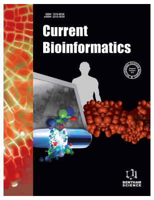
Current Bioinformatics, 2018, *13*, 182-192

RESEARCH ARTICLE

Hybrid High Exploration Particle Swarm Optimization Algorithm Improves the Prediction of the 2-Dimensional Hydrophobic-Polar Model for Protein Folding



Cheng-Hong Yang[1,2], Yu-Shiun Lin[1], Sin-Hua Moi[1], Kuo-Chuan Wu[1,3], Li-Yeh Chuang[4,*] and Hsueh-Wei Chang[5,6,7,*]

[1]Department of Electronic Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan; [2]Graduate Institute of Clinical Medicine, Kaohsiung Medical University, Kaohsiung, Taiwan; [3]Department of Computer Science and Information Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan; [4]Department of Chemical Engineering & Institute of Biotechnology and Chemical Engineering, I-Shou University, Kaohsiung, Taiwan; [5]Institute of Medical Science and Technology, National Sun Yat-sen University, Kaohsiung, Taiwan; [6]Department of Medical Research, Kaohsiung Medical University Hospital, Kaohsiung, Taiwan and [7]Department of Biomedical Science and Environmental Biology, Kaohsiung Medical University, Kaohsiung, Taiwan

ARTICLE HISTORY

Received: March 04, 2017
Revised: June 02, 2017
Accepted: September 26, 2017

DOI:
10.2174/1574893612666171006161527

Abstract: ***Background:*** Protein folding depends on the nature of the amino acid sequence. Once folding process of the amino acid sequence is successful, the protein becomes functional. Recently, a two-dimensional hydrophobic-polar (2D HP) model algorithm has been developed for the effective prediction of protein folding. However, the particular 2D HP models still lack an algorithm for protein folding prediction. Objective: Some developed algorithms still require further improvement in terms of accuracy and search stability.

Method: In order to evaluate its improvement for protein folding of the 2D HP model in this study, we propose the hybrid high exploration particle swarm optimization (HHEPSO) method, which employs the HEPSO algorithm for optimization which combines both hill climbing and greedy algorithms for local search.

Results: Several algorithms for protein structure prediction on the 2D square and triangular lattice models are compared with HHEPSO. In terms of accuracy and stability, our proposed HHEPSO revealed better performance than most of the test algorithms. HHEPSO also successfully deals with protein structure prediction problems for the longer amino acid sequences.

Conclusion: Our proposed HHEPSO algorithm is accurate and effective for protein structure prediction for a 2D triangular lattice model.

Keywords: Hybrid algorithm, HHEPSO, particle swarm optimization, hill climbing algorithm, greedy algorithm, protein folding, hydrophobic-polar (HP) model.

1. INTRODUCTION

Protein functions depend on the success of the folding process of polymerized amino acid strands [1]. Only proteins with correct folding can show normal functioning and can interact correctly with its downstream/ upstream proteins in a signaling cascade. In contrast, proteins with an incorrect folding may cause abnormal cellular functioning. Abnormal protein folding processes are associated with various diseases such as Alzheimer, Parkinson diseases. Accordingly, the correct and effective prediction of protein folding processes may provide a key for the understanding of the causes and pathways of various diseases.

To date, the 3-D protein structure is commonly resolved by x-ray crystallography or nuclear magnetic resonance spectroscopy. However, it is time and cost consuming for specific experts of protein structure and crystallography. Protein function [2] and structure [3] can computationally be predicted. The prediction of protein structure by computation may be time-effective. But simulating the folding mechanism

*Address correspondence to this author at the Department of Chemical Engineering & Institute of Biotechnology and Chemical Engineering, I-Shou University, Kaohsiung, 840 Taiwan; Department of Biomedical Science and Environmental Biology, Kaohsiung Medical University, Kaohsiung 80708, Taiwan; (LYC) Tel: +886-7-657-7711; ext. 3421; Fax: +886-7-312-5339; E-mail: chuang@isu.edu.tw; (HWC) Tel: +886-7-312-1101 ext. 2691; Fax: +886-7-312-5339; E-mail: changhw@kmu.edu.tw

2212-392X/18 $58.00+.00

of proteins without simplifying is highly complicated and time consuming because of the complex atomic structure and folding characteristics of proteins. How to simplify this complex information for protein folding by computation remains a challenge.

Many computational methods generally rely on the primary structure of a protein, *i.e.*, its amino acid sequence. In 1985, a hydrophobic-polar (HP) protein folding model [4] was proposed, which defined the protein folding mechanism by applying two- and three-dimensional (2D and 3D) lattices. This lattice protein model has been introduced as a highly simplified method for simulating the folding process of proteins [4]. The central structure of a protein comprises nonpolar amino acids that are hydrophobic (H), and the surface of a protein comprises polar (P) amino acids, being hydrophilic. Each amino acid is considered as a point, which is divided into either H or P type. In general, the HP model calculates the intensity of an adjacent hydrophobic amino acid and assigns it a negative weight. A lower weight (more negative) represents generally a more stable protein structure, which is highly similar to its tertiary structure [4].

Although the HP model has greatly simplified the protein folding information, the complexity of HP application fails to provide optimal solutions and leads to a nondeterministic polynomial-time-hard (NP-hard) problem [5]. Several optimization and random search algorithms have been applied for the evaluation of protein folding processes [6, 7]. Although several algorithms are able to provide solutions that are close to the optimal solution, the stability and prediction accuracy remains challenging. A HP model typically involves a 2D lattice model (typically square and triangular ones). However, a triangular lattice model exhibits a higher performance than a square model [8].

To further enhance the local search ability, we proposed a hybrid high exploration particle swarm optimization (HEPSO) algorithm (HHEPSO). In HHEPSO, the hill climbing and greedy algorithms were employed in the rear section after HEPSO [9]. The hill climbing algorithm possesses the characteristics of the genetic algorithm [10] which was used to exchange particle information. The greedy algorithm was chosen to improve its local optimization searching ability. The particle of HEPSO provided by the greedy algorithm is given a mutation probability in each dimension and the search result of the original prediction may be replaced if the mutation has a better search result.

In this study, we propose the HHEPSO algorithm to determine the optimal protein folding process and evaluate the accuracy and stability of the HHEPSO by comparing with other protein folding prediction algorithms.

2. METHOD

2.1. Particle Swarm Optimization (PSO) Algorithm

Kennedy and Eberhart [11] developed the PSO algorithm, which has been commonly used in recent years because of its effectiveness in addressing [6] scientific and engineering problems [12]. PSO is based on the food-seeking behavior of birds, in which a bird is represented by a particle (individual) and all particles share information that leads to an efficient food searching method for the population.

The operating steps of the PSO algorithm include initializing the population, updating *pbest* and *gbest*, and updating velocity and position. Population initialization involves randomly generating particles with possible solutions in the search space. The value of a particle experience is evaluated using a fitness function, and the value is called the fitness value. Moreover, *pbest* and *gbest* represent the previous best fitness value of a particle and global best fitness value in the population, respectively. In each iteration, the particle can update its *pbest* if the current fitness value is higher than *pbest*, and *gbest* is updated if the fitness value of a particle is higher than *gbest*. The velocity is used to adjust the particle's position in the vector space; in this space, the velocity is calculated according to the *pbest* and *gbest* vectors. Therefore, the particles can be converged toward a favorable location for determining the optimal objective in the search space.

In this study, the current position of the *i*th particle is represented as vector $X_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$ and the symbol D represents the total number of D dimensions in x_i. The $pbest_i$ value is represented as the previous best vector of the *i*th particle, denoted as $Pbest_i = (pbest_{i1}, pbest_{i2}, \ldots, pbest_{iD})$. The velocity of the *i*th particle is denoted as vector $V_i = (v_{i1}, v_{i2}, \ldots, v_{iD})$. The *gbest* value is represented using $Gbest_i = (gbest_1, gbest_2, \ldots, gbest_D)$. Equations (1) and (2) represent the updating equations of velocity and position, respectively.

$$v_{id}^{new} = w \times v_{id}^{old} + c_1 \times r_1 \times \left(pbest_{id} - x_{id}^{old} \right) + c_2 \times r_2 \times \left(gbest_d - x_{id}^{old} \right) \tag{1}$$

$$x_{id}^{new} = x_{id}^{old} + v_{id}^{new} \tag{2}$$

where both r_1 and r_2 are random decimals between zero and one; c_1 and c_2 are the learning factors, which control the distance that a particle covers in a generation, and in each move, c_1 and c_2 influence the cognitive and social behavior of the particle, respectively; v_{id}^{new} and v_{id}^{old} are the new and old velocities of the *i*th particle, respectively; x_{id}^{new} and x_{id}^{old} are the current and renewed positions of the *i*th particle, respectively; and w is the inertia weight that controls the effect of the *i*th particle on its current velocity.

2.2. HEPSO Algorithm

The HEPSO [9] is an improved approach that contains a mechanism as an operator. In order to improve the converging process and escape from local minima, Mahmoodabadi *et al.* [9] proposed that 1) the multi-crossover genetic algorithm (GA) [13] that uses the *gbest* as the premier parent and *pbest* as the second parent to generate the new velocity; 2) the food source finding operator of the artificial bee colony (ABC) [14] algorithm is used for the selected particles in the PSO strategy. In the current study, GA [13] and ABC [14] algorithms are integrated into the HEPSO [9] algorithm to enhance its search ability. The HEPSO algorithm applies an ABC algorithm in its initial iterations to overcome a problem encountered when the search value drops to a level lower than the local optimal value. GA is used for enhancing the search ability in the final

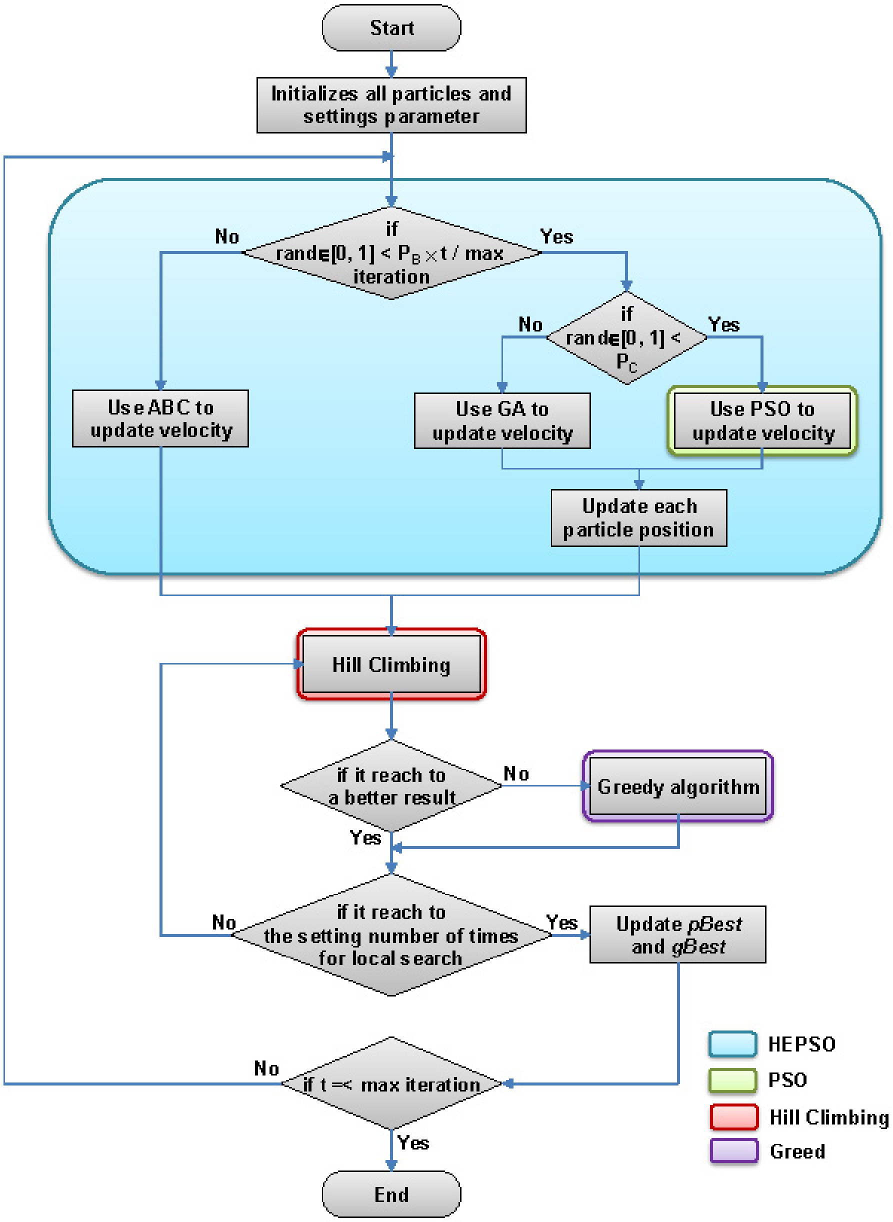

Start
Initializes all particles and settings parameter
if rand∈[0, 1] < P_B × t / max iteration
No
Yes
if rand∈[0, 1] < P_C
No
Yes
Use ABC to update velocity
Use GA to update velocity
Use PSO to update velocity
Update each particle position
Hill Climbing
if it reach to a better result
No
Greedy algorithm
Yes
if it reach to the setting number of times for local search
No
Yes
Update *pBest* and *gBest*
if t =< max iteration
No
Yes
End
HEPSO
PSO
Hill Climbing
Greed

Fig. (1). HHEPSO flow chart.

few iterations. Pseudocodes of our proposed algorithm are provided as shown in the Supplementary material.

2.3. Hill Climbing

The hill climbing algorithm [15] is an effective local search approach. This algorithm first determines the maximum or minimum fitness value. If a higher fitness value is determined, a new search set is conducted instead of the original search set.

2.4. Greedy Algorithms

The greedy algorithm [16] determines the best fitness value in each step. It can be used to solve the spanning tree and coin-changing problems. However, optimization algorithms cannot determine the optimal solution in most situations. Moreover, they may provide solutions that are inferior to normal solutions. We can obtain solutions for optimization problems by solving each of them individually. Both the hill climbing and greedy algorithms were alternately used to enhance the search performance.

2.5. Hybrid HEPSO (HHEPSO)

In the current study, two algorithms included hill climbing [15] and greedy algorithms [16] were hybridized after performing the HEPSO [9]. Fig. (**1**) illustrates the flowchart of the HHEPSO algorithm which integrates the hill climbing and greedy algorithms. As shown in this figure, P_B is the probability of using the ABC algorithm and P_c is the probability of using the GA. The HEPSO algorithm applies a decision function to derive the velocity updating function (ABC or GA). Next, it applies the hill climbing algorithm to

Table 1. Particle information of amino acid point directions.

Amino Acid Point	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
Model	**Movement Direction Toward the Next Point*[1]**										
Square lattice	-	2	1	1	4	4	4	3	2	2	-
Triangular lattice	-	6	1	2	3	3	3	5	6	6	-
HP characters*[2]	H	H	P	P	H	P	P	H	P	P	H

*[1] The number is the movement direction toward the next amino acid point as indicated in Fig. (**2**). For example, the number listed in point 2 indicates the direction from point 2 toward point 3.
- indicates the starting and end points.
*[2] The HP characters for each point are hypothesized for the prediction purpose for Fig. (**3**).

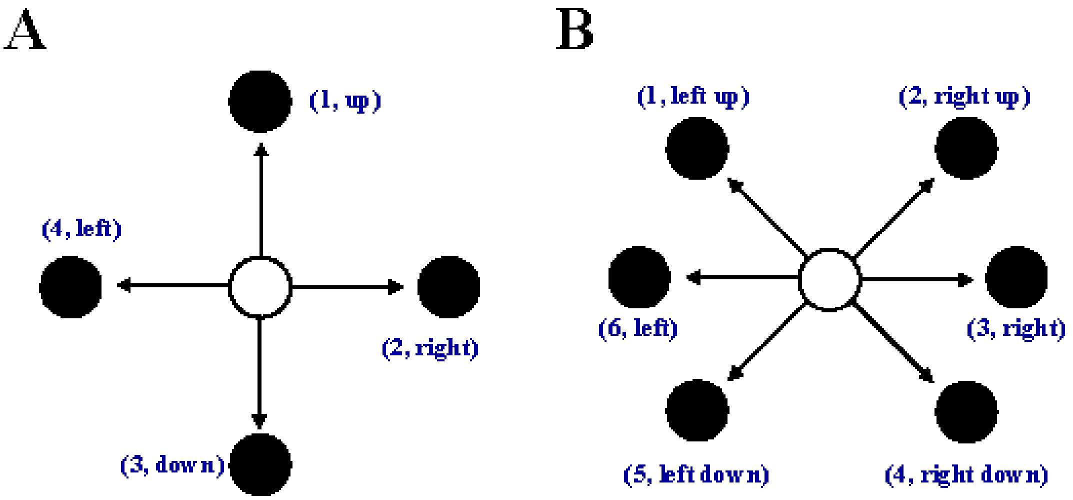


Fig. (2). Dataset encoding for protein folding. (**A**) The encoding for square model. (**B**) The encoding for triangular model. Central white circle indicates the starting point of amino acid. Black circles indicate the possible movement of the next amino acid. Different movements for protein folding are indicated by different numbers and directions for the mathematical operation by computer.

conduct a local search. If the search result is inferior to that of hill climbing algorithm conducted earlier, the greedy algorithm is applied for searching particles. The hill climbing algorithm stops the search process when the number of iterations is reached. The *pbest* and *gbest* values are updated with new values. Subsequently, the HEPSO algorithm moves to the next iteration. If the number of iterations is equal to the parameter set, the HHEPSO algorithm stops searching and visualizes the results. The flow chart for the hybrid HEPSO (HHEPSO) is displayed in Fig. (**1**). The detailed information for these algorithms is as follows:

2.5.1. Encoding

Protein folding involves folding the base unit (amino acid) to form a functional protein. Each amino acid in a data set is considered as a point, and each point has a folding direction that determines the protein folding mechanism. Figs. (**2A**) and (**2B**) respectively illustrate the square and triangular lattice protein models. The example of particle initialization based on these models is shown below.

2.5.2. Particle Initialization

In this study, each particle dimension was denoted as an amino acid. The first amino acid point is always at the center in a 2D space. The data for each dimension are denoted as the folding direction toward the next amino acid point. Table **1** presents the particle information of an amino acid sequence with 11 points. In fact, there are several possible folding processes. Table **1** is provided as an example to demonstrate how the protein folding can be predicted and recorded by the algorithm.

Since the direction of the first point toward the second point do not influence the protein folding, its movement direction toward the next point is not recorded and marked with "-" (Table **1**). Moreover, the last point has no next point to move and it is also marked with "-". Without the direction of the head and end points, the particle dimension number listed in Table **1** is equal to 9, *i.e.*, the total length of the amino acid minus 2.

Each particle is generated using a random number according to the encoding for direction labels, *i.e.*, within the range of 1-4 for the square model and 1-6 for the triangular model as mentioned in Fig. (**2**). The visualization of the protein folding process and outcome is shown in Fig. (**3**). For example, the second point labels with 2 in the square model (Table **1**). It means that the second point moves toward the third point in (2, right; Fig. (**2**)) direction as the visualization in Fig. (**3A**). The third point labels with 1 in the square

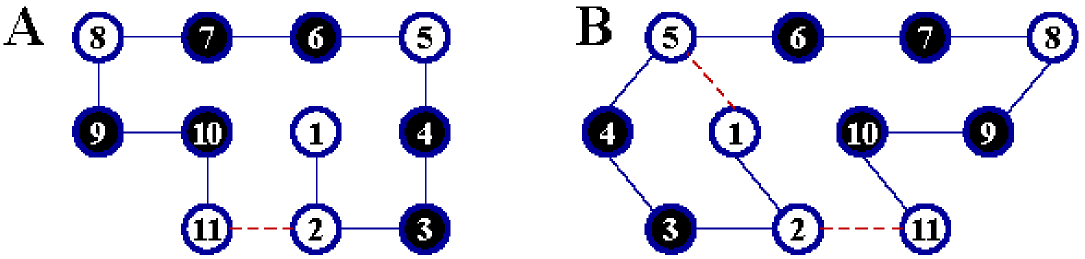


Fig. (3). Data visualization of protein folding prediction. (**A**) Square model. (**B**) Triangular model. Data set and folding information are compared in Table **1** and the folding method is indicated in Fig. (**2**). Black circle indicates the polar amino acid. White circle indicates the hydrophobic amino acid. The numbers indicate the order of amino acids. A dashed line indicates the hydrophobic interaction and is regarded as the high intensity between connecting amino acids.

model (Table **1**). It means that the third point moves toward the fourth point in (1, up; Fig. (**2**)) direction as the visualization in Fig. (**3A**). Similarly, the triangular model (Table 1) follows the movement as indicated in Fig. (**2B**) and is visualized in Fig. (**3B**).

Moreover, more hydrophobic interactions, *i.e.*, H point connects to other H point (marked with dashed line in Fig. (**3**)), are regarded as more stable for protein folding and become more correct to natural protein structure.

2.5.3. Velocity Update and Position Update

In this study, HEPSO used three methods to update the particle. The first method includes the food source finding operator of the ABC algorithm [9] to update the particle, the operator is shown as an equation (3).

$$x_{id}^{new} = x_{id}^{old} + (2r - 1) \times (x_{id}^{old} - x_{jd}^{old}) \qquad (3)$$

where *r* is a random number from 0 to 1, *d* and *i* are the dimension and current particle, and *j* is a random number from 1 to the population size, but *j* cannot be equal to *i*.

The second method is updating the function from the original PSO that is mentioned above as equations (1) and (2). In here, the parameters of origin PSO were modified. The weight *w* of PSO affects the search ability, a large weight speeds a global search and small weight helps a local search. Therefore, we refer to the literature [9] that the weight is dynamically adjusted between 0.4 and 0.9 as shown in equation (4).

$$w(f) = \frac{1}{1+1.5e^{-2.6f}} \in [0.4,\ \ 0.9] \qquad (4)$$

where

$$f = \frac{d_g - d_{min}}{d_{max} - d_{min}} \in [0, 1] \qquad (5)$$

where *f* is the evolutionary factor as shown in equation (5). d_i is the average distance of particle *i* to the all particles present, using Euclidian metrics as shown in equation (6). d_g is the best fitness value of a particle from d_i. d_{min} and d_{max} are determined by comparing all d_i's with their maximum and minimum distances.

$$d_i = \frac{1}{N-1}\sum_{j=1,j\neq i}^{N}\sqrt{\sum_{k=1}^{D}(x_i^k - x_j^k)^2} \qquad (6)$$

where *N* is the particle population, *i* is the current particle, *k* is the dimension of the particle.

In the learning factors c_1 and c_2, with a large value of c_1 and small value of c_2, particles trend to move around *pbest*, contrary to the particles trend to move around *gbest*. The learning factors are expressed as equations (7) and (8):

$$c_1 = c_{1i} - \left(c_{1i} - c_{1f}\right) \times \left(\frac{t}{\text{max iteration}}\right) \qquad (7)$$

$$c_2 = c_{2i} - \left(c_{2i} - c_{2f}\right) \times \left(\frac{t}{\text{max iteration}}\right) \qquad (8)$$

where *i* and *f* are the initial and end values of the learning vector and *t* is the current iteration. Learning vector c_1 decreases as the number of iterations increases. By contrast, learning vector c_2 increases with the number of iterations decreases.

Therefore, the algorithm can focus on global search during the initial iterations and on local search during the final iterations.

Finally, the concept of GA is used to integrate velocity in the calculation, as shown in equation (9).

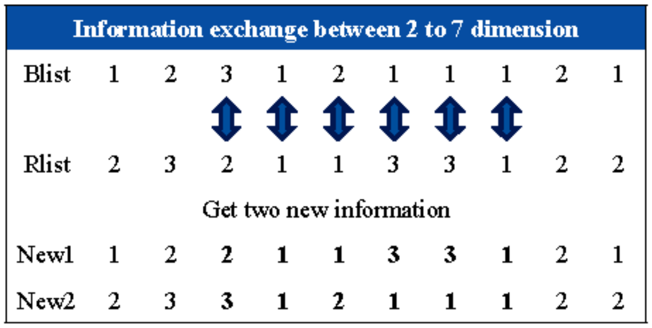
$$v_{id}^{new} = r\times\left(\frac{c_2}{2}\right) \times gbest_d - pbest_{id} - x_{id}^{old} \qquad (9)$$

where *r* is a random number ranging from 0 to 1 and c_2 is obtained from equation (8). The HEPSO algorithm applies equation (2) to update the position of particles after calculating their new velocities.

2.5.4. Local Search and Fitness Function

An elite particle set, "Blist," comprises the top five particles with the best fitness values. The local search selects one particle randomly to be the "Rlist." Subsequently, each member of Blist exchanges information with members of Rlist. Fig. (**4**) illustrates this process.

As shown in Fig. (**4**), we generated the start and stop points randomly in every exchange process. The Blist and Rlist sets have two new particles that exchange information between the start and stop points. In the greedy algorithm, a mutation is performed in every possible particle dimension. When the mutation is complete, a folding process is attempted in every direction of the selected mutation dimension to obtain the optimal result.



Information exchange between 2 to 7 dimension										
Blist	1	2	3	1	2	1	1	1	2	1
			⇕	⇕	⇕	⇕	⇕	⇕		
Rlist	2	3	2	1	1	3	3	1	2	2
Get two new information										
New1	1	2	**2**	**1**	**1**	**3**	**3**	**1**	2	1
New2	2	3	**3**	**1**	**2**	**1**	**1**	**1**	2	2

Fig. (4). Information exchange. Blist and Rlist are two different particles. When they exchange information from dimension 2 to dimension 7, two new results (new 1 and new 2) are generated for the sources for further simulation in protein folding prediction. Numbers in bold number indicate the outcome for information exchange.

Fig. (**5**) illustrates the new particle information obtained through the hill climbing (exchanging) and greedy algorithms (mutation). The star mark at the fourth amino acid point denotes the mutation position. The gray points are marked as type H, and the orange points are marked as type P. The numbers at the center of the points indicate that the current point is the *N*th amino acid point in the sequence. Each amino acid point in the sequence is connected using black lines beginning from the first amino acid point. The fitness values are relevant among the H type amino acid points without the black line connection. Dotted lines connect the relevant H type amino acid points. Each dotted line is added to the fitness value. The graphs representing the right folding and right up folding methods have only one dotted line. By contrast, the graph representing the left up folding method has three dotted lines. The hill climbing and greedy algorithms exchange the original folding mechanisms and search for a more optimal solution. The new solution replaces the old solution from the Blist set, and the particles are returned to the Blist set.

2.5.5. Updating Gbest and Pbest

The optimal fitness value of each particle in the past is *pbest*. The optimal *pbest* value is called *gbest*. The search result of each particle in every iteration is compared with *pbest* and *gbest*. If the new result is superior to the original result, *pbest* is replaced, and *gbest* is updated when *pbest* updating is complete.

3. EXPERIMENTAL RESULTS

3.1. Data Set

In this study, the protein folding simulation data sets [17] include eight amino acid sequences (Table **2**) ranging from 20 to 64 amino acids for simulation and the best results of all folding possibilities was reported [17, 18]. The data sets are used for determining the prediction accuracy and stability of the 2D square (sqr) and triangular (tra) lattice model prediction algorithms. Each test set was run for 30

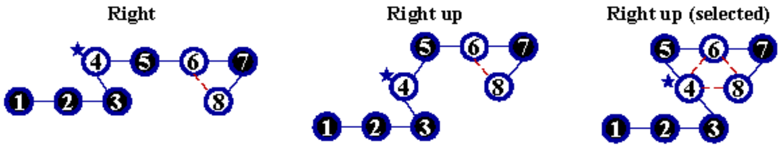


Fig. (5). Visualization of the greedy algorithm in protein folding prediction. In this example, the fourth point (marked with *) is mutated and all possible foldings are displayed such as the movement direction for right, right up, and left up. The dashed line indicates the hydrophobic interaction and is regarded as the high intensity between connecting amino acids. Among them, the left up prediction is selected because it generates more hydrophobic interactions and is regarded as the best folding result.

Table 2. Amino acid sequences of the test dataset.

Seq	Len	Amino Acids Sequence	E*(sqr)	E*(tra)
1	20	$(HP)^2PH(HP)^2(PH)^2HP(PH)^2$	-9	-15
2	24	$H^2P^2(HP^2)^6H^2$	-9	-17
3	25	$P^2HP^2(H^2P^4)^3H^2$	-8	-12
4	36	$P(P^2H^2)^2P^5H^5(H^2P^2)^2P^2H(HP^2)^2$	-14	-24
5	48	$P^2H(PH^3)^2P^5H^{10}P^6(H^2P^2)^2HP^2H^5$	-23	-43
6	50	$H^2(PH)^3PH^4PH(P^3H)^2P^4(HP^3)^2HPH^4(PH)^3PH^2$	-21	-41
7	60	$P(PH^3)^2H^5P^3H^{10}PHP^3H^{12}P^4H^6PH^2PHP$	-36	nd
8	64	$H^{12}(PH)^2((P^2H^2)^2P^2H)^3(PH)^2H^{11}$	-42	nd

Seq, Sequence; Len, length; H, hydrophobic; P, polar; H3, HHH; HP^2, HPHP; PH^2, PHPH; Sqr, square lattice model; Tra, triangular lattice model.
* The negative value indicates the number of hydrophobic interactions. Increasingly negative means that the predicted protein folding is more stable with more hydrophobic interactions. Using the model, the best results of all folding possibilities was reported [17, 18]. "nd" indicates no determination according to the existing literature [18].

experiments. The protein sequence is provided with P or H property in a row and when amino acids with the same property are connected this is shown repeatedly. For example: $(HP)^2$ = HPHP; the length of the first sequence is 20 and that of the second sequence is 24. The negative values of E(sqr) and E(tra) indicate the number of hydrophobic interactions. Under the same sequence, more negative means that the predicted protein folding is more stable with more hydrophobic interactions. For example of Seq 1, the E(sqr) and E(tra) values of the first sequence is -9 and -15, respectively. Accordingly, the predicted protein folding of E(tra) is more stable than that of E(sqr).

3.2. Parameter Settings

The values of P_B and P_C are set to 0.05 and 0.95, respectively. The particle population is set to 100, and the number of iterations is set to 200. The initial value of the learning factor C_{1i} is set to 2.5 and that of C_{2i} is set to 0.5. The end value of the learning factor C_{1f} is set to 0.5 and that of C_{2f} is set to 2.5. The parameters of HEPSO refer to Mahmoodabadi *et al.* [9]. The integrated local search algorithm (hill climbing and greedy algorithms) executes 64 iterations per PSO iteration. The initial velocity values are randomly generated from V_{min} to V_{max}. The setting of V_{min} and V_{max} are from diverse directions of protein folding model: square lattice and triangular lattice. The direction labels of the square lattice model are 1-4 for diverse directions. The direction labels of the triangular lattice model are 1–6 for various directions. The V_{max} and V_{min} values depend on different lattice models. The V_{max} value of the square lattice model is +3, and the V_{min} value is -3. The V_{max} value of the triangular lattice model is +5, and the V_{min} value is -5. The maximum weight W_{max} is 0.9 and the minimum weight W_{min} is 0.4.

3.3. Comparison Between the Prediction Accuracies of HHEPSO and Other Algorithms Under the Square Lattice Model

Table **3** shows the comparison between the prediction accuracies of the square lattice model, genetic algorithms (GA) [19], multimeme memetic algorithm (MMA) [20], differential evolutionary (DE) algorithm [6], and hybrid DE [21], indicating that the accuracy and search stability of these algorithms do not vary considerably in terms of the square lattice model. The experimental results show, that these algorithms can easily obtain best results in sequence 1 to 7 (*i.e.* -9 to -21). Other than sequence 7 and 8 are unable to find the best model but not bad results in our proposed method, *i.e.*, -35 (E = -36) and -39 (E = -42), respectively. Consequently, the accuracy of the square lattice folding model displays approximately the optimal solutions among these algorithms. It also validates that our proposed HHEPSO algorithm has a moderate ability for predicting protein folding in terms of square lattice model. Fig. (**6A**) illustrates a visualization of the square lattice search result. This graph shows that H points normally fold at the center of the protein, and P points fold on the outside around the H points. Therefore, the shape of the predicted protein folding is stable because it matches the functional protein that is hydrophobic inside and hydrophilic outside.

3.4. Comparison between the Prediction Accuracies of HHEPSO and Other Algorithms Under the Triangular Lattice Model

The prediction accuracy of the square lattice folding model may be further improved by another lattice model such as the triangular one. Because the triangular lattice prediction is complicated, we compare several algorithms not only the prediction accuracies results but also the prediction stability.

Table **4** shows a comparison between the prediction accuracies of the triangular lattice model for simple genetic algorithm (SGA), hybrid genetic algorithm (HGA) [22], tabu search (TS) [23], elite-based reproduction strategy-genetic algorithm (ERS-GA), hybrid of hill-climbing and genetic algorithm (HHGA) [7], and HHEPSO. The results show that HHEPSO prediction accuracies were -15, -17, -12, -24, -40 and -41 in sequence 1 to sequence 6 that obtain the best solutions but expect for the sequence 5. In sequence 7 and sequence 8, the HHEPSO obtained the best results, -70 and -73, respectively. In conclusion, most of the algorithms exhibit high performance in searching for short amino acid sequences, except for SGA. Except for the sequence 5, our proposed HHEPSO has the same ability for predicting accuracies with respect to the triangular lattice model. For

Table 3. Comparison between the prediction accuracies of the square lattice model and other algorithms*[1].

	Sequence (Amino Acid Lengths)*[2]							
Method	**1 (20)**	**2 (24)**	**3 (25)**	**4 (36)**	**5 (48)**	**6 (50)**	**7 (60)**	**8 (64)**
GA [19]	**-9**	**-9**	**-8**	**-14**	-22	**-21**	-34	-37
MMA [20]	**-9**	nd	**-8**	**-14**	-22	**-21**	nd	-39
DE [6]	**-9**	**-9**	**-8**	**-14**	**-23**	**-21**	**-35**	-39
Hybrid DE [21]	**-9**	**-9**	**-8**	**-14**	**-23**	**-21**	**-35**	**-42**
HHEPSO	**-9**	**-9**	**-8**	**-14**	**-23**	**-21**	**-35**	-39
E*[3]	-9	-9	-8	-14	-23	-21	-36	-42

*[1] The HP model defines the hydrophobic (H-H) interaction by assigning a negative (favorable) weight to interactions between adjacent, non-covalently bound H [4]. More negative means protein has a more stable structure.
*[2] The lengths of amino acid sequences are listed in Table **2**.
*[3] This provides the best result among all possibilities [21].
Numbers in bold denote the optimal solutions among all tested algorithms. "nd" indicates that no determination is provided from the literature.

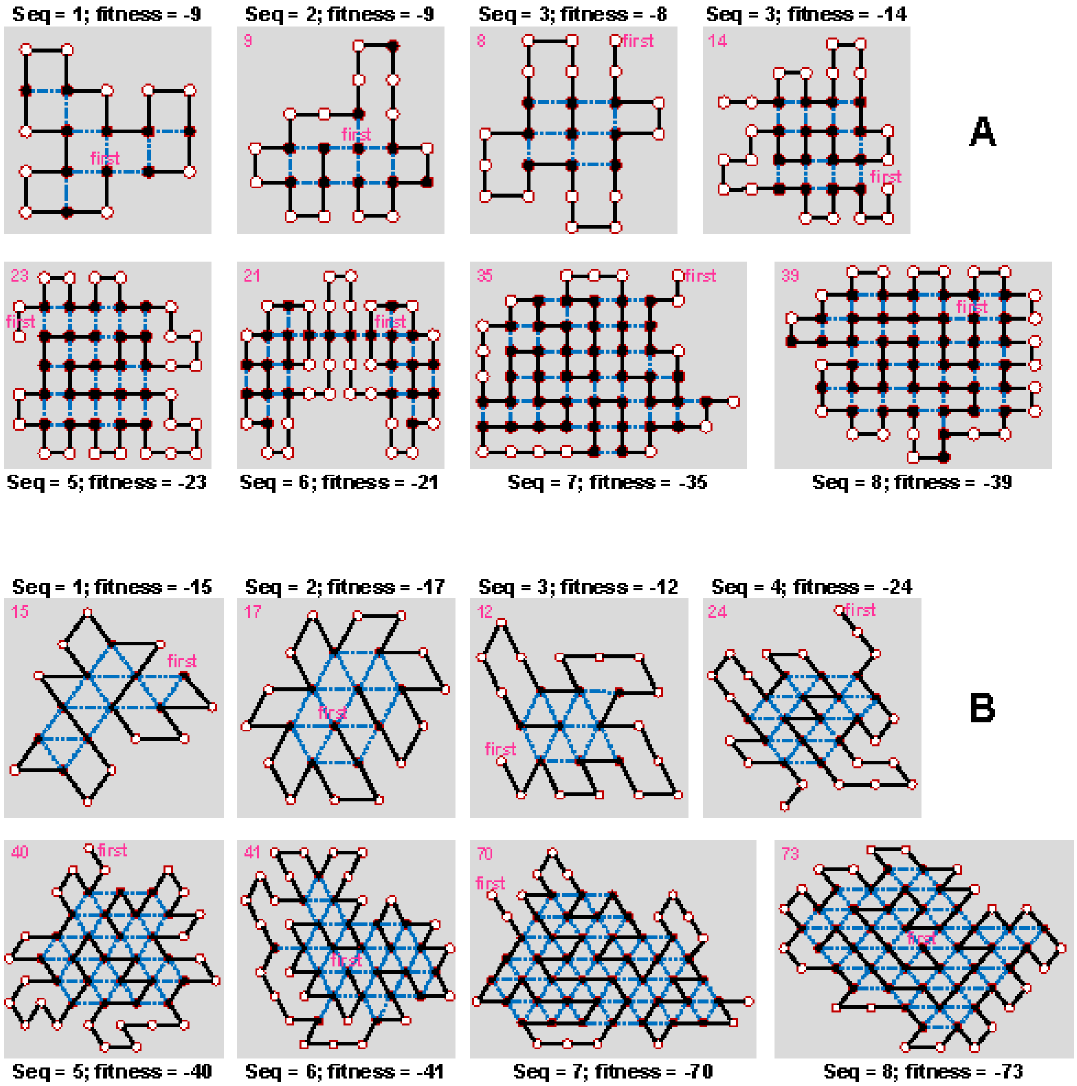


Fig. (6). Visualization of the prediction result of the square and triangular lattice models. (**A**) Square lattice models. (**B**) Triangular lattice models. Black and white points are hydrophobic and polar, respectively. The letter "first" indicates the beginning of the simulated amino acid sequence. The number in the left-upper corner indicates the intensity of simulated results. The solid line indicates the connection between the amino acids in the order of the test sequences. The dashed line indicates the hydrophobic interaction between two hydrophobic amino acid residues. This visualization is the best result among 30 simulations.

sequence 5, HHEPSO also listed on the top-two ranks among these algorithms. These results also validated that our proposed HHEPSO algorithm has a better performance for predicting protein folding with respect to the triangular lattice model. Fig. (**6B**) shows a visualization of the 2D triangular lattice search results. Therefore, the shape of the predicted protein folding is stable because it matches with the functional protein that is hydrophobic inside and hydrophilic outside. It also suggests that our proposed HHEPSO performs well in predicting protein folding.

3.5. Comparison between the Prediction Stability of HHEPSO and Other Algorithms Under the Triangular Lattice Model

The prediction stability of those non-HHEPSO algorithms was not provided under the square lattice model; therefore, we cannot compare the prediction stability of HHEPSO with non-HHEPSO in terms of the square lattice model.

Table **5** shows the prediction stability comparison of the test algorithms with respect to the triangular lattice model. The optimal results of 30 experiments are averaged for comparison. The experimental results show that HHEPSO, respectively, obtain the average values of -14.93, -15.43, -11.97, -22.93, -38.03, -37.03, -65.60, and -69.79 in sequence 1 to sequence 8 that are better than the ERS-GA and HHGA. In conclusion, when the average value is higher, the stability of predicting protein structure is higher and the result becomes satisfactory. Except the optimal solution of sequence 5, the performances for all sequences in HHEPSO are better than that of other algorithms.

Table 4. Comparison between the prediction accuracies of the triangular lattice model and other algorithms*[1].

	Sequence (Amino Acid Lengths)*[2]							
Method	**1 (20)**	**2 (24)**	**3 (25)**	**4 (36)**	**5 (48)**	**6 (50)**	**7 (60)**	**8 (64)**
SGA [22]	-11	-10	-10	-16	-26	-21	-40	-33
HGA [22]	**-15**	-13	-10	-19	-32	-23	-46	-46
TS [23]	**-15**	**-17**	**-12**	**-24**	-40	nd	**-70**	-50
ERS-GA [7]	**-15**	-13	**-12**	-20	-32	-30	-55	-47
HHGA [7]	**-15**	**-17**	**-12**	-23	**-41**	-38	-66	-63
HHEPSO	**-15**	**-17**	**-12**	**-24**	-40	**-41**	**-70**	**-73**
E*[3]	-15	-17	-12	-24	-43	-41	nd*[4]	nd*[4]

*[1] The HP model defines the hydrophobic (H-H) interaction by assigning a negative (favorable) weight to interactions between adjacent, non-covalently bound H [4]. Increasingly negative means that proteins have a more stable structure.
*[2] The lengths of the amino acid sequences are listed in Table **2**.
*[3] E is the best result among all possibilities [18].
*[4] "nd" indicates no determination is available from the literature [18].
Numbers in bold denote the optimal solutions among all of the test algorithms. "nd" indicates no determination is available from the literature.

Table 5. Comparison of the predicted stability of the triangular lattice model and other algorithms.

	Sequence (Length of Amino Acids)*[1]							
Method*[2]	**1 (20)**	**2 (24)**	**3 (25)**	**4 (36)**	**5 (48)**	**6 (50)**	**7 (60)**	**8 (64)**
ERS-GA [7]	**-15**/-12.5	-13/-10.20	**-12**/-8.47	-20/-16.17	-32/-28.13	-30/-25.30	-55/-49.43	-47/-42.37
HHGA [7]	-15/-14.73	**-17**/-14.93	**-12**/-11.57	-23/-21.27	**-41**/-37.30	-38/-34.10	-66/-61.83	-63/-56.53
HHEPSO	**-15/-14.93**	**-17/-15.43**	**-12/-11.97**	**-24/-22.93**	-40/**-38.03**	**-41/-37.03**	**-70/-65.6**	**-73/-69.79**

*[1] The lengths of amino acid sequences are listed in Table **2**.
*[2] Values indicate the performance for optimal solutions/the average solutions after 30 time simulation.
Numbers in bold denote the optimal solutions among all test algorithms.

3.6. Time Complexity Analysis

The computational complexity of the protein folding problem [24] is O(n^2), where n is the number of amino acid. In the PSO, the computation complexity is described as P×I×N, *i.e.*, O(PIn^2), where P is population size of particles, I is the number of iterations and N is the number of computational complexity of protein folding problems for each particle. In the HEPSO, the computational complexity has the same PSO origin, because the GA and ABC algorithm were integrated into the PSO as computational operators. The computational complexity of Hill climbing is O(n^2) and that of the Greedy algorithm is also O(n^2). Therefore, the time complexity of our proposed method HHEPSO is O(n^2) + O(n^2) + O(n^2).

4. DISCUSSION

A single algorithm can not completely solve the problem of protein structure prediction. The local optimum problem is the most common issue here and is studied widely. Many local search algorithms were proposed for escaping local optimum traps. For this the concept of hybrid algorithms were used especially, such as HEPSO [9] and PSOBBO [25]. Hence, we proposed to combine characteristics of several methods to enhance the search ability. Consequently, the experimental results show that our proposed method can provide a better solution to this problem. Several studies have tried hybrid techniques to generate the synergy effect of different algorithms for the improvement of protein structure prediction [26-28]. Accordingly, optimization algorithm (HEPSO) combined with local search algorithms (hill climbing and greedy) may increase the accuracy of protein structure prediction. For example, a hybrid of hill-climbing and genetic algorithm (HHGA) has been developed for 2D triangular protein structure prediction [7] and its performance is better than that of GA alone. Similarly, in this study we proposed a hybrid HEPSO algorithm which consists of the HEPSO algorithm for executing a global search and the hill climbing and greedy algorithms for enhancing the local search performance of the algorithm for predicting a protein conformation. The performance comparison between HHGA and HHEPSO is listed in Table **4** and its discussion is mentioned later.

Algorithms for predicting structure with long amino acid sequences often show a poor performance compared to short amino acid sequences [29, 30]. Similarly, most algorithms such as SGA, HGA [22], TS [23], ERS-GA, and HHGA [7] exhibit high performance for short sequences but display low performance for long sequences (≥ 60 amino acids) in terms of the triangular model prediction (Table **4**). In contrast, our proposed HHEPSO algorithm may overcome this problem. The HHEPSO algorithm is designed to integrate with the local search algorithms such as hill climbing and greedy algorithms and exhibit high performance for both short and long sequences in terms of the triangular model.

Moreover, different lattice models seem to have different performances. For example, the search performance of the square lattice model generally dropped to a level below the local optimal values due to the overfitting natures of the hill climbing and greedy algorithms. In contrast, the performance of each test sequence based on triangular lattice model (Table **4**) is better than that of square lattice model (Table **3**), suggesting that the protein structure prediction by the triangular lattice folding way is better than the square lattice.

In general, protein structure prediction for a long sequence is time-consuming but the prediction results are not guaranteed to become better. Search stability of the chosen algorithm is very important to the performance of protein prediction. As shown in Table **5**, our proposed HHEPSO shows higher search stability for protein prediction in terms of triangular model than the other tested algorithms. In the hybrid algorithms (ERS-GA and HHGA) [7], the GA is chosen to search local optimal solution. In contrast, HHEPSO is an improved algorithm of HEPSO [9], which originally developed to combine the GA and artificial bee colony algorithms. In HHEPSO, the greedy algorithm is further added to the HEPSO and found to be more effective in protein prediction.

CONCLUSION

The current study proposes a hybrid high exploration particle swarm optimization (HHEPSO) method to improve the HEPSO algorithm with the integration of local search algorithms (hill climbing and greedy algorithms) for the prediction of protein folding structure. For test algorithms and our proposed HHEPSO algorithm, the prediction accuracies of the triangular lattice model are performed better than that of square lattice model. Compared with other algorithms, the HHEPSO algorithm exhibits superior performance in predicting protein folding of long amino acid sequences in either square or triangular lattice models. Because the local search algorithms are repeated for each PSO iteration, the local search efficiency for short and long sequences is well processing by HHEPSO. In conclusion, our proposed HHEPSO algorithm is accurate and effective for protein structure prediction in 2D triangular lattice model.

CONSENT FOR PUBLICATION

Not applicable.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

This work was supported by funds of the Ministry of Science and Technology, Taiwan (MOST 102-2221-E-151-024-MY3 and MOST 104-2320-B-037-013-MY3), the National Sun Yat-sen University-KMU Joint Research Project (#NSYSU-KMU 106-p001). The authors thank Dr. Hans-Uwe Dahms for English editing.

SUPPLEMENTARY MATERIAL

Supplementary material is available on the publisher's website along with the published article.

REFERENCES

[1] Anfinsen CB, The formation and stabilization of protein structure. Biochem J 1972; 128(4): 737-49.

[2] Lee D, Redfern O, Orengo C. Predicting protein function from sequence and structure. Nat Rev Mol Cell Biol 2007; 8(12): 995-1005.

[3] Marks DS, Hopf TA, Sander C. Protein structure prediction from sequence variation. Nat Biotechnol 2012; 30(11): 1072-80.

[4] Dill KA. Theory for the folding and stability of globular proteins. Biochemistry 1985; 24(6): 1501-9.

[5] Crescenzi P, Goldman D, Papadimitriou C, Piccolboni A, Yannakakis M. On the complexity of protein folding. J Comput Biol 1998; 5(3): 423-65.

[6] Jana ND, Sil J. Protein structure prediction in 2D HP lattice model using differential evolutionary algorithm. Proceedings of the International Conference on Information Systems Design and Intelligent Applications 2012 (India 2012) 2012; 132: 281-90.

[7] Su SC, Lin CJ, Ting CK. An effective hybrid of hill climbing and genetic algorithm for 2D triangular protein structure prediction. Proteome Sci 2011; 9(Suppl 1): S19.

[8] Bechini A. On the characterization and software implementation of general protein lattice models. PLoS One 2013; 8(3): e59504.

[9] Mahmoodabadi MJ, Mottaghi ZS, Bagheri A. HEPSO: High exploration particle swarm optimization. Inf Sci 2014; 273: 101-11.

[10] Davis L. Handbook of genetic algorithms. Van Nostrand Reinhold, 1991.

[11] Kennedy J. Particle swarm optimization. Springer, 2010.

[12] García-Gonzalo E, Fernández-Martínez JL. A brief historical review of particle swarm optimization (PSO). J Bioinf Intell Control 2012; 1(1): 3-16.

[13] Chang WD. A multi-crossover genetic approach to multivariable PID controllers tuning. Expert Syst Appl 2007; 33(3): 620-6.

[14] Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J Global Optimization 2007; 39(3): 459-71.

[15] Skiena SS. The algorithm design manual: Text. Springer Science & Business Media: New York, 1998.

[16] Faigle U, Fujishige S. A general model for matroids and the greedy algorithm. Math Program 2009; 119(2): 353-69.

[17] Jiang T, Cui Q, Shi G, Ma S. Protein folding simulations of the hydrophobic–hydrophilic model by combining tabu search with genetic algorithms. J Chem Phys 2003; 119(8): 4592-6.

[18] Islam MK, Chetty M, Murshed M. In Evolutionary Computation (CEC), 2011 IEEE Congress on, 2011, pp 1003-11.

[19] Unger R, Moult J. Genetic algorithms for protein folding simulations. J Mol Biol 1993; 231(1): 75-81.

[20] Krasnogor N, Blackburne B, Burke EK, Hirst JD. Multimeme algorithms for protein structure prediction. Springer: Berlin, 2002.

[21] Santos J, Diéguez M. Differential evolution for protein structure prediction using the HP model. Springer: Berlin, 2011.

[22] Hoque MT, Chetty M, Dooley LS. A hybrid genetic algorithm for 2D FCC hydrophobic-hydrophilic lattice model to predict protein folding. Springer, Berlin, 2006.

[23] Böckenhauer H-J, Dayem Ullah AZM, Kapsokalivas L, Steinhöfel K. A local move set for protein folding in triangular lattice models. Springer: Berlin, 2008.

[24] Ngo JT, Marks J, Karplus M. In The protein folding problem and tertiary structure prediction. Merz KM, Le Grand, S.M., Eds.; Birkhäuser Boston: Boston, MA, 1994, pp 433-506.

[25] Yogesh CK, Hariharan M, Ngadiran R, *et al.* A new hybrid PSO assisted biogeography-based optimization for emotion and stress recognition from speech signal. Expert Syst Appl 2017; 69: 149-58.

[26] Citrolo AG, Mauri G. A hybrid Monte Carlo ant colony optimization approach for protein structure prediction in the HP model. arXiv preprint arXiv:1309.7690 2013.

[27] Shatabda S, Newton M, Pham DN, Sattar A. A hybrid local search for simplified protein structure prediction. arXiv preprint arXiv:1310.8583 2013.

[28] Rashid MA, Newton MAH, Hoque MT, Sattar A. In Evolutionary Computation (CEC), 2013 IEEE Congress on, 2013, pp 1091-8.

[29] Garza-Fabre M, Rodriguez-Tello E, Toscano-Pulido G. Constraint-handling through multi-objective optimization: The hydrophobic-polar model for protein structure prediction. Comput Oper Res 2015; 53: 128-53.

[30] Cutello V, Nicosia G, Pavone M, Timmis J. An immune algorithm for protein structure prediction on lattice models. IEEE Trans Evol Comput 2007; 11(1): 101-17.