Energy-landscape paving for prediction of face-centered-cubic hydrophobic-hydrophilic lattice model proteins

Jingfa Liu,[1,2] Beibei Song,[1,3] Zhaoxia Liu,[2] Weibo Huang,[1,3] Yuanyuan Sun,[1,3] and Wenjie Liu[1,3]
[1]*Jiangsu Engineering Center of Network Monitoring, Nanjing University of Information Science and Technology, Nanjing 210044, China*
[2]*Network Information Center, Nanjing University of Information Science and Technology, Nanjing 210044, China*
[3]*School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China*
(Received 7 August 2013; published 6 November 2013)

Protein structure prediction (PSP) is a classical NP-hard problem in computational biology. The energy-landscape paving (ELP) method is a class of heuristic global optimization algorithm, and has been successfully applied to solving many optimization problems with complex energy landscapes in the continuous space. By putting forward a new update mechanism of the histogram function in ELP and incorporating the generation of initial conformation based on the greedy strategy and the neighborhood search strategy based on pull moves into ELP, an improved energy-landscape paving (ELP+) method is put forward. Twelve general benchmark instances are first tested on both two-dimensional and three-dimensional (3D) face-centered-cubic (fcc) hydrophobic-hydrophilic (HP) lattice models. The lowest energies by ELP+ are as good as or better than those of other methods in the literature for all instances. Then, five sets of larger-scale instances, denoted by S, R, F90, F180, and CASP target instances on the 3D FCC HP lattice model are tested. The proposed algorithm finds lower energies than those by the five other methods in literature. Not unexpectedly, this is particularly pronounced for the longer sequences considered. Computational results show that ELP+ is an effective method for PSP on the fcc HP lattice model.

DOI: 10.1103/PhysRevE.88.052704 PACS number(s): 87.15.Cc, 87.15.ak, 05.10.Ln

I. INTRODUCTION

Protein engineering is a frontier in modern biotechnology and the prediction of protein structure is crucial to pharmacology and medical science. There are some experimental methods to find the native state of a protein, e.g., nuclear magnetic resonance (NMR), x-ray crystal diffraction, etc. However, these methods are costly, time consuming, and labor intensive. So using a computer to simulate the protein structure has been an important method to solve the protein structure prediction (PSP) problem.

According to Anfinsen's thermodynamic hypothesis, states of minimum free energies and the tertiary structures of proteins can be predicted from the linear sequences of their amino acids [1]. However, even for the simplest hydrophobic-hydrophilic (HP) lattice model [2,3], PSP has been proven to be "NP-complete" [4,5]. Since deterministic approaches are not helpful in identifying minimum energy conformations [6], to find a nondeterministic heuristic approach that can extract minimal energy conformations efficiently is of great importance [7]. In addition, an appropriate energy function which can generally distinguish the native state from non-native states of a protein molecule is another vital factor to predict protein structure successfully. The greatest difficulty lies in the huge search space, as well as the complexity of the energy surface, which contains a lot of local minima and a few global minima.

To simplify many of the required calculations, we choose the lattice model which captures the main features of the PSP. In this paper, we focus on the face-centered-cubic (fcc) HP lattice model which is shown to yield very good approximations of real protein structures [8–10]. Some outstanding heuristic approaches, such as tabu search (TS) [11] with pull moves [6], evolutionary algorithm (EA) with lattice rotation for crossover and K-site move for mutation [12], tabu-based local search method (LS-Tabu) [13,14], tabu-based spiral search algorithm (SS-Tabu) [14], simple genetic algorithm (SGA) [7,15], and its variations [simple genetic algorithm with twin removal (SGA + TR) [15], hybrid genetic algorithm (HGA) [7,15,16] which combines generalized short pull moves and improved crossover and mutation operations, hybrid genetic algorithm with twin removal (HGA + TR) [11,15,16], genetic algorithm with elite-based reproduction strategy (ERS-GA) [9], hybrid of hill-climbing and genetic algorithm (HHGA) [9] based on ERS-GA, memetic algorithm (MA) [17], and large neighborhood search (LNS) [18], were applied to a fcc HP lattice model. All of these methods cannot guarantee one to obtain optimal results in polynomial time. Later, a constraint-based protein structure prediction (CPSP) approach [19] was put forward. Once the corresponding H core [19] is given, the approach can ensure that all predicted structures are globally optimal.

The ELP [20] method is a class of global optimization methods. ELP was originally proposed by Hansmann and Wille [20] to simulate protein structure of all-atom proteins, such as pentapeptide Met-enkephalin and 36-residue peptide (HP-36) modeled by the ECEPP/2 force field, and hereafter was introduced for the off-lattice model proteins [21–23] and the circular packing problems [24,25]. In this paper, an ELP+ method which incorporates the generation of initial conformation based on the greedy strategy and the neighborhood search strategy based on pull moves [6] into the ELP method is proposed for PSP. In addition, an update mechanism of the histogram function is put forward. To the best of our knowledge, few researchers have applied ELP to the discrete optimization problem. In our current work, to demonstrate the efficiency of ELP in discrete space, we further improve the ELP method and use it as a tool to fold up given sequences on a fcc lattice model. Numerical results show that

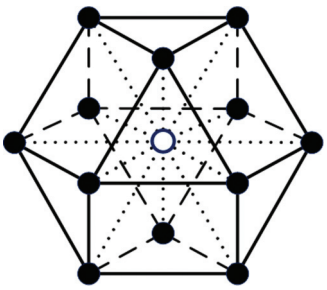


FIG. 1. (Color online) A unit of the 3D fcc HP lattice model. An amino acid at layer 2 has 12 neighbors, three of which are from the top layer, six are from the middle layer, and the other three are from the bottom layer.

ELP+ is an effective algorithm for solving PSP in a fcc HP lattice model.

II. fcc HP LATTICE MODEL

The HP lattice model [2,3] is the most frequently used model, which is based on the observation that the hydrophobic interaction between amino acids is the main driving force for protein folding, i.e., the development of native states in proteins [2]. In this model, amino acids are represented as a reduced set of H (hydrophobic or nonpolar) and P (hydrophilic or polar) according to the hydrophobicity of a single amino acid. Despite the fact that two-dimensional (2D) square and three-dimensional (3D) cube models [2,3,26] have been used mostly among HP lattice models, there exists a significant drawback that if two amino acids are at any even distance in the primary sequence, they cannot be neighbors in the lattices. To address this issue, Hart *et al.* [10] introduced a fcc HP lattice model which is parity problem free, that is to say, an odd indexed amino acid in the sequence can be the neighbor of both odd and even indexed amino acids in the sequence and vice versa. In addition, the famous Kepler conjecture [27,28] implies that fcc is the densest sphere-packing model, where an amino acid can have 12 neighbors in the 3D fcc lattice (see Fig. 1) and six neighbors in the 2D fcc lattice which form a hexagon (see Fig. 2).

A folding of a protein in the HP lattice model means that amino acids are embedded in the lattice such that adjacent amino acids in the sequence occupy adjacent grid points in the lattice and no grid point in the lattice is occupied by more than one amino acid. This is also called self-avoiding walk (SAW). In fact, the 2D fcc lattice is the infinite graph $G = (V,L)$, where

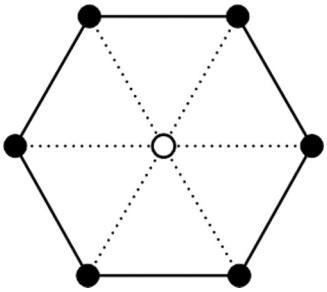


FIG. 2. A unit of the 2D fcc HP lattice model. Each amino acid has at most six neighbors.

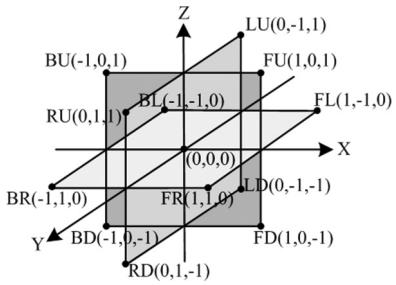


FIG. 3. 12 basis vectors of the 3D fcc HP lattice model.

the vertex set $V = (\sqrt{3}ZZ) \cup [(\sqrt{3}Z + \sqrt{3}/2)(Z + 1/2)]$, and the edge set $L = \{(x,x')|x,x' \in V, \|x - x'\| = 1\}$. Here Z denotes the integer set and $\|x - x'\|$ denotes the Euclidean distance between x and x'. The 3D fcc grids can be described as a stack of 2D fcc grids, where every individual 2D grid is slightly offset with respect to the grids above and below it [11]. The basis vectors are $(1,-1,0)$, $(-1,1,0)$, $(-1,-1,0)$, $(1,1,0)$, $(0,-1,1)$, $(0,-1,-1)$, $(1,0,1)$, $(1,0,-1)$, $(0,1,1)$, $(-1,0,1)$, $(0,1,-1)$, and $(-1,0,-1)$, denoted by forward-left (FL), backward-right (BR), backward-left (BL), forward-right (FR), left-up (LU), left-down (LD), forward-up (FU), forward-down (FD), right-up (RU), backward-up (BU), right-down (RD), and backward-down (BD) (see Fig. 3), respectively. Two 3D fcc points $P_i(x_i,y_i,z_i)$ and $P_j(x_j,y_j,z_j)$ are adjacent if and only if $(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2 = 2$. The energy $E(c)$ of a given conformation c is defined as the number of topological neighboring (TN) contacts between those Hs, which are not sequential with respect to the sequence. In other words, if a conformation denoted as $c = l_1 l_2 \cdots l_n$, where l_i is H if the ith amino acid in the sequence is hydrophobic and P if it is hydrophilic, has exactly m such H-H TN contacts, its energy $E(c) = m(-1)$. Figure 4 shows a conformation with the energy of -15 in the 2D fcc HP lattice model. Since each amino acid has two covalent neighbors, except the first and the last amino acids, a nonterminal and a terminal amino acid can have a maximum of four TNs and five TNs, respectively.

PSP can be formally defined as follows: Given an HP sequence $s = s_1 s_2 \cdots s_n$, we try to find a conformation with minimum energy of s, that is, to find $c^* \in C(s)$ such that $E(c^*) = \min\{E(c)|c \in C(s)\}$, where $C(s)$ is the set of all valid conformations (i.e., SAW) of s.

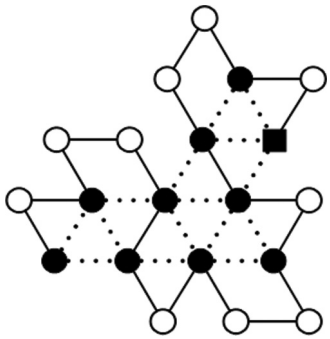


FIG. 4. A conformation with the energy of -15. "•" ("■") and "○" indicate the hydrophobic and hydrophilic amino acids, respectively. "–" denotes the binding edge and "⋯" is the topological neighboring contact edge. "■" is the first amino acid of the sequence.

III. METHODS

A. ELP method

The ELP method [20–25], which merges ideas from tabu search [29] with energy-landscape deformation [30], is a class of heuristic global optimization algorithm and a generation of Monte Carlo (MC) method. Simple canonical MC is easily get trapped in local minima. To avoid the search entrapping in local minima while exploring minimal energy conformations, it redefines the energy function so that there is little chance for ELP to search the regions that have been explored. This means if a conformation c is hit at a MC sweep t, the energy $E(c,t)$ is increased by a "penalty" and replaced by energy $\widetilde{E}(c,t) = E(c,t) + f(H(q,t))$. Here, the penalty term $f(H(q,t))$ is a function of the histogram $H(q,t)$ in prechosen "order parameter" q. In this paper, we set $q = E$ and choose $kH(E(c,t),t)$ as the replacement for $f(H(q,t))$. Here, k is a constant and $H(E(c,t),t)$ is the histogram function in energy at a MC sweep t. In fact, the histogram function $H(E(c,t),t)$ from all previously visited energies helps the simulation escape local entrapments and surpass the high-energy barrier more easily. If $E(c,t)$ falls into a certain bin, the corresponding bin is increased by 1, where a "bin" denotes an entry of the histogram and all bins in the histogram are the same at the beginning of the algorithm. We set the size of every bin as 1 in the fcc HP lattice model. The sampling weight is defined as $\omega(\widetilde{E}(c,t)) = \exp[-\widetilde{E}(c,t)/k_B\mathrm{T}]$, where k_BT is the thermal energy at the low temperature T and k_B is the Boltzmann constant. The more time the system stays in a local minimum, the less the sampling weight of a local minimum state.

ELP deforms the energy landscape locally until the local minimum is no longer favored and the system will explore higher energies. It will then either fall in a new local minimum or walk through this high-energy region until the corresponding histogram entries all have similar frequencies and the system again has a bias toward low energies. However, there is a technical flaw in ELP [24,25]. After new conformation c_2 generates from the current conformation c_1 by the conformation update mechanism, the algorithm accepts c_2 only by satisfying the condition expression random(0,1) $<$ $\exp\{\widetilde{E}(c_1,t) - \widetilde{E}(c_2,t)]/k_BT\}$, where random(0,1) is the random number between 0 and 1. However, by this acceptability condition, ELP may miss some lower-energy conformations near c_1. To avoid it, Liu *et al.* [24,25] give an alternative version of ELP. Now the acceptability of c_2 is determined by a comparison between $E(c_1,t)$ and $E(c_2,t)$, where two cases are possible: (a) $E(c_2,t) < E(c_1,t)$ and (b) $E(c_2,t) \geqslant E(c_1,t)$. For case (a), c_2 is accepted unconditionally and a new round of iteration starts; for case (b), if c_2 satisfies the condition expression random(0,1) $< \exp\{[\widetilde{E}(c_1,t) - \widetilde{E}(c_2,t)]/k_BT\}$, then c_2 is still accepted and another round of iteration starts; otherwise c_2 is not accepted and c_1 is restored as the current conformation.

We note that, in the original version of ELP by Hansmann and Wille [20] and its alternative version by Liu *et al.* [24,25], once a new conformation is generated, the histogram function is updated no matter whether the new conformation is accepted. This will cause some newly generated and unaccepted conformations which locate at the surrounding energy barriers of the minima to still not be accepted in subsequent simulations because the accumulated histogram function $H(E(c,t),t)$ gradually modifies their surrounding energy barriers. Thus it will be hard for the simulation to escape from the minima, especially from those located at narrow and deep valleys of energy landscape. To overcome this, we further improve ELP and propose an update mechanism of the histogram function, i.e., we update the histogram only when the newly generated conformation is accepted by the above-mentioned acceptability criteria.

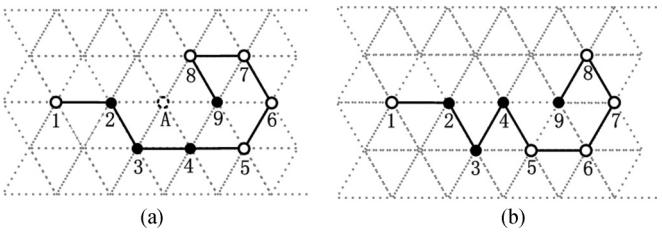


FIG. 5. An example of the forward pull move on the 2D fcc HP lattice model. "•" and "○" indicate the hydrophobic and hydrophilic amino acids, respectively. If position A is free, then amino acid 4 can be placed at A, and a forward pull move in (a) can be executed, where amino acid 5 is moved to the position of amino acid 4, 6 to 5, 7 to 6, and 8 to 7, then a valid conformation [indicated in (b)] is obtained.

B. Greedy strategy for the initial conformation

The improved ELP algorithm must start with a valid initial conformation, differently from the original version [20] and the alternative version of ELP [24,25], where the initial conformations can be invalid. To reduce the cost of the generation of a valid initial conformation and enhance the efficiency of the search of the improved ELP method, we use the greedy strategy to get the initial conformation for a given protein sequence with length n. The detailed steps are as follows: We put the first amino acid and the second one at two adjacent fixed positions, respectively, for example, at (0,0) and (0,1) for two dimensions, and at (0,0,0) and (0,1,1) for three dimensions. Subsequently we pseudoplace the ith ($3 \leqslant i \leqslant n$) amino acid at every position that is adjacent to the $(i-1)$th amino acid and not occupied by other amino acids, where "pseudoplace" means that the ith amino acid is placed temporarily and will be removed from the corresponding position after computing the energy of the partial conformation, which consists of the previous $i-1$ amino acids and the ith amino acid. If such positions exist, we put formally the ith amino acid at the position where the energy of the corresponding partial conformation is lowest; otherwise we remove the $(i-1)$ amino acid and continue to grow the partial conformation from the $(i-2)$ amino acid. This process is repeated until a conformation with n amino acids is produced.

C. Neighborhood search strategy with pull moves

In ELP, each MC step must update the current conformation. We use the neighborhood search strategy with pull moves [6] to update the conformation. The neighborhood of a conformation c is a set of valid conformations that are obtained by applying a specific set of perturbations to c. The pull move, originally introduced by Lesh *et al.* [6] for square and cube

lattices, has been proven to be very efficient in the HP model under a variety of local search methods [6,31]. The set of pull moves is complete and reversible [6,11] for square, cube, and fcc HP lattice models, which makes it efficient for updating the conformation and essential to guarantee reachability of the global minimum. As the parity problem is absent in fcc HP lattice models, the pull move does not need to be moved diagonally to start as an ordinary pull move in square and cube lattices.

First, we briefly describe the main idea of the pull move on the 2D fcc lattice. We choose randomly a vertex from the chain with length n such that there exists a free position in the grid adjacent to both this vertex and either its predecessor or successor in the chain and then move it to this free position [see Fig. 5(a)]. This might break the chain, so we need to repair the chain. This repairing is done via pulling the chain, i.e., the old position of the moved vertex will be occupied by its successor (or predecessor), again leaving a free position where the next vertex of the chain is moved, and so on, until a valid conformation is reached.

The pull move of an amino acid can be performed only when there exists at least one free position of its neighbors. During the process of the pull move, if the ith amino acid is moved first, we define the pull move as the pull move of the ith amino acid. Consider the pull move of the ith amino acid, whose position is (x_i,y_i). We define two kinds of moves. One is the direct move, and the other is the forward-backward pull move. The detailed description of pull moves is as follows. If there exists an index $i \in \{2,\ldots,n-1\}$ and a vertex A which is empty and adjacent to both the ith and $(i-1)$ amino acid, we can perform a forward pull move. If A is also adjacent to the $(i+1)$ amino acid, we move directly amino acid i to A. Thus a new legal conformation is reached. We call this pull move the direct move. Otherwise, we move amino acid i to A, $i+1$ to i, $i+2$ to $i+1$, and so on, until a valid conformation is reached. An example of the forward pull move is shown in Fig. 5, where $i = 4$ and $n = 9$. If $i = 1$, we call the pull move front-end pull move which is a special case of forward pull move. Backward and back-end pull moves can be defined similarly.

In the improved ELP method, for a conformation c, we randomly choose the ith $(1 \leqslant I \leqslant n)$ amino acid and first "pseudomove" it to its every legal adjacent position. Then we complete remaining moves by pull-move rules until new legal conformations are reached. After computing the energies of the corresponding conformations for all legal positions, we move formally the ith amino acid to the position where the energy of the conformation obtained by pull moves is lowest. This process is repeated until a new conformation is accepted or pull moves on all n amino acids are executed but no conformation is accepted. If the latter happens, we restore the previous conformation $\bar{c}$ of the current conformation c as the new current conformation and continue a new round of iteration of ELP.

The pull move on a 3D fcc lattice can be similarly defined as above. Differently, in a 2D fcc lattice, the ith $(1 \leqslant i \leqslant n)$ amino acid may at most be moved to six adjacent positions (see Fig. 2), but in a 3D fcc lattice it may be moved at most to 12 adjacent positions (see Fig. 1).

D. Description of algorithm

By putting forward a new update mechanism of the histogram function in ELP and incorporating the generation of an initial conformation based on the greedy strategy and the neighborhood search strategy with pull moves into ELP, an ELP+ algorithm is proposed. The calculating procedure is presented as follows.

(1) Generate a valid initial conformation c based on the greedy strategy. Let $\bar{c} = c$, $c_{\min} = c$. Initialize k, T, and the largest iterative step number l. Let $t = 1$ and compute $E(c,t)$. Initialize the histogram function $H(\mathrm{E}(c,t),t)$, i.e., if the energy $E(c,t)$ of conformation c falls into a certain bin, then let the frequency of the corresponding bin be 1, and those of other bins be 0. Let $\tilde{E}(c,t) = E(c,t) + kH(E(c,t),t)$.

(2) Choose randomly an integer i from $N = \{1,2,\ldots,n\}$.

(3) Execute pull moves for all legal move positions of the ith amino acid of the current conformation c. If at least a pull move is executed successfully, we compute the energies of the corresponding legal conformations obtained by pull moves, and pick out the conformation with the lowest energy as an updated conformation of c, denoted as c', and go to step (4); otherwise go to step (7).

(4) Compute $E(c',t)$. If $E(c',t) < E(c_{\min},\mathrm{t})$, then let $c_{\min} = c'$, $E(c_{\min},t) = E(c',t)$.

TABLE I. Twelve sequences for fcc HP lattice model.

Instance	Length	Sequence
1	20	$HPHP_2H_2PHP_2HPH_2P_2HPH$
2	24	$H_2P_2(HP_2)_6H_2$
3	25	$P_2HP_2(H_2P_4)_3H_2$
4	36	$P_3H_2P_2H_2P_5H_7P_2H_2P_4H_2P_2HP_2$
5	48	$P_2H(PH_3)_2P_5H_{10}P_6(H_2P_2)_2HP_2H_5$
6	50	$H_2(PH)_3PH_4P(HP_3)_3P(HP_3)_2HPH_4(PH)_4H$
7	54	$H_2(PH)_3PH_4P(HP_3)_4P(HP_3)_2HPH_4(PH)_4H$
8	60	$P(PH_3)_2H_5P_3H_{10}PHP_3H_{12}P_4H_6PH_2PHP$
9	64	$H_{12}(PH)_2\,[(P_2H_2)_2P_2H]_3(PH)_2H_{11}$
10	85	$H_4P_4H_{12}P_6(H_{12}P_3)_3HP_2(H_2P_2)_2HPH$
11	100_a	$P_3H_2P_2H_4P_2H_3(PH_2)_3H_2P_8H_6P_2H_6P_9HPH_2PH_{11}P_2H_3PH_2PHP_2HPH_3P_6H_3$
12	100_b	$P_6HPH_2P_5H_3PH_5PH_2P_2(P_2H_2)_2PH_5PH_{10}PH_2PH_7P_{11}H_7P_2HPH_3P_6HPH_2$

TABLE II. The step number l of iterations and the time number t of runs by ELP+ in different cases.

	2D fcc			3D fcc						
Model instance	1–4	5–9	10–12	1–4	5–12	S	R	F_90	F_180	CASP targets
l	5×10^5	3×10^6	3×10^7	5×10^5	5×10^6	3×10^7	3×10^7	5×10^6	3×10^7	3×10^7
t	50	50	30	30	30	20	20	30	20	20

(5) If $E(c',t) < E(c,t)$, then let $\bar{c} = c$, $c = c'$, and update $H(E(c',t),t)$, and let $\tilde{E}\ (c',t) = E(c',t) + kH(E(c',t),t)$, and go to step (8); otherwise go to step (6).

(6) If random(0,1) $<$ exp{$[\tilde{E}\ (c,t) - \tilde{E}\ (c',t)]/k_BT$}, then let $\bar{c} = c$, $c = c'$, and update $H(E(c',t),t)$, and let $\tilde{E}(c',t) = E(c',t) + kH(E(c',t),t)$, go to step (8); otherwise go to step (7).

(7) If all integer numbers between 1 and n are chosen, then let $c = \bar{c}$, and go to step (8); otherwise let $N = N - \{i\}$ and choose randomly another integer j from N. Let $i = j$, and go to step (3).

(8) If $t >$ l, then output the lowest energy conformation $c_{\min}$ and stop; otherwise let $t = t + 1$, and go to step (2).

IV. NUMERICAL RESULTS

We test the ELP+ algorithm on both 2D and 3D fcc HP lattice models. The tested instances include 12 general instances listed in Table I which have been partly used in literature [11,12,16–18] and the five sets of longer sequences denoted by the S, R, F90, F180, and CASP target instances. The S, R, F90, and F180 instances are taken from Ref. [20] and six CASP target instances from the CASP website http://predictioncenter.org/casp9/targetlist.cgi. The corresponding CASP target IDs for proteins *3mse*, *3mr7*, *3mqz*, *3no6*, *3no3*, and *3on7* are *T0521*, *T0520*, *T0525*, *T0516*, *T0570*, and *T0563*. To fit in the HP model, the CASP targets are converted to HP sequences based on the hydrophobic properties. The 20 constituent amino acids of proteins are broadly divided into two categories: (a) hydrophobic amino acids denoted as H (Gly, Ala, Pro, Val, Leu, Ile, Met, Phe, Tyr, and Trp); and (b) hydrophilic or polar amino acids denoted as P (Ser, Thr, Cys, Asn, Gln, Lys, His, Arg, Asp, and Glu). To make the ELP+ algorithm perform well, we do a parameter study in the same way as we have done in Ref. [25]. In this paper, we set $k = 0.1$ and $T = 5$. For every instance, the step number l of iterations and the time number t of runs by ELP+ are shown in Table II. Since our algorithm is a stochastic algorithm, we cannot guarantee the algorithm can obtain the optimal result in each run within given step numbers. So, we give the lowest energy and the corresponding average value in all runs for each instance. We implement the algorithm in Java language and run it on a desktop computer with an Intel Core 2 Duo, 1.6 GHz processor and 2.0 GB of RAM.

A. Numerical results on 2D HP fcc lattice model

First, we test the ELP+ algorithm on the 2D fcc HP lattice model and compare our results with those from SGA [7], HGA [7] which combines generalized short pull moves and improved crossover and mutation operations, HGA + TR [11], ERS-GA [9], HHGA [9] based on ERS-GA, and TS [11]. The lowest energies found by ELP+ and the other methods on 12 general instances listed in Table I are shown in Table III. From Table III, we can see that our algorithm can reach lowest energies so far for four short sequences and find new lower free energies than the other six methods for five larger instances. For instances 1–4, TS and ELP+ can easily obtain the lowest energies presented in literature. For instances 5–9, it

TABLE III. Comparison of computational results by different methods on the 2D fcc lattice model.

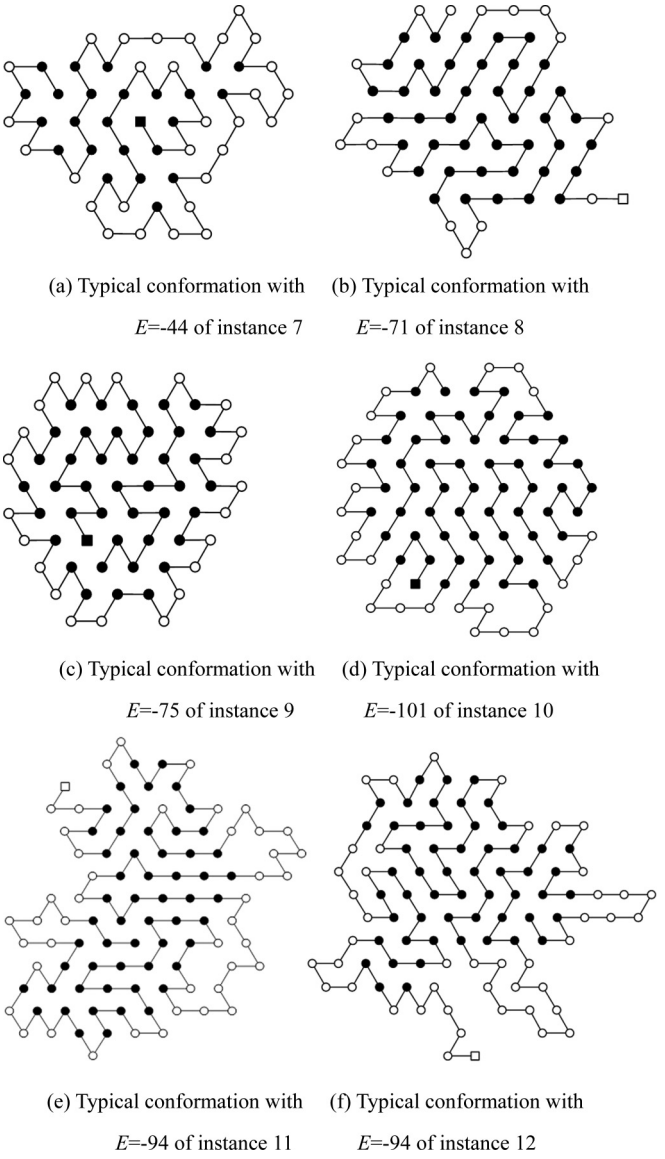
Instance[a]	SGA	HGA	HGA + TR	ERS-GA	HHGA	TS	ELP	ELP+	E-eval.[c]	CPU time(s)[d]
1	−11	**−15**	**−15**	−15	**−15**	**−15**	**−15**(−15)[b]	−15(−15)	796	0.001[d]
2	−10	−13	−13	−13	**−17**	**−17**	**−17**(−17)	−17(−17)	12120	0.189
3	−10	−10	−10	**−12**	**−12**	**−12**	−12(−12)	**−12**(−12)	7384	0.051
4	−16	−19	−19	−20	−23	**−24**	**−24**(−24)	**−24**(−24)	114162	0.049
5			−32	−32	−41	−40	−43(−43)	**−44**(−43.3)	2205135	21.33
6				−30	−38		−36(−34.4)	**−42**(−40.1)	1889451	45.01
7	−21	−23	−23			−31	−41(−38.1)	**−44**(−41.9)	2444678	20.12
8	−40	−46	−46	−55	−66	−70	−70(−67.5)	**−71**(−70.1)	2886590	91.02
9	−33	−46	−46	−47	−63	−50	−72(−67.7)	**−75**(−74.1)	2689831	119.94
10							−100(−97.8)	**−101**(−100.2)	7457973	4009.5
11							−89(−87.8)	**−94**(−93)	11434980	12556.97
12							−92(−90.4)	**−94**(−93.2)	25652309	13012.06

[a]Instances are taken from Table I.

[b]Numbers in bold indicate the lowest energies so far and the numbers in parentheses denote the average energies found over the given run times in Table II.

[c]Average number of energy evaluation (E-eval.) before the lowest energy is found by ELP+ .

[d]Average CPU time (second) needed to find the lowest energy by ELP+ over the given run times in Table II.



(a) Typical conformation with E=-44 of instance 7

(b) Typical conformation with E=-71 of instance 8

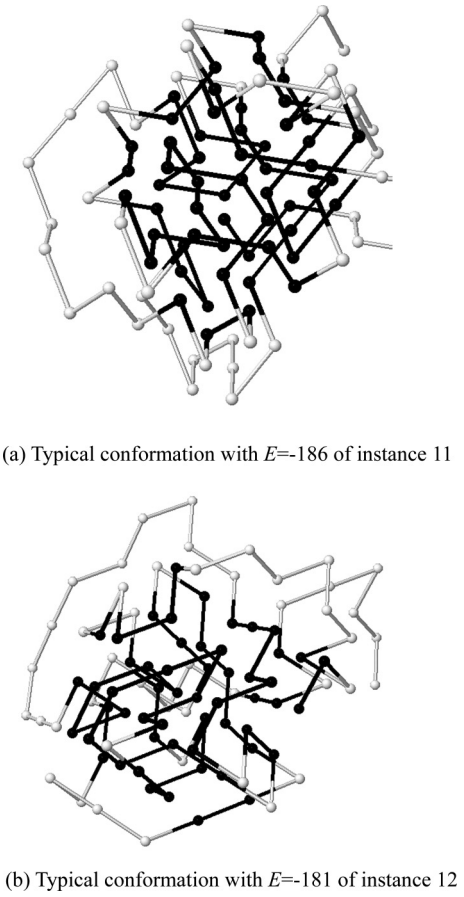
(c) Typical conformation with E=-75 of instance 9

(d) Typical conformation with E=-101 of instance 10

(e) Typical conformation with E=-94 of instance 11

(f) Typical conformation with E=-94 of instance 12

FIG. 6. Conformations with the lowest energies found by the ELP+ algorithm on the 2D fcc HP lattice model. "•" ("■") and "○" ("□") indicate the hydrophobic and hydrophilic amino acids, respectively. "■" or "□" denotes the first amino acid of each sequence.



(a) Typical conformation with E=-186 of instance 11

(b) Typical conformation with E=-181 of instance 12

FIG. 7. Conformations with the lowest energies found by the ELP+ algorithm for instances 11 and 12 on the 3D fcc HP lattice model. The black ball and the white ball indicate the hydrophobic and hydrophilic amino acids, respectively.

is noteworthy that we find lower energies of −44, −42, −44, −71, and −75, respectively, which are missed by the other six methods. Three longer sequences with lengths from 85 to 100 taken from Ref. [17] are also tested for future comparison with other methods.

To investigate the effects of the new update strategy of histogram function, we further test the ELP+ method without this update strategy (the following briefly describes ELP). The parameters used for ELP and ELP+ are the same, so the

TABLE IV. Comparison of computational results by different methods on the 3D fcc lattice model for longer eight instances in Table I.

Instance[a]	SGA	SGA + TR	HGA	HGA + TR	MA	TS	EA	ELP+
5				−69		**−74**[b]	**−74**	**−74**(−74)[c]
6	−55	−56	−59		−69		**−73**	**−73**(−72.6)
7				−59		**−77**		−77(−76.6)
8	−97	−112	−114	−117	−122	**−130**	**−130**	**−130**(−130)
9	−81	−90	−98	−103	−114	**−132**	**−132**	**−132**(−132)
10					−165			**−189**(−188.2)
11					−156			**−186**(−185)
12								**−181**(−180.2)

[a]Instances are taken from Table I.

[b]Numbers in bold indicate the lowest energies by the eight methods.

[c]The numbers in parentheses are the average energies obtained by ELP+ .

comparison will be more convincing. Table III shows that the results by the ELP+ algorithm are as good as or better than the ELP algorithm. This is reasonable because the ELP+ method without this update strategy (ELP) is harder to escape the minima located at narrow and deep valleys of energy landscape than ELP+, and may miss some lower-energy conformations during limited CPU times. The average CPU time of the lowest energy by ELP+ for every instance is also listed in Table III. However, because the six methods in the literature do not report their running times, we cannot compare the speed of our algorithm with the other methods' in detail. Considering the differences in the performances of the running computers and the programming languages, to detailedly compare the effectiveness of ELP+ with other methods in the future, we also give the average number of energy evaluation (i.e., the number of valid conformations scanned) before the lowest energy is found by ELP+ for each instance. Figure 6 shows typical representatives of the lowest-energy conformations obtained by ELP+ for instances 7–12. It is obvious that each of these conformations possesses a compact hydrophobic core.

B. Numerical results on 3D HP fcc lattice model

Further, to verify the effectiveness of the ELP+ algorithm, we apply it on the 3D fcc HP lattice model. First, we test 12 general instances listed in Table I. For four shorter sequences, the ELP+ algorithm can easily obtain the optimal results in literature, and for eight longer ones, the computational results of our algorithm are listed in Table IV, in comparison with the other seven algorithms, including SGA [15], SGA + TR [15], HGA [15,16] which combines generalized short pull moves and improved crossover and mutation operations, HGA + TR [11,15,16], MA [17], TS [11], and EA [12] with lattice rotation for crossover and K-site move for mutation. As seen from Table IV, the results of our algorithm are as good as or better than those of the other seven algorithms. The lowest energies by the ELP+ algorithm for instances 5, 8, and 9 are quite consistent with those of TS and EA, and are lower than those by the other five algorithms. For instance 6, both our algorithm and EA find the lowest energy which is missed by SGA, SGA + TR, HGA, and MA. For instance 7, our method also gets the optimal energy which is also obtained by TS. Compared with the results by MA, the ELP+ algorithm obtains much lower energies for instances 10 and 11. Instance 12 is also tested for future comparison with other methods. Typical conformations with the lowest energies found by ELP+ for instances 11 and 12 are shown in Fig. 7.

We also test the other five sets of larger-scale instances, denoted as the S, R, F90, F180, and CASP target instances, respectively. Table V summarizes the ELP+ algorithm's performance together with those by multiple sequence reoptimized LNS (LNS-MULT) [18], 3D structure reoptimized LNS (LNS-3D) [18], tabu-based local search method (LS-Tabu) [14], memory-based local search method (LS-Mem) [14], and tabu-based spiral search algorithm (SS-Tabu) [14]. The numerical results show that our algorithm wins over LNS-MULT, LNS-3D, LS-Tabu, LS-Mem, and SS-Tabu over those proteins with a significant margin on both the lowest energies and average lowest energies.

From Table V, one can see that our algorithm gets the native energies for all the F90 instances, and explore the conformation surfaces more efficiently than LNS-MULT, LNS-3D, and

TABLE V. Comparison of computational results by different methods for the S, R, F90, F180, and CASP target instances on the 3D fcc HP lattice model.

Instance	Len.	Native E.[a]	LNS-MULT	LNS-3D	LS-Tabu	LS-Mem	SS-Tabu	ELP+
S1	135	−357	−349(−332.37)	−351(−336.74)	−351(−341)		−355(−347)	**−355**(−354.23)[b]
S2	151	−360	−349(−328.98)	−353(−334.17)	−355(−343)		−354(−347)	**−359**(−356.84)
S3	161	−367	−351(−323.77)	−353(−329.80)	−355(−340)		−359(−350)	**−364**(−362.63)
S4	164	−370	−346(−323.98)	−354(−334.22)	−354(−343)		−358(−350)	**−365**(−362.63)
R1	200	−384	−313(−287.98)	−330(−305.54)	−332(−318)	−353(−326)	−359(−345)	**−369**(−362.44)
R2	200	−383	−331(−289.83)	−333(−308.31)	−337(−324)	−351(−330)	−358(−346)	**−366**(−362.60)
R3	200	−385	−325(−288.49)	−334(−307.76)	−339(−323)	−352(−330)	−365(−345)	**−370**(−362.82)
F90_1	90	−168	−164(−156.83)	−165(−157.39)	−164(−160)		−168(−166)	**−168**(−166.23)
F90_2	90	−168	−163(−155.05)	−163(−155.81)	−165(−158)		−168(−164)	**−168**(−167.13)
F90_3	90	−167	−163(−156.23)	−163(−157.20)	−165(−159)		−167(−165)	**−167**(−166.00)
F90_4	90	−168	−164(−156.20)	−163(−156.54)	−165(−159)		−168(−165)	**−168**(−167.24)
F90_5	90	−167	−163(−155.77)	−164(−157.46)	−165(−159)		−167(−165)	**−167**(−166.18)
F180_1	180	−378	−289(−264.06)	−293(−269.07)	−338(−327)	−360(−334)	−357(−340)	**−363**(−357.68)
F180_2	180	−381	−302(−280.84)	−312(−287.21)	−345(−334)	−362(−340)	−359(−345)	**−364**(−362.83)
F180_3	180	−378	−306(−286.78)	−313(−295.31)	−352(−339)	−357(−343)	−362(−353)	**−368**(−363.45)
3mse	179	−323			−266(−249)	−278(−254)	−289(−280)	**−291**(−287.72)
3mr7	189	−355			−301(−287)	−311(−292)	−328(−313)	**−351**(−347.42)
3mqz	215	−474			−401(−383)	−415(−386)	−420(−403)	**−439**(−435.88)
3no6	229	−455			−390(−373)	−400(−375)	−411(−391)	**−415**(−411.43)
3no3	258	−494			−388(−359)	−397(−361)	−412(−393)	**−462**(−457.32)
3on7	279	?			−491(−461)	−499(−463)	−512(−485)	**−548**(−546.85)

[a]Native E. is the optimal energy and is obtained by using HPSTRUCT [32].

[b]Numbers in bold indicate the lowest energies by the six methods. The numbers in parentheses are the average lowest energies.

TABLE VI. Absolute walks of optimal conformations by ELP+ for three CASP target instances in the fcc lattice model.

Instance	Structure sequence
3no6	LDFDBLBDRDRUFDFRBUBDBUFUFLFRRDRUBLBRRUFUFLLULDFRFLBUFLBLFULUBRLUFRFRBRRDFRFRBUBLFU BLRUBDFRBRLDBUBDFDLUBULDBLFDFDRUFUBULUFDFUFDRDFUFDBDLUBDBDFRFRBDBURUBLLDFDFDLUFD LUFDBLFLRUBUBDLUBLBRBDBLFDFRLUBURURDBDLUBDBDFRRDFRRURDFLFRFLFLLUFDLDLUBDRURDBULUL ULDFDBDRUBRRUBDLURUFUFUBRLUBURDBLFDBLRDRDBDBUBUFRFDLUBLBDFDLUFRLUBLLDFDFRBDBLRDF RBUBDFRBRLUFRRDFUBRBLFUFRRUFLBURUFRLDLUBULDFDFRRDRDFUBUFLLDFRFDFLLULUFLRUBLBDLUFLF DBDBDLUBLBDRDBRFDBDBRFRFDLULDFRFUBURDRUBLRURDFDRDBD[a]
3no3	LUBDLURUBDBDFRRDLDFDRDBLBULUFDBDBUFLFLLUFLBUBRRDBUFRFRFRFRFLBLFDBRRDBURDBUBDFLRDF DBLFLFRBDFLFRLUBLFDFUBLLDBDRDBURDFRLUBRBLRUFLLUFUFRBRBLBLBLBLBLBDBLFDRDFUBRFUBURUF DBRFRLDFURDRDFUFLBDRDLDFDLUFDBLLUFLLURUBRFUFUFUFULDBDFRLDBLFLBUBDFDFRBRLDBURULUFU BRFUBUFRRDBDBLBLBLBUBRFDFRBRBUFRBRFRRDFULUBRRDFUBUFLLUBLLDRDBURUFDRUFRLUFDFRBDRUF UBLFLLDFRBDFRBDFDFLLDBLFLFDBLFDBLFDBDLDBURDFDRUFURUBDBULULDBDRDLDBLBLFUBURUBULULU FDRDLDBRRDBDRDFRFUFURUFLBUBRBDRDFLLUBULDBDFRFLLDFULULUFLRDRUBUFRBDBRLUFRFRRURUFLB LBRFUFLLDBUBURDRULUBRRUFRBU
3on7	LUFDRUFDBRBRLUFUFDBRRURDFLRULUFDFLRURDRDFUFDLDLDLDBRBDFDRDFDLDBLBRBRBUFRLUFURDRU BUBUFURUFLFULDFDLDRDRDFLLUFRRUBDLUFLLUFULUBUBDFLBDBDBDBLBRBDBDRUBDBUFRFDRDFLLDFU BULDFLRUBLFLFDRDFRRUFLRUBUFLLDFRFRRDBDBUFRBUBRLURUBDFRBRBURUFLFDBLLUFRFRLUFLFLRDBL LUFUBRBUBLFLBUBUBDFLFDFLBDLDRDBUBRBRBRLDFLBLLDBLRDFRFUFULDBDFRFLFDRDRUBRBUBRBRBRR DFLFLFLRURDBRLUFUBLBRRUFUFLBDFLFDFULULUBDBUBUBUFLLDFRLDLUBDBLFDLUBDRDFDRUFLFDRUBU FURURDFDBDBRFRBDBULDBLBDRULUBRFRFDRUFUFLRUFLRDFUFLFULULDBDLDLDBDFDBRFUFLFDBRFDBDF RBRBDBRBRBLFULDBLFULURURUFUFUFDFUFDFRBDBDBDFRFLLUBLRULULUBDFLRDBRRDBRFU

[a]Every two letters indicate a basis vector (see Fig. 3).

LS-Tabu. For the S instances, we also get lower energies than those by LNS-MULT, LNS-3D, LS-Tabu, and SS-Tabu. For all the R and F180 instances, it is noted that, we obtain lower energies which are missed by the other five methods. We also find different lower energies than LS-Tabu, LS-Mem, and SS-Tabu for the six CASP target instances. Although we cannot find the native energies for the S, R, F180, and six CASP target instances, we get much better results than the other five methods listed in Table V. In all cases, the native energy is obtained by using HPSTRUCT [32] which is state-of-the-art software and can exactly compute the native energy for the fcc HP lattice if one has access to the precomputed H cores by a different method. It is obvious that, HPSTRUCT outperforms our method for all instances, except for the five F90 instances; however, if an HP sequence has m H residues and there is no m-residue H core [32], then HPSTRUCT cannot run and if not all size m H cores are available, then HPSTRUCT may not converge. Even if H cores are available, HPSTRUCT may not converge within a prespecified time limit, in which case no answer is returned [18]. With regard to average lowest energies, from Table V, one can see that the energies by ELP+ are much lower than those by LNS-MULT, LNS-3D, LS-Tabu, LS-Mem, and SS-Tabu, especially for larger instances, for example, *3no3* and *3on7*. Structure sequences of typical lowest-energy conformations obtained by ELP+ for three CASP target instances are listed in Table VI, where every two letters indicate a basis vector (see Fig. 3).

V. CONCLUSION

Because of the huge search space and the roughness of protein free-energy landscape, it is easy for the search method to get trapped in local minima during the process of finding the ground-state conformation of a protein. To address this problem, a global optimization method ELP is proposed. The Metropolis sampling and the accumulated histogram function helps ELP escape local minima. The ELP method has been applied successfully in continuous space, e.g., the off-lattice protein-folding problems and the circular packing problems. However, few researchers have applied ELP to discrete space. To demonstrate the efficiency of the ELP method in discrete space, in this paper we apply an improved ELP method (ELP+) to solving the 2D and 3D fcc HP lattice protein-folding problems. In ELP+, we put forward an update mechanism of the histogram function and incorporate the generation of an initial conformation based on the greedy strategy and the neighborhood search strategy based on pull moves into ELP.

The numerical results show that ELP+ is a competitive optimization method for both 2D and 3D FCC HP lattice protein-folding problems. We get the optimal energies for most of the tested instances. In the future, we would hope to find a better combination between the global optimization method and the local search method to reduce the cost to find the minimal energy, as well as to apply our algorithm to predict the structures of real proteins.

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China (Grants No. 61373016 and No. 61103235) and its subproject (Grant No. 20110150-1), the Natural Science Foundation of Jiangsu Province (Grant No. BK2010570), the "Six Talent Peaks" of Jiangsu Province (Grant No DZXX-041), and Special Foundation of China Postdoctoral Science Foundation (Grant No. 201104572).

[1] C. B. Anfinsen, Science **181**, 223 (1973).
[2] K. F. Lau and K. A. Dill, Macromolecules **22**, 3986 (1989).
[3] K. A. Dill, Biochemistry **24**, 1501 (1985).
[4] R. Unger and J. Moult, Bull. Math. Biol. **55**, 1183 (1993).
[5] W. E. Hart, and S. Istrail, J. Comput. Biol. **4**, 1 (1997).
[6] N. Lesh, M. Mitzenmacher, and S. Whitesides, in *Proceedings of the Seventh Annual International Conference on Research in Computational Molecular Biology, Berlin, 2003*, edited by M. Vingron, S. Istrail, P. Pevzner, and M. Waterman (ACM, New York, 2003), p.188.
[7] M. T. Hoque, M. Chetty, and L. S. Dooley, in *Proceedings of the19th Australian Joint Conference on Artificial Intelligence, Hobart, 2006*, edited by A. Sattar and B. H. Kang (Springer, Berlin, 2006), LNAI, Vol. 4304, p.867.
[8] B. H. Park and M. Levitt, J. Mol. Biol. **249**, 493 (1995).
[9] S. C. Su, C. J. Lin, and C. K. Ting, Proteome Sci. **9**, S19 (2011).
[10] W. E. Hart and S. Istrail, J. Comput. Biol. **4**, 241 (1997).
[11] H. J. Bökenhauer, A. D. Ullah, L. Kapsokalivas, and K. Steinhöel, in *Proceedings of the 8th International Workshop on Algorithms in Bioinformatics, Karlsruhe, 2008*, edited by K. A. Crandall and J. Lagergren (Springer, Berlin, 2008), LNBI, Vol. 5251, p.369.
[12] S. C. Su and J. J. Tsay, in *Proceedings of the 2012 IEEE International Conference on Bioinformatics and Biomedicine, Philadelphia, 2012* (unpublished), p. 1.
[13] M. Cebrián, I. Dotú, P. V. Hentenryck, and P. Clote, in *Proceedings of the 23rd National Conference on Artificial intelligence, Chicago, 2008* (AAAI, Menlo Park, 2008), p. 241.
[14] M. A. Rashid, M. A. H. Newton, M. T. Hoque, S. Shatabda, D. N. Pham, and A. Sattar, BMC Bioinf. **14**, S16 (2013).
[15] M. T. Hoque, M. Chetty, and A. Sattar, in *Proceedings of the IEEE Congress Evolutionary Computation, Singapore, 2007* (unpublished), p. 4138.
[16] M. T. Hoque, M. Chetty, A. Lewis, and A. Sattar, IEEE/ACM Trans. Comput. Biol. Bioinform. **8**, 234 (2011).
[17] J. J. Tsay and S. C. Su, in *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine Workshop, Atlanta, 2011* (unpublished), p. 315.
[18] I. Dotu, M. Cebrián, P. V. Hentenryck, and P. Clote, IEEE/ACM Trans. Comput. Biol. Bioinform. **8**, 1620 (2011).
[19] M. Mann, S. Will, and R. Backofen, BMC Bioinf. **9**, 230 (2008).
[20] U. H. E. Hansmann and L. T. Wille, Phys. Rev. Lett. **88**, 068105 (2002).
[21] A. Schug, W. Wenzel, and U. H. E. Hansmann, J. Chem. Phys. **122**, 194711 (2005).
[22] J. F. Liu and W. Q. Huang, J. Theor. Comput. Chem. **5**, 587 (2006).
[23] J. F. Liu, S. J. Xue, D. B. Chen, H. T. Geng, and Z. X. Liu, J. Biol. Phys. **35**, 245 (2009).
[24] J. F. Liu, S. J. Xue, Z. X. Liu, and D. H. Xu, Comput. Ind. Eng. **57**, 1144 (2009).
[25] J. F. Liu and G. Li, Sci. China Inform Sci. **53**, 885 (2010).
[26] J. F. Liu, G. Li, and J. Yu, Phys. Rev. E. **84**, 031934 (2011).
[27] N. J. A. Sloane, Nature (London) **395**, 435 (1998).
[28] T. C. Hales, Ann. Math. **162**, 1065 (2005).
[29] D. Cvijovic and J. Klinowski, Science **267**, 664 (1995).
[30] G. Besold, J. Risbo, and O. G. Mouritsen, Comput. Mater. Sci. **15**, 311 (1999).
[31] K. Steinhöfel, A. Skaliotis, and A. A. Albrecht, in *Proceedings of the First International Conference on Bioinformatics Research and Development, Berlin, 2007*, edited by S. Hochreiter and R. Wagner (Springer, Berlin, 2007), Vol. 4414, p. 381.
[32] R. Backofen and S. Will, in *Proceedings of the 19th International Conference on Logic Programming, Mumbai, 2003,* edited by C. Palamidessi (Springer, Berlin, 2003), Vol. 2916, p. 49.