ISSN No. 0976-5697

Volume 2, No. 5, Sept-Oct 2011

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Parallel Recombinative Simulated Annealing Technique for PSP using the Subsequence Secondary Structures: A Novel Protein Fold Lock Model

Sudhanshu Shrivastava*, Garima Srivastava and Navita Shrivastava
Dept. of Computer Science, Awadhesh Pratap Singh University, Rewa (M.P.), India
sudhanshu_nec@yahoo.com
garima.gs17@gmail.com
navita.srivastava@gmail.com

Abstract: The Subsequence Secondary Structures are a set of regular patterns (common motif) frequently occurring in the amino acid sequence of proteins [9]. These motifs are seen to be conserved among species, thus, forming a group of functionally related sequences, sharing a specific biological function. Determination of compact groups of these common patterns is a prerequisite to ably simulate a protein structure prediction algorithm (PSP). In this work, a novel parallel recombinative simulated annealing technique is proposed for an efficient prediction of minimum energy structure on a HP lattice model. The implementation comprising of hybrid heuristics, combines the population based evolutionary Genetic Algorithm and the cooling scheme of Simulated Annealing amid provision of protein subsequence secondary fold lock. A GUI implementation is done for user based parameter testing with 5 lock options={α1,α2,β,loop,core} derived from [5] [7]. The empirical results are compared with the existing EMC method [5] on similar datasets. It is found that the PRSA technique outperforms its ancestor with drastic improvement in computational speed to reach the minimum energy protein structure.

***Keywords*:** Protein Secondary Structure, Parallel Recombinative Simulated Annealing, Protein Fold Lock Model, Genetic Algorithm, Simulated Annealing.

I. INTRODUCTION

Denatured proteins automatically refolds from their random disordered state into a well-defined unique structure, where its biological activity is completely restored. Thereby, protein sequences fold into a unique native state within seconds. As pointed out by Levinthal [8] and Wetlauer [16], the number of possible structures of a polypeptide chain is too large to involve an exhaustive search because proteins folds too fast, by at least tens of orders of magnitude. This apparent contradiction leads to the Levinthal paradox: How can a protein find a globally optimal state without a global exhaustive search? Consequently, the protein folding problem is a question of what is the physical basis of cooperativity by which proteins avoid exhaustive searching of structural space [3]. So, the protein folds to its native state according to a relatively small number of pathways, which means that it folds by a specific sequence of molecular events from the unfolded random coil to a uniquely folded metastable state [8], [4].

To understand the mechanisms of protein folding, it is crucial to characterize the structures of folding intermediates. The intermediate protein fold cooperativity is mainly driven by two types of interactions: First, the secondary structure, e.g. the helix, is found by local interaction by which each individual tetrapeptide in the sequence finds a hydrogen-bonded helical structure, and second, the non-local interactions by which a compact hydrophobic core is formed [5]. However, there is controversy, whether the secondary structure of a protein forms before the growth of the hydrophobic core, as was postulated by the framework model [6] or whether or the hydrophobic residues collapse to form a compact unfolded state or a molten globule, on which the secondary structure grows, as was postulated by the hydrophobic collapse model [3]. In either case, the conserved secondary structures forms the basis of the protein folding pathways. Convinced with these observations of cooperativity, Liang and Wong, 2001 [5], employed secondary structures in exploring the structural space of a protein to speed up their simulation algorithm. The idea was further developed by Bui and Sundarraj, 2005 [7], in their Secondary Structure Genetic Algorithm (SSGA), being tested and evaluated by many others since then [1][9][10][12][13][19][20].

Motivated with the extensive literature and work done on computational algorithms employing secondary structures, this work introduces a Hybrid Genetic Algorithm called as the Parallel Recombinative Simulated Annealing, with a new protein fold lock concept where, a given length of the sequence can be locked to maintain a particular secondary structure throughout the execution procedure. This specific length position (the subsequence secondary structure) is the conserved domain of DNA translation process. Following steps are used to in order to speed up the simulations of protein folding:

(i) Identify the subsequences which will possibly fold to secondary structures in the native state of a given protein sequence;
(ii) Use a hybrid heuristics with integrative advantage of the combined techniques, in order to resolve complexity issues with speedy executions.
(iii) Perform sampling on the constrained structural space where some subsequences are subject to possible secondary structures.

The experimental results of proposed algorithm assuming constraint search space show the rapid iterative simulations and fast convergence to the global minimum (optimum solution).

II. MATERIALS AND METHODS

Hardware used: Processor - Intel core 2 duo i3 380M Pentium µp, 2.53 GHz ; RAM 4 GB ; Hard Disk 320 GB.
Software Used : Operating System –Windows XP.
Implementation Platform: ANSI C (Initial software prototype) / MATLAB 7.0 (final test module).

III. THE PROTEIN STRUCTURE PREDICTION PROBLEM (PSP)

The Protein Structure Prediction Problem (or The Protein Folding Problem) is defined as the prediction of the three dimensional native (ground energy) structure of a given protein from its amino acid chain (primary structure). It is the study of the way in which a protein - a sequence of amino acid residues - will 'fold' into its natural state hence determining the fold transitions. The problem is simply stated; however, solving is intractable [14], and is considered as a Grand Challenge problem [15].

For simplicity, thus, it is widely studied under 2D HP lattice Model [2], where it can be viewed abstractly as the problem of optimization from a pool of probable structure orientations in the presence of constraints. The optimization sequence may be benchmarks, further extended to real proteins. Several investigations have been done till date discussing various computing algorithms to optimize the sequence in order to find its ground state, thus, identifying transition of the folding stages to ascertain the natural existing structure of a protein [23][24]. Heuristic Algorithms, that can learn and exploit the search space regularities in the form of probabilistic derivations with a comprehensive decision making, gives an optimum choice to solve the NP hard PSP problem. Though being the best, the computational efficiency of heuristics in terms of space and time complexity still remains a big question. The use of subsequence secondary structures within the framework of hybrid heuristics can speed up simulations, hence increasing the computational efficiency for the same.

IV. PERSISTENCE OF SUBSEQUENCE SECONDARY STRUCTURES IN PROTEINS

Many biological sequences which belong to a group of functionally related genes or proteins, usually contain a number of *biologically active sequence patterns* shared among some (sometimes all) members of the functional group. While it is not clear yet, exactly how these active functional DNA motifs, (such as transcription factor binding sites), have evolved in complex organisms, *evolutionary algorithms with their heuristic nature* provide relatively simple paradigms that could be close to some possible ways of evolution in linking the species. These functional DNA motifs are the part of a protein sequence - *the sub-sequence patterns*, which forms the secondary structure namely, α helix and β sheet. It is found that such sub - sequence patterns are conserved among species giving rise to some similar biological processes (cell activities) in all [25]. An example is that of a gene called *"Eyeless"* found in fruit fly - *Drosophila Melanogaster*. The absence of this gene results in non development of eyes in the fly.

Another gene called as *"Aniridia"* , (responsible for eye formation) is found in humans. Now if *Aniridia* is inserted in place of absent *Eyeless* gene, it results in development of eyes in the fly. Hence its concluded that there exists a function similarity between the genes having similar sequence and is found to be conserved during course of evolution of organisms. Persuaded with the fact, it can be said that the sub sequence forming secondary structures persists among the evolutionary linked species, giving rise to the conserved domains among proteins.

Hence these set conserved patterns need not be optimized since they remain fixed throughtout procedure of achieving a lowest energy stable protein structure. As such, avoiding the search for conserved sub-sequences, makes the total length of complete protein sequence smaller. This significantly reduces the optimization stages, which are otherwise un-necessarily scanned wasting time and computational evaluations.

While it is not implied that conserved sub-sequence structures exactly follows the same pattern always, the algorithm is developed to exploit the idea with abstraction of most probable recurrent motif groups (sub-sequence ranges) which may prove efficient in finding the optimum solution in a reduced computational time.

V. PSP USING CONSERVED SECONDARY STRUCTURES

The most crucial aspects of computer implementation of a protein structure prediction (PSP) algorithm includes the representation of protein sequence vector [9], the model constructs with parameter setting, and the locking rules, for computation of sequence with conserved set of motifs in order to determine the native (ground energy) structure[13]. The subsections below describes the protocols for model construction followed by pseudocode steps of the implemented algorithm and the briefs of the designed graphical user interface.

A. A Novel Protein Fold Lock Model :

Secondary structures are mainly formed through hydrogen bonds between backbone atoms. There are three types of backbone structures: α-helices, β -sheets and loops. The α-helices and β -sheets are preferably located at the core of the protein, whereas loops are rather found in outer regions. To simulate these frequent conserved DNA α / β motifs, this experiment presumes the following assumptions -

(i) The implementation model is a two dimensional hydrophobic - hydrophilic (polar) lattice grid model [2]. This model, though being simple, has a potential to reveal several computational aspects of search algorithms without involving complex data structures for its implementation. Thus, one can concentrate on testing the algorithm performance rather than on programming language constructs[21].

(ii) The length and number of secondary sub-sequences are *user-driven*. Unlike Liang-Wong [5] and Bui-Sundaraj [7], where a secondary structure is a conformation of a sub-sequence consisting only of hydrophobic H-H residues, there is neither a restriction on what kind of amino acid sub-sequence (hydrophobic or polar) to choose nor its position within the complete amino acid sequence. Hence, *the implementation should allow to observe the development of a general constrained structures across the generations.* The *Figure 1* shows the five possible formations that can be selected as sub-

sequences for a certain range of arbitrarily combined polar and hydrophobic amino acids, namely α helix with direction 1/2, Extended β -sheet, loop and compact core structure The assumption falls close to the Liang and Wong [5] secondary sub-structures of H-H α, H-H β and Bui Sundarraj's longest H-H sequence. The work of Bui-Sundarraj [7] introduced a secondary structure library for the longest subsequence of hydrophobic residues. Due to the number of possible secondary structures even for a small subsequence of hydrophobic residues, they used a genetic algorithm to systematically evolve the secondary structures which are then used as building blocks to evolve the best structure for the given input sequence. Additionally, they put two constraints on the structures: The secondary structures of the hydrophobic sub-sequences are required to be symmetric to either one of the two lattice axes,and end lattice sites should have at least one unoccupied lattice neighbour. We employed a contrary scheme with more flexible lock options.

(iii) The selected ranges of secondary sub sequence structures once locked in the initial phase, remains fixed throughout the run of the algorithm.

(iv) The Mutation Operator : When the point of mutation is randomly selected and this point happens to be in a locked range, another point is randomly chosen until it falls outside the locked sub-sequences.

(v) The Crossover Operator: Since all individuals, including the two selected parents, are locked at the same points, two structures can still be recombined even if the cutpoint happens to be within a locked subsequence. However, in this case the second part is not turned to ensure that the newly created individual contains the respective sub-formation.

(vi) The Cooling Factor : Cooling is done slowly according to the Boltzman distribution, with acceptance probability as $1 / \{1+\exp[\Delta E / T]\}$, where, $T=\Delta T/K$

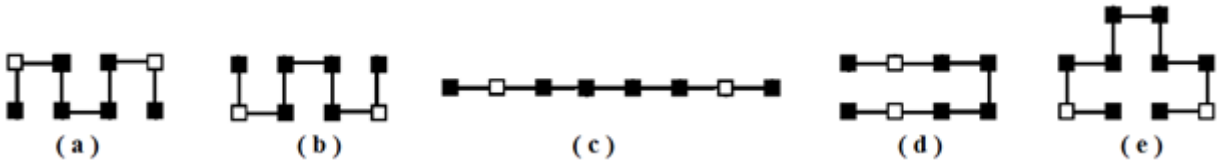

Figure 1 : Secondary structures that can be chosen as locked sub-sequences for an arbitrary range of amino acids: (a) α helix with direction 1 (b) α helix with direction 2 (c) Extended β -sheet (d) a single fold - loop (e) compact core structure.

In this experiment, in contrast with Liang-Wong EMC and Bui-Sundarraj SSGA, *several ranges can be chosen, whereby also the different kinds of sub-structures can be combined*. The size or position of the ranges is irrelevant as long as the different ranges do not overlap. Thus, though not quite meaningful, it is theoretically possible to select the complete sequence and assigned a certain structure to it. All individuals of the population are initialized with these sub-sequences. The selected ranges are then *locked* which means that they will keep this form for the course of the algorithm.

B. The PRSA Technique :

Simulated Annealing and Genetic algorithms are naturally motivated, general purpose optimization procedures and possess many similarities. The main difference between the two, is that GA s are naturally parallel algorithms which search from a population of points, where as SA searches from a single point and is not easily run on parallel processors. The advantage of SA over GA is that we can exercise extreme control over convergence in SA while GAs employ no such concept of cooling so its convergence is not easily forestalled or controlled. A better approach is where an effective combination of GA and SA is attempted. This is called Parallel Recombinative Simulated Annealing (PRSA)[17]. It retains the desirable asymptotic convergence properties of simulated annealing, while adding the populations approach and recombinative power of genetic algorithms.

a. Genetic Algorithm - The GA Step :

Genetic Algorithms (GAs) are adaptive heuristic search algorithm premised on the evolutionary ideas of natural selection and genetic transformation. Genetic Algorithms are designed for intelligent exploitation of a random search within a defined search space to optimize a solution. In the present context, GA starts with a population of individual chromosomes - the probable variations of initial structure of the amino acid sequence. Three genetic operators select, recombine and mutate the individuals to generate a number of offsprings - the changed structures. An iterative loop evaluates the fitness (energy) of all, retaining the best parents/offsprings. A new generation is formed with improved fitness of the initial population. The algorithm proceeds repeating the process generation by generation, finally converging to the optimum (lowest energy) structure.

b. Simulated Annealing - The SA Step :

The idea for Simulated Annealing (SA) based optimization originally given by *Kirkpatrick* et.al. [11] relies on the principles of thermodynamics and resembles the process in which a solid material is first melted and then allowed to cool by slowly reducing temperature. This approach is very suitable for discrete combinatorial optimization problems, such as the protein structure prediction problem[12]. A description of the SA procedure is as follows - Assume that a search algorithm is looking for a configuration that minimizes a certain cost function ***E***. The steps of SA can then be formulated as given below -

Step 1 - Starting off at an initial configuration, a sequence of iterations is generated.

Step 2 - Each iteration consists of the random selection of a configuration from the neighbourhood of the current configuration and the calculation of the corresponding change in cost function ΔE.

Step 3 - The neighbourhood is defined by the choice of a generation mechanism, i.e. a ``prescription'' to generate a transition from one configuration into another by a small perturbation.

Step 4 - **If** change in cost function (ΔE) is negative, the transition is unconditionally accepted;

If the cost function increases, the transition is accepted with a probability based upon the *Boltzmann distribution* :

P (Δ E)~exp(--ΔE / kT), where ***k*** is a constant and the temperature ***T*** is a control parameter.

Step 5 - This temperature ***T*** , is gradually lowered throughout the algorithm from a sufficiently high starting value (i.e. a temperature where almost every proposed transition, both positive and negative, is accepted) to a ``freezing'' temperature, where no further changes occur.

In practice, the temperature is decreased in stages, and at each stage the temperature is kept constant until thermal quasi-equilibrium is reached. The whole of parameters determining the temperature decrement (initial temperature, stop criterion, temperature decrement between successive stages, number of transitions for each temperature value) is called the ***cooling schedule.***

Consequently the four key ``ingredients'' for the implementation of simulated annealing are:

i. the definition of configurations;
ii. a generation mechanism, i.e. the definition of a neighborhood on the configuration space;
iii. the choice of a cost-function;
iv. a cooling schedule.

C. Parallel Recombinative Simulated Annealing - The Prediction Algorithm :

The working of Parallel Recombinative Simulated Annealing (PRSA) technique can be imagined as if several copies of SA Algorithm are running in Parallel, using population heuristics of GA with mutation as a neighborhood operator, and crossover to recombine independent solutions[18]. The new copies are accepted according to the Metropolis criterion. The disruption by crossover and mutation is not a problem in PRSA due to the non destructive nature of the algorithm. Convergence is strictly regulated by a cooling schedule. Because of its slow cooling and diversity maintaining operators , PRSA has little problem with genetic drift.

// Pseodocode : The PRSA Algorithm //

1. Set : Population with initial configuration C_{pop}
   Temperature with initial value To , final value T_N
2. Repeat till Ti <= T_N,
   Make random perturbation to system, changing to new population from current (C_{pop} → C* $_{pop}$)

2.1 Randomly pair all population elements
2.2 For each such pair of parents
  2.2.1 Generate 2 children using a recombination operator (crossover), followed by a neighborhood operator (mutation)
  2.2.2 Calculate energy for parents and children
  2.2.3 Apply metropolis algorithm to decide whether or not to accept the change -
  - hold trials between child and parent
  - keep parent with the probability , $1 / \{1+\exp[(E_{parent} – E_{child})/T] \}$

2.3 Increase counter , i = i + 1
2.4 Lower temperature , $Ti = T_O – i * (T_O – T_N) / N$

3. End

PRSA starts with a very high temperature and generates a large number of random structures (initial population). A small perturbation is made to form new population of structures. Crossover and mutation are applied. Individuals are retained as determined by Boltzmann trials. Subsequently, the temperature is lowered by a small amount, and creation of a new generation starts. If the 'temperature cooling schedule' is made slower, it increases probability of finding the optimum solution - the global minima. The cost, however, is a longer computation time. The SA step is thus, a bit subjective but, has a ability to escape local minima. Another advantage is its very simple implementation. SA is sometimes called a "biased random walk". This due to the fact that iteration steps are made randomly and they do not contain an 'intelligent' move as most of the other optimization techniques, so it does not require the knowledge of the search space[22]. A balance is made by the GA step which induces the parallelism with population of structure orientations and intelligent decision making for fast convergence[20]. On the whole PRSA inherits natural escape from local minima with temperature control of SA and generational improvement of fitness value with smart moves of GA.

D. GUI Implementation on Standard Test and User Defined Amino Acid String Instances :

With intent to provide an implementation that is suitable for user interactions, a graphical implementation is done with user options. The user is provided a number of benchmark sequences with the corresponding optimum energy. Moreover, provision is given for an arbitrary sequence which can be defined by the user. The program allows to determine sub-sequences to be locked, that means to maintain a certain structure, namely a α helix (dir 1 or 2), a β -sheet, a single fold and compact core. If no sub-sequence is locked, the program is executed with the normal PRSA algorithm. In addition to the size of the population and the number of iterations to be performed, the user can also select how many individuals of a population should be displayed as graphical representation. Though this value can still be varied after running the algorithm.

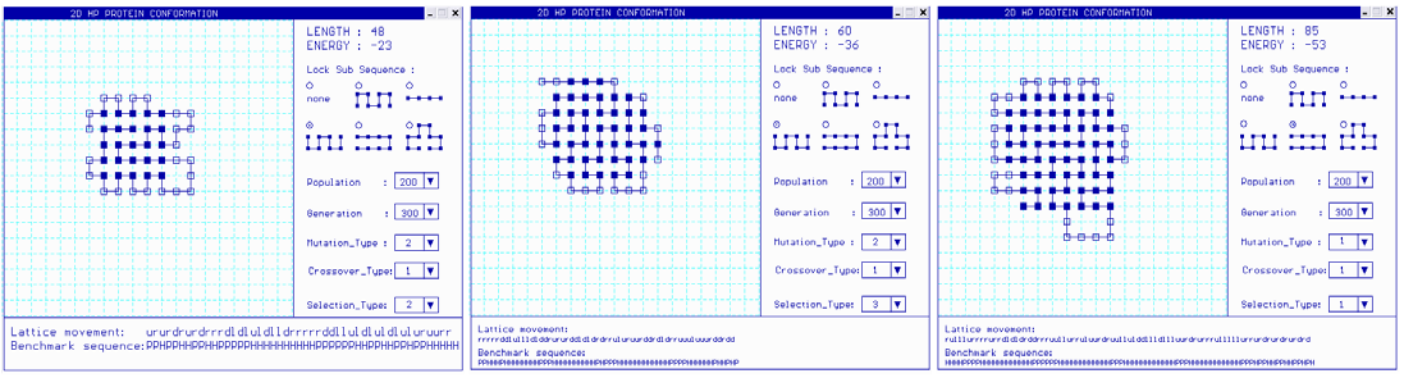

Figure 2 : Ground energy structure (native state) of protein length (a) L = 48 (b) L = 60 (c) L = 85 found by the PRSA sampler with sub-sequence secondary fold lock constraints.

By sampling on the constrained sructural space, PRSA folds the L(48,64,85) sequence length rapidly to their putative ground energy states. The computational results are summarized in *Table I*. The putative ground states found by the constrained PRSA sampler for the 48, 64 and 85 sequence length are shown in *Figure 2*.

With constraint (a) and (c), the PRSA folds the sequence rapidly to native state with energy -23, as compared to EMC taking higher number of evaluations. With slightly stronger constraints (e),(f) and (g), EMC folds the sequence to many ground states in perturbed evaluations. With (g), EMC fails to reach the ground energy state, while PRSA is found as a stronger contender with continuous success and lower number of evaluations for similar constraints. For example, in one run with constraint (f), EMC scans 44,029 valid structures before ground energy state was found in contrast with PRSA which takes only 32,723 evaluations. Though in one case with lock constraint range (d), PRSA was not able to converge to ground E-42, the overall performance is more efficient than EMC in terms of computational time and speed of convergence. Two string instances are included in the table with a user defined lock range of a core, a helix and a loop for length 24 and 25. Comparison could not be done due to unavailability of similar data in [5].

VI. CONCLUSION AND FUTURE PROSPECTS

For PRSA, two points needs to stress are its flexible structure and learning capability. The structure of PRSA is "flexible" in the sense that any efficient move developed for protein folding can be incorporated as a mutation operator or a crossover operator. The "learning" capability of PRSA refers to its ability to modify its behavioral tendency by experience. The use of population heuristics makes it possible for PRSA to learn from its historical samples and at the same time, controlled cooling schedule leads to appropriate folding pathways guiding towards the putative ground energy states. In the SA step of PRSA, the simulation at high temperatures can help the system make a much global exploration over the whole sample space. At high temperatures, random mutations are easily accepted, hence a high energy structure will, climb up the temperature ladder, consequently it will be eliminated from the population. While a low energy structure will climb down the temperature ladder.At low temperatures, random mutations are difficult to accept and low energy structure will be stored there for a relatively long time period. The iterative process of the GA step with integrated genetic operators promotes low energy structures to have a maximum spread (number of offsrings one can produce) thus, improving the overall energy fitness vector of chromosomes. Hence the PRSA technique proceeds learning from its historical samples.

We showed that the PRSA can be effectively applied to simulations of protein folding on lattice models using the sub sequence secondary fold locks. In all cases it did better than the Evolutionary Monte Carlo, in terms of both reduced computational time and number of valid structures scanned to reach the optimum. We also designed a graphical user interface with selection options. The numerical results showed that it is very successful in finding low energy states. Although we have considered only 2D HP models in this work, we stress that the extension to 3D HP and real protein models is straightforward.

Table I - S.No. gives the reference for the constraint range set, (a) for (17-26) , (c) for (33-44) ... ; L is the length of the amino acid sequence, with derived ground energy E_{min} putative from Literature ; Lock constraints specify the ranges which is not considered in the optimization process presuming a fixed secondary structure ; EMC stands for Evolutionary Monte Carlo [5] compared to PRSA[# proposed model]; Number of structures are the valid states scanned by the algorithm before reaching the global minimal energy state. * ground energy is not found /reported, --~ data unavailable / not tested with EMC

S. No.	*Length With E_{min}*	*Lock Constraints (Sub Sequence range of secondary structures)*	*EMC [5] Ground Energy*	*EMC [5] no. of valid structures*	*PRSA # Ground Energy*	*PRSA # No. of valid structures*
(a)	L : 48, E:-23	(17 - 26)	-23	53,263	-23	42,958
(b)	L : 60, E:-36	Without lock constraints	-35*	--*	-36	36,425
(c)	L : 60, E:-36	(33 - 44)	-36	40,334	-36	34,599
(d)	L : 64, E:-42	(1 - 10) ,(55 - 64)	-42	77,287	-41*	65,864
(e)	L : 85, E:-52	(09 - 20) (27 - 38) (42 - 53) (57 - 68)	-52	17,794	-52	10,026
(f)	L : 85, E:-52	(09 - 18) (27 - 36) (42 - 51) (57 - 66)	-52	44,029	-52	32,723
(g)	L : 85, E:-52	(11- 18) (29 - 36) (44 - 51) (59 - 66)	-51*	--*	-52	35,642
(h)	L : 24, E:-9	(11-20) core	--~	--~	-9	7,564
(i)	L : 25, E:-8	(1-8) helix, (18-25) loop	--~	--~	-8	8,500

VII. REFERENCES

[1] Huang Chenhua, Yang Xiangbo and Zhihong He, Protein folding simulations of 2D HP model by the genetic algorithm based on optimal secondary structures GAOSS , Computational Biology and Chemistry Volume 34, Issue 3, Pages 137-142, June 2010.

[2] Hoque T., Chetty M, Sattar A., Extended HP model for protein structure prediction. J Computational Biol. 16(1):85-103, Jan 2009.

[3] Dill Ken A. , Fiebig K.M., and Chan Hue S.,Cooperativity in protein-folding kinetics. Proceedings of the National Academy of Sciences USA, 90:1942–1946, 1993.

[4] Creighton Thomas E. , Experimental Studies of Protein Folding and Unfolding. Progress in Biophysics and Molecular Biology, 33:231–297, 1978.

[5] Liang Faming,Wong Wing Hung, Evolutionary Monte Carlo for protein folding simulations. Journal of Chemical Physics, 115(7), 2001.

[6] Kim Peter S. and Baldwin Robert L. , Intermediates in the Folding Reactions of Small Proteins. Annual Review of Biochemistry, 59:631–660, 1990.

[7] Bui Thang N. and Sundarraj Gnanasekaran. An Efficient Genetic Algorithm for Predicting Protein Tertiary Structure in the 2D HP model. In Genetic And Evolutionary Computation Conference, pages 385–392. ACM, 2005.

[8] Levinthal Cyrus, Are there Pathways for Protein Folding? Journal de Chimie Physique et de Physico-Chimie Biologique, 65(1):44, 1869.

[9] Birzele Fabian, Kramer Stefan, A new representation for protein secondary structure prediction based on frequent patterns, Bioinformatics Vol. 22 no. 21, pages 2628–2634, 2006.

[10] Liang Yang, Enli Huang and Vladimir B Bajic, Some Implementation Issues Of Heuristic Methods For Motif Extraction From DNA Sequences, International Journal of Computers, Systems and Signals, Vol. 6, No. 2, 2005.

[11] Kirkpatrick S. , Gelatt C. and Vecchi, M. Optimization by Simulated Annealing, Science, 220(4598): 671-680, 1983.

[12] Xiaomei Li, Protein Folding Based On Simulated Annealing Algorithm, Proceedings of Third International Conference,ICNC'07 on Natural Computation, 2007.

[13] Gabriela Ochoa, Gabi Escuela, and Krasnogor Natalio, Incorporating Knowledge of Secondary Structures in a L-System-Based Encoding for Protein Folding, Lecture Notes in Computer Science (LNCS), Springer-Verlag, vol. 3871, pp. 247–258, 2006.

[14] Khimasia M. M. L., NP complete problems, http://www.tcm.phy.cam.ac.uk/mmlk2/report13/node31.html , 1996.

[15] M. Committee on Physical and E. Sciences. "Grand Challenges 1993: High Performance Computing and Communications.", Office of Science and Technology Policy, 1982.

[16] Wetlaufer Donal B. , Nucleation, Rapid Folding, and Globular Intrachain Regions in Proteins. Proceedings of the National Academy of Sciences USA, 70(3):697–701, 1973.

[17] Mahfoud S.W., and Goldberg David E. ,Parallel Recombinative Simulated Annealing: A Genetic Algorithm, Parallel Computing, Volume 21, Issue 1, Pages 1-28 January 1995.

[18] Kurbel K., Schneider B., Singh K., Solving optimization problems by parallel recombinative simulated annealing on a parallel computer-an application to standard cell placement in VLSI design, Volume: 28, Issue: 3, Pages: 454-461, 1998.

[19] Lopes H.S., Scapin M.P., A Hybrid Genetic Algorithm for the Protein Folding Problem Using the 2D HP Lattic Model.Studies in Computational Intelligence, Vol.92. pp.121-140, 2008.

[20] Haijuan Guo, Qiang Lu, Jinzhen Wu, Xu Huang, Peide Qian, Solving 2D HP Protein Folding Problem by Parallel Ant Colonies, 2nd International Conference on Biomedical Engineering and Informatics, 2009. BMEI '09, pp. 1-5, 2009

[21] Fidanova Stefka, An Improvement of the Grid-based Hydrophobic-hydrophilic Model, INT. J. BIOAUTOMATION, 14(2), pp. 147-156, 2010.

[22] Simons K., Kooperberg C., Huang E., and Baker D., "Assembly of Protein Tertiary Structures from Fragments with Similar Local Sequences Using Simulated Annealing and Bayesian Scoring Function," Journal of Molecular Biology, Vol. 268, pp. 209-225,1997.

[23] Jones D.T., "Predicting Novel Protein Folds by Using FRAGFOLD" Proteins: Structure, Function, and Genetics Suppl., Vol. 5, pp. 127-132, 2001.

[24] Kolinski A. and Skolnick J., "Reduced Models of Proteins and their Applications" Polymer, Vol. 45, pp. 511–524, 2004.

[25] Lin Hsin-Nan, Sung Ting-Yi, Ho Shinn-Ying, Hsu Wen-Lian, Improving protein secondary structure prediction based on short subsequences with local structure similarity,BMC Genomics 11(Suppl 4):S4, 2010.

