DFS Based Partial Pathways in GA for Protein Structure Prediction

Md Tamjidul Hoque[1], Madhu Chetty[2], Andrew Lewis[1], and Abdul Sattar[1]

[1] Institute for Integrated and Intelligent Systems (IIIS),
Griffith University, Nathan QLD 4108, Australia
T.Hoque@griffith.edu.au, {A.Lewis,A.Sattar}@griffith.edu.au
[2] Gippsland School of Information Technology (GSIT)
Monash University, Churchill VIC 3842, Australia
Madhu.Chetty@infotech.monash.edu.au

Abstract. Nondeterministic conformational search techniques, such as Genetic Algorithms (GAs) are promising for solving protein structure prediction (PSP) problem. The crossover operator of a GA can underpin the formation of potential conformations by exchanging and sharing potential sub-conformations, which is promising for solving PSP. However, the usual nature of an optimum PSP conformation being compact can produce many invalid conformations (by having non-self-avoiding-walk) using crossover. While a crossover-based converging conformation suffers from limited pathways, combining it with depth-first search (DFS) can partially reveal potential pathways. DFS generates random conformations increasingly quickly with increasing length of the protein sequences compared to random-move-only-based conformation generation. Random conformations are frequently applied for maintaining diversity as well as for initialization in many GA variations.

Keywords: Depth-first search, protein structure prediction, genetic algorithm, lattice model.

1 Introduction

We are seeking to solve the *ab initio* (meaning 'from the origin') or the *de novo* protein structure prediction problem [1]. In an *ab initio* approach, the building of a 3D conformation (structure) is essentially based on the properties of amino acids, where protein is a three dimensionally folded molecule composed of amino acids [2] linked together (called the primary structure) in a particular order specified by the DNA sequence of a gene. Particular folded structures are essential for the functioning of living cells as well as for providing body structure. Protein structure prediction (PSP) is a problem of determining the native state of a protein from its primary structure and is of great importance because three dimensionally folded structures determine the biological function [3] and hence proves extremely useful in applications like drug design [4].

M. Chetty, A. Ngom, and S. Ahmad (Eds.): PRIB 2008, LNBI 5265, pp. 41–53, 2008.

For investigating the underlying principles of protein folding, lattice protein models introduced by Dill [5] are widely used [6]. Protein conformation as a *self-avoiding walk* in the lattice model has been proven to be *NP-complete* [7, 8]. Therefore a deterministic algorithm for folding prediction is not feasible. So, a nondeterministic approach with robust strategies that can extract minimal energy conformations efficiently from these models becomes necessary. Still, this is a very challenging task as there exists an astronomical number of possible conformations even for a very short sequence of amino acids [9, 10].

We have chosen the Genetic Algorithm (GA) as a vehicle for providing solutions to the PSP problem for better performance, where crossover is regarded as the key operation of GA [11]. The core concepts of GAs and their components are often adapted by many PSP solving algorithms for the effectiveness [12-16]. While crossover can be very effective in joining two different potential sub-conformations, it can be repeatedly unsuccessful as the converging conformations (hence the sub-conformations), being compact in nature, leave limited pathways to a valid (i.e., self-avoiding-walk) conformation. This means many potential conformations may be lost, which motivates us to apply partial pathways based on depth first search (DFS) [17] to regain potential conformations, leading to effective PSP solution.

2 Background and Preliminaries

In nature, a protein folds remarkably quickly, requiring between a tenth of a millisecond and one second in general, whereas any algorithm on any modern computer is still unable to simulate this task in anything approaching similar time[11, 18]. For the immensely complex protein structure prediction problem, there are several issues and approaches which are yet to be considered [11, 19, 20]:

First, the energy function, which is a combination of several factors that determines the free energy of a folded protein, is not fully understood. Therefore, existing formulations for energy functions do not suggest any obvious path to solution of the PSP problem.

Second, conformational search algorithms are promising approaches toward this hard optimization problem, but the PSP problem still needs considerable research to find an effective algorithm. The aim of the search is to identify an optimum conformation within a huge and very convoluted search landscape.

Third, Cyrus Levinthal postulated, in what is popularly known as the Levinthal paradox, that proteins fold into their specific 3D conformations in a time-span far shorter than it would be possible for the molecule to actually search the entire conformational space (which is astronomically large) for the lowest energy state [21]. As proteins cannot, while folding, be sampling all possible conformations, therefore folding pathways must exist.

While focusing on the second issue [22-27], we are utilizing DFS strategies, developing novel search algorithms in a form to address the pathway hypothesis. It has been concluded that conformational searching is the bottleneck in protein folding prediction and the observed folding rates have been found to be proportional to the number of microscopic folding routes [28]. These routes can be captured by the crossover operation from suboptimal conformations and then partial DFS can mimic

the existing microscopic path guided by the converging sub-conformation, whereas a crossover operation alone can encounter more collisions [13] (while mating dissimilar converging conformations) before having a SAW conformation and thus often can reject the potential sub-conformation as being unfit when paired with the available counterpart of the crossover portion (from a dissimilar conformation). To determine the effect of DFS in such situations we will rely on empirical results.

2.1 The HP Lattice Model

The simplified HP lattice model [29, 30] is based on *hydrophobicity* [31], dividing the amino acids into two different beads – *hydrophobic* (H) and *hydrophilic* (or *polar* (P)). The model allows HP protein sequences to be configured as self-avoiding walks (SAW) on the lattice path favoring an energy free state due to HH interaction. The energy of a given conformation is defined as the number of *topological neighboring* (TN) contacts between those Hs, which are not adjacent in the sequence. This contact between two neighboring H residues (or HH contact) is TN and is assigned a value for the potential, termed *interaction potential* which is define as -1 for the regular HP model [32]. Further, the HP interaction and PP interaction potential value is assigned 0, which basically implies that there is no interaction between an H and a P of HP contact or between the Ps of PP contacts.

To define PSP formally, assume for an amino-acid sequence $s = s_1, s_2, s_3, \cdots, s_n$, a conformation *c* needs to be formed where $c^* \in C(s)$, $C(s)$ is the set of all valid (i.e., SAW) conformations of *s, n* is the total number of amino acids in the sequence and energy $E^* = E(C) = \min\{E(c) \mid c \in C\}$ [15]. If the number of TNs (for HH contact) in a conformation *c* is *q* then the value of $E(c)$ is defined as $E(c) = -1 \times q = -q$ and the *fitness function* is $F = -q$. The optimum conformation will have a maximum possible value of |F|. In a 2D HP square lattice model (Figure 1), a non-terminal and a terminal residue, each with 4 neighbours, can have a maximum of 2 TNs and 3 TNs, respectively. In this paper, we will confine ourselves to using the 2D HP square lattice model only, as this model will be sufficient for our needs. However, its simplicity may encourage interested readers to do further research, which would otherwise be very difficult. The HP lattice model is also very popular with the research community [11, 23, 29, 30, 33-39], since it allows easy development, validation and comparison of new techniques for protein structure prediction (PSP) [22-24, 26, 27, 40].

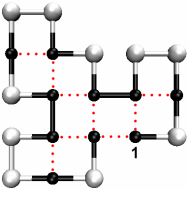


Fig. 1. HP conformation in the 2D HP model shown by a solid line. 2D square lattice having fitness = - (TN Count) = -9. ● indicates a hydrophobic and ● indicates a hydrophilic residue. The dotted line indicates a TN. Starting residue is indicated by a '1' in the figure.

2.2 Complexity of the Lattice Model

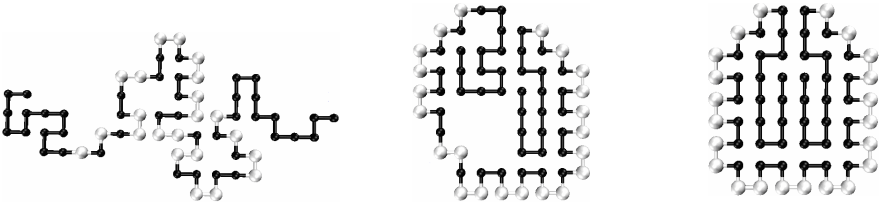
Even if we use this simplified model we have an inordinate number of valid (i.e., SAW) conformations, even for a shorter sequences [9, 10, 41]. For instance, for a sequence of n amino acids, the number of valid conformations is proportional to μ^n, where the connective constant or the effective coordinate number μ, is lattice dependent [10]. Prediction of the optimal conformation using the lattice model is also an *NP-complete* problem [7, 8]. To predict the backbone conformation of the folded protein from its amino acid sequence based on global interactions such as *hydrophobicity*, lattice models are used for approximation [29, 30, 33-35]. For *ab initio* prediction in *Critical Assessment of Structure Prediction* (CASP) [33-35], most successful approaches followed the hierarchical paradigm where the lattice-based, backbone conformational sampling works very effectively at the top of the hierarchy. With further advancement toward all-atom or full modeling from the lattice, the energy functions include atom-based potentials from molecular mechanics packages such as CHARMM, AMBER, ECEPP and so on [42, 43]. Conformational search algorithms built on lattice models, which play a key role in solving PSP, are discussed next.

2.3 Nondeterministic Conformational Search Algorithms

For solving *ab initio* PSP using the lattice model numerous nondeterministic approaches have been investigated: *Monte Carlo* (MC) simulation, *Evolutionary* MC (EMC) [12, 13], *Simulated Annealing* (SA), *Tabu Search* with *Genetic Algorithm* (GTB) [14], *Ant Colony Optimisation* [15], and *Immune Algorithm* (IA) based on *Artificial Immune System* (AIS) [44]. Due to their simplicity and search effectiveness, *Genetic Algorithms* (GAs) [11, 26, 32, 45-48] are the most attractive. They also provided superior performance over MC [46, 47]. The concepts of GAs are also widely adapted within these algorithms. For instance, a new MC algorithm [12] adopted the population-based cut-and-paste (i.e. crossover) operation to achieve higher fitness. The evolutionary Monte Carlo (EMC) [13] algorithm incorporated the evolutionary features of genetic algorithms, such as a population which is updated by crossover and mutation operations. Jiang *et al.* applied the GA with *Tabu* (GTB) search to solve PSP using lattice models [14]. Also, the *conformational space annealing* (CSA) [16, 49] algorithm is based on GA concepts, where the population is renamed as a "bank".

2.4 Focus of the Paper

Given the widespread adaptation of GAs for PSP, the heart of a GA, i.e. the crossover operation, can be made more effective by combining it with DFS which can have a significant positive impact on solving the PSP problem. In a conventional GA, since the optimum conformation is mostly compact physically (see Figure 2), a crossover-based converging conformation suffers from limited pathways and the algorithm increasingly generates invalid conformations. Our hypothesis is that the combination of depth-first search (DFS) with crossover can instead reveal potential pathways in solving PSP. Thus, a repeatedly failing crossover with a congested but potential sub-conformation can be allowed a limited number of pathways for possible candidate crossover counterparts obtained by using DFS if there exists at least one path.



(a): Fitness = -15 (b): Fitness = -33 (c): Fitness = - 42

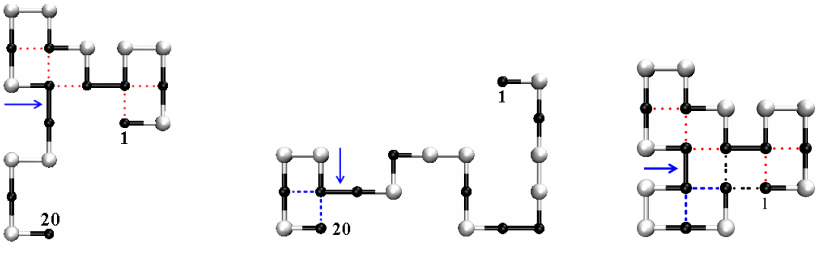
Fig. 2. As the search proceeds the conformation gets more compact: For a typical run, conformations at generation 1, 1434 and 5646 have been shown in (a), (b) and (c) respectively, showing the fitter conformation is relatively more compact.

2.5 Defining the GA Operators for PSP Problem

Here, we define the GA operators for the PSP problem based on the HP lattice model:

Crossover operation: For PSP, this aids the construction of global solutions by the cooperative combination of many local substructures [11]. We particularly followed the commonly-used crossover operation pioneered by Unger *et al.* [46], as illustrated in Figure 3, a single-point crossover. We follow this single-point crossover, since otherwise the converging conformation, being compact in nature, would generate more collisions or invalid conformations [13]. The ability to rotate before joining within the crossover, in addition, provides a mutation-equivalent operation. With the help of *relative encoding* [40], this can be seen easily. For example, if we emulate the crossover in Figure 3 without the rotation, we can write using relative encoding that:

Crossover (a:'LFLLRRLRLLFLRFRLFL', b:'RFFFRFRFLFLRFRLLFL') → would output, c': 'LFLLRRLRLLFL*RLLFL' without the rotation before joining. (Here, '*' indicates an undefined move in relative encoding but here it indicates a non-SAW move.) But, with rotation, the conformation can have SAW, i.e. c: 'LFLLRRLRLLFL**R**RLLFL'.



(a) (b) (c)

Fig. 3. An example of the crossover operation [46]. Conformations are randomly cut and pasted with the cut point chosen randomly between residues 14 and 15. The first 14 residues of (a) are rotated first as needed (as allowed by the degree of freedom by the model configuration) and then joined with the last 6 residues of (b) to form (c), where fitness, F = -9. '→' indicates crossover positions.

Comparing c':'LFLLRRLRLLFL*RLLFL'and c:'LFLLRRLRLLFL**R**RLLFL', it becomes clear that the '*' is replaced by an 'R' after the rotation, which is genotypically a single-point mutation.

Crossover failure: This implies that before joining two parts all, possible, rotated positions at the joining point have been tried but failed to produce at least one valid conformation (i.e., a SAW).

Combination of crossover and DFS: For generating a conformation this implies that a DFS-generated random and partial path has been joined with the first half of the sub-conformation.

DFS after crossover failed: This implies that 'combination of crossover and DFS' has been performed after an occurrence of 'crossover failure'.

Mutation operation: This involves *pivot rotation* (Figure 4) as basically pioneered by Unger *et al.* [46]. We employed single-point mutation to avoid more collisions.

(a) (b)

Fig. 4. An example of the mutation operation [46]. Dotted lines indicate TN. Residue number 11 is chosen randomly as the pivot. For the move to apply, a 180° rotation (among a number of possible degree of freedom defined by the model configuration) alters (a) with F = -4 to (b) F = -9. '→' indicates the mutation residue.

Ordinary random conformation generation: This implies the generation of a SAW conformation based on random-move-only (RMO). In a 2D square lattice model *Left*, *Right* and *Forward* moves are permissible but *Backward* move is prohibited. For a conformation, once a path search has failed after looking in the three possible degrees of the freedom the whole process restarts.

Random conformation generation by DFS: This implies that we apply DFS to generate a SAW conformation. As the DFS proceeds, it stores the possible pathways using a stack-memory [17] and, upon total failure after trying all possible degrees of freedom on a particular location (i.e. lattice point), it can backtrack to restart from the stored options instead of restarting the creation of the whole conformation.

3 Experiments and Results

We carried out experiments to empirically verify our hypothesis that combining DFS with crossover will be advantageous. The simple GA (SGA) applied for PSP is

1. Initialize the fixed size current population (Pop_z) of randomly generated conformations.
2. Obtain a new solution (S_{new}) from the current population by using ***Crossover*** and ***Mutation*** operations at the pre-specified rates (p_c and p_m respectively).
3. Assess the quality or fitness, *F,* of S_{new}.
4. Promote the obtained S_{new}, and elite and untouched chromosomes, to the next generation and assign the new generation as the current population.
5. IF END-OF-SOLUTION is not reached THEN repeat from Step 2.

Fig. 5. Genetic Algorithm for solving PSP problem[†]

(a)	(b)
1. DO single-point ***Crossover.*** 2. IF '***Crossover failure***' = TRUE then 3. REPLACE one of the parents. 4. DO single-point ***Crossover***. END IF	1. DO single-point ***Crossover.*** 2. IF '***Crossover failure***' = TRUE then 3. DO '***DFS after crossover failed***'. END IF
(c)	(d)
1. DO single-point '***Combination of crossover and DFS***'.	1. DO apply option: (a). 2. IF no improvement for 5 consecutive generations, 3. DO apply option: (b). END IF

Fig. 6. ***Crossover*** operation and variation details

illustrated in Figure 5 and the crossover variations with the possible implementation have been shown in Figure 6. As shown in Figure 6, we have experimented with four variations of the crossover operation. *Crossover* (a) (see Figure 6(a)) represents a conventional crossover operation for PSP without DFS. *Crossover*(b) (see Figure 6(b)) applies DFS-based partial path generation with the sub-conformation immediately the sub-conformation fails to join with its counterpart sub-conformation after trying all possible degrees of freedom. *Crossover*(d) (see Figure 6(d)) is similar to *Crossover*(b) in operation but allows more time to a failed crossover to search for a suitable counterpart sub-conformation to match. *Crossover*(c) is a the most dissimilar variation of *Crossover*(d) where, instead of a sub-conformation looking for its counterpart sub-conformation in the population, *Crossover*(c) directly uses DFS to generate the rest of the path to complete the conformation. This alternative was investigated to determine an effective rate of DFS.

The default GA parameters for all experiments were set as population size (Pop_z) to 200, crossover rate (p_c) to 0.85 or 85%, mutation rate (p_m) to 5% and for elitism the elite rate was set to 5% [50, 51].

The fold for longer PSP problems generally has complex energy landscapes [30, 52-57], and hence those sequences will take longer to converge. So we chose those longer sequences to highlight the true benefit of this approach. A maximum of 2000

[†] Terms in **bold** and *italic* are explained in section 2.5.

Table 1. Benchmark protein sequences for 2D HP model

Length	Sequences	Ref.
50	H2(PH)3PH4PH(P3H)2P4H(P3H)2PH4P(HP)3H2	[59]
60	P2H3PH8P3H10PHP3H12P4H6PH2PHP	[59]
64	H12(PH)2(P2H2)2P2HP2H2PPHP2H2P2(H2P2)2(HP)2H12	[59]
85	4H4P12H6P12H3P12H3P12H3P1H2P2H2P2H2P1H1P1H	[58]
100	3P2H2P4H2P3H1P2H1P2H1P4H8P6H2P6H9P1H1P2H1P11H2P3H1P2H 1P1H2P1H1P3H6P3H	[58]

'H' and 'P' in the sequence indicate hydrophobic and hydrophilic amino acids, respectively.

Table 2. Run results of 10 iterations on each PSP sequence (see Table 1 for the sequences). GA runs with four different crossover options (shown in Figure 6), have been compared.

Length	*X(a)*	*X(b)*	*X(c)*	***X(d)***	*CSA*	*UGA*
50	-17.3/-**20**	-17.6/-**20**	-14.5/-17	-**18**/-**20**	-17 / -19	-16.6 / -18
60	-29.2/-**32**	-29.8/-**32**	-27.8/-31	-**30.5**/-**32**	-30.4/-**32**	-29/-31
64	-29.1/-31	-29.3/-31	-25.2/-29	-**32**/-**35**	-29/-30	-27.8/-31
85	-39.4/-44	-39.6/-45	-34.5/-38	-**43.4**/-**46**	-43.2/-**46**	-41.4/-**46**
100	-37.1/-39	-37.6/-41	-30.2/-37	-**38.5**/-**42**	-37.2/-38	-37.4/-40

The format of column entries is 'Average / Minimum'. The *X* implies ***Crossover*** operation. Thus, *X*(a) indicates ***Crossover***(a) as described above, and so on. CSA and UGA indicate Conformational Space Annealing Algorithm [16] and Unger's GA [46], respectively. **Bold** entries indicate the row-wise best values obtained.

generations was allocated for each of the 10 iterations carried out per sequence, per category of experiments. Benchmark PSP sequences shown in Table 1 for the 2D square HP lattice model [5], length ranging from 50 to 100 were used [58, 59]. The results are shown in Table 2.

It may be noted that in Table 2, we include two other algorithms in their generic form: Unger's GA (UGA [46]) and Conformational Space Annealing (CSA) algorithm [16, 49] with our proposed algorithm for solving the PSP problem. UGA has already outperformed many MC variations, as reported in [11, 46]. We emulated UGA in our experiment with the same parameter for cooling, i.e. the cooling temperature was set to 2 at the start and decreased by 0.99 every 200000 steps until the temperature became 0.15.

We abstracted the general form of the CSA algorithm by removing the heuristic-based special moves, keeping the generic form intact, to provide a fair comparison in our experiment. Comparison with CSA algorithm is particularly important for our work, since the CSA approach has recently been applied in the PSP software ROSETTA [33, 60-63]. Both UGA and CSA ran 2000 GA generation equivalent runs per iteration.

4 Discussion of the Experimental Results

We have introduced the concept of finding potential partial pathways using a depth-first search (DFS) strategy when a converging, potential sub-conformation in a crossover failed to find a matching counterpart to produce a valid (i.e., having a self-avoiding-walk) conformation. Crossover variation *X*(c) has the worst result in Table 2. *X*(c) involves applying DFS constantly at the same rate as the crossover operation to generate the other half of the crossover portion, which is misguiding the optimum results more that guiding them. *X*(a) represents the crossover-only approach, that is, crossover with DFS, and *X*(b) is the variant where DFS is applied whenever a crossover fails. *X*(b) is a slight improvement over *X*(a). *X*(d) performed the best, with results comparable to the UGA and CSA algorithms. This is because, in *X*(d), crossover was applied exhaustively by allowing a failed crossover to look for more counterparts to match and when there is no improvement at all in the whole population for consecutive few generations, the failed crossover is combined with DFS to generate the possible potential pathways. It is interesting to note that, in our experiment we find *DFS has zero failure in finding pathways*. Thus, a constantly failing sub-conformation in a crossover operation, which is likely to have few possible pathways, can be salvaged using DFS to unravel the hidden paths effectively. As an alternative to DFS, breadth-first search (BFS) [17] could have been used; however, BFS is both memory and time intensive.

5 Supplementary Applications of DFS in PSP

It is important to remember that ***ordinary random conformation generation***† takes exponential time (fitted curve: $y = 2.8723\ e^{0.0326x}$ with square of coefficient of determination, $R^2 = 0.9832$) with increasing sequence length using the random-move-only

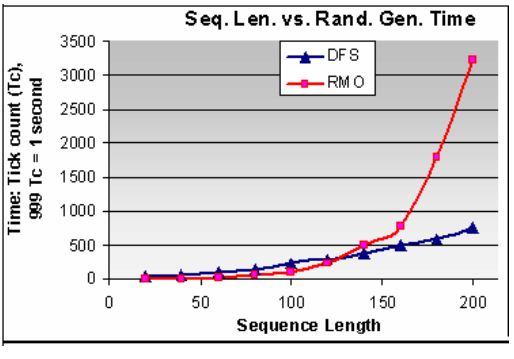


Fig. 7. Random conformation generation: DFS approach versus random-move-only (RMO) approach. An average of 100 iterations is taken for a particular length of a single random conformation generation.

† Terms in **bold** and *italic* are explained in section 2.5.

(RMO) approach. In contrast, the run-time for ***random conformation generation by DFS*** remains quadratic (fitted curve: $y = 0.02\ x^2\text{-}0.5717x\text{+}54.789$, with $R^2 = 0.9996$) (see Figure 7).

The application of ***random conformation generation by DFS*** may have a generally lower impact because totally random conformations are only generated for initialization of the population. To maintain diversity many GA approaches replenish the population a considerable amount and at frequent intervals [64, 65]. For example, Hoque *et al.* have shown removal of chromosomes having 80-90% or greater similarity from a GA population helps it to perform better [64]. After removal it is necessary to replenish the population by random conformations of 20 to 30% in each generation. Thus, in such a case, for longer sequences, ***random conformation generation by DFS*** would make the GA search far more efficient.

6 Conclusions

A depth-first search (DFS) strategy at a low rate has been applied in combination with a powerful crossover operation. Together they revealed convoluted and microscopic pathways in solving protein structure prediction problem. Experiments using a variety of longer, standard benchmark sequences from the literature have demonstrated the efficacy and improved performance characteristics of this approach. The search strategy developed was inspired by the pathway hypothesis. Further work will be directed to exploring the biological significance and relevance of this novel approach.

Acknowledgement

Support from Australian Research Council (grant no DP0557303) is thankfully acknowledged.

References

1. Chivian, D., Robertson, T., Bonneau, R., Baker, D.: AB INITIO METHODS. In: Bourne, P.E., Weissig, H. (eds.) Structural Bioinformatics. Wiley-Liss, Inc., Chichester (2003)
2. Allen, F., et al.: Blue Gene: A vision for protein science using a petaflop supercomputer. IBM System Journal 40, 310–327 (2001)
3. Pietzsch, J.: The importance of protein folding. Nature (2003) (last access, 2007), `http://www.nature.com/horizon/proteinfolding/background/importance.html`
4. Petit-Zeman, S.: Treating protein folding diseases. Nature (last access, 2008), `http://www.nature.com/horizon/proteinfolding/background/treating.html`
5. Dill, K.A.: Theory for the Folding and Stability of Globular Proteins. Biochemistry 24, 1501–1509 (1985)
6. Backofen, R., Will, S.: A Constraint-Based Approach to Fast and Exact Structure Prediction in Three-Dimensional Protein Models. Constraints Journal 11 (2006)

7. Crescenzi, P., Goldman, D., Papadimitriou, C., Piccolboni, A., Yannakakis, M.: On the complexity of protein folding (extended abstract). In: 2nd Intl. conference on Computational molecular biology, pp. 597–603. ACM, New York (1998)
8. Berger, B., Leighton, T.: Protein Folding in the Hydrophobic-Hydrophilic (HP) Model is NP-Complete. Journal of Computational Biology 5, 27–40 (1998)
9. Schiemann, R., Bachmann, M., Janke, W.: Exact Enumeration of Three – Dimensional Lattice Proteins. In: Computer Physics Communications, p. 166. Elsevier Science, Amsterdam (2005)
10. Guttmann, A.J.: Self-avoiding walks in constrained and random geometries. Elsevier, Amsterdam (2005)
11. Unger, R., Moult, J.: On the Applicability of Genetic Algorithms to Protein Folding. The Twenty-Sixth Hawaii International Conference on System Sciences 1, 715–725 (1993)
12. Bastolla, U., Frauenkron, H., Gerstner, E., Grassberger, P., Nadler, W.: Testing a new Monte Carlo Algorithm for Protein Folding. National Center for Biotechnology Information 32, 52–66 (1998)
13. Liang, F., Wong, W.H.: Evolutionary Monte Carlo for protein folding simulations. J. Chem. Phys. 115 (2001)
14. Jiang, T., Cui, Q., Shi, G., Ma, S.: Protein folding simulation of the hydrophobic-hydrophilic model by computing tabu search with genetic algorithms. ISMB, Brisbane Australia (2003)
15. Shmygelska, A., Hoos, H.H.: An ant colony optimization algorithm for the 2D and 3D hydrophobic polar protein folding problem. BMC Bioinformatics 6 (2005)
16. Lee, J.: Conformational space annealing and a lattice model Protein. Journal of the Korean Physical Society 45, 1450–1454 (2004)
17. Cormen, T.H., leiserson, C.E., Rivest, R.L.: Introduction to Algorithms. MIT Press, Cambridge (1998)
18. Toma, L., Toma, S.: Folding simulation of protein models on the structure based cubo-octahedral lattice with the Contact interactions algorithm. Protein Science 8, 196–202 (1999)
19. Baker, D.: A surprising simplicity to protein folding. Nature 405, 39–42 (2000)
20. Pande, V.S., Rokhsar, D.: Folding pathway of a lattice model for proteins. PNAS 96, 273–278 (1999)
21. Levinthal, C.: Are there pathways for protein folding? Journal of Chemical Physics 64, 44–45 (1968)
22. Hoque, M.T., Chetty, M., Dooley, L.: A Guided Genetic Algorithm for Protein Folding Prediction Using 3D Hydrophobic-Hydrophilic Model. In: IEEE CEC (2006)
23. Hoque, M.T., Chetty, M., Dooley, L.S.: Significance of Hybrid Evolutionary Computation for Ab Initio Protein Folding Prediction. In: Grosan, C., Abraham, A., Ishibuchi, H. (eds.) Hybrid Evolutionary Algorithms, vol. 75, pp. 241–268. Springer, Berlin (2007)
24. Hoque, M.T., Chetty, M., Dooley, L.S.: A Hybrid Genetic Algorithm for 2D FCC Hydrophobic-Hydrophilic Lattice Model to Predict Protein Folding. In: Sattar, A., Kang, B.-h. (eds.) AI 2006. LNCS (LNAI), vol. 4304. Springer, Heidelberg (2006)
25. Hoque, M.T., Chetty, M., Dooley, L.S.: Fast computation of the fitness function for protein folding prediction in a 2D hydrophilic-hydrophobic model. Journal published in the special issue of the International Journal of Simulation Systems, Science and Technology 6, 27–37 (2005)
26. Hoque, M.T., Chetty, M., Dooley, L.S.: A New Guided Genetic Algorithm for 2D Hydrophobic-Hydrophilic Model to Predict Protein Folding. In: IEEE CEC, Edinburgh, UK (2005)

27. Hoque, M.T., Chetty, M., Sattar, A.: Protein Folding Prediction in 3D FCC HP Lattice Model Using Genetic Algorithm Bioinformatics special session. In: IEEE CEC, Singapore (2007)
28. Ghosh, K., Ozkan, S.B., Dill, K.A.: The Ultimate Speed Limit to Protein Folding Is Conformational Searching. Journal of American Chemical Society 129, 11920–11927 (2007)
29. Lau, K.F., Dill, K.A.: A lattice statistical mechanics model of the conformational and sequence spaces of proteins. Macromolecules 22, 3986–3997 (1989)
30. Dill, K.A., Bromberg, S., Yue, K., Fiebig, K.M., Yee, D.P., Thomas, P.D., Chan, H.S.: Principles of protein folding – A perspective from simple exact models. Protein Science 4, 561–602 (1995)
31. Dill, K.A., Ozkan, S.B., Weikl, T.R., Chodera, J.D., Voelz, V.A.: The protein folding problem: when will it be solved? Current Opinion in Structural Biology 17, 246–342 (2007)
32. Corne, D.W., Fogel, G.B.: An Introduction to Bioinformatics for Computer Scientists. In: Fogel, G.B., Corne, D.W. (eds.) Evolutionary Computation in Bioinformatics, pp. 3–18 (2004)
33. Baker, D.: Prediction and design of macromolecular structures and interactions. Phil. Trans. R. Soc. B361, 459–463 (2006)
34. Schueler-Furman, O., Wang, C., Bradley, P., Misura, K., Baker, D.: Progress in Modeling of Protein Structures and Interactions. Science 310, 638–642 (2005)
35. Xia, Y., Huang, E.S., Levitt, M., Samudrala, R.: Ab Initio Construction of Protein Tertiary Structures using a Hierarchical Approach. J. Mol. Biol. 300, 171–185 (2000)
36. Backofen, R., Will, S., Clote, P.: Algorithmic approach to quantifying the hydrophobic force contribution in protein folding. Pacific Symp. On Biocomputing 5, 92–103 (2000)
37. Yue, K., Dill, K.A.: Sequence-structure relationships in proteins and copolymers. Phys. Rev. E48, 2267–2278 (1993)
38. Toma, L., Toma, S.: Contact interactions methods: A new Algorithm for Protein Folding Simulations. Protein Science 5, 147–153 (1996)
39. Bornberg-Bauer, E.: Chain Growth Algorithms for HP-Type Lattice Proteins. In: RECOMB, USA (1997)
40. Hoque, M.T., Chetty, M., Dooley, L.S.: Non-Isomorphic Coding in Lattice Model and its Impact for Protein Folding Prediction Using Genetic Algorithm. In: IEEE CIBCB, Toronto, Canada (2006)
41. Chen, M., Lin, K.Y.: Universal amplitude ratios for three-dimensional self-avoiding walks. Journal of Physics A: Mathematical and General 35, 1501–1508 (2002)
42. Roterman, I.K., Lambert, M.H., Gibson, K.D., Scheraga, H.: A comparison of the CHARMM, AMBER and ECEPP potentials for peptides. II. Phi-psi maps for N-acetyl alanine N'-methyl amide: comparisons, contrasts and simple experimental tests. J. Biomol. Struct. Dynamics 7, 421–453 (1989)
43. Cornell, W.D., Cieplak, P., Bayly, C.I., Gould Jr., I.R., Merz Jr., K.M., Ferguson, D.M., Spellmeyer, D.C., Fox, T., Caldwell, J.W., Kollman, P.A.: A second generation force field for the simulation of proteins and nucleic acids. J. Am. Chem. Soc. 117, 5179–5197 (1995)
44. Cutello, V., Nicosia, G., Pavone, M., Timmis, J.: An Immune Algorithm for Protein Structure Prediction on Lattice Models. IEEE Transactions on Evolutionary Computation 11 (2007)
45. Takahashi, O., Kita, H., Kobayashi, S.: Protein Folding by A Hierarchical Genetic Algorithm. AROB (1999)
46. Unger, R., Moult, J.: Genetic Algorithms for Protein Folding Simulations. J. of Mol. Bio. 231, 75–81 (1993)

47. Unger, R., Moult, J.: Genetic Algorithm for 3D Protein Folding Simulations. In: Conference on GAs, pp. 581–588 (1993)
48. König, R., Dandekar, T.: Refined Genetic Algorithm Simulation to Model Proteins. Journal of Molecular Modeling 5 (1999)
49. Lee, J., Scheraga, H.A., Rackovsky, S.: New Optimization Method for Conformational energy Calculations on Polypeptides: Conformational Space Annealing. J. of Comp. Chemistry 18, 1222–1232 (1997)
50. Haupt, R.L., Haupt, S.E.: Practical Genetic Algorithms (2004)
51. Digalakis, J.G., Margaritis, K.G.: An experimental Study of Benchmarking Functions for Genetic Algorithms Intern. J. Computer Math. 79, 403–416 (2002)
52. Flores, S.D., Smith, J.: Study of Fitness Landscapes for the HP model of Protein Structure Prediction. In: IEEE CEC (2003)
53. Mousseau, N., Barkema, G.T.: Exploring High-Dimensional Energy Landscape. Computing in Science & Engineering 1, 74–80, 82 (1999)
54. Hansmann, U.H.E.: Protein Folding in Silico: An Overview. In: IEEE CS and the AIP (2003)
55. Skolnick, J., Kolinski, A.: Computational Studies of Protein Folding. IEEE COMPUTING IN SCIENCE & ENGINEERING 3, 40–50 (2001)
56. Cui, Y., Wong, W.H., Bornberg-Bauer, E., Chan, H.S.: Recombinatoric exploration of novel folded structures: A heteropolymer-based model of protein evolutionary landscapes. PNAS 99, 809–814 (2002)
57. Schreiner, K.: Distributed Project Tackle Protein Mystery. Computing in Science & Engineering. IEEE 3, 13–16 (2001)
58. Lesh, N., Mitzenmacher, M., Whitesides, S.: A Complete and Effective Move Set for Simplified Protein Folding. In: RECOMB, Berlin, Germany (2003)
59. Hart, W.E., Istrail, S.: HP Benchmarks vol. 2005 (2005)
60. Rosetta, Y.: 2.1.0., Copyright © 2007-2008 The Rosetta Commons (last access, 2008), http://www.rosettacommons.org/tiki/tiki-index.php?page=Change+Log
61. Bonneau, R., Tsai, J., Ruczinski, I., Chivian, D., Rohl, C., Strauss, C.E.M., Baker, D.: Rosetta in CASP4: Progress in Ab Initio Protein Structure Prediction. PROTEINS: Struct. Func. and Genetics 5, 116–119 (2001)
62. Bradley, P., et al.: Rosetta Predictions in CASP5: Success, Failure, and Prospects for Complete Automation. PROTEINS: Structure, Function, and Genetics 53, 457–468 (2003)
63. Simons, K.T., Bonneau, R., Ruczinski, I., Baker, D.: Ab Initio Protein Structure Prediction of CASP III Target Using ROSETTA. PROTEINS: Structure, Function, and Genetics 3, 171–176 (1999)
64. Hoque, M.T., Chetty, M., Dooley, L.S.: Generalized Schemata Theorem Incorporating Twin Removal for Protein Structure Prediction. In: PRIB, Singapore (2007)
65. Koumousis, V.K., Katsaras, C.P.: A Saw-Tooth Genetic Algorithm Combining the Effects of Variable Population Size and Reinitialization to Enhance Performance. TEVC 10, 19–28 (2006)