Fast Tree Search for A Triangular Lattice Model of Protein Folding

Xiaomei Li* and Nengchao Wang

Computer Science and Technology Institute, Huazhong University of Science and Technology, Wuhan 430074, China.

Using a triangular lattice model to study the designability of protein folding, we overcame the parity problem of previous cubic lattice model and enumerated all the sequences and compact structures on a simple two-dimensional triangular lattice model of size 4+5+6+5+4. We used two types of amino acids, hydrophobic and polar, to make up the sequences, and achieved $2^{23}+2^{12}$ different sequences excluding the reverse symmetry sequences. The total string number of distinct compact structures was 219,093, excluding reflection symmetry in the self-avoiding path of length 24 triangular lattice model. Based on this model, we applied a fast search algorithm by constructing a cluster tree. The algorithm decreased the computation by computing the objective energy of non-leaf nodes. The parallel experiments proved that the fast tree search algorithm yielded an exponential speed-up in the model of size 4+5+6+5+4. Designability analysis was performed to understand the search result.

Key words: triangular lattice model, protein folding, fast search tree, designability

Introduction

The prediction of a protein structure from its primary sequence is one of the most interesting problems in computational biology (*1*). Native proteins usually fold much too fast (by at least tens of magnitude) to involve an exhaustive search (*2*). It is a classical puzzle of the protein folding that biological proteins could not have originated from random sequences. Despite a tremendous amount of efforts and progresses over many decades, the problem remains essentially unsolved.

In 1963, Anfinsen and his colleagues made a remarkable discovery that the amino acid sequence of a protein was fully sufficient to specify the molecule's ultimate 3D shape and biological activity (*3*). For most single domain proteins, the information coded in the amino acid sequence is sufficient to determine the three-dimensional folded structure, which is the minimum free energy structure. Based on this theory, the protein is described by the complete list of the atoms in a molecule, with connectivities, bond lengths, angles, and force constants between all pairs of atoms (*4*). This all-atom model involves complex energy force and needs astronomical computational time. To understand the folding mechanism, it is useful to study simplified models such as the Hydrophobic-Hydrophilic model introduced by Dill (*5*, *6*). Lattice protein folding models have been playing important roles in theoretical studies of protein folding (*7*). In these models, a protein is represented by a self-avoiding chain of beads placed on a discrete lattice, with two types of beads used to mimic hydrophobic and polar (HP). The advantage of HP lattice models is that they are simple enough to be amenable to thorough theoretical study, which can provide fruitful insights to feed back to or test against realistic models and experiments.

The biological foundation of this model is the believed theory that the first-order driving force of protein folding (*8*, *9*) is due to a "hydrophobic collapse" in which those residues that prefer to be shielded from water (hydrophobic residue) are driven to the core of the protein, while those that interact more favorably with water (polar residues) remain on the outside of the protein. Previous papers researched the problem of protein folding on the cubic lattice model whose goal was to find the fold with the maximum number of contacts between non-covalently linked hydrophobic amino acids. Yet, a significant drawback of the cubic lattice is the "parity problem". In this paper, we present a triangular lattice model that overcomes the shortcomings of the cubic lattice model. Based

*** Corresponding author.**
E-mail: lxmdwj@163.com

on this model, we enumerated all possible compact structures and HP sequences.

For a 3×3×3 cubic lattice model of size N=27, the total number of compact structures is 103,346 and the number of all possible HP sequences is 2^{27}. In the enumeration study, if the energy of every sequence folded into every compact structure is valued, the total number of evaluation will be $2^{27}\times 103{,}346 \approx 1.39\times 10^{13}$. For the triangular models of size 3+4+5+4+3 and size 4+5+6+5+4, the computation is $2^{19}\times 20{,}486 \approx 1.07\times 10^{10}$ and $2^{24}\times 1{,}474{,}782 \approx 2.48\times 10^{13}$. It will be strongly desirable to enumerate larger size system if possible. So it is crucial to find a fast search algorithm. In this paper, we present a fast search algorithm by constructing a fast search tree. The algorithm decreased the computation by computing the objective energy of tree non-leaf nodes. The parallel experiments proved that the fast tree search algorithm yielded an exponential speed-up factor of $\Theta\left(1.486^{\log_2 M-\log_2 Max_{op}}\right)$, in which M is the number of different compact construct strings, Max_{op} is the optimal string bound per leaf.

Model

The methods currently used for the tertiary structure modeling are based on cubic lattice models to enumerate the minimization of the energy as a function of the topological contacts. As previously stated, the quadrate lattice model exists a defect referred as the "parity problem", in which only the residue in the even position of the primary sequence and another one in the odd site can form the topological contact. The non-bonded neighbor can't be made between two even residues or two odd residues. This situation can never be found in the triangular lattice models.

We generated and enumerated all the compact self-avoiding walks on the triangular lattice model of size 4+5+6+5+4 (Figure 1). There were 5,903,128 different paths in this model. We may use symmetries to reduce the conformational space significantly. There are two types of symmetries for walks on this model. The first is the two-rotation symmetry. The second is the two-mirror symmetry, of which one is the axial reflection and the other is the diagonal reflection. So we got the differently directed 5,903,128/4=1,475,782 self-avoiding walks with this model. Among the 1,475,782 paths, we obtained reversal symmetries of 738,189 pairs and head-tail symmetries of 596 pairs (Figure 1), so the number of different compact structures in this model was 738,189+596=738,785.

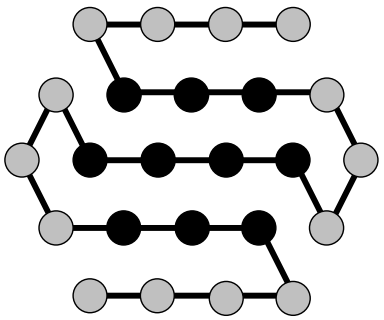

Fig. 1 The 4+5+6+5+4 lattice model with core sites (black) and surface sites (gray).

It is energetically favorable for hydrophobic amino acids to occupy core sites, where there is low exposure to water. In this model, we denoted a sequence of amino acids by σ_i, and took only two types of amino acids, hydrophobic and polar. The energy of a sequence folded into a structure was taken to be the sum of the contributions from each amino acid upon burial away from water:

$$E = -\sum_{i=1}^{N} \sigma_i s_i,$$

where s_i is a structure-dependent number characterizing the degree of burial of the i^{th} amino acid in the chain. Larger s_i corresponds to a smaller surface area accessible to the solvent. For a structure on a 3D lattice, there are four different kinds of sites: center, face, edge, and corner. Therefore, in principle, there could be four different values of s_i. On a 2D lattice model, we took only two values for s_i, and defined a string s_i for each structure with $s_i = 1$ if the i^{th} site is a core and $s_i = 0$ if it is a surface (Figure 1).

Out of the 1,475,782 compact structures, the number of distinct structure strings was 219,093, among which there were 25,825 lattice conformations, and each represented exactly one structure. The 219,093 distinct structures would decrease to 109,497 excluding reversal symmetries.

Fast Search Tree

Each structure string has exactly ten 1's and fourteen 0's. Our goal was to find a target structure string $\{s_j\}$, which had unique and minimum energy corresponding to the target sequence string $\{\sigma_i\}$. It is obvious that the target structure string must possess a

certain similarity with the target sequence string. According to the observation, we organized the structure string into a binary tree and clustered similar structure strings into the same tree node (Figure 2). The algorithm decreased the computation by computing the objective energy of non-leaf nodes to locate the target structure string. The distinct structure strings was 219,093 in the model of 4+5+6+5+4. There were thus 109,656 distinct strings that we kept in the calculation excluding the reversal structure strings. Each node of the tree represented a subset of these strings and maintained the following three kinds of information. First, a structure string would have the value 1 at the i^{th} position if and only if all the strings corresponding to this node have 1's at the i^{th} position. We named this kind of sites the known ones (K). Second, a structure string would have the value 1 at the i^{th} position if and only if there is a table entry in this node that has a 1 or 0 at the i^{th} position. We named this kind of sites the unknown ones (U). Third, each string in a node would have 1's at some undecided positions. For each string, missing ones are the sum of these 1's. By construction, each string has exactly ten 1's, so the number of missing ones is equal to 10 minus the sum of known ones. That is, missing ones are single integers no greater than 10 for each node. We named this kind of sites the missing ones (M).

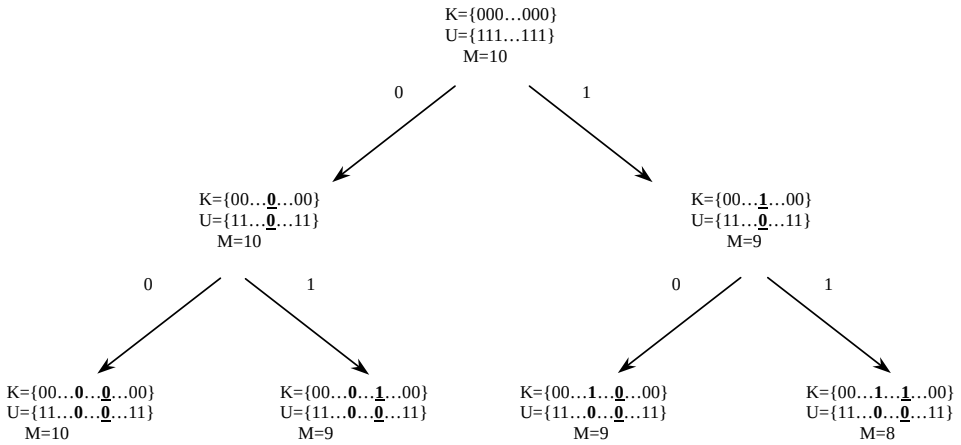

Fig. 2 A cluster tree. Underlined bold face denotes UK-renewing information; bold face denotes UK-renewed information; ellipsis of K denotes "0"; ellipsis of U denotes "1".

We splited each node at the position that made two child nodes as tightly clustered as possible, and measured this clustering for each child as the entropies for each site, with the tightest clustering corresponding to the minimum entropy. Specifically, for each node we regarded the site of minimum entropy S of its set of structure strings as a branch point (*10*):

$$S = Min\left\{-(p_i \log p_i + q_i \log q_i)\right\},$$

in which p_i is the probability of the i^{th} position being 1, and q_i is the probability of the i^{th} position being 0.

Those nodes partition the strings in the parent according to the value of the given position i: one child has the entire parent strings where i=1 and the other child has all of the strings where i=0. Each leaf node at the end of the tree contains a small list of structure strings—in the following experiments we defined the parameter as Max.

Given a sequence string $\{\sigma_i\}$ and a node of the tree, we hoped to obtain the bounds of all structure strings represented by the node. Clearly, for all strings in the node the upper bound can be expressed by:

$$B_{upper} = Min\{\sigma_i * (U + K), \sigma_i * K + m\}.$$

Given such a tree, where the root corresponds to all of the structure strings, the question here is how to search the tree. We computed the upper bound of the sequence string according to previous formula, and called this the objective value. If there are any leaf nodes that achieve this objective, go to next sequence. If not, repeat with the reversed version of the sequence strings. Again, if we achieve the objective value, go to next sequence. If not, decrease the objective value by 1 and try again. Repeat until the goal is satisfying.

Given an objective value, we searched the tree as follows, starting at the root node. If the upper bound on the node indicates that objective value is unachievable, backdate and search other nodes. If the node

is a leaf node, check each structure string. If one string or none has been found that satisfies the objective, backdate and search other nodes, else if two strings have been found that achieve this objective, exit and transfer the next sequence string. If the current search node is not a leaf node, try each of the child nodes. We first tried the child that matches $\{\sigma_i\}$. The following experiments proved that only this step would yield a speed-up factor 1.782 in the model of size 4+5+6+5+4.

Experiments

Using the fast search tree, we were able to completely enumerate all possible HP sequences and compact structures in the 4+5+6+5+4 triangular lattice model. For every sequence, we rapidly computed the energy value of all the compact structures, found the structure with the minimal energy value, and recorded the minimal energy value and energy value of the first excited state (the second minimal energy value).

The overall computation is highly parallelizable because each sequence can be done independently. In order to implement the calculation of ground states for all sequences in parallel, it is useful to divide the sequences into groups and use these groups as the unit of parallelism. We performed our computations with all $2^{23}+2^{12}$ HP sequences excluding reversed strings, which produced 129 groups. These 129 groups were executed on the Legend Group DeepComp1800 -P4 Xeon 2 GHz -Myrinet/ 512 Large Array Multiple Processors, which is a collection of 24 computers, each containing two Intel Pentium Pro microprocessors of 2 GHz. Every machine ran the Linux operating system and had at least 512 MB of memory.

The parallel algorithm is described as follows. We divided the space of sequences into 129 groups f_i ($0 \leq i < 129$). Parameter "num" denotes the sign of group with the initial value of 0; Parameter "count" denotes the number of performed group with the initial value of 0. The operation of main processor P_0 includes: (1) take out m groups f_0, f_1, ..., f_{m-1}, send the groups to processor P_1, P_2, ..., P_m, respectively; num=m; count=0; (2) perform the data of $f_{num}{}^{th}$ group, count added by 1 and num added by 1; (3) receive the result of processors P_i ($1 \leq i \leq m$); count added by 1; (4) if num<n−1 (n denotes the number of groups), send the data of $f_{num}{}^{th}$ group to processor P_i, else notify processor P_i to exit; num added by 1; (5) if num<n−1, transfer to step 2; if count equals to n, transfer to step 6; (6) collect the results of all the other processors P_i ($1 \leq i \leq m$) and exit. The operation of other processors P_i ($1 \leq i \leq m$) includes receiving the group sent by the main processor P_0, performing the fast tree search, and sending the result to the main processor P_0.

Each leaf node at the end of the tree contains a small list of structure strings. We considered the maximum structure string number (Max) per leaf as a variant in our experiment. The creating tree time (T_t), search time (T_c), and total time (Total=T_t +T_c) through the experiment are shown in Figure 3. The computation time showed in Figure 3 is the CPU time and its unit is second.

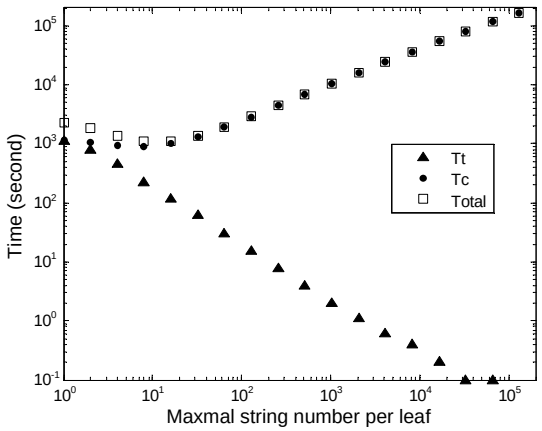

Fig. 3 Search time vs. Max.

It is obvious that search time and total time are the lowest when Max is 8. The creating tree time decreases exponentially as Max increases (Figure 3). The reason is that when creating the tree, every node needs to search all the structure strings and the number of nodes in the tree increases exponentially as Max decreases, so the creating tree time increases exponentially.

At the same time, the total time increases exponentially as Max increases. The reason is that the leaf node of tree increases exponentially as Max decreases. Futile search is eliminated by computing the objective value and aim structure string is located rapidly; hence the search time decreases exponentially. However, when Max is less than 8, search time becomes longer. This is because when Max is small, the leaf node contains less structure strings, but information in its parent node is enough to describe that (Figure 4), so continuous bisection will increase the computation of objective energy value, which makes search time increase. Surely, the optimal Max will differ with different models. When Max equals to the total num-

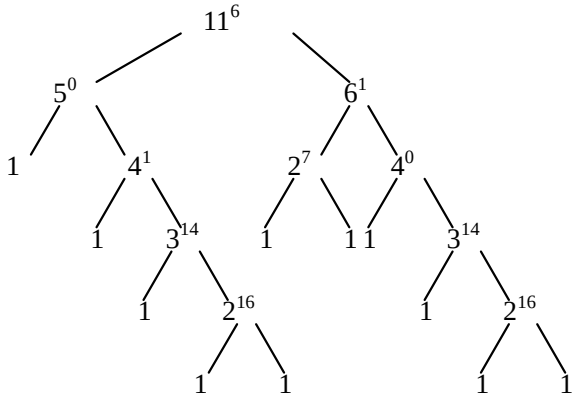

Fig. 4 The Fractional cluster tree (Max=1). The date denotes structure string number of node and superscript denotes the branch site.

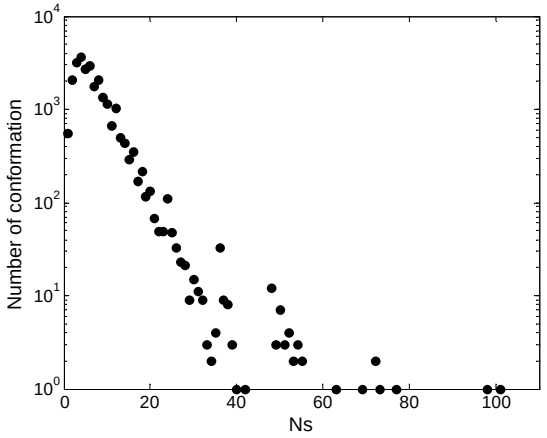

Fig. 5 Number of structures vs. Ns.

ber of structure strings, there is only a node in the tree and the situation equals to enumerate all the structure strings.

The experiments proved that the fast tree search algorithm yielded an exponential speed-up factor of $\Theta\left(1.486^{\log_2 M - \log_2 Max_{op}}\right)$ in the 4+5+6+5+4 triangular lattice model, in which M is the total number of distinct structure strings, and Max_{op} is optimal Max.

Statistical Analysis

Whenever a ground state structure string is found for a sequence, the reversed sequence necessarily has the reversed structure string as a ground state. We excluded the reversed structure string and reversed sequence string in order to simplify and statistically analyze the process. There were totally 109,656 such structure strings unrelated by rotational, reflection, or reverse labeling symmetries. For a given sequence, the ground state structure is found by calculating the energy of all compact structures. We completely enumerated all the ground states of all $2^{23}+2^{12}$ possible sequences, and found that only 181,375 sequences have unique ground states, and then we calculated the designability of each compact structure. There are structures that can be designed by an enormous number of sequences, and there are poor structures that can only be designed by a few sequences. The top structure can be designed by 101 different sequences. The number of structures decreases monotonically as the number of sequences increases (Figure 5). The result offers the evidence that the highly designable structures can tolerate more mutations during evolution because these structures can be designed by more different sequences.

Under different designability situations, we obtained that the ratio of sequences with the ground state energy –10, –9, –8, and –7 among 181,375 non-degeneracy sequences was 62.64%, 33.50%, 3.83%, and 0.32%, respectively. The statistical relationship between the sequences with ground state energy –10, –9, –8 and the designability is shown in Figures 6 and 7. From the figure we can find that the sequences with the ground state energy –10 account for 95% in the structure of the highest designability, while those with the ground state energy –9 and –8 account for 5% and 0, respectively; the sequences with the ground state energy –10 account for 52% in the structure of the lowest designability, while those with the ground state energy –9 and –8 account for 37% and 11%, respectively. The data shows that highly designable structures tend to have lower energy of ground. This indicates that the compact structures of highly designability are, on average, thermodynamically more stable than other structures.

Considering a structure string to be a chain of 0's and 1's linked by N−1 links of three types, 0-0, 1-0 or 0-1, and 1-1, with N_{00}, N_{10} or N_{01}, and N_{11} being the numbers of such links, respectively, we analyzed the relationship between designability and N_{01} (Figure 8). When N_{01}=10, the structures of high designability occur more frequently (Figure 8A), and the relatively high frequency of low designability structures are due to the large number of structure strings with designability "1" (Ns=1). The tendency is approximately the same as N_{01}=12 (Figure 8A), except lower frequency. When N_{01}= 8, 9, 11 (Figure 8B), the structures of high designability occur less frequently while the structures of low designability occur more frequently. The tendencies of N_{01}= 6, 7, 13 (Figure 8C) are approximately the same as each other and

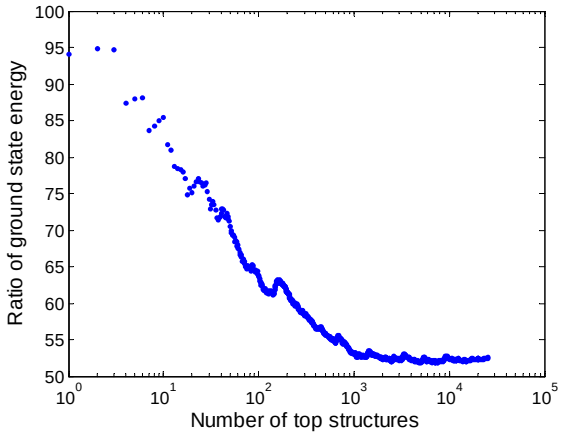

Fig. 6 Ground state energy –10 vs. Ns.

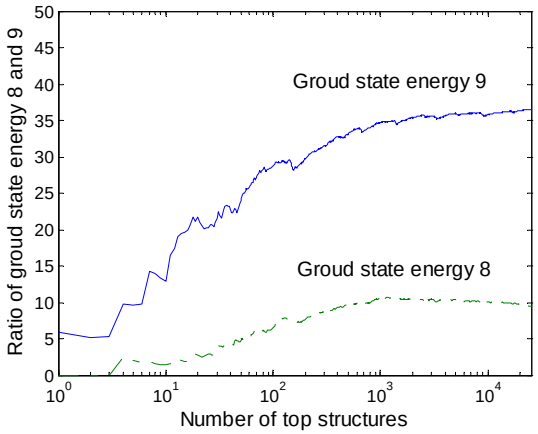

Fig. 7 Ground state energy –9 and –8 vs. Ns.

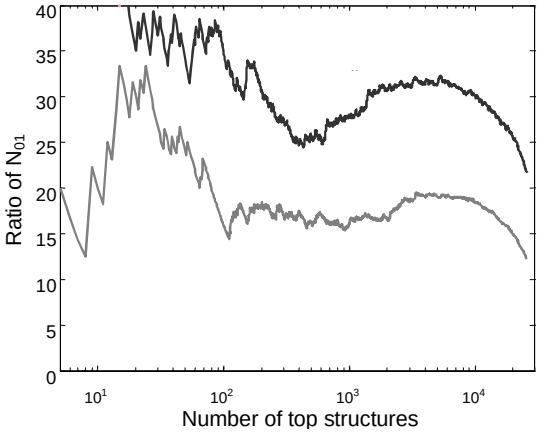

Fig. 8A N_{01}=10 (black line), 12 (gray line) vs. Ns.

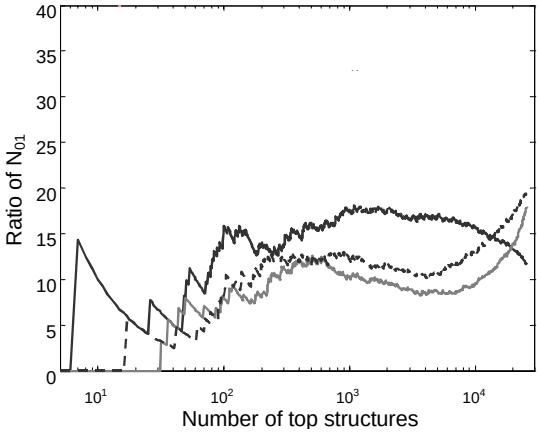

Fig. 8B N_{01}=8 (black solid line), 9 (black dash line), 11 (gray line) vs. Ns.

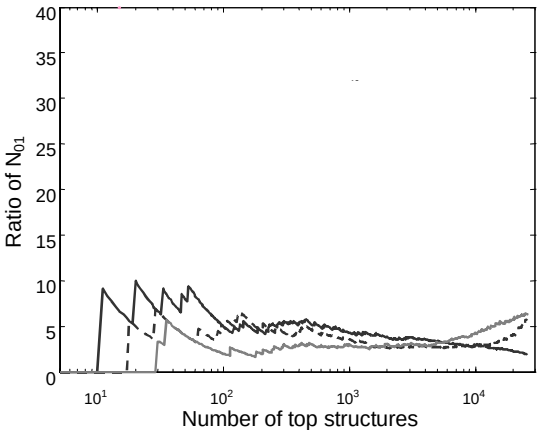

Fig. 8C N_{01}=6 (black solid line), 7 (black dash line), 13 (gray line) vs. Ns.

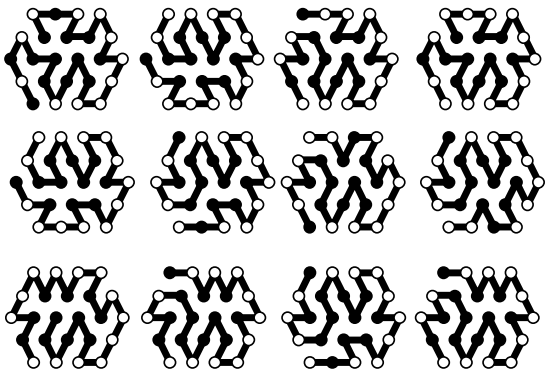

Fig. 9 Top 12 compact structures in the 4+5+6+5+4 triangular lattice model.

have no much effect on the designability.

The occurrence of such kind of phenomenon is because that the structures of high designability have regular secondary structures. In the top twelve structures of high designability (Figure 9), the number of N_{01} should be in a suit scope to form a regular second structure; it is appropriate to form a regular helix structure when N_{01}=10. This is similar to the natural protein structure. There is no regular sheet in the structures of high designability because of the energy

computation simplification of the lattice model.

We suppose that each dot in the 24D hypercube represents the sequence and structure in the lattice model. In the HP model, the energy of a sequence folded into a particular structure is the Hamming distance between their binary strings. Hence, the number of sequences that fold uniquely to a particular structure—the designability of the structure—is the set of vertices lying closer to that structure than to any others.

It happens that in the hypercube the smallest Hamming distance between two structures is approximately proportional to the difference in their respective N_{10} numbers. This is evident in Figure 10, where the smallest Hamming distance is plotted against the difference in N_{01} for all the pairs among the 25,825 binary structures on a 4+5+6+5+4 lattice, and is consistent with results given by Li *et al* (*11*) in which x(p) (the degree of clustering of hydrophobic residues) is analogous to N_{01}.

To see whether what we have observed so far has anything to do with real proteins, we compared five sequences, P_{1-5}, each being a concatenation of a set of real proteins or (4+5+6+5+4) lattice binary peptides. P_1, the representative non-redundant 350 proteins culled from Protein Data Bank (PDB; www.rcsb.org/pdb/); P_2, the sections in P_1 that fold into helices; P_3, the sections in P1 that fold into sheets; P_4, the 2,919 peptides mapped to the highest designabilities; P_5, the 2,077 peptides mapped to the lowest designabilities.

Firstly we defined the frequency distribution function as:

$$F_i^{(l)}(m) = \frac{f_i^{(l)}(m) - \overline{f}_i^{(l)}}{Z},$$

in which $f_i^{(l)}(m)$ denotes the frequency of the m^{th} binary word of length l occurring in sequence P_i and there are 2^l words of length l.

$\overline{f}_i^{(l)}$ denotes the average frequency of $f_i^{(l)}(m)$:

$$\overline{f}_i^{(l)} = \frac{\sum\limits_m f_i^{(l)}(m)}{2^l}.$$

$Z = (\sum\limits_m (f_i^{(l)}(m) - \overline{f}_i^{(l)})^2)^{1/2}$ denotes the normalized frequency distribution function.

The pairwise overlaps were defined as:

$$O_{ij}^{(l)} = \sum_{m=1}^{2^l} F_i^{(l)}(m) F_j^{(l)}(m),$$

where $i = 2, 3; j = 4, 5; l = 4 \sim 14$.

The relationship between pairwise overlaps of different sequences is shown in Figure 11. It is seen that P_4 (P_5) is positively (negatively) correlated with P_3. For all values of l, the strongest correlation occurs between the model sequence of high designability (P_4) and the real protein sequence rich in sheets (P_3). The sequence of low designability is strongly correlated with the sequence rich in helix (P_2), and the strong correlation occurs between the two model sequences of high (P_4) and low (P_5) designabilities. The strong correlation between P_2 and P_5 is to some extent an artifact of the lattice model. Since we only considered the simple structure string, fractional compact structures will be washed out due to path degeneracy.

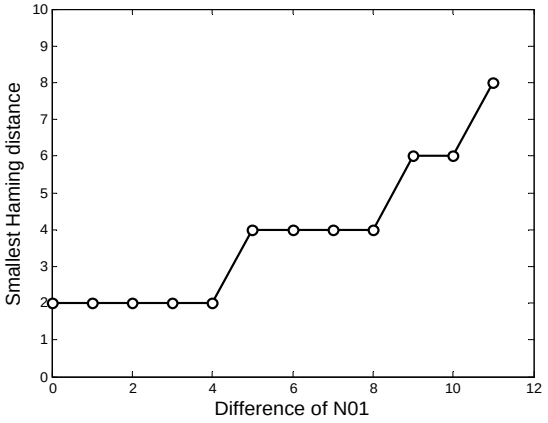

Fig. 10 Smallest Hamming distance vs. ΔN_{01}.

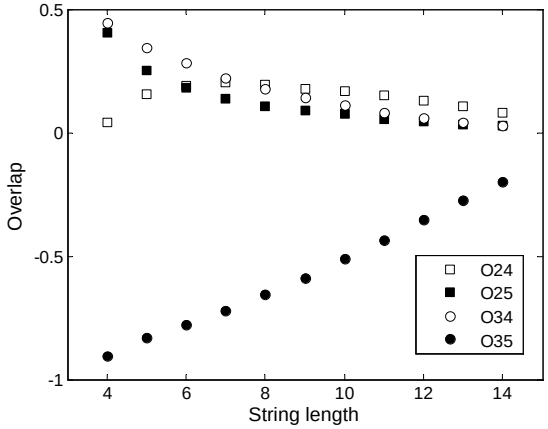

Fig. 11 Overlap vs. String length.

Conclusion

In the opinion of algorithm, we created the cluster tree by using the relationship between structure strings. The algorithm decreased the computation by computing the objective energy of non-leaf nodes. The parallel experiments proved that the fast tree search algorithm yielded an exponential speed-up in models of size 4+5+6+5+4. In this paper, we have presented the two-dimensional triangular lattice model to study the designability of protein folding. Through enumerating all the possible compact structures and HP sequences, we found that different compact structures have rather different designability. The compact structures of high designability exhibit lower ground state energy, showing that these structures are, on average, thermodynamically more stable than other ones. The research offers strong evidence that compact structures of high designability are more regular and geometrically symmetric. In the opinion of structure, the triangular lattice model has no parity problem and its surface is more similar to the natural proteins, therefore it is possible to get more information from the research of highly designable compact structures on triangular lattice models.

References

1. Anfinsen, C.B. 1973. Principles that govern the folding of protein chains. *Science* 181: 223-230.
2. Richards, F.M. 1991. The protein folding problem. *Sci. Am.* 264: 54-57, 60-63.
3. Levinthal, C. 1968. Are there pathways for protein folding? *J. Chem. Phys.* 65: 44-45.
4. Dill, K.A. 1999. Polymer principles and protein folding. *Protein Sci.* 8: 1166-1180.
5. Dill, K.A. 1985. Theory for the folding and stability of globular proteins. *Biochemistry* 24: 1501-1509.
6. Dill, K.A, *et al.* 1995. Principles of protein folding: a perspective from simple exact models. *Protein Sci.* 4: 561-602.
7. Chandru, V., *et al.* 2003. The algorithmics of folding proteins on lattices. *Disc. App. Math.* 127: 145-161.
8. Chan, H.S. and Dill, K.A. 1989. Intrachain loops in polymers: effect of excluded volume. *J. Chem. Phys.* 90: 492-509.
9. Li, H., *et al.* 1997. Nature of driving force for protein folding: a result from analyzing the statistical potential. *Phys. Rev. Lett.* 79: 765-768.
10. Cejtin, H., *et al.* 2002. Fast tree search for enumeration of a lattice model of protein folding. *J. Chem. Phys.* 116: 352-359.
11. Li, H., *et al.* 1998. Are protein folds atypical? *Proc. Natl. Acad. Sci. USA* 95: 4987-4990.