Protein structure prediction in simplified models with estimation of distribution algorithms

Roberto Santana, Pedro Larrañaga, José A. Lozano

Intelligent Systems Group

Department of Computer Science and Artificial Intelligence

University of the Basque Country

P.O. Box 649, 20080 San Sebastián - Donostia, Spain

rsantana@si.ehu.es, Pedro.Larranaga@ehu.es, lozano@si.ehu.es

Abstract

In this paper we discuss the use of probabilistic modeling in the solution of the protein structure prediction problem. Estimation of distribution algorithms (EDAs) based on Markov models are presented as an alternative to other nature-inspired optimization algorithms for the solution of protein structure simplified models.

1 Introduction

Solving the protein structure problem consists in finding the 3-d structure of the protein given its sequence of aminoacids. The problem has proven itself to be a great challenge in bioinformatics. Given the difficulties associated with modelling the molecular potentials and other functions that influence the folding process, simplifications are usually required. These simplifications change according to the level of detail with which the proteins are modelled, and are in general difficult to scale to a fine level of detail.

In this paper we concentrate on a class of coarse-grained or simplified models that have been extensively used to study approximations of the protein folding problem. This is the class of the hydrophobic-polar (HP) model [7]. The HP model is based on the fact that hydrophobic interactions are a dominant force in protein folding. The search for the optimal configuration is transformed into the search for the optimum of an objective function that takes into account the HP interactions that arise in the model.

Although more complex models have been proposed, the HP model remains a focus of research in computational biology [2, 3, 5]. Among other reasons, it is considered a useful model of an exhaustive sequence-structure map for the study of evolution [3], and it has been acknowledged that the hydrophobicity pattern in real proteins has statistical properties similar to those of 2-d HP model proteins [5]. Also in evolutionary computation [9, 10, 11, 20, 21, 23] the model keeps to be employed given its simplicity and its usefulness as a test bed for new evolutionary optimization approaches.

EDAs [13, 15] construct an explicit probability model of a set of selected solutions. This model can capture, by means of probabilistic dependencies, relevant interactions among the variables of the problem. The model can be conveniently used to generate new promising solutions. In this paper we show that an EDA that uses a Markov probabilistic model can outperform other nature-inspired methods in the solution of the 3-d HP model, and of a variant of the HP model called functional model protein defined in two dimensions. Both models are defined in regular lattices.

Actas del IV Congreso Español sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados, MAEB2005, pp.245-252
ISBN: Vol I:84-9732-440-4 Vol II: 84-9732-467-6

2 Previous evolutionary methods

There exist a number of nature-inspired approaches to the simplified protein structure problem.

In [23], the authors proposed a GA that used heuristic based crossover and mutation operators. The GA was able to outperform a number of variants of Markov chain methods at different sequences. In [4], an evolutionary algorithm for the 3-d HP problem was proposed. By using a backtracking based repairing procedure, the algorithm guarantees that the search is constrained to the space of legal solutions.

The multimeme algorithm (MMA) for protein structure prediction [12] is a GA combined with a set of local searches. From this set, the algorithm self-adaptively selects which local search heuristic to use for different instances, states of the search, or individuals in the population. This algorithm was used to find solutions of the functional model protein. A relevant issue of this algorithm is the use of a contact map memory as a way to collect and use important problem information. Contact maps abstract the geometric details of the structures, keeping only the essential topological features of the configurations.

In [16], MMA was extended by the incorporation of fuzzy-logic based local searchers. The modifications allowed to obtain a more robust algorithm that improved previous MMA results in the protein structure prediction problem. Memetic algorithms were also combined with a population of rules [22] to solve the HP model in a 2-d triangular lattice. The proposed algorithm outperformed simple versions of GAs and memetic algorithms.

Inmune algorithms (IA) [6] have been recently proposed for the HP problem. These evolutionary algorithms inspired in the theory of clonal selection, use hyper-macromutation and aging (an operator that resembles GAs elitism) as important operators to proceed the search. In [6] the algorithm is shown to find the optimal configurations of the regular 2-d HP model for the smallest problems. The algorithm fails to find the optimum for the largest instances.

An algorithm that incorporates, to a certain scale, the modelling step is the ant colony optimisation (ACO) method presented in [20, 21]. In this approach, the simulated ants contruct candidate conformations for a given HP protein sequence, apply local search to achieve further improvement, and update a probability value based on the quality of the solutions found. In ACO's terminology, this value is called the pheromone trail.

3 Simplified models

This section briefly introduces the HP and functional model protein and the codification used to represent the solutions.

3.1 The HP and functional model protein

In the HP model a sequence comprises residues of only two types: hydrophobic (H) and hydrophilic or polar (P). Residues are located in regular lattice models forming self-avoided paths. There is a zero contact energy between P-P and H-P pairs. Different values can be taken to measure the interaction between hydrophobic non-consecutive residues; a common choice that we use in this paper is -1. The energy interactions can be represented by the following matrix:

$$E = \begin{array}{c|cc} & \text{H} & \text{P} \\ \hline \text{H} & -1 & 0 \\ \text{P} & 0 & 0 \end{array}$$

The function evaluation of a given configuration or protein conformation is simplified to the sum of every two hydrophobic residues that are non-consecutive nearest neighbors on the lattice.

In this paper, we consider the 2-d regular lattice. In the linear representation of the sequence, hydrophobic residues are represented with the letter H and polar ones with P. In the graphical representation, hydrophobic proteins are represented by black beads and polar proteins by white beads. Figure 1 shows an optimal folding for the sequence $S1 = HPHPPHHPHPPHPHHPPHPH$.

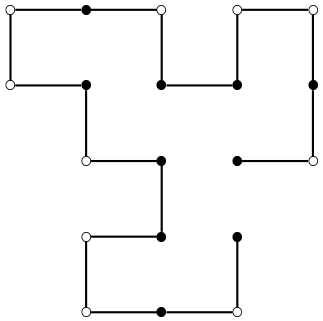

Figure 1: Best solution found for sequence $HPHPPHHPHPPHPHHPPHPH$

The optimal energy corresponding to this sequence is -9.

3.2 Functional model protein

The functional model protein is a 'shifted' HP model. The name comes from the fact that the model supports a significant number of proteins that can be characterized as functional. This model has native states, some of which are not maximally compact. Thus, in some cases, they have cavities or potential binding sites, a key property that is required in order to investigate ligand binding using these models [8]. The energy matrix associated with the model contains both attractive and repulsive interactions. Its representation is as follows:

$$E = \begin{array}{c|cc} & \text{H} & \text{P} \\ \hline \text{H} & -2 & 1 \\ \text{P} & 1 & 1 \end{array}$$

3.3 Problem representation

We use X_i to represent a discrete random variable. A possible value of X_i is denoted x_i. Similarly, we use $\mathbf{X} = (X_1, \ldots, X_n)$ to represent an n-dimensional random variable and $\mathbf{x} = (x_1, \ldots, x_n)$ to represent one of its possible values. n is the sequence length. For a given sequence and lattice, X_i will represent the relative move of residue i in relation to the previous two residues. Taking as a reference the location of the previous two residues in the lattice, X_i takes values in $\{0, 1, \ldots, a-1\}$, where a is the number of movements allowed in the given lattice. These values respectively mean that the new residue will be located in one of the a number of possible directions with respect to the previous locations. Therefore, values for X_1 and X_2 are meaningless. The locations of these two residues are fixed.

We use 2-d and 3-d regular lattices. For regular d-dimensional lattices, $a = 2d - 1$, where d is the lattice dimension.

4 Estimation of distribution algorithm

EDAs use probability models that estimate the probability of the selected solutions, and are employed to simulate new promising solutions. These algorithms can be seen as a development of GAs. By recombining a subset of selected solutions, GAs are able to process the information learned during the search, and to orient the exploration to promising areas of the search space. However, it has been demonstrated that GAs experience limitations in their capacity to deal with problems where there are complex interactions between different components of the solutions [14].

$D_0 \leftarrow$ `Generate` M `individuals (the initial population) randomly`
$l = 1$
`Do`
  $D^s_{l-1} \leftarrow$ `Select` $N \leq M$ `individuals from` D_{l-1} `according to a selection method`
  $p_l(\boldsymbol{x}) = p(\boldsymbol{x}|D^s_{l-1}) \leftarrow$ `Estimate the joint probability of selected individuals`
  $D_l \leftarrow$ `Sample` M `individuals (the new population) from` $p_l(\boldsymbol{x})$
`Until a stop criterion is met`

Algorithm 1: Main scheme of the EDA approach

Although the results of the GAs can be improved by designing heuristic based genetic operators, the improvements are constrained by the narrow scope of application of this type

of knowledge based operators. A more robust solution is the use of evolutionary algorithms able to use probabilistic models of the search space. The success of EDAs in the solution of different practical problems has been documented in the literature [13]. There exist as well recent successful applications of EDAs in bioinformatics [1, 17, 19, 18], all these papers refer to the use of EDAs as optimization methods.

The general scheme of the EDA approach is shown in Algorithm 1. Individuals are represented using the vector representation introduced in Section 3.3. A key characteristic and crucial step of EDAs is the construction of the probabilistic model. These models may differ in the order and number of the probabilistic dependencies that they represent.

We propose to consider a k-order Markov model where the configuration of variable X_i depends on the configuration of the previous k variables, where $k \geq 0$ is a parameter of the model. $p(\mathbf{x})$ can be factorised as follows:

$$p_{MK}(\mathbf{x}) \tag{1}$$
$$= p(x_1, \ldots, x_{k+1}) \prod_{i=k+2}^{n} p(x_i \mid x_{i-1}, \ldots, x_{i-k})$$

The main feature of the EDA approach to the protein structure problem is the use of the Markov model to estimate and sample the distribution. However, other issues have to be considered. For instance, in the chosen representation there might be invalid vectors that correspond to self-intersecting sequences. To enforce the validity of the solutions, we employ a variation of the backtracking method used in [4]. A solution is incrementally repaired in such a way that the self-avoidance constraint is fulfilled. At position i, the backtracking call is invoked only if self-avoidance cannot be fulfilled with any of the possible assignments to X_i. The order of the assignment of variables is random. If all the possible values have been checked, and self-avoidance is not fulfilled yet, backtracking is invoked.

On the other hand, if the number of backtracking calls have reached a pre-specified threshold, the repair procedure is abandoned. This is a compromise solution for situations where the repair procedure can be too costly in terms of time. Further details about the original backtracking algorithm can be found in [4].

In our implementation of EDAs, the truncation selection of parameter $T = 0.1$ is used. Let M be the population size. In this type of selection, the best $N = T \cdot M$ individuals according to their function evaluations, are selected. We use best elitism, a replacement strategy where the population selected at generation t is incorporated into the population of generation $t + 1$. Thus, only $M - N$ individuals are generated at each generation except the first one. We name MK-EDA$_k$ the Markov EDA, where k is the order of the interactions. This algorithm has been implemented in C++ language. All the experiments have been executed in a Pentium III processor with 933MHz.

5 Experimental results

In this section we present the results of the MK-EDA$_k$ in the optimization of different instances, and compare these results with those achieved by other evolutionary methods.

5.1 Problem instances

The HP instances used in our experiments (see Table 1), have been previously used in [4, 20, 21, 23].

Table 2 shows sequences $s9$-$s19$ that belong to the functional model protein and were previously used as a benchmark in [12]. All these intances have size 23. They are an example of a challenging set of problems with only one solution.

5.2 Results for the 3-d HP model

In the following experiment we investigate the behaviour of EDAs in the solution of the HP model in the regular 3-d lattice. MK-EDA$_k$ ($k \in \{1, 2\}$) is compared with the hybrid GA that uses a repair based approach with absolute encoding [4]. Although in [4] experiments with the relative encoding (the same used by

inst.	size	sequence
$s1$	20	$HPHPPHHPHHPHPHHPPHPH$
$s2$	24	$HHPPHPPHPPHPPHPPHPPHPPHH$
$s3$	25	$PPHPPHHP^4HHP^4HHP^4HH$
$s4$	36	$P^3HHPPHHP^5H^7PPHHP^4HHPPHPP$
$s5$	48	$PPHPPHHPPHHP^5H^{10}P^6$ $HHPPHHPPHPPH^5$
$s6$	50	$HHPHPHPHPH^4PHP^3HP^3HP^4$ $HP^3HP^3HPH^4\{PH\}^4H$
$s7$	60	$PPH^3PH^8P^3H^{10}PHP^3$ $H^{12}P^4H^6PHHPHP$
$s8$	64	$H^{12}PHPH\{PPHH\}^2PPH\{PPHH\}^2$ $PPH\{PPHH\}^2PPHPHPH^{12}$

Table 1: HP instances used in the experiments

name	opt.	sequence
$s9$	-20	$PHPPHPPHHHHPPHPPHPHPPHH$
$s10$	-17	$PHPPHPPHHHHHPPPPHPPHPPH$
$s11$	-16	$HPHPHPHHHPPHPPPHPHHPPHH$
$s12$	-20	$HHHPHHHPPHHPPPHPHPPHHHH$
$s13$	-17	$PHPPPPPPHPHHPHPHHHHPHPH$
$s14$	-13	$HHPHPPHPPPPHPPPPHPPPHHH$
$s15$	-26	$PHPHHPHHHHHHPPHHHPHHHHH$
$s16$	-16	$HPHPPPHHHHPHPPPPHPHPHHH$
$s17$	-15	$PHPHHPHHPHHPHPHPHPPPPPH$
$s18$	-14	$HPHPHPPPPPHHPPPHPHPHPHH$
$s19$	-15	$PHPPHHHPHPPHPHHPHPPPPPH$

Table 2: Functional model protein instances of size 23 used in the experiments

EDAs) were also conducted, the hybrid GA used in our comparison was better than any of the other five versions of the hybrid GA tested.

The hybrid GA used a population size of 100 individuals and a maximum of 10^5 evaluations. This is not an appropriate population size for EDAs. We use a population size of 5000 individuals and a maximum of 1000 generations. In almost all sequences, this implies a higher number of function evaluations than the 10^5 used in [4]. The average number of evaluations needed by EDAs was between $6 \cdot 10^4$ for the smallest instances and 10^6 for the bigger ones.

The results are shown in Table 3. In this table, the best values achieved by each algorithm are represented by $H(\mathbf{x}^*)$. In this case the optimum is not known and it is critical to establish whether differences among the methods are statistically significant. We have used the Kruskal-Wallis test to accept or reject the null hypothesis that the samples have been taken from equal populations. The test significance level was 0.01. For all the instances considered significant statistical differences have been found between the hybrid GA results and those achieved by the Markov (MK-EDA$_1$ and MK-EDA$_2$) EDAs. The EDAs have a better average of solutions, showing that the algorithms clearly outperform the hybrid GA. Furthermore, as can be observed in Table 3, for sequences $s5$, $s6$ and $s8$ EDAs find new best solutions.

5.3 Results for the 2-d functional model protein

Now we evaluate the EDA for the functional model protein instances. The number of evaluations needed by MMA to optimize the functional model protein instances are shown in Table 4. The results correspond to the best out of five experiments where the optimum has been reached at least once. To make a fair comparison, we run our algorithm in similar conditions. We find the population size (M) for which the EDA finds the optimum in at least one of the five experiments. Additionally, we present the number of times (S) the optimum was found with the given population size and the number of evaluations (e) for the best run where it was found.

For the functional model protein instances, we found that the simple EDA with $k = 1$ was able to improve the results achieved by MMA. In Table 4, it can be appreciated that the EDA is able to find the optimum for all the instances with a number of evaluations that is, in all cases, lower than the number needed by MMA. Another observation is that, for eight of the eleven instances treated, the optimum could be found with the minimum population size tested, i.e. $M = 500$.

6 Discussion

We have shown that EDAs based on Markov models are able to deal with the simplified protein structure problem. The approach of using probabilistic dependencies to improve the search efficiency has a strong theoretical basis. Its operational simplicity and applicability makes it an advantageous method in relation to other widely-applied evolutionary algorithms. The experiments confirm that EDAs are a feasible alternative for the HP and functional model protein problems. The same strategy presented in this paper can be extended to other optimisation problems that arise in bioinformatics.

	EDA				*MMA*
inst.	k	S	M	e	e
$s9$	1	4	500	12650	15170
$s10$	1	5	500	2750	61940
$s11$	1	1	1000	35900	132898
$s12$	1	3	1000	15900	66774
$s13$	1	4	500	20950	53600
$s14$	1	5	500	5420	32619
$s15$	1	1	500	19450	114930
$s16$	1	1	1500	10350	28425
$s17$	1	1	500	4950	25545
$s18$	1	2	500	8950	111046
$s19$	1	5	500	2950	52005

Table 4: Results for the 2-d functional model protein instances

	hybrid GA		MK-EDA$_1$		MK-EDA$_2$	
	$H(\mathbf{x}^*)$	$mean \pm \sigma$	$H(\mathbf{x}^*)$	$mean \pm \sigma$	$H(\mathbf{x}^*)$	$mean \pm \sigma$
$s1$	-11	-10.52 ± 0.54	-11	-10.80 ± 0.40	-11	-10.82 ± 0.38
$s2$	-13	-11.28 ± 0.90	-13	-12.56 ± 0.75	-13	-12.02 ± 0.94
$s3$	-9	-8.54 ± 0.64	-9	-8.90 ± 0.30	-9	-8.96 ± 0.19
$s4$	-18	-15.76 ± 1.05	-18	-16.34 ± 0.79	-18	-16.40 ± 0.80
$s5$	-28	-24.60 ± 1.57	$\mathbf{-29}$	-27.10 ± 1.00	$\mathbf{-29}$	-27.24 ± 0.92
$s6$	-26	-23.02 ± 1.48	-28	-25.68 ± 1.24	$\mathbf{-29}$	-25.70 ± 1.26
$s7$	-49	-41.18 ± 2.75	-49	-46.52 ± 1.25	-49	-46.30 ± 2.04
$s8$	-46	-40.40 ± 2.50	-50	-46.16 ± 2.16	$\mathbf{-52}$	-46.78 ± 2.28

Table 3: Results of the EDAs and the hybrid GA in the 3-d lattice

Acknowledgements

The authors thank Carlos Cotta for providing the experimental results used in the comparison between EDAs and the hybrid GA. This work was supported in part by the Spanish Ministerio de Ciencia y Tecnología under grant TIC2001-2973-C05-03, by SAIOTEK-PEO4UN25 and Etortek-Biolan projects from the Basque Government, and by the University of the Basque Country under grant 9/UPV 00140.226-15334/2003.

References

[1] R. Blanco, P. Larrañaga, I. Inza, and B. Sierra. Selection of highly accurate genes for cancer classification by estimation of distribution algorithms. In *Proceedings of the Workshop 'Bayesian Models in Medicine' held within AIME 2001*, pages 29–34, 2001.

[2] H. Cejtin, J. Edler, A. Gottlieb, R. Helling, H. Li, J. Philbin, N. Wingreen, and C. Tang. Fast tree search for enumeration of a lattice model of protein folding. *Journal of Chemical Physics*, 116:121–144, 2002.

[3] H. S. Chan and E. Bornberg-Bauer. Perspectives on protein evolution from simple exact models. *Applied Bioinformatics*, 1(3):121–144, 2002.

[4] C. Cotta. Protein structure prediction using evolutionary algorithms hybridized with backtracking. In *Artificial Neural Nets Problem Solving Methods*, volume 2687 of *Lecture Notes in Computer Science*, pages 321–328. Springer Verlag, 2003.

[5] Y. Cui, W. H. Wong, E. Bornberg-Bauer, and H. S. Chan. Recombinatoric exploration of novel folded structures: A heteropolymer-based model of protein evolutionary landscapes. *Proceedings of the National Academy of Sciences*, 99:809–814, 2002.

[6] V. Cutello, G. Nicosia, and M. Pavone. An immune algorithm with hyper-macromutations for the Dill's 2d Hydrophobic-Hydrophilic model. In *Proceedings of the 2004 Congress on Evolutionary Computation CEC-2004*, volume 1, pages 1074–1080, Portland, Oregon, 2004. IEEE Press.

[7] K. A. Dill. Theory for the folding and stability of globular proteins. *Biochemistry*, 24(6):1501–1509, 1985.

[8] J. D. Hirst. The evolutionary landscape of functional model proteins. *Protein Engineering*, 12:721–726, 1999.

[9] M. Khimasia and P. Coveney. Protein structure prediction as a hard optimization problem: The genetic algorithm ap-

proach. *Molecular Simulation*, 19:205–226, 1997.

[10] R. König and T. Dandekar. Improving genetic algorithms for protein folding simulations by systematic crossover. *Biosystems*, 50:17–25, 1999.

[11] N. Krasnogor, B. Blackburne, E. K. Burke, and J. D. Hirst. Algorithms for protein structure prediction. In *Parallel Problem Solving from Nature - PPSN VII*, volume 2439 of *Lecture Notes in Computer Science*, pages 769–778. Springer Verlag, 2002.

[12] N. Krasnogor, B. Blackburne, E. K. Burke, and J. D. Hirst. Algorithms for protein structure prediction. In J. J. Merelo, P. Adamidis, H.-G. Beyer, J.-L. Fernandez-Villacañas, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature - PPSN VII*, volume 2439 of *Lecture Notes in Computer Science*, pages 769–778, Granada, Spain, 2002. Springer Verlag.

[13] P. Larrañaga and J. A. Lozano. *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*. Kluwer Academic Publishers, 2002.

[14] H. Mühlenbein, T. Mahnig, and A. Ochoa. Schemata, distributions and graphical models in evolutionary optimization. *Journal of Heuristics*, 5(2):213–247, 1999.

[15] H. Mühlenbein and G. Paaß. From recombination of genes to the estimation of distributions I. Binary parameters. In *Parallel Problem Solving from Nature - PPSN IV*, volume 1141 of *Lecture Notes in Computer Science*, pages 178–187. Springer Verlag, 1996.

[16] D. Pelta and N. Krasnogor. *Recent Advances In Memetic Algorithms*, chapter Multimeme algorithms using fuzzy logic based memes for protein structure prediction, pages 49–64. Springer, 2004.

[17] J. M. Peña, J. A. Lozano, and P. Larrañaga. Unsupervised learning of bayesian networks via estimation of distribution algorithms: an application to gene expression data clustering. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 12(1):63–82, 2004.

[18] Y. Saeys, S. Degroeve, D. Aeyels, P. Rouzé, and Y. VandePeer. Feature selection for splice site prediction: A new method using EDA-based feature ranking. *BMC Bioinformatics*, 5:64, 2004.

[19] Y. Saeys, S. Degroeve, D. Aeyels, Y. VandePeer, and P. Rouzé. Fast feature selection using a simple estimation of distribution algorithm: A case study on splice site prediction. *Bioinformatics*, 19(2):ii179–ii188, 2003.

[20] A. Shmygelska, R. A. Hernández, and H. H. Hoos. An ant colony optimization algorithm for the 2D HP protein folding problem. In *Proceedings of the Third International Workshop on Ant Algorithms*, pages 40–53. Springer Verlag, 2002.

[21] A. Shmygelska and H. H. Hoos. An improved ant colony optimization algorithm for the 2D HP protein folding problem. In *Advances in Artificial Intelligence*, volume 2671 of *Lecture Notes in Computer Science*, pages 400–417. Springer Verlag, 2003.

[22] J. Smith. *Recent Advances In Memetic Algorithms*, chapter The co-evolution of memetic algorithms for protein structure prediction, pages 105–128. Springer, 2004.

[23] R. Unger and J. Moult. Genetic algorithms for protein folding simulations. *Journal of Molecular Biology*, 231:75–81, 1993.