**Improved growth algorithms for protein folding in lattice model**

Li XiaoMei, Faculty of computer, Guangdong University of technology, Guangzhou, 510090

Abstract—PERM is the most efficient approach for protein folding problem in lattice model. In this paper we presented an improved growth algorithm which simplify the calculation of weight when choosing different branches and apply different upper thresholds according to different type of monomers. The experiment results show that the improved growth algorithm can find the known lowest ground state faster for the given HP sequences in square lattice model.

I. INTRODUCTION

In the 1963, Christian R. Anfinsen and his colleagues at the National Institutes of health made a remarkable discovery. It seemed that the amino acid sequence of a protein was fully sufficient to specify the molecule’s ultimate three-dimensional shape and biological activities[1]. Despite a great deal of effort and improvement over past 40 years, the protein folding problem[2] remains essentially unsolved. Two classic puzzles of protein folding are the Blind Watchmaker’s Paradox that biological proteins could not have originated from random sequences and Levinthal’s Paradox that the folded state of a protein cannot be found by random search[3]. Both paradoxes are traditionally framed in terms of random unguided searches through vast spaces, and vastness is equated with impossibility. But both processes are partly guided[4].

Now we can research the protein folding properties through the simple lattice model[5,6]. There so-called “HP lattice models” can capture some essential features of the protein folding problem. The biological foundation of this model is the believed that the first-order driving force of protein folding[7,8] is due to a “hydrophobic collapse” in which those residues which prefer to be shielded from water (hydrophobic residue) are driven to the core of the protein[9], while those which interact more favorably with water (polar residues) remain on the outside of the protein. The simple cubic lattice model have been applied to a lot of problems including protein folding pathway[10], designability of lattice model [11-14], folding in lattice models with side chains[15,16] and triangular model research[17].

The previous paper research the problem of protein folding on the cubic lattice model in which the goal[18,19] is to find the fold with the maximum number of contacts between non-covalently linked hydrophobic amino acids. In the model, each amino acid is represented as a bead, each bond is straight line, bond angles are a few discrete options rather than a continuum, different conformations conform to lattices in two or three dimension, and the 20 amino acids are condensed into a two-letter alphabet: hydrophobic (H) or polar (P).

For 2D square lattice model of size $N=22$, the total number of different conformation is 301,100,754 excluding the rotation symmetries and mirror reflection; It will be strong desirable to evaluate larger size system. In this paper, we use the growth algorithms together with other methods[21,22] to find the lowest ground state energy conformation. We studied the five 2D HP-sequences of length $N=60, 64, 85, 100$ and 100. The experiment results show that the improved growth algorithm can find the known lowest ground state faster for the given HP sequences in lattice model.

II. THE HP MODEL

In the HP model a protein, i.e. an amino acid sequence, is abstracted as a string describing the hydrophobicity of each amino acid in the sequence. The linear sequence is composed of amino acids of only two types: hydrophobic (H) and hydrophilic (P). Throughout this paper we use $N$ to denote length of an amino acid sequence. A folding of a protein in HP model is embedding of amino acid sequence in the lattice such that adjacent characters in sequence occupy adjacent grid points in the lattice, and no grid point in the lattice is occupied by more than one character.

In the lattice model, a structure is specified by the position $r_i$ ($i = 1, 2 \ldots N$) of each residue along the chain. The energy for a sequence folded into a structure is taken to be the sum of the topological contact energy, that is:

$$ H = \sum_{\sigma_i \sigma_j} E_{\sigma_i \sigma_j} \Delta(r_i - r_j) $$

Where $E_{\sigma_i \sigma_j}$ is the constant energy between residue types $\sigma_i$ and $\sigma_j$. There are three kind of the topological contact energy $E_{\text{HH}}$, $E_{\text{HP}}$, and $E_{\text{PP}}$. We choose these interaction parameters as $(-1, 0, 0)$ in this paper. $\Delta(r_i - r_j) = 1$ if $r_i$ and $r_j$ are adjoining lattice sites with $i$ and $j$ not adjacent along the chain, and $\Delta(r_i - r_j) = 0$ otherwise.

Therefore, the globally optimal conformations in this model are simply those with the maximum possible number of HH contact. Then the folding problem is transformed to an optimization problem maximizing the number of HH contact.
III. The Improved Grown Algorithm

The algorithm is built on the PERM to use a biased growth algorithm for polymer\(^{[23]}\), where the biased is corrected by means of giving a weight to each sample conformation. While the chain grows by adding monomers, this weight will fluctuate along with the number of the hydrophobic contacts. PERM suppresses these fluctuations by pruning conformations with too low weight, and by enriching the sample with copies of high-weight conformation. These copies are made while the chain is growing and continue to grow independently of each other.

Since conformations are invariant through rotations, the position of the first two amino acids can be fixed without loss of generality. In the weight initialization phase of improved grown algorithm(IGA), its direct successor in the sequence arbitrarily assign to neighboring position in the lattice model according to a uniform random. During the grown procession, the weight of partial conformation of length \(n\) is defined as \(W_n\) :

\[
W_1 = W_2 = 1; \quad W_n = W_{n-1}e^{\Delta E/T} (N \geq n \geq 2)
\]  

(1)

In which \(\Delta E\) denotes added energy when the \(n\)th amino acid is placed and \(T\) is a temperature parameter and its value is between 0.25 and 0.35. It is obvious that if the \(n\)th amino acid is polar the added energy equals to zero and the weight of partial conformation of length \(n\) equals to the weight of partial conformation of length \(n - 1\). The partition sums \(Z_n\) of length \(n\) conformation can be initialized as the weight of partial conformation of length \(n\), namely \(Z_n = W_n (N \leq n \leq 1)\)

The chain grown or weight update phase of improved grown algorithm can be described as:

(1) Calculate the average qualities \(q_n\) of all possible legal positions of placing the \(n\)th monomer:

\[
q_n = (\sum_{k=1}^{k_{free}} \Delta E_k) / k_{free}
\]  

(2)

In which \(k_{free}\) counts the number of all possible legal positions where the \(n\)th monomer can be placed and \(\Delta E_k\) denotes the added energy when the \(n\)th monomer is placed in \(k\)th possible legal position.

(2) Calculate two thresholds \(W_n^>\) and \(W_n^<\) which depend on partition sums \(Z_n\). \(Z_n\) means the average value of \(n\)-monomer chains weight \(W_n\) of length \(n\) conformation already created during run.

We used \(W_n^> = C^> Z_n\) and \(W_n^< = C^< W_n^>\), where \(C^>\) and \(C^<\) are constant factor. For the \(C^>\), we choose different value in different situation, such as we define it as 0.6–0.8 when \(n^{th}\) monomer was hydrophobic residue and 0.05–0.2 when \(n^{th}\) monomer was polar residue.

(3) When we have a partial conformation with \(n - 1\) monomers, we first estimate its expectation predicted weight \(W_n^{pred}\) for the next step.

\[
W_n^{pred} = W_{n-1}e^{\Delta E/T}
\]  

(3)

(4) Pruning and enrichment were done depending on two thresholds \(W_n^>\) and \(W_n^<\) and the current expectation predicted weight \(W_n^{pred}\) with partial conformation of \(n\) monomers.

If the current expectation predicted weight \(W_n^{pred}\) of \(n\)-monomer chain is less than \(W_n^<\), the chain is discarded with probability 1/2, or it is kept and its weight is doubled.

If the current expectation predicted weight \(W_n^{pred}\) of \(n\)-monomer chain is more than \(W_n^<\) and less than \(W_n^>\), the chain is grown with only one free site and the weight is unchanged.

If the current expectation predicted weight \(W_n^{pred}\) of \(n\)-monomer chain is more than \(W_n^>\), the chain is grown with \(k\) free sites and we used \(k = \min\{k_{free}, \text{ceiling}(\frac{W_n^{pred}}{W_n^>})\}\) and the weight multiple a factor \(\frac{k}{k_{free}}\).

(5) Implement chain grown and calculate the weight of partial conformation of length \(n\) according to formula (1) and update partition sums \(Z_n\).

IV. Implementation and Results for the Improved Grown Algorithm

We use one data set (see table 1) to test the performance of improved grown algorithm. These 2D HP sequences are typical set of well known difficult instances in nowadays literature.

At first R.Konig test the 60-mer HP sequence in [25]. It needed only 10 seconds to hit the lowest energy -36 in the square lattice model on the DEC 21264 using [23] method. We simulate the 60-mer HP sequence to hit the lowest energy -36 on the SANYO 1.7G in the square lattice model (see Figure 1a) in the several seconds using improved growth algorithm.

Using the previous algorithm it could find all known lowest energy conformation for all sequence tested in [25] except the 64-mer. This sequence is particularly difficult for any growth algorithm in the square lattice model. Using [23] method the average CPU time per hit the ground state energy –42 was 30h on the DEC 21264. To find the ground state energy for the 64-mer, it took 84 minutes/hit on the SANYO 1.7G in the square lattice model (see Figure 1b). The reason that exits the
long CPU time is perhaps that it has to grow a long arc which at first seems very unnatural and which is stabilized only much late.

Table 1 2D HP standard benchmark instances.

<table>
<thead>
<tr>
<th>No</th>
<th>Length</th>
<th>E</th>
<th>Protein sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>60</td>
<td>-36</td>
<td>P2HPH3PHH10PHH2P2H6PH2PH</td>
</tr>
<tr>
<td>2</td>
<td>64</td>
<td>-42</td>
<td>H1LP2H2P2H6PH2P2HP2HP</td>
</tr>
<tr>
<td>3</td>
<td>85</td>
<td>-53</td>
<td>H4P4H18P6(H12P3)3HP2HP2HP</td>
</tr>
<tr>
<td>4</td>
<td>100</td>
<td>-50</td>
<td>P3H2P2HP3PH2P2HP4P8HP2HP</td>
</tr>
<tr>
<td>5</td>
<td>100</td>
<td>-48</td>
<td>P6HPH2P5PH3PH2P2H4P2HP2HP3P6HPH2</td>
</tr>
</tbody>
</table>

Fig 1 The lowest conformation with E=-53, -42, -50, -48 of N=85, 60, 64 and two 100 HP on improved growth algorithm in square lattice and triangular lattice.

The 85-mer 2D HP sequence was presented in [24], where it was claimed to have minimal energy -52. Using the growth algorithm[23] it took about 10 minutes CPU time to find energy -53 between successive hits on the Sun ULTRA 1 and using the improved growth algorithm it only took 20 seconds to get energy -53 between successive hits on the SANYO 1.7G in the square lattice model (see Figure 1c).

Lastly, we studied the two 2D HP sequences of length 100 presented in [26]. For the two sequences, using [23] algorithm the lowest energy conformation with -48 and -50 could be found within the 5.8 hours and 2.6 minutes between successive hits on a 667MHz DEC21264. Using the improved growth algorithm we found the ground state energy conformation -48 and -50 within 5 minutes and 8 minutes in the square lattice models (see Figure 1 d and e).

V. CONCLUSION

We present an improvement of growth algorithm for the 2D HP model. Through numerical analysis, we show that the improved growth algorithm hit the lowest energy conformation faster than previous methods excluding the one 2D HP sequence. All of these results show that the improved growth algorithm is a more efficient algorithm with further improvement for protein folding problem.

REFERENCES