Neural Comput & Applic (2017) 28:3385–3391
DOI 10.1007/s00521-016-2252-5

ORIGINAL ARTICLE

Biological complexity: ant colony meta-heuristic optimization algorithm for protein folding

Aman Chandra Kaushik[1] · **Shakti Sahi**[1]

Received: 19 June 2015 / Accepted: 19 February 2016 / Published online: 7 March 2016

Abstract Ant colony meta-heuristic optimization (ACO) is one of the few algorithms that can help to gain an atomic level insight into the conformation of protein folding states, intermediate weights and pheromones present along the protein folding pathway. These are analysed by nodes (amino acids), and these nodes depend upon the probability of next optimized node (amino acids). Nodes have conformational degrees of freedom as well as depend upon the natural factors and collective behaviour of biologically important molecules like temperature, volume, pressure and other ensembles. This biological quantum complexity can be resolved using ACO algorithm. Ants are visually blind and important behaviour of communication among individuals or colony of ant environment is based on chemicals (pheromones) deposited by the ants. Just like ants, proteins are also a group of colony; amino acids are node (amino acid) attached to each others with the help of bonds. This paper is aimed to determine the factors affecting protein folding pattern using ant colony algorithm. Protein occurs structurally in a compact form and determining the ways of protein folding is called NP hard (non-deterministic polynomial-time hard) problem. Using the ACO, we have developed an algorithm for protein folding. It is interesting to note that based on ants ability to find new shorter path between the nest and the food, proteins can also be optimized for shorter path between one node to another node and the folding pattern can be predicted for an unknown protein (ab initio). We have developed an application based on ACO in Perl language (PFEBRT) for determining optimized folding path of proteins.

Keywords ACO · Node · Heuristics · Pheromones · NP hard problems · GAFF

1 Introduction

Various kinds of optimization algorithms have been implemented to tackle folding in homology modelling, threading and ab initio based on artificial intelligence and hybrid approaches. Protein folding is non-deterministic polynomial-time hard problem for identifying protein conformation and folding process. Ant colony optimization (ACO) is used in various problems like routing vehicles, dynamic problems, stochastic problems, multi-target implementation, multi-target parallel implementations, software testing, travelling salesman problem and protein folding. Ant colony meta-heuristic optimization is one of the few algorithms that can help to gain an atomic level insight into the conformation of protein folding states and its intermediate weights and pheromone present along the protein folding. This ant colony optimization algorithm is inspired by research on the behaviour of real ant colonies. Ant colonies are distributed systems which perform complex tasks for finding food in an optimized way. Ant algorithms are derived from the observation of real ant's behaviour and optimized for distributed control problems [1]. The different aspects of the behaviour of ant colonies are useful in solving computational protein folding problems. The main logic behind ant colony algorithms is to

Electronic supplementary material The online version of this article (doi:10.1007/s00521-016-2252-5) contains supplementary material, which is available to authorized users.

✉ Shakti Sahi
shaktis@gbu.ac.in

[1] School of Biotechnology, Gautam Buddha University, Greater Noida, Uttar Pradesh, India

solve the NP hard problem of protein folding mechanism [2]. The ants exhibit a complex behavioural pattern which helps them to find optimized shorter path between the nest and the food. Ants are visually impaired and an important aspect of their communication is based on chemicals released and deposited (pheromones) by them [3]. They tend to choose paths marked by strong pheromone concentration (pheromone trail) for food. Proteins may also be considered as a colony where amino acids are nodes attached to each other with the help of peptide bonds. Protein structure folds in a compact form. Using ACO behaviour, researchers can identify the folding pattern of proteins [4]. Similar to the behavioural pattern exhibited by ants, proteins can also be optimized to have a shorter connecting path between one amino acid to another amino acid [5]. An interesting experiment was conducted on ants involving a double bridge connecting the ant nest to food source to study the pheromone trail laying and following behaviour. Initially, only the long branch was opened to ants and later the short branch was also offered. Since the starting pheromone concentration was high on the long branch with slow evaporation of pheromone, so majority of ants always chose the long branch even after the appearance of shorter branch [6, 7]. Similarly, in protein structure starting amino acid bonding is very strong on the long branch with standard distance factors using general AMBER force field (GAFF), and other factors like temperature, pressure, volume, degree of freedom and total time of protein folding. With these ensembles, proteins are also dependent on standard distance factors; when distance factors are changed in proteins, it causes random or specific mutation in proteins resulting in diseases.

1.1 Monte Carlo simulation

The Monte Carlo simulation (MC) algorithms is based on the energy distribution for a given protein temperature and executes temperature simulation for protein folding pattern [8–10]. MC algorithms are based on conformational states and their searching efficiency has been enhanced in the Basin Hopping approach, which couples large step Monte Carlo jumps with gradient-driven local minimization [11, 12]. After searching the final energy distribution, it modifies the transition probability to accelerate the transition between different states [13]. Monte Carlo minimization has been successfully applied to the conformational searching in protein folding pattern [2] by executing the local energy minimization of each trajectory of protein folding pattern [14]. Monte Carlo approach allows efficient exploration of protein conformation and is comparable with genetic algorithm and other heuristic approaches.

1.2 Molecular dynamics

Molecular dynamics (MD) simulation is based on Newton's equations of motion. It monitors atom movements during protein folding pattern [15] and is one of the most useful methods for known biological complexity of protein folding problem [16, 17]. The long MD simulation is a major limiting factor as the incremental timescale is in the order of femtoseconds, while the fastest protein folding pattern timing of a small protein less than 100 residues is in the millisecond range [18–22]. There are many softwares for MD simulation which are also used for the structure refinements of low-resolution model [23, 24]. Number of parameters like implemented torsions, ensembles and coarse-grained energy functions are used for refinement [25, 26]. Molecular dynamics simulation is based on Newton's equations of motion to all atoms concurrently over a small time step to conclude new atomic positions and velocities. In cases of Monte Carlo (MC) and molecular dynamics (MD), the force field controls the total energy, which concludes the evolution of the systems. Molecular dynamics simulation is proven to be a powerful approach for studying protein dynamics.

1.3 Genetic algorithm

Conformational space annealing is based on one of the genetic algorithms (GAs) for protein folding pattern. Using GA folding pattern of the proteins can be identified [27, 28]. The MC algorithms are based on local minima, searches whole conformation of the protein and generate low-energy conformation, while conformational space annealing applies various global optimizations, searches whole conformation of the protein and generates low energy for folding pattern of the protein [29, 30]. Conformational space annealing has been successfully applied to ab initio modelling of the protein.

2 Materials and methods

The flow chart (Fig. 1) represents the movement of ant (protein) based on two parameters A (heuristics as a force fields of amino acid to other amino acids) and B (pheromones as a distance of amino acid to other amino acids).

2.1 Ant colony algorithms

1. *Implementation* The methodology for ant colony optimization algorithm implementation for the protein folding pattern, which is NP hard problem [31], and generation of various conformational states using ACO [32–37] is shown in Fig. 1. A and B are two

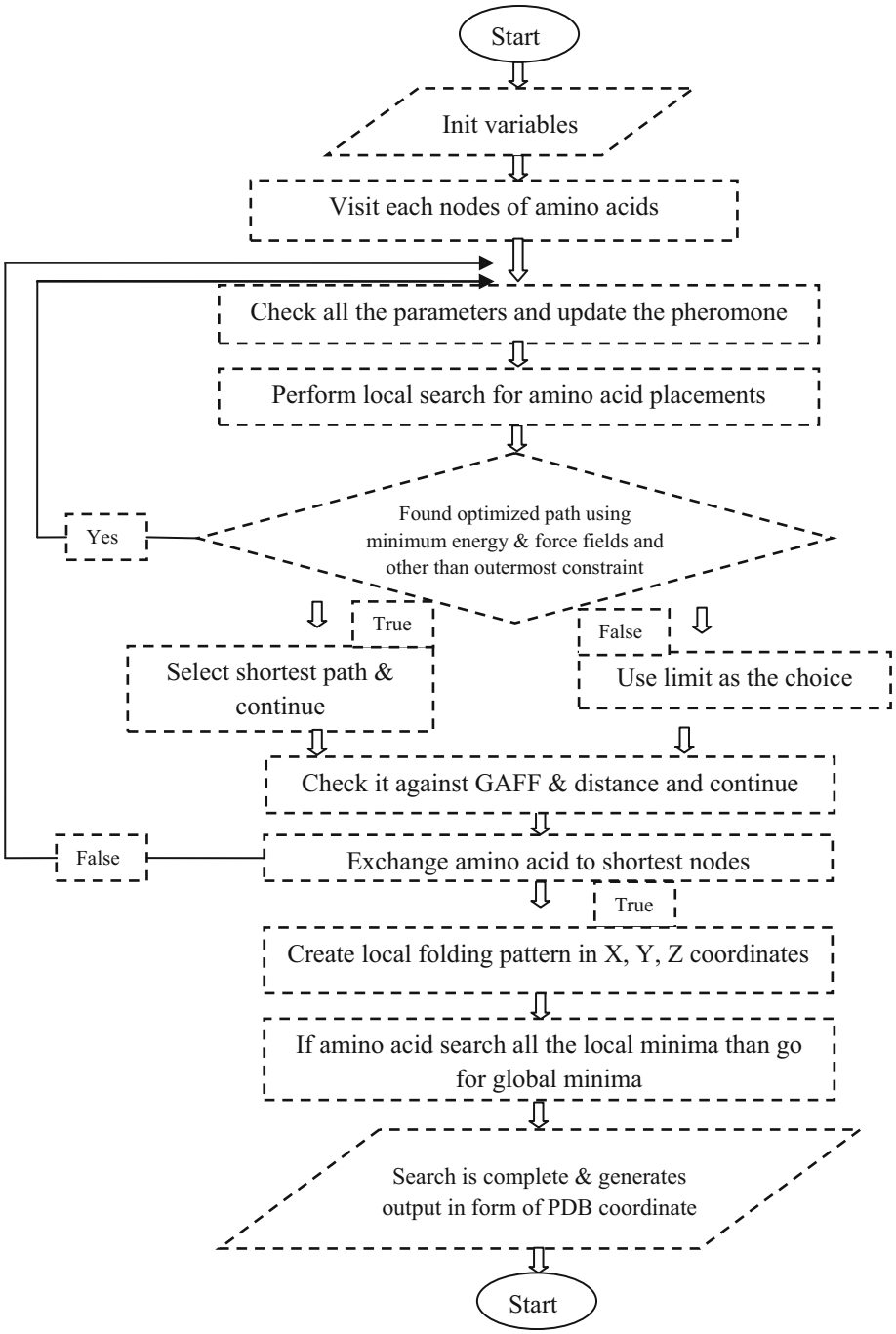

Fig. 1 Flow chart representation of ACO algorithm, movement of ant (protein) based on two parameters

parameters where A (heuristics as a force fields of amino acid to other amino acids) and B (pheromones as a distance of amino acid to other amino acids) determine the relative influence of the force fields (taken from the GAFF) or distance factors (taken from the GAFF) [38]. These factors are responsible for the trail of amino acids and the heuristic information on the path traversed by amino acids. Initialization (*I*) is initial distance factor and heuristic value deposited on each amino acid in proteins, respectively. Initially, (*I*) is set equal to the depth of proteins and *I* is set equal to the number of decision node (amino acid). This variable specifies the count status of node (amino acid) by the amino acids *k*, initially set to 0. N_i^k is the feasible neighbourhood of amino acids *k* when being at node *i*, initially set to zero. Key = End node in proteins, Pframe = [Total no of node (amino acids in the protein)]. Pframe = [1, 2, 3, 4, 5…EndAA], NC = total node sequence covered up to now, calculation of depth of proteins using algorithm ACO_ DEPTH.

Initialization Init accordingly and determination of number of decision node in proteins and setting up the initial value. When expanded a fractional conformation $I_k \ldots I_i$ to I_{i+1} during the edifice phase of ant colony optimization algorithms, next to kin direction d of I_{i+1} to I_{i-1} is resolved based on heuristic (force fields K_{ij}) and pheromone (distance of amino acids r_{ij}) values according to following probabilities $P_i d = \left(T_{xy}^{\alpha}\right)\left(\eta_{xy}^{\beta}\right) / \sum_y \in \text{allowed}_y \left(T_{xy}^{\alpha}\right)\left(\eta_{xy}^{\beta}\right)$.

2. *Application development* We developed Perl application, for validation of proposed algorithm and also to find out protein folding pattern of the protein using ACO algorithm. In this application by selecting PFP button of the application, the desired protein in PDB format can be selected. This would generate coordinates of given PDB file and save it into analysis file folder. This output can be used to investigate the optimized folding pattern of the given PDB file.

3 Results and discussion

We report ACO algorithm for protein (including membrane protein) folding prediction. We have designed new algorithm for protein conformation and protein folding. The algorithm traverses each node and prioritizes the path according to the path strength. Paths having the standard distance factors strength are given the highest priority for testing followed by next lower standard distance factors strength. This algorithm finds optimized path for protein folding (native conformations) within nanoseconds of CPU. The protein with the PDB code: 4BEY (night blindness causing G90D rhodopsin) was used as an example (Fig. 1) to prioritize the various conformation states using ant colony optimization algorithms. The algorithm was applied as follows.

GAFF (general AMBER force field) is well matched to the AMBER force fields, GAFF is appropriate to study range of molecules, and generally, all the organic molecules are made of C, H, S, O, N, P, F, Br, Cl and I. The interaction force fields (K_{ij}) parameter between one molecule to another molecule and interaction distance (r_{ij}) parameter between one molecule to another molecule are described in supplementary Table 1. AMBER and GAFF force fields have been reported to work well in case of drug designing, biological molecules and organic molecules. GAFF is more compatible for rational drug designing and applies harmonic function form as following

$$E_{\text{Pair}} = \sum_{\text{Bonds}} K_r\left(r - r_{\text{eq}}\right)^2 + \sum_{\text{Angles}} K_\theta\left(\theta - \theta_{\text{eq}}\right)^2 + \sum_{\text{Dihedrals}} V_n/2[1 + \text{COS}(n\varphi - \Upsilon)] + \sum_{i<j}\left[A_{ij}/R_{ij}^{12} - B_{ij}/R_{ij}^{6} + q_i q_j / R_{ij}\right].$$

3.1 ACO algorithms implementation for identification of protein folding pattern using standard distance and generalized AMBER force fields (GAFF) parameters

1. Firstly, the amino acids starting node A and its neighbourhood $N_s^k = 1$ was defined. Heuristic was known from the generalized AMBER force fields. One per cent distance factors is evaporated from the node covered up to now, i.e. node A − 1 initially has approximately 1 distance factor value which after evaporation is left to 0.082. Optimized path covered after finding each next node to be visited upon is calculated and thus its length, i.e. A − 1 the optimized path is A − 1 where length is equal to 1. The next node to be moved upon is node 1.
2. As the next node G to be moved upon is not equal to key node (i.e. end node), the amino acids starting node are now node 1 and the above steps are again followed.
3. Applied random proportional rule to decide the next node of ant, probabilities of visiting the ant from node to node, the process is repeated till the ant reaches the destination node END_AA.
4. As node covered up to now = i which is not equal to Pframe = {M, C, G, T, E), the amino acid again starts from the start node i.
5. As node covered up to now = (M, C, G, T, E). The ant had now covered one full path starting from node 1 to node END_AA. The amount of distance factors deposited on each edge in the traversed path up to now is calculated, and the net pheromone is updated on each edge in the path equal to amount left after evaporation of distance factors + amount deposited after path traversed.
6. The change in heuristic (GAFF) is calculated, i.e. $\Delta n_i^j = n_{ij}/C_i^k$.
7. As node covered until now = (M, C, G, T, E) which is not equal to Pframe = (M, C, G, T, E), the ant again starts from the start node i.
8. The process is continued till all nodes are covered.
9. When node covered = Pframe, the strength of each path was calculated, for example, for the path (M, C, G, T, E), and the strength is calculated as $I - 1$ = (final pheromone value) × (final heuristic value). Similarly, for the others edges in the path we calculate

Table 1 Probability-based result summary of different moves of an ant path, here we use PDB:4BEY [39] PDB file (night blindness causing G90D rhodopsin) for calculation of protein folding pattern using ant colony optimization algorithms

Move no.	NC Node covered	t_0 Initial pheromone	$t_e \leftarrow (1-\rho)^* t_0$ Evaporation	C_i^k Length of tour	$\Delta t_{ij} = 1/C_i^k$	n_o	$\Delta n_{ij} \leftarrow n_o/C_i^k$ Left heuristic
First	M–C	0.738	0.730	0.25	0.980	4.661	1.165
	C–G	0.738	0.730	0.25	0.980	4.661	1.165
	G–T	0.738	0.730	0.25	0.980	4.661	1.165
	T–E	0.738	0.730	0.25	0.980	4.661	1.165
Second	M–G	0.738	0.730	0.5	1.230	4.661	2.330
	G–E	0.738	0.730	0.5	1.230	4.661	2.330
Third	M–T	0.738	0.730	0.5	1.230	4.661	2.330
	T–E	0.980	0.970	0.5	1.470	1.165	0.615
Fourth	M–E	0.738	0.730	1	1.730	4.661	4.661
Fifth	M–T	1.230	1.210	0.5	1.710	2.330	1.165
	T–E	1.455	1.455	0.5	1.955	0.615	0.307
Sixth	M–G	1.230	1.210	0.5	1.710	2.330	1.165
	G–E	1.230	1.210	0.5	1.710	2.330	1.165
Seventh	M–T	1.710	1.539	0.5	2.039	1.165	0.582
	T–E	1.955	1.759	0.5	2.259	0.307	0.153
Eighth	M–G	1.710	1.539	0.5	2.039	1.165	0.582
	G–E	1.710	1.539	0.5	2.039	1.165	0.582
Ninth	M–C	0.980	0.970	0.33	1.300	1.165	0.388
	C–T	0.738	0.730	0.33	1.060	4.661	1.553
	T–E	2.259	2.236	0.33	2.566	0.153	0.051
Tenth	M–T	2.039	2.018	0.5	2.518	0.582	0.291
	T–E	2.566	2.540	0.5	3.040	0.512	0.256
Eleventh	M–G	2.039	2.018	0.5	2.518	0.582	0.291
	G–E	2.039	2.018	0.5	2.518	0.582	0.291
Twelfth	M–T	2.518	2.493	0.5	2.993	0.291	0.145
	T–E	3.040	3.009	0.5	3.509	0.256	0.128
Thirteenth	M–G	2.518	2.493	0.33	2.823	0.291	0.097
	G–T	0.980	0.970	0.33	1.300	1.165	0.388
	T–E	3.509	3.474	0.33	3.804	0.128	0.042
Fourteenth	M–C	1.300	1.287	0.5	1.787	0.388	0.194
	C–E	0.738	0.730	0.5	1.238	4.661	2.330

edge strengths. Finally, adding all edge's strength in the path gives the final value for the path strength (Table 1)

Further, the algorithm using was compared and validated using PDB:4BEY protein. There was a good correlation in the results of observed and the experimental results. Figure 2a represents the folding pattern priority on the basis of nearest distance and favourable force fields using ant colony optimization (ACO) where different colours represent amino acids of small peptide (PDB:4BEY) and their M to E movement. The Fig. 2b represents the folding pattern of PDB:4BEY on the basis of nearest distance and favourable force fields using ant colony optimization (ACO) as in 2a in graphical form, where blue colour represents MET (node 1), cyan colour represents CYS (node 2), green colour represents GLY (node 3), orange colour represents THR (node 4), and red colour represents GLU (node 5) amino acids. The best optimized and prioritized folding paths are given in Table 2 covering all the residues of PDB:4BEY.

The final path according to the total path strength and priority is shown in Table 2. The path having the maximum combined strength of pheromone and heuristic has been given the highest priority.

A Perl-based application protein folding energy-based recognition tool (PFEBRT) was developed as shown in Fig. 3a, b to determine the protein folding pattern of the protein using ant colony optimization algorithms. With this, users can easily select desired protein in PDB format

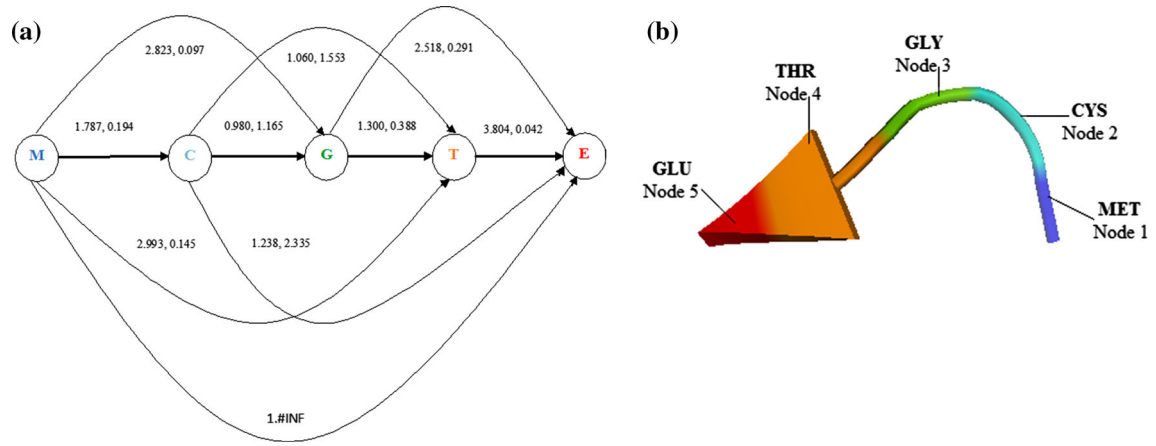

Fig. 2 Panel **a** represents the PDB:4BEY protein folding pattern using ant colony optimization. Panel **b** represents the small peptide folding pattern of PDB:4BEY

Table 2 Independent path's strength versus priority

S. no.	All independent paths	Strength	Priority
1	M, C, G, T, E	2.150	4
2	M, E	1.#INF	1
3	M, C, T, E	2.148	5
4	M, C, E	3.230	2
5	M, C, G, E	2.219	3
6	M, G, E	1.005	6
7	M, G, T, E	0.936	7
8	M, T, E	0.592	8

and generate coordinates of given PDB file and save it into analysed file folder. In this work, a crucial role is played by CPU time. The maximum number of local minima search of sequences using ant colony algorithm is improved with respect to time and minimum number of CPU usage.

4 Conclusion

We have developed an ant colony-based algorithm and implemented in Perl language for protein folding. Using this application, the conformation of protein folding states, intermediates weights and pheromone present along the protein folding can be determined using amino acids interaction, ensembles and force fields. Protein folding is non-deterministic polynomial-time hard (NP hard) problem, using PDB files identification of protein conformation and folding process can be done. The ant colony meta-heuristic optimization is one of the few

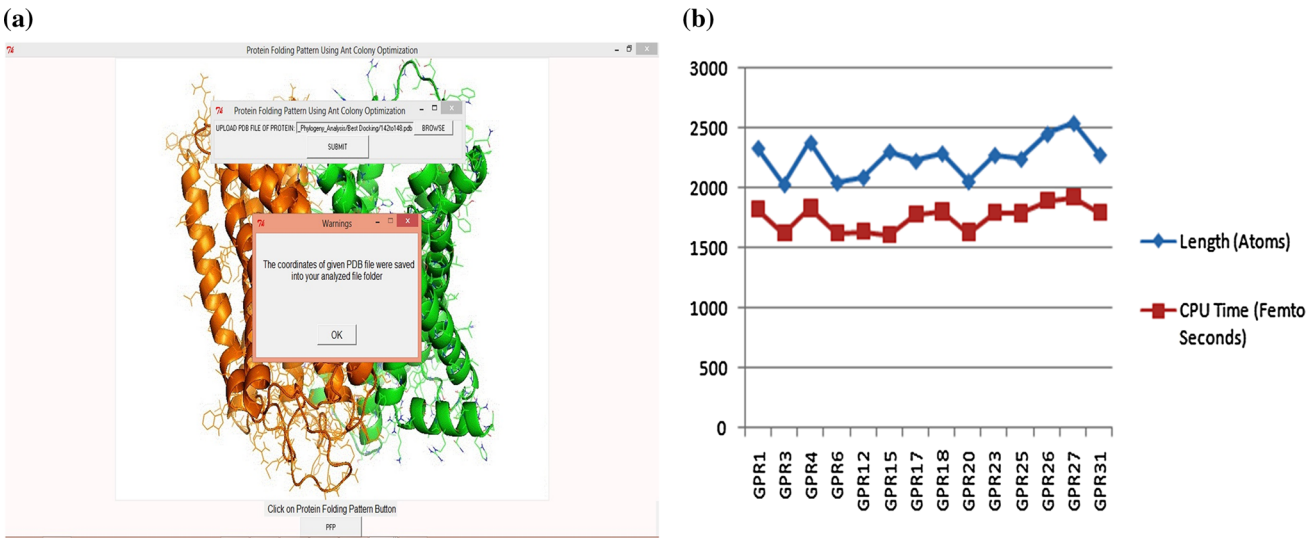

Fig. 3 **a** Represents the front panel view of protein folding energy-based recognition tool, and **b** represents the performance of protein folding energy-based recognition tool, where *x* axis represents different GPCRs at different length and *y* axis represents the CPU timing in femtoseconds

algorithms that can help to gain an atomic level insight into the conformation of protein folding states and intermediates weights and pheromone present along the protein folding. It finds a more optimized path of folding states. Development of ACO algorithms is more realistic and CPU-based models for protein structure folding pattern, i.e. NP hard problem.

References

1. Dorigo M (2005) Ant colony optimization theory: a survey. Elsevier 344:243–278
2. Kuwajima K (1989) The molten globule state as a clue for understanding the folding and cooperativity of globular-protein structure. Proteins Struct Funct Genet 6:87–103
3. Dorigo M (1996) The ant system: optimization by a colony of cooperating agents. IEEE Trans 26:29–41
4. Dorigo M (1997) Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans Evolu Comput 1:53–66
5. Dorigo M (1997) Ant colonies for the traveling salesman problem. Bio-Systems 43:73–81
6. Dorigo M, Maniezzo V, Colorni A (1991) Positive feedback as a search strategy. Tech rep., pp 91–116
7. Dorigo M, Di Caro G (1999) New ideas in optimization. In: Corne D, Dorigo M, Glover F (eds) New ideas in optimization. McGraw-Hill, New York, pp 63–76
8. Bastolla U, Fravenkron H, Gestner E, Grassberger P, Nadler W (1998) Testing a New Monte Carlo algorithm for the protein folding problem. Proteins 32:52–66
9. Georgopoulos C, Liberek K, Zylicz M, Ang D (1994) Heat-shock proteins in biology and medicine. Cold Spring Harbor Laboratory Press, Cold Spring Harbor, New York, pp 209–249
10. O'Toole EM, Panagiotopoulos AZ (1992) Monte Carlo simulation of folding transitions of simple model proteins using a chain growth algorithm. J Chem Phys 97:8644–8652
11. Ramakrishnan R, Ramachandran B, Pekny JF (1997) A dynamic Monte Carlo algorithm for exploration of dense conformational spaces in heteropolymers. J Chem Phys 106:2418–2424
12. Irback A (1998) Monte Carlo approach to biopolymers and protein folding. World Scientific, Singapore, pp 98–109
13. Sali A, Shakhnovich E, Karplus M (1994) How does a protein fold? Nature 369:248–251
14. Kim PS, Baldwin RL (1990) Intermediates in the folding reactions of small proteins. Annu Rev Biochem 59:631–660
15. Backofen R (2001) The protein structure prediction problem: a constraint optimization approach using a new lower bound. Springer 6:223–255
16. Richards FM (1977) Areas, volumes, packing, and protein structures. Annu Rev Biophys Bioeng 6:151–176
17. Chikenji G, Kikuchi M, Iba Y (1999) Multi-self-overlap ensemble for protein folding: ground state search and thermodynamics. ARXIV 27:1–4
18. Dill KA, Fiebig KM, Chan HS (1993) Cooperativity in protein-folding kinetics. Proc Natl Acad Sci USA 90:1942–1946
19. Beutler T, Dill K (1996) A fast conformational search strategy for finding low energy structures of model proteins. Protein 5:2037–2043
20. Yue K, Dill KA (1995) Forces of tertiary structural organization in globular proteins. Proc Natl Acad Sci USA 92:146–150
21. Backofen R, Will S (2003) A constraint-based approach to structure prediction for simplified protein models that outperforms other existing methods. In: Proceedings of XIX international conference on logic programming, pp 49–71
22. Torrie GM, Valleau JP (1977) Nonphysical sampling distributions in MC free energy estimation: umbrella sampling. J Comput Phys 23:187–199
23. Berg BA, Neuhaus T (1992) Multicanonical ensemble: a new approach to simulate first-order phase transitions. Phys Rev Lett 68:9–12
24. Plaxco KW, Simons KT, Baker D (1998) Contact order, transition state placement and the refolding rates of single domain proteins. J Mol Biol 277:985–994
25. Hoos HH, Stützle T (2004) Stochastic local search: foundations and applications. Elsevier, Amsterdam, pp 1–156
26. Krasnogor N, Pelta D, Lopez PM, Mocciola P, de la Canal E (1998) Genetic algorithms for the protein folding problem: a critical view. In: Proceedings of engineering of intelligent systems. ICSC Academic Press, pp 353–360
27. Patton AWP, Goldman E (1995) A standard GA approach to native protein conformation prediction. In: Proceedings of the 6th international conference in genetic algorithms Morgan Kaufmann Publishers, pp 574–581
28. Unger R, Moult J (1993) Genetic algorithms for protein folding simulations. J Mol Biol 231:75–81
29. Unger R, Moult J (1993) A genetic algorithm for three dimensional protein folding simulations. In: Proceedings of the 5th international conference on genetic algorithms Morgan Kaufmann Publishers, pp 581–588
30. Hsu HP, Mehra V, Nadler W, Grassberger P (2003) Growth algorithm for lattice heteropolymers at low temperatures. J Chem Phys 51:118–444
31. Bin W, Zhongzhi S (2011) An ant colony algorithm based partition algorithm for TSP. Chin J Comput 24:1328–1333
32. Gambardella LM, Dorigo M (1999) Ant colonies for the quadratic assignment problem. J Oper Res Soc 50:167–176
33. Shmygelska A, Hernandez R, Hoos H H (2002): An ant colony optimization algorithm for the 2d hp protein folding problem. In: Proceedings of the 3rd international workshop on ant algorithms, pp 40–52
34. Shmygelska A, Hoos HH (2005) An ant colony optimization algorithm for the 2d and 3d hydrophobic polar protein folding problem. BMC Bioinform 30:97–112
35. He LL, Shi F, Zhou HB (2011) Application of improved ant colony optimization algorithm to the 2D HP model. Wuhan Univ J (Nat Sci Edn) 51:33–38
36. Xudong Wu (2012) A two-stage ant colony optimization algorithm for the vehicle routing problem with time windows. IJACT 4:485–491
37. Liu Fang (2012) A dual population parallel ant colony optimization algorithm for solving the travelling salesman problem. JCIT 7:66–74
38. Wang J, Wolf RM, Caldwell JW, Kollman PA, Case DA (2004) Development and testing of a general amber force field. J Comput Chem 25:1157–1174
39. Singhal A, Ostermaier MK, Vishnivetskiy SA, Panneels V, Homan KT, Tesmer JJ, Veprintsev D, Deupi X, Gurevich VV, Schertler GF, Standfuss J (2013) Insights into congenital stationary night blindness based on the structure of G90D rhodopsin. EMBO Rep 14:520–526