Novel Memetic Algorithm for Protein Structure Prediction

Md. Kamrul Islam and Madhu Chetty

GSIT, Moansh University,
3842, VIC, Australia
{kamrul.islam,madhu.chetty}@infotech.monash.edu.au

Abstract. A novel Memetic Algorithm (MA) is proposed for investigating the complex *ab initio* protein structure prediction problem. The proposed MA has a new fitness function incorporating domain knowledge in the form of two new measures (H-compliance and P-compliance) to indicate hydrophobic and hydrophilic nature of a residue. It also includes two novel techniques for dynamically preserving best fit schema and for providing a guided search. The algorithm performance is investigated with the aid of commonly studied 2D lattice hydrophobic polar (HP) model for the benchmark as well as non-benchmark sequences. Comparative studies with other search algorithms reveal superior performance of the proposed technique.

Keywords: Memetic Algorithm, Pair-wise-interchange, Tabu Search, Modified fitness function, Schema preservation, Guided search space.

1 Introduction

The protein folding problem has remained one of the grand challenges in computational molecular biology. For PSP investigations, hydrophobic-polar (HP) protein model [1] is most commonly applied. The model considers hydrophobic (lacking affinity for water) and hydrophilic (water loving) interactions as the two main dominant forces in protein folding process and amino acids are therefore represented as either hydrophobic (H) or hydrophilic (P). For 2D modeling, these residues are located in square lattice ensuring a self-avoiding walk (SAW) so that the two residues do not occupy same space position. The fitness function measures the energy of the conformation which is obtained by evaluating the topological contacts between two hydrophobic residues (H-H) as -1 (provided they are not neighbors in given sequence) while topological contacts for other possible pairs (H-P, P-H, and P-P) are evaluated as 0. The energy matrix E_{TN} of the HP model [2] is given by *eqn. 1*. Protein conformation can be encoded in various ways such as absolute, relative and so on. In relative encoding, a conformation has three possible moves relative to current position, namely *forward (F)*, *left (L)* and *right (R)*. The first move is always considered as forward (F).

Protein structure prediction (PSP), even in simplified hydrophobic-polar (HP) model, is NP-complete [3]. Hence, not only GA [4,5,6] but a plethora of other

A. Nicholson and X. Li (Eds.): AI 2009, LNAI 5866, pp. 412–421, 2009.

evolutionary algorithms [7] including Ant Colony Optimization (ACO), Tabu Search (TS), Monte Carlo (MC), Memetic Algorithm (MA) are being investigated. Since the PSP problem has a large and complex search space, algorithms which emphasis only on global optimization (e.g. GA) might not be able to perform properly. MA, a powerful combination of GA and local search (LS), due to its ability to combine local search (LS) techniques refines individual population and improves their fitness [8]. Usually, the flexible architecture of MA allows it to include different approaches for local search, i.e. gradient descent, pair wise interchange (PWI), tabu search (TS). In this paper, MA with pair wise interchange is referred as pair-wise MA (PMA) and MA with Tabu search is referred as tabu MA (TMA). Recent studies in various domains [9,10,11] show that MA is both efficient (less computations) and effective (higher accuracy) compared to other EAs. Comparisons between several EAs and MA with pair-wise-interchange (PWI) show better performance for MA [12]. However, limited work has been reported on its application to NP-complete PSP problem. Recently, hybridization of GA with Tabu search on PSP [6] showed a satisfactory performance. With changes in population size based on *complexity* of the protein sequence, its application is, however, limited because it is not always possible to know the complexity upfront. Krasnogor *et al.* [8,13,14,15] and Smith [16,17] applied MA to solve PSP problem using techniques such as fuzzy logic, multi-meme, co-evolution with limited improvement.

$$E_{TN} = \sum_{j=1}^{n-1} \sum_{k=j+1}^{n} N_{jk} \quad \text{where,}$$
$$N_{jk} = \begin{cases} -1 \text{ if } j \text{ and } k \text{ are both H residues and topological neighbour;} \\ 0 \quad \text{otherwise.} \end{cases} \tag{1}$$

An appropriate fitness function, capturing domain knowledge is very important for enhancing fitness function and improving the accuracy of prediction. For example, Radius of Gyration (RG), measuring radial distance from a given axis, was applied [4] to capture the domain characteristics. In an effort to use the characteristics of the amino acids, hydrophobic property was included in the fitness function [5]. However, there has been no effort to use the equally significant second hydrophilic (P) property of the residues. We propose a novel fitness function which not only maintains the significance of the existing fundamental fitness parameters but also incorporates domain knowledge to bring H type amino acids close to the H-core and pushing P type residues close to the boundary. This is achieved by developing two new measures for H and P characteristics, namely H-compliance and P-compliance. The proposed algorithm also includes a new technique for dynamically preserving the fit schema based on domain knowledge. Further, we also propose a novel approach to add interim individuals in a guided manner (rather than randomly) and also maintain the necessary diversity in population. Experiments are performed using the 2D HP lattice model and using the bench mark as well as non benchmark sequences. Comparisons with other techniques are also carried out which show a superior performance of the proposed algorithm.

2 Proposed Memetic Algorithm

In this section, we present the three novel aspects of the proposed MA which enhances its potential for solving the complex PSP problem.

2.1 Modified Fitness Function

An ideal empirical energy function contains only a few energy terms, is computationally efficient, which can be easily derived from experimental data [18]. Further, for PSP problem it should account for effects such as hydrophobic packing and include penalties for undesirable effects. We will address hydrophobic packing as it can prove to be important in removing the limitations of the existing fitness function *eqn. 1.* This is done by including two new fitness terms for capturing the H and P characteristics of the residues (i) H-compliance factor and (ii) P-compliance factor. The resulting new fitness function obtained by including the two new fitness terms will be referred as 'modified fitness function'.

H-compliance. As we mentioned earlier, the H residues lack affinity for water and tend to be located within the protein fold. We define the H-compliance of a H-type residue as a measure of how compactly (i.e. closely) a residue is located to the H-core centre. It is measured as the radial distance of H residues from H-core centre. The smaller the value of H-compliance, the closer the residue is to the H-core centre. The sum of the distance of all the H-type residues in the sequence gives the H-compliance of the conformation under consideration.

H-compliance of i^{th} H type residue is denoted as h_i. To calculate h_i, we determine the center of a hypothetical rectangle "enclosing" the residues forming the H core as shown in Fig. 1(a). The coordinates (x_{rect}, y_{rect}) of the "center" are obtained as: $x_{rect} = (x_h max - x_h min)/2$, $y_{rect} = (y_h max - y_h min)/2$. Further, if coordinates of any i^{th} hydrophobic residue are given as (x_{hi}, y_{hi}), the overall H-compliance of the j^{th} conformation can be obtained as $H_j = \sum_{i=1}^{n_h} h_i$. That is

$$H_j = \sum_{i=1}^{n_h} h_i = \sum_{i=1}^{n_h} (x_{rect} - x_{hi})^2 + (y_{rect} - y_{hi})^2 \tag{2}$$

The H-compliance of j^{th} conformation can be added as a fitness term $E_{H-compliance}$ to the function of *eqn. 1.* $E_{H-compliance}$ is the average of the H-compliance of the conformation where n_h is the total number of H type residues in the sequence.

$$E_{H-compliance} = H_j / n_h \tag{3}$$

P-compliance. The P-compliance of a P type residue is a measure of how close the P residue is to any of the sides ($x_p min$, $x_p max$, $y_p min$ and $y_p max$) of a *P-boundary rectangle* (Fig. 1(b)). P-compliance is defined with the help of P-boundary rectangle rather than H-core because the P residues are located close to the outer periphery of a conformation and it is not possible to measure this from a H-core centre. The smaller the value of P-compliance, the closer it

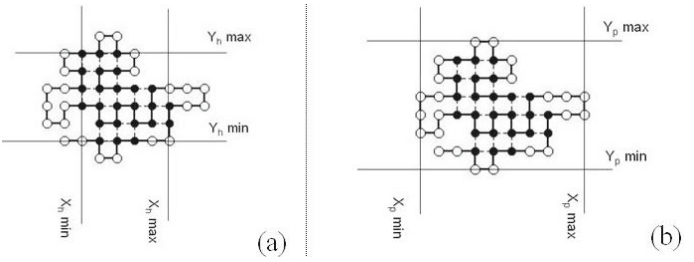

Fig. 1. Boundary rectangles for (a) H residues. (b) P residues.

is to the P-boundary rectangle. The sum of the P-compliance of all the P-type residues gives the P-compliance of the conformation under consideration.

For measuring the P-compliance p_i, we determine the minimum distance of an i^{th} P-type residue from P-boundary rectangle shown in Fig. 1(b).With coordinates of i^{th} P residue given as (x_{pi}, y_{pi}), the P-compliance of the j^{th} conformation is given as follows

$$P_j = \sum_{i=1}^{n_p} p_i = \sum_{i=1}^{n_p} (\min\{|x_p min - x_{pi}|, |x_p max - x_{pi}|, \\ |y_p min - y_{pi}|, |y_p max - y_{pi}|\}) \tag{4}$$

Again, to determine the corresponding fitness term $E_{P-compliance}$ to be included in *eqn. 1*, the average P-compliance of the conformation is used as given below. The term n_p is the total number of P residues in the individual.

$$E_{P-compliance} = P_j / n_p \tag{5}$$

Finally, the 'Modified Fitness Function (MFF)' for the j^{th} conformation which is a total fitness is given below

$$E_j^{mff} = aE_{TN} + E_{H-compliance} + E_{P-compliance} \tag{6}$$

Here E_{TN} is fitness for the j^{th} confirmation computed from *eqn. 1*. The original fitness function E_{TN} is multiplied by high integer constant value a so aE_{TN} of *eqn. 6* will remain integer and the later parts of *eqn. 6* will be in decimal and it will ensure that the original fitness term E_{TN} continues to have an influential effect on the MFF.

2.2 Schema Preservation

A conformation in any configuration (2D lattice, FCC etc) can be represented as a two dimensional matrix M_{Ci}. In general, if X is the set of all possible moves and size$|X| = n$, then $X_q \in X$ with $q = 1, 2, \cdots, n$. For relative 2D encoding, $n = 3$ and we have the set of possible moves as $(X_0 = F, X_1 = L, X_2 = R)$. If l is

the length of the sequence (i.e. number of residues), then for 2D relative encoding a conformation will have only $(l-2)$ moves [19], because the first move is always F. Thus the size of matrix M_{Ci} will be $(l-2) \times n$. The matrix M_{Ci} is populated as $M_{Ci} = [a_{rq}]_{r=1,\ldots,l-2, q=1,\ldots,n}$. Now, if the r^{th} position of a conformation C_i is X_q , then $a_{rq} = \epsilon \times F(C_i)$ otherwise $a_{rq} = 0$. The constant $\epsilon = -1$ and $F(C_i)$ is the fitness of the i^{th} conformation.

To find out the highly probable schema that is likely to occur in subsequent generation, we obtain a matrix $\pi = \sum_{i=1}^{N} M_{Ci}$ or the entire population, N. Multiplying π with a column vector $[1 \quad 1 \quad 1]^T$, we obtain another column vector $\Lambda = \pi \times [1 \quad 1 \quad 1]^T = [\rho_1 \quad \cdots \quad \rho_{l-2}]^T$. The r^{th} row of Λ presents the cumulative weight, ρ_r of r^{th} position of all conformations. To obtain the probability of occurrence of each move at a given position, we multiply each row of matrix π by $(1/\rho_r)$ to obtain another matrix π'. This matrix π' is important because it contains the relevant information about the probability of occurrence of a schema. To establish a move in a given position is highly probable, we define a cut off value χ $(= 0.8)$. If any element of matrix π' has value greater than χ, then that position is fixed for finite number (=50) of generations. However, if the probability of this position changes after 50 iterations, we may get a new schema. However, based on the Hollands schemata theorem which underpins the working of MA, we note that the probability of changes in schema will reduce as the solution converges. This novel technique of schema preservation enables us to establish the highly probable moves in a conformation. By applying the technique, if two moves (say, first and third move) are fixed as say F, then for a sequence of length 10 the conformation would be FxFxxxxxx where x is a *dont care* move. This fixing of schema significantly reduces the search space (hence computational time) and restrict search to those individuals which contain highly probable moves.

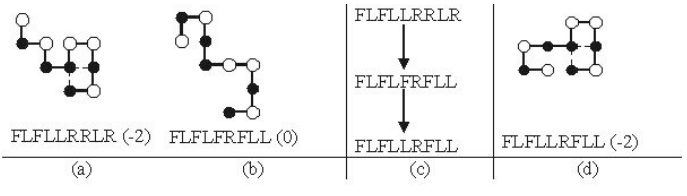

Fig. 2. (a) Conformation from the best individual set with fitness of -2, (b) Newly generated random conformation with a fitness of 0, (c) Implementing move changes (F to L) for the confirmation of (b), (d) Modified new individual conformation with a high fitness of -2

2.3 Guided Search Space

Realizing that best individuals will preserve best schemas, rather than a purely random generation of new interim individuals, we propose a guided search. For this, a record is maintained of all those individuals having fitness equal to the

current best fitness value. This set of best individuals is used as templates for generating New Fit Individual (NFI). The strategy is best illustrated by considering an arbitrary toy sequence HPHPPHHPHP. As shown in Fig. 2(a), consider a conformation FLFLLRRLR with fitness -2 from the current best individuals set. Next, we consider a randomly generated conformation to be, say, FLFLFRFLL with fitness 0 (Fig. 2(b)). If the 5th move (i.e. F) of this conformation is changed with the corresponding move (i.e. 5th move, L) of the best individual, the resulting individual FLFLLRFLL can be seen to improve its fitness equal to the fitness of the best individual (i.e. -2). The whole process is shown in Fig. 2(c) and the conformation is shown in Fig. 2(d).

Simple GA (SGA) is essentially based on fitness function defined by *eqn. 1* is modified incorporating all of the above three features which we refer as enhanced GA (EGA). Its fitness function is given by *eqn. 6* and it incorporates the schema preservation features of 2.2 and the guidance of 2.3.

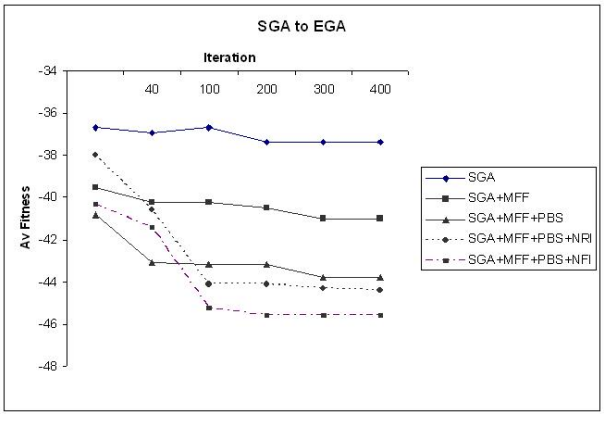

Fig. 3. Effect of enhancement features on SGA applied to benchmark sequence *b7*

3 Results

For investigations, we consider a set of benchmark sequences (*b1, b2, b3, b4, b5, b6, b7*) from [13,2] and also two non-benchmark sequences (*n2, n3*) from [2] given in Appendix. We begin investigations of the performance of the algorithm by selecting one bench sequence (i.e. sequence *b7*) from a set of benchmark protein sequences. Sequence *b7* is chosen as it is a reasonably long sequence with 85 residues possessing necessary level of complexity to discriminate between various techniques.

i) Enhancements to SGA. For various enhancements to SGA under investigations, we compute the average fitness value of the best 10 conformations in each generation. Fig. 3 shows the variation of average fitness value as a function

Table 1. Comparison of SGA and simple MAs using two local search approaches, i.e. SGA+PWI and SGA+TS

Test Run	Avg. Iteration					Success Rate
	b1	n2	n3	b2	b3	
SGA	11.4	34.2	29	16	84.2	86.20%
SGA+PWI	13	32.4	18	12.2	20.2	96.15%
SGA+TS	6.4	10.2	8.4	6.4	9	100%

of generation in different cases. We see that SGA (without any enhancement), has a poor performance. The performance improves by progressively applying improvements (i) modified fitness function (SGA+ MFF) (ii) preserved best schema (SGA+MFF+PBS). (iii) Add New Fit Individuals (NFI) using the technique explained in sec 2.3 (SGA+MFF+PBS+NFI). Finally, for the sake of comparison, instead of NFI we add New Random Individuals (NRI) to the population (SGA+MFF+PBS+NRI). We see that (SGA+MFF+PBS+NFI) has the best performance.

ii) Effect of local search on simple MA. We will first study the effect of local search on a simple GA and then compare its performance with the two variants of simple MA (LS with PWI and TS). For investigating the performance of the two variants of the proposed MA, i.e. PMA and TMA, we randomly consider three benchmark sequences (*b1, b2, b3*) and two new sequences (*n2, n3*) for experimentation. The results are shown in Table 1. Our aim is to obtain five values of E* value for each of the sequences. Hence, for each of the 5 sequences, number of simulation runs were carried out till we achieved 5 successful (which results in E*) results. The number of iterations required for the successful runs are averaged and shown in Table 1. For evaluating the algorithm performance in another manner, we further define a new measure called $SuccessRate = ((25 \div totalnumberofattempts) \times 100)$. The constant 25 appears in the definition because each of the 5 sequences are successful 5 times. But the total number of attempts required to achieve this 25 successful runs are different for different algorithms. In our studies, we found that SGA achieves a success rate of 86.2% (25 optimum values in 30 attempts) whereas SGA+PWI had a success rate of 96.15% (25 optimum values in 26 attempts) and SGA+TS had 100% of success. These results are tabulated in Table 1. It shows that although both variants of simple MA perform better than SGA, simple MA with TS as local search has best performance.

Enhancement to SGA showed that EGA incorporating: (i) modified fitness function (ii) preserving schema and (iii) guided search performs better than the simple GA. Hence we will consider this EGA for further investigations. Effect of the two local search techniques, PWI and TS on EGA performance is studied using several benchmark protein sequences given in Appendix. The results of the studies are presented in Table 2.

Table 2. Results for EGA, PMA and TMA (when an optimal is not reached the number of iteration, each of the algorithms first time reached the suboptimal are given in bracket)

	EGA		PMA		TMA	
Seq.	Iteration	Fitness	Iteration	Fitness	Iteration	Fitness
b1	11	-9	9	-9	4	-9
n2	8	-4	13	-4	2	-4
n3	7	-8	6	-8	1	-8
b2	6	-9	6	-9	3	-9
b3	19	-8	8	-8	2	-8
b4	36731	-13	3457	-14	10	-14
b5	17251	-21(2687)	14189	-21(1291)	1507	-23
b6	179	-21	441	-21	11	-21

From the Table, it can be observed that while in general, EGA and PMA have a somewhat similar performance, TMA shows a significant improvement over other techniques with regard to both, the optimum value and also the number of iteration required in reaching that optimal value.

iii) Comparison of TMA with other approaches. Since the previous experiments establish that TMA has the best performance, this approach is investigated further by comparing it with five other known approaches, i.e. guided GA (GGA), guided Tabu search (GTB), expected Monte Carlo (EMC), simple GA (SGA), and Monte Carlo (MC), which have been reported in literature. In each of these simulations, for each run, 200 randomly generated individuals were included. The algorithm is set to run up to a maximum of 6 hours if optimum is not reached earlier. The time limit as a termination condition ensures that that all algorithms irrespective of their complexity are compared for similar time duration. Table 3 gives the comparisons. The best results obtained by TMA are compared with the results given in [5,19]. It can be seen that for all smaller

Table 3. TMA compared with other search algorithms (number of iteration required for GGA to reach the fitness are given in bracket and E* denotes optimal fitness value)

		TMA						
Seq.	E*	Iteration	Fitness	GGA	GTB	EMC	GA	MC
b1	-9	4	-9	-9(2)	-9	-9	-9	-9
b2	-9	3	-9	-9(83)	-9	-9	-9	-9
b3	-8	2	-8	-8(124)	-8	-8	-8	-8
b4	-14	10	-14	-14(814)	-14	-14	-12	-13
b5	-23	1507	-23	-23(3876)	-23	-23	-22	-20
b6	-21	11	-21	-21(720)	-21	-21	-21	-21

sequences, TMA outperforms GGA (Guided GA) which is the best result of [5,19] for both, fitness and number of iterations required.

4 Conclusion

In this paper, we show that MA with superior local search proves very useful for PSP prediction. To make MA suitable for the complex PSP problem, the global search algorithm is enhanced by a novel fitness function which includes two new measures: H-compliance and P-compliance for the H and P residues. The enhancements also include novel techniques for schema preservation and guided search. Number of benchmark sequences and new sequences are used for investigations. Comparison with other known search algorithms for PSP problem is also reported. We observe that the enhanced global search (with the three features of novel fitness function, schema preservation and guided search) and incorporating tabu search for local optimization has a superior performance compared to other known algorithms. Experiment with other complex sequences are in progress.

References

1. Dill, K.A., Bromberg, S., Yue, K., Chan, H.S., Ftebig, K.M., Yee, D.P., Thomas, P.D.: Principles of protein folding - a perspective from simple exact models. Protein Science 4(4), 561–602 (1995)
2. Cutello, V., Nicosia, G., Pavone, M., Timmis, J.: An immune algorithm for protein structure prediction on lattice models. IEEE Transaction on Evolutionary Computation 11(1), 101–117 (2007)
3. Berger, B., Leighton, T.: Protein folding in the hydrophobic-hydrophilic (hp) model is np-complete. Journal of Computational Biology 5(1), 27–40 (1998)
4. Lopes, H.S., Scapin, M.P.: An enhanced genetic algorithm for protein structure prediction using the 2d hydrophobic-polar mode. In: Talbi, E.-G., Liardet, P., Collet, P., Lutton, E., Schoenauer, M. (eds.) EA 2005. LNCS, vol. 3871, pp. 238–246. Springer, Heidelberg (2006)
5. Hoque, M., Chetty, M., Dooley, L.: A new guided genetic algorithm for 2d hydrophobic-hydrophilic model to predict protein folding. In: IEEE Congress on Evolutionary Computation, vol. 1, pp. 259–266 (2005)
6. Jiang, T., Cui, Q., Shi, G., Ma, S.: Protein folding simulations of the hydrophobic-hydrophilic model by combining tabu search with genetic algorithms. The Journal of chemical physics 119(8), 4592–4596 (2003)
7. Zhao, X.: Advances on protein folding simulations based on the lattice hp models with natural computing. Applied Soft Computing 8(2), 1029–1040 (2007)
8. Krasnogor, N., Smith, J.: A tutorial for competent memetic algorithms: model, taxonomy, and design issues. IEEE Transactions on Evolutionary Computation 9(5), 474–488 (2005)
9. Tang, M., Yao, X.: A memetic algorithm for vlsi floorplanning. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 37(1), 62–69 (2007)
10. Hasan, S.M.K., Sarker, R., Essam, D., Cornforth, D.: Memetic algorithms for solving job-shop scheduling problems. Memetic Computing 1(1), 69–83 (2008)

11. Fallahi, A.E., Prins, C., Calvo, R.W.: A memetic algorithm and a tabu search for the multi-compartment vehicle routing problem. Computers and Operations Research 35(5), 1725–1741 (2008)
12. Elbeltagi, E., Hegazy, T., Grierson, D.: Comparison among five evolutionary-based optimization algorithms. Advanced Engineering Informatics 19(1), 43–53 (2005)
13. Krasnogor, N., Blackburne, B.P., Burke, E.K., Hirst, J.D.: Multimeme algorithms for protein structure prediction. In: Guervós, J.J.M., Adamidis, P.A., Beyer, H.-G., Fernández-Villacañas, J.-L., Schwefel, H.-P. (eds.) PPSN 2002. LNCS, vol. 2439, pp. 769–778. Springer, Heidelberg (2002)
14. Pelta, D.A., Krasnogor, N.: Recent Advances in Memetic Algorithms. In: Multimeme Algorithms Using Fuzzy Logic Based Memes For Protein Structure Prediction, pp. 49–64. Springer, Berlin (2005)
15. Krasnogor, N., Hart, W., Smith, J., Pelta, D.: Protein structure prediction with evolutionary algorithms. In: Proceedings of the genetic and evolutionary computation (1999)
16. Smith, J.: Protein structure prediction with co-evolving memetic algorithms. In: The 2003 Congress on Evolutionary Computation, vol. 4, pp. 2346–2353 (2003)
17. Smith, J.E.: The Co-Evolution of Memetic Algorithms for Protein Structure Prediction. Studies in Fuzziness and Soft Computing, vol. 166. Springer, Heidelberg (2005)
18. Greenwood, G.W., Shin, J.M.: On the evolutionary search for solutions to the protein folding problem. Morgan Kaufmann, San Francisco (2003)
19. Hoque, M.T.: Genetic algorithm for ab initio protein structure prediction based on low resolution models. PhD thesis, GSIT, Monash University (2007)

Appendix

Benchmark sequences (b1, b2, b3, b4, b5, b6, b7) and non benchmark sequences (n2, n3) (E* denotes optimal fitness value)

Inst.	Size	Sequence	E*	Ref
b1	20	2(hp)p2hph2php2hp2(ph)	-9	[2,13]
b2	24	2h2ph2p5(h2p)2h	-9	[2,13]
b3	25	2ph2p3(2h4p)2h	-8	[2,13]
b4	36	3p2h2p2h5p7h2p2h4p2h2ph2p	-14	[2,13]
b5	48	2ph2p2h2p2h5p10h6p2(2h2p)h2p5p	-23	[2,13]
b6	50	2h3(ph)p4hp2(h3p)h4p2(h3p)hp4h3(ph)p2h	-21	[2,13]
b7	85	4h4p12h6p12h3p12h3p12h3ph2p2h2p2h2phph	-53	[2]
n2	18	2h5p2h3ph3php	-4	[2,13]
n3	18	hphp3h3p4h2p2h	-8	[2,13]