A Branch-and-Bound Constraint Optimization Approach to the HPNX Structure Prediction Problem

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

Abstract

The protein structure prediction problem is one of the most important problems in computational biology. Because of the complexity of this problem, simplified models like Dill's HP-lattice model [9, 10] 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 [4, 2].

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 [3], which considers hydrophobic amino acids as well as positively and negatively 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.

1 Introduction

The protein structure prediction is one of the most important unsolved problems of computational biology. Many results in the past have shown the problem to be NP-hard. 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. Some commonly used simplifications 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. 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 [5]. 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 [13, 1, 6, 8, 7, 11].

An important representative of lattice models is the HP-model, which has been introduced by [9, 10]. 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 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 neighbor positions in the conformation. A

conformation with *minimal energy* 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 [2, 4].

An example of the use of lattice models is the work by Šali, Shakhnovich and Karplus [13]. The same lattice model is used by several other people, e.g., [1, 7]. The authors investigate in [13] under which conditions a protein folds into its native structure. For this purpose, they have performed computer simulations of protein folding on 200 proteins in the cubic lattice. The simulation of protein folding was done by using a Monte Carlo method. A protein was defined to be foldable if the Monte Carlo method finds the minimal energy (= native) structure. 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. For the energy function used by [13], there is no *exact* algorithm for finding the minimal structure. To be computational feasible, they have restricted in [13] the search for the native structure on the $3 \times 3 \times 3$-cube. 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 [13]); 2.) even then, it is not guaranteed that the minimal conformation is on this cube 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 using the HP-model. But the HP-model has the problem that its degeneracy (i.e., the number of structures of a sequence that have minimal energy) is large [5]. 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 as the HPNX-model [3] have been introduced, which we are considering in this paper. 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)

2 Structure Prediction as a Constraint Problem

Let $s = s_1 \dots s_n$ be an HPNX-sequence of length n. We say that a monomer with number i in s is even (resp. odd) if i is even (resp. odd). For convenience, we talk of a PNX-monomer meaning either a P, N or X monomer. With $\|\vec{p} - \vec{p}'\|$, we denote the Euclidean distance between $\vec{p}$ and $\vec{p}'$. 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 $\|c(i) - c(i+1)\| = 1$
2. and for all $i \neq j$ we have $c(i) \neq c(j)$ (the conformation c is self-avoiding).

Given a conformation c of a sequence s and two monomers i and j with $i+1 < j$, then i and j *form a contact* in c if $||c(i) - c(j)|| = 1$

With $\vec{e}_x$, $\vec{e}_y$ and $\vec{e}_z$ we denote the unit vectors $(1,0,0)$, $(0,1,0)$ or $(0,0,1)$, respectively. We say that two points $\vec{p}, \vec{p}' \in \mathbb{Z}^3$ are *neighbours* if $\|\vec{p} - \vec{p}'\| = 1$. This is equivalent to the proposition that $\vec{p} = \vec{p}' \pm \vec{e}$ with $\vec{e} \in \{\vec{e}_x, \vec{e}_y, \vec{e}_z\}$. Given a conformation c, the *H-surface* $\mathrm{HSurf}_s(c)$ of c is defined as the number of pairs of neighbour

positions, where the first is occupied by an H-monomer, but the second not. I.e.,

$$\mathrm{HSurf}_s(c) \quad = \quad \left| \left\{ \; (c(i), \vec{p}) \; \middle| \; \begin{array}{c} s_i = H \;\wedge\; \|\vec{p} - c(i)\| = 1 \\ \wedge \quad \forall j : (s_j = H \Rightarrow c(j) \neq \vec{p}) \end{array} \right\} \right|$$

Now Yue and Dill [14] made the observation that there is a simple linear equation relating H-surface and the number of HH-contacts of c (denoted by $\mathrm{HHContact}_s(c)$). This equation uses the fact that every monomer has 6 neighbours in the $\mathbb{Z}^3$, each of which is in any conformation either filled with either an H-monomer, a PNX-monomer, or left free. Let $n_H(s)$ be the number of H-monomers in s, then we have for every conformation c that

$$6 \cdot n_H(s) \quad = \quad 2 \cdot [\mathrm{HHContact}_s(c) + \mathrm{HHBonds}_s] + \mathrm{HSurf}_s(c) \tag{2}$$

where $\mathrm{HHBonds}_s$ is the number of bonds between H-monomers (i.e., the number of H-monomers whose successor in s is also a H-monomer). Since $\mathrm{HHBonds}_s$ is constant for all conformations c of sequence s, this implies that minimizing the surface is the same as maximising the number of HH-contacts.

Given a conformation, the *frame* of the conformation is the minimal rectangular box that contains all H-monomers of the sequence (see [14]). Given a vector $\vec{p}$, we denote with $(\vec{p})_{\mathrm{x}}$, $(\vec{p})_{\mathrm{y}}$ and $(\vec{p})_{\mathrm{z}}$ the x-,y- and z-coordinate of $\vec{p}$, respectively. The *dimensions* (fr_x, fr_y, fr_z) of the frame are the numbers of monomers that can be placed in x-, y- and z-direction within the frame. I.e.,

$$fr_x \quad = \quad \max\{|(c(i) - c(j))_{\mathrm{x}}| \mid 1 \leq i, j \leq n \wedge s_i = H \wedge s_j = H\} + 1.$$

fr_y and fr_z are defined analagously. We define s_x to be $\min\{c(i)_{\mathrm{x}} \mid 1 \leq i, j \leq n \wedge s_i = H\}$. s_y and s_z are defined analogously. (s_x, s_y, s_z) is called *starting point* of the frame.

2.1 Basic Constraints and Search Algorithm

We start with the basic constraint formulation that underlies our search algorithm. Our algorithm is based on constraint optimization, which is the combination of two principles, namely generate-and-constraint with branch-and-bound. For using constraint optimization, 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 on these variables. In the following, we fix a sequence s of length n.

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 these coordinates are all integers. But we can even restrict the possible values of these variables to the finite domain $[1..2n]$.[1] This is expressed by introducing the constraints

$$\mathtt{X}_i \in [1..(2 \cdot n)] \wedge \mathtt{Y}_i \in [1..(2 \cdot n)] \wedge \mathtt{Z}_i \in [1..(2 \cdot n)] \tag{3}$$

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 $i \neq j$.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 $1 \leq i < n$ three

[1] We even could have used $[1..n]$. But the domain $[1..2n]$ 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.

variables $\texttt{Xdiff}_i$, $\texttt{Ydiff}_i$ and $\texttt{Zdiff}_i$. These variables have values 0 or 1. Then we can express the unit-vector distance constraint by

$$\begin{aligned} \texttt{Xdiff}_i &= |\texttt{X}_i - \texttt{X}_{i+1}| \qquad \texttt{Zdiff}_i = |\texttt{Z}_i - \texttt{Z}_{i+1}| \\ \texttt{Ydiff}_i &= |\texttt{Y}_i - \texttt{Y}_{i+1}| \qquad 1 = \texttt{Xdiff}_i + \texttt{Ydiff}_i + \texttt{Zdiff}_i. \end{aligned}$$

The constraints described above span the space of all possible conformations. I.e., every valuation of $\texttt{X}_i, \texttt{Y}_i, \texttt{Z}_i$ 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 $\texttt{X}_i, \texttt{Y}_i, \texttt{Z}_i$ (expressed by additional constraints as introduced by the search algorithm), we call a conformation c *compatible* with these constraints on $\texttt{X}_i, \texttt{Y}_i, \texttt{Z}_i$ if c is admissible and c satisfies the additional constraints.

But in order to use constraint optimization, 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. For this purpose we introduce for every pair (i, j) of monomers with $i + 1 < j$ a variable $\texttt{Contact}_{i,j}$. $\texttt{Contact}_{i,j}$ is 1 if i and j have a contact in every conformation which is compatible with the valuations of $\texttt{X}_i, \texttt{Y}_i, \texttt{Z}_i$, and 0 otherwise. Then we can express this property in constraint programming as follows:

$$\begin{aligned} \texttt{Xdiff}_{i,j} &= |\texttt{X}_i - \texttt{X}_j| \qquad \texttt{Zdiff}_{i,j} = |\texttt{Z}_i - \texttt{Z}_j| \\ \texttt{Ydiff}_{i,j} &= |\texttt{Y}_i - \texttt{Y}_j| \qquad \texttt{Contact}_{i,j} \in \{0, 1\} \\ (\texttt{Contact}_{i,j} = 1) &\leftrightarrow (\texttt{Xdiff}_i + \texttt{Ydiff}_i + \texttt{Zdiff}_i = 1) \end{aligned} \tag{4}$$

where $\texttt{Xdiff}_{i,j}$, $\texttt{Xdiff}_{i,j}$ and $\texttt{Zdiff}_{i,j}$ are new variables. The constraint (4) is called a reified constraint, and can be encoded directly in many modern constraint programming languages. Using the variables $\texttt{Contact}_{i,j}$, we can now easily encode the energy function, which is subject to constraint optimization. We introduce the variables `HHContacts`, `PNContacts`, `PPContacts` and `NNContacts`, which count the number of contacts between monomers of the specified type. Thus, `HHContacts` is defined by

$$\texttt{HHContacts} = \textstyle\sum_{i+1<j \wedge s(i)=H \wedge s(j)=H} \texttt{Contact}_{i,j}.$$

The variables `PNContacts`, `PPContacts` and `NNContacts` are defined analogously. Finally, we can now define a variable `Energy`, where we have the constraint

$$\texttt{Energy} = -4 \cdot \texttt{HHContacts} - \texttt{PNContacts} + \texttt{PPContacts} + \texttt{NNContacts}.$$

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, an undetermined variable *var* out of the set of variables $\{\texttt{X}_i, \texttt{Y}_i, \texttt{Z}_i \mid 1 \leq i \leq n\}$ 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

$$\texttt{Energy} < E_c \tag{5}$$

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 (5). 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 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.

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 constraints that allow early elimination of invalid configurations, and 2.) the ability for implementing a search strategy that tends to enumerate low energy conformations first.

2.2 Additional Variables and Constraints

We start with defining the additional variables used in our formulation. With $(\mathtt{Frx}, \mathtt{Fry}, \mathtt{Frz})$, we denote the dimension of the frame. In [14] it is shown, that setting the frame dimension first allows to exclude many conformations having a non-optimal number of HH-contacts. Clearly, one has to search through different frame dimensions to find the optimal conformation in general. By using the lower bound of the H-surface given the sequence and the H-frame dimensions as defined in [14], one usually needs to search through a tiny number of frame dimensions to find the optimal conformation.

Hence, we start with setting the frame dimension $(\mathtt{Frx}, \mathtt{Fry}, \mathtt{Frz})$. If these variables are determined, we fix the frame starting point.[2] Having this, we can add for every H-monomer i the constraints $s_x \leq \mathtt{X}_i \leq s_x + \mathtt{Frx} - 1$, $s_y \leq \mathtt{Y}_i \leq s_y + \mathtt{Fry} - 1$ and $s_z \leq \mathtt{Z}_i \leq s_z + \mathtt{Frz} - 1$.

The remaining variables consider the different positions that a monomer can occupy. For simplifying the description in this paper, we will assume that we consider every position in $[1..(2 \cdot n)] \times [1..(2 \cdot n)] \times [1..(2 \cdot n)]$, and that every monomer can (initially) occupy every position. With fixing the frame, this is not true, since many monomers can only be placed onto a restricted set of positions. We have used this optimisation in our implementation, but skip it here for simplicity of presentation.

The first set of variables is related to planes parallel to the ones of the coordinate axis. An *x-layer* is a plane defined by the equation $x = c$ for some integer c. *y-layers* and *z-layers* are defined analogously. For the membership of monomers to layers, we introduce additional Boolean variables. For every monomer i and every integer $1 \leq c \leq 2 \cdot n$, we introduce a variable $\mathtt{Elem}_i^{x,c}$. $\mathtt{Elem}_i^{x,c}$ is 1 if the monomer is in the x-layer defined by $x = c$.[3] Thus, we have the reified constraint $(\mathtt{Elem}_i^{x,c} = 1) \;=\; (\mathtt{X}_i = c)$. The distribution of monomers to x-layers is restricted by the following constraints valid for the cubic lattice. If two monomers i and $i+2$ are in the same x-layer, then $i+1$ and must also be the same x-layer. I.e., for every $1 \leq c \leq 2 \cdot n$ we have

$$(\mathtt{Elem}_i^{x,c} = 1 \wedge \mathtt{Elem}_{i+2}^{x,c} = 1) \quad \rightarrow \quad (\mathtt{Elem}_{i+1}^{x,c} = 1)$$

If two monomers i and $i+3$ are in the same x-layer, then $i+1$ and $i+2$ must also be in one x-layer. I.e., for every $1 \leq c \leq 2 \cdot n$ we have

$$(\mathtt{Elem}_i^{x,c} = 1 \wedge \mathtt{Elem}_{i+3}^{x,c} = 1) \quad \rightarrow \quad \mathtt{X}_{i+1} = \mathtt{X}_{i+2}$$

We treat y-layers and z-layers analogously.

Finally, we have variables related to positions that can be occupied by monomers. Let $\vec{p} = (p_x, p_y, p_z)$ be some position and i be a monomer. The *occurrence variable* $\mathtt{O}_i^{\vec{p}}$ is

[2] this can always be done in a way which is compatible with $(\mathtt{Frx}, \mathtt{Fry}, \mathtt{Frz})$ and the constraint (3)

[3] We do not have to consider all pairs of i, c in reality since the frame is fixed first.

a Boolean variable that is 1 if the monomer i occupies the position $\vec{p}$, and 0 otherwise. This variable can be defined by

$$(\mathtt{O}_i^{\vec{p}} = 1) \quad \leftrightarrow \quad (\mathtt{Elem}_i^{x,p_x} = 1 \wedge \mathtt{Elem}_i^{y,p_y} = 1 \wedge \mathtt{Elem}_i^{z,p_z} = 1).$$

With the constraint $\left(\sum_{1 \le i \le n} \mathtt{O}_i^{\vec{p}}\right) \le 1$ we guarantee that every position may be occupied by at most one monomer.

Since the major part of the search tree is spanned over all possible assignments of monomers to positions, it is important to exclude invalid assignments as soon as possible. We do this by relating the different occurrences of neighbor positions. For every positions $\vec{p}$ and every monomer $1 < i < n$, we introduce the constraint

$$(\mathtt{O}_i^{\vec{p}} = 1) \rightarrow \left(\sum_{\vec{p'} \text{ neighb of } p} \mathtt{O}_{i+1}^{\vec{p}}\right) \ge 1 \quad \text{and} \quad (\mathtt{O}_i^{\vec{p}} = 1) \rightarrow \left(\sum_{\vec{p'} \text{ neighb of } p} \mathtt{O}_{i-1}^{\vec{p}}\right) \ge 1.$$

For $i = 1$ we introduce only the first constraint, for $i = n$ only the second. This constraint just states that i can only occupy the position $\vec{p}$ if both monomers $i-1$ and $i+1$ occupy a neighbor position of $\vec{p}$. This generalize the concept of the *tether length* as introduced in [14] and extended in [15], which only states which H-monomers can occupy which positions in the H-frame, not taking into account where the neighbor monomers can be placed. Thus, our constraint prunes the search tree given partial distribution of monomers to positions, which is not true for the tether constraint.

The final set of constraints relates occurrence variables and the energy variable in various ways. For every position $\vec{p}$, we introduce the Boolean variables $\mathtt{Htype}_{\vec{p}}$, $\mathtt{Ptype}_{\vec{p}}$, $\mathtt{Ntype}_{\vec{p}}$ and $\mathtt{Xtype}_{\vec{p}}$. These variables are 1 if the positions are occupied by an H-monomer of the corresponding type. Thus, $\mathtt{Htype}_{\vec{p}}$ is defined by

$$(\mathtt{Htype}_{\vec{p}} = 1) \quad \leftrightarrow \quad \left(\left(\sum_{1 \le i \le n \wedge s_i = H} \mathtt{O}_i^{\vec{p}}\right) \ge 1\right)$$

$\mathtt{Ptype}_{\vec{p}}$ and $\mathtt{Ntype}_{\vec{p}}$ are defined analogously, but we need a special definition for $\mathtt{Xtype}_{\vec{p}}$, since a position has X-type if it is occupied by an X-monomer, or not occupied by a monomer at all. This condition and the condition that a position has a unique type is expressed by the following two constraints:

$$\begin{aligned} (\mathtt{Xtype}_{\vec{p}} = 1) \quad &\leftrightarrow \quad (\mathtt{Htype}_{\vec{p}} + \mathtt{Ptype}_{\vec{p}} + \mathtt{Ntype}_{\vec{p}} = 0) \\ 1 \quad &= \quad \mathtt{Xtype}_{\vec{p}} + \mathtt{Htype}_{\vec{p}} + \mathtt{Ptype}_{\vec{p}} + \mathtt{Ntype}_{\vec{p}} \end{aligned}$$

Additionally, we have $\sum_{\vec{p}} \mathtt{Htype}_{\vec{p}} = n_H(s)$, $\sum_{\vec{p}} \mathtt{Ptype}_{\vec{p}} = n_P(s)$ and $\sum_{\vec{p}} \mathtt{Ntype}_{\vec{p}} = n_N(s)$, where $n_H(s)$, $n_P(s)$ and $n_N(s)$ is the number of H-, P- and N-monomers in s, respectively.

Finally, we have constraints relating the type variables of positions and H-surface contributions. As already mentioned, the number of HH-contacts can be more easily approximated from the surface of all H-surface. Thus, we introduce the Boolean variables $\mathtt{HSurf}_{\vec{p}}^{\vec{p'}}$ for all neighbor positions $\vec{p}$ and $\vec{p'}$, which is defined by $\mathtt{HSurf}_{\vec{p}}^{\vec{p'}} = (\mathtt{Htype}_{\vec{p}} = 1 \wedge \mathtt{Htype}_{\vec{p'}} = 0)$. Of course, we get $\mathtt{HSurf} = \sum_{\vec{p},\vec{p'} \text{ neighbours}} \mathtt{HSurf}_{\vec{p}}^{\vec{p'}}$. The variable $\mathtt{HSurf}$ is then used to constrain the variable $\mathtt{Energy}$ as described by equation 2.

Search Strategy As the first step, we search the maximal number of HH-contacts that is possible for the sequence s. Once we have found the maximal number m of $\mathtt{HHContacts}$, we start with finding a conformation with m $\mathtt{HHContacts}$ that maximizes $\mathtt{PNContacts} - \mathtt{PPContacts} - \mathtt{NNContacts}$. The next step is to decrease $\mathtt{HHContacts}$ to $m-1$, and to restart the search for the conformation which maximizes $\mathtt{PNContacts} - \mathtt{PPContacts} - \mathtt{NNContacts}$. Since one HH-contact correspond to 4 PN-contacts, and since (by branch and bound) we need to find a better conformation, this implies that we need to find a conformation where at least 5 more PN-contacts, which does not

exist in many cases. But this case may not be excluded for completeness, which shows that it is not enough to search through all minimal conformation of the corresponding HP-sequence.

Within the major search steps described above, we select the variables according to the following order.[4] First, we determine the frame dimensions $\texttt{Frx}, \texttt{Fry}, \texttt{Frz}$. After that, we determine the x-values of the H-monomers. This allows one to apply the lower bound on HSurf_s, which is in fact an upper bound on $\texttt{HHContacts}$. This is described in the next section. Finally, we determine the positions of the monomers.

2.3 Lower Bound on HHContacts

For every x-layer defined by the equation $x = k$, we define the variables Lseh_k and Lsoh_k counting the number of even and odd H-monomer in that layer. In order to apply the lower bound, we have fix these numbers. A way to achieve this is to fix an assignment of H-monomer to x-layers (which implies that the the variable $\texttt{X}_i$ is determined for every i with $s_i = H$), as it is done by our search strategy.

Let c be some conformation of s. We now distinguish between surface contribution in x-direction, and surface contributions in the single x-layers (i.e., contributions in y- and z-direction). For this purpose, we define

$$\begin{aligned}
\mathrm{HSurf}_{\mathrm{s}}^{\mathrm{x}}(c) &= \left| \left\{ (c(i), \vec{p}) \;\middle|\; \begin{array}{ll} & s_i = H \;\wedge\; \vec{p} - c(i) = \pm \vec{e}_x \\ \wedge & \mathrm{Htype}_{\vec{p}}(c) = 0 \end{array} \right\} \right| \\
\mathrm{LayHSurf}_s^{x=k}(c) &= \left| \left\{ (c(i), \vec{p}) \;\middle|\; \begin{array}{ll} & s_i = H \;\wedge\; \|\vec{p} - c(i)\| = 1 \\ \wedge & c(i)_x = k = \vec{p}_x \wedge \mathrm{Htype}_{\vec{p}}(c) = 0 \end{array} \right\} \right|
\end{aligned}$$

where $\mathrm{Htype}_{\vec{p}}(c)$ is defined by $\mathrm{Htype}_{\vec{p}}(c) = 1 \Leftrightarrow \exists i : (s_i = H \wedge c(i) = \vec{p})$. Clearly, we have $\mathrm{HSurf}_s(c) = \mathrm{HSurf}_{\mathrm{s}}^{\mathrm{x}}(c) + \sum_{1 \le k \le 2 \cdot n} \mathrm{LayHSurf}_s^{(x=k)}(c)$.

Given a point $(x, y, z) \in \mathbb{Z}^3$, we say that (x, y, z) is *odd* (resp. *even*) if $x + y + z$ is odd (resp. even). We write $(x, y, z) \equiv (x', y', z')$ iff $x + y + z \equiv x' + y' + z' \mod 2$.

Proposition 2.1 *Let c be a conformation of s. Then $c(i) \equiv c(j)$ iff $i \equiv j \mod 2$.*

From this we get the following lower bound on $\mathrm{HSurf}_{\mathrm{s}}^{\mathrm{x}}(c)$, provided that we know how many even and odd monomers are placed on the j^{th} layer. It is easy to see that the Lsoh_1 monomers generate Lsoh_1 surface points in $-$x direction. Furthermore, there are Lsoh_1 points in $+$x direction, which are candidates for surface points. But all these candidates are even points. If $\mathrm{Lsoh}_1 > \mathrm{Lseh}_2$, then we have minimal $\mathrm{Lsoh}_1 - \mathrm{Lseh}_2$ surface points in the second layer. If $\mathrm{Lsoh}_1 \le \mathrm{Lseh}_2$, then a similar argumentation shows that we have at least $\mathrm{Lseh}_2 - \mathrm{Lsoh}_1$ surface points in the first layer.

Lemma 2.2 *Let c be a conformation of s, and let $(\mathrm{Lsoh}_1, \mathrm{Lseh}_1, \ldots, \mathrm{Lsoh}_{2n}, \mathrm{Lseh}_{2n})$ be the number of H-monomers of c that are placed in the different layers. Then $\mathrm{HSurf}_{\mathrm{s}}^{\mathrm{x}}(c) \ge \mathrm{Lsoh}_1 + \mathrm{Lseh}_1 + \mathrm{Lsoh}_{2n} + \mathrm{Lseh}_{2n} + \sum_{1<j<2n} |\mathrm{Lsoh}_j - \mathrm{Lseh}|$*

For calculating the yz-surface of a specific layer, we introduce the concept of a coloring. A coloring just states which points are occupied by some H-monomer. A *coloring* is a function $f : \mathbb{Z}^2 \to \{0, 1\}$. We say that a point (x, y) is colored black by f iff $f(x, y) = 1$. In the following, we consider only colorings different from the empty coloring f_e (which satisfies $\forall \vec{p} : f_e(\vec{p}) = 0$). Given a coloring f, define

$$\begin{aligned}
e(f) &= |\{(x, y) \mid f(x, y) = 1 \text{ and } x + y \text{ even}\}| \\
o(f) &= |\{(x, y) \mid f(x, y) = 1 \text{ and } x + y \text{ odd}\}| \\
\mathrm{ColSurf}(f) &= |\{(\vec{p}, \vec{p}') \in \mathbb{Z}^2 \mid \vec{p}, \vec{p}' \text{ neighbours} \wedge f(\vec{p}) = 0 \wedge f(\vec{p}') = 1 \;\}|
\end{aligned}$$

[4] We present only an oversimplified description of the search strategy for clarity of presentation; the implemented strategy is much more complex in the way the single variables are selected

$\mathrm{ColSurf}(f)$ is called the surface of f. Given a pair (e,o) of integers, we define $\mathrm{ColSurf}(e,o)$ to be the minimum of $\{\mathrm{ColSurf}(f) \mid f \text{ colouring with } e(f) = e \wedge o(f) = o\}$. The next lemma relates the surface of colorings with the yz-surface of a conformation.

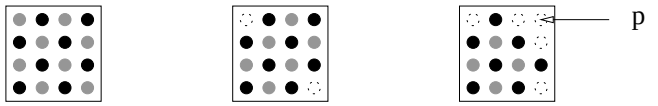
Lemma 2.3 *Let c be a conformation of s having* Lseh_j *even (resp.* Lsoh_j *odd) points in the j^{th} x-layer. Then* $\mathrm{LayHSurf}_s^{(x=j)}(c) \geq \mathrm{HSurf}(\mathrm{Lseh}_j, \mathrm{Lsoh}_j)$

Thus, Lemma 2.2 together with Lemma 2.3 provide a lower bound on the surface. Since $\mathrm{ColSurf}(e,o) = \mathrm{ColSurf}(o,e)$, it is sufficient to treat the case where $e \leq o$. In the following theorem, we handle the simple case where $|e-o| \leq 1$.

Theorem 2.4 *Let (e,o) be a pair of integers with $|e-o| \leq 1$. Let $a = \lceil \sqrt{e+o} \rceil$ and $b = \lceil \frac{e+o}{a} \rceil$. Then* $\mathrm{ColSurf}(e,o) = 2a + 2b$.

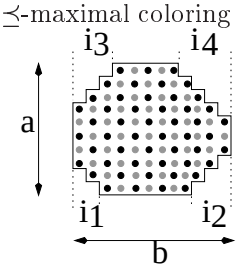
The remaining case is to calculate $\mathrm{ColSurf}(e,o)$ where $e < o-1$, without the need to search through all possible colorings f. A point $(x,y) \in \mathbb{Z}^2$ is a *caveat in f* if $(x,y) = 0$ and (x,y) is contained in the hull (over $\mathbb{Z}^2$) of the points colored black in f. We handle only caveat-free colorings in this paper. The case of a coloring with caveats can be reduced to the caveat-free case.

Given a coloring f, we denote with the *frame* (a,b) the maximal dimension of the coloring in y- and x-direction. Since we are considering caveat-free colorings, we get that the surface of f is $2 \cdot a + 2 \cdot b$, where (a,b) is the frame of f. Hence, we can calculate the surface of (e,o) by finding a minimal frame (a,b) such that there is a coloring of (e,o) having this frame. The first condition is clearly that $a \times b \geq e + o$. This condition is exactly the case that is treated in Theorem 2.4. But in the case that we have $e < o - 1$, this condition is not sufficient. The reason is that given a fixed frame (a,b), it may well be that we can color $e+1$ even and o odd points in the frame (a,b), but not e even and o odd. E.g., consider a fixed frame of size $(4,4)$. Grey points indicate even points, black ones odd points. We define $d(f) = o(f) - e(f)$. Then three maximal colorings for different values of $d(f)$ are



If we have the same number of even and odd points ($d(f) = 0$), then we can color at most 16 points in that frame. If $d(f) = 2$, then we can color at most 14 points. But if $d(f) = 3$, we can color at most 11 points, because we have to remove one odd position (e.g. p) before we can reduce the number of even positions. This leads to the following definition. The partial order $\preceq$ on caveat-free colorings is defined by $f \preceq f'$ if and only if $\mathrm{height}(f) = \mathrm{height}(f')$, $\mathrm{length}(f) = \mathrm{length}(f')$ and $d(f) = d(f')$.

Now we have $f \preceq f'$ implies that f, f' have the same surface. The nice thing is that $\preceq$-maximal colorings have a simple normal form, from which $d(f)$ can easily be read off. An example of such $\preceq$-maximal coloring (called simple coloring) f is



(6)

Again, we use black beads for odd positions colored by f, and grey for even. (a,b) is the frame of f, and $i_1, \ldots, i_4$ are the side length of triangles excluded at the corners. The tuple (a,b,i_1,i_2,i_3,i_4) is called the characteristics of this coloring (here it is $(10,12,2,3,3,4)$).

Theorem 2.5 *Every coloring can be extended to a simple coloring with the same surface. Let f be a simple coloring with characteristics $(a, b, i_1, i_2, i_3, i_4)$. Then* $\mathrm{ColSurf}(f) = 2a + 2b$. *Furthermore, we have*

$$e(f) + o(f) = a \times b - \sum_{j=1}^{4} \frac{i_j(i_j+1)}{2} \quad \textit{and} \quad d(f) = \frac{i_1+i_2+i_3+i_4}{2} + 1$$

This can be used for calculating $\mathrm{ColSurf}(e, o)$ as follows. We start with the minimal frame (a, b) for $e + o$ as stated in Theorem 2.4. Then we search for numbers i_1, i_2, i_3, i_4 satisfying the above constraints. Note that we do not have to search through all possible numbers for i_1, i_2, i_3, i_4, since Lemma B.10 gives a good restriction on the possible characteristics of maximal colorings. If we find an appropriate valuation for i_1, i_2, i_3, i_4, then the $\mathrm{ColSurf}(e, o)$ is given by $2a+2b$. Otherwise, we have to search for the next bigger frame.

2.4 Bound on the PN-Energy

The PN-Energy is $-$`PNContacts`$+$`PPContacts`$+$`NNContacts`. To get a lower bound on this energy, we need an upper bound on `PNContacts` and lower bounds on `PPContacts` and `NNContacts`. We need some additional variables and constraints.

For an upper bound on the `PNContacts`, we have to count the number of N-neighbours of positions, which are occupied by a P-monomer (or equivalently, the number of P-neighbours of position occupied by an N-monomer). For this purpose, we introduce for every position $\vec{p}$ Boolean variables $\mathtt{PNcons}^0_{\vec{p}} \dots \mathtt{PNcons}^6_{\vec{p}}$. $\mathtt{PNcons}^i_{\vec{p}}$ is true if $\vec{p}$ is occupied by a P-monomer, and has exactly i neighbour position occupied by an N-monomer. This is defined by the following reified constraint:

$$(\mathtt{PNcons}^i_{\vec{p}} = 1) \quad \leftrightarrow \quad (\mathtt{Ptype}_{\vec{p}} = 1) \wedge (\mathtt{Nneighs}_{\vec{p}} = i),$$

where $\mathtt{Nneighs}_{\vec{p}}$ is an integer variable with $0 \leq \mathtt{Nneighs}_{\vec{p}} \leq 6$, which is defined by $\mathtt{Nneighs}_{\vec{p}} = \sum_{\vec{p}' \text{ neighbour of } \vec{p}} \mathtt{Ntype}_{\vec{p}'}$. Analogously, we define the variables $\mathtt{NPcons}^i_{\vec{p}}$, $\mathtt{NNcons}^i_{\vec{p}}$, and $\mathtt{PPcons}^i_{\vec{p}}$.

Now we can get an upper bound for `PNContacts` using the following consideration. We count in the variables $\mathtt{NumPNcons}^0, \dots, \mathtt{NumPNcons}^6$ the number of positions occupied by a P-monomer, that have $0, \dots, 6$ N-neighbours, respectively. This is defined by $\mathtt{NumPNcons}^i = \sum_{\vec{p}} \mathtt{PNcons}^i_{\vec{p}}$. Note that in some configuration *con* at some specific search step, not all position types will necessarily be determined. Thus, it is clear that $\mathtt{NumPNcons}^i$ has a range associated. E.g., we could have a configuration where $\mathtt{NumPNcons}^6 = 0$, $\mathtt{NumPNcons}^5 = [0..1]$, $\mathtt{NumPNcons}^4 = [0..2]$, $\mathtt{NumPNcons}^3 = [0..2]$, and so on. For a sequence s containing 4 P-monomers, we could now derive that the best conformation compatible with this configuration can have 1 P-monomer occupying a position with 5 N-neighbours, 2 P-monomer occupying positions with 4 N-neighbours, and the last one occupying a position with 3 N-neighbours. This gives an upper bound of 16 PN-contacts. Note that the more position types are determined (using the constraints), the smaller the ranges get, and henceforth the better the upper bound will be.

Theorem 2.6 *Let con be a configuration, where $[b_i \dots t_i]$ is the range associated to the variable* $\mathtt{NumPNcons}^i$. *Let $n_P(s)$ be the number of P-monomers in s. Let k be the smallest number such that $\sum_{i \geq k} t_i \geq n_P(s)$. Then $U_{PN} = \sum_{i>k}(i \cdot t_i) - k(\sum_{i>k} t_i - t_k)$ is an upper bound on* `PNContacts` *for every conformation compatible with con.*

Analogously, we get an upper bound on `PNContacts` using $\mathtt{NumNPcons}^i$, and lower bounds on `PPContacts` (resp. `NNContacts`) using $\mathtt{NumPPcons}^i$ (resp. $\mathtt{NumNNcons}^i$) (where we start with the lowest neighbour number 0 instead of the highest 6).

3 Results

We have implemented the above described constraints and bounding functions in the constraint programming language Oz 2.0 [12]. We investigate a set of test sequences as shown in Table 1. We select 5 HPNX-sequences and show them together with their corresponging HP-sequences, since we intend to compare the results of the HPNX-sequences with the ones of its HP-sequence. The HP-sequences S1hp through S5hp were treated by [14] before, where they are named L1,...,L5. The algorithm finds the native structure of all sequences listed in Table 1. It is optimized for the search of one best conformation, but we are also able to determine and count all optimal conformations (see Table 2 for detailed results). When comparing the degeneracy of the HPNX-sequences with the corresponding HP-sequences, one can see that the degeneracy is usually strongly reduced in the HPNX-model (except for S4, due to its high percentage of H-monomers). But further, it shows that our algorithm for finding best HPNX-conformations performs significantly better than an algorithm, wich had to go through all HP-optima.

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] 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.

[3] Erich Bornberg-Bauer. Chain growth algorithms for HP-type lattice proteins. In *Proc. of the First Annual International Conferences on Compututational Molecular Biology (RECOMB97)*, pages 47 – 55, Santa Fe, New Mexico, 1997. ACM Press.

[4] 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.

[5] 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.

[6] 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.

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

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

[9] 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.

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

[11] 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, 1998.

[12] 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.

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

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

[15] Kaizhi Yue and Ken A. Dill. Forces of tertiary structural organization in globular proteins. *Proc. Natl. Acad. Sci. USA*, 92:146 – 150, 1995.

A Tables of Results

S1	HXXNNHHHHXPHXHXHHHNHPPHHXPH	S1hp	HPPPPHHHHPPHPHPHHHPHPPHHPPH
S2	HXNNHHHHXHXHHNXNHXHHNHPPXHP	S2hp	HPPPHHHHPHPHHPPPHPHHPHPPPHP
S3	HPHHNXHHPNHHHHXXXHXPXHHHPXH	S3hp	HPHHPPHHPPHHHHPPPHPPPHHHPPH
S4	HHXHHPHHXHHHHHHPNHHHHHPNHHHHHHHH	S4hp	HHPHHPHHPHHHHHHPPHHHHHPPHHHHHHH
S5	XHXNHXXHNPHXXHPXHXXHXNHPXHNXHPXHXPHX	S5hp	PHPPHPPHPPHPPHPPHPPHPPHPPHPPHPPHPPHPPHP

Table 1: Test sequences

Sequence	Search Steps Best HPNX	Search Steps All HP	HPNX Degeneracy	HP Degeneracy
S1	14402	167662	61	37244 [5]
S2	733	2998	4	297
S3	411	155693	195	25554
S4	46	11036	1023	1114
S5	1629	55086	16	3528

Table 2: Results

B Proofs for the lower on `HHContacts`

Let f be some coloring. With $\min_x(f)$ we denote the integer

$$\min\{x \in \mathbb{Z} \mid \exists y \in \mathbb{Z} : f(x,y) = 1\}.$$

$\max_x(f)$, $\min_y(f)$ and $\max_y(f)$ are defined analogously. Furthermore, we define

$$\begin{aligned} \text{length}(f) &= \max_x(f) - \min_x(f) + 1 \\ \text{height}(f) &= \max_y(f) - \min_y(f) + 1. \end{aligned}$$

The pair $(\text{height}(f), \text{length}(f))$ is called the *frame* of f. We say that a point $(x,y) \in \mathbb{Z}^2$ is *within the frame of* f if $\min_x(f) \le x \le \max_x(f)$ and $\min_y(f) \le y \le \max_y(f)$. Given $1 \le i \le \text{height}(f)$, then the i^{th} *row* (denoted $(\text{row}(i,f))$) is the coloring r defined by

$$r(x,y) = \begin{cases} f(x,y) & \text{if } x = \min_y(f) + i - 1 \\ 0 & \text{else.} \end{cases}$$

Furthermore, we define

$$\begin{aligned} \text{indent}_l(i,f) &= \min_x(\text{row}(i,f)) - \min_x(f) \\ \text{indent}_r(i,f) &= \max_x(f) - \max_x(\text{row}(i,f)). \end{aligned}$$

For a row $r = \text{row}(i,f)$ with $1 \le i \le \text{height}(f)$, we write $\text{yval}(r)$ for $\min_y(r)$ $(= \max_y(r))$. The line $y = \text{yval}(r)$ contains all points colored black by the row r. The *leftmost* (resp. *rightmost*) point in a row r is the leftmost (resp. rightmost) point colored black by r, i.e., the point $(\min_x(r), \text{yval}(r))$ (resp. $(\max_x(r), \text{yval}(r))$).

Definition B.1 *The partial order $\preceq$ on caveat-free colorings is defined by $f \preceq f'$ if and only if* $\text{height}(f) = \text{height}(f')$, $\text{length}(f) = \text{length}(f')$ *and* $d(f) = d(f')$

[5] This number differs from the degeneracy as given in [14]. The reason is just the following. We have found 36691 caveat-free conformations and 553 conformations including caveats. In [14], only the number of caveat-free conformations is listed (i.e., 36691).

Proposition B.2 *Let f, f' be two caveat-free colorings with $f \preceq f'$. Then* $\mathrm{HSurf}(f) = \mathrm{HSurf}(f')$.

Proof. Since both f and f' are caveat-free, we know that $\mathrm{HSurf}(f) = 2\mathrm{height}(f) + 2 \times \mathrm{length}(f)$. Similarly, we get $\mathrm{HSurf}(f') = 2 \times \mathrm{height}(f) + 2\mathrm{length}(f')$. Since $f \preceq f'$, we have $\mathrm{height}(f) = \mathrm{height}(f')$ and $\mathrm{length}(f) = \mathrm{length}(f')$, which implies $\mathrm{HSurf}(f) = \mathrm{HSurf}(f')$. □

We will show that every f can be extended to a $\preceq$-maximal coloring f' (which has the same surface by the last proposition). This implies that the surface of $\preceq$-maximal colorings extending f is a lower bound on the surface of f. To calculate the surface of $\preceq$-maximal colorings, we can show, that every $\preceq$-maximal coloring f has a simple form, as e.g. shown in (6).

Definition B.3 *Let f be a caveat-free coloring with $d(f) > 1$. Then f is called* simple *if it satisfies the following conditions: 1.) for all $1 \leq i < \mathrm{height}(f)$ we have*

$$\mathrm{indent}_l(i, f) \neq 0 \vee \mathrm{indent}_l(i+1, f) \neq 0 \Rightarrow |\mathrm{indent}_l(i+1, f) - \mathrm{indent}_l(i, f)| = 1$$
$$\mathrm{indent}_r(i, f) \neq 0 \vee \mathrm{indent}_r(i+1, f) \neq 0 \Rightarrow |\mathrm{indent}_r(i+1, f) - \mathrm{indent}_r(i, f)| = 1.$$

and 2.) the leftmost and the rightmost point of the first and the last row are odd.

Definition B.4 *Let f be a simple coloring with frame $(a, b) = (\mathrm{height}(f), \mathrm{length}(f))$. Then the tuple*

$$(a, b, \mathrm{indent}_l(1, f), \mathrm{indent}_r(1, f), \mathrm{indent}_l(a, f), \mathrm{indent}_r(a, f))$$

is called the characteristics *of f. A tuple $(a, b, i_1, i_2, i_3, i_4)$ is called a* characteristics *if it is the characteristics of some simple coloring.*

First, we show some easy corelation between simple colorings and their characteristics.

Proposition B.5 *A simple coloring f is uniquely determined (up to translation) by its characteristics, i.e., for two simple colorings f, f' having the same characteristics, there is a vector $\vec{v} \in \mathbb{Z}^2$ such that*

$$\forall \vec{p} \in \mathbb{Z}^2 : [(f(\vec{p}) = 1) \Leftrightarrow (f'(\vec{p} + \vec{v}) = 1)].$$

Proof (sketch). E.g., consider the left lower corner. Now Lemma B.9 implies that the rows 1 to $i+1$ have left indents $i, i-1, \ldots, 0$, where $i = \mathrm{indent}_l(1, f)$. The same holds for the other corners. Since f simple, this uniquely determines f (up to translation). □

Proposition B.6 *Let $C = (a, b, i_1, i_2, i_3, i_4)$ be a tuple. Then C is a characteristics if and only if*

1. *$a - i_1 - i_3 \geq 1$, $a - i_2 - i_4 \geq 1$, $b - i_1 - i_2 \geq 1$ and $b - i_3 - i_4 \geq 1$;*
2. *and*
$$\begin{array}{lcl} a \text{ odd} & \Rightarrow & (i_1 \equiv i_3 \mod 2) \wedge (i_2 \equiv i_4 \mod 2) \\ a \text{ even} & \Rightarrow & (i_1 \not\equiv i_3 \mod 2) \wedge (i_2 \not\equiv i_4 \mod 2) \\ b \text{ odd} & \Rightarrow & (i_1 \equiv i_2 \mod 2) \wedge (i_3 \equiv i_4 \mod 2) \\ b \text{ even} & \Rightarrow & (i_1 \not\equiv i_2 \mod 2) \wedge (i_3 \not\equiv i_4 \mod 2) \end{array}$$

Proof (sketch). Claim 1 follows directly from the definition of a characteristics of a simple coloring. Claim 2 follows from the fact that the leftmost and rightmost point of the first and last row must be odd, which implies that the first and last row must have an odd number of points colored black. The same argument can be applied to the first and last column. □

Corollary B.7 *Let $(a, b, i_1, i_2, i_3, i_4)$ be a characteristics. Then the top and bottom point of the first and last column are odd.*

The advantage of a simple coloring f is that one can easily calculate $e(f) + o(f)$ and $d(f)$ out of the characteristics, as shown in Theorem 2.5. For the proof of this theorem we need an additional proposition.

Proposition B.8 *Let f be a connected, caveat-free coloring with* $\text{height}(f) = a$. *Then*

$$\begin{aligned} d(f) = a \; - \; & |\{i \mid \textit{the leftmost point of } \text{row}(i, f) \textit{ is even}\}| \\ - \; & |\{i \mid \textit{the rightmost point of } \text{row}(i, f) \textit{ is even}\}| \end{aligned}$$

Proof. Via induction on $a = \text{height}(f)$. For the base case $a = 1$, it holds trivially. For the induction step, let f be a coloring with $\text{height}(f) = a + 1$. Let f' be the coloring which is generated by deleting the $a + 1^{st}$ row in f. Then $\text{height}(f') = a$, and we get

$$\begin{aligned} d(f') = a \; - \; & |\{i \mid \text{the leftmost point of } \text{row}(i, f) \text{ is even}\}| \\ - \; & |\{i \mid \text{the rightmost point of } \text{row}(i, f) \text{ is even}\}| \end{aligned}$$

by induction hypothesis. Let $r = \text{row}(a + 1, f)$. Then

$$d(f) = d(f') + 1 - \begin{cases} 0 & \text{if the leftmost } \textit{and} \text{ rightmost point of } r \text{ are odd} \\ 2 & \text{if the leftmost } \textit{and} \text{ rightmost point of } r \text{ are even} \\ 1 & \text{else,} \end{cases}$$

which proves the claim. □

Proof of Theorem 2.5. Let f be given as defined by the theorem. Then $e(f) + o(f)$ is just the number of points $\vec{p} \in \mathbb{Z}^2$ with $f(\vec{p}) = 1$. But this is exactly $a \times b$ minus the points that are excluded at the corners. Given the indents $i_1, \ldots, i_4$, we get that we exclude exactly

$$\sum_{j=1}^{4} \frac{i_j(i_j + 1)}{2}$$

points at the corners.

For proving that $d(f) = \frac{i_1+i_2+i_3+i_4}{2} + 1$, we have to count the number of times the starting (resp. end point) of the $\text{row}(i, f)$ is even (according to Proposition B.8). This happens only if $\text{indent}_l(i, f)$ (resp. $\text{indent}_r(i, f)$) is zero. Now there are $a - i_1 - i_3$ integers i with $\text{indent}_l(i, f) = 0$, and $a - i_2 - i_4$ integers i with $\text{indent}_r(i, f) = 0$. Since they all have indent 0, one can see that exactly every second row starts or ends with an even point. Furthermore, Corollary B.7 guarantees that

1. $a - i_1 - i_3$ and $a - i_2 - i_4$ are both odd; and
2. that there are more i's with $\text{indent}_l(i, f) = 0$ that start with an odd monomer.

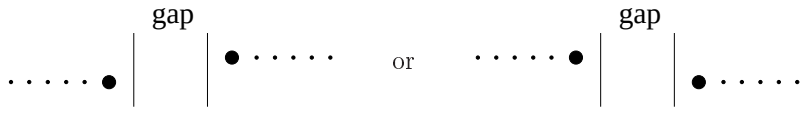
The same holds for the right side. Hence, we get

$$\begin{aligned} d(f) &= a - \frac{a - i_1 - i_3 - 1}{2} - \frac{a - i_2 - i_4 - 1}{2} \\ &= a - \frac{a - i_1 - i_3 - 1 + a - i_2 - i_4 - 1}{2} \\ &= a - \frac{2a - 2 - i_1 - i_3 - i_2 - i_4}{2} \; = \; \frac{i_1 + i_2 + i_3 + i_4}{2} + 1 \end{aligned}$$

□

The remaining part is to show that a $\preceq$-maximal coloring is simple. We will first show that this holds for a subclass of caveat-free colorings, namely connected colorings.

We say that a coloring f is *connected* if there is no i such that there is a gap between the i^{th} and $i+1^{st}$ row, i.e., they have the form



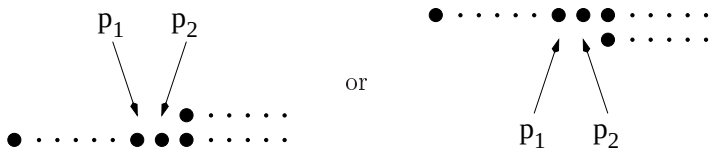
Lemma B.9 *Let f be a connected, caveat-free coloring with $d(f) > 1$ that is $\preceq$-maximal. Then f is simple.*

Proof. First, we show that Condition B.3 is satisfied by every $\preceq$-maximal coloring. Suppose that f does not satisfy (7). Then there is some $1 \leq i < \text{height}(f)$ with

$$\text{indent}_l(i, f) \neq 0 \wedge |\text{indent}_l(i+1, f) - \text{indent}_l(i, f)| \neq 1.$$

We distinguish the following cases:

1. $|\text{indent}_l(i+1, f) - \text{indent}_l(i, f)| > 1$. Since f is connected, the i^{th} and $i+1^{st}$ row of f have the form



with positions $p_1 = (x_1, y_1)$ and $p_2 = (x_1+1, y_1)$ (for some x_1, y_1) being free. Now p_1 and p_2 are positions with distance 1, which implies that they have different parities. Define f' by

$$f'(x,y) = \begin{cases} 1 & \text{if } (x,y) = p_1 \text{ or } (x,y) = p_2 \\ f(x,y) & \text{else.} \end{cases}$$

Since f is caveat-free, we know that f' is also caveat-free. Furthermore, we know that $\text{length}(f) = \text{length}(f')$ and $\text{height}(f) = \text{height}(f')$. Since p_1 and p_2 have different parity, we know that

$$e(f') = e(f) + 1 \quad \text{and} \quad o(f') = o(f) + 1.$$

Hence, $d(f) = d(f')$, which implies $f \prec f'$. But this is a contradiction to the $\preceq$-maximality of f.

2. $\text{indent}_l(i+1, f) = \text{indent}_l(i, f)$. This case can be reduced to the previous one by rotating f by 90°.

The case that f does not satisfy (7) is analogous.

Now suppose that f does not satisfy the Condition B.3. Let $a = \text{height}(f)$, $b = \text{length}(f)$, $i_1 = \text{indent}_l(1, f)$, $i_2 = \text{indent}_r(1, f)$, $i_3 = \text{indent}_l(a, f)$ and $i_3 = \text{indent}_r(a, f)$ After possibly applying reflections, we can assume that the leftmost point of the first row is even. We distinguish the following cases:

1. $i_1 \neq 0$. Let $r = \text{row}(1, f)$ and define

$$p_1 = (\min_x(r) - 1, \text{yval}(r))$$

(the point left to the leftmost point of r). Then p_1 is an odd point and within the frame of f. Since $d(f) > 1$, Proposition B.8 implies that there must a j such that the row $r' = \text{row}(j, f)$ starts or ends with an odd point, and has non-empty indent. If row r' starts with an odd point, then take

$$p_2 = (\min_x(r') - 1, \text{yval}(r')),$$

otherwise define

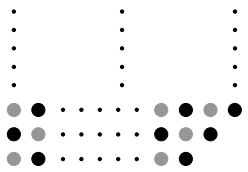
$$p_2 = (\max_x(r') + 1, \mathrm{yval}(r')).$$

Then p_2 is an even point which is within the frame of f. Define f' by

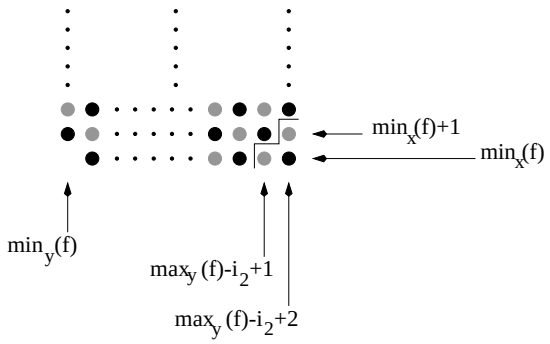
$$f'(p) = \begin{cases} f(p) & \text{if } p \neq p_1 \text{ or } p \neq p_2 \\ 1 & \text{else} \end{cases}$$

Then f' is caveat-free and connected with $f \prec f'$, which is a contradiction.

2. $i_1 = 0$. Since $d(f) > 1$, we know that by Proposition B.8 that not all of i_2, i_3, i_4 can be lower or equal 1. Suppose that $i_2 > 2$. By the last case, we can assume that last point of the first row is odd. Hence, the first three rows of f are of the form



where again black beads indicated odd positions (x, y) with $f(x, y) = 1$, and grey beads represent even positions. Let f' be the coloring which is f except for the first three rows, where f' is of the form



I.e., f' is defined by

$$f'(x, y) = \begin{cases} 0 & \text{if } (x, y) = (\min_x(f), \min_y(f)) \\ 1 & \text{if } (x, y) = (\min_x(f), \max_y(f) - i_2 + 1) \\ 1 & \text{if } (x, y) = (\min_x(f), \max_y(f) - i_2 + 2) \\ 1 & \text{if } (x, y) = (\min_x(f) + 1, \max_y(f) - i_2 + 2) \\ f(x, y) & \text{else} \end{cases}$$

It is easy to check that $d(f) = d(f')$. Since we didn't change the height or length of f, and since we have added two points, this implies

$$f \prec f'.$$

But this is a contradiction to the $\preceq$-maximality of f. The other cases $i_3 > 2$ and $i_4 > 2$ are analogous.

□

We can even further restrict the characteristics of $\preceq$-maximal colorings.

Lemma B.10 *Let f be a connected, $\preceq$-maximal coloring such that $d(f) > 1$. Then f has a characteristics $(a, b, i_1, i_2, i_3, i_4)$ such that*

$$\forall k, l \in [1..4] : |i_k - i_l| \leq 2.$$

Proof. Let f be $\preceq$-maximal with characteristics $(a, b, i_1, i_2, i_3, i_4)$. Assume that f does not satisfy the condition of the lemma. I.e., there is a i_k and i_l with $k \neq l \in [1..4]$ such that

$$i_l < i_k - 2$$

After applying possibly reflection or rotation, we can assume that

$$i_1 = \min\{i_1, i_2, i_3, i_4\}.$$

and that there is an $i_k \in [2..4]$ with $i_1 < i_k - 2$. Note that by definition of characteristics, $a - i_1 - i_3 \geq 1$ and $b - i_1 - i_2 \geq 1$. We distinguish the following cases:

1. $a - i_1 - i_3 > 1$ and $b - i_1 - i_2 > 1$. By Condition B.3 and Corollary B.7, this implies that $a - i_1 - i_3 \geq 3$ and $b - i_1 - i_2 \geq 3$.

   Suppose that i_2 satisfies $i_1 < i_2 - 2$. Consider $C = (a, b, i_1', i_2', i_3, i_4)$ with $i_1' = i_1 + 2$ and $i_2' = i_2 - 2$. By Proposition B.6 we know that C is a characteristics, which implies that there is some simple coloring f' having characteristics C. By Lemma 2.5, we know that

   $$d(f') = \frac{(i_1 + 2) + (i_2 - 2) + i_3 + i_4}{2} + 1 = \frac{i_1 + i_2 + i_3 + i_4}{2} + 1 = d(f).$$

   Since f and f' have the same length and height, we need only to show that $e(f') + o(f') > e(f) + o(f)$ for showing that $f \prec f'$. But this is equivalent to show that

   $$nd(f', f) = e(f') + o(f') - e(f) + o(f) > 0.$$

   By Lemma 2.5, we get

   $$\begin{aligned}
   nd(f', f) &= a \times b - (\frac{i_1'(i_1' + 1)}{2} + \frac{i_2'(i_2' + 1)}{2} + \frac{i_3(i_3 + 1)}{2}) + \frac{i_4(i_4 + 1)}{2} \\
   &- (a \times b + \sum_{j=1}^{4} \frac{i_j(i_j + 1)}{2}) \\
   &= \frac{-(i_1 + 2)(i_1 + 3) - (i_2 - 2)(i_2 - 1) + i_1(i_1 + 1) + i_2(i_2 + 1)}{2} \\
   &= \frac{-(i_1^2 + 5i_1 + 6) - (i_2^2 - 3i_2 + 2) + i_1^2 + i_1 + i_2^2 + i_2}{2} \\
   &= \frac{-4i_1 + 4i_2 - 8}{2} \\
   &= 2((i_2 - 2) - i_1) \\
   &> 0 \quad \text{(since } i_1 < i_2 - 2 \text{ by assumption)}
   \end{aligned}$$

   Hence, $f \prec f'$, which is a contradiction. The other cases $i_1 < i_3 - 2$ and $i_1 < i_4 - 2$ are analogous.

2. $a - i_1 - i_3 = 1$ and $b - i_1 - i_2 > 1$. Note that by condition $a - i_1 - i_3 = 1$, we cannot just enlarge i_1 without simultaneously decreasing i_3 by the same value. Hence, we can consider only characteristics of the forms

   $$(a, b, i_1 + k, i_2 + l, i_3 - k, i_4 - l) \quad \text{or} \quad (a, b, i_1 + k, i_2 - l, i_3 - k, i_4 + l)$$

   We distinguish the following cases:

(a) $i_1 < i_3 - 2$. We can then show that there is an f' with $f \prec f'$ by considering the characteristics $(a, b, i_1 + 2, i_2, i_3 - 2, i_4)$ similar to the previous case.

(b) $i_1 \geq i_3 - 2 \wedge (i_1 < i_2 - 2) \vee (i_1 < i_4 - 2)$.
Suppose that $i_1 < i_2 - 2$. Since $a - i_1 - i_3 = 1$ we get $i_3 = a - i_1 - 1$. Furthermore, we know that $a - i_2 - i_4 \leq 1$, which implies

$$i_4 \leq a - i_2 - 1 < a - (i_1 + 2) - 1 = i_3 - 2 \tag{7}$$

Consider the tuple $C = (a, b, i_1+1, i_2-1, i_3-1, i_4+1)$, which is a characteristics by Proposition B.6. Hence, there is a simple f' having the characteristics C. We get again $d(f) = d(f')$, and we have to show that

$$nd(f', f) = e(f') + o(f') - e(f) + o(f) > 0.$$

By Lemma 2.5, we get

$$\begin{aligned}
nd(f', f) &= \frac{-(i_1+1)(i_1+2) - (i_2-1)i_2 - (i_3-1)i_3 - (i_4+1)(i_4+2)}{2} \\
&+ \frac{i_1(i_1+1) + i_2(i_2+1) + i_3(i_3+1) + i_4(i_4+1)}{2} \\
&= \frac{-i_1^2 - 3i_1 - 2 - i_2^2 + i_2 - i_2^2 + i_3 - i_4^2 - 3i_4 - 2}{2} \\
&+ \frac{i_1^2 + i_1 + i_2^2 + i_2 + i_3^2 + i_3 + i_4^2 + i_4}{2} \\
&= \frac{-2i_1 + 2i_2 + 2i_3 - 2i_4 - 4}{2} \\
&= i_2 - i_1 + i_3 - i_4 - 2 \\
&> 2 + 2 - 2 = 2 \quad \text{since } i_1 < i_2 - 2 \text{ and } i_4 < i_3 - 2 \text{ by (7)} .
\end{aligned}$$

which shows that $f \prec f'$.
The case that $i_1 < i_4 - 2$ can be proved analogous to the case that $i_1 < i_2 - 2$. We can then show that $i_2 < i_3$, and prove the existence of an f' with $f \prec f'$ by using the characteristics $(a, b, i_1 + 1, i_2 + 1, i_3 - 1, i_4 - 1)$.

(c) $a - i_1 - i_3 = 1$ and $b - i_1 - i_2 = 1$. Analogous to the previous case.

□

Finally, we have to treat unconnected, caveat-free colorings, and colorings containing caveats. For simplicity, we will only sketch the proofs for these kind of colorings.

Lemma B.11 *Let f be a caveat-free coloring that is not connected. Then there is a coloring f' with $f \prec f'$.*

Proof (sketch). Let f have frame (a, b). If one has n unconnected subparts with frames $(a_1, b_1), \ldots, (a_n, b_n)$, then the caveat-freeness of f implies that

$$a = a_1 + \ldots + a_n \quad \text{and} \quad b = b_1 + \ldots + b_n.$$

Then one can show that one finds always a characteristics for the frame (a, b) which has the same difference than the sum of differences of the characteristics of the subparts of f, but which has more points colored black. □

We can even show that we will always find a caveat-free coloring with minimal surface, but we will skip the proof.

Lemma B.12 *Let f be a coloring that is not caveat-free. Then there is some caveat-free coloring f' such that $d(f') = d(f)$, $\mathrm{HSurf}(f') \leq \mathrm{HSurf}(f)$, and $e(f') + o(f') \geq e(f) + o(f)$.*