Algorithm for protein folding problem in 3D lattice HP model

METODI TRAYKOV
Department of Electrical Engineering, Electronics and Automatics
University Center for Advanced Bioinformatics Research
South-West University "Neofit Rilski"
66 Ivan Mihaylov Str., 2700 Blagoevgrad
BULGARIA
metodi.gt@gmail.com

NICOLA YANEV
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Acad. Georgi Bonchev Str., Block 8, 1113 Sofia
BULGARIA
choby@math.bas.bg

RADOSLAV MAVREVSKI
Department of Electrical Engineering, Electronics and Automatics
University Center for Advanced Bioinformatics Research
South-West University "Neofit Rilski"
66 Ivan Mihaylov Str., 2700 Blagoevgrad
BULGARIA
radoslav_sm@abv.bg

BORISLAV YURUKOV
Department of Informatics
South-West University "Neofit Rilski"
66 Ivan Mihaylov Str., 2700 Blagoevgrad
BULGARIA
bobyur@swu.bg

Abstract: - The prediction of a protein's tertiary structure from the amino acid sequence of a protein is known as the protein folding problem. The protein folding problem in 3D lattice Hydrophobic-Polar model is problem of finding the lowest energy conformation. This is the NP-complete problem. In this article we propose extension of the heuristic algorithm described by some of authors to solve the protein folding problem in 3D cubic lattice in HP model. For computational experiments we use 8 HP sequences that are known in the literature benchmarks for 3D lattice in HP model. We compare the obtained results with results obtained by algorithms for solving the problem in 3D lattice HP model as genetic algorithms, ant-colony optimization algorithm, and Monte Carlo algorithm.

Key-Words: - Bioinformatics, Protein Folding Problem, 3D Cubic Lattice, HP Model, Heuristics, HP folding, 3D structure

1 Introduction

The 3D structure of proteins is the major factor that determines their biological activity. The synthesis of new proteins and the crystallographic analysis of their 3D structure is very slow and very expensive process. If we can predict the 3D structure of many proteins, than only proteins with expected properties have to be synthesized.

The mistakes, arising in the protein folding process lead to occurrence of proteins with unusual forms, which are the main causes of many diseases such as cystic fibrosis, Alzheimer's disease and mad cow. If we can predict, with high accuracy, the tertiary structures of proteins from their primary structure, we will be able to better treat these diseases. The knowledge of the tertiary structures of proteins, there are other applications, such as in drug design [1].

The common practice for predicting of the tertiary structure on the proteins is to use models that simplify the possible conformations search space. These models reflect the different global characteristics of the proteins structure. In the Hydrophobic-Polar (HP) model the primary amino acids sequence of the protein (which may be represented as a string over twenty-letters alphabet) is simplified to a sequence of hydrophobic (H) and polar (P) amino acids and thus presented as a sequence over {H,P} alphabet [2].

Hydrophobic-Polar (HP) model describes a protein sequence based on the fact that hydrophobic amino acids must have less contact with water as opposed to the polar amino acids [2]. The way of folding is determined by the polarity or the hydrophobicity of different amino-acids, so the 3D structure with minimum energy is the real case, i.e. the optimal conformation of protein folding in HP model is the one that has the maximum number of H-H contacts (Figure 1), which give the lowest energy value [3].

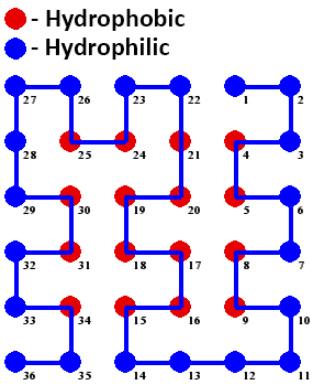

Fig. 1. Optimal conformation for HP sequence with length 36 amino acids in 2D lattice (14 contacts).

The prediction of the 3D structure of proteins, from the primary structure (the amino acid sequence), is known as Protein folding problem. It is proved that the Protein folding problem in HP model for 2D and 3D is *NP*-hard [4].

In 2D, the heuristic algorithm described by Traykov et al. [15] generated folds that are better than the folds obtained by approximate algorithms as Monte Carlo Algorithm, Newman's algorithm, Hart-Istrail algorithm, and close to the folds obtained by the Mixed Search Algorithm, and Genetic Algorithm [5, 6, 7, 8]. Here, we will present extension on this heuristic to solve the protein folding problem in 3D.

2 HP Folding in lattice method

The processes, related with the protein folding are very complex and only minority of them are explained and understood from the scientists. For this reason the simplified models such as Dill's HP model, have become one of the main tools for study of proteins [2]. HP model is based on the observation that the hydrophobic interaction between the amino acid is the driving force in the protein folding process. In the HP model, the energy of the conformation is defined as the number of contacts between hydrophobic amino acids, which are not neighbors in the protein sequence. More specifically conformation c with n H-H contacts have energy value $E(c) = n$.

In the HP lattice model, 20-th amino acids are reduced to two types – H (Hydrophobic) and P (Polar). In lattice model hydrophobic (H) interactions are the driving force in the protein folding process. Also, in the lattice model, each sequence is presented as **self-avoiding walk**. The self-avoiding walk is a sequence of moves in the lattice, which do not pass through the same position more than once.

The connections between the H-H amino acids are constructive [9]. The natural conformation of the HP sequence is defined as the conformation with the largest number of H-H contacts. Basing on the number of H-H contacts, we calculate the energy value of the conformation. The energy value should be minimized in order to obtain the best 3D structure. Figure 2 shows a schematic representation of the 3D HP lattice model.

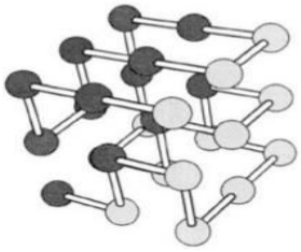

Fig. 2. 3D HP lattice model [9].

The protein folding problem in 3D HP lattice model can be defined as follows. Given an amino acid sequence, $S = s_1, s_2, ..., s_n$ (sequence of letters over the {H,P} alphabet) and a lattice. The goal is to find conformation of S with lowest energy value, i.e. Maximize:

The number of H-H contacts

Subject to:

1. (*Assignment*) Each amino acid must occupy one lattice point.
2. (*Non-overlapping*) No two amino acids may share the same lattice point.
3. (Connectivity) Each two amino acids that are consecutive in the protein's sequence must also occupy adjacent lattice points.

For solving the protein folding problem in 3D HP lattice model, are proposed a number of known heuristic optimization methods, including Evolutionary Algorithms (EA), Monte Carlo (MC) algorithms, Ant Colony Optimization (ACO) [10, 11, 12, 13].

3 Integer programming formulation

Let n to be the length of the protein sequence. Let $L(i, k)$ to be 3D lattice, with side N:

- $N = n$;
- $N = 2\sqrt{n}$;
- $N = \frac{n}{2}$.

So, the size of the lattice $L(i, k)$ is N^3.

We define HP model in 3D lattice. For simplification we convert 3D lattice model in 1D as follows [14]: we present three-dimensional coordinates (x, y, z) as one $i = N^2(z-1) + N(y-1) + x$.

Each cell in column $i \in L$ and row $k \in L$ on the lattice may be occupied by element of the protein sequence. We define the following variables

$$x_{i,k} = \begin{cases} 1, if\ the\ k^{th}\ amino\ acid\ is\ in\ the\ cell\ i \\ 0, otherwise, \end{cases}$$

where $i = 1 \dots N^3$, $k = 1 \dots n$.

$$y_{i,k,j,l} = \begin{cases} 1, if\ we\ have\ H-H\ contact \\ 0, otherwise, \end{cases}$$

where $i, j = 1 \dots N^3$, $k, l = 1 \dots n$.

Each element k can be placed in only one cell of the lattice (*Assignment*):

$$\sum_{i=1}^{N^3} x_{i,k} = 1, \quad \forall k, \qquad (1)$$

where $k = 1 \dots n$.

Each cell i can contain only one element of the input sequence (*Non-overlapping*):

$$\sum_{k=1}^{n} x_{i,k} \leq 1, \quad \forall i, \qquad (2)$$

where $i = 1 \dots N^3$, $k = 1 \dots n$.

Each two neighboring elements of the protein sequence should be placed in the adjacent cells in the lattice (*Connectivity*):

$$x_{i,k} \leq \sum_{j \in G(i)} x_{j,k+1}, \qquad (3)$$

where $i = 1 \dots N^3$, $k = 1 \dots n$. These constraints define **self-avoiding walk** (Figure 3).

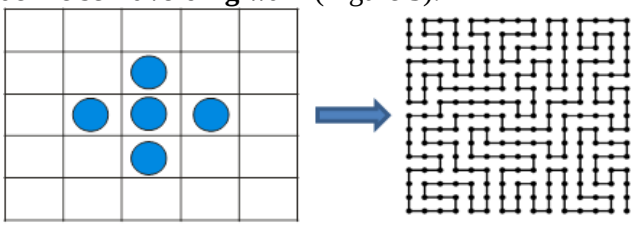

Fig. 3. Self-avoiding walk.

The additional variable $y_{i,k,j,l}$, which has value 1, if two adjacent cells are occupied by hydrophobic amino acids that are not adjacent in the protein sequence and 0, otherwise:

$$x_{i,k} \geq \sum_{j \in G(i)} y_{i,k,j,l}, \quad \forall i, k, \qquad (4)$$

$$x_{j,l} \geq \sum_{i \in G(j)} y_{i,k,j,l}, \quad \forall j, l, \qquad (5)$$

where

$i, j = 1 \dots N^3$,

$k, l = 1, \dots, n$,

$G = \{l - k > 2 \cap S_k = H \cap S_i = H\}$,

$G(j)$ – set of cells, which are neighbor of j-th cell

Our goal is to maximize the number of contacts between the hydrophobic amino acids, i.e.

$$\max \sum y_{i,k,j,l}.$$

4 Algorithm for solving the problem

The heuristic algorithm described in [15] uses sequence of move to generate self-avoiding walk in 2D. To solve the problem in 3D we extend this set of moves to generate self-avoiding walk in a cubic lattice, i.e. in 3D case the possible directions for the

movement of amino acids in the lattice are six: L (Left), R (Right), U (Up), D (Down), F (Forward) and B (Back). The main idea of algorithm is as follow.

We consider a sequence S with length n in a cubic lattice. The size of the cubic lattice is selected so that the first two amino acids of the sequence to be fixed in center of the lattice, or with 1 cell displacement from it. We divide the sequence S on parts with predefined size, i.e. $S = S_1 \cup S_2 \cup \ldots \cup S_m$, $S_i \cap S_{i+1} = \varnothing$. After that, we take i-th part from S and generate all possible folds. On the next step we choose the fold with maximum number of contacts and put it in a cubic lattice. To already obtained fold we add (i+1)-th part from S and find all possible folds against already selected fold. From the obtained new folds we choose the fold with maximum number of contacts and put it in a cubic lattice. This concept allows us to reach a solution for protein with any length.

5 Computational experiments

For computational experiments we use 8 HP sequences that are known in the literature benchmarks for 3D lattice in HP model (Table 1).

Length	Protein Sequence
20	$(HP)_2PH(HP)_2(PH)_2HP(PH)_2$
24	$H_2P_2(HP_2)_6H_2$
25	$P_2HP_2(H_2P_4)_3H_2$
36	$P(P_2H_2)_2P_5H_5(H_2P_2)_2P_2H(HP_2)_2$
46	$P_2H_3PH_3P_3HPH_2PH_2P_2HPH_4PHP_2H_5PHPH_2P_2H_2P$
48	$P_2H(P_2H_2)_2P_5H_{10}P_6(H_2P_2)_2HP_2H_5$
50	$H_2(PH)_3PH_4PH(P_3H)_2P_4(HP_3)_2HPH_4(PH)_3PH_2$
60	$P(PH_3)_2H_5P_3H_{10}PHP_3H_{12}P_4H_6PH_2PHP$

Table 1 HP benchmarks for 3D lattice.

The symbols H_i, P_i and $(\ldots)_i$ in table 1 shows i repeats of character or sequence.

In table 2 we compare the obtained results by Extended Heuristic Algorithm (the column EHA) with known in the literature results obtained by Meta-Heuristic Ant Colony Optimization Algorithm (the column ACO-Metaheuristic) [12, 16], Genetic Algorithm (the column GA) [17], and Evolutionary Algorithm with Backtracking (the column Backtracking-EA) [18]. The column BKS show best known solution for these HP sequences.

Length	Contacts				
	BKS	GA	Backtracking-EA	EHA	ACO-metaheuristic
20	11	11	11	11	10
24	13	13	13	13	8
25	9	9	9	9	6
36	18	18	18	18	10
46	32	–	–	29	21
48	29	25	25	31*	–
50	26	23	23	26	–
60	49	37	39	55*	–

Table 2 Computational results obtained for 8 HP sequences in 3D.

With * we note the protein sequence for which we improve the best know energy value

From table 2 we can see that the EHA generates the best know solution for sequences with length 20, 24, 25, 36 and 50 amino acids. For sequences with length 48 (Figure 4), and 60 amino acids (Figure 5) the algorithm generates folds that are greater than the best know solution for these protein sequences.

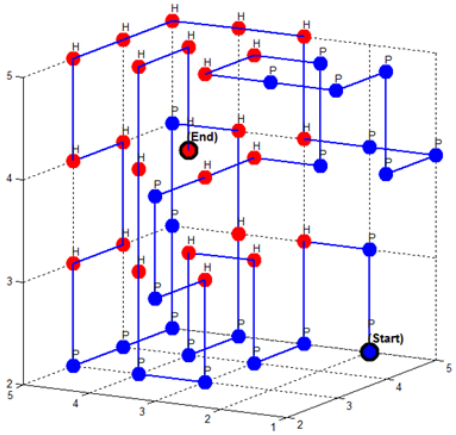

Fig. 4. Protein folds with length 48 amino acids (31 contacts).

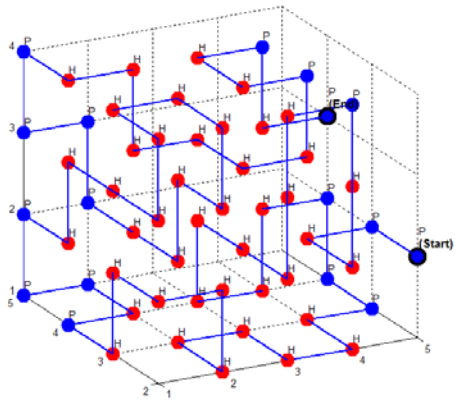

Fig. 5. Protein folds with length 60 amino acids (55 contacts).

A table 3 shows the execution time for each of the test sequences.

Length	HP Sequence	CPU time (sec.)
20	$(HP)_2PH(HP)_2(PH)_2HP(PH)_2$	44
24	$H_2P_2(HP_2)_6H_2$	410
25	$P_2HP_2(H_2P_4)_3H_2$	103
36	$P(P_2H_2)_2P_5H_5(H_2P_2)_2P_2H(HP_2)_2$	82
46	$P_2H_3PH_3P_3HPH_2PH_2P_2HPH_4P$ $HP_2H_5PHPH_2P_2H_2P$	216
48	$P_2H(P_2H_2)_2P_5H_{10}P_6(H_2P_2)_2HP_2$ H_5	93
50	$H_2(PH)_3PH_4PH(P_3H)_2P_4(HP_3)_2$ $HPH_4(PH)_3PH_2$	369
60	$P(PH_3)_2H_5P_3H_{10}PHP_3H_{12}P_4H_6P$ H_2PHP	811

Table 3 CPU time for run on extended heuristic algorithm.

The machine that we use for realization of the computational experiments is laptop with Intel Core i5 430M (2.26 GHz, 3MB L3 cache) processor and 4GB RAM. We not compare the execution time with the other algorithms because they have different mode of operation.

6 Conclusion

In this work is shown that the heuristic algorithm for 2D lattice HP model, described by Traykov et al. (2016), can be successfully applied to the 3D protein folding problem. Simulation results indicate that our approach performs better than those of Evolutionary Algorithm with Backtracking, Meta-Heuristic Ant Colony Optimization Algorithm and Genetic Algorithm.

7 Acknowledgment

This work is partially supported by the project of the Bulgarian National Science Fund, entitled: Bioinformatics research: protein folding, docking and prediction of biological activity, code NSF I02/16, 12.12.14.

References:

[1] K. Dill, Theory for the folding and stability of lobular proteins, *Biochemistry-US*, Vol. 24, 1985, pp. 1501-1509.

[2] K. Dill, K. Lau, A Lattice Statistical Mechanics Model of the Conformational Sequence Spaces of Proteins, *Macromolecules*, Vol. 22, 1989, pp. 3986-3997.

[3] C. Lin, M. Hsieh, An efficient hybrid Taguchi-genetic algorithm for protein folding simulation, *Expert Systems with Applications*, Vol. 36, 2009, pp. 12446-12453.

[4] J. Blazewick, K. Dill, P. Lukasiak, et al., A Tabu Search Strategy For Finding Low Energy Structures Of Proteins In Hp-Model, *CMST*, Vol. 10, 2004, pp. 7-19.

[5] S. Istrail, A. Hurd, R. Lippert, et al., *Prediction of self-assembly of energetic tiles and dominoes: Experiments, mathematics, and software*, Technical Report SAND2002, 2000, Sandia National Laboratories.

[6] S. Istrail, F. Lam, Combinatorial algorithms for protein folding in lattice models: A survey of mathematical results, *Commun. Inf. Syst.*, Vol. 9, 2009, pp. 303-346.

[7] M. Chen, W. Huang, A branch and bound algorithm for the protein folding problem in the HP Lattice Model, *Genomics Proteomics Bioinformatics*, Vol. 3, 2005, pp. 225-230.

[8] L. Toma, S. Toma, Contact interactions method: A new algorithm for protein folding simulations, *Protein Sci.*, Vol. 5, 1996, pp. 147-153.

[9] Y. Zhang, J. Skolnick, Tertiary structure predictions on a comprehensive benchmark of medium to large size proteins, *Biophys J.*, Vol. 87, 2004, pp. 2647-2655.

[10] N. Krasnogor, D. Pelta, P. Lopez, et al., Genetic Algorithms for the Protein Folding Problem: A Critical View, *Proc. Engineering of intelligent systems*, 1998, pp. 353-360.

[11] F. Liang, W. Wong, Evolutionary Monte Carlo for Protein Folding Simulations, *J. Chem. Phys.*, Vol. 115, 2001, pp. 444-451.

[12] A. Shmygelska, H. Hoos, An ant colony optimisation algorithm for the 2D and 3D hydrophobic polar protein folding problem, *BMC Bioinformatics*, Vol. 6, 2005, Article No.30.

[13] N. Yanev, M. Traykov, P. Milanov, B. Yurukov, Protein folding prediction in a cubic lattice in hydrophobic-polar model, *Journal of Computational Biology*, Vol. 24, 2017, pp. 412-421.

[14] N. Yanev, P. Milanov, I. Mirchev, Integer programming approaches to HP folding, *Serdica J. Computing*, Vol. 5, 2011, pp. 359-366.

[15] M. Traykov, S. Angelov, N. Yanev, A New Heuristic Algorithm for Protein Folding in the HP Model, *J Comput. Biol.*, Vol. 23, 2016, pp. 662-668.

[16] N. Thilagavathi, T. Amudha, Aco-metaheuristic for 3D-hp protein folding optimization, *ARPN Journal of Engineering and Applied Sciences*, Vol. 10, 2015, pp. 4948-4953.
[17] C. Lin, S. Su, Protein 3D HP Model Folding Simulation Using a Hybrid of Genetic Algorithm and Particle Swarm Optimization, *Int. J. Fuzzy Syst*, Vol. 13, 2011, pp. 140-147.
[18] C. Cotta, Protein structure prediction using evolutionary algorithms hybridized with backtracking, *Artificial Neural Nets Problem Solving Methods*, Vol. 2687, 2003, pp. 321-328.