Laboratoire d'Informatique Fondamentale, de Recherche Opérationnelle, de Combinatoire et d'Économétrie

Availaible on line at http://www.liforce.usthb.dz
bulletin-liforce@usthb.dz
Bulletin du Laboratoire 02 (2023) 01 - 24

# A new hybrid flower pollination algorithm for protein structure prediction problem

**Nabil BOUMEDINE & Sadek BOUROUBI**

USTHB, Faculty of Mathematics,
P.B. 32 El-Alia, 16111, Bab Ezzouar, Algiers, Algeria.

nboumedine@usthb.dz & sbouroubi@usthb.dz

**Abstract :** Proteins are biological molecules that perform various functions in the cells of our body. Each protein has a specific amino acid sequence and structure, known as the native structure, which determines its biological function. Any changes in this structure can lead to a loss of function and can cause dangerous diseases such as Alzheimer's and Parkinson's disease. Therefore, predicting the tertiary structure of proteins is one of the most challenging topics in molecular biology and biophysics. With the numerous optimization methods developed in operations research, the problem of Protein Structure Prediction (PSP) based on the real folding process is being addressed with different optimization methods. In this paper, we present a discrete hybrid power pollination algorithm called HFPA for solving the PSP problem in a 2D triangular lattice based on the simplified hydrophobic-polar model. The proposed algorithm combines the Flower Pollination Algorithm (FPA), Tabu Search (TS), and Hybrid Genetic Algorithm (HGA).We implemented the HFPA algorithm and used it to solve a set of benchmark instances. We also compared its performance against state-of-the-art algorithms.

**Keywords:** Protein Structure Prediction; Flower Pollination Algorithm; Tabu Search Algorithm; Genetic Algorithm; Hydrophobic-Polar Model; Lowest-Energy Structure.

# 1 Introduction

Predicting the original structure of proteins from their amino acid sequences is one of the most difficult and challenging problems confronted by scientists in many areas, including biology, chemistry, and even mathematics [21, 8, 9, 10, 49]. The main drivers of this high degree of interest are the biological roles of proteins in living cells. A protein's native structure, which consists of a specific number and order of amino acids, determines its biological functions (i.e., the primary structure) [68]. Protein misfolding has also been linked to several serious diseases, including Alzheimer's, Parkinson's, and mad cow disease [35, 32, 33, 15]. As a result, developing an effective prediction approach based on the natural protein folding mechanism could help in the detection and treatment of a wide range of diseases. Currently, protein structure is determined using X-ray crystallography [13] and Nuclear Magnetic Resonance (NMR) [1]. However, these procedures take a long time and require a lot of equipment. As a result, computational approaches based on simplified models have been widely used to address the PSP problem. One of the most widely used models for simulating protein folding based on amino acid sequences is the Hydrophobic-Polar (HP) model, which is implemented on different lattices, including 2D square, 3D cubic, 2D triangular, and Face-Centered Cubic (FCC) (2D or 3D) lattices [9, 24, 50, 17].

This model not only simplifies and reduces the complexity of the Protein Folding Problem (PFP), but it is also based on the actual process of protein folding, where hydrophobic interactions between amino acids are the most important force that guides proteins to fold into their native state [2, 3]. Despite reducing the complexity of the protein folding problem, predicting the optimal protein structure under the HP model is still an NP-hard optimization problem [18, 19]. Predicting the native structure of proteins in the HP model using different metaheuristics has been widely investigated by scientists over the last 50 years. In the present work, we are interested in solving the PSP problem in the 2D triangular HP lattice model. In this model, Hoque et al. proposed a Hybrid Genetic Algorithm (HGA) to solve the PSP problem [20]. After the crossover and mutation operations, HGA employs an efficient strategy to change the conformations without causing any collisions. The experimental results showed that the HGA algorithm performs significantly better than the Standard Genetic Algorithm (SGA). Then, in [21], Su et al. suggested an effective hybrid algorithm (HHGA) based on a new elite-based reproduction strategy. HHGA combines the Genetic Algorithm (GA) and Hill-Climbing Algorithm at the level of crossover and mutation operations. According to the obtained results on a set of data benchmarks, the authors demonstrate that the proposed HHGA algorithm outperforms both SGA and HGA algorithms. Yang et al. proposed a new hybrid method called IMOG in [53], the proposed IMOG algorithm combines the Ion Motion Optimization (IMO) algorithm with the greedy algorithm. The statistical results based on the best and mean energy indicate that IMOG outperforms HGGA, HGA, and SGA and achieves high-quality results for the majority of tested instances. To solve the PSP problem, Guo et al. proposed an Extended version of the Particle Swarm Optimization algorithm called EPSO [13]. Experimental results showed that EPSO achieved better conformation quality than SGA and produced comparable results to HHGA.

However, in terms of execution time, EPSO outperformed HHGA. Recently, Boumedine et al. proposed a new hybrid genetic algorithm (GATSLS) that incorporates a novel reproduction scheme, where the crossover operation is guided by the tabu search strategy. They further improve the quality of the solutions using the local search algorithm [38]. Based on the experimental results, the authors demonstrated that the proposed GATSLS algorithm can easily obtain the best-known solution for short protein instances and a near-optimal solution for long protein instances. Additionally, the authors showed that the GATSLS algorithm can quickly obtain the best-known solution for short protein instances and near-optimal solutions for long and complex protein instances.

Furthermore, the GATSLS algorithm outperforms all current state-of-the-art algorithms. Recently, Yang et al. introduced the Flower Pollination Algorithm (FPA) [54], which is based on the real pollination process of flowering plants. Since its introduction in 2012, this algorithm has been widely employed to solve many challenging optimization problems. Due to its high ability to explore the search space, the FPA algorithm has been successful in solving many continuous optimization problems more efficiently than other popular nature-based algorithms, such as genetic algorithms and particle swarm optimization [23]. The efficiency and interesting results achieved by FPA have motivated us to apply it to the PSP problem, which represents a discrete optimization problem. In this paper, we introduce a discrete hybrid ower pollination algorithm labeled HFPA to solve the PSP problem in a 2D triangular lattice.

The remainder of this paper is structured as follows: In Section 2, we offer the HP model and the 2D triangular lattice used to conduct this study. Then, in Section 3, we present the different basic concepts of the Flower Pollination Algorithm (FPA). Section 4 provides a brief description of the proposed Hybrid Flower Pollination Algorithm (HFPA) and the motivation behind this hybridization. In Section 5, we report and discuss the results obtained by the proposed algorithm on two sets of benchmark instances. Finally, in the last section, we offer some interesting conclusions and perspectives for future work.

## 2 Protein folding in lattices under the HP model

The hydrophobic amino acids form the core of the native conformation during the folding of proteins, through interactions between them that reduce the free energy of the structure to its lowest possible level [7, 22]. This leads to the stability of the structure in its active conformation. Based on this theory, Dill et al. proposed a simplified model of hydrophobic-polar folding called the HP model [24]. The 20 amino acids that make up the primary structure of proteins are divided into two groups in this model: H (hydrophobic) and P (polar). The energy value of the conformation is determined by the number of contacts between hydrophobic amino acids (H-H contacts) that are not neighbors in the protein sequence and are located in two adjacent positions on the grid. The optimal conformation is the one with the lowest energy value, which corresponds to the maximum number of H-H contacts [3]. Figure 1 shows the optimal conformation for a protein sequence of 20 amino acids, with the corresponding sequence in the HP model being HPHPPHHPHPPHPHHPPHPH. This conformation has 15 topological contacts between hydrophobic amino acids, resulting in a free energy of $E = -15$. In lattice models, protein conformations are represented by a sequence of lattice movements that do not pass through the same position twice [38].

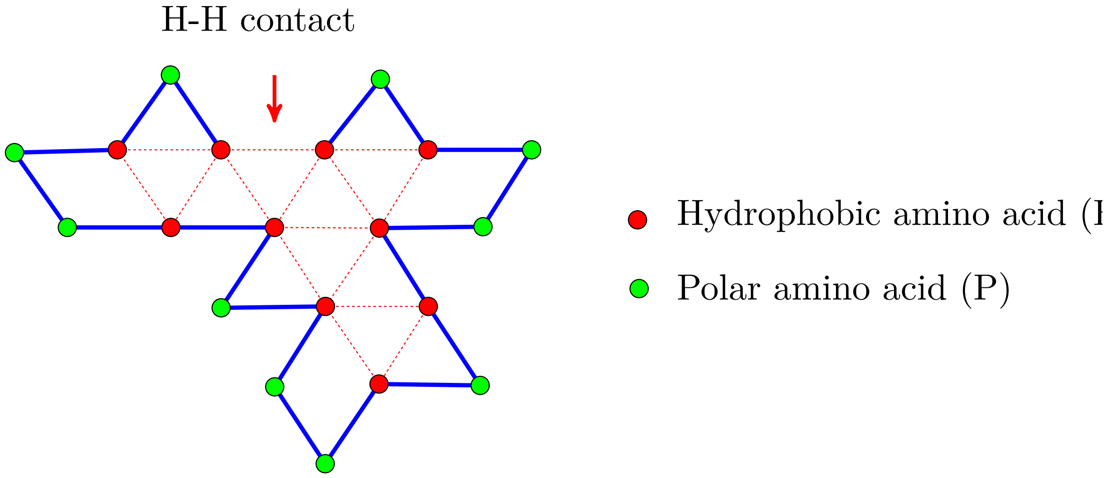


Figure 1: An optimal conformation refers to the most favorable shape of a protein sequence containing 20 amino acids on a 2D triangular lattice.

## 2.1 The determination of free energy

In HP lattice models, the quality of a given protein conformation is evaluated based on the number of H-H contacts. For a protein sequence $p = p_1p_2 \dots p_n$ of $n$ amino acids, let $C(p)$ be the set of all feasible conformations for $p$. The free energy $E(C(p))$ can be determined using the following mathematical formula [26]:

$$E(C(p)) = -\sum_{i=1}^{n-1} \sum_{i=i+1}^{n} \theta_{i,j} r_{i,j},$$

where

$$\theta_{i,j} = \begin{cases} 1, & \text{if the } i^{th} \text{ and } j^{th} \text{ amino acids are hydrophobic, i.e., } p_i = p_j = H, \\ 0, & \text{else,} \end{cases}$$

and

$$r_{i,j} = \begin{cases} 1, & \text{if } p_i \text{ and } p_j \text{ are adjacent in the grid but not consecutive in the sequence } P, \\ 0, & \text{else.} \end{cases}$$

The main challenge in protein structure prediction using HP lattice models is to identify the conformation $c^*$ that minimizes the free energy [29]:

$$E(c^*) = \min\{E(c) \mid c \in C(p)\}.$$

# 3 Flower Pollination Algorithm (FPA)

Recently, in 2012, Yang et al. developed the Flower Pollination Algorithm (FPA) [54], a metaheuristic inspired by the behavior of a variety of owers during their pollination for reproduction [55]. The flower pollination process can be characterized by the following rules:

1. Biotic pollination: can be considered a form of global pollination that relies on pollinators like birds, insects, and bees to carry out long-distance or cross-pollination.
2. Abiotic pollination: refers to self-pollination or local pollination that occurs within a flower and does not require external pollinators.
3. Constancy in pollination: refers to the tendency of pollinators to visit flowers of the same species due to their similarity. The likelihood of successful reproduction is correlated with the probability of pollinators providing consistent pollen transfer between flowers.
4. The Switch Probability, denoted as $p$ and ranging from 0 to 1: is used to achieve a balanced approach between local and global pollination.

To develop the FPA algorithm, Yang et al. mathematically represented each pollination rule:

## 3.1 Global pollination

Global pollination is a search strategy that aims to explore new regions of the search space. It is defined by the following mathematical formula [34]:

$$x_i^{t+1} = x_i^t + \alpha L(\lambda)(x^* - x_i^t), \tag{1}$$

where

$$L(\lambda) = \frac{\lambda\Gamma(\lambda)\sin(\lambda)}{\pi} \cdot \frac{1}{s^{1+\lambda}} \qquad s \gg s_0 \gg 0,$$

and $x_i^t$ represents the $i$th pollen at the current iteration $t$ (i.e., the current solution), and $x^*$ represents the best solution identified in all previous generations (i.e., the current best solution). The parameter $\alpha$ is a scaling factor that controls the step size. $L(\lambda)$ represents the Lévy flight step size, which is one of the most important and in influential search mechanisms in the FPA algorithm. The Lévy flight step size is determined by the standard gamma distribution $\Gamma$.

### 3.2 Local pollination

Local pollination is a search method that focuses on finding the local optimum within a specific region of the search space. Within the FPA algorithm, this is accomplished through the use of the following mathematical formula, as stated by Yang [54]:

$$x_i^{t+1} = x_i^t + \epsilon(x_j^t - x_k^t). \tag{2}$$

Here, $x_i^t$ represents the pollen of iteration $t$ for the $i$th solution. The values $x_j^t$ and $x_k^t$ correspond to two different solutions randomly selected from the current generation t, while $\epsilon$ is a random number generated from a uniform distribution over the interval [0 1]. The pseudo-code for the CSA algorithm is presented in Algorithm 1.

**Algorithm 1** Flower Pollination Algorithm: FPA

**Require:** Problem instance $I$.

**Ensure:** The best-found solution for $I$.

```
Begin
    Objective Function $f(x)$;
    Generate initial population $F$ of $m$ flowers in random way, $x_1, \dots, x_m$;
    $x^*$ : The best solution in the initial population $F$;
    Set the probability of the switch $p$;
    $t = 0$;
    while $t \leq$ Max (maximum number of iterations) do
        for each flower $x_i^t, i = 1, \dots, m$ do
            if $rand < p$ then
                Define a step vector $L$ that according to a Lévy distribution.
                Perform a global pollination to generate a new solution $x_i^{t+1}$ via Equation 1.
            else
                Generate $\epsilon$ from the uniform distribution.
                Randomly select two solutions from the current population.
                Perform a local pollination to generate a new solution $x_i^{t+1}$ via Equation 2.
            end if
            if $f(x_i^{t+1}) < f(x_i^t)$ then
                $x_i^t = x_i^{t+1}$;
            end if
        end For
        Update the current optimal solution $x^*$;
        $t = t + 1$;
    end while
End
```

### 3.3 Application of FPA

Since the initial publication of the FPA algorithm, it has been successfully applied to solve numerous challenging real-world optimization problems in various domains, including electrical engineering problems such as PV model parameter estimation [52, 37], economic load dispatch [36,37], reactive power dispatch [41, 25], and optimal power flow [46, 45]. The algorithm has also been applied in wireless and network domains, such as wireless sensor network optimization [44, 14], antenna array optimization [42, 43], and multiplexing of optical divisions [18]. In the clustering domain, FPA has been used for data clustering [19, 36] and neural network feedforward training [5]. In signal and image processing, the algorithm has been applied to tasks such as medical image segmentation [51], localization of retinal vessels, and shape matching. Furthermore, the FPA algorithm has been utilized in mechanical engineering problems, such as designs of mechanical and structural engineering [30, 28, 31], as well as for several global optimization problems [27].

## 4 Hybrid Flower Pollination Algorithm (FPA)

Most optimization problems have multiple local optima, and the efficiency of the optimization method used has a substantial impact on finding the global optimal solution. To effectively solve a given optimization problem, the metaheuristic employed should strike a balance between global and local search. In light of this, numerous hybrid metaheuristics integrating multiple approaches have been developed to address various optimization problems.

In this work, we present an efficient hybrid algorithm, called the Hybrid Flower Pollination Algorithm (HFPA), which combines the Genetic Algorithm (GA) with the standard FPA algorithm. The main goal of this integration is to create a more powerful optimization method that leverages the advantages of various pure techniques. Our proposed method involves two search phases, each consisting of global and local search. We control the balance between the two search phases using the switching probability of the FPA algorithm. For the first search phase, we use an adaptive Lévy distribution for global search and the Tabu Search (TS) algorithm to improve the quality of the solutions obtained. We also propose an adaptive Lévy distribution for discrete optimization problems like the Permutation Flow Shop Problem (PSP). For the second search phase, we use the mutation operator of the GA algorithm for diversification and the crossover operation to generate high-quality solutions. The pseudo-code for the proposed HFPA algorithm is presented in Algorithm 2 and is described in detail below.

**Algorithm 2** Hybrid Flower Pollination Algorithm: HFPA

**Require:** Problem instance $I$.

**Ensure:** The best-found solution for $I$.

```
Begin
    Objective Function $f(x)$;
    Generate initial population $F$ of $m$ flowers in random way, $x_1, \ldots, x_m$;
    $x^*$ : The best solution in the initial population $F$;
    Set the probability of the switch $p$;
    $t = 0$;
    while $t \leq$ Max (maximum number of iterations) do
        for each flower $x_i^t, i = 1, \ldots, m$ do
            if $rand < p$ then
                First step;
                Perform a global pollination: generate a new solution $x_i^{t+1}$ via Lévy distribution.
                Perform a local pollination: improve $x_i^{t+1}$ using the tabu search algorithm;
            else
                Second step;
                Perform a global pollination: generate a new solution $x_j^t$ by applying the mutation operator to $x_i^t$.
                Select a random solution $x_k^t$ from the current population.
                Perform a local pollination: generate a new solution $x_i^{t+1}$ by applying the crossover operator to $x_j^t$ and $x_k^t$;
            end if
            if $f(x_i^{t+1}) < f(x_i^t)$ then
                $x_i^t = x_i^{t+1}$;
            end if
        end For
        Update the current optimal solution $x^*$;
        $t = t + 1$;
    end while
End
```

## 4.1 Initialization

Each node of the 2D triangular lattice has six neighbors, which we assign a folding direction to, denoted by a number $i$ where $i, i \in \{1, 2, 3, 4, 5, 6\}$. Specifically, we assign the direction of ”right” as (1), ”right up” as (2), ”left up” as (3), ”left” as (4), ”left down” as (5), and ”right down” as (6) (see Figure 2). For each amino acid, we assign the folding direction that corresponds to its position in the lattice [38]. The solution (b) in Figure 2) can be represented by its direction vector movement $v$ as follows:

$$v = (3, 5, 4, 6, 1, 1, 5, 1, 5, 6, 2, 1, 3, 1, 2, 4, 5).$$

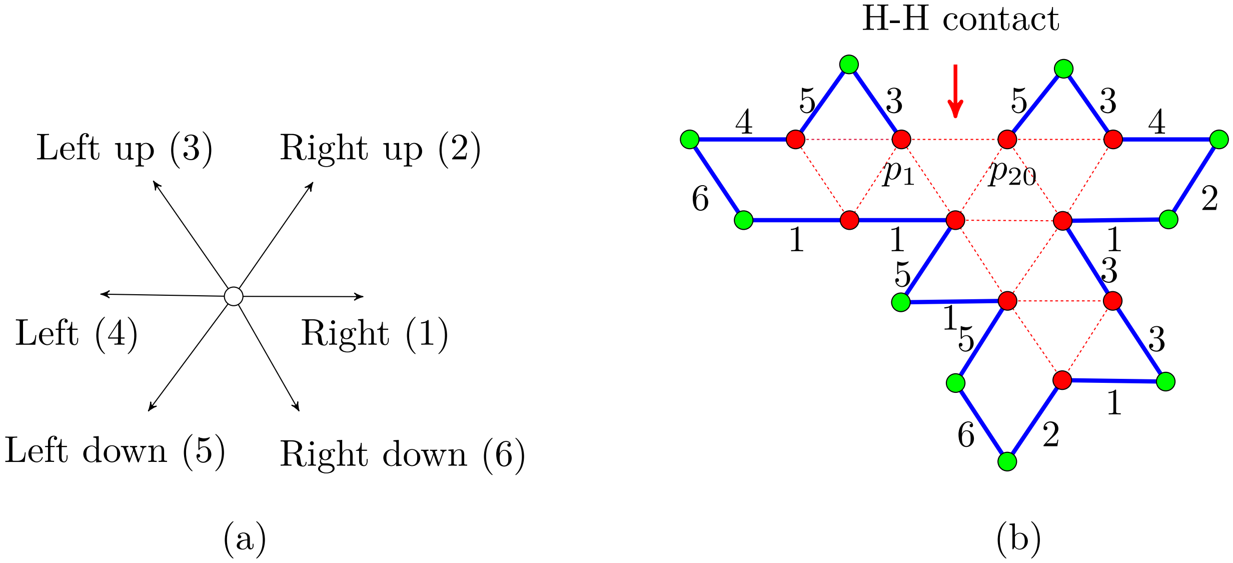


Figure 2: (a): Encoding of neighboring triangular lattice nodes. (b): Encoding of a feasible solution for a protein sequence containing 20 amino acids.

## 4.2 Global pollination via Lévy distribution

As mentioned earlier, Lévy flights are one of the most effective search mechanisms in the FPA algorithm, owing to their benefits and the ability to efficiently explore the search space with their step length properties [54]. However, they have been employed in the FPA algorithm for continuous optimization problems only. To utilize them in our discrete algorithm without forfeiting their advantages, we associate each value generated by the Lévy flights with a protein sub-sequence that we rotate using the rotation rules presented below [4]. Suppose n is the length of a given protein sequence $p_1 p_2 \dots p_n$. We divide the interval $[0, 1]$ into $k$ subintervals, where $k \leq n$. We then generate a Lévy flight value $v$ and define the step length $L$ according to the following table.

| | | | | |
|---|---|---|---|---|
| $v$ | $\left[0, \frac{1}{k}\right[$ | $\left[\frac{1}{k}, \frac{2}{k}\right[$ | $\cdots$ | $\left[\frac{k-1}{k}, 1\right]$ |
| $L$ | $\left[1, \frac{n}{k}\right[$ | $\left[\frac{n}{k}, \frac{2n}{k}\right[$ | $\cdots$ | $\left[\frac{(k-1)n}{k}, n\right]$ |

After defining the range of the generated value $v$ in the first row, we generate a random number $d$ within the corresponding range in the second row. The generated value $d$

represents the sub-sequence length that will be rotated as follows: Let $u$ be a random number such that $u \in \{1, \ldots, n-d\}$. The sub-sequence to be rotated is defined as $s = p_u p_{u+1} \ldots p_d$. We rotate $s$ by 60°, 120°, 180°, 240°, or 300°, based on a random number $j$, where $j \in \{1, 2, 3, 4, 5\}$ as illustrated in Table 1.

| | The corespondent direction after rotation | | | | |
|---|---|---|---|---|---|
| Direction | 60° | 120° | 180° | 240° | 300° |
| 1 | 2 | 3 | 4 | 5 | 6 |
| 2 | 3 | 4 | 5 | 6 | 1 |
| 3 | 4 | 5 | 6 | 1 | 2 |
| 4 | 5 | 6 | 1 | 2 | 3 |
| 5 | 6 | 1 | 2 | 3 | 4 |
| 6 | 1 | 2 | 3 | 4 | 5 |

Table 1: Rotation rules in 2D triangular lattice.

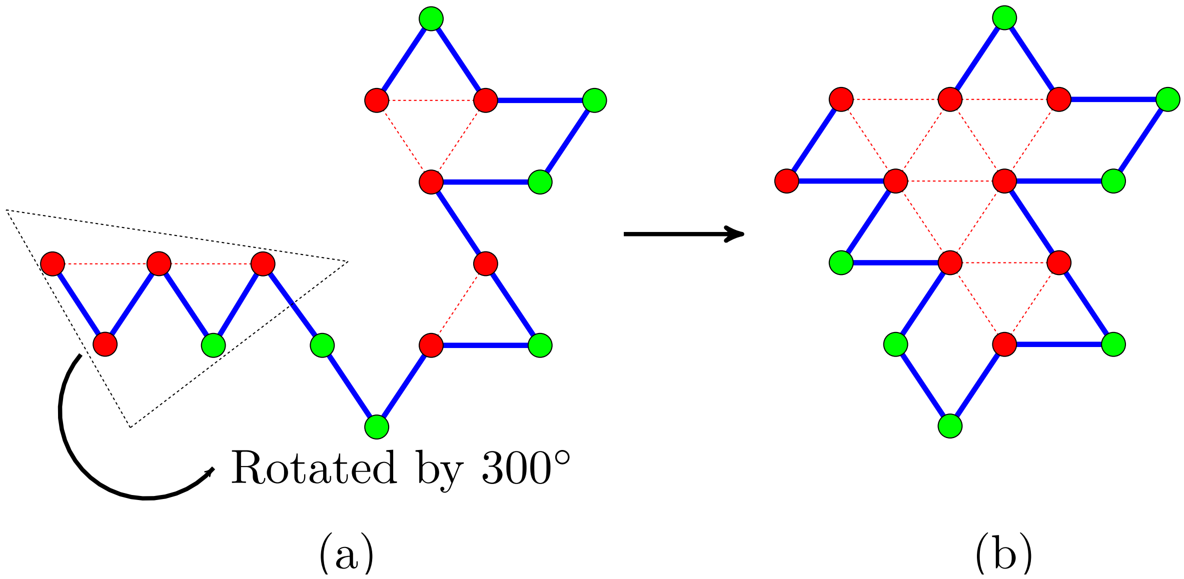


Figure 3: An example of rotation rules: the selected subsequence of the solution (a) is rotated by 300° to produce the solution (b).

## 4.3 Global pollination via mutation operator

We use the mutation operator of GA to increase the diversity of the search process [20]. In our suggested algorithm, the mutation operation is performed by selecting a random position in a given solution $x$, and replacing the value at that position with a randomly chosen value from the solution space, as illustrated in Figure 4.

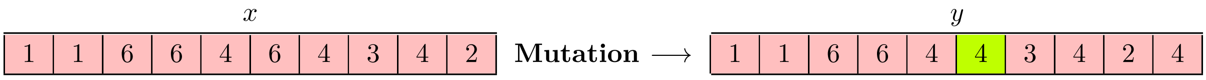


Figure 4: An example of the mutation operation is illustrated by replacing the value at the 6th position of solution $x$ from 6 to 4, resulting in a new solution $y$.

## 4.4 Local pollination via tabu search

To enhance the quality of solutions generated by the Lévy flight step, we adopt each solution as an initial solution in the Tabu Search (TS) algorithm [12, 11]. In the proposed TS algorithm, the neighbors of a given solution are determined through local movements. Specifically, a diagonal move is applied to a randomly selected amino acid by swapping its direction with that of its preceding amino acid, as shown in Figure 5). To avoid cyclic movements, we add each selected amino acid to the tabu list $T$. Any existing amino acids in the tabu list are excluded from future selections. The proposed algorithm follows the FIFO principle; that is, if the tabu list reaches its capacity, the first introduced element will be removed from the list (see Algorithm 3).

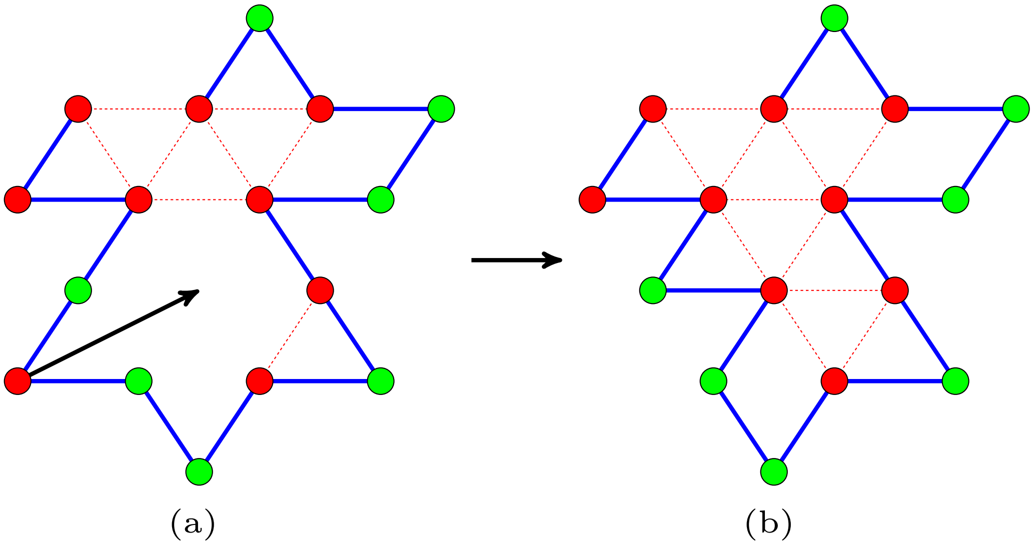


Figure 5: An example of the diagonal move, the obtained solution (b), $E(b) = -12$ is better than the initial solution a, $E(a) = -8$.

**Algorithm 3** The proposed Tabu Algorithm: TS

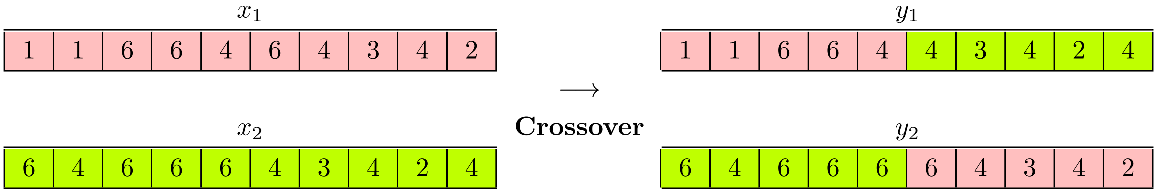
**Require:** A feasible solution $x = (x_1, \dots, x_n)$.

**Ensure:** The best found solution for $x^*$.

```
Begin
Calculate $f(x)$;
$x^* = x$;
$i = 0$
$T = \{\emptyset\}$// Tabu list;
while $i \leq$ Max (maximum number of iterations) do
    $j$ = random amino acids, $j \in \{2, 1, ..., n-1\}$
    if $j \in T$ then
        $x'$= diagonal pull-move $(x, j)$;
        $T = T \cup \{j\}$;
        if $f(x') < f(x)$ then
            $x = x'$;
        end if
    end if
    if $f(x) < f(x^*)$ then
        $x^* = x$;
    end if
    $i = i + 1$;
end while
End
```

## 4.5 Local pollination via crossover operator

The crossover operation consists of combining two or more solutions to create new high-quality solutions [20]. For our algorithm, we use a crossover operation with a crossover point. It consists of generating a random position $c$ and exchanging the directions of movement between two selected solutions called parents $(x_1, x_2)$. As we show in Figure 6, this operation generates two new solutions called children $(y_1, y_2)$. For our algorithm, we choose the best of them to be a candidate solution for the next step.



$x_1$: 1 1 6 6 4 6 4 3 4 2

$x_2$: 6 4 6 6 6 4 3 4 2 4

$y_1$: 1 1 6 6 4 4 3 4 2 4

$y_2$: 6 4 6 6 6 6 4 3 4 2

Figure 6: An example of a crossover operation is illustrated by permuting the subsequence limited to the fifth position in the selected parent solutions $x_1$ and $x_2$, resulting in the new solutions $y_1$ and $y_2$.

# 5 Experimental Results

In this section, we evaluate the effectiveness and stability of the proposed HFPA for solving the PSP problem. To conduct this experimental study, we use two datasets containing various protein sequences, as reported in Table 2 [39, 50, 6] and Table 3 [47, 22]. The symbol $(...)^k$ denotes the repetition of the sub-sequence in brackets $k$ times. The last column shows the best energy values reported in the literature denoted by $E^*$. We compare the performance of the proposed algorithm against the most efficient state-of-the-art methods reported in the literature. The comparison is based on the best, average, and worst results obtained from 20 independent runs for each instance.

To conduct our experimental study, we varied the value of each parameter in our algorithm and selected the best values that achieved the optimal results. Based on our preliminary experimental results, the appropriate values for the parameters of the HFPA algorithm are reported in Table 4.

We present the results of our proposed algorithm, HFPA, and compare them with state-of-the-art algorithms such as Standard Genetic Algorithm (SGA), Hybrid Genetic Algorithm (HGA) [16], ERS-GA (a hybrid genetic algorithm that combines GA algorithm and ERS strategy), Hybrid Hill-Climbing-Genetic Algorithm (HHGA) [48], IMOG (a hybridization of Ions Motion Optimization algorithm and Greedy algorithm) [53], Tabu Search (TS) [2], Extended Particle Swarm Optimization algorithm (EPSO) [13], and GALSTS (a hybrid version of genetic algorithm that combines Genetic Algorithm, Local Search algorithm, and Tabu Search strategy) [40]. The best predictions achieved by each algorithm are reported in Table 5.

From Table 5, we clearly see that the suggested HFPA algorithm achieves the best known energy value for the majority of instances and produces high-quality structures when compared to state-of-the-art algorithms. Although all the listed algorithms are able to produce the optimal structure for short and medium-sized instances, i.e., from the instance $A_1$ to $A_3$, with its efficient search strategy, which establishes a good balance between exploration and exploitation, HFPA achieves better results than state-of-the-art algorithms, with a significant gap for large and complex instances (i.e., from the instance $A_7$ to $A_{10}$). In comparison, we observe that GALSTS is the most competitive algorithm against the proposed algorithm.

For the second benchmark instance, Table 6 presents results that demonstrate the high performance of the proposed algorithm in predicting the optimal structure for all short and medium protein sequences. In comparison, simulation results show that the IMOG [53] and the proposed HFPA algorithm outperform the Multimeme algorithm presented in [22] (column MMA) for some sequences, with a significant difference observed for the sequence $B_{20}$. The IMOG and HFPA algorithms achieved the lowest energy values for all tested instances without exception. However, MMA was unable to find the optimal structure for instances $B_{15}$, $B_{18}$, $B_{19}$, and $B_{20}$.

To analyze the convergence behavior and stability of our algorithm, Table 7 presents the statistical results obtained by the suggested HFPA algorithm and state-of-the-art algorithms for the first set of benchmark problems. The table reports the best values and average values over 20 independent runs for each instance.

Based on the results presented in Table 7, we observe that for the first four instances (i.e., from $A_1$ to $A_4$), all the compared algorithms perform well and can generate high-quality structures with the best known energy value. However, compared to IMOG, GATSLS, ERS-GA, and HHGA, our proposed HFPA algorithm is more stable and robust in terms of average results. The statistical results for the last five instances, which involve more complex and longer sequences, demonstrate a significant improvement achieved by the proposed algorithm with a considerable gap in the best and average results compared to state-of-the-art algorithms, especially when compared to ERS-GA, HHGA, and IMOG. For instance, for $A_7$, the difference between HFPA and ERS-GA, HHGA, and IMOG ranges from -5 to -16, and for instance $A_8$, it ranges from -10 to -26.

To evaluate the performance and search capabilities of the proposed HPFA algorithm compared to the standard FPA and GA algorithms, we implemented all three algorithms using the same initial population of 200 randomly generated solutions and the same number of generations. The GA algorithm had a crossover rate of 0.85 and a mutation rate of 0.05, as used in [16]. The FPA algorithm used Lévy flights for global pollination and a local search algorithm for local pollination. Each algorithm was run 15 times independently for each instance, and the statistical results are presented in Table 8. The results clearly show that the HPFA algorithm outperforms the GA and FPA algorithms for the majority of tested instances in terms of the best, worst, and average results. Additionally, the worst results produced by HPFA are better than the best results produced by GA and FPA. This excellent balance between global and local search techniques integrated into the proposed HPFA justifies its efficiency. The HPFA is more efficient in the local search phase using the crossover operation and tabu search algorithm, while Lévy flights and mutation improve its capability during the global search phase.

| Seq. | Length | Sequence | $E^*$ |
|---|---|---|---|
| $A_1$ | 20 | $(HP)^2PH(HP)^2(PH)^2HP(PH)^2$ | -15 |
| $A_2$ | 24 | $H^2P^2(HP^2)^6H^2$ | -17 |
| $A_3$ | 25 | $P^2HP^2(H^2P^4)^3H^2$ | -12 |
| $A_4$ | 36 | $P(P^2H^2)^2P^5H^5(H^2P^2)^2P^2H(HP^2)^2$ | -24 |
| $A_5$ | 48 | $P^2H(P^2H^2)^2P^5H^{10}P^6(H^2P^2)^2HP^2H^5$ | -43 |
| $A_6$ | 50 | $H^2(PH)^3PH^4PH(P^3H)^2P^4(HP^3)^2HPH^4(PH)^3PH^2$ | -40 |
| $A_7$ | 60 | $PH(PH^3)^2P(PH^2PH)^2H(HP)^3(H^2P^2H)^2PHP^4(H(P^2H)^2)^2$ | NA |
| $A_8$ | 64 | $H^{12}(PH)^2((P^2H^2)^2P^2H)^3(PH)^2H^{11}$ | NA |
| $A_9$ | 85 | $H^4P^4H^{12}P^6(H^{12}P^3)^3HP^2(H^2P^2)^2HPH$ | NA |
| $A_{10}$ | 100 | $P^3H^2P^2H^4P^2H^3(PH^2)^3H^2P8H^6P^2H^6P^9HPH^2$<br>$PH^{11}P^2H^3PH^2PHP^2HPH^3P^6H^3$ | NA |

NA: refers for 'data not accessible in literature'.

$E^*$: refers for 'best know energy value'.

Table 2: The first set of benchmark instances for the HP problem is in a 2D triangular lattice.

| Seq. | Length | Protein sequence in the H-P model | $E^*$ |
|---|---|---|---|
| $B_1$ | 12 | $H(HP)^5H$ | -11 |
| $B_2$ | 14 | $HHPP(HP)^5$ | -11 |
| $B_3$ | 14 | $H(HPP)^2(HP)^3H$ | -11 |
| $B_4$ | 16 | $HHP(HPP)^4H$ | -11 |
| $B_5$ | 16 | $H(HPP)^2(HP)^3PHP$ | -11 |
| $B_6$ | 17 | $H(HPP)^5H$ | -11 |
| $B_7$ | 17 | $H(HH)^7HH$ | -17 |
| $B_8$ | 20 | $H(HPP)^2(HP)^3(PH)^3H$ | -17 |
| $B_9$ | 20 | $H(HP)^4H(PPH)^3H$ | -17 |
| $B_{10}$ | 21 | $H(HPP)^2(HPHPP)^2HPHH$ | -17 |
| $B_{11}$ | 21 | $HHP(HPP)2(HP)2(HPP)2HH$ | -6 |
| $B_{12}$ | 21 | $HHPP(HP)^3(PH)^2(PPH)^2H$ | -17 |
| $B_{13}$ | 22 | $H(HPP)^2(HP)^3(PHP)^2PHH$ | -17 |
| $B_{14}$ | 23 | $HH(HP)^9HHH$ | -25 |
| $B_{15}$ | 24 | $H(HPP)^7HH$ | -17 |
| $B_{16}$ | 24 | $HH(HP)^3(PH)^7HH$ | -25 |
| $B_{17}$ | 24 | $HH(HP)^4(PH)^6HH$ | -25 |
| $B_{18}$ | 30 | $HH(HPP)^4H(PHPPH)^2PPHHH$ | -25 |
| $B_{19}$ | 30 | $HH(HPP)^3(HP)^2(PH)^2(PPH)^3HH$ | -25 |
| $B_{20}$ | 37 | $HH(HPP)^3(HP)^2H(PPH)^3(P)^5(HP)^2HHH$ | -29 |

$E^*$: refers for 'best know energy value.'

Table 3: The second set of HP benchmark instances in 2D triangular lattice.

| Parameter | The tested values | The best value |
|---|---|---|
| Switch probability $p$ | $p \in \{0.1, 0.2, 0.4, 0.6, 0.8\}$ | 0.6 |
| Step Lévy flight $\lambda$ | $b \in \{0.5, 1, 1.5, 2, 2.5\}$ | 1.5 |
| Population size $m$ | $m \in \{50, 80, 100, 150, 200, 250\}$ | 80 if $n \leq 50$<br>150 if $n > 150$ |
| Tabu search size $T$ | $T \in \left\{\frac{n}{5}, \frac{n}{4}, \frac{n}{3}, \frac{n}{2}\right\}$ | $\frac{n}{4}$ |

$n$ is the protein sequence length.

Table 4: Parameter Settings of HFPA algorithm.

| Seq. | Length | SGA | HGA | TS | ERS-GA | HHGA | EPSO | IMOG | GALSTS | HFPA |
|---|---|---|---|---|---|---|---|---|---|---|
| $A_1$ | 20 | -11 | **-15** | **-15** | **-15** | **-15** | NA | **-15** | **-15** | **-15** |
| $A_2$ | 24 | -10 | -13 | **-17** | -13 | **-17** | **-17** | **-17** | **-17** | **-17** |
| $A_3$ | 25 | -10 | -10 | **-12** | -12 | **-12** | **-12** | **-12** | **-12** | **-12** |
| $A_4$ | 36 | -16 | -19 | **-24** | -20 | -23 | **-24** | **-24** | **-24** | **-24** |
| $A_5$ | 48 | -26 | -32 | -40 | -32 | -41 | -40 | -40 | **-43** | **-43** |
| $A_6$ | 50 | -21 | -23 | NA | -30 | -38 | NA | **-40** | **-40** | **-40** |
| $A_7$ | 60 | -40 | -46 | **-70** | -55 | -66 | NA | -67 | -70 | **-71** |
| $A_8$ | 64 | -33 | -46 | **-50** | -47 | -63 | NA | -63 | -67 | **-73** |
| $A_9$ | 85 | NA | NA | NA | NA | NA | NA | NA | -98 | **-100** |
| $A_{10}$ | 100 | NA | NA | NA | NA | NA | NA | NA | -87 | **-91** |

Values in bold represent the best energy value for the correspondent instance.

NA refers for 'data not accessible in literature.'

Table 5: The best results were achieved by the proposed HFPA for the first set of benchmark instances against the state-of-the-art algorithm.

| Seq. | Length | $E^*$ | MMA | IMOG | HFPA |
|---|---|---|---|---|---|
| $B_1$ | 12 | **-11** | NA | **-11** | **-11** |
| $B_2$ | 14 | **-11** | **-11** | **-11** | **-11** |
| $B_3$ | 14 | **-11** | **-11** | **-11** | **-11** |
| $B_4$ | 16 | **-11** | **-11** | **-11** | **-11** |
| $B_5$ | 16 | **-11** | **-11** | **-11** | **-11** |
| $B_6$ | 17 | **-11** | **-11** | **-11** | **-11** |
| $B_7$ | 17 | **-17** | **-17** | **-17** | **-17** |
| $B_8$ | 20 | **-17** | **-17** | **-17** | **-17** |
| $B_9$ | 20 | **-17** | **-17** | **-17** | **-17** |
| $B_{10}$ | 21 | **-17** | **-17** | **-17** | **-17** |
| $B_{11}$ | 21 | **-17** | **-17** | **-17** | **-17** |
| $B_{12}$ | 21 | **-17** | **-17** | **-17** | **-17** |
| $B_{13}$ | 22 | **-17** | **-17** | **-17** | **-17** |
| $B_{14}$ | 23 | **-25** | **-25** | **-25** | **-25** |
| $B_{15}$ | 24 | **-17** | -16 | **-17** | **-17** |
| $B_{16}$ | 24 | **-25** | **-25** | **-25** | **-25** |
| $B_{17}$ | 24 | **-25** | **-25** | **-25** | **-25** |
| $B_{18}$ | 30 | **-25** | -24 | **-25** | **-25** |
| $B_{19}$ | 30 | **-25** | -24 | **-25** | **-25** |
| $B_{20}$ | 37 | **-29** | -26 | **-29** | **-29** |

Table 6: The best results were achieved by the proposed HFPA for the second set of benchmark instances against the state-of-the-art algorithms.

| | | ERS-GA | | HHGA | | IMOG | | GALSTS | | HFPA | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Seq. | Length | Best | Mean | Best | Mean | Best | Mean | Best | Mean | Best | Mean |
| $A_1$ | 20 | **-15** | -12.50 | **-15** | -14.73 | **-15** | -14.73 | **-15** | -14.86 | **-15** | **-15** |
| $A_2$ | 24 | -13 | -10.20 | **-17** | -14.93 | **-17** | -14.93 | **-17** | -15.53 | **-17** | **-17** |
| $A_3$ | 25 | -12 | -8.47 | **-12** | - 11.57 | **-12** | -11.57 | **-12** | **-12** | **-12** | **-12** |
| $A_4$ | 36 | -20 | -16.17 | -23 | -21.27 | -23 | -21.27 | **-24** | -21.93 | **-24** | **-23.85** |
| $A_5$ | 48 | -32 | -28.13 | -41 | - 37.30 | -41 | -37.30 | **-43** | -39.86 | **-43** | **-42.10** |
| $A_6$ | 50 | -30 | -25.30 | -38 | -34.10 | -38 | -34.10 | -40 | -37.6 | **-40** | **-38.60** |
| $A_7$ | 60 | -55 | -49.43 | -66 | - 61.83 | -66 | -61.83 | -70 | -68.26 | **-71** | **-69.10** |
| $A_8$ | 64 | -47 | -42.37 | -63 | - 56.53 | -63 | -56.53 | -67 | -58.46 | **-73** | **-69.43** |

Table 7: A comparative study on the stability and best prediction of the suggested HFPA against the most efficient state-of-the-art algorithms.

| | | | GA | | | FPA | | | HFPA | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Seq. | Lenght | $E^*$ | Best | Worst | Mean | Best | Worst | Mean | Best | Worst | Mean |
| $A_1$ | 20 | **-15** | **-15** | -10 | -11.70 | **-15** | **-15** | **-15** | **-15** | **-15** | **-15** |
| $A_2$ | 24 | **-17** | -16 | -12 | -14.50 | **-17** | -14 | -15.65 | **-17** | **-17** | **-17** |
| $A_3$ | 25 | **-12** | **-12** | -9 | -10.50 | **-12** | -11 | -11.55 | **-12** | **-12** | **-12** |
| $A_4$ | 36 | **-24** | -19 | -16 | -18.20 | -23 | -20 | -21.35 | **-24** | **-23** | **-23.80** |
| $A_5$ | 48 | **-43** | -35 | -31 | -32.70 | -41 | -36 | -38 | **-43** | **-41** | **-42.55** |
| $A_6$ | 50 | **-40** | -36 | -31 | -33.65 | -38 | -34 | -35.80 | **-40** | **-39** | **-39.75** |
| $A_7$ | 60 | **NA** | -61 | 52 | -55.55 | -67 | -61 | -63.35 | **-71** | **-68** | **-70.20** |
| $A_8$ | 64 | **NA** | -60 | -50 | -56.85 | -63 | -54 | -59.05 | **-73** | **-68** | **-71.10** |

Table 8: Comparison of statistical results. The best, worst and average results obtained by HFPA are compared to those obtained by FA, and GA. $E^*$ is the best energy value.

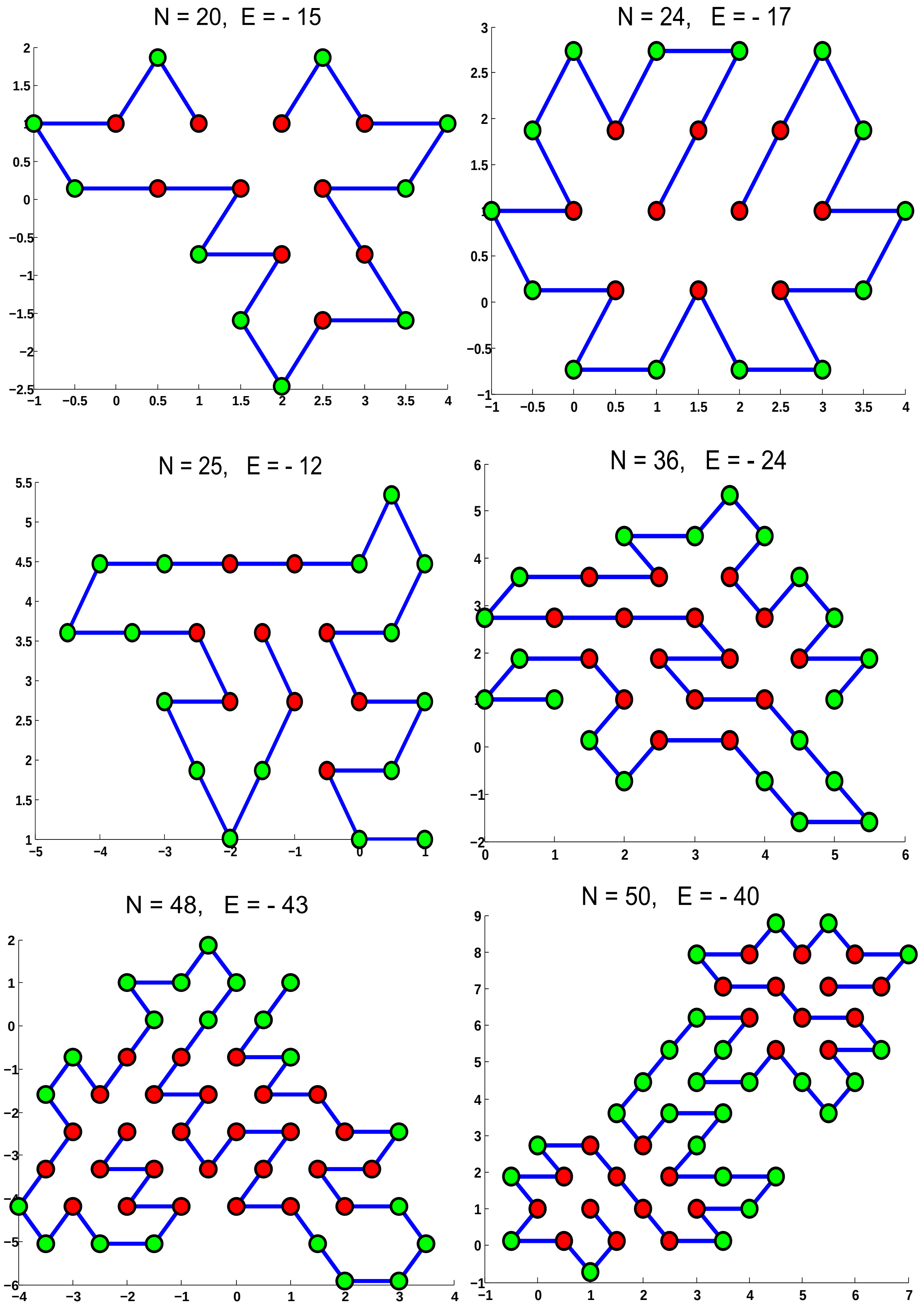


Figure 7: The conformations obtained by HFPA for instances from $A_1$ to $A_6$.

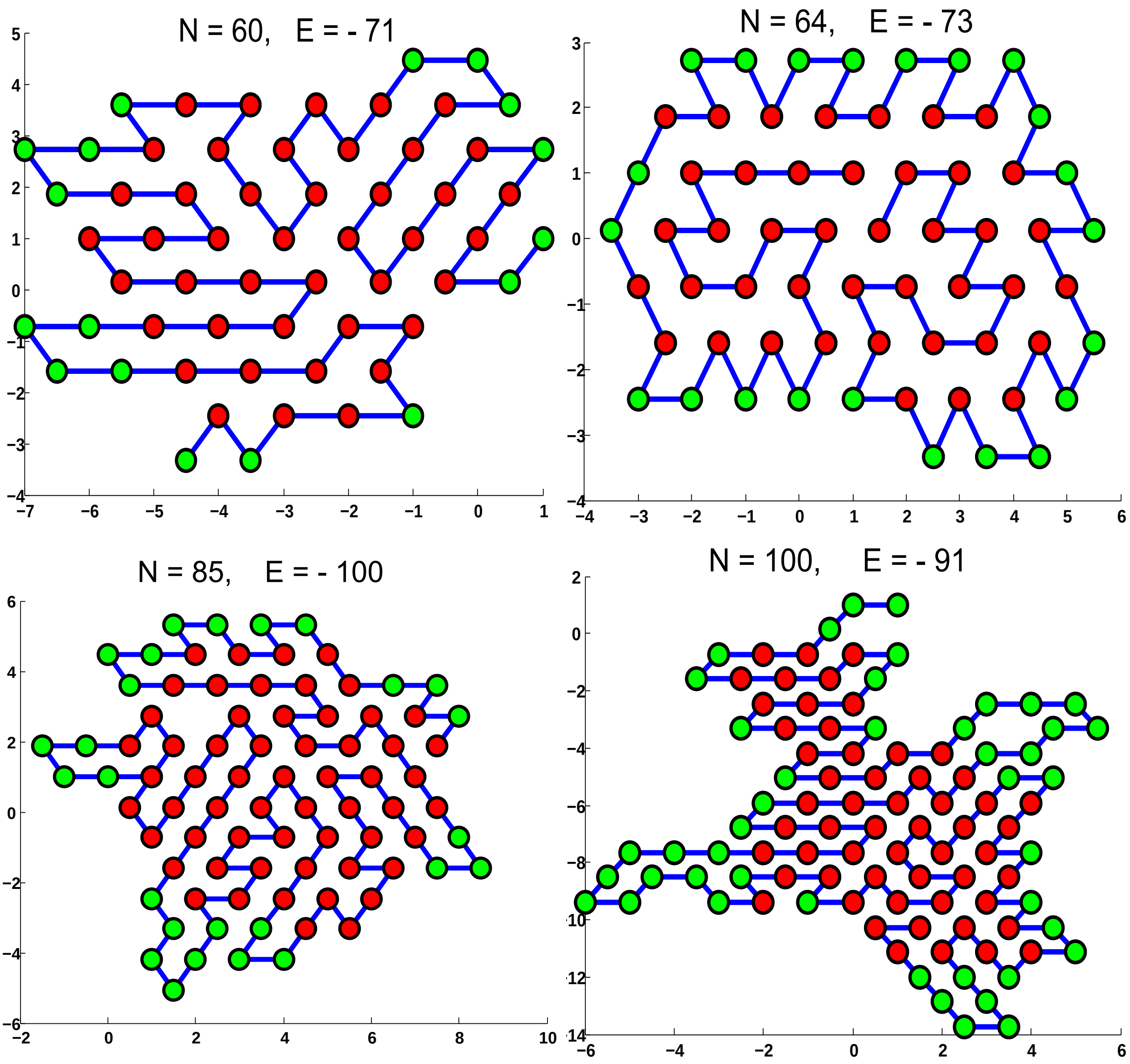


Figure 8: The conformations obtained by HFPA for instances from $A_7$ to $A_{10}$.

# 6 Conclusion

Predicting the native structure of a protein from its amino acid sequence is one of the most challenging optimization problems in biology and bioinformatics. The difficulty of efficiently solving this problem arises from the enormous solution space, even for a sequence with a small number of amino acids. Although simplified models like Dill's Hydrophobic-Polar (HP) model have been proposed to reduce the conformational space, the Protein Structure Prediction (PSP) problem remains an NP-hard problem. In recent years, many evolutionary algorithms and metaheuristics have been used to address the PSP problem under the HP simplified model. The flexible balancing of exploration and exploitation of solutions is one of the most essential components of metaheuristics, and it plays a key role in their effectiveness. In this work, we proposed a discrete hybrid ower pollination algorithm that combines classical FPA with genetic and tabu search algorithms. We chose the FPA algorithm due to its demonstrated high exploration in solving various optimization problems. To enhance the PFA's capacity in the intensification phase, we integrated the crossover operator of GA and the Tabu Search algorithm, both of which are very efficient in exploiting the solution search space. The simulation results showed that our algorithm successfully predicted the optimal structure for several instances of PSP under the HP model in the 2D triangular lattice. Furthermore, the proposed HFPA algorithm outperformed existing state-of-the-art algorithms. In future work, we aim to use the proposed algorithm to solve other discrete optimization problems. Additionally, we can develop other hybrid metaheuristics based on FPA for global optimization problems, such as combining FPA with Particle Swarm Optimization (PSO).

# References

[1] Eric T. Baldwin, Irene T. Weber, Robert St Charles, Jian-Cheng Xuan, Ettore Appella, Masaki Yamada, Kouji Matsushima, B. F. Edwards, G. Marius Clore, and Angela M. Gronenborn. Crystal structure of interleukin 8: symbiosis of NMR and crystallography. *Proceedings of the National Academy of Sciences*, 88(2):502–506, 1991. Publisher: National Acad Sciences.

[2] Hans-Joachim Bockenhauer, Abu Zafer M. Dayem Ullah, Leonidas Kapsokalivas, and Kathleen Steinhofel. A local move set for protein folding in triangular lattice models. In *International Workshop on Algorithms in Bioinformatics*, pages 369–381. Springer, 2008.

[3] Nabil Boumedine and Sadek Bouroubi. An Improved Simulated Annealing Algorithm for Optimization of Protein Folding Problem. In *2020 2nd International Workshop on Human-Centric Smart Environments for Health and Well-being (IHSH)*, pages 246–251. IEEE, 2021.

[4] Nabil Boumedine and Sadek Bouroubi. Protein folding in 3D lattice HP model using a combining cuckoo search with the Hill-Climbing algorithms. *Applied Soft Computing*, 119:108564, 2022. Publisher: Elsevier.

[5] Dwaipayan Chakraborty, Sankhadip Saha, and Samaresh Maity. Training feedforward neural networks using hybrid flower pollination-gravitational search algorithm. In *2015 international conference on futuristic trends on computational analysis and knowledge management (ABLAZE)*, pages 261–266. IEEE, 2015.

[6] Thomas Dandekar and Patrick Argos. Folding the main chain of small proteins with the genetic algorithm. *Journal of Molecular Biology*, 236(3):844–861, 1994. Publisher: Elsevier.

[7] S. Decatur and Serafim Batzoglou. Protein folding in the Hydrophobic-Polar model on the 3D triangular lattice. In *6th Annual MIT Laboratory for Computer Science Student Workshop on Computing Technologies*, 1996.

[8] Ken A. Dill. Dominant forces in protein folding. *Biochemistry*, 29(31):7133–7155, 1990. Publisher: ACS Publications.

[9] Ken A. Dill and Justin L. MacCallum. The protein-folding problem, 50 years on. *science*, 338(6110):1042–1046, 2012. Publisher: American Association for the Advancement of Science.

[10] Ivan Dotu, Manuel Cebrian, Pascal Van Hentenryck, and Peter Clote. On lattice protein structure prediction revisited. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(6):1620–1632, 2011. Publisher: IEEE.

[11] Fred Glover. Tabu search part-I. *ORSA Journal on computing*, 1(3):190–206, 1989. Publisher: Informs.

[12] Fred Glover and Manuel Laguna. Tabu search. In *Handbook of combinatorial optimization*, pages 2093–2229. Springer, 1998.

[13] Yuzhen Guo, Zikai Wu, Ying Wang, and Yong Wang. Extended particle swarm optimisation method for folding protein on triangular lattice. *IET systems biology*, 10(1):30–33, 2016. Publisher: IET.

[14] Faten Hajjej, Ridha Ejbali, and Mourad Zaied. An efficient deployment approach for improved coverage in wireless sensor networks based on flower pollination algorithm. *NETCOM, NCS, WiMoNe, GRAPH-HOC, SPM, CSEIT*, pages 117–129, 2016.

[15] John Hardy. Alzheimer's disease: the amyloid cascade hypothesis: an update and reappraisal. *Journal of Alzheimer's disease*, 9(s3):151–153, 2006. Publisher: IOS press.

[16] Md Tamjidul Hoque, Madhu Chetty, and Laurence S. Dooley. A hybrid genetic algorithm for 2D FCC hydrophobic-hydrophilic lattice model to predict protein folding. In *Australasian Joint Conference on Artificial Intelligence*, pages 867–876. Springer, 2006.

[17] Md Kamrul Islam and Madhu Chetty. Clustered memetic algorithm with local heuristics for ab initio protein structure prediction. *IEEE Transactions on Evolutionary Computation*, 17(4):558–576, 2012. Publisher: IEEE.

[18] Prince Jain, Shonak Bansal, Arun Kumar Singh, and Neena Gupta. Golomb ruler sequences optimization for FWM crosstalk reduction: multi-population hybrid flower pollination algorithm. In *Progress in electromagnetics research symposium (PIERS), Prague, Czech Republic*, pages 2463–2467, 2015.

[19] R. Jensi and G. Wiselin Jiji. Hybrid data clustering approach using k-means and flower pollination algorithm. *arXiv preprint arXiv:1505.03236*, 2015.

[20] Holland John. Holland. genetic algorithms. *Scientific american*, 267(1):44–50, 1992.

[21] Martin Karplus. The Levinthal paradox: yesterday and today. *Folding and design*, 2:S69–S75, 1997. Publisher: Elsevier.

[22] Natalio Krasnogor, B. P. Blackburne, Edmund K. Burke, and Jonathan D. Hirst. Multimeme algorithms for protein structure prediction. In *International Conference on Parallel Problem Solving from Nature*, pages 769–778. Springer, 2002.

[23] Sahil Lalljith, Ismail Fleming, Umeshan Pillay, Kiveshen Naicker, Zachary Naidoo, and Akshay Kumar Saha. Applications of Flower Pollination Algorithm in Electrical Power Systems: A Review. *IEEE Access*, 2021. Publisher: IEEE.

[24] Kit Fun Lau and Ken A. Dill. A lattice statistical mechanics model of the conformational and sequence spaces of proteins. *Macromolecules*, 22(10):3986–3997, 1989. Publisher: ACS Publications.

[25] Kanagasabai Lenin and Bhumanapally RavindhranathReddy. Reduction of real power loss by using Fusion of Flower Pollination Algorithm with Particle Swarm Optimization. *Journal of the Institute of Industrial Applications Engineers Vol*, 2(3):97–103, 2014.

[26] Cheng-Jian Lin and Ming-Hua Hsieh. An efficient hybrid Taguchi-genetic algorithm for protein folding simulation. *Expert systems with applications*, 36(10):12446–12453, 2009. Publisher: Elsevier.

[27] Szymon \Lukasik and Piotr A. Kowalski. Study of flower pollination algorithm for continuous optimization. In *Intelligent Systems' 2014*, pages 451–459. Springer, 2015.

[28] Ong Kok Meng, Ong Pauline, Sia Chee Kiong, Hanani Abdul Wahab, and Noormaziah Jafferi. Application of modified flower pollination algorithm on mechanical engineering design problem. In *IOP conference series: materials science and engineering*, volume 165, page 012032. IOP Publishing, 2017. Issue: 1.

[29] Boumedine Nabil and Bouroubi Sadek. Protein structure prediction in the HP model using scatter search algorithm. In *2020 4th International Symposium on Informatics and its Applications (ISIA)*, pages 1–5. IEEE, 2020.

[30] Sinan Melih Nigdeli, G. Bekdaş, and X. S. Yang. Optimum tuning of mass dampers for seismic structures using flower pollination algorithm. *Int. J. Theor. Appl. Mech*, 1:264–268, 2016.

[31] Sinan Melih Nigdeli, Gebrail Bekdaş, and Xin-She Yang. Optimum tuning of mass dampers by using a hybrid method using harmony search and flower pollination algorithm. In *International Conference on Harmony Search Algorithm*, pages 222–231. Springer, 2017.

[32] Brian K. Nunnally and Ira S. Krull. *Prions and mad cow disease.* CRC Press, 2003.

[33] Robert L. Nussbaum and Christopher E. Ellis. Alzheimer's disease and Parkinson's disease. *New england journal of medicine*, 348(14):1356–1364, 2003. Publisher: Mass Medical Soc.

[34] Ilya Pavlyukevich. Lévy flights, non-local search and simulated annealing. *journal of computational physics*, 226(2):1830–1844, 2007. Publisher: Elsevier.

[35] Stanley B. Prusiner. The prion diseases. *Scientific American*, 272(1):48–57, 1995. Publisher: JSTOR.

[36] Meera Ramadas, Ajith Abraham, and Sushil Kumar. Using data clustering on ssFPA/DE-a search strategy flower pollination algorithm with differential evolution. In *International Conference on Hybrid Intelligent Systems*, pages 539–550. Springer, 2016.

[37] Ahmed K. Ryad, Ahmed M. Atallah, and Abdelhaliem Zekry. Photovoltaic parameters estimation using hybrid flower pollination with clonal selection algorithm. *Turkish J. Electromechanics Energy*, 3(2):15–21, 2018.

[38] BOUROUBI SADEK and NABIL BOUMEDINE. A new hybrid genetic algorithm for protein structure prediction on the 2Dtriangular lattice. *Turkish Journal of Electrical Engineering & Computer Sciences*, 29(2):499–513, 2021. Publisher: The Scientific and Technological Research Council of Turkey.

[39] BOUROUBI SADEK and Nabil Boumedine. A new hybrid genetic algorithm for protein structure prediction on the 2Dtriangular lattice. *Turkish Journal of Electrical Engineering & Computer Sciences*, 29(2):499–513, 2021. Publisher: The Scientific and Technological Research Council of Turkey.

[40] BOUROUBI SADEK and Nabil Boumedine. A new hybrid genetic algorithm for protein structure prediction on the 2Dtriangular lattice. *Turkish Journal of Electrical Engineering & Computer Sciences*, 29(2):499–513, 2021. Publisher: The Scientific and Technological Research Council of Turkey.

[41] S. Sakthivel, P. Manopriya, S. Venus, S. Ranjitha, and R. Subhashini. Optimal reactive power dispatch problem solved by using flower pollination algorithm. *International Journal of Applied Engineering Research*, 11(6):4387–4391, 2016.

[42] Rohit Salgotra and Urvinder Singh. A novel bat flower pollination algorithm for synthesis of linear antenna arrays. *Neural Computing and Applications*, 30(7):2269–2282, 2018. Publisher: Springer.

[43] Prerna Saxena and Ashwin Kothari. Linear antenna array optimization using flower pollination algorithm. *SpringerPlus*, 5(1):1–15, 2016. Publisher: Springer.

[44] Marwa Sharawi, E. Emary, Imane Aly Saroit, and Hesham El-Mahdy. Flower pollination optimization algorithm for wireless sensor network lifetime global optimization. *International Journal of Soft Computing and Engineering*, 4(3):54–59, 2014. Publisher: Citeseer.

[45] C. Shilaja and K. Ravi. Optimal line flow in conventional power system using euclidean affine flower pollination algorithm. *Int. J. Renew. Energy Res. C*, 6(1), 2016.

[46] C. Shilaja and K. Ravi. Multi-objective optimal power flow problem using enhanced flower pollination algorithm. *Gazi University Journal of Science*, 30(1):79–91, 2017.

[47] J. E. Smith. The co-evolution of memetic algorithms for protein structure prediction. In *Recent advances in memetic algorithms*, pages 105–128. Springer, 2005.

[48] Shih-Chieh Su, Cheng-Jian Lin, and Chuan-Kang Ting. An efficient hybrid of hill-climbing and genetic algorithm for 2D triangular protein structure prediction. In *2010 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW)*, pages 51–56. IEEE, 2010.

[49] Douglas H. Turner, Naoki Sugimoto, and Susan M. Freier. RNA structure prediction. *Annual review of biophysics and biophysical chemistry*, 17(1):167–192, 1988. Publisher: Annual Reviews 4139 El Camino Way, PO Box 10139, Palo Alto, CA 94303-0139, USA.

[50] Ron Unger and John Moult. Genetic algorithms for protein folding simulations. *Journal of molecular biology*, 231(1):75–81, 1993. Publisher: Elsevier.

[51] Rui Wang, Yongquan Zhou, Chengyan Zhao, and Haizhou Wu. A hybrid flower pollination algorithm based modified randomized location for multi-threshold medical image segmentation. *Bio-medical materials and engineering*, 26(s1):S1345–S1351, 2015. Publisher: IOS Press.

[52] Shuhui Xu and Yong Wang. Parameter estimation of photovoltaic modules using a hybrid flower pollination algorithm. *Energy Conversion and Management*, 144:53–68, 2017. Publisher: Elsevier.

[53] Cheng-Hong Yang, Kuo-Chuan Wu, Yu-Shiun Lin, Li-Yeh Chuang, and Hsueh-Wei Chang. Protein folding prediction in the HP model using ions motion optimization with a greedy algorithm. *BioData mining*, 11(1):1–14, 2018. Publisher: Springer.

[54] Xin-She Yang. Flower pollination algorithm for global optimization. In *International conference on unconventional computing and natural computation*, pages 240–249. Springer, 2012.

[55] Xin-She Yang, Mehmet Karamanoglu, and Xingshi He. Flower pollination algorithm: a novel approach for multiobjective optimization. *Engineering optimization*, 46(9):1222–1237, 2014. Publisher: Taylor & Francis.