Multimodal HP Model Protein Folding using Clustering and Differential Evolution

Master's Degree Thesis Summary

Beryl Ramadhian Aribowo
Department of Computational Science
Institut Teknologi Bandung
Bandung, Indonesia
berylramadhian@gmail.com

Supervisor I: Kuntjoro Adji Sidarto
Department of Mathematics
Institut Teknologi Bandung
Bandung, Indonesia
sidarto@math.itb.ac.id

Supervisor II: Rukman Hertadi
Department of Chemistry
Institut Teknologi Bandung
Bandung, Indonesia
rukman@chem.itb.ac.id

***Abstract*—The goal of protein folding is to predict the protein native structure given a sequence of amino acid. The protein folding problem is modelled into lattice model. Within this research, Differential Evolution algorithm is employed to optimize a protein objective function under single target global optimum assumption. The combination of Clustering and Differential Evolution is adopted to optimize the protein objective function under multimodal function mapping problem. The protein conformations are also analyzed on multimodal problem formulation. It has been discovered that if there exist several different global optimum points, the final protein conformations generated are equivalent, which signifies that an amino acid sequence only yields one stable protein conformation.**

***Index Terms*—protein folding, multimodal optimization, differential evolution, clustering**

I. Introduction

Proteins are composed of several amino-acids, these amino-acids bond together and create a polypeptide chain. Protein is the native state of the polypeptide chain after folding process, where different combinations of amino-acids sequence in polypeptide chain will result in different protein structures. Proteins play important roles in living beings, and each protein has different function depending on the structure. Naturally, proteins fold from random coil or unfolded state into native state in split second, meanwhile the precise description of the process and mechanism of protein folding itself is unknown. These conditions rise the importance of the protein folding simulation and protein structure prediction.

Protein folding and protein structure prediction (PSP) are interchangeable terms and mostly refer to the prediction of the tertiary structure or the native state of a protein. Native state of a protein is assumed to be the state of protein conformation with the lowest free energy. Ab-initio (or De novo) approach of protein folding is the most prominent, by taking only the information contained within protein primary structure (polypeptide chain) to predict the tertiary structure [1]. The process of protein folding from the unfolded state to the native state i.e. the prediction of the final native structure of the protein is noted to be having a very high degree of freedom, that it requires astronomical amount of time to sample all of the possible conformations (levinthal paradox) [2].

The protein folding process is often simplified into various mathematical optimization models to reduce the computational complexity. The model simplification is done by taking several assumptions from the physical protein interactions set. One of the most favored protein folding model is lattice HP-model proposed by [3]. HP-model protein takes into account the hydrophobicity property from the 20-types of amino-acids. Despite the simplification, it has been noted that computationally HP-model protein folding's complexity is within the class of NP-hard problems [4].

Many optimization techniques have been applied to protein folding problem including Evolutionary Algorithms (EA), such as Genetic Algorithm (GA) [5], Differential Evolution (DE) [6]. Wong [5] particularly addressed the multimodality of protein folding problem, in the assumption that a native structure of proteins may also be found within the local minimum of the function funnel. However we deemed this assumption as inaccurate for the case of single protein chain (tertiary protein structure). The depiction of the protein folding function in figure 1 of [7] is a merge of tertiary (left part) and quaternary (right part) function structure of protein, hence the native structure of a protein chain is still found in the global optimum point.

The objective within this research work is to simulate and analyze Ab-initio protein folding on HP lattice model with multimodal approach, by taking into account global and local minimum structures found by the multimodal technique. Our hypothesis of the protein folding function multimodality is that the identification of meta-stable structures (local minimum structures) may become important in order to understand the nature of diseases which are caused by protein-misfolds [8], [9].

II. Methodology

A. HP Model Protein

Lattice HP model captures the hydrophobic - polar interactions of amino-acids within the polypeptide chain. The 20 types of amino acids are further grouped into hydrophobic (H) and polar (P) amino-acids. To represent the HP model protein as a mathematical model, the amino acids are embedded into lattice coordinates (2D or 3D). The degree of freedom of each amino acid is represented by the movesets (absolute moveset, relative moveset) inside the lattice. Here we used relative moveset, in 2D lattice it is represented by $\{F, R, L\}^{n-2}$ (Forward, Right, Left). Then the moveset is encoded into real values, i.e. $\mathbf{x}_{move} = (move_1, move_2, ...move_n)$ is transformed into a sequence vector $\mathbf{x} = (x_1, x_2, ...x_n) \in \mathbb{R}^n$.

B. Differential Evolution

DE is a metaheuristic optimization algorithm within the EA class, first proposed by Storn and Price [10]. As a metaheuristic algorithm, DE's flexibility comes from its capabilities to perform operations without any assumptions on the problem, such as not requiring the objective function to be differentiable. Due to DE's ability to perform on any types of function, it has been chosen to be paired with the Clustering algorithm to obtain multiple solutions from multimodal protein folding problem. DE's selection phase is modified to accommodate Clustering's selection scheme, a selection for maximum point vector is employed instead of the usual minimum point selection per iteration of the DE. Moreover, Cauchy's convergence test is also applied to enhance the convergence strength [11].

C. Clustering

Clustering technique proposed by [12] has a main purpose to search the roots of nonlinear systems. Then in the subsequent research work [13] the Clustering method was utilized for multimodal optimization problems, to find all local and global optimum points with just a single run.

The main concept of the Clustering technique is to divide the search space into local search spaces contained within a cluster (diversification phase), each of the clusters has a potential local or global optimum point. The initialization phase is executed by incorporating Sobol sequence as the initial points generator to provide uniformity and deterministic results [14]. Then the diversification phase is done by recursively comparing the function values on the available points to obtain the cluster center candidates, then each points are rotated within the search space to obtain the more suitable cluster center (the "better" point), finally this phase results in several clusters. The intensification phase on each clusters is performed by DE, where in [12] and [13] Spiral Dynamics Optimization Algorithm (SDOA) by [15] was employed for the intensification phase. Each clusters results in one potential local or global optimum point, then the potential local or global points from all clusters are selected based on the clustering parameters to be set as the final local or global optimum points. The detailed steps for the clustering algorithm is shown below.

Input:

1) m_{cl}: Number of points distributed within the feasible domain.
2) k_{cl}: Number of cluster iteration.
3) r_{cl}: Contraction constant for point rotation.
4) θ_{cl}: Angle constant for point rotation.
5) ϵ $(0 < \epsilon < 1)$: parameter for optimum points acceptance.
6) δ $(0 < \delta < 1)$: cut-off distance parameter to distinguish the points of potential optimum candidates.

Algorithm:

1) Generate m_{cl} Sobol sequence initial points $\mathbf{x}_i(0) \in \mathbb{R}^n, i = 1, 2, ..., m_{cl}$ in the feasible region $D = [a_1, b_1] \times [a_2, b_2] \times ... \times [a_n, b_n] \subset \mathbb{R}^n$, and $k = 0$.
2) Set $\mathbf{x}^* = \mathbf{x}_{ig}(0)$, $i_g = \arg\max_i F(\mathbf{x}_i(0))$, $i = 1, 2, ..., m_{cl}$ as center of the first cluster with radius equal to $\frac{1}{2}(\min_l b_l - a_l)$, $l = 1, 2, ..., n$.
3) if $\mathbf{x}_i$ is not the center of existing cluster, then do the function cluster with $\mathbf{x}_i$ as the input.
   Function Cluster (input $\mathbf{y}$*)*:
   a) Find a cluster with its center nearest to $\mathbf{y}$. Let C be such cluster, with center at $\mathbf{x}_c$.
   b) Set $\mathbf{x}_m$ as mid-point between $\mathbf{y}$ and $\mathbf{x}_c$.
   c) Compare $F(\mathbf{y}), F(\mathbf{x}_c), F(\mathbf{x}_m)$:
      - if $F(\mathbf{x}_m) < F(\mathbf{y})$ and $F(\mathbf{x}_m) < F(\mathbf{x}_c)$: Set a new cluster with center at $\mathbf{y}$ and radius $||\mathbf{y} - \mathbf{x}_m||$.
      - Else, if $F(\mathbf{x}_m) > F(\mathbf{y})$ and $F(\mathbf{x}_m) > F(\mathbf{x}_c)$: Set a new cluster with $\mathbf{y}$ as its center and radius $||\mathbf{y} - \mathbf{x}_m||$. Do *Function Cluster*(input $\mathbf{x}_m$).
      - Else, if $F(\mathbf{y}) > F(\mathbf{x}_c)$, set $\mathbf{y}$ as the center of C.
   d) Change the radius of C equal to $||\mathbf{y} - \mathbf{x}_m||$.
4) Set $\mathbf{x}^* = \mathbf{x}_{ig}$, where $i_g = \arg\max_i F(\mathbf{x}_i(k))$, $i = 1, 2, ..., m_{cl}$.
5) Update by rotating the points $\mathbf{x}_i$: $\mathbf{x}_i(k+1) = S_n(r_{cl}, \theta_{cl})\mathbf{x}_i(k) - (S_n(r_{cl}, \theta_{cl}) - I_n)\mathbf{x}^*$, $i = 1, 2, ..., m_{cl}$, and set $k = k + 1$.
6) Do steps 3 to 5 k_{cl} times.
7) Each cluster $C_1, C_2, ...C_{n_C}$ has a center $\mathbf{x}_{C_i}$ and radius of ρ_i $(i = 1, 2, ..., n_C)$. For each cluster, perform intensification phase by employing DE with $D_i = [\mathbf{x}_{1,i} - \rho_i,\ \mathbf{x}_{1,i} + \rho_i] \times [\mathbf{x}_{2,i} - \rho_i,\ \mathbf{x}_{2,i} + \rho_i] \times ... \times [\mathbf{x}_{n,i} - \rho_i,\ \mathbf{x}_{n,i} + \rho_i] \subset \mathbb{R}^n$ as the domain, where $\mathbf{x}_{C_i} = (\mathbf{x}_{1,i}, \mathbf{x}_{2,i}, ..., \mathbf{x}_{n,i})^T$ $i = 1, 2, ..., n_C$.
8) Keep candidate of maximum points which satisfy the conditions $F(\mathbf{x} - \epsilon) < F(\mathbf{x})$ $F(\mathbf{x} + \epsilon) < F(\mathbf{x})$.
9) Suppose from step 8 there results in n_g candidate of points. Select only candidates which satisfy $||\mathbf{x}_i - \mathbf{x}_j|| > \delta$, where $i, j = 1, 2, ..., n_g$. If $||\mathbf{x}_i - \mathbf{x}_j|| \leq \delta$, select only $\mathbf{x}_i$ as maximum point if $F(\mathbf{x}_i) \geq F(\mathbf{x}_j)$, otherwise select $\mathbf{x}_j$ as maximum point. Finally all of the selected candidates are put into the list of the selected local and/or global maximum points.

Output: List of local and global optimum points.

Further details on the Clustering algorithm, at step 5 of the clustering algorithm, the points are rotated in n dimensional plane with $\mathbf{x}^*$ as the center of the rotation. The rotation equation is shown in (1) for rotation in 2-dimensional plane.

For $n > 2$ dimensions, (1) is extended with several rules. Let $R^{(2)}$ be the rotation in 2D-plane, the rotation in n-dimensional plane matrix $R^{(n)}$ with $n \times n$ size is defined in (2). The entries of $R^{(n)}_{i,j}$ matrix are $r_{ii} = r_{jj} = cos(\theta), r_{ji} = sin(\theta), r_{ij} = -sin(\theta)$ and $r_{st} = \delta_{st}$ for all other entries of $R^{(n)}_{i,j}$ (where $\delta_{st} = 1$ if $s = t$ and $\delta_{st} = 0$ if $s \neq t$). The total rotation matrices for n dimensional rotation is $\binom{n}{2} = \frac{n!}{2!(n-2)!} = \frac{1}{2}n(n-1)$.

$$S_2(r, \theta) = diag_2(r, r).R^{(2)},$$

$$diag_2(r, r) = \begin{pmatrix} r & 0 \\ 0 & r \end{pmatrix}, R^{(2)} = \begin{pmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & cos(\theta) \end{pmatrix}, \tag{1}$$

with $0 < r < 1$ and $0 < \theta < 2\pi$.

$$R^{(n)} = \prod_{i=1}^{n-1} (\prod_{j=1}^{i} R^{(n)}_{n-1,n+1-j}) \tag{2}$$

III. Computational Experiments

A. Parameters

There are two scheme of protein folding employed, first is unimodal target (only obtaining the single global optimum), second is multimodal target (obtaining multiple global and local solutions). Each scheme has different parameters of DE and Clustering algorithm. Each algorithms parameters in each scheme will be explained in the respective sections. Within the figures of conformations, H amino-acid is indicated by red bead, meanwhile P amino-acid is indicated by green bead.

The algorithms are implemented using Python 3.7 with NumPy [16], SciPy [17], and Python Sobol [18] library dependencies. The simulation experiments are conducted using a 6-cores 12-threads Ryzen 5 2600x 3.9 Ghz CPU (OC) with 16 GB of RAM. The source code is available at https://github.com/berylgithub/HP-Protein.

B. Protein Folding with single global optimum target

The dataset used is the amino acids with length 20 until 64 provided in [19]. The protein parameters used are shown in table I and DE parameters used for all of the unimodal protein folding are shown in table II. The result of minimum energy found from each sequence is shown in table III. The DE employed was able to find the global minimum of seq-20 and seq-24, however for the rest of the sequences, it is not guaranteed to find the global optimum, may also be local optimum or intermediate structures. To show the folding process, the samples of conformations from seq-20 (figure 1) and seq-24 (figure 2) are taken. The rest of the conformations, which are seq-25 until seq-64 conformations are shown in figure 3. For protein sequences 20-25, the simulation execution time to complete is around 2-4 hours, while for the three longest sequences $\{50, 60, 64\}$, require $\{1.35, 2.5, 3.5\}$ days to be completed respectively.

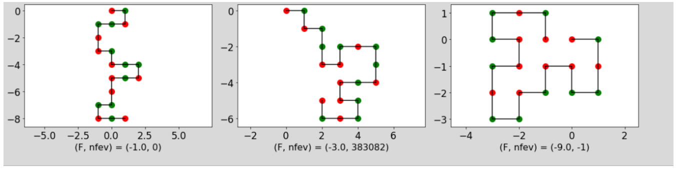


Fig. 1: Sample of seq-20 conformations.

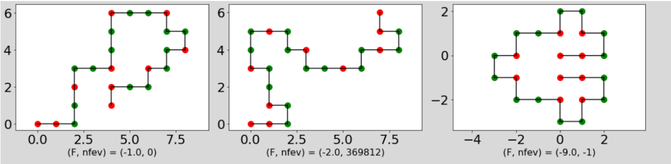
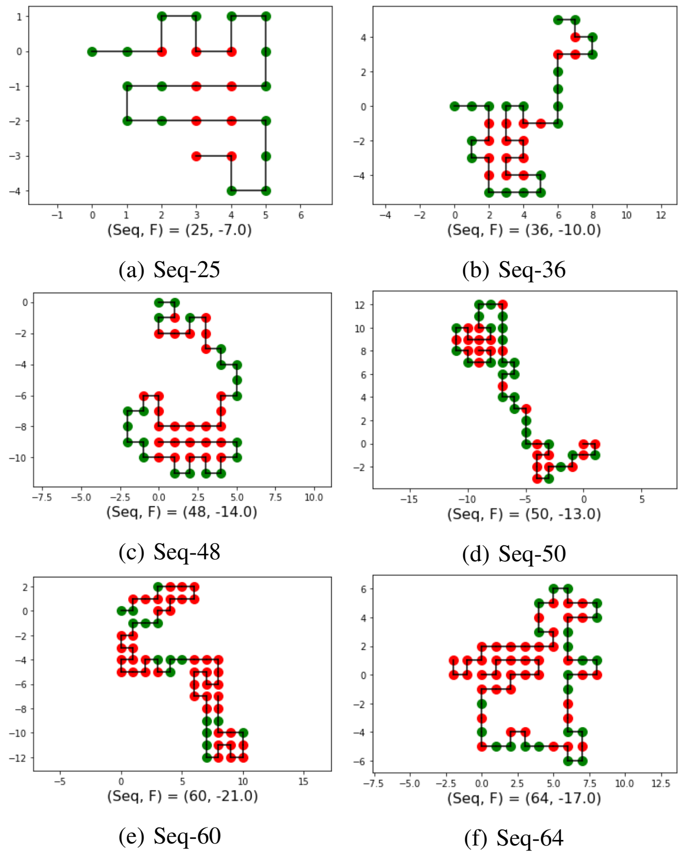


Fig. 2: Sample of seq-24 conformations.



(a) Seq-25

(b) Seq-36

(c) Seq-48

(d) Seq-50

(e) Seq-60

(f) Seq-64

Fig. 3: Conformation with lowest energy value of each sequence.

TABLE I: Protein settings for unimodal.

Protein Settings	**Value**
HH Contact	-1
HP Contact	0
PH Contact	0
PP Contact	0
”F”-range	[0, 3)
”R”-range	[3, 6)
”L”-range	[6, 9)
Infeasible Conformation Energy	0

TABLE II: DE settings for unimodal.

DE Settings	**Value**
Scheme	best/1/bin
Mutation	dithering ([0.5, 1))
Crossp	0.7
Pop_size	length (sequence) * 20
Pop_init	rand_uniform
Max_iter	1000

C. *Protein Folding with multimodal target*

Multimodal protein folding experiments uses only the seq-20. The protein settings used are the same with the unimodal case, except the contact energy values. There are three set of values for contact energy, the first is $\{HH = -1, HP = PH = PP = 0\}$ (experiment 1-4), the second is $\{HH = -2.3, HP = -1, PH = 0, PP = 0\}$ (experiment 5), the third is $\{HH = -2.3, HP = PH = -1, PP = 0\}$ (experiment 6). DE parameters used for all experiments are shown in table IV. The Clustering parameters are shown in table V.

In all of the experiments, local optimum conformations in a energy level are all unique, the sample of local optimum conformations is shown in figure 4. Meanwhile the global optimum conformations are equivalent, e.g. conformation on point A is a rotation result of conformation on point B, such as shown in the conformations of global optimum points shown in figure 5, 6, 7. The 8 conformations with the lowest energy of experiment 5 and 6 are shown in figure 8 and 9 respectively.

IV. Conclusion

The protein folding process can be modeled as the process of finding optimal points on an objective function (mathematical optimization) by using the assumption of a lattice

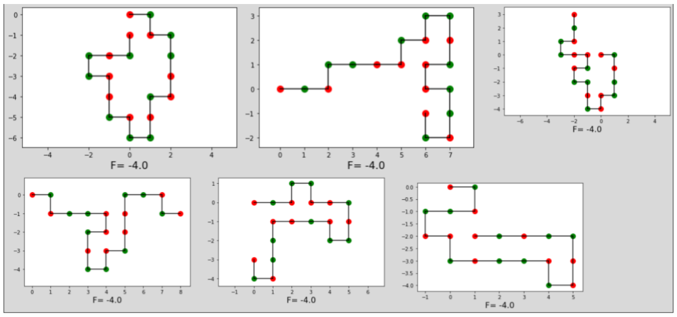


Fig. 4: Local optimum conformations sample from experiment 1.

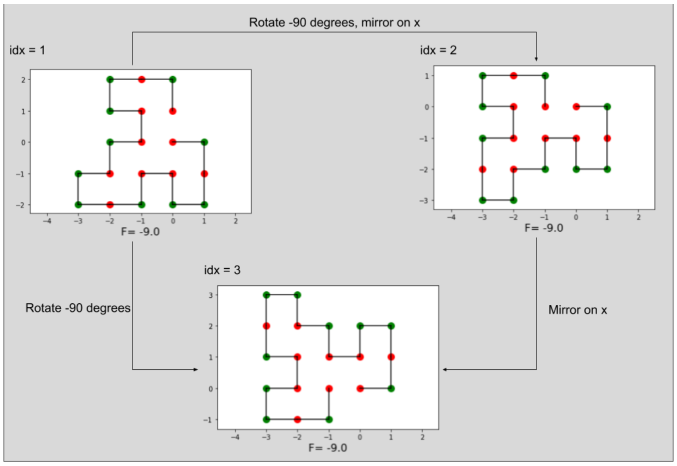


Fig. 5: All global optimum conformations of experiment 2.

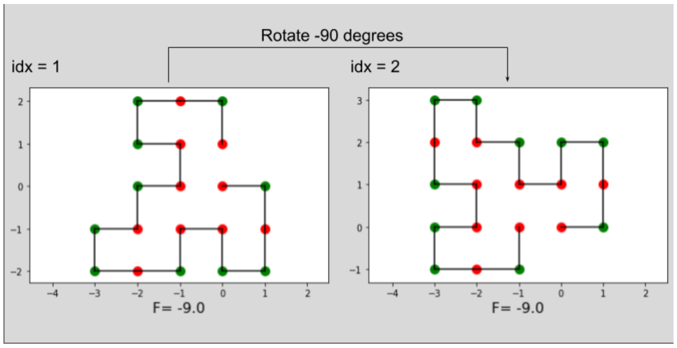


Fig. 6: All global optimum conformations of experiment 3.

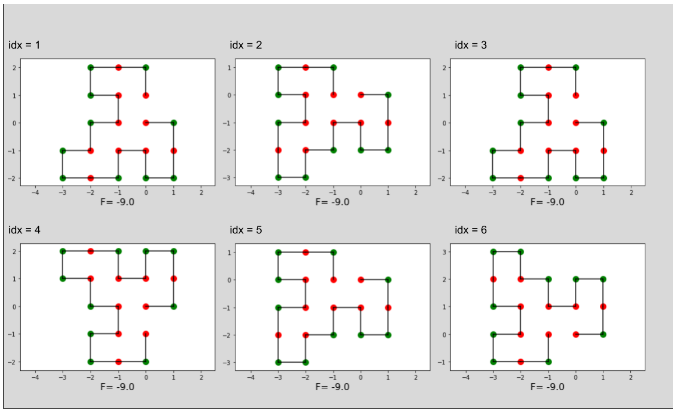


Fig. 7: All global optimum conformations of experiment 4.

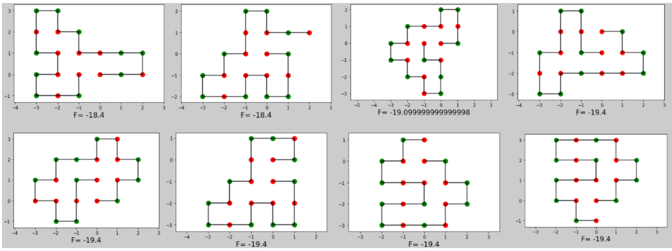


Fig. 8: 8 conformations with lowest energy from experiment 5.

TABLE III: Minimum energy found for unimodal protein folding.

Sequence	$F(\mathbf{x})$
20	-9
24	-9
25	-7
36	-10
48	-14
50	-13
60	-21
64	-17

TABLE IV: DE settings for multimodal experiments.

DE Settings	**Value**
Scheme	rand/1/bin
Mutation	0.8
Crossp	0.7
Pop_size	500
Pop_init	Sobol
Max_iter	7500
Cauchy_iter	1000
64	-17

TABLE V: Cluster settings for multimodal protein folding.

No.	$m_{cluster}$	$epsilon$	$delta$	$k_{cluster}$	r	$theta$
1	300	1	0.1	1	0.95	45
2	600	0.5	0.1	1	0.95	45
3	300	0.5	0.1	10	0.95	45
4	1000	0.5	0.1	10	0.95	45
5	300	0.5	0.1	10	0.95	45
6	300	0.5	0.1	10	0.95	45

model that converts physical representations of proteins into an objective function. In this research, protein folding is modeled as both single target optimal point or unimodal and multimodal optimization. Unimodal protein folding were performed as a benchmark to the multimodal problem. Multimodal protein folding uses the seq-20 as a benchmark for achieving optimal global points. Several experiments were carried out which produced a number of different local optimal and global optimal points. Based on the analysis, for each experiment, although there are several different optimal global points, it turns out that these points have equivalent protein conformations, hence there is only one global optimal point for a protein sequence. On the other hand the local conformations found may be beneficial for meta-stable structure analysis.

References

[1] Anfinsen, C. B. (1973). Principles that Govern the Folding of Protein Chains. Science, 181(4096):223-230.

[2] Levinthal, C. Are there pathways for protein folding? J. Chem. Phys., 65:44-45, 1968.

[3] Lau, K.F., Dill, K.A. (1989). A lattice statistical mechanics model of the conformational and sequence spaces of proteins. Macromolecules. 22 (10): 3986-97.

[4] Blazewick, J., Dill, K.A, Lukasiak, P., et al. (2004). A Tabu Search Strategy For Finding Low Energy Structures Of Proteins In Hp-Model, CMST, Vol. 10, pp. 7-19.

[5] Wong, K.C., Leung, K.S., Wong, M.H. (2010). Protein Structure Prediction on a Lattice Model via Multimodal Optimization Techniques. GECCO'10, July 7-11, 2010, Portland, Oregon, USA.

[6] Lopes, H., Bitello, R. (2007). A Differential Evolution Approach for Protein Folding using a Lattice Model. Journal of Computer Science and Technology 22(6): 904-908 Nov 2007.

[7] Jahn, R.T., Radford, S.E. (2008). Folding versus aggregation: Polypeptide conformations on competing pathways. Arch Biochem Biophys. 2008 Jan 1; 469(1): 100-117.

[8] Ghosh D.K., Ranjan A. (2020). The metastable states of proteins. Protein Sci. ;29(7):1559-1568. doi:10.1002/pro.3859.

[9] Lee, C., Park, S. H., Lee, M. Y., Yu, M. H. (2000). Regulation of protein function by native metastability. Proceedings of the National Academy of Sciences of the United States of America, 97(14), 7727–7731. https://doi.org/10.1073/pnas.97.14.7727

[10] Storn R, Price K. (1997). Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces. J Global Optimization;11:341-59.

[11] Bradley, R. E., Sandifer, C. E. (editors). (2009). Cauchy's Cours d'analyse. An Annotated Translation, pp. 85-on.

[12] Sidarto, K.A., Kania, A. (2015). Finding All Solutions of Systems of Nonlinear Equations Using Spiral Dynamics Inspired Optimization with Clustering. Journal of Advanced Computational Intelligent Informatics. Vol.19 No.5, 2015.

[13] Sidarto, K.A., Kania, A., Sumarti, N. (2017). Finding Multiple Solutions of Multimodal Optimization Using Spiral Optimization Algorithm with Clustering. Mendel. Soft Computing Journal, Volume 23, No.1, June 2017, Brno, Czech Republic.

[14] Joe, S., Kuo, S. Y. (2008) Constructing Sobol sequences with better two dimensional projections," SIAM J. Sci. Comput., Vol.30, pp. 2635-2654.

[15] Tamura, K.,Yasuda, K. (2011). Spiral Dynamics Inspired Optimization. J. of Advanced Computational Intelligence and Intelligent Informatics (JACIII), vol. 15, No. 8, pp. 1116-1122.

[16] Stéfan van der Walt, S. Chris Colbert and Gaël Varoquaux. (2011). The NumPy Array: A Structure for Efficient Numerical Computation, Computing in Science Engineering, 13, 22-30, DOI:10.1109/MCSE.2011.37

[17] K. Jarrod Millman and Michael Aivazis. (2011). Python for Scientists and Engineers, Computing in Science Engineering, 13, 9-12, DOI:10.1109/MCSE.2011.36

[18] J. Burkardt and C. Chisari, Sobol sequence implementation in python, Retreived from https://github.com/naught101/sobol_seq.

[19] Unger, R., Moult, J. (1993). Genetic Algorithms for Protein Folding Simulations. J. Mol. Bid. (1993) 231, 75-81.

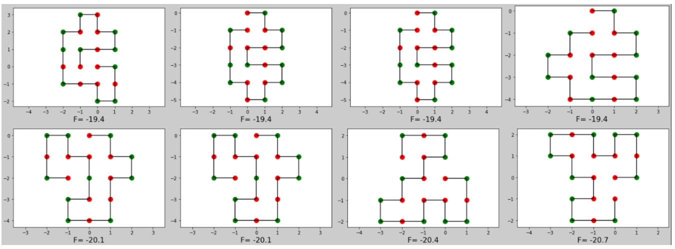


Fig. 9: 8 conformations with lowest energy from experiment 6.

TABLE VI: Result of multimodal protein folding experiments.

Experiment No.	**Found Conformations**	**Feasible Conformations**	$[F(\mathbf{x}_1), F(\mathbf{x}_2), ...]$	**Execution Time (s)**
1	30	12	[-3. -4. -4. -4. -4. -4. -4. -5. -6. -7. -7. -8.]	9195.395307
2	65	45	[-1. -1. -1. -1. -1. -1. -1. -2. -2. -2. -2. -2. -3. -3. -4. -4. -4. -4. -5. -5. -5. -6. -6. -6. -6. -6. -7. -7. -7. -7. -7. -7. -7. -7. -7. -7.-7. -7. -7. -8. -8. -8. -9. -9. -9.]	26773.32246
3	27	15	[-1. -2. -2. -2. -3. -4. -4. -4. -4. -5. -6. -7. -8. -9. -9.]	10209.01037
4	124	70	[-1. -1. -1. -1. -2. -2. -2. -2. -2. -2. -2. -3. -3. -3. -3. -4. -4. -4.-4. -4. -4. -4. -4. -5. -5. -5. -5. -5. -6. -6. -6. -6. -6. -6. -7. -7.-7. -7. -7. -7. -7. -7. -7. -8. -8. -8. -8. -8. -8. -8. -8. -8. -8. -8.-8. -8. -8. -8. -8. -8. -8. -8. -8. -8. -9. -9. -9. -9. -9. -9.]	51677.73747
5	43	28	[-4.3 -4.6 -8.9 -8.9 -9.2 -9.9 -10.9 -11.5 -13.5 -14.5 -15.8 -17.1 -17.1 -17.1 -18.1 -18.1 -18.1 -18.1 -18.1 -18.4 -18.4 -18.4 -19.1 -19.4 - 19.4 -19.4 -19.4 -19.4]	25679.00976
6	51	36	[-1. -2.3 -2.3 -3. -3.3 -3.3 -3.3 -4.3 -4.6 -4.6 -6.3 -6.6 -8.6 -10.2 -10.2 -11.2 -11.2 -13.2 -13.2 -16.5 -16.8 -18.1 -18.1 -18.1-18.1 -18.1 -18.1 -18.8 -19.4 -19.4 -19.4 -19.4 -20.1 -20.1 -20.4 -20.7]	25189.4605