PSPGA: A New Method for Protein Structure Prediction based on Genetic Algorithm

Arash Mazidi[1*], Fahimeh Roshanfar[2]

Abstract – Bioinformatics is a new science that uses algorithms, computer software and databases in order to solve biological problems, especially in the cellular and molecular areas. Bioinformatics is defined as the application of tools of computation and analysis to the capture and interpretation of biological data. Protein Structure Prediction (PSP) is one of the most complex and important issues in bioinformatics, and extensive researches has been done to solve this problem using evolutionary algorithms. In this paper, we propose a genetic based method in order to solve protein structure prediction problem with increasing the accuracy of prediction, using a crossover operator based on pattern mask. Further, we compare two genetic based method to evaluate the proposed method. The results of the implementation of our proposed algorithm on five standard test sequences show that the use of a pattern mask-based crossover operator in the genetic algorithm can significantly improve the accuracy compared to previous similar algorithms.

Keywords: Protein Structure Prediction, Evolutionary Algorithm, Genetic Algorithm, HP Model.

1. Introduction

Data collection is one of the basic steps in any scientific research. Biological researchers, like other sciences, first collect biological data. In the past, this data was extracted with the help of simple laboratory facilities, so the data volume was generally small and easily manageable. With the advancement of technology and the introduction of new laboratory facilities in the field of biology, the volume of biological data increased rapidly, to the extent that other old methods for managing this data and process them were not enough[1]. The use of computers to process biological data led to the birth of a new science called bioinformatics. Bioinformatics is the knowledge of using computer science, statistics and probabilities in the field of molecular biology[2].

One of the main goals in bioinformatics is the analysis of biological data. For example, data analysis and protein structure, comparing the sequence of one protein with other proteins, and predicting protein structure are applications of bioinformatics. The use of protein molecules depends on their spatial shape and three-dimensional structure. Genes play a role in the function of the proteins they make. Therefore, complete knowledge of genes requires complete knowledge of proteins. Various models and methods are used to predict the structure of proteins, an example of these methods is the use of network models and exploratory algorithms[2][3][4].

For the first time, Berger et al. [5]introduced the problem of predicting protein structure as a NP-complete problem. Therefore, to solve these problems, due to the time-consuming methods, indefinite methods are used. In their paper, Unger et al. [6] used a genetic algorithm using the Monte Carlo mutation and crossover to reduce the energy of the sequences. Lau et al. [7] proposed a network-based model for representing protein structure that is called HP model. Tantar et al. [8] presented a parallel method and a combination of genetic algorithms with hill climbing search method as local search in order to solve the problem of predicting protein structure. Cutello et al. [9] proposed a new algorithm for predicting protein structure using a genetic algorithm. The new mutation operators reproduced the new populations. It has been tested on various models such as the HP model and the results show the good efficiency of the proposed algorithm compared to the others. Unger et al. [10] examined the benefits of using a genetic algorithm to find the three-dimensional structure of proteins in their linear order and the challenges in this area.

As it was observed, in most of the researches done to

[1*] **Corresponding Author** : Department of Computer Engineering, Faculty of Engineering, Golestan University, Gorgan, Iran.
E-Mail: arash_mazidi_67@yahoo.com
[2] Department of Nanotechnology and Advanced Materials, Materials and Energy Research Center, Karaj, Iran.

Received 2020.01.25 ; Accepted 2020.05.10

solve the problem of protein structure prediction, genetic algorithm has been used and we intend to improve the genetic algorithm. In this paper, we predict the protein structure based on the HP lattice model using a pattern mask-based crossover operator in the genetic algorithm. The results of our experiments show that the crossover operator improves the detection of protein structure.

In the rest of this paper, Section 2 reviews the basic concepts of the problem. Section 3 describes the proposed algorithm; Section 4 discusses the implementation and evaluation of proposed algorithm. In the end, Section 5 concludes the paper and provides suggestions for future works.

2. HP Model and Genetic Algorithm

2.1. HP Model

Protein is a biomolecule made up of amino acid sequences that plays a key role in many cellular functions. During protein synthesis, amino acids join together from the endpoints to form a peptide bond, forming a sequence of these peptide bonds in the protein backbone [11].Unlike the structure of other biomolecules, proteins have a complex and irregular structure. The functional properties of a protein depend on its three-dimensional structure, so predicting the structure of a protein is an important challenge in molecular biology.

Many computational techniques have been developed to predict protein structure, but few of these methods are accurate. Many of these methods use counting techniques or search strategies to predict the structure of a protein, which itself requires the evaluation and study of the structure of thousands of proteins, so researchers are looking for methods with lower cost.

Network models are very useful tools for predicting the structure of proteins, because they ignore atomic details and therefore rely on the extraction of basic principles, which leads to model and predict in less time. One of the most popular network models is the HP model which introduced by Dale in 1989[7].

In the HP model, all 20 types of amino acids are divided into two separate groups based on hydrophobicity and hydrophobicity of amino acids. The hydrophobic group is indicated by the symbol H in the form of a black circle and the hydrophilic group is represented by the symbol P in the form of a white circle. According to this property, hydrophilic amino acids form the outer surface of the protein and hydrophobic amino acids form the hydrophobic core of the protein.

This two-dimensional lattice model shows the protein amino acid sequence as a flat, tortuous path with no overlap. This property is called self-avoidance of overlap, and the existence of this property causes the structural protein to find a minimum amount of energy [12]. The characteristics of the HP model used to show the structure of the protein are:

- Amino acid sequences: Two amino acids with a covalent bond between them will be consecutive. When a bond is formed between two non-sequential H amino acids, this bond reduces the free energy of the resulting molecule. Thus, protein reaches its lowest energy level when the number of non-sequential bonds of type H amino acids is maximized[13].
- Proximity: Two H amino acids is also due to the adjacent H-H bond.
- Establishment: H amino acids are concentrated in the center and the P amino acids surround them.
- Topology of the Graph: Each compound of amino acids is a graph whose vertices represent the amino acids and its edges represent the peptide bonds (covalent).
- Square model: In a two-dimensional square model, all amino acids except the first and last have two covalent bonds and a maximum of two topological bonds.

2.2.Genetic Algorithm

Genetic algorithm is an evolutionary algorithm and a statistical method for optimization. The main idea of the genetic algorithm is the transfer of inherited traits by genes. The genetic algorithm were introduced at the University of Michigan in 1962, and then in 1975, its mathematical foundations were published in a book entitled "Adaptation in Natural and Artificial Systems"[14].

An important feature of the genetic algorithm is its robustness, in which there is a flexible balance between performance and characteristics necessary for survival in a variety of environments and conditions. In fact, if the compatibility increases, that system will be able to operate for a longer period of time and in a more favorable manner. The genetic algorithm starts with a set of solutions that are represented by chromosomes. This set of solutions is called the initial population. In this algorithm, the solutions obtained from one population are used to generate the next population. In this process, it is hoped that the new population will be better than the previous population. The selection of some solutions from the total solutions (parents) in order to create new solutions(children) based on their desirability or fitness. It is natural that more appropriate solutions have a better chance of reproducing. This process continues until a predetermined condition is met. In the genetic algorithm, there are basic concepts, which are described following:

Objective function: The objective function gives us an indicator of how people perform in the problem space. In fact, the function we want to optimize is known as the objective function.

Fitness Function: The fitness function is used to convert the values of the objective function into a scale for the relative compatibility and efficiency of individuals.
Chromosome: A chromosome is a set of genes or the same characteristics. A chromosome is the solution for the problem.

Population: A population is a set of chromosomes, or sets of solutions. In a genetic algorithm, the number of people in a population is usually constant. The whole effort of the genetic algorithm is to increase the competence of the population according to the limitations of the problem.

Search space: The space of all acceptable solutions is the search space. Each point in the search space indicates an acceptable solution.

Operators: In the genetic algorithm, there are three main operators, selection, crossover and mutation operators.

The crossover operator, with a probability that is often between 0.65 and 0.7, acts on the parent chromosomes and combines them to produce new chromosomes, or child. The mutation operator, with a probability of less than 0.05, is performed on chromosomes derived from crossover operator, randomly selects a gene from a chromosome, and then changes the content of that gene. It should be noted that a mutation is a single-operative actuator and is applied to a chromosome or part of a gene. Figure 1 shows an overview of the steps of the genetic algorithm.

1. Random production of a population that includes a certain number of chromosomes.

2. Evaluates the fitness function for each chromosome in the population.

3. Creating a new population based on the repetition of the following steps:

3.1. Selection of two parents from the population according to selection operator

3.2. Use crossover operator and creation new generation.

3.3. Use mutation and change in some of the offspring genes

3.4. Replacing new children in a new population.

4. Using the new population for future implementations of the algorithm.

5. Stop running the algorithm if you see the termination condition and announce the best answer, otherwise return to the second stage.

Figure 1: Genetic algorithm

3. Proposed Algorithm

In this section, we will implement three methods of genetic algorithm. A typical genetic algorithm is a basic algorithm of genetic that has been implemented to compare with other methods [6]. Another method is the genetic algorithm with depth first search, which was first presented in 2010 [15] and implemented in this section, and the results were obtained. The main one presented in the paper is implemented and the results of this algorithm are compared with the results of the previous two algorithms.

3.1. Basic Genetic Algorithm

To implement a typical genetic algorithm, the various parameters are examined and the operations required for implementation are as follows.

- **Inputs**

The input of this algorithm is strings of letters P and H that represent amino acid sequences. The letter P stands for hydrophilic amino acids and the letter H stands for hydrophobic amino acids, denoted by a white circle and a black circle, respectively. For example, the PHHHPHPPH input string is shown in Figure 2.

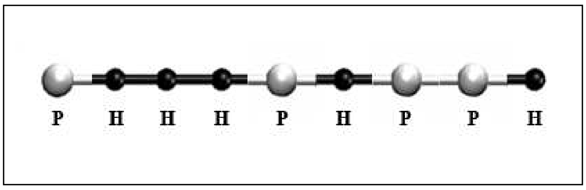


Figure 2: HP model of PHHHPHPPH sequence

By receiving the input string, the algorithm tries to find a structure that has the lowest energy level for the desired sequence[16].

- **Coding of chromosomes**

To encode chromosomes in this problem, there are two methods of independent encoding and dependent encoding which are explained below.

Independent coding: This method uses four letters R, L, U and D to represent the path, which represent right, left, up and down, respectively. The remarkable point in this method is that each letter represents a fixed and definite movement and has no dependence on the letters before and after it. For example, the RURULLDL code is for the input string shown in Figure 3.

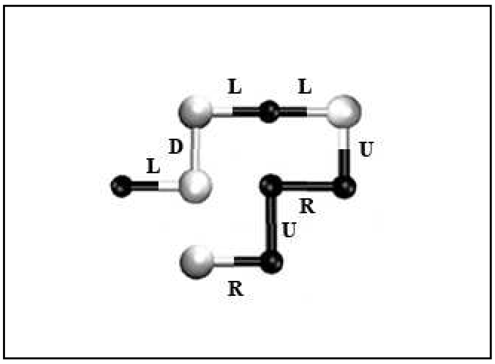


Figure 3: Independent coding for the PHHHPHPPH input string

Dependent coding: In this method, three letters F, L and R are used to display the path, which indicate the continuation of movement in the opposite path, change of direction to the left of the path and change of direction to the right of the path, respectively. In this method, the direction of movement of each letter is not clear and fixed and is determined according to the path in which it is located. For example, the FLRLLFLR code is for the input string is shown in Figure 4.

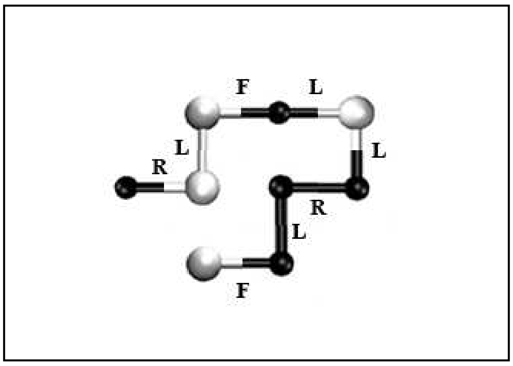


Figure 4: Dependent coding for the PHHHPHPPH input string

The dependent coding method has a high degree of flexibility and to change the structure, changing one letter is enough, but in another method of applying the change requires many changes in the original code, which is very time consuming. Therefore, the dependent coding method is used to encode chromosomes.

- **Initial population**

In the genetic algorithm, the initial population is generated randomly, but in the protein structure prediction problem, the solutions should be self-avoidance of overlap.

- **Fitness function**

Proteins are placed in space to have the lowest possible energy level. The protein reaches its lowest energy level when the number of non-sequential bonds of type H amino acids is maximized. Therefore, the goal can be expressed as minimizing the fitness function, which in fact represents the maximum number of non-sequential amino acid bonds of type H. it is represented in Equation 1.

$$E(c_i)=\sum_{j=1}^{n-1}\sum_{k=j+1}^{n}N_{jk} \qquad where \qquad N_{jk}=\begin{cases}-1\\0\end{cases} \tag{1}$$

j and k are amino acid bonds of type H
O.W

- **Selection operator**

Most algorithms in this category use a roulette wheel for the selection operator. In the roulette wheel selection, each chromosome is first assigned a selection probability according to its fitness.

- **Crossover operator**

The implementation of the crossover operator is single-point, in which two selected chromosomes are randomly cut from a single point and cross-linked. Furthermore, multi-point crossover were proposed, but not much improvement was achieved over single-point crossover. Figure 5 shows a single-point on a chromosome of two desired chromosomes.

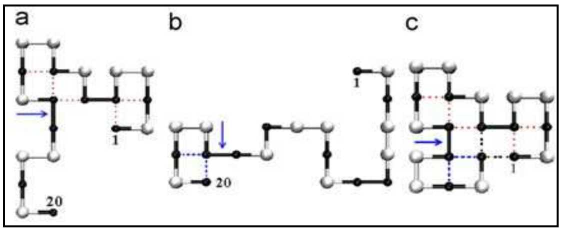


Figure 5: crossover operation on two chromosomes a and b and the resulting chromosome c

- **Mutation operator**

To implement the mutation, it is sufficient to randomly select one of the genes on the chromosome and transfer the protein from that point. You can see this in Figure 6.

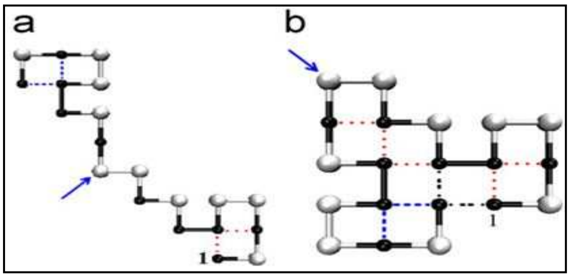


Figure 6: Action of a mutation on chromosome a resulting chromosome b

3.2. Genetic algorithm with depth first search

In 2010, a method for constructing the initial population and the crossover operator in the genetic algorithm was used, this method reduces the complexity of exponentially constructing the initial population randomly and doing so in a linear time. Also, the integration of crossover with the depth first search results in the formation of novel chromosomes that would not normally be available at random. In this method, the fitness function, the selection operator, and the mutation operator are similar to the basic genetic algorithm, so in the following we will only explain the construction of the initial population and the crossover that is different from basic genetic algorithm.

- **Initial population**

In the basic genetic algorithm, self-avoidance is checked to avoid overlapping of the path produced at the end of chromosome production, but in this algorithm, this property is checked at each stage of chromosome production, and in the event of a collision, one step is reversed. Then, we continue the path from another direction.

For a better understanding of the subject, see Figure 7. In this figure, the steps of making a chromosome are shown randomly, no collision has occurred until stage 5, but at this stage the sequence collides, which invalidates the sequence. Unlike a basic genetic algorithm, instead of making the chromosome from the first, one step is reversed and another is selected to make the sequence valid. It is clear that the cost of making chromosomes with this method is much lower than basic genetic algorithm.

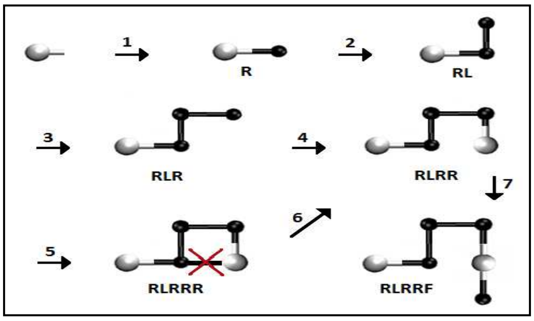


Figure 7: Steps of making a chromosome by the depth first search method

- **Crossover operator**

To implement the crossover operator, first, the crossover operator is performed as a single point. If the chromosome from the crossover was valid, the job is done; otherwise, two chromosomes from one point is cut randomly and the continuation of the sequence path is generated by the depth first search method.

3.3. Genetic algorithm with crossover operator based on pattern mask

As mentioned different methods are proposed for the crossover operator in the genetic algorithm. In this section we try to implement a genetic algorithm with a crossover operator based on a pattern mask. In this method, first the chromosomes of a population are arranged according to the value of their fitness function. If the population has N solutions, the number of N/4 chromosomes that have the highest fitness and the same number of chromosomes that have the lowest fitness are selected to produce positive and negative masks, respectively. The main idea of this method is that first group of chromosomes are expected to have better genes and characteristics second group of chromosomes.

By statistically extracting the information of chromosomes with high and low finesses, positive mask and negative masks are produced, respectively. The maximum voting method is used to prepare positive and negative masks.

Depending on how the positive and negative masks are formed, it is expected that the positive mask contains the superior characteristics and the negative mask contains the weak characteristics. To prepare a pattern mask, if the corresponding bits in the positive and negative masks are different from each other, the value of their corresponding bits in the pattern mask is the same as the positive mask bit, otherwise it is considered insignificant. In this method, after the pattern mask is made for a population, the crossover operator is performed with the participation of the parent chromosomes and the pattern mask.

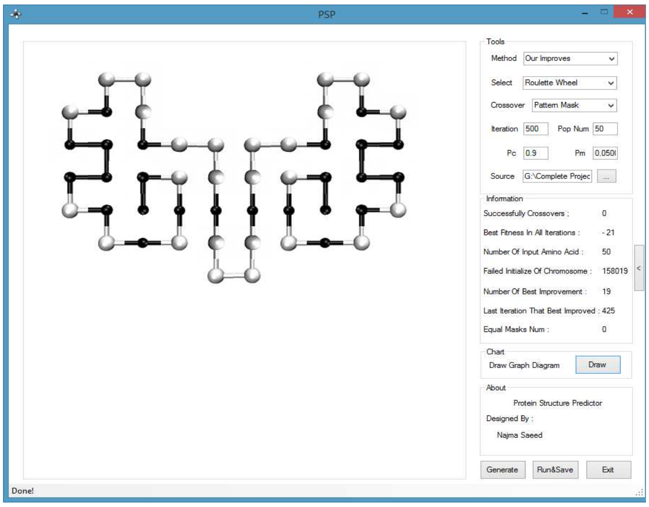
4. Implementation and Evaluation

All algorithms are implemented in C# programming language and the results of running the algorithms on the test data shown in Table 1 are compared in Table 2. The results obtained for different algorithms are the average of 10 times on standard test data.

Table 1: Test data[17][18]

Test No.	*Sequence length*	*Sequence*
1	50	H2(PH)3PH4PH(P3H)2P4 H(P3H)2PH4P(HP)3H2
2	60	P2H3PH8P3H10PHP3H12 P4H6PH2PHP
3	64	H12(PH)2(P2H2)2P2HP2 H2PPHP2H2P2(H2P2)2(HP)2 H12
4	85	H4P4H12P6(H12P3)3HP2 (H2P2)2HPH
5	100	P3H2P2H4P2H3(PH2)2P H4P8H6P2H6P9HPH2PH11P 2H3PH2PHP2HPH3P6H3

Figure 8 shows the software user interface designed to implement the algorithms. Figure 8 shows an image of running software for predicting protein structure with a sequence length of 50 in the genetic algorithm with the crossover pattern mask operator.



Method: Our Improves
Select: Roulette Wheel
Crossover: Pattern Mask
Iteration: 500 Pop Num: 50
Pc: 0.9 Pm: 0.050(
Source: G:\Complete Projec

Successfully Crossovers : 0
Best Fitness In All Iterations : - 21
Number Of Input Amino Acid : 50
Failed Initialize Of Chromosome : 158019
Number Of Best Improvement : 19
Last Iteration That Best Improved : 425
Equal Masks Num : 0

Draw Graph Diagram

Protein Structure Predictor
Designed By :
Najma Saeed

Figure 8: User interface of software for sequence with length 50

As shown in the table 2, the genetic algorithm with the crossover pattern mask operator has better results than the two basic genetic algorithms and the depth first search genetic algorithm. Therefore, it can be concluded that to solve the problem of predicting the structure of the protein, crossover operator with pattern mask can give an acceptable improvement to the genetic algorithm.

Table 2: Results obtained from implementation

Test No.	*Sequence length*	*Basic genetic algorithm*	*Genetic algorithm with depth first search*	*Genetic algorithm with pattern mask*
1	50	-18	-19	-21
2	60	-31	-32	-32
3	64	-33	-35	-37
4	85	-46	-46	-46
5	100	-40	-42	-42

5. Conclusions and future work

In this paper, we investigate the efficiency of one of the heuristic algorithms for solving the problem of protein structure prediction, using the HP model. Genetic algorithm is one of the heuristic algorithms that has been widely used to solve this problem, and by adding the crossover operator to this algorithm, we tried to improve the algorithm to solve the problem of predicting protein structure. The idea presented in the paper, according to the results obtained from the implementations, has shown a positive effect on the genetic algorithm. The results also showed that the genetic algorithm with the crossover pattern mask operator, compared to the basic genetic algorithm and the improved genetic algorithm with the depth first search, obtained better and more acceptable results in the standard test data.

In future works, we intend to examine the performance of other heuristic algorithms and to perform similar experiments on other standard test data to ensure the performance of the proposed algorithm.

References

[1] A. Mazidi, F. Roshanfar, and V. Parvin Darabad, “A Review of Outliers: Towards a Novel Fuzzy Method for Outlier Detection ,” J. Appl. Dyn. Syst. Control, vol. 2, no. 1, pp. 7–17, Jun. 2019.

[2] B. Patel, V. Singh, and D. Patel, “Structural Bioinformatics,” in Essentials of Bioinformatics, Volume I, Cham: Springer International Publishing, 2019, pp. 169–199.

[3] A. Mazidi, M. Fakhrahmad, and M. Sadreddini, “A meta-heuristic approach to CVRP problem : local search optimization based on GA and ant colony,” J. Adv. Comput. Res., vol. 7, no. December, pp. 1–22, 2016.

[4] A. Mazidi and E. Damghanijazi, “Meta-Heuristic Approaches for Solving Travelling Salesman Problem A meta-heuristic approach to CVRP problem View project Meta-Heuristic Approaches for Solving Travelling Salesman Problem View project Meta-Heuristic Approaches for Solving Travelling Salesman Problem,” Int. J. Adv. Res. Comput. Sci., vol. 8, no. 5.

[5] B. Berger and T. Leighton, “Protein folding in the hydrophobic-hydrophilic (HP) model is NP-complete, Mathematics Department and Laboratory for Computer Science,” 1998.

[6] R. Unger and J. Moult, “Genetic algorithms for protein folding simulations,” J. Mol. Biol., vol. 231, no. 1, pp. 75–81, May 1993.

[7] K. F. Lau and K. A. Dill, “A Lattice Statistical Mechanics Model of the Conformational and Sequence Spaces of Proteins,” Macromolecules, vol. 22, no. 10, pp. 3986–3997, Oct. 1989.

[8] A. A. Tantar, N. Melab, E. G. Talbi, B. Parent,

and D. Horvath, “A parallel hybrid genetic algorithm for protein structure prediction on the computational grid,” Futur. Gener. Comput. Syst., vol. 23, no. 3, pp. 398–409, Mar. 2007.

[9] V. Cutello, G. Nicosia, M. Pavone, and J. Timmis, “An immune algorithm for protein structure prediction on lattice models,” IEEE Trans. Evol. Comput., vol. 11, no. 1, pp. 101–117, Feb. 2007.

[10] R. Unger, “The Genetic Algorithm Approach to Protein Structure Prediction,” Springer, Berlin, Heidelberg, 2004, pp. 153–175.

[11] H. D. D. Ziero, L. S. Buller, A. Mudhoo, L. C. Ampese, S. I. Mussatto, and T. F. Carneiro, “An overview of subcritical and supercritical water treatment of different biomasses for protein and amino acids production and recovery,” J. Environ. Chem. Eng., p. 104406, Sep. 2020.

[12] A. Mazidi, E. Damghanijazi, and S. Tofighy, “An Energy-efficient Virtual Machine Placement Algorithm based Service Level Agreement in Cloud Computing Environments,” Circ. Comput. Sci., vol. 2, no. 6, pp. 1–6, 2017.

[13] A. Mazidi, M. Golsorkhtabaramiri, and M. Y. Tabari, “Autonomic resource provisioning for multilayer cloud applications with K-nearest neighbor resource scaling and priority-based resource allocation,” Softw. Pract. Exp., Apr. 2020.

[14] Sharapov RR, “Genetic Algorithms: Basic Ideas, Variants and Analysis,” 2007.

[15] M. T. Hoque, M. Chetty, A. Lewis, A. Sattar, and V. M. Avery, “DFS-generated pathways in GA crossover for protein structure prediction,” Neurocomputing, vol. 73, no. 13–15, pp. 2308–2316, Aug. 2010.

[16] A. Mazidi, M. Golsorkhtabaramiri, and M. Yadollahzadeh Tabari, “An autonomic risk- and penalty-aware resource allocation with probabilistic resource scaling mechanism for multilayer cloud resource provisioning,” Int. J. Commun. Syst., p. e4334, Feb. 2020.

[17] I. S. Hart W, “HP Benchmarks,” 2005.

[18] N. Lesh, M. Mitzenmacher, and S. Whitesides, “A complete and effective move set for simplified protein folding,” in Proceedings of the Annual International Conference on Computational Molecular Biology, RECOMB, 2003, pp. 188–195.