CPSP-tools - Exact and Complete Algorithms for High-throughput 3D Lattice Protein Studies

Martin Mann, Sebastian Will and Rolf Backofen*

Bioinformatics Group, University of Freiburg, Georges-Köhler-Allee 106, 79110 Freiburg, Germany

Email: CPSP-tools - cpsp@informatik.uni-freiburg.de; R. Backofen*- backofen@informatik.uni-freiburg.de;

*Corresponding author

Abstract

Background: The principles of protein folding and evolution pose problems of very high inherent complexity. Often these problems are tackled using simplified protein models, e.g. lattice proteins. The CPSP-tools package provides programs to solve exactly and completely the problems typical of studies using 3D lattice protein models. Among the tasks addressed are the prediction of (all) globally optimal and/or suboptimal structures as well as sequence design and neutral network exploration.

Results: In contrast to stochastic approaches, which are not capable of answering many fundamental questions, our methods are based on fast, non-heuristic techniques. The resulting tools are designed for high-throughput studies of 3D-lattice proteins utilising the Hydrophobic-Polar (HP) model. The source bundle is freely available at `http://www.bioinf.uni-freiburg.de/sw/cpsp/`

Conclusions: The CPSP-tools package is the first set of exact and complete methods for extensive, high-throughput studies of non-restricted 3D-lattice protein models. In particular, our package deals with cubic and face centered cubic (FCC) lattices.

Background

The organisation of bio-molecules, in particular proteins, in the sequence and structure space has recently been attracting increased attention. Particularly questions concerning finding the native structure or investigating the kinetics and evolution of proteins have been widely studied. These problems are often tackled using simplified models such as the Hydrophobic-Polar (HP) model (e.g. Jacob *et al.* [1]). Though abstract, these models are computationally feasible and do allow for deeper insights into fundamental and general principles [1–3].

Several recurring tasks can be identified in such studies using simplified models. Namely, predicting the native structure, classifying whether a sequence is protein-like, calculating its degeneracy and stability, or the design of sequences that optimally fold to a given structure. The problems associated with these tasks are computationally very hard (NP-complete) [4–6]. Nevertheless, these tasks demand for exact and complete (i.e. non-heuristic) methods. It is important to note that stochastic methods cannot be used for proving optimality and in particular proving that a sequence has a unique lowest energy

(protein-like) fold [7].

Consequently, with the exception of Yue and Dill [8], all studies requiring complete and exact answers to optimal structure prediction were based on exhaustive enumeration. These studies were, hence, confined to small sequence lengths. In other approaches, structures are artificially restricted to be maximally compact (e.g. filling a 3x3x3 cube) [9]. This allows for complete enumeration but artificially biases the energy function towards overall hydrophobicity.

Furthermore, many studies are confined to extremely simplified models on the 2D-square or 3D-diamond-lattice [1, 10]. The coordination number, a measurement of lattice complexity, is four in both cases. The use of lattices with such a low complexity may lead to oversimplified models that are not able to reproduce real world properties. Park and Levitt [11] have shown that lattices with higher coordination number provide a much better fit to real protein structures. A further hint toward the simplicity of the 2D-lattice is the low computational complexity of inverse folding when compared to the 3D-cubic lattice [6].

The *Constraint-based Protein Structure Prediction (CPSP)* approach by Backofen and Will [12] provides a way to overcome the aforementioned obstacles. The method is tailored to the HP model introduced by Lau and Dill [13]. This model is widely used in the literature [14, 15]. CPSP supports complex 3D lattices (currently cubic and face centered cubic) without artificial restrictions (e.g. to be maximally compact). The approach predicts all globally optimal structures together with a proof of optimality. No naive, exhaustive enumeration of all structures is performed and it is as fast as stochastic methods that cannot prove optimality. Backofen and Will [12] showed that the CPSP-approach could fold even sequences of length 200 to optimality within seconds. In contrast, exhaustive structure enumeration as e.g. done by Blackburne and Hirst [16] is restricted to short sequence lengths. For instance, on a 3D-cubic lattice it is only viable to enumerate up to about length 20. In fact, the exact number of structures is only known up to length 23 where there are already more than 5×10^{15} [17]. CPSP uses constraint programming that is commonly applied to hard (NP-complete) problems and, thus, avoids the complete expansion of the whole search space. Hence, constraint-programming techniques are a powerful tool to handle the high complexity that typifies problems related to protein structure. Constraint-programming techniques have successfully been applied to structure prediction with given secondary structure information [18], analysis of NMR data [19], and modeling of protein complexes [20].

Currently, we are not aware of any other complete approach that ensures optimality of the predicted structures in different lattices. There is an alternative to CPSP for the 3D-cubic lattice, the constraint-based hydrophobic core construction method by Yue and Dill [8]. This allows the prediction of optimal structures and proves their optimality. However, using the CPSP-approach, Backofen and Will showed that the method developed by Yue and Dill is not always complete in enumerating all optimal structures [12].

Complex Lattices. As mentioned before, complete structure enumeration is only applicable to simple, low coordination number lattices. In contrast, the CPSP-approach is built for the more complex 3D-cubic and 3D-face-centered-cubic (FCC) lattices with higher coordination numbers of 6 and 12, respectively.

A main feature of the CPSP-tools is their applicability to the unrestricted FCC lattice. The FCC lattice lacks one of the main problems of the 3D-cubic lattice, namely that only sequence positions with different parities form contacts; the parity problem [21]. Modeling protein structures on a FCC lattice, Park and Levitt [11] demonstrated that a good approximation of real protein structures is possible. They achieved a coordinate root mean square deviation of 1.78 Å, whereas a deviation of 2.84 Å was obtained in the 3D-cubic lattice. Recently, Bagci *et al.* [22] have shown that the neighborhood of amino acids in proteins closely resembles a distorted FCC lattice, and that the FCC is best suited for modeling proteins. The CPSP-approach is the first exact method that allows the prediction of provable optimal structures in the FCC lattice. An example is given in Figure 1.

Implementation

CPSP-tools provides a set of programs that enable typical, modern research tasks to be calculated efficiently and accurately. Here we list the programs each with a typical example application. HPSTRUCT predicts (all) optimal and suboptimal structures

as required for investigating properties of low energy conformations, as e.g. studied by Jacob and Unger [15]. The statistical analysis of protein-like sequences, see Blackburne and Hirst [10], requires a degeneracy-based classification of sequences. This is possible with HPDEG. For the exploration of protein evolution, similar to Wroe and Chan [23], one needs to investigate the sequence-structure space. We provide HPDESIGN for sequence design and HPNNET for neutral network computation.

All methods can be applied to HP-sequences in the cubic and the more complex face centered cubic lattice model. Before giving a detailed description of the tools, we first introduce the idea of H-cores, central to these methods.

H-core database

In the HP lattice models, two monomers form a *contact* if they occupy neighboring positions in the lattice. The *energy* of a structure is defined by the number of contacts between H-monomers, i.e. *HH-contacts*. Thus, an optimal (minimum energy) conformation maximizes the number of HH-contacts. An important observation is that optimal structures show an almost optimal (maximally compact) packing of the H-monomers. Such dispersions of H-monomers without any chain connectivity are called *H-cores*. The compactness of the H-cores is a basic feature that can be used for structure prediction and sequence design. Note that optimal H-cores are independent of a particular sequence and depend only on the number of H-monomers. Hence, compact and nearly compact H-cores can be precalculated and stored in a database. HPSTRUCT and HPDESIGN use this database as a starting point for their calculations (details later). Thereby, redundant computation is avoided, which significantly speeds up the CPSP-approach and related applications.

The enumeration of all optimal H-cores in complex lattice models such as FCC is a computationally hard problem by itself and was solved by Backofen and Will using constraint-programming techniques [24]. Firstly an upper bound on the number of possible contacts for a given number of monomers is calculated via dynamic programming. Subsequently, this information is used to enumerate all compact optimal and almost optimal (*suboptimal*) H-cores for a given number of H-monomers using constraint-programming.

Some statistics on the number of H-cores in the 3D-cubic lattice are given in Fig. 4. It shows that the number of H-cores grows exponentially in H-core size but still much slower than the number of structures for a corresponding sequence length.

HPstruct

Motivation. HPSTRUCT implements the CPSP approach, as introduced by Backofen and Will [12], to predict provably optimal structures of 3D lattice proteins in the HP-model. For a given HP-sequence S and a given lattice type (cubic or face centered cubic), (all) optimal structures are calculated. The CPSP approach computes the global minimal energy for S.

Methods. The CPSP-approach is based on the H-core database as described before. For a concrete sequence S the approach systematically examines the list of H-cores compatible with S in decreasing maximal contact number. For each core, it attempts to thread the sequence through the core. Threading means to find a placement of the monomers of S in a self-avoiding walk such that all H-monomers are elements of the given H-core and all P-monomers are outside of the core. Since the H-cores are considered in the order of decreasing contacts, the first successful threading results in a structure with global minimal energy. Note that at this point the algorithm has *proven* that there is no structure of S that forms more HH-contacts.

Technically, the threading of a sequence through a core is performed by a constraint program. For this purpose, we formulate the threading problem as a constraint satisfaction problem (CSP) [25]. It constrains the H-monomers of the sequence to the positions in the H-core. Further, it enforces successive monomers along the sequence to be neighbored in the lattice and prohibits the multiple use of a single position. The constraint-programming machinery allows for the enumeration of all valid placements according to the given constraints. In this way, all (sub)optimal structures for a given sequence can be calculated. For a more detailed description of the CSP definition and the mechanisms for solving it see [12].

Advanced Features. All resulting structures of HPSTRUCT are returned in absolute move string representation. This compactly encodes the lattice position vectors between successive monomers in the structure and reduces the space consumption for

huge data sets.

To handle the common case of highly degenerated sequences (with many optima), HPstruct offers the possibility to limit the number of predicted structures or to generate only a representing subset. Such a subset only contains structures that are separated by at least (a user defined) distance k. The distance measure is the hamming distance on the absolute move strings.

HPdeg

Motivation. The degeneracy of an HP-sequence S is the number of optimal structures S can adopt. It can be calculated using HPdeg and is the base to determine the stability of structures [26]. HPdeg specializes HPstruct and completely counts all optimal structures.

An important application of HPdeg is the classification of sequences as protein-like or not. A sequence is protein-like if it can adopt only one optimal structure (degeneracy 1), a definition applied by Li *et al.* [9] and Huard *et al.* [3] among others.

Methods. HPdeg is directly based on the CPSP-approach to compute the degeneracy. Here, all solutions for all arbitrary H-cores/CSPs are calculated. In addition, a significant acceleration of the process can be achieved by the search decomposition methods we introduced in [27]. This is done by identifying sub-chains of the sequence that can be placed independently from each other. Their placements are calculated separately and the resulting numbers are multiplied to the overall structure number of the whole chain. This decomposition strategy results in a speedup of 3-times and higher on average.

HPdesign

Motivation. HPdesign solves the inverse folding problem, i.e. the design of sequences that form a given structure X as their unique optimum. It allows deeper investigations of sequence-structure relations and a better understanding of general properties of protein folding [28].

The inverse folding problem (IFP) in 3D lattices has been shown by Berman *et al.* [6] to be NP-complete, i.e. it is, as the protein folding problem, a hard computational problem. In contrast, as the same authors show, the IFP in the simple 2D lattice is solvable in polynomial time. This indicates once more the higher complexity of three-dimensional lattice models. To our knowledge, HPdesign is the only method applicable to a 3D-model that calculates the desired sequence properties without exhaustive sequence space enumeration.

Methods. The approach is based on the CPSP H-core database in order to get a set of good candidate sequences C. First, using H-cores ordered by decreasing size and optimality, a matching of the core and the structure is done. For each match a candidate sequence is derived and added to C. Afterwards, each $c \in C$ is evaluated concerning degeneracy and checked if X is its optimal structure.

The candidate set C, produced by the filtering step using the H-cores, consists of sequences that can adopt X with an optimal or slightly sub-optimal H-core. Therefore, their probability to form X as their unique optimum is very high and the size of C very small compared to the whole sequence space. The latter is of high importance for the performance of the method.

Advanced Features. Often sequences with a special ratio of H/P occurrences or with only limited degeneracy are of interest. Both can be specified using HPdesign.

Furthermore, the number of evaluated H-cores is selectable to allow a balancing between runtime and completeness. This is done by adjusting their allowed level of optimality used in the filtering step.

HPnnet

Motivation. The organisation of sequence space in neutral networks provides insights into evolutionary principles [14, 29]. Such networks can be expanded using HPnnet. A neutral network for a given structure X is an undirected binary graph, where each node represents a sequence that forms X as its unique optimal structure. Edges connect evolutionary related sequences, i.e. sequences that differ only in one sequence position, a point mutation. HPnnet expands a neutral network starting from an initial sequence (or a set of sequences) S that folds into the structure X.

Methods. The method follows the generate-and-test paradigm. Recursively, all neighboring sequences of S are tested if they adopt X as their unique optimum. If so, they are added to the network and their neighbors are checked. Therefore,

HPNNET is capable of detecting and expanding connected neutral networks of different structures.

Advanced Features. Running HPNNET with S as the only start sequence results in the connected component of the network S belongs to. However, Blackburne and Hirst [16] have shown by exhaustive enumeration in restricted models that neutral networks may consist of several connected components. To find and study them in complex three-dimensional lattices a combination of HPDESIGN and HPNNET can be used. The independently designed sequences resulting from HPDESIGN have a high chance to belong to different components. HPNNET supports as input such a set of sequences and expands all corresponding connected components. An example is later shown in the results section.

Utility tools

In addition to those described above, CPSP-tools provides a set of utility programs helpful for lattice protein studies. For instance using HPCONVERT, it is possible to convert between absolute move strings, the 3D-position data in XYZ-, Protein Data Bank (PDB-) and Chemical Markup Language (CML-) format. A move string normalization, as well as a conversion into an orientation independent relative move string, is available for a symmetry independent structure comparison.

HPVIEW interactively visualizes structures in 2D-square, 3D-cubic, and 3D-FCC lattices using the Jmol interface (http://jmol.sourceforge.net/).

Installation and Usage

The package supplies standard installation procedures for Linux based on common tools (GNU automake) and can be compiled and installed easily on current 32- and 64-bit Linux systems (including Cygwin for Microsoft Windows^{TM}). The programs are written in C++ for highest performance and provide a slim text-based user interface for efficient pipelining as required for high-throughput experiments. A web front end is under development.

All constraint programming based algorithms utilize the open source Gecode system [30].

The validity of the algorithms has been tested and confirmed on a large set of benchmark problems. The functionality of H-core database access, structure prediction, and degeneracy computation are collected in the C++ CPSP-library. A complete API is included which allows the embedding, extension, and use of the CPSP approach in new programs.

To reduce package size, only a small fraction of the H-core database is included in the source package. This already enables the use of CPSP-tools for short sequences. The complete database is available on request.

Results and Discussion

For illustration, we provide some scenarios that exemplify the use of CPSP-tools in extending known or enabling new studies. All examples are performed in the unrestricted 3D-cubic lattice with HP-sequences of length 27. Note that for this length there are already more than 10^{19} possible structures, which makes an exhaustive enumeration inapplicable. Table 1 outlines the performance of programs from CPSP-tools. Table 2 shows the sequences used for Table 1, their optimal energy (E), and degeneracy (deg). All tasks were performed on an Intel P4 3GHz (using CPSP-2.0.0).

(1) Studies of sequence or structure features of proteins as done by Huard *et al.* [3] require a classification of sequences as protein-like. One way is to classify them by the number of optimal structures, i.e. their degeneracy. The fast calculation of this sequence property by HPDEG allows production of sufficiently large benchmark sets for detailed studies. To illustrate this, we run HPDEG for a random HP-sequence S_0 revealing an enormous degeneracy, which is a frequent finding in the HP-model. As a starting point for the following scenarios, we evaluate the degeneracy of S_1, a sequence with a single optimal structure. The very short runtimes for both checks are given in Table 1.

(2) Calculating the globally optimal structure for a given sequence is the main task in many studies, e.g. see Jacob and Unger [15]. Furthermore, in stochastic folding simulation approaches knowing the minimal possible energy is favorable. Both can be calculated extremely rapidly using HPSTRUCT. Again, We demonstrate this with sequences S_0 and S_1. This results in an energy of -13 and -22 and the optimal structures X_0 and X_1, respectively. Both structures are visualized in Figure 2.

(3) To study protein evolution on the sequence level, neutral networks are widely utilized [16]. Using HPNNET we can span the connected component

of the neutral network for a given sequence with a unique optimal structure. Applied to S_1 with X_1 we find four sequences $S_2 .. S_4$ sharing X_1 as their unique optimal structure. Note, this can be done *without* exhaustive sequence enumeration for a given structure.

(4) The detailed study of neutral networks by Blackburne and Hirst [16] has shown that neutral networks may decompose into connected components. Their results are based on full enumeration of sequences and structures in the diamond lattice. This approach does not extend to complex lattice models due to the enormous size of the structure space as discussed above.

HPDESIGN can overcome that problem by directly designing sequences of the neutral network. Recall that the neutral network contains only sequences with the same unique optimal structure. The described design approach allows one to generate sequences of independent components in the neutral network without exhaustive enumeration. Afterwards, the full components can be expanded via HPNNET.

We apply this approach to the neutral network of the structure X_1. HPDESIGN calculates 12 members of the network ($S_1 .. S_{12}$), including the four sequences $S_1 .. S_4$ known from scenario (3). Expanding the network N from these sequences via HPNNET reveals two further sequences S_{13}, S_{14} and two independent connected components as shown in Figure 3.

Preliminary studies performed with CPSP-tools indicate that neutral networks as large as N with several large independent components are rare in the unrestricted 3D-cubic model.

Conclusions

For complex 3D models, mainly heuristic and/or stochastic approaches to search for optimal structures of a given sequence are available [7, 31]. However, these methods are (a) incomplete and (b) cannot ensure the global optimality of the predicted structures. In consequence, the investigation of problems requiring this information was only possible using exhaustive enumeration, which is not possible for longer sequence lengths.

The CPSP approach is as fast as common stochastic methods *but ensures* that all predicted structures are globally optimal, and that none are missing. This is done without exhaustive structure space exploration applying constraint-programming techniques. Therefore, it is well suited to many studies in complex 3D models; especially for finding protein-like sequences, the investigation of neutral networks or sequence design. Further applications range from the generation of candidate sets to the validation of results of folding simulations and stochastic optimization methods.

The CPSP-tools package combines several applications in the field of bioinformatics concerning 3D lattice proteins. It allows advanced investigation of problems related to protein structure prediction, sequence evolution, inverse folding, and energy landscapes.

Availability and requirements

Project name: CPSP-tools

Project home page:
http://www.bioinf.uni-freiburg.de/sw/cpsp/

Operating system(s): all Linux based systems (including Cygwin for MS Windows^{TM})

Programming language: C++

Other requirements: Gecode and BIU library (a source bundle is provided)

License: BSD-style license

Any restrictions to use by non-academics: none

Authors contributions

Implementation and software design was done by MM and SW. The CPSP method was developed by RB and SW and extended by SW and MM. The CPSP derived algorithms are designed by all authors. All authors have approved and contributed to the final manuscript.

Acknowledgements

Martin Mann is supported by the EU project EMBIO (EC contract number 012835). Sebastian Will is partially supported by the EU Network of Excellence REWERSE (project reference number 506779). Further, thanks to the reviewers of an earlier version of the manuscript for their helpful comments and Rhodri Saunders for proofreading.

References

1. Jacob E, Horovitz A, Unger R: **Different mechanistic requirements for prokaryotic and eukaryotic chaperonins: a lattice study**. *Bioinformatics* 2007, **23**:240–248.
2. Wolfinger MT, Will S, Hofacker IL, Backofen R, Stadler PF: **Exploring the lower part of discrete polymer model energy landscapes**. *Europhysics Lett* 2006, **74**:725–732.
3. Huard FP, Deane CM, Woo GR: **Modelling sequential protein folding under kinetic control**. *Bioinformatics* 2006, **22**:202–210.
4. Unger R, Moult J: **Finding the lowest free energy conformation of a protein is an NP-hard problem: proof and implications**. *Bull Math Biol* 1993, **55**:1183–1198.
5. Berger B, Leighton T: **Protein folding in the hydrophobic-hydrophilic (HP) model is NP-complete**. *J Comp Biol* 1998, **5**:27–40.
6. Berman P, DasGupta B, Mubayi D, Sloan R, Turán G, Zhang Y: **The protein sequence design problem in canonical model on 2D and 3D lattices**. In *Combinatorial Pattern Matching*, *Volume 3109*, Springer 2004:244–253.
7. Hsu HP, Mehra V, Nadler W, Grassberger P: **Growth-based optimization algorithm for lattice heteropolymers**. *Phys Rev E* 2003, **68**:021113.
8. Yue K, Dill KA: **Forces of tertiary structural organization in globular proteins**. *Proc Natl Acad Sci* 1995, **92**:146–150.
9. Li H, Tang C, Wingreen NS: **Designability of protein structures: a lattice-model study using the Miyazawa-Jernigan matrix**. *Proteins* 2002, **49**:403–412.
10. Blackburne BP, Hirst JD: **Population dynamics simulations of functional model proteins**. *J Chem Phys* 2005, **123**:154907–9.
11. Park BH, Levitt M: **The complexity and accuracy of discrete state models of protein structure**. *J Mol Biol* 1995, **249**:493–507.
12. Backofen R, Will S: **A constraint-based approach to fast and exact structure prediction in three-dimensional protein models**. *Constraints* 2006, **11**:5–30.
13. Lau KF, Dill KA: **A lattice statistical mechanics model of the conformational and sequence spaces of proteins**. *Macromolecules* 1989, **22**:3986–3997.
14. Cui Y, Wong WH, Bornberg-Bauer E, Chan HS: **Recombinatoric exploration of novel folded structures: a heteropolymer-based model of protein evolutionary landscapes**. *Proc Natl Acad Sci* 2002, **99**:809–814.
15. Jacob E, Unger R: **A tale of two tails: why are terminal residues of proteins exposed?** *Bioinformatics* 2007, **23**:225–230.
16. Blackburne BP, Hirst JD: **Three-dimensional functional model proteins: structure function and evolution**. *J Chem Phys* 2003, **119**:3453–3460.
17. Sloane NJA: **Number of n-step self-avoiding walks on cubic lattice**. In *On-Line Encyclopedia of Integer Sequences* 2007. [`www.research.att.com/~njas/sequences/A001412`].
18. Dal Palu A, Dovier A, Fogolari F: **Constraint Logic Programming approach to protein structure prediction**. *BMC Bioinformatics* 2004, **5**:186.
19. Krippahl L, Barahona P: **PSICO: Solving protein structures with constraint programming and optimization**. *Constraints* 2002, **7**:317–331.
20. Krippahl L, Moura JJ, Palma PN: **Modeling protein complexes with BiGGER**. *Proteins* 2003, **52**:19–23.
21. Decatur SE: **Protein folding in the generalized hydrophobic-polar model on the triangular lattice** 1996. [Technical Memo MIT-LCS-TM-559, Massachusetts Institute of Technology].
22. Bagci Z, Jernigan RL, Bahar I: **Residue coordination in proteins conforms to the closest packing of spheres**. *Polymer* 2002, **43**:451–459.
23. Wroe R, Chan HS, Bornberg-Bauer E: **A structural model of latent evolutionary potentials underlying neutral networks in proteins**. *HFSP J* 2007, **1**:79–87.
24. Backofen R, Will S: **Optimally Compact Finite Sphere Packings — Hydrophobic Cores in the FCC**. In *Proc of the 12th Annual Symposium on Combinatorial Pattern Matching*, *Volume 2089*, Springer 2001:257–272.
25. Marriott K, Stuckey PJ: *Programming with Constraints: an Introduction*. The MIT Press 1998.
26. Shortle D, Chan HS, Dill KA: **Modeling the effects of mutations on the denatured states of proteins.** *Prot Sci* 1992, **1**:201–215.
27. Will S, Mann M: **Counting protein structures by DFS with dynamic decomposition**. In *Proc of Workshop on Constraint Based Methods for Bioinformatics* 2006:83–90.
28. Gupta A, Manuch J, Stacho L: **Structure-approximating inverse protein folding problem in the 2D HP model**. *J Comp Biol* 2005, **12**:1328–1345.
29. Schuster P, Stadler PF: **Networks in molecular evolution**. *Complexity* 2002, **8**:34–42.
30. **Gecode - generic constraint development environment** 2007. [`http://www.gecode.org`].
31. Shmygelska A, Hoos HH: **An ant colony optimisation algorithm for the 2D and 3D hydrophobic polar protein folding problem**. *BMC Bioinformatics* 2005, **6**:30.

Figures

Figure 1 - Structure in FCC lattice model

One optimal structure of sequence S_1 from Table 2 with 50 HH-contacts in the 3D-face centered cubic (FCC) lattice model. The coloring shows H-monomers in green and P-monomers in grey.

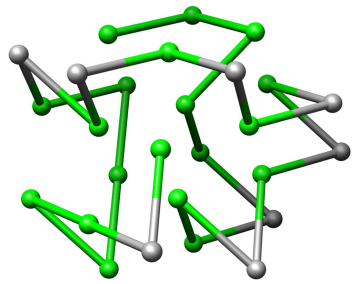


Figure 2 - Structures in 3D-cubic lattice

An optimal structure X_0 for sequence S_0 and the unique optimal structure X_1 of S_1 from Table 2 in the 3D-cubic lattice. The coloring shows H-monomers in green and P-monomers in grey.

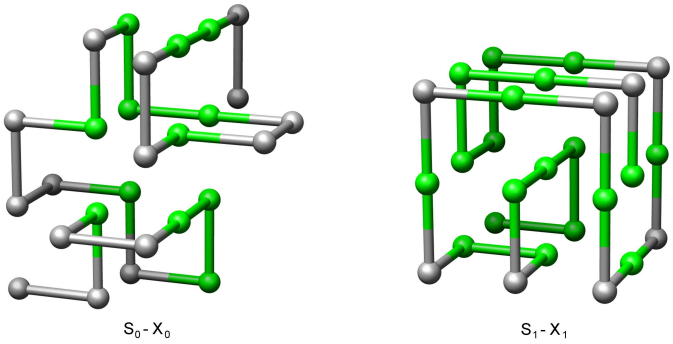


Figure 3 - Neutral net

Known independent components of the neutral network for structure X_1 from Table 2 in the 3D-cubic lattice. The border size corresponds to the node degree. The structure is visualized in Figure 2.

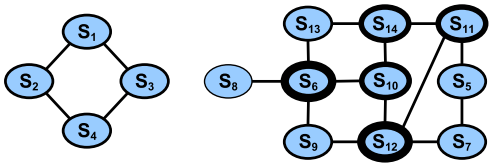
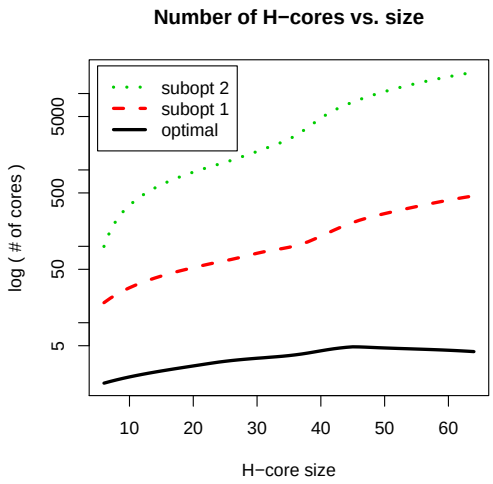


Figure 4 - H-core database statistics

The number of different H-cores for several number of H-monomers (H-core size) in the 3D-cubic lattice. The three curves represent different levels of optimality of the H-cores.



Tables

Table 1 - Exemplary runs and data

Example runs of the exemplified CPSP-tools application scenarios. The corresponding sequences and structures are given in Table 2. The neutral net N is given in Figure 3.

appl.	tool	parameter	result	runtime
1	HPDEG	S_0	471354	2.5s
1	HPDEG	S_1	1	0.2s
2	HPSTRUCT	S_0	$X_0, E = -13$	0.01s
2	HPSTRUCT	S_1	$X_1, E = -22$	0.06s
3	HPNNET	$X_1, S_1, deg = 1$	$S_1 .. S_4$	9s
4	HPDESIGN	$X_1, minH = 17, so = 2$	$S_1 .. S_{12}$	13m43s
4	HPNNET	$X_1, S_1 .. S_{12}, deg = 1$	$N, S_1 .. S_{14}$	1m

Table 2 - Data of exemplary runs

The corresponding sequences and structures for the exemplary runs of CPSP-tools in the 3D-cubic lattice. For each sequence its optimal energy (E) and degeneracy (deg) is listed. The optimal structures of the sequences are given in absolute move string representation (**F**orward, **B**ackward, **L**eft, **R**ight, **U**p and **D**own). The corresponding neutral net of sequences $S_1 .. S_{14}$ is given in Figure 3.

id	sequence	E	deg
S_0	PPHPPHHHPHPPPHPHHHPPHPPHHPP	-13	471354
S_1	HHHHHPHHPHPHPHPHPHPHHHHHHPH	-22	1
S_2	HHHHHPHHPHHHPHPHPHPHHHHHHPH	-23	1
S_3	HHHHHPHHPHPHPHPHHHPHHHHHHPH	-23	1
S_4	HHHHHPHHPHHHPHPHHHPHHHHHHPH	-24	1
S_5	HHHHHPHHPHPHHHPHPHHHPHHPHHH	-23	1
S_6	HHHHHPHHPHPHPHPHHHPHPHHPHPH	-22	1
S_7	HHHHHPHHPHPHHHPHHHHHPHHPHHH	-24	1
S_8	HHHHHPHHPPPHPHPHHHPHPHHPHPH	-20	1
S_9	HHHHHPHHPHPHHHPHHHPHPHHPHPH	-22	1
S_{10}	HHHHHPHHPHPHPHPHHHPHPHHPHHH	-22	1
S_{11}	HHHHHPHHPHPHHHPHPHPHPHHPHHH	-22	1
S_{12}	HHHHHPHHPHPHHHPHHHPHPHHPHHH	-23	1
S_{13}	HHHHHPHHPHPHPHPHPHPHPHHPHPH	-21	1
S_{14}	HHHHHPHHPHPHPHPHPHPHPHHPHHH	-21	1
X_0	FLUFDDRBLBULFLDRFFUBULDDDR	S_0	
X_1	FLUURDBULLFFRRDDLLBBRULFFR	$S_1 .. S_{14}$	