay}(\xi, c_1, c_2).con$ to the sum $\sum_{1 \leq i \leq 4} \mathbf{ilay}(\xi, c_1, c_2).i$.

Search strategy We start a search by enumerating the variables m_x, m_y, m_z, $\min_x$, $\min_y$, and $\min_z$. This fixes the surrounding cube and allows in an implementation to construct all data structures. Afterwards, we distribute over the point variables to fix the core. To exclude rotation and reflection symmetries, we employ symmetry excluding search.[6] This search is a special form of constrained search, which only finds solutions modulo given symmetries

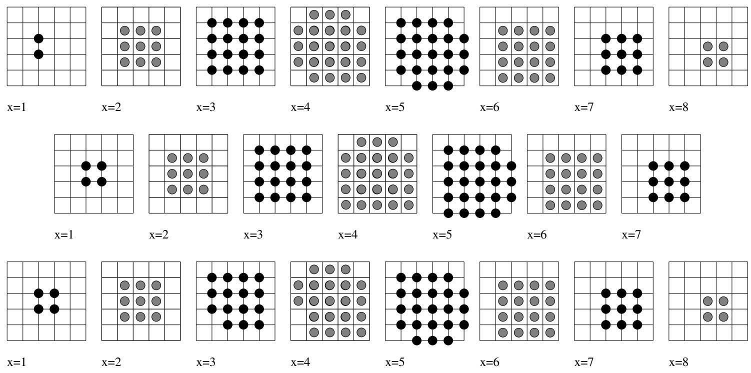

Figure 3: Plane sequence representations of 3 optimally compact cores of size $n = 100$.

and employs this to prune the search tree.

5 Results

All sets of layer parameter sequences $S_{\mathrm{con}}(n, \mathrm{con})$ without gaps for $n \leq 100$ were computed in about ten days on a standard PC. After this precomputation, which has to be done only once, the set of all optimally compact cores usually is found within a few seconds to minutes by our search program. Some results are shown in Table 1. Currently, the search program implements most of the presented ideas as well as additional redundant constraints.

Further, some optimal cores for $n = 100$ elements are shown in Figure 3. The cores are shown in plane sequence representation. This representation shows a core by the sequence of its occupied x-layers rotated by 45°. For each x-layer $x = x_0$ the lower left corner of the grid has coordinates $(x_0, 0, 0)$. The grid-lines are parallel to the lattice lines in x-layers and have distance $\sqrt{2}$. The core points in each x-layer are shown as filled circles.

Finally, we are able to thread sequences to hydrophobic cores for structure prediction, which is described in detail elsewhere.[7] There, we experimentally evaluate the ability of our algorithm to predict the structure for random sequences with 100 H-monomers and chain lengths of up to 160. We are able to find structures for 60% of the sequences with length 160 within 15 minutes. This percentage increases to 82%, when we allow one hour search time.

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

Table 1: Search for all optimally compact cores with n elements, given the layer sequences. We list the number of contacts, the number of nodes and depth of the search tree, and time of the constraint search for every core size n.

n	# contacts	# search-nodes	depth	time
40	152	167	17	17.2 s
60	243	182	72	4.6 s
82	349	220	37	14.2 s
102	447	54	20	8.2 s

2. V.I. Abkevich, A.M. Gutin, and E.I. Shakhnovich. Computer simulations of prebiotic evolution. In *PSB'97*, pages 27–38, 1997.
3. Richa Agarwala, Serafim Batzoglou, Vlado Dancik, Scott E. Decatur, Martin Farach, Sridhar Hannenhalli, S. Muthukrishnan, and Steven Skiena. Local rules for protein folding on a triangular lattice and generalized hydrophobicity in the HP-model. *Journal of Computational Biology*, 4(2):275–296, 1997.
4. Rolf Backofen. Constraint techniques for solving the protein structure prediction problem. In *Proceedings of* 4^{th} *International Conference on Principle and Practice of Constraint Programming (CP'98)*, volume 1520 of *Lecture Notes in Computer Science*, pages 72–86. Springer Verlag, 1998.
5. Rolf Backofen. An upper bound for number of contacts in the HP-model on the face-centered-cubic lattice (FCC). In *Proc. of the 11th Annual Symposium on Combinatorial Pattern Matching (CPM2000)*, volume 1848 of *Lecture Notes in Computer Science*, pages 277–292, Berlin, 2000. Springer–Verlag.
6. Rolf Backofen and Sebastian Will. Excluding symmetries in constraint-based search. In *Proceedings of* 5^{th} *International Conference on Principle and Practice of Constraint Programming (CP'99)*, volume 1713 of *Lecture Notes in Computer Science*, pages 73–87, Berlin, 1999. Springer–Verlag.
7. Rolf Backofen and Sebastian Will. Fast, constraint-based threading of HP-sequences to hydrophobic cores. In *Proceedings of* 7^{th} *International Conference on Principle and Practice of Constraint Programming (CP'2001)*, Lecture Notes in Computer Science, Berlin, 2001. Springer–Verlag. To appear.
8. Rolf Backofen and Sebastian Will. Optimally compact finite sphere packings — hydrophobic cores in the FCC. In *Proc. of the 12th Annual Symposium on Combinatorial Pattern Matching (CPM2001)*, Lecture Notes in Computer Science, Berlin, 2001. Springer–Verlag.
9. Rolf Backofen, Sebastian Will, and Erich Bornberg-Bauer. Application of constraint programming techniques for structure prediction of lattice proteins with extended alphabets. *J. Bioinformatics*, 15(3):234–242, 1999.
10. B. Berger and T. Leighton. Protein folding in the hydrophobic-hydrophilic (HP) modell is NP-complete. In *Proc. of the Second Annual International Conferences on Compututational Molecular Biology (RECOMB98)*, pages 30–39, New York, 1998.

11. 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.
12. P. Crescenzi, D. Goldman, C. Papadimitriou, A. Piccolboni, and M. Yannakakis. On the complexity of protein folding. In *Proc. of STOC*, pages 597–603, 1998. Short version in *Proc. of RECOMB'98*, pages 61–62.
13. 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.
14. Ken A. Dill, Klaus M. Fiebig, and Hue Sun Chan. Cooperativity in protein-folding kinetics. *Proc. Natl. Acad. Sci. USA*, 90:1942 – 1946, 1993.
15. 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.
16. S. Govindarajan and R. A. Goldstein. The foldability landscape of model proteins. *Biopolymers*, 42(4):427–438, 1997.
17. Patrice Koehl and Michael Levitt. A brighter future for protein structure prediction. *Nature Structural Biology*, 6:108–111, 1999.
18. 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.
19. Hao Li, Robert Helling, Chao Tnag, and Ned Wingreen. Emergence of preferred structures in a simple model of protein folding. *Science*, 273:666–669, 1996.
20. Britt H. Park and Michael Levitt. The complexity and accuracy of discrete state models of protein structure. *Journal of Molecular Biology*, 249:493–507, 1995.
21. Gert Smolka. The Oz programming model. In *Computer Science Today*, Lecture Notes in Computer Science, vol. 1000, pages 324–343. Springer-Verlag, Berlin, 1995.
22. R. Unger and J. Moult. Genetic algorithms for protein folding simulations. *Journal of Molecular Biology*, 231:75–81, 1993.
23. Ron Unger and John Moult. Local interactions dominate folding in a simple protein model. *Journal of Molecular Biology*, 259:988–994, 1996.
24. A. Šali, E. Shakhnovich, and M. Karplus. Kinetics of protein folding. *Journal of Molecular Biology*, 235:1614–1636, 1994.
25. Yu Xia, Enoch S. Huang, Michael Levitt, and Ram Samudrala. Ab initio construction of protein tertiary structures using a hierarchical approach. *Journal of Molecular Biology*, 300:171 – 185, 2000.
26. Kaizhi Yue and Ken A. Dill. Forces of tertiary structural organization in globular proteins. *Proc. Natl. Acad. Sci. USA*, 92:146 – 150, 1995.