Research Article

Solving Two-Dimensional HP Model by Firefly Algorithm and Simplified Energy Function

Yudong Zhang,¹ Lenan Wu,¹ and Shuihua Wang²

¹ School of Information Science and Engineering, Southeast University, Nanjing, Jiangsu 210096, China

² Grover School of Engineering, The City College of New York, New York, NY 10031, USA

Correspondence should be addressed to Lenan Wu; wuln@seu.edu.cn

Received 18 December 2012; Accepted 9 January 2013

Academic Editor: Saeed Balochian

Copyright © 2013 Yudong Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to solve the HP model of the protein folding problem, we investigated traditional energy function and pointed out that its discrete property cannot give direction of the next step to the searching point, causing a challenge to optimization algorithms. Therefore, we introduced the simplified energy function into a turn traditional discrete energy function to continuous one. The simplified energy function totals the distance between all pairs of hydrophobic amino acids. To optimize the simplified energy function, we introduced the latest swarm intelligence algorithm, the firefly algorithm (FA). FA is a hot nature-inspired technique and has been used for solving nonlinear multimodal optimization problems in dynamic environment. We also proposed the code scheme strategy to apply FA to the simplified HP model with the clash test strategy. The experiment took 14 sequences of different chain lengths from 18 to 100 as the dataset and compared the FA with standard genetic algorithm and immune genetic algorithm. Each algorithm ran 20 times. The averaged energy convergence results show that FA achieves the lowest values. It concludes that it is effective to solve 2D HP model by the firefly algorithm and the simplified energy function.

1. Introduction

Protein folding is the process by which a protein structure assumes its functional shape or conformation. It is the physical process by which a polypeptide folds into its characteristic and functional three-dimensional structure from random coil. Each protein exists as an unfolded polypeptide or random coil when translated from a sequence of mRNA to a linear chain of amino acids [1]. This polypeptide lacks any developed three-dimensional structure. Amino acids interact with each other to produce a well-defined three-dimensional structure, the folded protein, known as the native state. The resulting three-dimensional structure is determined by the amino acid sequence [2, 3].

The protein folding has a challenging search space, since nature identifies the global minimum from more than 10^{50} possible conformations for the backbone of a small protein [4]. A successful prediction requires two major components: (1) a set of free energy components for the protein, which are computationally inexpensive enough to be used in the search

procedure and sufficiently accurate to ensure the uniqueness of the native fold; (2) an efficient optimization procedure which is capable of finding an appropriate minimum for the strongly anisotropic function of hundreds of variables [5, 6].

Scholars tend to use the 2D lattice model (HP model) for protein folding. The HP model was proposed by Lau and Dill [7]. In this model, proteins consist of two different kinds of residues, hydrophobic and hydrophilic. The task is to minimize the energy function, which is defined as the counting of every two hydrophobic residues that are nonconsecutive nearest neighbors on the lattice [8].

The recent literatures solving the 2D HP model were reported as follows. Lin and Hsieh [9] proposed an efficient hybrid Taguchi genetic algorithm that combines genetic algorithm, Taguchi method, and particle swarm optimization, in order to enhance the performance of predicting protein structure. In addition, Lin presented the PSO inspired by a mutation mechanism in a genetic algorithm. In the experiment, Lin demonstrated that their algorithm can be applied successfully to the protein folding problems based on

the hydrophobic-hydrophilic lattice model. Zhang and Wu [10] investigated the bacterial chemotaxis optimization (BCO) on the 2D lattice model. He compared BCO with standard genetic algorithm, immune genetic algorithm, and artificial immune system for various chain lengths. He concluded that the BCO has the highest successful rate. Albrecht et al. [11] presented results from three-dimensional protein folding simulations in the HP model on ten benchmark problems. Their simulations are executed by a simulated annealing-based algorithm with a time-dependent cooling schedule. The neighborhood relation is determined by the pull-move set. The results of 10 benchmark problems provided experimental evidence of the relationship between the maximum depth, the stopping criterion, the mean of nonzero difference of the objective function, and their bounds. Chen et al. [12] suggested that protein folding cores form early in the process of folding, and proteins may have evolved to optimize both folding speed and native-state stability. Chen developed a set of empirical potential functions and applied them to analyze interaction energies among secondary structure elements in two proteins. Their work demonstrated that the predicted folding core also harbors residue that form native like interactions early in the folding reaction. Using a set of 29 unrelated proteins, Chen demonstrated that the average prediction result from their method is significantly better than predictions based on other computation methods. Sharma et al. [13] provided a description which is consistent with other natural processes, that the protein folding is formulated from the principle of increasing entropy. It then became evident that protein folding is an evolutionary process among many others. During the course of folding protein, structural hierarchy builds up in succession by diminishing energy density gradients in the quest for a stationary state determined by surrounding density in energy. Evolution toward more probable states, eventually attaining the stationary state, naturally selects steeply ascending paths on the entropy landscape that correspond to steeply descending paths on the free energy landscape. The dissipative motion of the non-Euclidian manifold is nondeterministic by its nature which clarifies why it is so difficult to predict protein folding. Zhang et al. [14] proposed a novel chaotic clonal genetic algorithm (CCGA) on a 2D lattice model. The algorithm combines chaos operator, clonal selection algorithm, and genetic algorithm. The experiments compared CCGA with standard genetic algorithm and immune genetic algorithm for various chain lengths, and found that CCGA not only find global minima more reliably, but also be significantly faster in convergence. Zhao [15] had introduced several natural computing approaches. In his paper, the evolutionary algorithm, memetic algorithm, ant colony optimization algorithm, tabu search, self-organizing map-based iterative approaches, and the typical chain growth computing approaches were reviewed. The advantages and disadvantages of various computing approaches and the current optimal results are analyzed.

However, those above mentioned algorithms cannot converge to the global optimal points within limited time. The reason lies at two sides. The first is that the energy function is discrete, so it cannot effectively guide the searching point

as the continuous energy function. The second is that the optimization algorithms themselves are not rapid enough.

To solve the two outstanding issues, we introduced a simplified energy function [16], and we introduced the latest firefly algorithm (FA). FA is a hot nature-inspired technique and has been used for solving nonlinear multimodal optimization problems in dynamic environment [17]. The algorithm is based on the behavior of the fireflies. In social insect colonies, each firefly seems to have its own plans, and yet the group acts as a whole appears to be highly organized. Scholars published immense literatures reporting that its performance, effectiveness, and robustness are superior to GA, PSO, and other global algorithms in a wide range of fields [17–19].

The structure of this paper was organized as follows. Section 2 gave a brief overview on the simplified 2D HP model and its energy function. Section 3 introduced the fundamental of firefly algorithm, giving its pseudocodes and the encoding schemes to simplified 2D HP model. Section 4 contained the experiment of 14 chains varying from 18 to 100 residues and compared the FA algorithm to standard genetic algorithm and immune genetic algorithm. Finally, Section 5 concluded the paper and summarized our contribution.

2. Overview of 2D HP Model

Proteins are the basis of biology. They are the driving force behind all of the biochemical reactions. They are the main constituent of our bones, muscles, hair, skin, and blood vessels. They recognize invading elements and allow the immune system to get rid of the unwanted invaders. For these reasons, scientists have sequenced the human genome, the blueprint for all of the proteins in biology. However, only knowing this sequence tells us little about what the protein does and how it does it. In order to carry out their function, they must take on a particular shape (fold). Therefore, proteins are genuinely amazing machines: before they do their work, they assemble themselves, which is called “folding.” [20] The protein folding problem is so difficult, so scholars prefer to use the 2D HP model proposed by Lau and Dill [7]. The task is to find the optimal folded configuration of a given sequence. Each potential configuration is associated with an energy function. The optimal configuration corresponds to the lowest energy.

2.1. The 2D HP Model. There are two types of amino acids in the realistic world: hydrophobic amino acid and hydrophilic amino acid. Since proteins always exist in the watery environment, the hydrophobic amino acid and hydrophilic amino acid tend to cling to each other, so as to apart from water to the maximal degree [16].

Figure 1 shows an 8-residue chain (01111110) before and after folding. We use 1 to denote hydrophobic residue (denoted by *red* circle), and 0 to denote hydrophilic residue (denoted by *green* circle). Figure 1 indicates that (1) the chain can turn 90° left or right, or continue ahead at each point; (2) the hydrophobic residues compact themselves to keep away from water.

FIGURE 1: Illustration of HP model of protein chain 01111110: (a) before Folding; (b) after Folding.

2.2. Simplified Energy Function. In traditional HP model, the energy function is simple as adding -1 for each direct (occupying nondiagonal neighboring lattice points) between two hydrophobic amino acids, those are not consecutive in the protein sequences. Figure 2(a) shows a protein sequence with energy function as -2 .

However, it is not easy to pinpoint the direct and not consecutive two hydrophobic amino acids. Therefore, a simplified energy scoring method via distance matrix is proposed at Mathworks contest in 2002 [16]. We introduced in this strategy in our work. The energy is scored by totaling the distance between all pairs of hydrophobic amino acids. For Figure 2(b), we used the blue lines to link all pairs of hydrophobic amino acids, and summed distances is calculated as 21.129. The detailed calculation procedure is shown as follows.

Step 1. Chain = [0 1 1 1 1 1 1 0].

Step 2. Remove the hydrophilic residues (1st and last residues), and reserve the hydrophobic residues (2nd, 3rd, 4th, 5th, 6th, and 7th residues).

Step 3. Record their coordination as (0, 2), (0, 1), (0, 0), (1, 0), (1, 1), and (1, 2). See Figure 1(b).

Step 4. Calculate the residue-to-residue distances matrix as

$$\text{dist} = \begin{bmatrix} 0 & 1 & 2 & \sqrt{5} & \sqrt{2} & 1 \\ 1 & 0 & 1 & \sqrt{2} & 1 & \sqrt{2} \\ 2 & 1 & 0 & 1 & \sqrt{2} & \sqrt{5} \\ \sqrt{5} & \sqrt{2} & 1 & 0 & 1 & 2 \\ \sqrt{2} & 1 & \sqrt{2} & 1 & 0 & 1 \\ 1 & \sqrt{2} & \sqrt{5} & 2 & 1 & 0 \end{bmatrix}. \quad (1)$$

Step 5. Sum up the triangle part of the distance matrix

$$\text{Energy} = 7 + 4\sqrt{2} + 2\sqrt{5} + 2 * 2 = 21.129. \quad (2)$$

2.3. Theory Analysis of Simplified Energy Function. We expect that the simplified energy function is better than the traditional function. The reason is that the traditional energy

FIGURE 2: Comparison of two energy functions: (a) traditional energy function with energy as -2 ; (b) simplified energy function with energy as 21.129.

function is discrete or stair wisely with -1 decrease. It will not give instructive information for the searching point. While the simplified energy function used the sum of the distance matrix so as to transform traditional discrete energy function to a continuous one.

Figure 3 gave an illustration. Figure 3(a) shows the discrete energy function, where the searching point found the energy function remains as 1 within the searching area, so it can not give the direction of the next step for the searching point. Figure 3(b) shows the continuous energy function, where the searching point found the energy function varies, so it will move towards the minimal point within the searching area.

3. Firefly Algorithm

A power optimization algorithm is of importance to solve 2D HP model. Firefly Algorithm (FA) is a nature-inspired algorithm based on the flashing behaviors of the firefly swarm [19]. The primary purpose for the flash of fireflies is to signal to attract other fireflies. The assumption of FA consists of three rules: (1) all fireflies are unisex, so that one firefly will be attracted to other fireflies regardless of their sex; (2) an important and interesting behavior of fireflies is to glow brighter mainly to attract prey and to share food with others; (3) attractiveness is proportional to their brightness, thus each agent firstly moves toward a neighbor that glows brighter.

In the FA, the fireflies are randomly distributed in the search space. The fireflies carry a luminescence quality, called luciferin, which emits light proportional to the quality [18]. Each firefly is attracted by the brighter glow of other approximated fireflies. The attractiveness decreases as their distance increases [21]. If there is no brighter one within the scope of a firefly, it will move randomly in the search space.

3.1. Variation of Light Intensity. The brightness is related to the objective values, so for an optimization problem, a firefly with higher intensity will attract another firefly with higher probability, and vice versa. Assume that there exists a swarm of n fireflies, and x_i represents a solution for a firefly i , whereas $f(x_i)$ denotes its corresponding energy value. Here,

FIGURE 3: The effect of energy function on the optimization procedures: (a) discrete energy function; (b) continuous energy function. The black dot denotes the searching point, and the black circle denotes the searching territory.

the brightness I of a firefly is equivalent to the simplified energy value

$$I_i = f(x_i), \quad 1 \leq i \leq n. \quad (3)$$

3.2. Movement towards Attractive Flies. The attractiveness β of the firefly is proportional to the light intensity received by the adjacent fireflies [22]. Suppose β_0 is the attractiveness with distance $r = 0$, so for two fireflies i and j at locations x_i and x_j , their attractiveness is calculated as

$$\begin{aligned} \beta_r(i, j) &= \beta_0 \exp\{-\gamma r(i, j)^2\} \\ r(i, j) &= \|x_i - x_j\|, \end{aligned} \quad (4)$$

where $r(i, j)$ denotes the distance between fireflies i and j , γ denotes the light absorption coefficient. Suppose that firefly j is brighter than firefly i , then firefly i will move to a new location as

$$x_i(t+1) = x_i(t) + \beta_0 \exp\{-\gamma r^2\} (x_j - x_i). \quad (5)$$

3.3. Pseudocodes. See Pseudocode 1.

3.4. Encoding Scheme. For the 2D Hp model, we set the firefly as a string of digits chosen from the set $\{0, 1, 2\}$. Each digit denotes a torsion angle as: $0 \rightarrow$ left, $1 \rightarrow$ right, and $2 \rightarrow$ continue [8]. The initialization of FA generated the fireflies with random digits, and the first link of the sequence is not included in the encoding. Figure 4 shows that the direction of the first link only makes the structure rotated. Their structures of the lattice model does not change, neither does the corresponding energy levels.

3.5. Clash Test. Sometimes the decoded string represents a clash, which means a lattice node is occupied by more than

one residue, as shown in Figure 5. If a clash occurred, this firefly will be replaced by a new randomly generated firefly. If the clash occurs in the new one, then the procedures are repeated until no clash occurs. Meanwhile, the clash tests need to be checked at every step after the fireflies are updated, see (5) for detailed information.

4. Experiments

The experiments were carried out on the platform of P4 IBM with 3.1 GHz processor and 2 G RAM, running under Windows XP operating system. The algorithm was in-house developed via Matlab 2012a. The experiment data contains 14 protein folding problems with different chain lengths varying from 18 to 100 generated randomly. Their sequences are listed in Table 1.

4.1. Effectiveness of Simplified Energy Function. Figure 6 gave the optimal solutions obtained by FA through the simplified HP model. It forces the hydrophobic residues to fold themselves together to the utmost extent. Obviously, all the hydrophobic residues are located at the inner side, while the hydrophilic residues are located outside. It satisfies the expectation of 2D HP model. Therefore, this simplified energy function is effective.

4.2. Algorithm Comparison. We compared the FA with traditional standard genetic algorithm (SGA) and immune genetic algorithm (IGA). We ran each algorithm 20 times and got the averaged converged energy values of total 20 times results. The parameters are set by hundreds of experiments to get the optimal values and are shown in Table 2.

The average convergence results were illustrated in Figure 7. It indicated that the firefly algorithm achieves the least averaged convergence values among all three algorithms

The pseudo-codes of the firefly algorithm can be summarized as

Step 1 Initialization.

Step 1.1 Create the initial population of n fireflies ($x_1, x_2, \dots, x_n$) within d -dimensional search space.

Step 1.2 Formulate light intensity of each firefly so as to be associated with the energy value $f(x)$.

Step 1.3 Define the parameters β_0 and γ .

Step 2 Perform.

while (termination criteria are not met)

for $i = 1$ to n

for $j = 1$ to n

if ($I_j < I_i$)

Move firefly j towards firefly i via (5).

end if

Update Attractiveness Table.

Evaluate New Solutions and Update light intensity.

end for j

end for i

Rank the fireflies and find the best.

end for while

Step 3 Post-processing and output.

PSEUDOCODE 1

FIGURE 4: The direction of the first link does not contribute to the encoding. The first link directs to: (a) North, (b) West, (c) South, and (d) East does not affect the protein structure. They are all encoded as “220022”, and their energies are all 21.129.

for all protein sequences. Therefore, the FA is the most effective among the three algorithms. Besides, the longer the proteins become (the index increases), the larger the advantages of FA turn.

4.3. Energy Function Comparison. In this chapter, we compared the simplified energy function with traditional energy function. We choose the first protein in the dataset, of which the minimal traditional energy function is -5 , and the minimal simplified energy function is 100.97 . Their convergence curves are shown in Figure 8.

We see that the simplified energy function merely costs about $1/3$ iterations of those needed by the traditional energy function. The reason lies in the fact that the traditional energy function is discrete and cannot give effective direction for the searching point to move to, whereas the simplified energy function can be regarded as a continuous function, so the searching point is easy to guide itself moving towards the minimal point within its search territory. Moreover, Figure 8 validates that our developed theory in Section 2.3 is coherent to the experimental results.

5. Conclusions and Future Work

This study introduced the firefly algorithm and the simplified energy function and applied them to the 2D HP lattice model of protein folding problem. The experimental results on 20 runs of 14 chains with different lengths showed that the FA can achieve the lowest averaged energy values compared to standard genetic algorithm and immune genetic algorithm. We also prove the superiority of the simplified energy function on traditional energy function.

Our contributions can be summarized as follows.

- (1) We applied firefly algorithm to the field of protein folding problem.
- (2) We introduced the simplified energy function.
- (3) We demonstrate the superiority of the simplified energy function on traditional energy function on both from the discrete and continuous energy function theory and experimental results.

The future work focuses on three points. First is to investigate more novel intelligence algorithms, including

FIGURE 5: Illustration of clashes: (a) no Clash (encoded as “220022”); (b) clash (encoded as “220002”).

FIGURE 6: Continued.

FIGURE 6: Optimal solutions of the 14 sequences found by FA: (a) 18 residues; (b) 26 residues; (c) 30 residues; (d) 39 residues; (e) 42 residues; (f) 49 residues; (g) 53 residues; (h) 66 residues; (i) 72 residues; (j) 79 residues; (k) 83 residues; (l) 89 residues; (m) 91 residues; (n) 100 residues.

FIGURE 7: The algorithm comparison by averaged convergence values.

- [5] B. C. Gin, J. P. Garrahan, and P. L. Geissler, "The limited role of nonnative contacts in the folding pathways of a lattice protein," *Journal of Molecular Biology*, vol. 392, no. 5, pp. 1303–1314, 2009.
- [6] Y. Zhang, S. Wang, L. Wu, and Y. Huo, "Artificial immune system for protein folding model," *Journal of Convergence Information Technology*, vol. 6, no. 1, pp. 55–61, 2011.
- [7] K. F. Lau and K. A. Dill, "Lattice statistical mechanics model of the conformational and sequence spaces of proteins," *Macromolecules*, vol. 22, no. 10, pp. 3986–3997, 1989.
- [8] R. König and T. Dandekar, "Improving genetic algorithms for protein folding simulations by systematic crossover," *BioSystems*, vol. 50, no. 1, pp. 17–25, 1999.
- [9] C. J. Lin and M. H. Hsieh, "An efficient hybrid Taguchi-genetic algorithm for protein folding simulation," *Expert Systems with Applications*, vol. 36, no. 10, pp. 12446–12453, 2009.
- [10] Y. Zhang and L. Wu, "Bacterial chemotaxis optimization for protein folding model," in *Proceedings of the 5th International Conference on Natural Computation (ICNC '09)*, pp. 159–162, Tianjian, China, August 2009.
- [11] A. A. Albrecht, A. Skaliotis, and K. Steinhöfel, "Stochastic protein folding simulation in the three-dimensional HP-model," *Computational Biology and Chemistry*, vol. 32, no. 4, pp. 248–255, 2008.
- [12] M. Chen, A. D. Dousis, Y. Wu, P. Wittung-Stafshede, and J. Ma, "Predicting protein folding cores by empirical potential functions," *Archives of Biochemistry and Biophysics*, vol. 483, no. 1, pp. 16–22, 2009.
- [13] V. Sharma, V. R. I. Kaila, and A. Annala, "Protein folding as an evolutionary process," *Physica A*, vol. 388, no. 6, pp. 851–862, 2009.
- [14] Y. Zhang, L. Wu, Y. Huo, and S. Wang, "Chaotic clonal genetic algorithm for protein folding model," in *Proceedings of the International Conference on Computer Application and System Modeling (ICCASM '10)*, pp. V3120–V3124, October 2010.
- [15] X. Zhao, "Advances on protein folding simulations based on the lattice HP models with natural computing," *Applied Soft Computing Journal*, vol. 8, no. 2, pp. 1029–1040, 2008.
- [16] "Protein Folding Contest Rules," <http://www.mathworks.com/matlabcentral/contest/contests/11/rules>.
- [17] J. Senthilnath, S. N. Omkar, and V. Mani, "Clustering using firefly algorithm: performance study," *Swarm and Evolutionary Computation*, vol. 1, no. 3, pp. 164–171, 2011.
- [18] M.-H. Horng and R.-J. Liou, "Multilevel minimum cross entropy threshold selection based on the firefly algorithm," *Expert Systems With Applications*, vol. 38, no. 12, pp. 14805–14811, 2011.
- [19] X.-S. Yang, S. S. S. Hosseini, and A. H. Gandomi, "Firefly Algorithm for solving non-convex economic dispatch problems with valve loading effect," *Applied Soft Computing*, vol. 12, no. 3, pp. 1180–1186, 2012.
- [20] "The science behind Folding," <http://folding.stanford.edu/English/Science>.
- [21] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Mixed variable structural optimization using Firefly Algorithm," *Computers & Structures*, vol. 89, no. 23–24, pp. 2325–2336, 2011.
- [22] M.-H. Horng, "Vector quantization using the firefly algorithm for image compression," *Expert Systems With Applications*, vol. 39, no. 1, pp. 1078–1091, 2012.