

Computational Approach for Protein Structure Prediction

Amouda Venkatesan, PhD, Jeyakodi Gopal, MPhil, Manimozhi Candavelou, MSc, Sowjanya Gollapalli, MSc, Kayathri Karthikeyan, MSc

Centre for Bioinformatics, Pondicherry University, Kalapet, Pondicherry, India

Objectives: To predict the structure of protein, which dictates the function it performs, a newly designed algorithm is developed which blends the concept of self-organization and the genetic algorithm. **Methods:** Among many other approaches, genetic algorithm is found to be a promising cooperative computational method to solve protein structure prediction in a reasonable time. To automate the right choice of parameter values the influence of self-organization is adopted to design a new genetic operator to optimize the process of prediction. Torsion angles, the local structural parameters which define the backbone of protein are considered to encode the chromosome that enhances the quality of the confirmation. Newly designed self-configured genetic operators are used to develop self-organizing genetic algorithm to facilitate the accurate structure prediction. **Results:** Peptides are used to gauge the validity of the proposed algorithm. As a result, the structure predicted shows clear improvements in the root mean square deviation on overlapping the native indicates the overall performance of the algorithm. In addition, the Ramachandran plot results implies that the conformations of phi-psi angles in the predicted structure are better as compared to native and also free from steric hindrances. **Conclusions:** The proposed algorithm is promising which contributes to the prediction of a native-like structure by eliminating the time constraint and effort demand. In addition, the energy of the predicted structure is minimized to a greater extent, which proves the stability of protein.

Keywords: Chromosomes, Peptide, Methionine Enkephalin, Islet Amyloid Polypeptide, Mutation

Submitted: November 21, 2012

Revised: 1st, February 22, 2013; 2nd, March 30, 2013

Accepted: April 1, 2013

Corresponding Author

Amouda Venkatesan, PhD

Centre for Bioinformatics, Pondicherry University, R. V. Nagar, Kalapet, Pondicherry 605014, India. Tel: +91-413-2655212, Fax: +91-413-2655211, E-mail: amouda@yahoo.com

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

© 2013 The Korean Society of Medical Informatics

1. Introduction

Resolving the functional conformation of a protein remains a central challenge in molecular biology [1]. Structural knowledge is imperative to analyze the functionality of protein; such knowledge contributes much to the development of better drug design, higher-yield crops, and even synthetic bio-fuels. Predicting the three-dimensional (3D) structure of a protein is one of the greatest challenge and a cornerstone for structural biologists.

Currently, there are three experimental techniques, namely, X-ray crystallography, nuclear magnetic resonance (NMR) spectroscopy, and electron microscopy, that can be used to determine the 3D structure of protein with accuracy. Due to technical difficulties, the gap between the number of known

sequences and the predicted structure is widening, which is a formidable challenge in terms of effort demand and time consumption. Thus, there is a greater need than ever before for a reliable computational method to address the problem of protein structure prediction (PSP) directly from the sequence.

The existing computational methods are categorized into three approaches based on the information used to model the protein. Homology modeling and threading utilize the structural information of similar proteins, whereas the *ab initio* method does not. The template free approach is followed by *ab initio* and it has been proven to be better in several cases.

Many successive computational studies have proposed solution for the conformation structure of protein, such as the evolutionary algorithm [2,3], Monte Carlo [4,5], and HP models [6,7]; however, such solutions are not powerful to search structures in a huge conformational space. The genetic algorithm (GA) has been chosen and has proven to be a reliable computational tool to address the PSP problem because of 1) its success in solving a complex problem by improving the speed using parallelization and 2) the computation process involved in the genetic search to find the optimal strategies for a large space.

Even though several studies have explored the implementation of GA variants [1,8-17] on PSP, the efficiency of GA can be realized only by setting appropriate parameter values. As various inputs require different parameter settings, they are usually obtained by ad-hoc experimentation with manual tuning by a trial-and-error method. This is time consuming and requires a lot of experience and knowledge. On the other hand, inappropriate selection of values leads to premature convergence and consequently, failure to find the optimum in a reasonable computational time, especially when the size of the conformational space is huge (PSP).

From the research work [18] using GA, it is clear that the confirmation results produced were not satisfactory. To overcome the previously mentioned limitations, the most important parameter values of genetic operators need to be self-organized. Therefore, this study focuses on the application of a novel genetic algorithm called the self-organizing genetic algorithm for PSP (SOGA-PSP), which self-configures the crossover and mutation rate during execution, thereby reducing the complexity, which in turn increases the performance efficiency.

Since Met-enkephalin (Protein Data Bank [PDB] ID: 1PLW) has been a test case in many studies [16,19,20], the developed algorithm was tested using the same protein. In addition, the NNFGAIL segment from islet amyloid poly-

peptide (PDB ID: 3DGJ) was also tested.

II. Methods

1. Genetic Algorithm

Algorithm begins with initial population

- 1) Fitness function $f(x)$ is calculated for every chromosome x in the population.
- 2) Reiterate the following procedure to regenerate the population;
 - Choose the parents on the basis of fitness score
 - Reproduce the offspring by crossover operator
 - Mutate the resultant chromosome
 - Calculate the fitness score for elite selection
 - Resultant chromosome reproduces the population
- 3) Algorithm terminates as it reaches the fitness value/number of generations.

2. Self-Organizing Genetic Algorithm

A self-organizing system operates based on information received locally without the guidance of the external environment. It is an adaptive methodology adopted for GA to discover the interaction of operators to collect the required information to set the parameter values in an automated way. The SOGA was developed with full knowledge of the test problem and parameter values which are incremented uniformly in each generation. Self-organization contributes desirable factors to GA by adding knowledge to guide and assist the choice of right parameter values.

3. Self-Organizing Genetic Algorithm to Protein Structure Prediction

Generally, knowledge regarding the interaction of parameters leads to better global solutions in less time, thereby making GA more robust. A lack of robustness in the choice of design parameters always leads to search imbalance, causing premature convergence and lower performance, which is completely eliminated by the developed algorithm given below:

- Step 1: Generate n chromosomes for the initial population
- Step 2: Encode the structure for n chromosomes using TINKER [18]
- Step 3: Calculate the energy value for each chromosome using Discovery Studio [21]
- Step 4: Select and save the elite (chromosome with minimal energy value)
- Step 5: Initialize the crossover and mutation rate
- Step 6: Repeat the following steps to create a new population
 - Calculate the crossover point for initial crossover

- Select the elite and replace, if it is low
- Increase crossover rate with uniform interval
- Perform mutation with initial mutation rate
- Select the elite and replace if it is low
- Increase mutation rate by uniform interval

Step 7: Algorithm terminates by reaching the optimal rate
 SOGA-PSP is pictorially represented in Figure 1.

4. Chromosome Representation

The chromosome representation varies for different problems, and as a set, it constitutes an individual structure called a gene, a set of solution representatives. In general, Cartesian coordinates, templates, torsion angles, side chains, and simplified residues are considered for chromosome representation in PSP [8]. In the present work, torsion [22] and side chain angles [23] were assumed to be best for encoding the chromosomes since both contribute much to the consistency and confirmation of protein structure.

5. Number of Generations

Self-organizing mutation and crossover rates are uniformly increased until they reach the optimal values; in turn the termination condition decides the number of generations.

6. Population Initialization

Population is created using frequently occurring torsion angles [15] which have the degree of freedom for adequate adaptability of GA; it helps in determining the 3D conformation, which is chosen for population size.

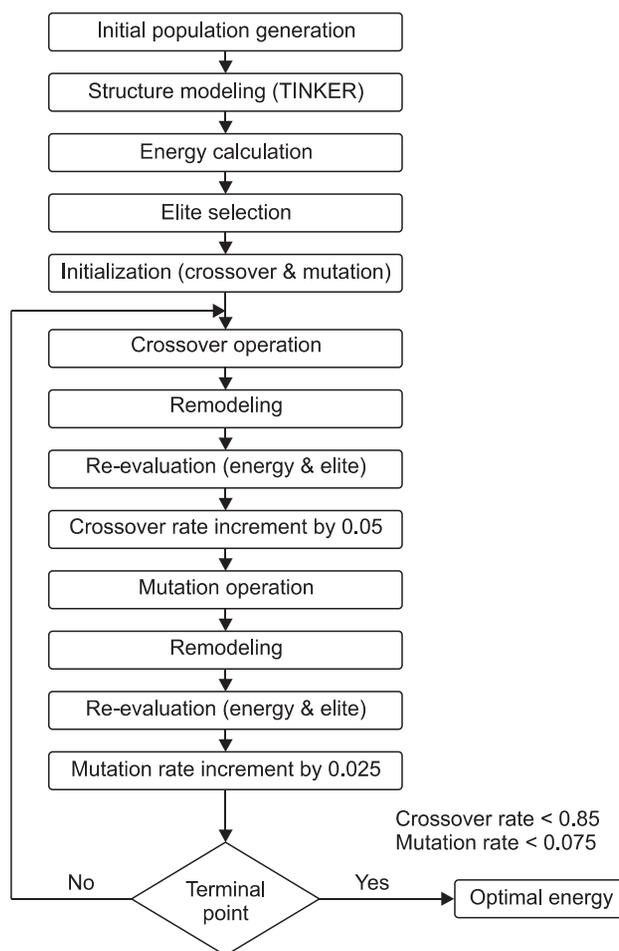


Figure 1. Flowchart for self-organizing genetic algorithm for protein structure prediction.

Sample:

If crossover rate = 0.7

Crossover point = $0.7 \times \text{length} = 19 \times 0.7 = 13.3$

Parent 1

-161.38	161.69	180.0	-67.50	82.50	0	0	0	-179.13	-178.91	180	82.50	-77.50	0	0	0	-179.13	-178.91	180.00
---------	--------	-------	--------	-------	---	---	---	---------	---------	-----	-------	--------	---	---	---	---------	---------	--------

Parent 2

-161.07	142.67	180.00	-62.50	-77.50	0	0	0	-162.75	-138.55	180	62.50	56.54	0	0	0	-162.75	-138.55	180.00
---------	--------	--------	--------	--------	---	---	---	---------	---------	-----	-------	-------	---	---	---	---------	---------	--------



Self-organizing crossover operator

Child 1

-161.38	161.69	180.0	-67.50	82.50	0	0	0	-179.13	-178.91	180	82.50	56.54	0	0	0	-162.75	-138.55	180.00
---------	--------	-------	--------	-------	---	---	---	---------	---------	-----	-------	-------	---	---	---	---------	---------	--------

Child 2

-161.07	142.67	180.00	-62.50	-77.50	0	0	0	-162.75	-138.55	180	62.50	-77.50	0	0	0	-179.13	-178.91	180.00
---------	--------	--------	--------	--------	---	---	---	---------	---------	-----	-------	--------	---	---	---	---------	---------	--------

Figure 2. Self-organizing crossover operation.

7. Fitness Evaluation

The population created in each generation is evaluated by fitness function for selection of the elite [24]. It projects the nature of each chromosome, and the energy value that determines the elite is calculated using the CHARMM force field chosen as the objective function:

$$E_{tot} = E_{bs} + E_{ab} + E_{UB} + E_{id} + E_{ta} + E_{VdW} + E_{cc} \quad (1)$$

- Etot: Total energy (fitness)
- Ebs: Energy of stretching bond
- Eab: Energy of bending angle
- EUB: Energy of Urey-Bradley
- Eit: Energy of improper dihedrals
- Eta: Energy of torsion angle
- EVdW: Energy of van der Waals
- Ecc: Energy of charge-charge [18]

8. Selection

The selection operator elitism, selects the best chromosome with minimal energy as elite. In each generation the elite may be replaced by the low-energy chromosome for genetic operations.

9. Self-Organizing Crossover Operator (SOCO)

To refine the search space, the crossover operator which is normally fixed in standard genetic algorithm (SGA) is self-organized. The least optimal rate is initially chosen for execution without termination at the convergence point, and it is increased uniformly until it reaches the optimal upper limit [25], or the rates are modified from high to low [26]. The reported range of the rate (0.70 to 0.85) is considered for self-organization.

The working principle of SOCO

- Select crossover point with the initial rate
- Perform single-point crossover operation
- Genes from two parents within the crossover point are interchanged to reproduce children
- Model the protein structure using TINKER [27]
- Calculate the energy value to select the elite
- Repeat the process with uniform interval

The crossover operation is pictorially represented in Figure 2.

10. Self-Organizing Mutation Operator (SOMO)

The purpose of mutation is to transform the chromosome to change the conformation and to prevent the population from stagnating at any local optimum. With a fixed mutation rate, the search space is compressed, which leads to premature convergence, and it may not be possible to predict the optimal solution. Self-organization is used to avoid this, and the mutation rate is self-configured. Only the constrained values are considered for the test problem since slight changes may occur in the sequential order of amino acids, which in turn may change the entire structure [17].

The working principle of the SOMO operator is:

- Select mutation point with initial rate
- Mutate the gene value
- Model the protein structure using TINKER
- Calculate the energy value to select the elite
- Repeat the process with uniform interval

The mutation operation is pictorially represented in Figure 3.

III. Results

The efficiency of the developed algorithm is implemented on a biological problem, PSP, using two peptides as test case for

Sample:
 If mutation rate = 0.1
 Mutation point = chromosome length*0.1 = 19*0.1 = 1.9
 *2 gene values should be mutated

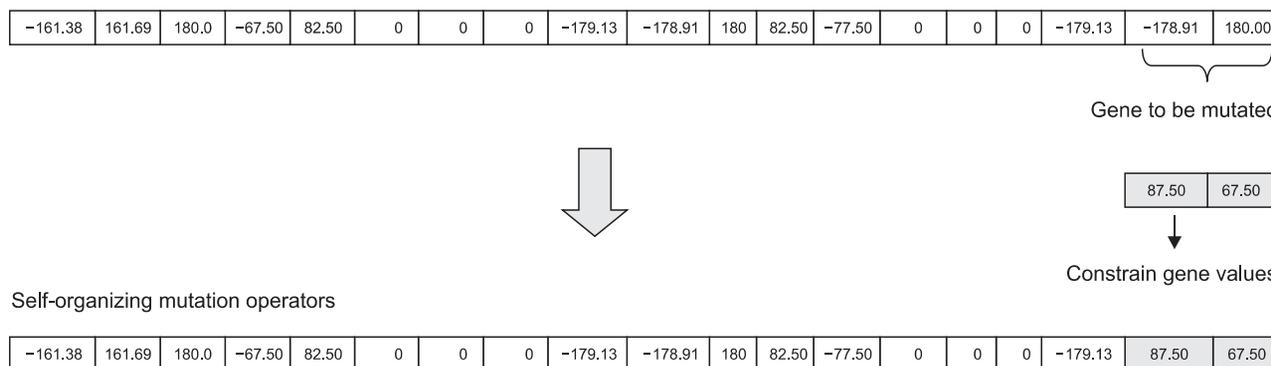


Figure 3. Self-organizing mutation operation.

Met-enkephalin and NNFGAIL.

1. Met-enkephalin

Met-enkephalin (PDB ID: 1PLW) was chosen because it plays a crucial role in generating tissue damage signals as initiated by both peripheral and central nervous system (CNS). Also, it is a simple pentamer used as a test case to measure the efficiency of various *ab initio* methods. Unlike natural crossover and mutation observed in real cellular chromosomes, randomness is restricted in SGA, where it is designed to propagate in a defined architecture due to fixed values. This limitation is relaxed by defining the rate range so that better results can be produced by switching between the intervals. The rate ranges of 0.75–0.85 and 0.025–0.075 were adopted for crossover and mutation to execute the developed SOGA, whereas the rates are 0.85 and 0.075 in SGA. Accordingly, three generations were produced using predefined rates as shown in Tables 1 and 2. With the inher-

ent limitations of SGA in choosing a crossover rate, peptide conformations are generated, and a huge deviation of energy is observed as seen in the plot. However, in the SOGA approach, predicted conformations are maintained in a standard low-energy distribution as shown in Figures 4 and 5. The best values scored by the protein structure indicate that it provides much better confirmation against a native-structure using NMR. Also similar backbone confirmation is indicated by a root mean square deviation (RMSD) value of 0.0009. Eventually, differences in side chain orientations are also clearly seen; this improves the backbone dihedral angles, which is very clear from Ramachandran plot shown in Table 3 and Figure 6 [28]. On closer view, differences in the orientation of side chains (Y1, F3, and M5) of the predicted and native structures clearly indicate better side chain selection. The overall results confirm the performance efficiency of SOGA-PSP to predict refined peptide structures free from steric hindrances.

Table 1. 1PLW - elite selection with minimal energy value for SGA

Generation 1 Crossover rate = 0.85 Mutation rate = 0.075		Generation 2 Crossover rate = 0.85 Mutation rate = 0.075		Generation 3 Crossover rate = 0.85 Mutation rate = 0.075	
Crossover	Mutation	Crossover	Mutation	Crossover	Mutation
-171.59229	-172.18845	-170.64494	-172.14163	-170.39186	-172.1935
-172.37836	-172.92455	-173.84797	-172.43567	-173.67424	-172.92455
-176.90749	-174.42711	-174.65798	-175.00852	-173.49961	-174.42711
-216.58899	-221.72898	-209.17892	-221.27071	-216.07702	-221.72898
-260.99627	-264.63671	-253.83570	-263.58868	-264.26139	-264.63671
1932.59728	1.02E+08	868.97016	1928.20332	869.13149	1923.52245
-231.63577	15488.3	-265.44626	1932.6694	-232.02055	-257.46188
-210.76364	-210.17976	-204.82853	-215.72846	-207.79661	-210.17976
-172.20762	-199.82153	-204.37479	-206.31075	-196.51983	-199.82153
-172.20762	-201.17168	-194.72876	-200.07026	-201.80910	-201.17168
-230.42117	-234.48144	-218.00101	-234.48144	-218.00101	-234.48144
-228.74409	-267.76956	-234.93003	-270.99066	-234.93003	-267.76956
-239.68646	-244.67124	-268.80995	-233.03179	-268.32918	-244.67124
-212.88365	-213.63925	-217.97674	-219.25625	-219.36322	-212.98686
-195.23965	-199.66229	-207.06528	-211.30368	-197.757744	-192.25972
-177.59569	-181.43619	-195.45288	-193.35084	-165.66421	-188.1002
-215.40841	-275.42566	-274.54747	-274.34997	-278.86180	-275.42566
-238.12503	-237.77215	-194.64695	-240.54593	-236.54583	-256.89867
-253.12312	-237.77513	-272.02328	-266.73044	-234.20797	-266.05603
-235.96314	-195.0875	-203.91085	-204.21397	-230.62773	-197.82337

1PLW: Met-enkephalin, SGA: standard genetic algorithm.

Table 2. 1PLW - elite selection with minimal energy value for SOGA (highlighted)

Generation 1 Crossover rate = 0.75 Mutation rate = 0.025		Generation 2 Crossover rate = 0.80 Mutation rate = 0.05		Generation 3 Crossover rate = 0.85 Mutation rate = 0.075	
Crossover	Mutation	Crossover	Mutation	Crossover	Mutation
-171.10760	-172.23923	-170.94641	-172.21268	-170.42172	-159.52444
-172.94472	-172.7326	-173.84371	-172.59388	-173.24245	-172.45245
-177.76095	-178.73269	-178.49301	-178.66430	-176.70001	-178.28913
-217.06141	-214.21062	-218.215	-214.31990	-213.57530	-213.52977
-264.86048	-271.23110	-239.45125	-242.10899	-270.96874	-271.23110
1932.52061	-254.92830	-270.09628	-260.45899	-268.65440	-255.29472
-231.63577	-233.83280	-277.30329	-255.00251	-232.59092	-257.26083
-210.76364	-205.27317	-255.68121	-204.74780	-199.19133	-192.53621
-198.43299	-197.903	-208.97408	-208.26187	-104.79681	-215.86607
-213.17293	-203.34033	-201.86809	-205.3305	-199.16423	-223.84774
-231.17293	-218.07886	1230.15828	-220.39921	-252.87856	-224.11319
-231.31206	-198.28606	-227.4667	-248.02747	-268.43895	-224.57840
-245.40767	-241.76029	-266.1819	-257.68881	-268.43895	-255.31500
-252.08365	-218.53758	-229.47055	-225.29209	-221.35872	-254.22995
-255.1311	-229.94733	-205.4311	-201.26061	-213.76205	-221.35872
-203.00470	-234.91466	-210.63538	-208.74070	-205.70740	-198.67929
-281.94584	-270.4918	-274.61564	-285.30224	-275.09863	-197.87270
-238.12503	-240.61146	-240.99259	-241.30708	-203.20855	-285.49187
-269.17322	-258.58849	-204.05672	-271.12167	-266.59104	-201.05081
-223.19976	-261.49866	-213.1576	-200.04069	-204.93426	-264.38387

1PLW: Met-enkephalin, SOGA: self-organizing genetic algorithm.

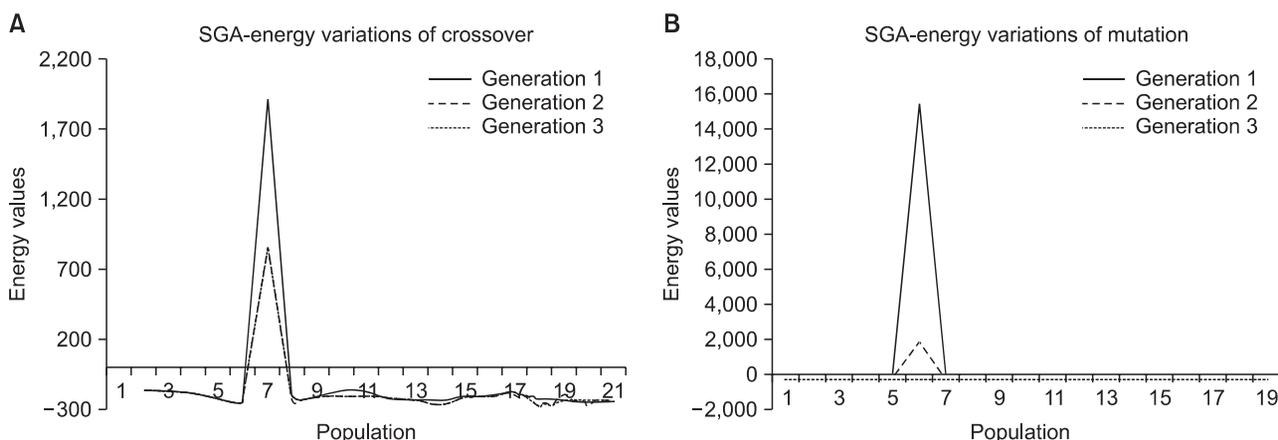


Figure 4. (A) 1PLW - graphical schema of energy variations of crossover in standard genetic algorithm (SGA). (B) Graphical schema of energy variations of mutation in SGA.

2. NNFGAIL Segment from Islet Amyloid Polypeptide (3DGJ)

NNFGAIL is a fragment of human islet amyloid polypeptide (IAPP or amylin) of 37-residue hormone found in pancreatic

extracts of type 2 diabetics. It plays a role in the function of the pancreas and contributes to glycemic control [29]. The structure of the peptide was predicted using SOGA; as a result,

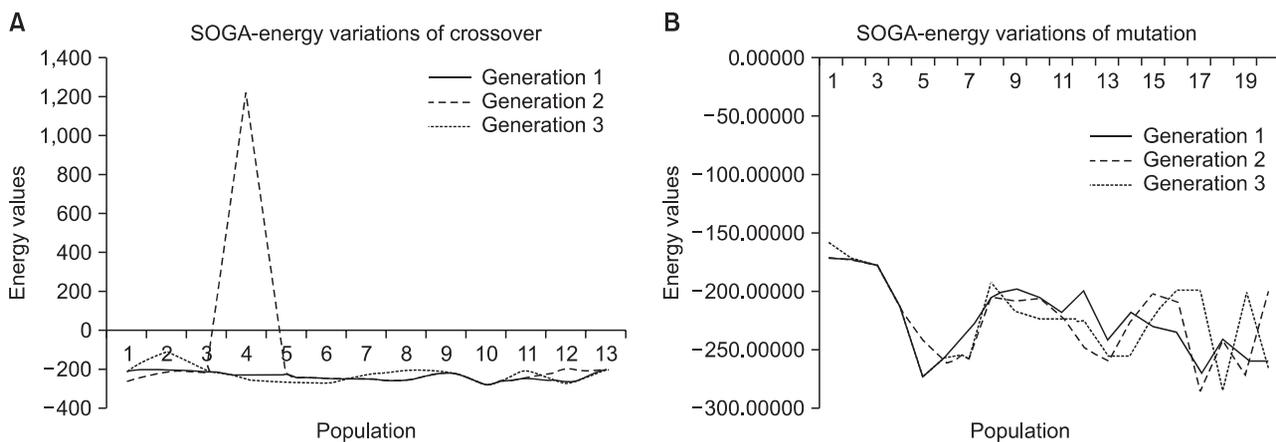


Figure 5. (A) 1PLW - graphical schema of energy variations in crossover of self-organizing genetic algorithm (SOGA). (B) Graphical schema of energy variations in mutation of SOGA.

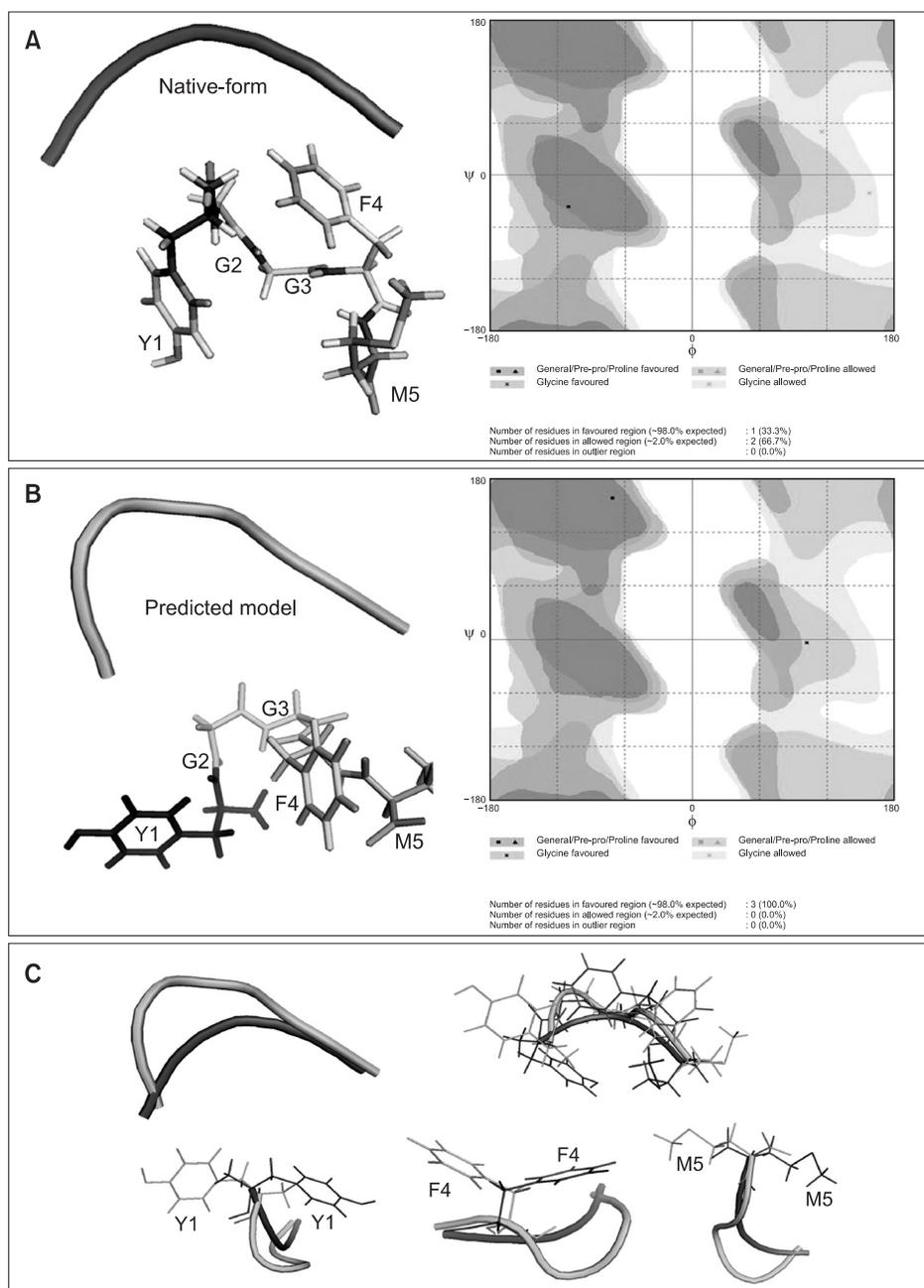


Figure 6. (A) Native structure and its Ramachandran plot of 1 PLW, (B) Predicted structure and its Ramachandran plot. (C) Differences between the native and predicted structures on superimposition.

the energy of the structure was comparatively minimal which indicates better conformation as shown in Table 4 and Figure 7. On superimposing with the native structure, an RMSD value of 0.939 was obtained and 5 hydrogen bonds were formed, a most important factor for structure stability. Furthermore, as an advantage out of two hydrogen bonds of X-ray structure, one is conserved. This conservation is also seen in the Ramachandran plot (Table 5 and Figure 8). On further examina-

tion, the differences in the orientation of the side chains of the predicted and native structures indicate that better side chain selection is achieved with the proposed approach.

IV. Discussion

It is well known that the prediction of protein structures through wet lab or experimental methods is very expensive

Table 3. Ramachandran plot of predicted and native structures of 1PLW

Protein	Region structure	Favored (%)	Additional allowed region (%)	Disallowed region (%)
1PLW	Predicted structure	100.0	0	0
	Native structure	33.3	66.7	0

1PLW: Met-enkephalin.

Table 5. Ramachandran plot of predicted and native structures of 3DGJ

Protein	Region structure	Favored (%)	Additional allowed region (%)	Disallowed region (%)
3DGJ	Predicted structure	75.0	25.0	0
	Native structure	75.0	25.0	0

3DGJ: islet amyloid polypeptide.

Table 4. 3DGJ - elite selection with minimal energy value for SOGA (highlighted)

Generation 1 Crossover rate = 0.75 Mutation rate = 0.025		Generation 2 Crossover rate = 0.80 Mutation rate = 0.05		Generation 3 Crossover rate = 0.85 Mutation rate = 0.075	
Crossover	Mutation	Crossover	Mutation	Crossover	Mutation
-281.19281	-278.02795	-281.84365	-281.84365	-280.75352	-275.70773
-278.92144	-265.13230	-278.88138	-278.88138	-263.13612	-275.01806
-272.38611	-263.94010	-276.08350	-268.26382	-264.01089	-264.05983
-292.70103	-302.48385	-293.87336	-281.76799	-302.74224	-302.74224
-255.36956	-251.58781	-252.07568	-252.07568	-253.29922	-253.29922
-292.60674	-278.72430	-292.82043	-292.82043	-293.62817	-293.62817
-270.31410	-265.06665	-269.86710	-269.86710	-266.22684	-266.26840
-280.97149	-292.38454	-269.04647	-269.04647	-285.85525	-285.85525
-279.69023	-295.65093	-275.60934	-282.21989	-281.19843	-281.19843
-282.48866	-278.70978	-282.53619	-282.53619	-275.24063	-275.24063
-266.11218	-268.82571	-270.72094	-271.79699	-266.16514	-244.31593
-265.87125	-268.05299	-257.67895	-257.67895	-275.57600	-266.74591
-266.17720	-270.56018	-275.94368	-275.97891	-277.89890	-277.89890
-248.01801	-239.58074	-259.06633	-259.53144	-235.51707	-234.21624
-280.79328	-279.98272	-279.69149	-279.69149	-283.42990	-283.42990
-268.82144	-276.02419	-286.53978	-282.86298	-276.93297	-276.93297
-290.67117	-290.02132	-283.28249	-283.28249	-286.38617	-286.37940
-274.93633	-270.26848	-273.70065	-273.70065	-270.32572	-270.32572
-278.83112	-272.68608	-288.25009	-288.22309	-275.70786	-275.70773
-278.24753	-274.51539	-272.78755	-272.78755	-269.64582	-269.64582

3DGJ: islet amyloid polypeptide, SOGA: self-organizing genetic algorithm.

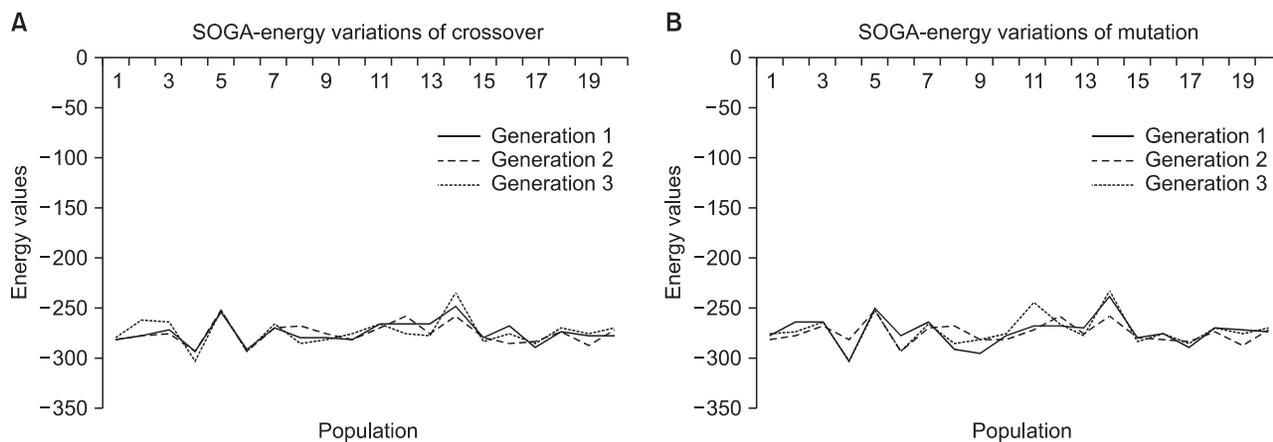


Figure 7. (A) 3DGJ - graphical schema of energy variations in crossover of self-organizing genetic algorithm (SOGA). (B) Graphical schema of energy variations in mutation of SOGA.

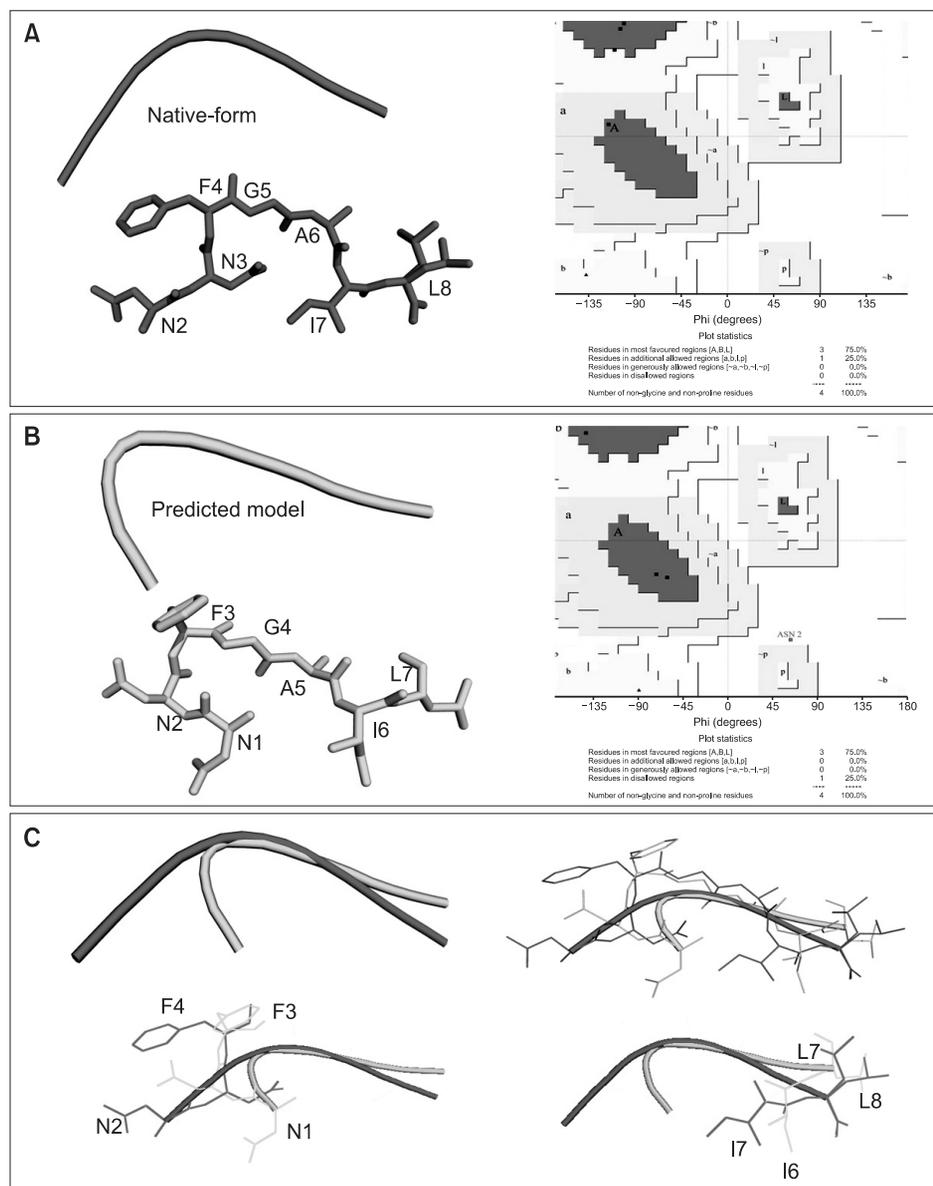


Figure 8. (A) Native structure and Ramachandran plot of 3DGJ. (B) Predicted structure and its Ramachandran plot. (C) Differences between the native and predicted structures on superimposition.

and time consuming. Hence, the challenge was taken up in this study to develop a novel algorithm, SOGA implemented on PSP. The proposed algorithm was proved to be better, and the result obtained demonstrated that it outperforms the approaches taken in previous work [18]. It also avoids the previously mentioned drawbacks of other approaches, which proves the relative efficiency of the algorithm. Furthermore, energy is minimized to a greater extent, and the RMSD value reflects a similar backbone conformation against native structures. The strength of this research also lies in considering only torsion angles as input. In general, the best mechanism adopted for PSP contributes much to the modeling of loops, which are the most common and functionally important substructures. Using current methods like homology modeling or database search, the loops are modeled with less accuracy. As SOGA-PSP has been shown to be a better method to predict the structure of proteins, and it can also be extended to loop modeling.

Conflict of Interest

No potential conflict of interest relevant to this article was reported.

Acknowledgments

Dr. V. Amouda acknowledges the receipt of financial support from University Grants Commission (UGC), Government of India, and Centre for Bioinformatics, Pondicherry University, India. The authors are grateful to Mr. M. Sundara Mohan and Mr. P. Manivel, Centre for Bioinformatics, Pondicherry University for their constant support all through the work.

References

1. Pedersen JT, Moult J. Genetic algorithms for protein structure prediction. *Curr Opin Struct Biol* 1996;6(2):227-31.
2. de Lima TW, Gabriel PH, Delbem AC, Faccioli RA, da Silva IN. Evolutionary algorithm to ab initio protein structure prediction with hydrophobic interactions. In: *Proceedings of the IEEE Congress on Evolutionary Computation*; 2007 Sep 25-28; Singapore. p. 612-9.
3. Piccolboni A, Mauri G. Application of evolutionary algorithm to protein folding prediction. In: Hao JK, Lut-ton E, Ronald E, Schoenauer M, Snyers D, editors. *Artificial Evolution*. Heidelberg, Germany: Springer; 1998. p. 123-35.
4. Carnevali P, Toth G, Toubassi G, Meshkat SN. Fast protein structure prediction using monte carlo simulations with modal moves. *J Am Chem Soc* 2003;125(47):14244-5
5. Lee B, Kurochkina N, Kang HS. Protein folding by a bi-ased Monte Carlo procedure in the dihedral angle space. *FASEB J* 1996;10(1):119-25.
6. Mandal S, Jana ND. Protein structure prediction using 2D HP lattice model based on integer programming ap-proach. In: *Proceedings of 2012 International Congress on Informatics, Environment, Energy and Applications*; 2012 Mar 17-18; Singapore. p. 171-5.
7. Benitez CM, Lopes HS. Protein structure prediction with the 3D-HP side-chain model using a master-slave parallel genetic algorithm. *J Braz Comput Soc* 2010;16(1):69-78.
8. Cui Y, Chen RS, Wong WH. Protein folding simulation with genetic algorithm and supersecondary structure constraints. *Proteins* 1998;31(3):247-57.
9. Zhang X, Wang T, Luo H, Yang JY, Deng Y, Tang J, et al. 3D protein structure prediction with genetic tabu search algorithm. *BMC Syst Biol* 2010;4 Suppl 1:S6.
10. Hoque MT, Chetty M, Sattar A. Genetic algorithm in ab initio protein structure prediction using low resolu-tion model: a review. In: Sidhu AS, Dillon TS, editors. *biomedical data and applications*. Heidelberg, Germany: Springer; 2009. p. 317-42.
11. Dandekar T, Argos P. Applying experimental data to protein fold prediction with the genetic algorithm. *Protein Eng* 1997;10(8):877-93.
12. Contreras-Moreira B, Fitzjohn PW, Offman M, Smith GR, Bates PA. Novel use of a genetic algorithm for protein structure prediction: searching template and se-quence alignment space. *Proteins* 2003;53 Suppl 6:424-9.
13. Goldberg DE. *Genetic algorithms in search, optimiza-tion and machine learning*. Reading (MA): Addison-Wesley Publishing Co.; 1989.
14. Kaiser CE, Merkle LD, Lamont GB, Gates GH Jr, Pachter R. Case studies in protein structure prediction with real-valued genetic algorithms. In: *Proceedings of the 8th SIAM Conference on Parallel Processing for Scientific Computing*; 1997 Mar 14-17; Minneapolis, MN.
15. Day RO, Zydallis JB, Lamont GB, Pachter R. Solving the protein structure prediction problem through a multi-objective genetic algorithm. In: *Proceeding of the In-ternational Conference on Computational Nanoscience and Nanotechnology*; 2002 Apr 21-25; San Juan, Puerto Rico. p. 32-5.
16. Deerman KR, Lamont GB, Pachter R. Linkage-learning genetic algorithm application to the protein structure

- prediction problem. In: Proceedings of the ACM Symposium on Applied Computing; 2001 Mar 11-14; Las Vegas, NV. p. 333-9.
17. Schulze-Kremer S, Tiedemann U. Parameterizing genetic algorithms for protein folding simulation. In: Proceedings of the 27th Hawaii International Conference on System Sciences; 1994 Jan 4-7; Wailea, HI. p. 345-54.
 18. Tragante-do-O V, Tinos R. A self-organizing genetic algorithm for protein structure prediction. *Learn Non-linear Model* 2010;8(3):135-47.
 19. Fogel GB, Corne DW. *Evolutionary computation in bioinformatics*. Boston (MA): Morgan Kaufmann Publishers; 2003.
 20. Klepeis JL, Pieja MJ, Floudas CA. Hybrid global optimization algorithms for protein structure prediction: alternating hybrids. *Biophys J* 2003;84(2 Pt 1):869-82.
 21. Accelrys Software Inc. *Discovery Studio modeling environment*, release 3.1. San Diego (CA): Accelrys Software Inc.; 2012.
 22. Srinivasan R. Phi, psi distribution of amino-acids from 42 proteins [Internet]. Baltimore (MD): RoseLab, Johns Hopkins University; [cited 2013 May 1]. Available from: <http://roselab.jhu.edu/~raj/Research/Linus/phispsi.html>.
 23. Tuffery P, Etchebest C, Hazout S, Lavery R. A new approach to the rapid determination of protein side chain conformations. *J Biomol Struct Dyn* 1991;8(6):1267-89.
 24. Thierens D. Adaptive mutation rate control schemes in genetic algorithms. In: Proceedings of the 2002 Congress on Evolutionary Computation; 2002 May 12-17; Honolulu, HI. p. 980-5.
 25. Bao-Juan H, Jian Z, De-Hong Y. A novel and accelerated genetic algorithm. *WSEAS Trans Syst Control* 2008;3(4):269-78.
 26. Breukelaar R, Back T. Self-adaptive mutation rates in genetic algorithm for inverse design of cellular automata. In: Proceedings of the 10th Annual Conference on Generic and Evolutionary Computation; 2008 Jul 12-16; Atlanta, GA. p. 1101-2.
 27. Jay Ponder Lab. TINKER: software tools for molecular design (TINKER 3.9, June 2001) [Internet]. St. Louis (MO): Washington University; c2013 [cited 2013 May 1]. Available from: <http://dasher.wustl.edu/tinker/>.
 28. Rampage. Cambridge: Department of Biochemistry, University of Cambridge; [cited 2013 May 1]. Available from: <http://mordred.bioc.cam.ac.uk/~rapper/rampage.php>.
 29. Wiltzius JJ, Sievers SA, Sawaya MR, Cascio D, Popov D, Riek C, et al. Atomic structure of the cross- β spine of islet amyloid polypeptide (amylin). *Protein Sci* 2008;17(9):1467-74.