3D Protein Structure Prediction
with BSA-TS Algorithm

Yan Xu, Changjun Zhou, Qiang Zhang[(✉)], and Bin Wang

Key Laboratory of Advanced Design and Intelligent Computing,
Dalian University, Ministry of Education, Dalian, China
`zhangq26@126.com`

Abstract. Three-dimensional protein spatial structure prediction with the amino acid sequence can be converted to a global optimization problem of a multi-variable and multimodal function. This article uses an improved hybrid optimization algorithm named BSA-TS algorithm which combines Backtracking Search Optimization Algorithm (BSA) with Tabu Search (TS) Algorithm to predict the structure of protein based on the three-dimensional AB off-lattice model. It combines the advantage of BSA which has a simple and efficient algorithm framework, less control parameters and less sensitivity to the initial value of the control parameters and the advantage of TS which has a strong ability for the global neighborhood search, and can better overcome the short-comings of traditional algorithms which have slow convergence rate and are easy to fall into local optimum. At last we experiment in some Fibonacci sequences and real protein sequences which are widely used in protein spatial structure prediction, and the experimental results show that the hybrid algorithm has good performance and accuracy.

Keywords: Protein structure prediction · BSA · TS · 3D AB off-lattice model

1 Introduction

The study of protein structure helps us understand the function of proteins and to understand how they exercise their biological function, and understanding protein-protein interaction (or other molecules) is very important for biology, medicine and pharmacy [1]. In recent years, although the test methods for the determination of protein structures develop well, it's still time-consuming, expensive, and not applied for some proteins difficult to crystallize. Therefore, we need to develop the theoretical analysis. With the development of computer technology, it has gradually become an important tool for processing large data of protein molecule.

Theoretical Prediction of protein three-dimensional structure is mainly divided into the following three steps: Firstly, the mathematical model proposed should reflect the interaction among the amino acid residues; Secondly, we have to establish a simple calculated energy function which can also distinguish correctly between natural proteins and other proteins based on the thermodynamic hypothesis, Thirdly, we have to find the appropriate global optimization methods on the corresponding model to find the minimum of energy function. Owing to the efforts of researchers, we have had lots

of achievements about the above steps. There are two widely used models named hydrophobic-polar (HP) model [2] and AB off-lattice model [3].The HP-lattice model uses two types of residues to present the amino acid chains which are hydrophobic (H) or non-polar residues and polar (P) or hydrophilic residues, and the residues on the vertices of stack cubic lattices are linked sequentially by unit-length chemical bonds [4–6]. This model is simple but it neglects local interactions which are important in protein folding [7]. The AB off-lattice model is more accurate than HP-lattice model, because the bond angles are free-to-rotate and it takes torsional energy of each pair of bonds into account [8]. Though this model neglects the effect of side chains and provides only a coarse-grained approximation to the real proteins, its off-lattice construction still reflects some basic features of real proteins [9, 10]. Due to the complexity of protein folding, it's difficult to establish an accurate protein energy function. So many simplified energy functions are proposed as in [1, 11, 12], etc. And in this paper we use one of the most widely used energy function which is the same as that of [11] to predict the protein 3D structure. As for the global optimization methods, many algorithms have been proposed. Today, many algorithms have been proposed to predict protein structure using HP-lattice model, such as Multi-Self-Overlap Ensemble (MSOE), Ant Colony Optimization (ACO) [13], Multi-crossover and mutation Partial Swarm Optimization-Tabu Search algorithm (MCMPSO-TS) [14], etc. The MOSE algorithm uses a heuristic bias function to help the theoretical protein structure form a hydrophobic core, but it is only efficient for the proteins of simple folding [13]. The ACO algorithm's inspiration is from the observation of ants' behavior of searching for food, and the algorithm structure can be divided into three parts: construct ants solutions, update pheromone, and daemon actions [13]. The MCMPSO-TS algorithm is a hybrid search algorithm which combines the particle swarm optimizer (PSO) algorithm and tabu search (TS) algorithm to get better global optimization ability [14]. The used algorithms to predict protein structure with the AB off-lattice model are as follows, PSO, Tabu Search-Particle Swarm Optimization (TPSO) [15], Levy Flight Particle Swarm Optimizer (LPSO) [16], Genetic-Particle Swarm Optimization-Tabu Search algorithm (PGATS) [17], Artificial Bee Colony algorithm (ABC) [18], etc. The PSO algorithm is put forward on the basis of information transmission process of migrating birds that every bird can not only remember the best place to find the distance of the food but also know the optimal location found by the population of all the birds [15]. The TPSO algorithm combines PSO and TS to take advantage of each algorithm to improve the search precision and the ability to jump out local optimal [15]. The LPSO algorithm adds a random process named levy flight to PSO algorithm to improve algorithm's precision [16]. The PGATS algorithm is proposed on the basis of GA-PSO algorithm which is a hybrid algorithm using the strategy of updating parameter to guarantee the diversity of population in the late run of algorithm, while enhanced TS algorithm and particle mutation strategy are combined to jump out the local minimum [17]. The ABC algorithm is inspired from the behavior of bees, and the main feature of the algorithm is that it doesn't need to know the special information of the problem while it only needs to compare the merits of the solutions [18]. Since the algorithms aim to predict the structure of proteins, the most important criterion is the lowest energy value of corresponding protein energy function. And this paper focuses on getting lower energy value. We propose a hybrid optimization algorithm based on

improved BSA and TS algorithm. It's not only less sensitivity to initial values of parameters but also more efficient than the algorithms above. Experiments using AB off-lattice model shows that it can get lower energy value than nearly all the algorithms above can get.

2   3D AB Off-Lattice Model

3D AB off-lattice model was proposed by Stillinger et al. [3] on the basis of two-dimensional AB off-lattice model, which considers that the main reason for residues chain to fold into a specific spatial structure is the hydrophobicity of amino acid residues. In this model, we divide the amino acid residues into hydrophobic (A) and hydrophilic (B) two types. So, protein sequences can be simplified as sequences consisted of only A and B [19, 20], and we can predict native structure of the proteins by the relation between bond angles and bond energy of corresponding amino acids.

In the 3D space of this model, the residues of amino acid sequences are sequentially connected into the polymer with no direction keys of unit length. We'd take the bond angles between adjacent keys and the torsional angles of the two planes constituted by the adjacent three key vectors into consideration when the model is used as 3D structure of a protein [21–23]. Thus, the 3D structure of n residues is determined by $n - 2$ bond angles $\theta_1, \theta_2, \ldots, \theta_{n-2}$ and $n - 3$ torsional angles $\beta_1, \beta_2, \ldots, \beta_{n-3}$. We set the condition $-\pi \leq \theta_i, \beta_i \leq \pi$. θ_i and β_i are positive when the angles clockwise rotate. Otherwise, the angles are negative. Figure 1 is the configuration diagram of AB off-lattice model.
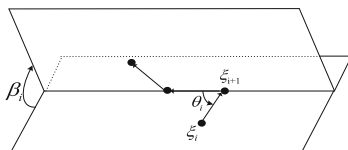
Fig. 1. Configuration diagram of AB off-lattice model

The energy function of AB off-lattice model consists of bending energy and potential energy of the Lennard-Jones type [24–27]. The energy function (E) of a sequence of n residues is expressed as Eq. (1) [28]:

$$E = \sum_{i=1}^{n-2} \frac{1}{4}(1 - \cos \theta_i) + \sum_{i=1}^{n-2} \sum_{j=i+2}^{n} 4[r_{ij}^{-12} - C(\xi_i, \xi_j)r_{ij}^{-6}] \tag{1}$$

The first part is bending potential energy which is only associated with bond angles. The second part is gravitational potential energy of any two non-adjacent residues which is not only related to polarity but also the distance of them. Here, r_{ij} represents spatial distance of non-adjacent residues, ξ_i represents the category of different residues. If the $i - th$ residue is A, $\xi_i = 1$, if it is B, $\xi_i = -1$.

Where:

$$C(\xi_i, \xi_j) = \begin{cases} +1 & \xi_i = 1, \xi_j = 1 \\ +0.5 & \xi_i = -1, \xi_j = -1 \\ -0.5 & \xi_i \neq \xi_j \end{cases} \tag{2}$$

We can see from Eq. (2) [28] that the pairs of residues AA has strong gravity, the pairs of residues BB has less gravity, and AB has weak repulsion. It reflects the true characteristic of real protein to a certain extent that hydrophobic cores form due to larger gravitational between hydrophobic residues when the protein fold into a spatial structure, at the same time, hydrophilic residues are excluded to the outside.

Therefore, the problem of protein structure prediction in the AB off-lattice model becomes the problem to find the $n - 2$ bond angles and $n - 3$ torsional angles to minimize the protein energy function E, as Eq. (3) [17]:

$$\min_{\theta_i, \beta_i \in (-\pi, \pi]} E(\theta_1, \theta_2, \ldots, \theta_{n-2}, \beta_1, \beta_2, \ldots, \beta_{n-3}) \tag{3}$$

3   BSA-TS Algorithm

3.1   BSA

Backtracking Search Optimization Algorithm (BSA) [29] was put forward by Pinar Civicioglu in 2013 for solving real-valued numerical optimization problems. Unlike many evolutionary algorithms, BSA only has one control parameter, and the initial value of the parameter has limited impact on the problem-solving performance. BSA has a simple but effective structure which enables it to easily solve multimodal problems.
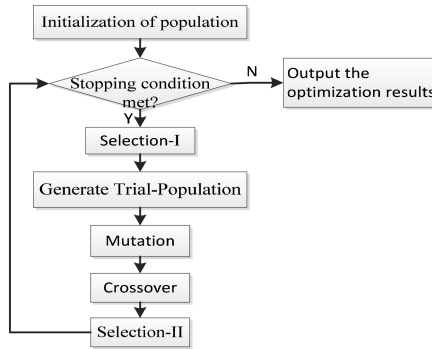
3.1.1   BSA's General Algorithm Framework
The algorithm framework of BSA is analogous to differential evolution algorithms, and it can be explained by dividing its functions into five parts: initialization of population, selection-I, mutation, crossover, and selection-II [29, 30]. Figure 2 is BSA's general algorithm framework:

It has two new crossover and mutation operators to generate a trial population, and the strategy for controlling the search-space boundaries and amplitude of the search-direction matrix enables it to have very strong exploitation capabilities.

3.1.2   Improved Strategies of BSA
(1) At the end of crossover strategy, we use two randomly choices to update the individuals beyond the search-space limits. One is the standard method that we regenerate the $j - th$ element of individual M_i beyond the search-space limits using $M_{ij} = rand * (up_j - low_j) + low_j$, where up_j means the upper bound and low_j means the lower bound of the $j - th$ component of problem; another boundary control method

Fig. 2. BSA's general algorithm framework

is that if the value of trial population is greater than the upper bound, M_{ij} will be set as up_j, and if the value of M_{ij} is less than low_j, M_{ij} will be set as low_j.

(2) Since BSA's convergence speed is slow, this paper will use Penalty strategy to replace Selection-II strategy: First, we sort the parent population P in ascending order according to the fitness value, then we take its first half as R and combine R and the trial population T obtained after the operation of crossover as a new population Mu. Secondly, the Euclidean distance d_{ij} between the individuals of Mu is calculated, and if d_{ij} is less than D (experience value, related to the length of protein sequence), the greater fitness value between Mu_i and Mu_j will be replaced by *Penalty* (experience value, this paper take 10^{13}), and then we sort the individuals of Mu in ascending order according to the fitness value and take the first *popsize* individuals to update population P.

3.2 TS Algorithm

TS algorithm was put forward by Fred Glover in 1986 [31–33] based on the local neighborhood search algorithm. Considering that the local neighborhood search is too greedy for a local search which is easy to lead to falling into local minimum, tabu strategy is used in TS to gain more search space to jump out of a local optimal solution (not completely abandoned) and the good individuals can avoid to be missed with the application of aspiration criterion.

3.2.1 Description of TS
The process of TS can be described as follows:

Step 1: Set the algorithm parameters, initialize the current solution randomly and set tabu list blank.

Step 2: Judge whether the termination criterion is satisfied; if it is, jump to step 6.

Step 3: Generate the neighborhood solutions of the current solution, and determine the candidate solutions.

Step 4: Judge whether the aspiration criterion is satisfied; if it is, set the solution as the current solution, put its fitness into tabu list, update the optimal state, and jump to step 2.

Step 5: Judge the taboo attributes of candidate solutions, set the optimal state of candidate solutions which is not in tabu list as current solution, and jump to step 2.

Step 6: Output the optimization results.

3.2.2 Improved Strategy of TS

TS algorithm is used in the second part of BSA-TS hybrid optimization algorithm so that the algorithm can jump out of a local optimal solution. In this paper, we use a mutation operator to generate the neighborhoods of current solution. On the one hand, the mutation operator can guarantee the diversity of particles in the early stage of the algorithm; on the other hand, it produces less disruption to the current solution to ensure the global convergence of the algorithm. The strategy is as follows:

Select the $k-th$ element of individual x, then perform mutation operator, the $k-th$ element of new individual x_{new} after mutation is as follows [31]:

$$x_{new}^k = x^k + 2 * \pi * f(r) * c * rate^i \tag{4}$$

Where c is a random number between 0 and 1, *rate* is scale factor, i is the current iteration the range of which is from 0 to K-1 (K is the size of neighbors). In this paper, we set *rate* as 0.95 which is the same as that in reference [31]. $f(r)$ is the correlation coefficient which is defined with Eq. (5) [17]:

$$f(r) = \begin{cases} 1, r \geq 0.5 \\ -1, r < 0.5 \end{cases} \tag{5}$$

Where r is a random number between 0 and 1.

3.3 BSA-TS Algorithm

3.3.1 Parameters

In this paper, the BSA-TS algorithm is realized through MATLAB R2012a in Windows 7 system. According to the references and the experience of experiment, the parameters are set as follows: the population size *popsize* is 200; the maximum number of iterations of the part of BSA *maxcycle* is 3000; crossover rate *mixrate* is 0.88; *Penalty* is 10^{13} (appropriate adjustments are needed according to different sequence lengths). The size of neighborhood K is 800 during the part of TS algorithm; the maximum iteration of the part of TS W is 2000; tabu length *tabulength* is 10 + N, and *rate* is 0.95.

3.3.2 Initialization

It's need to initialize the coordinates of each individual when we calculate the corresponding fitness value. The coordinates are all calculated with Eq. (6).

$$(x_i, y_i, z_i) = \begin{cases} (0,0,0) & i = 1 \\ (0,1,0) & i = 2 \\ (\cos\theta_1, \sin\theta_1, 0) & i = 3 \\ (x_{i-1} + \cos(\theta_{i-2})\cos(\beta_{i-3}), y_{i-1} + \sin(\theta_{i-2})\cos(\beta_{i-3}), z_{i-1} + \sin(\beta_{i-3})) & 4 \le i \le n \end{cases}$$
$$(6)$$

3.3.3   Description of BSA-TS Algorithm

During the BSA-TS algorithm based on AB off-lattice model, the first step is to initialize each parameter, and generate population P and population P^{old} randomly which both contain N individuals whose range is $[-\pi, \pi]$. Then the improved BSA algorithm is used to obtain the local optimal solution. At last, the result gained by improved BSA is used as the initial solution of TS algorithm in order to avoid the low efficiency of search caused by casual initialization.
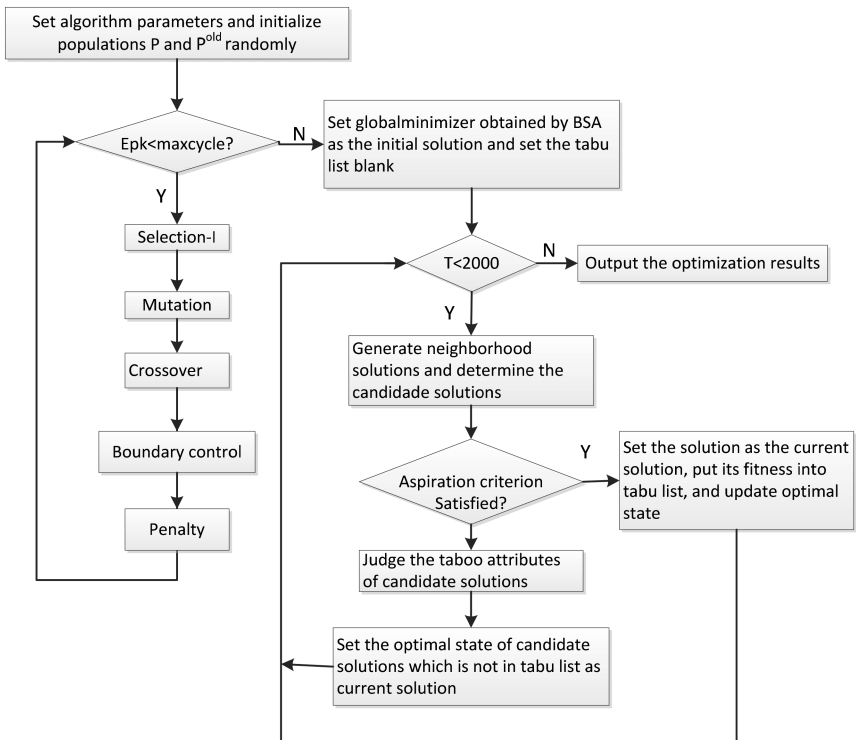
The algorithm framework of BSA-TS is as Fig. 3:

Fig. 3. Algorithm framework of BSA-TS

As we can see from Fig. 3, the process of BSA-TS can be briefly described as follows:

Step 1: Set algorithm parameters and initialize the population P and P^{old} randomly.

Step 2: Judge the termination criterion of the part of BSA that whether $epk < maxcycle$; if it does, continue execution, else jump to step 4.

Step 3: Execute the Selection-I, mutation, crossover, boundary control, and penalty sequentially.

Step 4: Set global optimal solution obtained by BSA as the initial current solution and set the tabu list blank.

Step 5: Execute the part of improved TS, and output the optimization result when the termination criterion of TS is met.

4   Experimental Results and Discussion

4.1   Prediction with Fibonacci Sequences

Fibonacci Sequences [34] usually used in the protein structure prediction are defined as follows:

$$S_0 = A, S_1 = B, \ldots, S_{i+1} = S_{i-1} * S_i \tag{7}$$

where '*' is a connection symbol, A denotes a hydrophilic amino acid and B denotes a hydrophobic amino acid.

Table 1 shows the trial Fibonacci Sequences. Table 2 shows the comparison of lowest energy values of the 3D structures of corresponding Fibonacci Sequences gained by PSO, TPSO, LPSO, PGATS, ABC and BSA-TS.

Table 1. Fibonacci Sequences

Length	Sequence
13	ABBABBABABBAB
21	BABABBABABBABBABABBAB
34	ABBABBABABBABBABABBABABBABBABABBAB
55	BABABBABABBABBABABBABABBABBABABBABBABABBABABBABBABABBBAB

We can see from the Table 2 that the lowest energy values of the Fibonacci Sequences gained by BSA-TS are lower than those obtained by PS0, TPSO, LPSO, PGATS and ABC except the sequence length of 13. The detailed description is as

Table 2. Lowest energy values of Fibonacci Sequences

Length	E(PSO [15])	E(TPSO [15])	E(LPSO [16])	E(PGATS [17])	E(ABC [18])	E(BSA-TS)
13	−1.2329	−4.7284	−4.6159	−0.6914	−0.8875	−1.3852
21	−2.7268	−8.7379	−6.6465	−7.2423	−8.1083	−9.9878
34	−4.6756	−10.7983	−7.3375	−8.9361	−10.3953	−11.3741
55	−6.9587	−13.0352	−13.0487	−18.3413	−22.4172	−23.1124

follows. When the sequence length is 13, the lowest energy value BSA-TS get is −1.3852 while the lowest energy value TPSO gets is −4.7284 and the lowest energy value LPSO gets is −4.6159 which means the 3D structure we get is less stable than TPSO and LPSO gets but more stable than the ones other algorithms can get. Additionally, when the sequence length is 21, the lowest energy value BSA-TS gets is −9.9878 while the lowest energy value other algorithms can get is −8.7379, and when the sequence length is 34 or 55, the lowest energy values BSA-TS get are −11.3741 and −23.1124 correspondingly while the lowest energy values others get are only −10.3953 and −22.4172 correspondingly. Therefore, we can conclude that the BSA-TS algorithm can effectively predict the 3D structure of protein using Fibonacci Sequences. In addition, Table 3 lists the bond angles and torsional angles of the corresponding lowest energy configurations predicted by BSA-TS algorithm.

Table 3. The bond angles and torsional angles of the Fibonacci Sequences

Length	Bond angles						Torsional angles					
13	-0.3628	0.7875	-0.7895	0.7097	-0.6639	-0.0994	2.8530	-0.3179	-2.8831	2.8440	-1.7237	0.3422
	1.2286	-0.2492	-0.1753	-0.1750	-0.1601		-2.7739	1.9743	0.4938	2.2847		
21	1.7060	-0.2291	-1.6908	-0.1045	-1.4767	-1.5979	-2.9754	0.0819	2.4444	-2.9474	2.1367	-0.5892
	0.2081	-1.1960	0.2501	-0.7140	-1.6455	-3.0239	2.9450	-2.5439	-0.6408	-2.8518	2.9076	3.0691
	1.6622	0.5601	-2.3997	0.2236	0.5165	0.0623	-1.2841	0.1008	-2.0082	2.7850	1.2461	-3.0852
	1.3040											
34	-0.4068	0.4435	-0.0085	-0.3523	-0.1423	1.1838	2.1896	1.6225	3.1086	-1.6649	3.1415	-0.8286
	0.2099	-0.5724	-0.0215	0.1309	-0.8883	-2.1602	-0.8557	-1.4109	-0.6137	-2.8643	2.5595	2.9795
	2.4730	1.4351	2.3091	0.3918	0.6735	-0.0749	1.1119	-3.0203	1.9413	-2.8573	-1.2443	0.12009
	-1.8545	0.6571	-0.3539	1.3239	-0.8056	2.6495	-1.2598	-0.8768	-0.2696	1.9029	3.1141	-0.9851
	-0.0305	-0.0302	-1.4765	0.4551	-0.1490	-0.9108	-0.4882	2.8255	-1.7022	2.8262	1.9140	1.9823
	-0.3853	-0.3279					0.1182					
55	2.2460	1.6626	0.4867	0.0342	-2.6134	0.3197	-0.7226	-3.0247	-1.4403	-3.1272	-0.3214	1.6093
	0.0549	-0.3452	1.1502	3.0766	0.7785	0.2919	-0.1444	-2.6634	0.5139	3.0125	2.3433	2.5964
	0.8690	0.4825	-0.3523	0.5528	0.0061	0.7276	-2.7821	-0.5445	-0.5445	0.8094	2.4836	0.3339
	-0.1749	1.0492	0.0944	-1.1531	0.3756	-1.4246	-2.9688	-0.9562	-2.2109	-2.0836	0.4244	2.1284
	-0.0682	-0.8392	-0.0050	0.1595	-0.4820	-0.8138	2.3701	2.8072	-1.5283	2.9689	2.5052	1.9350
	0.3366	1.8634	-1.0231	-0.6281	-0.4771	0.8827	-2.8566	-0.7047	-0.8226	0.8132	-2.8299	0.0178
	-0.7381	-0.4107	1.9268	0.5257	1.7866	0.0016	-1.4940	-0.0431	-2.1355	0.2835	-1.5955	-2.0911
	0.5750	0.8842	0.6126	0.2646	0.6930	-2.5114	-3.0494	1.3662	1.6923	0.9561	0.3247	0.0241
	-0.5513	-0.8263	0.0431	-0.0702	0.8027		-1.1321	-1.7392	-1.5901	0.2267		

Table 4. Fibonacci Sequences

Name	Real protein sequences
2KGU	GYCAEKGIRCDDIHCCTGLKCKCNASGNCVCRKK
1CRN	TTCCPSIVARSNFNVCRLPGTPEAICATYTGCIIIPGATCPGDYAN
2KAP	KEACDAWLRATGFPQYAQLYEDFLFPIDISLVKREHDFLDRAIEALCRRLNTLNKCAVMK
1PCH	AKFSAIITDKVGLHARPASVLAKEASKFSSNITIIANEKQGNLKSIMNVMAMAIKTGIEITIQADGNDADQAIQAIKQTMIDTALIQG

4.2 Prediction with Real Protein Sequences

In this section, we use the real protein sequences which are the same as those in reference [16]. These protein sequences can be downloaded from PDB database. The real protein sequences used in this paper are as follows:

In Table 4 above, A, C, G, I, L, P, M and V are hydrophobic amino acids, and D, E, F, H, K, N, Q, R, S, T, W and Y are hydrophilic amino acids.

Table 5 shows the lowest energy values of real protein sequences.

**Table 5.** Lowest energy values of real protein sequences

Length	E(PSO [15])	E(TPSO [15])	E(LPSO [16])	E(PGATS [17])	E(ABC [18])	E (BSA-TS)
2KGU	−8.3635	−21.725	−20.9633	−32.2599	−31.9480	−34.2979
1CRN	−20.1826	−53.249	−28.7591	−49.6487	−52.3249	−57.6271
2KAP	−8.0448	−	−15.9988	−28.1052	−30.3643	−36.1682
1PCH	−18.4408	−	−46.4964	−49.5729	−63.4272	−64.2034

We can see from Table 5 that the lowest energy values of the real protein sequences obtained by BSA-TS are lower than all those gained by PSO, TPSO, LPSO, PGATS and ABC. When the real protein sequence is 2KGU, 1CRN, 2KAP or 1PCH, the lowest energy values BSA-TS get are −34.2979, −57.6271, −36.1682 and −64.2034 correspondingly while the lowest energy values others get are −32.2599, −52.3249, −30.3643 and −63.4272 correspondingly. So we can conclude that our algorithm is feasible and effective to predict the 3D structure of real proteins.

Tables 2 and 5 show us that BSA-TS algorithm is efficient especially in real protein 3D structure prediction. When we predict protein 3D structure using Fibonacci Sequences, our algorithm is better-behaved as the sequence length is longer.

Table 6 lists the bond angles and torsional angles of the real protein sequences.

The following Figs. 4, 5, 6 and 7 show the 3D structures of real protein sequences predicted by BSA-TS algorithm and the real 3D protein structures downloaded from PDB database. We can conclude that the 3D structures predicted by BSA-TS algorithm are similar to the native structures of real protein sequences.
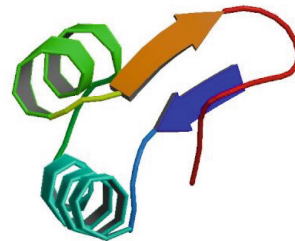

(a) 2KGU predicted by BSA-TS   (b) 2KGU in PDB
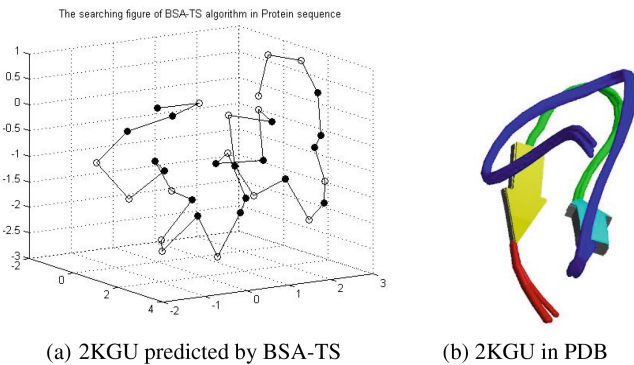
Fig. 4. Comparison of 2KGU

Table 6. The bond angles and torsional angles of the real protein sequences

Length	Bond angles						Torsional angles					
2KGU	-0.1669	-2.1412	0.5268	1.0975	0.6641	-0.6898	-0.2863	-2.4203	-0.8223	0.6329	0.3648	-0.4762
	-0.0870	-0.9542	1.2376	-0.5065	1.2986	0.1739	-2.8107	-2.2882	-2.7528	0.6952	-0.8064	1.0594
	0.9168	-0.5940	0.1924	-0.7341	-2.1755	-0.1885	3.0347	2.2126	-0.0201	3.0931	1.9193	-0.0871
	-0.2038	-1.6865	1.4663	-0.0153	1.9151	0.3536	0.2158	-1.4376	-0.4826	0.1887	-0.8352	2.9711
	-0.8108	-0.5097	-0.8404	-0.3395	-0.4083	-0.0663	2.8954	0.8762	0.4116	2.3652	2.6159	3.1284
	0.2804	-0.0695					-2.0677					
1CRN	-1.0309	1.3664	2.6891	3.1397	2.3024	-0.2724	-2.8446	-3.1320	-3.0271	1.1321	3.0215	0.1266
	3.0375	0.4026	0.7938	0.1767	-0.9387	0.5290	2.9643	3.0683	-1.6614	-0.1102	0.6941	-3.1415
	-2.5929	2.8024	1.0391	-1.3766	-1.3262	-2.1526	3.0557	2.9317	2.3983	3.0544	-3.1379	2.9053
	-1.3986	-0.9282	-0.1275	1.8422	0.4736	2.2159	1.0244	3.0738	-2.9137	-2.9375	3.0696	1.0664
	1.3804	-3.0722	-2.7121	3.1415	-1.3889	3.0151	1.3315	2.2323	-2.3403	-1.9422	-1.8556	3.0852
	-2.7292	0.7079	0.8372	2.7774	-1.0847	2.8761	-1.8929	-3.0891	-2.9064	1.8116	-2.92622	0.6080
	-1.2265	0.0101	2.9683	2.3007	1.5776	0.0646	-2.9443	-0.6404	0.7859	2.0981	1.7601	1.1664
	-0.4970	-0.2516					2.9888					
2KAP	-0.8093	-2.6397	-1.4801	2.7926	-0.9642	1.1006	2.7281	2.5670	1.2743	1.9036	3.0824	1.0521
	0.5790	-2.4767	-0.6579	-0.1268	1.5880	0.5674	2.6126	-1.1302	-1.1949	2.6947	3.1385	-2.2617
	-0.1187	-0.0890	3.1111	-1.3020	0.3872	-0.1273	-1.5645	0.2669	-0.8673	2.5987	1.5325	2.7385
	3.0466	-0.1876	-0.6100	0.6134	-0.2001	-2.2949	-0.3973	-1.7132	2.8115	-1.4590	-2.6958	-1.4388
	0.0518	0.4158	-1.0209	0.6070	-0.5793	-0.4586	2.8513	-0.5225	2.1355	2.9261	2.7558	2.1957
	-0.6472	1.5600	1.6117	-0.0652	-0.0821	1.3515	1.9057	2.8829	0.6679	-0.8247	-1.8422	3.0278
	-0.8105	0.3333	2.4782	-0.2398	1.3061	-0.2418	-2.8976	-2.1764	-0.1172	-2.8647	-1.4149	2.6605
	-2.5771	1.8776	-0.1988	-0.0387	1.1343	-1.8387	1.8947	1.5988	1.6319	1.2730	2.6464	-2.6444
	2.8144	0.8608	0.0760	-2.6360	0.1188	0.4682	-0.3847	-0.6969	-1.1354	3.1373	-2.4517	-0.9580
	-0.6844	1.6049	-1.8006	-2.8131			-3.1240	1.9528	2.4207			
1PCH	2.3837	0.1135	2.6722	1.9575	-0.5914	0.2811	1.5935	2.9751	-2.8858	1.7250	1.6557	2.7454
	-3.1011	-1.3585	2.6625	-0.7666	-0.7084	-1.7524	-2.8892	-1.1813	-0.7274	-1.7867	-2.8427	1.8842
	0.2117	-0.5654	-2.8569	2.7067	0.4575	-0.5063	-3.0660	2.1229	3.0021	0.8008	-3.1403	-2.7152
	-0.8123	-0.1471	-2.8126	2.5275	-3.1160	-2.5324	-1.6630	-2.0733	2.8908	1.3000	-3.0625	-0.9310
	0.4686	-0.4004	-0.5082	0.6098	1.8765	2.6603	-2.6517	-0.8686	-3.1177	-1.8400	-2.4446	-2.8957
	-1.5606	2.6624	1.5806	1.0635	-0.0267	3.0000	-2.8986	-3.1346	1.1236	1.6356	0.6595	1.3642
	-0.1611	1.7959	2.1088	-0.0154	2.9183	-1.1740	1.8976	-3.1415	-1.8336	3.0343	-1.8864	-1.4394
	-0.2116	-0.1058	-2.9054	3.0543	-1.1305	-0.5472	-1.8338	-1.9093	-3.1348	-2.4358	2.6950	-1.7528
	0.8016	-0.1468	0.2492	-2.1617	-1.7842	3.0632	-2.6725	1.6564	-3.0195	-2.8017	2.9032	-0.0004
	-1.5919	-2.8640	-1.4369	-0.3443	1.0223	2.9372	-2.6430	-2.4618	-0.9320	2.9721	-2.9941	-0.0785
	-0.5600	-1.0349	0.1674	-1.6850	-0.5322	-1.9706	-2.0973	-2.6544	2.7631	-1.9588	3.0461	1.2265
	-0.1369	-1.0825	0.7210	2.5296	1.5400	-1.3621	-3.0795	0.3000	3.0480	3.0234	0.9044	-2.5036
	0.2749	0.1890	0.0591	1.4680	-1.8232	2.9289	-1.2303	-1.7681	-0.9573	2.3549	2.1931	-2.9392
	3.0575	2.3969	0.9394	2.3725	1.3217	-1.5052	-3.0481	2.7907	0.4025	0.4407	-2.5874	0.1971
	0.6403	1.8016					-3.1274					
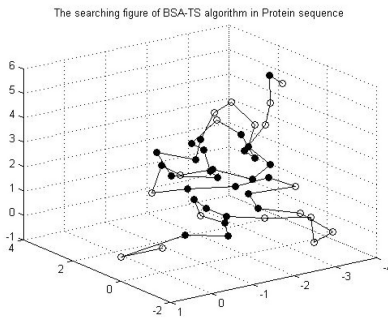
(a) 1CRN predicted by BSA-TS   (b) 1CRN in PDB

Fig. 5. Comparison of 1CRN

(a) 2KAP predicted by BSA-TS   (b) 2KAP in PDB

Fig. 6. Comparison of 2KAP

(a) 1PCH predicted by BSA-TS   (b) 1PCH in PDB
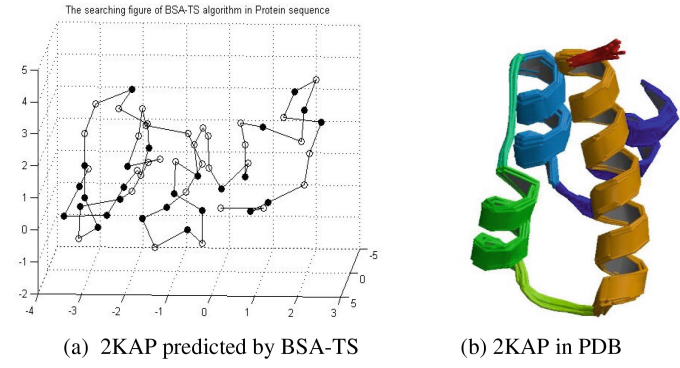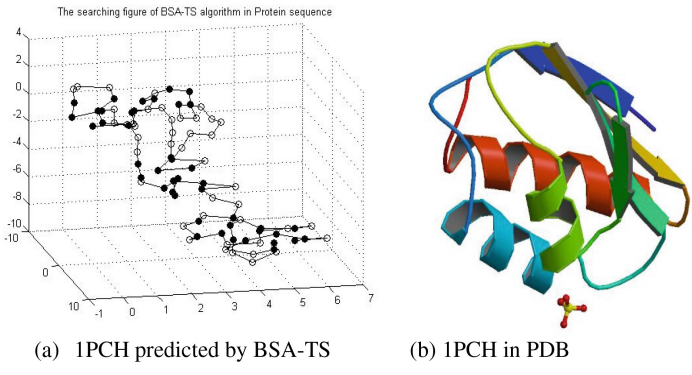
Fig. 7. Comparison of 1PCH

5   Conclusion

Due to the importance of 3D protein structure prediction, many theoretical prediction algorithms are proposed. But on account of the complexity of prediction, the present algorithms' accuracy is usually unsatisfactory. Since the most important criterion is the accuracy of algorithm, we focus on getting more stable structure with some widely used Fibonacci Sequences and real protein sequences.

This paper proposes a BSA-TS hybrid optimization algorithm which integrates the advantages of BSA that it has an efficient, fast structure, less control parameters and powerful search capability and the advantage of TS that it has strong local search capability and high precision of search. Additionally, the addition of penalty method can effectively reduce the probability of falling into local optimum caused by general BSA's Selection-II strategy. Then we use BSA-TS to predict the structure of protein based on the 3D AB off-lattice model. The experimental result shows that BSA-TS can get more stable structures which have lower energy compared with the structures gained by other algorithms. That indicates the feasibility of BSA-TS on protein structure prediction problem.

References

1. Lin, X.L., Zhang, X.L., Zhou, F.L.: Protein structure prediction with local adjust Tabu search algorithm. BMC Bioinformatics **15**(Suppl 15), S1 (2014)
2. Lin, C.J., Shih, C.S.: Protein 3D HP model folding simulation using a hybrid of genetic algorithm and particle swarm optimization. Int. J. Fuzzy Syst. **13**(2), 140–147 (2011)
3. Stillinger, F.H., Head-Gordon, T., Hirshfel, C.L.: Toy model for protein folding. Phys. Rev. **E48**, 1469–1477 (1993)
4. Custódio, F.L., Barbosa, H.J.C., Dardenne, L.E.: A multiple minima genetic algorithm for protein structure prediction. Appl. Soft Comput. **15**, 88–99 (2014)
5. Liu, J.F., Song, B.B., Yao, Y.L., Yu, X., Liu, W.J., Liu, Z.X.: Wang-Landau sampling in face-centered-cube hydrophobic-hydrophilic lattice model proteins. Phys. Rev. E **90**(3), 042715 (2014)
6. Liu, J.F., Li, G., Yu, J., Yao, Y.L.: Heuristic energy landscape paving for protein folding simulation in the three-dimensional HP lattice model. Comput. Biol. Chem. **38**(3), 17–26 (2012)
7. Irback, A., Peterson, C., Potthast, F., Sommelius, O.: Local interactions and protein folding: a three-dimensional off-lattice approach. J. Chem. Phys. **107**, 273–282 (1997)
8. Li, B., Chiong, R., Lin, M.: A balance-evolution artificial bee colony algorithm for protein structure optimization based on a three-dimensional AB off-lattice model. Comput. Biol. Chem. **54**, 1–12 (2015)
9. Liu, J.F., Sun, Y.Y., Li, G., Song, B.B., Huang, W.B.: Heuristic-based tabu search algorithm for folding two-dimensional AB off-lattice model proteins. Comput. Biol. Chem. **47**(3), 142–148 (2013)
10. Mansour, R.F.: Applying an evolutionary algorithm for protein structure prediction. Am. J. Bioinform. Res. **1**, 18–23 (2011)
11. Wang, W.H.: Ordering of unicyclic graphs with perfect matching by minimal energies. MATCH Commun. Math. Comput. Chem. **66**, 927–942 (2011)
12. Jin, X., Zhang, F.: The jones polynomial for polyhedral links. MATCH Commun. Math. Comput. Chem. **65**(2), 501–520 (2011)
13. Guo, H., Lv, Q., Wu, J.Z., Xu, H., Qian, P.: Solving 2D HP protein folding problem by parallel ant colonies. In: International Conference on Biomedical Engineering and Informatics, pp. 1–5 (2009)
14. Zhou, C.J., Hou, C.X., Zhang, Q., Wei, X.P.: Enhanced hybrid search algorithm for protein structure prediction using the 3D-HP lattice model. J. Mol. Model. **19**(9), 3883–3891 (2013)
15. Guo, H., Lan, R., Chen, X., Wang, Y.X.: Tabu search-particle swarm algorithm for protein folding prediction. Comput. Eng. Appl. **47**(24), 46–50 (2011)
16. Chen, X., Lv, M.W., Zhao, L.H., Zhang, X.D.: An improved particle swarm optimization for protein folding prediction. Int. J. Inf. Eng. Electron. Bus. **3**(1), 1–8 (2011)

17. Zhou, C.J., Hou, C.X., Wei, X.P., Zhang, Q.: Improved hybrid optimization algorithm for 3D protein structure prediction. J. Mol. Model. **20**(7), 1–12 (2014)
18. Li, Y.Z., Zhou, C.J., Zheng, X.D.: Artificial bee colony algorithm for the protein structure prediction based on the Toy model. Fundamenta Informaticae **136**(3), 241–252 (2015)
19. Mario, G.F., Eduardo, R.T., Gregorio, T.P.: Comparative analysis of different evaluation functions for protein structure prediction under the HP model. J. Comput. Sci. Technol. **28** (5), 868–889 (2013)
20. Zhang, X.L., Wang, T., Luo, H.P., Yang, J.Y., Deng, Y.P., Tang, J.S., Yang, M.Q.: 3D protein structure prediction with genetic tabu search algorithm. BMC Syst. Biol. **4**(Suppl 1), S6 (2010)
21. Nanda, D.J., Jaya, S.: Particle swarm optimization with backpacking in protein structure prediction problem. J. Mol. Model., pp. 734–738 (2012)
22. Gao, Y., Xie, S.L.: Particle swarm optimization algorithm based on simulated annealing. Comput. Eng. Appl. **40**(1), 47–50 (2004)
23. Van, D.B.F., Engelbrecht, A.P.: A study of particle swarm optimization particle trajectories. Inf. Sci. **176**(8), 937–971 (2006)
24. Bachmann, M., Arkin, H., Janke, W.: Multicanonical study of coarse-grained off-lattice models for folding heteropolymers. Phys. Rev. E **71**, 031906 (2005)
25. Kim, S.Y., Lee, S.B., Lee, J.: Structure optimization by conformational space annealing in an off-lattice protein model. Phys. Rev. E **72**, 011916 (2005)
26. Kim, J., Straub, J.E., Keyes, T.: Structure optimization and folding mechanisms of off-lattice protein models using statistical temperature molecular dynamics simulation: statistical temperature annealing. Phys. Rev. E **76**, 011913 (2007)
27. Liu, J.F.: Structure optimization by heuristic algorithm in a coarse-grained off-lattice model. Chin. Phys. B **18**, 2615–2621 (2009)
28. Li, W.Y., Wang, Y.: Multi-population genetic algorithm for three-dimensional protein structure prediction. Fujian Comput. **28**(11), 20–24 (2012)
29. Pinar, C.: Backtracking search optimization algorithm for numerical optimization problems. Appl. Math. Comput. **219**, 8121–8144 (2013)
30. Wang, X.J., Liu, S.Y., Tian, W.K.: Backtracking search optimization algorithm with high efficiency mutation scale factor and greedy crossover strategy. Comput. Appl. **34**(9), 2543–2546 (2014)
31. Zhang, X., Cheng, W.: An improved Tabu search algorithm for 3D protein folding problem. In: Ho, T.-B., Zhou, Z.-H. (eds.) PRICAI 2008. LNCS (LNAI), vol. 5351, pp. 1104–1109. Springer, Heidelberg (2008)
32. Glover, F.: Future paths for integer programming and links to artificial intelligence. Comput. Oper. Res. **13**, 533–549 (1986)
33. Glover, F., Kelly, J.P., Laguna, M.: Genetic algorithms and tabu search: hybrids for optimization. Comput. Oper. Res. **22**(1), 111–134 (1995)
34. Hsu, H.P., Mehra, V., Grassberger, P.: Structure optimization in an off-lattice protein model. Phys. Rev. E **68**(3), 1–4 (2003)