Available online at www.sciencedirect.com

SCIENCE DIRECT®

Computer Physics Communications 166 (2005) 8–16

Computer Physics Communications

www.elsevier.com/locate/cpc

Exact enumeration of three-dimensional lattice proteins

Reinhard Schiemann, Michael Bachmann, Wolfhard Janke *

Institut für Theoretische Physik, Universität Leipzig, Augustusplatz 10/11, 04109 Leipzig, Germany

Received 22 September 2004; accepted 23 September 2004

Available online 8 December 2004

Abstract

We present an algorithm for the exhaustive enumeration of all monomer sequences and conformations of short lattice proteins as described by the hydrophobic-polar (HP) model. The algorithm is used for an exact identification of all designing sequences of HP proteins consisting of up to 19 monomers whose conformations are represented by interacting self-avoiding walks on the simple cubic lattice. Employing a parallelized implementation on a Linux cluster, we generate the complete set of contact maps of such walks.

PACS: 05.10.-a; 87.15.Aa; 87.15.Cc

Keywords: Lattice proteins; HP model; Contact maps; Self-avoiding walks; Polymers

1. Introduction

The numerical treatment of protein models is highly nontrivial. On one hand, the design of realistic models suffers from the fact that the atomic interactions among the constituents of proteins and with their aqueous cellular environment are by no means well understood [1]. On the other hand, the computational effort increases drastically with the length of the molecules. Therefore, significant simplifications of the realistic situation have to be introduced in order to facilitate a detailed analysis based on computational methods and, in particular, to allow studies of the relation between sequence and conformation spaces of model proteins.

Herein, we will consider two versions of the very simple HP lattice model [2,3] which makes the following assumptions: Instead of considering all 20 different kinds of amino acids that occur in real proteins the model comprises only two prototypes of residues: hydrophilic (or polar, P) and hydrophobic (H) monomers, respectively. This is to account for the fact that most of the naturally occurring amino acids can be classified in that way [1, p. 154]. Also the atomar interactions are drastically simplified. Short-range repulsion between monomers is taken into account by modeling the conformations of HP proteins

* Corresponding author.
E-mail addresses: Reinhard.Schiemann@itp.uni-leipzig.de (R. Schiemann), Michael.Bachmann@itp.uni-leipzig.de (M. Bachmann), Wolfhard.Janke@itp.uni-leipzig.de (W. Janke).

0010-4655/$ – see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.cpc.2004.09.011

as *self-avoiding walks* on regular lattices. The simple cubic (sc) lattice was used in this study. In addition one considers in the most simple formulation of the model exclusively a nearest-neighbor attractive interaction between hydrophobic residues non-adjacent in the polymer chain [2]. Slightly more involved variants also take into account nearest-neighbor contacts between *HP* and/or *PP* pairs [3]. This is an effective way of describing the interaction of the molecule with the aqueous environment [4].

Exact enumeration results obtained for short HP proteins can be used as cross-checks for other non-exact methods that search the conformational and sequence spaces of proteins. These include Monte Carlo and genetic algorithms (e.g. [5,6]), generalized ensemble techniques (e.g. [7]), chain growth algorithms (e.g. [8,9]), and combinations thereof (e.g. [10,11]). More importantly, the complete treatment of all sequences and conformations allows one to carry out systematic statistical analyses of HP proteins. Our results for the sc lattice described in more detail in Ref. [12] complement prior exact enumeration studies on the square lattice [13] and for HP proteins with conformations *restricted* to regular cuboids on the sc lattice [14,15].

In the next section we introduce the HP models used here in a little more formal way. Section 3 explains the exact enumeration procedure in terms of which our results are obtained. The concept of exact enumeration is first illustrated with a naive implementation. What remains of Section 3 is dedicated to improvements of that simple implementation and describes how these improvements apply to a simple example case. In Section 4 we show how our exact results can be applied for a comparison of the numbers of self-avoiding walks and contact matrices and for the determination of designing sequences in the HP model. Section 5 concludes this article with a summary and an outlook on further statistical analyses based on the results of the enumerations presented here.

2. HP models

An HP protein is defined by its sequence of monomers. We will denote the type of monomer by σ_i, where $i = 1, \ldots, N$ is the position of the monomer in a polymer chain of length N and by convention $\sigma_i \in \{0 \widehat{=} P, 1 \widehat{=} H\}$. Its conformation, which is a self-avoiding walk on the lattice (with lattice spacing $a = 1$), is represented by an ordered collection of lattice vectors that contain the positions of the residues: $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$. The distance between monomers i and j is denoted by $x_{ij} = |\mathbf{x}_i - \mathbf{x}_j|$. The attractive interaction between pairs of residues is short-ranged on the underlying lattice. It is considered only between residues that are on nearest-neighbor positions but not covalently bound in the molecular chain. Such a pair of residues is said to be in *contact*. This is expressed by the following energy function that is assigned to each HP protein:

$$E = \sum_{i,j>i+1} C_{ij}\, U_{\sigma_i \sigma_j}, \tag{1}$$

where $C_{ij} = (1 - \delta_{i+1\,j})$ for $x_{ij} = 1$, and zero otherwise, is a symmetric $N \times N$ matrix called *contact map* and

$$U_{\sigma_i \sigma_j} = \begin{pmatrix} u_{HH} & u_{HP} \\ u_{HP} & u_{PP} \end{pmatrix} \tag{2}$$

is the 2×2 interaction matrix.

In the present study, the HP model comes in two versions that are different from one another in the way attractive interactions between the amino acids are considered. In the original version of the model [2], which we will refer to as HP model in the following, only a pair of hydrophobic residues in contact contributes to the energy function (1) and the only non-zero entry in the interaction matrix is $u_{HH}^{\mathrm{HP}} = -1$. A modification of the model [3] also takes into account an interaction between hydrophobic and polar monomers. We call it the MHP (mixed HP) model. Its interaction matrix entries read $u_{HH}^{\mathrm{MHP}} = -1$, $u_{HP}^{\mathrm{MHP}} = -1/2.3 \approx -0.435$, and $u_{PP}^{\mathrm{MHP}} = 0$. The magnitude of u_{HP}^{MHP} is motivated by an analysis of inter-residue contact energies between different types of real amino acids [4].

A sequence of monomers is called a *designing sequence* if there exists exactly one conformation (up to trivial symmetries, to be explained in more detail below) for its state of lowest energy. The interest in designing sequences is based on a generally accepted biochemical principle that sequence specifies conformation, and, in turn, the conformation of a polymer determines its biological function. Accepting this principle also in the framework of the highly simplified HP model leads directly to the concept of designing sequences. The conformation of the lowest-energy state is uniquely determined for designing sequences only. Furthermore, the number of designing sequences is very small compared to the total number of 2^N HP sequences of a given chain length N. Thus, the ability of identifying designing sequences may be seen as a benchmark for algorithms that search the sets of conformations *and* sequences of HP proteins.

3. Exact enumeration

3.1. Naive implementation

A straightforward method of identifying designing sequences of a given length is to perform an exact enumeration. This means to run through the whole set of sequences and for each sequence through the whole set of conformations. Consider such a deliberately naive enumeration for short sequences of length $N = 4$ in the HP model. Trivially, there are $2^4 = 16$ different sequences and $6 \times 5 \times 5 = 150$ self-avoiding walks on the sc lattice.

Up to symmetries, Fig. 1 shows all conformations for $N = 4$. Only the conformation designated by FLL has a contact between its first and last residues. Consequently, all four HP sequences with a hydrophobic monomer in the first and last positions of the sequence must be designing: there is only one conformation for the lowest energy $E = -1$. All other sequences are non-designing since the energy $E = 0$ is obviously degenerate.

When increasing the number of monomers N by one, the number of sequences doubles. Also, the number of self-avoiding walks (SAW) is known to increase asymptotically by a factor of $\mu_{\mathrm{SAW}} \approx 4.684$ [16–18]. Consequently, the computational effort scales roughly as 9.37^N, i.e. exponentially fast with the chain length N. This is why improvements of the naive enumeration become necessary even for rather short chain lengths.

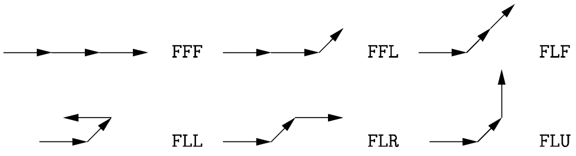


Fig. 1. All relevant conformations for $N = 4$ together with their configurational chain codes explained in the text.

3.2. Improvements

Some of these improvements are very obvious. Firstly, there is no point in carrying out the enumeration for two sequences that contain the same monomers but in reverse order. For example, the sequences HPPP and PPPH are equivalent. Simply counting the number of relevant sequences, R_N, that have to be considered in the enumeration yields

$$R_N = 2^{N-1} + \begin{cases} 2^{\frac{N}{2}-1} & \text{if } N \text{ even,} \\ 2^{\frac{N-1}{2}} & \text{if } N \text{ odd.} \end{cases} \tag{3}$$

3.2.1. Symmetries on the sc lattice

Furthermore, not all self-avoiding walks are independent but related to each other by symmetry operations. On the sc lattice, there are six directions for the first bond of any conformation. Fixing the direction of the first bond, we are left with $(1/6) f_{\mathrm{s}}(\mathbf{X})$ mutually symmetric conformations, where $f_{\mathrm{s}}(\mathbf{X})$ is the total number of mutually symmetric self-avoiding walks starting from the origin. Fig. 2 illustrates these symmetries for the conformation encoded by FLU in Fig. 1. For a given conformation $\mathbf{X}$, the symmetry factor is given by

$$f_{\mathrm{s}}(\mathbf{X}) = \begin{cases} 6 & \text{if } \mathbf{X} \text{ linear,} \\ 24 & \text{if } \mathbf{X} \text{ planar,} \\ 48 & \text{otherwise.} \end{cases} \tag{4}$$

We represent conformations by means of chain codes that encode the steps of self-avoiding walks in terms of a sequence of relative moves. On the sc lattice there are five kinds of such moves which we denote by F (“forward”), L (“left”), R (“right”), U (“up”), and D (“down”). The chain codes for all independent conformations consisting of four monomers are shown in Fig. 1. Two vectors are needed in order to define the five moves on the sc lattice: Let $\mathbf{o}_i$ be a unit vector attached to the monomer at $\mathbf{x}_i$ and $\mathbf{s}_i$ another unit vector at $\mathbf{x}_i$ perpendicular to $\mathbf{o}_i$ determining the direction of the ith step of the self-avoiding walk as shown in Fig. 3. Given a chain code, we determine the conformation $\mathbf{X}$ by initially choosing $\mathbf{x}_1 = (0, 0, 0)$, $\mathbf{o}_1 = (0, 0, 1)$, $\mathbf{s}_1 = (0, 1, 0)$ and by specifying how to go over from $\{\mathbf{x}_i, \mathbf{o}_i, \mathbf{s}_i\}$ to $\{\mathbf{x}_{i+1}, \mathbf{o}_{i+1}, \mathbf{s}_{i+1}\}$ for each

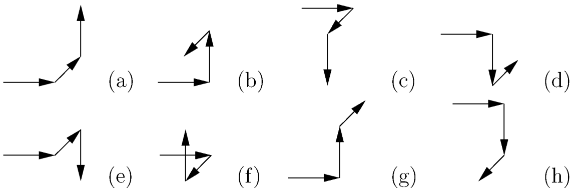


Fig. 2. All $(1/6) f_s(\mathbf{X}) = 8$ conformations are symmetric on the sc lattice and their first bonds show into the same direction. Symmetry operations applied to the conformation (a) are rotations about the first bond (b, c, d), reflections at a lattice plane (e, f), and two compositions of rotations and reflections (g, h).

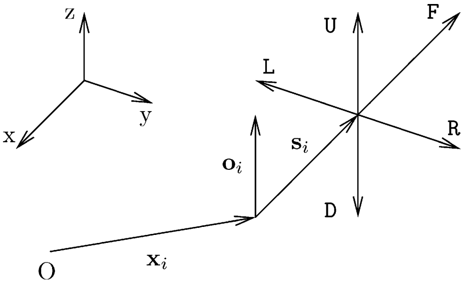


Fig. 3. Encoding a conformation in terms of relative moves on the lattice. Vectors designated by F, L, R, U, D are $\mathbf{s}_{i+1}(\mathsf{F})$, $\mathbf{s}_{i+1}(\mathsf{L})$, etc.

of the five moves:

$$\mathbf{o}_{i+1} = \begin{cases} \mathbf{o}_i & \text{for moves F, L, R,} \\ -\mathbf{s}_i & \text{for U,} \\ \mathbf{s}_i & \text{for D,} \end{cases} \tag{5}$$

$$\mathbf{s}_{i+1} = \begin{cases} \mathbf{s}_i & \text{for F,} \\ \mathbf{o}_i \times \mathbf{s}_i & \text{for L,} \\ -\mathbf{o}_i \times \mathbf{s}_i & \text{for R,} \\ \mathbf{o}_i & \text{for U,} \\ -\mathbf{o}_i & \text{for D,} \end{cases} \tag{6}$$

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \mathbf{s}_i. \tag{7}$$

Eqs. (5), (6), and (7) can be read off from Fig. 3.

In exact enumeration, it is not desirable to enumerate conformations that are symmetric to each other. This is easily achieved by enumerating the chain codes for conformations of a given length N considering only codes that satisfy a chosen set of rules. The choice of $\mathbf{x}_1$, $\mathbf{o}_1$, and $\mathbf{s}_1$ determines the first move which we call by convention F. Furthermore, we require the first move that makes the walk deviate from a linear conformation to be an L-move and, subsequently, we require the first step into the third coordinate direction to be a U-move. For conformations of length $N = 4$ modeled as self-avoiding walks of three steps there are six different chain codes obeying these rules for not mutually symmetric conformations (see again Fig. 1).

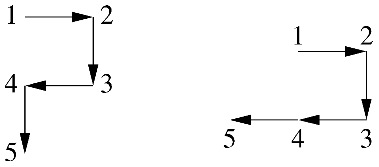


Fig. 4. Two different conformations which have a single contact between their first and fourth monomers. Both belong to the same contact map.

3.2.2. Contact maps

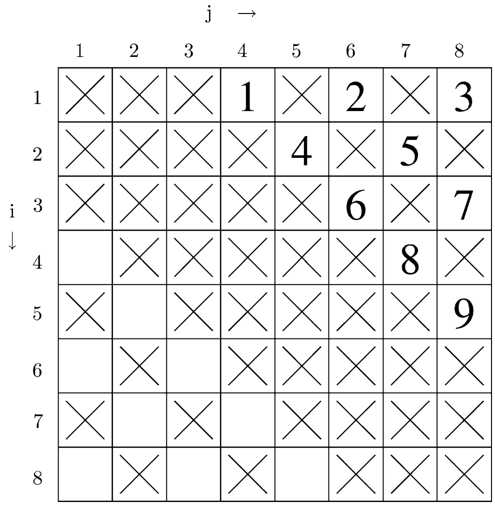
As defined in (1), the energy of an HP protein does not depend explicitly on its conformation but only on the information of which pairs of monomers form contacts. This information is contained in the contact map. In general, more than one conformation corresponds to a given contact map (see Fig. 4). Therefore, it is possible to improve exact enumeration in terms of contact maps: First, all self-avoiding walks of a given length are enumerated *once* in order to generate the complete set of contact maps. In a second step, designing sequences are identified by running through the set of contact maps for each sequence. A sequence is identified as designing if there is exactly one contact map that corresponds to the lowest energy and, in turn, if there is exactly one self-avoiding walk corresponding to that contact map.

The first step of this enumeration procedure requires all contact maps to be stored in memory. Some straightforward properties of contact maps can be used in order to occupy as little memory as possible. Apart from symmetry ($C_{ij} = C_{ji}$) and the trivial facts that self-contacts are not meaningful ($C_{ii} = 0$) and that, by definition, covalently bound monomers are not counted as contacts ($C_{ij} = 0$ if $|i - j| = 1$), these properties are:

$$C_{ij} = 0 \quad \text{if } |i - j| = 2n,\ n = 1, 2, \ldots, \tag{8}$$

$$\sum_{i=1}^{N} C_{ij} \leqslant \begin{cases} 5 & \text{if } j = 1 \text{ or } j = N, \\ 4 & \text{otherwise,} \end{cases} \tag{9}$$

which are easily seen to be consequences of the considered sc lattice geometry. Fig. 5 illustrates these properties for the $N = 8$ case. There are $Z_8 = 9$ pos-



j →

i ↓	1	2	3	4	5	6	7	8
1	×	×	×	1	×	2	×	3
2	×	×	×	×	4	×	5	×
3	×	×	×	×	×	6	×	7
4		×	×	×	×	×	8	×
5	×		×	×	×	×	×	9
6		×		×	×	×	×	×
7	×		×		×	×	×	×
8		×		×		×	×	×

Fig. 5. Properties of contact maps of conformations consisting of $N = 8$ monomers. Contacts that correspond to entries that are crossed out cannot be formed by conformations on the sc lattice. There are nine pairs of residues that can possibly be in contact (indexed fields).

sible contacts for conformations of that length, i.e. nine bits are required to store any such contact map in memory. In general, this number can be calculated to be

$$Z_N = \begin{cases} \frac{1}{4}(N-2)^2 & \text{if } N \text{ even}, \\ \frac{1}{4}(N-3)(N-1) & \text{if } N \text{ odd}. \end{cases} \quad (10)$$

For each contact map C, we also accumulate the number $g_c(C)$ of self-avoiding walks corresponding to that contact map. In the determination of $g_c(C)$ the trivial symmetries described above are automatically excluded. We include them in a separate quantity $g_s(C)$ given by $\sum f_s(\mathbf{X})$, where the sum runs over all $g_c(C)$ conformations that correspond to C and $f_s(\mathbf{X})$ is given by (4). Knowledge of $g_s(C)$ is necessary for the calculation of thermodynamic quantities and we store it for each contact map, too. Furthermore, we retain for each contact map C the last chain code that we enumerate and whose conformation corresponds to C. In particular, this yields all conformations corresponding to contact matrices with $g_c(C) = 1$ which allows to determine the ground-state conformations of designing sequences.

3.2.3. *Parallelization*

The number of contact maps that can be simultaneously held in memory was increased by distributing them over several individual processors (IPs). We implemented the corresponding program according to the Message Passing Interface (MPI) standard [19] and executed it on the local Linux cluster *Hagrid*,[1] consisting of 40 Athlon 1800+ MHz processors with 100 Mbit Ethernet communication. Fig. 6 shows the basic structure of the program. The master process P_1 generates all self-avoiding walks $\mathbf{X}$ and the corresponding contact maps $C(\mathbf{X})$. For each self-avoiding walk, it can behave in three different ways:

(1) If the memory associated to P_1 is not yet filled up and $C(\mathbf{X})$ was not stored in this memory partition before, $C(\mathbf{X})$ will be appended to the list of contact maps stored in P_1.
(2) If $C(\mathbf{X})$ is already stored in P_1 the corresponding counters $g_c(C)$ and $g_s(C)$ will be increased by one and $f_s(\mathbf{X})$, respectively.
(3) If $C(\mathbf{X})$ is not stored in P_1 and its memory is completely filled, $C(\mathbf{X})$ will be stored in the master's output buffer O_1. When the output buffer is filled up all contact maps in the buffer will be transferred to the next IP's input buffer I_2.

This way of storing contact maps is termed *selective insertion* in Fig. 6. The behavior of the slave processes $P_2, P_3, \ldots$ is very similar. The difference is that their input buffers serve as sources of contact maps, they do not perform any kind of enumeration. Their mere purpose is lookup and storage of contact maps.

The second step of the enumeration, i.e. going through all contact maps for all sequences, can be trivially parallelized in order to reduce the running time. We achieved this by simply distributing the set of contact maps over all IPs. Then, each IP performs the enumeration with respect to its subset of contact maps. Finally, all IPs send their enumeration results to the master process which compares the lowest energies that were found by the slaves in order to find the "globally" minimal energies and the correct degeneracies g_c and g_s. The speed-up factor due to this parallelization

[1] http://www.physik.uni-leipzig.de/Computer/Hagrid.

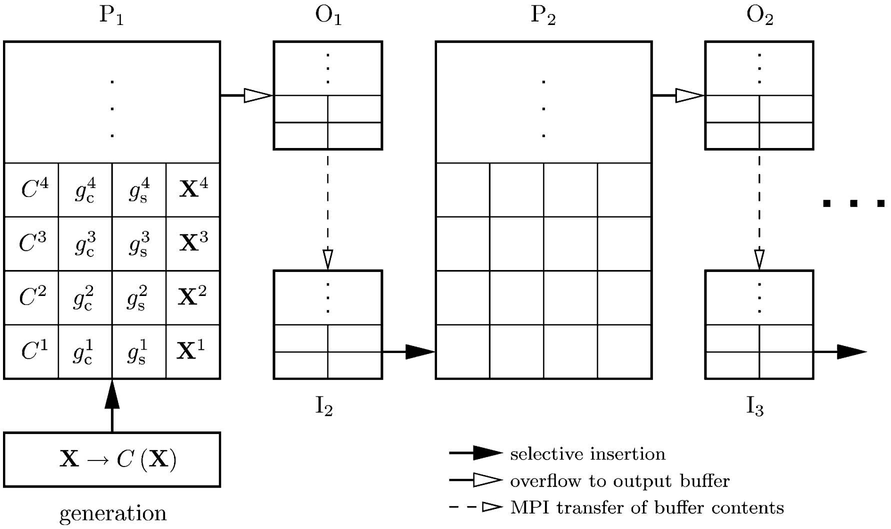

P1
O1
P2
O2
C^4 g_c^4 g_s^4 $\mathbf{X}^4$
C^3 g_c^3 g_s^3 $\mathbf{X}^3$
C^2 g_c^2 g_s^2 $\mathbf{X}^2$
C^1 g_c^1 g_s^1 $\mathbf{X}^1$
I2
I3
$\mathbf{X} \to C(\mathbf{X})$
generation
selective insertion
overflow to output buffer
MPI transfer of buffer contents

Fig. 6. Generating contact maps in terms of parallel distributed programming. Two individual processors (IPs), P_1 and P_2, are shown. Each IP but P_1 disposes of an input buffer I_i, and there is an output buffer O_i for all IPs but the last one. Buffering allows for faster information communication as less MPI messages have to be sent between the IPs.

is virtually equal to the number of available processors.

3.3. A simple example

In the following, we illustrate briefly how the improvements discussed above apply to the very simple $N = 4$ example. There are $R_4 = 10$ relevant sequences which we store in the array

$$\mathbf{S} = [\mathsf{PPPP}, \mathsf{PPPH}, \mathsf{PPHP}, \mathsf{PPHH}, \mathsf{PHPH}, \mathsf{PHHP}, \mathsf{PHHH}, \mathsf{HPPH}, \mathsf{HPHH}, \mathsf{HHHH}]. \tag{11}$$

Also, there are six relevant conformations as shown in Fig. 1. In the first enumeration step, all contact maps are determined. Five conformations (FFF, FFL, FLF, FLR, and FLU) have no contacts and belong to the trivial empty contact map which we will call $C^{(0)}$. The conformation encoded by FLL has a single contact between its first and fourth residues; we refer to its contact map as $C^{(1)}$. Thus, there are only two contact maps with $g_c(C^{(0)}) = 5$ and $g_c(C^{(1)}) = 1$, and we compute $g_s(C^{(0)}) = f_s(\mathsf{FFF}) + f_s(\mathsf{FFL}) + f_s(\mathsf{FLF}) + f_s(\mathsf{FLR}) + f_s(\mathsf{FLU}) = 126$ and $g_s(C^{(1)}) = f_s(\mathsf{FLL}) = 24$.

In the second step, we run through all sequences from (11) for each of the two contact maps. The enumeration requires four more arrays of length $R_4 = 10$ in order to store the lowest energy, $\mathbf{E}$, the accumulated degeneracies, $\mathbf{G}_c$ and $\mathbf{G}_s$, and an example conformation for each sequence, $\mathbf{W}$. After evaluation of the energy function (1) for all sequences with respect to $C^{(0)}$ these arrays read

$$\mathbf{E} = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], \tag{12}$$

$$\mathbf{G}_c = [5, 5, 5, 5, 5, 5, 5, 5, 5, 5], \tag{13}$$

$$\mathbf{G}_s = [126, 126, 126, 126, 126, 126, 126, 126, 126, 126], \tag{14}$$

$$\mathbf{W} = [\mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}]. \tag{15}$$

Calculating now the energies with respect to $C^{(1)}$ yields once more $E = 0$ for the first seven sequences and a lower energy $E = -1$ for the last three sequences in (11). The arrays have to be updated accordingly. This means that for the sequences with energy $E = 0$ the counter for the conformations, $\mathbf{G}_c$, is incremented, while it is reset for the other three sequences, since their energies are lower now. The degeneracy $\mathbf{G}_s$ which includes the symmetry factors for the conformations is accumulated appropriately and the new conformations possessing lower energies than those in the previous step (15) are stored in $\mathbf{W}$:

$$\mathbf{E} = [0, 0, 0, 0, 0, 0, 0, -1, -1, -1], \tag{16}$$

$$\mathbf{G}_{\mathrm{c}} = [6, 6, 6, 6, 6, 6, 6, 1, 1, 1], \tag{17}$$

$$\mathbf{G}_{\mathrm{s}} = [150, 150, 150, 150, 150, 150, 150, 24, 24, 24], \tag{18}$$

$$\mathbf{W} = [\mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FFF}, \mathsf{FLL}, \mathsf{FLL}, \mathsf{FLL}]. \tag{19}$$

This shows that there are three designing sequences HPPH, HPHH, and HHHH corresponding to the entries equal to unity in (17), and from the corresponding entries in (19) we read off that, in this example, all three sequences possess the same unique ground-state conformation FLL with energy $E = -1$, as stored in (16).

Of course, parallelization must seem quite artificial in this very simple example. It would correspond to distributing the two contact maps $C^{(0)}$ and $C^{(1)}$ over two different IPs.

4. Applications

Table 1 and the corresponding Fig. 7 show how the number of self-avoiding walks, C_N, grows in comparison to the number of contact maps, M_N, for chain lengths $N \leqslant 19$. For a given chain length, there are many more self-avoiding walks than contact maps. The ratio between both numbers shown in the rightmost column of Table 1 keeps growing with $n = N - 1$. The exponential growth, as suggested by Fig. 7, can generically be described by the following scaling ansatz [20,21]:

$$C_n = A\mu^n n^{\gamma-1}, \tag{20}$$

where γ is a universal exponent and μ the effective coordination number. For self-avoiding walks (SAW) we reproduce the well-known results $\mu_{\mathrm{SAW}} \approx 4.684$ and $\gamma \approx 1.16$ [16–18] by means of a ratio method analysis (see, e.g., Refs. [20,22,23]). Assuming a scaling form (20) also for the number of contact maps (CM), a similar analysis yields $\mu_{\mathrm{CM}} \approx 4.38$, i.e. their (still) exponential growth is slower than that of self-avoiding walks (see Ref. [12] for more details).

In the second enumeration step we determined all designing sequences of length $N \leqslant 19$, their numbers are shown in Table 2. As the interaction is more complicated in the MHP case, it is intuitively clear that degeneracies are lifted and that there are hence more designing sequences for that model than in the HP case. We also note that there are fewer designing sequences in the HP model on the sc lattice than for the same model and the same lengths N on the square lattice [13].

5. Summary and outlook

In the first part of our exact enumeration procedure we generated the complete sets of contact maps for self-avoiding walks of $n \leqslant 18$ steps, i.e. for conformations of up to $N = 19$ monomers. We parallelized

Table 1
Number of self-avoiding conformations C_N and contact maps M_N on a sc lattice

N	$n = N - 1$	$\frac{1}{6}C_N$	M_N	$\frac{1}{6}C_N/M_N$
4	3	25	2	12.5
5	4	121	3	40.3
6	5	589	9	65.4
7	6	2 821	20	141.1
8	7	13 565	66	205.5
9	8	64 661	188	343.9
10	9	308 981	699	442.0
11	10	1 468 313	2 180	673.5
12	11	6 989 025	8 738	799.8
13	12	33 140 457	29 779	1 112.9
14	13	157 329 085	121 872	1 290.9
15	14	744 818 613	434 313	1 714.9
16	15	3 529 191 009	1 806 495	1 953.6
17	16	16 686 979 329	6 601 370	2 527.8
18	17	78 955 042 017	27 519 000	2 869.1
19	18	372 953 947 349	102 111 542	3 652.4

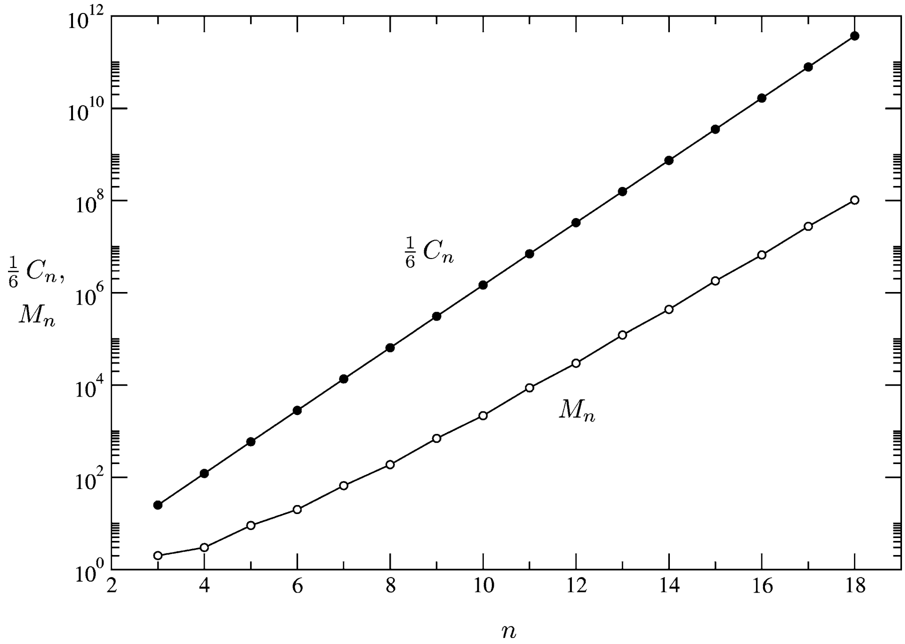


Fig. 7. Semi-log plot of the numbers C_n of self-avoiding walks and numbers M_n of contact maps vs. the walk length $n = N - 1$.

Table 2
Numbers of designing sequences S_N (only relevant sequences, see text) in the HP and MHP models

N	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
S_N^{HP}	3	0	0	0	2	0	0	0	2	0	1	1	1	8	29	47
S_N^{MHP}	7	0	0	6	13	0	11	8	124	14	66	97	486	2196	9491	4885

our program such as to distribute the set of contact maps over several memory partitions of a Linux cluster. In the second step of enumeration, we determined all designing sequences for both types of interactions considered. Here, parallelization is used to decrease the required computer time.

The results obtained this way can be used in a statistical analysis of designing sequences and their ground-state conformations. First, in the space of sequences, we can discuss the *hydrophobicity*, i.e. the H-content of designing sequences as well as *hydrophobicity profiles* describing the distribution of hydrophobic monomers in the polymer chain. Additionally, it enables us to investigate in how far monomers are involved in the formation of *HH* contacts (and *HP* contacts in case of the MHP model) by defining *hydrophobic contact density profiles*. Second, in the space of conformations, the data obtained herein allow for the study of the *end-to-end distances* and *radii of gyration* as measures of the compactness of designed conformations. The consideration of the distribution of the *designability* of designed conformations shows that some conformations are preferred over others as ground-state conformations of designing sequences. The complete statistical analysis can be found in Ref. [12].

Finally, it should be pointed out that a slight variation of the enumeration procedure explained herein also allows for the *exact* determination of the density of states $g(E)$, i.e. the number of conformations corresponding to all energy levels and not just to the ground-state energy. This number includes all symmetries which is why we store the degeneracies $g_{\mathrm{s}}(C)$ for all contact maps C (see Section 3.2.2). For a given HP sequence, $g(E)$ can be used to determine the temperature dependence of energetic quantities, in particular that of the specific heat whose peaks can be associated with conformational transitions [12].

Acknowledgements

This work was partially supported by the German–Israel-Foundation (GIF) under Grant No. I-653-181.14 /1999. One of us (R.S.) acknowledges support by the Studienstiftung des deutschen Volkes.

References

[1] T.E. Creighton, Proteins: Structures and Molecular Properties, second ed., W.H. Freeman, New York, 1993.

[2] K.A. Dill, Biochemistry 24 (1985) 1501;
K.F. Lau, K.A. Dill, Macromolecules 22 (1989) 3986.

[3] C. Tang, Physica A 288 (2000) 31.

[4] S. Miyazawa, R.L. Jernigan, J. Mol. Biol. 256 (1996) 623.

[5] A.D. Sokal, Monte Carlo methods for the self-avoiding walk, in: K. Binder (Ed.), Monte Carlo and Molecular Dynamics Simulations in Polymer Science, Oxford University Press, New York, 1995, p. 51.

[6] R. Unger, J. Moult, J. Mol. Biol. 231 (1993) 75.

[7] Y. Okamoto, Recent Res. Develop. Pure Appl. Chem. 2 (1998) 1.

[8] E. Bornberg-Bauer, Chain growth algorithms for HP-type lattice proteins, in: Proceedings of the First International Conference on Computational Molecular Biology, Santa Fe, ACM Press, New York, 1997, p. 47.

[9] P. Grassberger, Phys. Rev. E 56 (1997) 3682.

[10] A. Mitsutake, Y. Sugita, Y. Okamoto, Biopolymers (Peptide Science) 60 (2001) 96.

[11] M. Bachmann, W. Janke, Phys. Rev. Lett. 91 (2003) 208105;
M. Bachmann, W. Janke, J. Chem. Phys. 120 (2004) 6779.

[12] R. Schiemann, M. Bachmann, W. Janke, q-bio.BM/0405009, J. Chem. Phys., submitted for publication.

[13] A. Irbäck, C. Troein, J. Biol. Phys. 28 (2002) 1.

[14] H. Li, R. Helling, C. Tang, N. Wingreen, Science 273 (1996) 666.

[15] H. Cejtin, J. Edler, A. Gottlieb, R. Helling, H. Li, J. Philbin, N. Wingreen, C. Tang, J. Chem. Phys. 116 (2002) 352.

[16] D. MacDonald, D.L. Hunter, K. Kelly, N. Jan, J. Phys. A: Math. Gen. 25 (1992) 1429.

[17] D. MacDonald, S. Joseph, D.L. Hunter, L.L. Moseley, N. Jan, A.J. Guttmann, J. Phys. A: Math. Gen. 33 (2000) 5973.

[18] M. Chen, K.Y. Lin, J. Phys. A: Math. Gen. 35 (2002) 1501.

[19] P.S. Pacheco, Parallel Programming with MPI, first printed ed., Morgan Kaufmann, San Francisco, CA, 1997.

[20] D.S. Gaunt, A.J. Guttmann, Asymptotic analysis of coefficients, in: C. Domb, M.S. Green (Eds.), Phase Transitions and Critical Phenomena, vol. 3, Academic Press, London, 1974, p. 181.

[21] J.L. Guttmann, A.J. Guttmann, J. Phys. A: Math. Gen. 26 (1993) 2485.

[22] H.E. Stanley, Introduction to Phase Transitions and Critical Phenomena, Oxford University Press, New York, 1987.

[23] A.J. Guttmann, Asymptotic analysis of power-series expansions, in: C. Domb, J.L. Lebowitz (Eds.), Phase Transitions and Critical Phenomena, vol. 13, Academic Press, London, 1989, p. 3.