An Ecologically-inspired Parallel Approach Applied to the Protein Structure Reconstruction from Contact Maps

César M.V. Benítez
Bioinformatics Laboratory
Federal Technological
University of Paraná
Curitiba, Brazil
cesarbenitez@utfpr.edu.br

Rafael Stubs Parpinelli
Applied Computing Graduate
Program
Santa Catarina State
University
Joinville, Brazil
rafael.parpinelli@udesc.br

Heitor Silvério Lopes
Bioinformatics Laboratory
Federal Technological
University of Paraná
Curitiba, Brazil
hslopes@utfpr.edu.br

ABSTRACT

The Protein Folding Problem (PFP) is considered one of the most important open challenges in Biology and Bioinformatics. This paper describes the application of a parallel ecology-inspired algorithm (pECO) to a hard problem related to the PFP: the protein structure reconstruction from Contact Maps. The fitness function proposed includes information not only about the free-energy of the conformation, but also similarity measures commonly used in classification systems. Experiments were done to evaluate the adequacy of the proposed approach. Results show that the combination of concurrent evolutionary approaches take advantage of both the coevolution effect and the different search strategies. Furthermore, it is observed that parallel processing was not only justified but also essential for obtaining results in reasonable processing time.

CCS Concepts

•Theory of computation → Parallel algorithms; •Computing methodologies → Parallel algorithms; •Applied computing → Bioinformatics;

Keywords

Protein Folding, Parallel Computing, Computational Intelligence, Contact Maps, 3D-AB *Off-Lattice* model

1. INTRODUCTION

Nowadays, one of the most important and challenging problems in Molecular Biology and Bioinformatics is to obtain a better understanding of the protein folding process. In this process, under physiological conditions, a protein folds into a specific three-dimensional structure, that determines their specific biological functionality. It is known that ill-formed proteins can be completely inactive or even harmful to the organism. This is the case of several diseases, which origin is believed to be the result of the aggregation of such ill-formed proteins, for instance, Alzheimer's disease and some types of cancer. Once more extensive knowledge about the formation of the tertiary structure of proteins can be acquired, important medical and biochemical advancements can take place, including the design of new drugs with specific functionality [8,22].

The structure of proteins is usually represented by all their atoms or by coarse-grained models, such as the AB *off-lattice* model [23]. However, an alternative and compact way is to represent those three-dimensional structures using Contact Maps (CM), which are minimalistic two-dimensional representations [25] capable of reducing the inherent complexity of computational simulations. Notwithstanding, the use of CMs to study the protein folding have been sparsely explored. In recent literature, methods have been developed for their prediction from sequence (for instance, [14, 21]). In addition, [7] presents a novel parallel approach for the induction of *transition rules* of two-dimensional Cellular Automata (2D-CA), using Gene Expression Programming, applied to the Protein Contact Maps prediction and folding pathway simulation. Furthermore, very few research groups proposed heuristic approaches for protein structure reconstruction from native CMs [9, 25].

A reconstruction procedure of the three-dimensional structure of proteins is needed after the CM prediction, which has been proved to be NP-hard [25]. Consequently, metaheuristic approaches seem to be the most reasonable algorithmic choice for dealing with the problem. The general objective of this work is the application of a parallel heterogeneous ecological-inspired approach (called pECO), formerly introduced in [6], to reconstruct the three-dimensional structure of proteins from Contact Maps. Basically, the aim is to find low energy conformations, using Contact Maps as guides.

This paper is organized as follows: Section 2 describes issues related to the Protein Folding Problem; Section 3 describes the ecological-inspired approach; Section 4 shows how the experiments were done; Section 6 presents the results obtained and their analysis; finally, in Section 7 some conclusions and future directions are pointed out.

2. THE 3D-AB OFF-LATTICE MODEL OF PROTEINS

The AB *off-lattice* model was introduced by [23] to represent protein structures. In this model each residue is represented by a single interaction site located at the Cα position. These sites are linked by rigid unit-length bonds ($\hat{b_i}$) to form the protein structure. The three-dimensional structure of a N-length protein is specified by the $N-1$ bond vectors $\hat{b_i}$, $N-2$ bond angles τ_i and $N-3$ torsional angles α_i.

In this model, the 20 proteinogenic amino acids are classified into two classes, according to their affinity to water (hydrophobicity): 'A' (hydrophobic) and 'B' (hydrophilic or polar). This model

GECCO'16 Companion, July 20-24, 2016, Denver, CO, USA

DOI: http://dx.doi.org/10.1145/2908961.2931719

does not describe the solvent molecules, but their effects, such as the formation of a hydrophobic core, are taken into account through interactions between residues, according to their hydrophobicity (species-dependent global interactions). When a protein is folded into its native conformation, the hydrophobic amino acids tend to be packed inside the protein, in such a way to get protected from the solvent by an aggregation of polar amino acids positioned outwards. Interactions between amino acids take place and the energy of the conformation tends to decrease. The conformation tends to converge to its native state, in accordance with the Anfinsen's thermodynamic hypothesis [3].

The energy function of a folding is given by Equation 1, as proposed by [11].

$$E(\hat{b_i};\sigma) = E_{Angles} + E_{torsion} + E_{LJ} \quad (1)$$

$$E_{Angles} = -k_1 \sum_{i=1}^{N-2} \hat{b_i} \cdot \hat{b_{i+1}} \quad (2)$$

$$E_{torsion} = -k_2 \sum_{i=1}^{N-3} \hat{b_i} \cdot \hat{b_{i+2}} \quad (3)$$

$$E_{LJ} = \sum_{i=1}^{N-2} \sum_{j=i+2}^{N} 4\varepsilon(\sigma_i,\sigma_j)(r_{ij}^{-12} - r_{ij}^{-6}) \quad (4)$$

where
E_{Angles} and $E_{torsion}$ are the energies from bond angles and torsional forces, respectively. The species-dependent global interactions are given by the Lennard-Jones potencial (E_{LJ}). $\hat{b_i}$ represents the ith bond that joins the $(i-1)$th and the ith residues and $k_1 = -1$; $k_2 = +1/2$ [11]. r_{ij} represents the distance between ith and jth residues; $\sigma = \sigma_0, ..., \sigma_N$ form a binary string that represents the protein sequence.
$\varepsilon(\sigma_i,\sigma_j)$ is chosen to favor the formation of the hydrophobic core ('A' residues). Thus, $\varepsilon(\sigma_i,\sigma_j)$ is 1 for AA interactions and 1/2 for BB/AB interactions. Finally, it is important to mention that the model can be explored for different values of k_1 and k_2 as stated by [11].

3. THE ECOLOGICAL-INSPIRED APPROACH

The ecologically-inspired algorithm, named ECO, represents a perspective to apply optimization strategies cooperatively in an ecosystemic context [20]. ECO is composed by populations of individuals (Q) and each population evolves according to an optimization strategy. Therefore, individuals of each population are modified according to the mechanisms of intensification and diversification, and the initial parameters, specific to each optimization strategy. The ECO system can be modeled in two ways: homogeneous or heterogeneous. A homogeneous model implies that all populations evolve in accordance to the same optimization strategy, configured with the same parameters. Any change in the strategies or parameters in at least one population characterizes a heterogeneous model.

The ecological inspiration stems from the use of some ecological concepts, such as: habitats, ecological relationships and ecological successions [4] [17]. A habitat is the actual location in the environment where an organism lives and consists of all the physical and biological resources available. In this way, populations of individuals that are scattered in the search space and established in the same region constitute an ecological habitat. The search surface of a problem being optimized represents the environment and, as well as in nature, populations can move around through all the environment. The movement of populations can be observed by changing the values of variables that affect function $f(.)$. However, each population may belong only to one habitat at a given moment of time t. Therefore, by definition, the intersection between all habitats NH at moment t is the empty set. The ecosystem can be composed of several habitats that can also interact to each other, as shown in the upper level of Figure 1.

With the definition of habitats, two categories of ecological relationships can be defined. Intra-habitats relationships that occur between populations inside each habitat, and inter-habitats relationships that occur between habitats. An example of five intra-habitats communication topologies is shown in the intermediate level of Figure 1. Individuals belonging to a given habitat can migrate to other habitats aiming at identifying promising areas for survival and mating. The inter-habitats communication topologies is represented in the upper level of Figure 1. Intra-habitats relationships are responsible for intensifying the search and inter-habitats relationships are responsible for diversifying the search.

It is important to highlight that the concept of interactions between populations is not new. An example is the well-know island model GA [26] and other algorithms that apply the same concept (e.g., PSO [18] and ACO [24]). However, the approach used in ECO differs from the others by presenting a new level of abstraction for the topologies of communication. There are two different topologies of communication, being the intra and inter-habitats communications. The difference between these two topologies can be visually observed an Figure 1. The formation of topologies is done probabilistically and is influenced by the distribution of populations on the surface of function $f(.)$. It can also be observed that the topologies are not static and do not follow a standard formation like ring, star or fully connected as performed by the island model. The topologies are dynamic, i.e., at every given moment t the topologies can assume different patterns [19].

In this work, we use a parallel heterogeneous ecological-inspired approach, called pECO [6], which is a parallel master-slave architecture that allows the application of the computational ecosystem in a reasonable computing time. In this approach, the processing load is divided into several processors (master and slaves), under the coordination of a master processor. Each processor (master or slave) is responsible for initializing the population, and performing

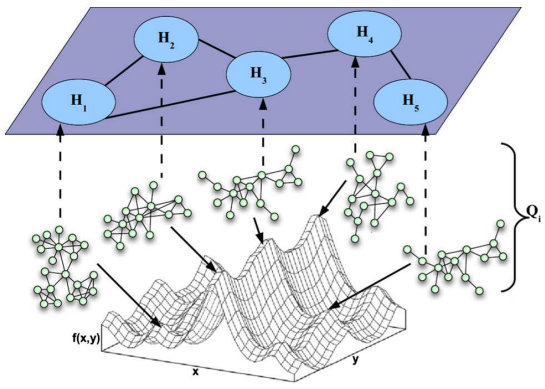


Figure 1: View of a computational ecosystem for optimization. Lower level: problem-dependent search space that defines a hypersurface. Intermediate level: intra-habitats communication topologies where each small circle represents a population. Upper level: five habitats connected through inter-habitats communication topology [20].

the evolutive period of a population independently. The master processor is also responsible for defining the communication topologies between populations and habitats. Figure 2 shows the pECO master-slave topology, where each species represents a population-based approach, S_i represents the i-th slave and n denotes the number of slaves.

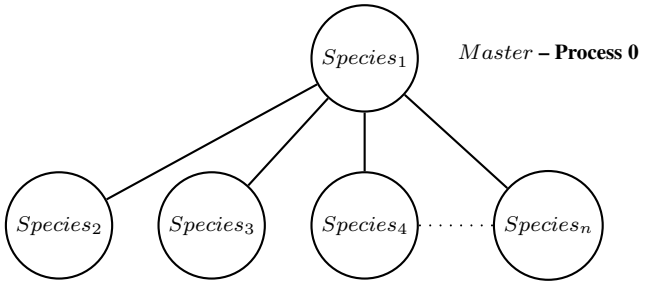


Figure 2: pECO architecture

3.1 Encoding of Candidate Solutions and initial Populations

An important issue when using population-based evolutionary approaches for a given problem is the encoding of the individual that represent a possible solution to the problem. The way variables are encoded can have a strong influence not only in the size of the search space, but also in the dynamics and efficiency of the algorithm. In this work, the encoding of the individuals is defined according to the set of relative bond rotation and torsion angles of the amino acids. Considering the folding of a protein with N amino acids, an individual has $(2N-5)$ variables, such that positions P_1 to P_{N-2} represent the bond rotation angles (τ_i), and P_{N-1} to P_{2N-5} represent the torsion angles (α_i), where τ and $\alpha \in [-\pi, \pi]$. A given conformation of the protein is represented as a set of bond rotation and torsion angles over a threedimensional space. To represent the position of the amino acids, their Cartesian coordinates are defined by a vector (x_i, y_i, z_i). This vector is obtained from the relative bond rotation and torsion angles of an amino acid and position of its predecessor. A folding begins in the origin of the three-dimensional Cartesian coordinates, such that the first amino acid is at $(0, 0, 0)$. The position of the remaining amino acids is computed following the bond rotation and torsion angles encoded in the individual. Figure 3 shows an example of an individual that represents the structure of a folded protein with 13 amino acids.

The initial populations (or swarms) are randomly generated by using the Mersenne Twister random number generator [16], which is known as one of the best generators for this purpose.

3.2 Fitness Function

A Contact Map (CM) is a matrix representation of the closeness between all pairs of amino acids. The CM for a protein sequence with N amino acids is a $N \times N$ binary symmetrical matrix (C), which is defined as follows: each position of the matrix (ith,jth) is 1 if the amino acid pair (ith and jth amino acids) fulfills the connectivity condition. Two amino acids are in contact when their Cα atoms are closer than a threshold distance [25]. CMs are populated primarily with *non-contacts* (or zeros). Therefore, similarity measures between two CMs based, for instance, on the Hamming distance or the Euclidean distance, do not work well because *contacts* (true) and *non-contacts* (false) values carry the same weight. Thus,

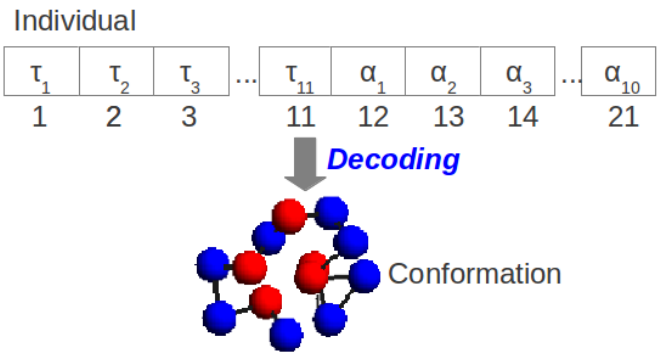


Figure 3: Example of individual-conformation decoding

different measures could be proposed in order to give more importance to the presence of *contacts* in CMs. In this work, a novel fitness function is proposed, that is better suited to this problem. Equation 5 shows the fitness function, using the 3D-AB model.

$$fitness = E(\hat{b_i}; \sigma) * S_C * S_{NC} \quad (5)$$

where: $E(\hat{b_i}; \sigma)$ represents the energy equation of the model. The energy is calculated using the coordinates of the amino acids that compose the structure of the protein which, in turn, is encoded in the individual. S_C, S_{NC} are based on the sensitivity and specificity measures, respectively. S_C and S_{NC} are computed using the input CM (of the target conformation) and the CM of the conformation obtained from the individual. S_C measures the similarity between both CMs from the *contact* point of view (see Equation 6). On the other hand, S_{NC} measures the similarity between both CMs from the *non-contact* point of view (see Equation 7).

$$S_C = \frac{T_C}{T_C + F_{NC}} \quad (6) \qquad S_{NC} = \frac{T_{NC}}{F_C + T_{NC}} \quad (7)$$

where:

- **True contacts (T_C):** number of contacts generated by the *transition rule* that, in fact, are contacts;
- **True non-contacts (T_{NC}):** number of non-contacts generated by the *transition rule* that, in fact, are non-contacts;
- **False contacts (F_C):** it counts the contacts generated by the *transition rule* that, in fact, are non-contacts;
- **False non-contacts (F_{NC}):** it counts the non-contacts generated by the *transition rule* that, in fact, are contacts;

It is important to be aware that a given conformation under evaluation may have collisions between amino acids. Obviously, such conformation is physically invalid, but, anyway, the corresponding individual can carry some promising information and should not be disposed by the search algorithm. Basically, this procedure has five parts: conversion of angles into Cartesian coordinates, computation of the energy, conversion of the Cartesian coordinates to the CM representation, computation of the metrics S_C and S_{NC}, and computation of the fitness. It is important to recall that the conversion of the Cartesian coordinates to the CM representation is done according to the same threshold value of the input CM.

4. COMPUTATIONAL EXPERIMENTS

All experiments done in this work were run in a cluster of 25 networked computers, running a minimal installation of Arch Linux

and using MPICH2 (available in: http://www.mcs.anl.gov/research/-projects/mpich2/), for the implementation of the message passing interface. All algorithms were implemented in ANSI-C programming language. Due to the stochastic nature of the algorithms compared in this work, 10 independent runs were done with different initial random seeds generated by the Mersenne Twister random number generator [16].

4.1 Protein Sequences

Table 1 shows the list of real protein sequences that were used in this work. These proteins were extracted from PDB files (available in *http://www.pdb.org*). In this table, the first column and second columns identify, respectively, the PDB code (with the size – N) and the equivalent AB sequence.

In order to convert the protein sequences of the PDB into the AB model alphabet (i.e.: 'A' and 'B' for hydrophobic and hydrophilic residues, respectively) we need to use an amino acid conversion table. In this work, we used the amino acid type classification shown in [2].

Table 1: Equivalent AB sequences of the proteins

PDB code (Size)	Equivalent AB sequence
2gb1 ($N = 56$)	$AB^3A^3BAB^2ABAB^4$ $B(AAB)^2AB^2A^2(BBBA)^3A(BA)^2B(BBBA)^2$ BAB^2
1pcy ($N = 99$)	$A(BAAAAABBA)^2(BA)^2AB^2$ $A^3B^3A^4B^2A^3B^4(AAB)^2AB(BA)^2B^4A^2$
2trx ($N = 108$)	$B^3A(AB)^2(BBBA)^2(AB)^2$ $A^2(AAAB)^2A^5BA^6B^2A^2B^4(AB)^2A^2(BA)^2$
3fxn ($N = 138$)	$ABA^2(BA)^3B^4A^2(BAAA)^2$ $(BBA)^3(BABBA)^2AB^3A^6BA$

4.2 Contact Map Generation

The performance of the pECO approach was evaluated using Contact Maps (CMs) which, in turn, were generated from 3D protein structures obtained by Molecular Dynamics (MD) simulations, as proposed by [5]. Figure 4 presents the Contact Maps generation procedure. It is important to recall that it depends on the CM obtained by MD simulations.

The 3D structures were generated for the protein sequences presented in Section 4.1 by MD simulations with time-step: $\delta t = 0.0001$ and stop criterion: $t_{max} = 300$, leading to 3×10^6 folding states for each protein sequence.

From the structures obtained by MD simulations at equal intervals of time between 0 and t_{max}, 100 CMs were generated for the following threshold values: 6.65, 7, 8, 9, 10, 11 and 12Å. The first value was obtained from the dimensionless value defined by [11]. They stated that two monomers i and j are taken to be in contact if $r_{ij}^2 < 1.75$. Considering that the unity dimensionless distance is 3.8Å, 1.75 is equal to 6.65Å. The other ones are typical threshold values considered in the literature. A total of 700 CMs were generated for each protein sequence and, thus, 2800 CMs for the four sequences. For instance, for the protein 2gb1, composed by 56 amino acids, each CM is a 56x56 matrix and represents a folding state of the folding process.

5. STRUCTURE VALIDATION

In this work, we assess the quality of the structures obtained by comparing them with the structures obtained by Molecular Dynamics simulations (see [5]). Basically, the procedure has three steps, where the first two steps are fitting procedures and the last one represents the quality assessment. In the two first steps, the structures obtained are fitted to off-lattice structures (the so-called "AB_like"), where all unit-length bonds are scaled to 3.8 Å, which is the mean distance between two consecutive Cα atoms [15]. Next, the similarity between the off-lattice structures obtained in steps 2 and 3 is measured using RMSD [15].

The RMSD evaluation depends on the superpositioning of the protein structures. Since the RMSD is a rotation-dependent measure, an optimised RMSD is done using the Kabsch algorithm [12] in order to obtain the lowest RMSD.

5.1 pECO – Control Parameters

The parameters used for the pECO algorithm are: number of populations (*Q*) that will be co-evolved, the initial population size (*POP*), number of cycles for ecological successions (*ECO-STEP*), the size of the evolutive period (*EVO-STEP*) that represents the number of function evaluations in each *ECO-STEP*, the minimum threshold distance (ρ), and the tournament size (*T-SIZE*) used to choose solutions to perform intra and inter-habitat communications. The values for these parameters were defined empirically with: Q = 40, POP = 50, *ECO-STEP = 2,000*, *EVO-STEP = 100* , $\rho = 0.5$ and *T-SIZE = 5*. The heterogeneous model of the pECO approach, combines all four algorithms (ABC-PSO-DE-jDE/BBO) in which $1/4$ of the populations behaves according to one of these strategies. In this work, the number of processors (m) is equal to the number of populations ($m = Q$).

5.2 Parameters of the Algorithms

Default parameters recommended in the literature were used in the algorithms employed. *POP* is a common parameter between all algorithms and is adjusted as mentioned in Section 5.1. For ABC algorithm, there is only one control parameter, *limit = 100* [13]. For PSO algorithm, besides *POP*, the parameters were set to standard values (Standard PSO (SPSO-07): http://www.particleswarm.info/-Programs.html) : inertia weight $W = 0.721$; cognitive and social components $\varphi_p = \varphi_g = 1.193$, respectively. For DE algorithm, the parameters are $F = 0.9$ (F controls the amplification of the differential variation) and $CR = 1.0$ (crossover constant) with DE/rand/1/bin approach. And for jDE/BBO the parameters used are $I = E = 1.0$ (maximum possible immigration and emigration rates), $CR = 0.9$, $F = 0.5$, and S_{max} =*POP* [10].

6. RESULTS AND ANALYSIS

6.1 Numerical Results

Table 2 presents the results obtained through pECO simulations, using CMs of the 2gb1 ($N = 56$) sequence. The first column shows the metrics of the best individuals. Next columns show their values for each threshold value. It is important to recall that the S_C, S_{NC} metrics and the RMSD are computed using the input CM and the CM of the obtained conformation. Overall, from the S_C and RMSD values, better results are obtained for larger threshold values, since the S_C increases and the RMSD decreases when increasing the threshold.

In addition, experiments were done using CMs of the 1pcy ($N = 99$), 2trx ($N = 108$) and 3fxn ($N = 138$) sequences obtained by MD simulations (with threshold of 7Å). The obtained results are shown in Table 3. As expected, from the S_C, RMSD and processing time values, the performance of the approach decreases with the protein length since the search space complexity grows exponentially.

6.2 Graphical Results

Figures 5(a), 5(b) and 5(c) show the best values of the metrics S_C, S_{NC} and RMSD obtained for each CM of the 2gb1 sequence

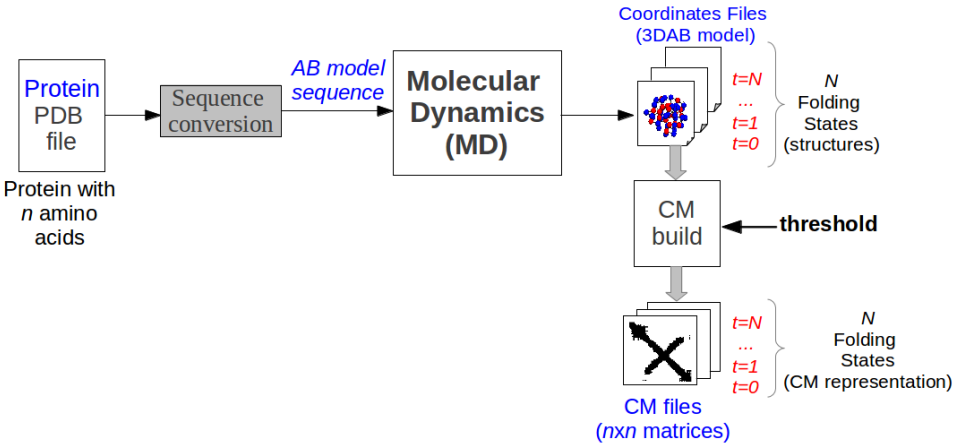


Figure 4: Contact Maps generation procedure

Table 2: Numerical results obtained using CMs with different threshold values – sequence 2gb1

Metric	CM threshold [Å]			
$Avg(Min/Max)$	6.65	7	8	9
Best $fitness$	-88.68 (-120.59/-82.27)	-84.47 (-114.80/-77.56)	-83.51 (-108.90/-77.52)	-83.87 (-106.16/-76.22)
T_C	268.68 (138/296)	506.22 (296/562)	781.06 (396/856)	999.00 (456/1108)
F_C	112.94 (68/242)	172 (102/292)	194.94 (116/352)	191.70 (124/346)
T_{NC}	2600.8 (2562/2742)	2206.3 (2120/2554)	1813.88 (1694/2378)	1561.38 (1436/2320)
F_{NC}	153.58 (14/180)	251.48 (10/310)	346.62 (10/422)	383.92 (14/482)
S_C	0.64 (0.59/0.907)	0.67 (0.60/0.967)	0.697 (0.64/0.975)	0.73 (0.66/0.976)
S_{NC}	0.96 (0.91/0.975)	0.93 (0.89/0.96)	0.90 (0.86/0.94)	0.89 (0.84/0.94)
$Energy$	-144.74 (-152.86/-128.79)	-135.52 (-146.08/-124.15)	-132.78 (-143.76/-123.53)	-129.65 (-137.65/-115.72)
Kabsch RMSD [Å]	7.24 (5.22/11.96)	6.15 (3.31/11.91)	5.82 (3.41/11.04)	5.508 (3.12/7.43)
Avg t_p(s)	304	295.49	330.996	330.66

Metric	CM threshold [Å]		
$Avg(Min/Max)$	10	11	12
Best $fitness$	-82.33 (-104.48/-74.09)	-80.59 (-103.29/-70.60)	-79.93 (-102.86/-64.15)
T_C	1210.34 (554/1310)	1506.16 (670/1634)	1734.00 (776/1918)
F_C	185.88 (106/306)	151.06 (82/328)	115.62 (66/380)
T_{NC}	1325.48 (1172/2260)	1010.18 (860/2130)	791.66 (664/1980)
F_{NC}	414.30 (16/552)	468.60 (8/622)	494.72 (6/712)
S_C	0.75 (0.67/0.97)	0.76 (0.70/0.99)	0.78 (0.69/0.99)
S_{NC}	0.88 (0.80/0.94)	0.87 (0.80/0.93)	0.87 (0.80/0.92)
$Energy$	-125.46 (-137.60/-116.01)	-120.98 (-130.76/-109.92)	-117.13 (-128.19/-105.53)
Kabsch RMSD [Å]	5.61 (3.41/7.31)	5.55 (3.64/7.27)	5.75 (4.14/7.18)
Avg t_p(s)	324.70	340.39	294.70

Table 3: Numerical results obtained using CMs with threshold = 7Å– sequences 1pcy, 2trx and 3fxn

Metric	Sequence		
$Avg(Min/Max)$	1pcy	2trx	3fxn
Best $fitness$	-114.72 $\pm$ 19.26 (-221.59/-90.79)	-99.72 $\pm$ 22.34 (-240.53/-67.75)	-81.55 $\pm$ 20.64 (-207.33/-56.72)
T_C	765.78 $\pm$ 43.95 (450/840)	741.64 $\pm$ 41.41 (556/828)	824.4 $\pm$ 52.82 (592/948)
F_C	377.84 $\pm$ 69.68 (230/692)	351.50 $\pm$ 100.35 (138/690)	248.5 $\pm$ 84.59 (94/752)
T_{NC}	7919.02 $\pm$ 155.93 (7709/8629)	9651.2 $\pm$ 178.71 (9382/10374)	16628.88 $\pm$ 231.40 (16346/17588)
F_{NC}	738.36 $\pm$ 144.13 (30/852)	919.66 $\pm$ 170.84 (44/1094)	1342.22 $\pm$ 243.04 (148/1564)
S_C	0.52 $\pm$ 0.08 (0.46/0.94)	0.45 $\pm$ 0.08 (0.38/0.93)	0.39 $\pm$ 0.07 (0.32/0.8)
S_{NC}	0.95 $\pm$ 0.008 (0.93/0.97)	0.96 $\pm$ 0.01 (0.94/0.98)	0.98 $\pm$ 0.005 (0.96/0.99)
$Energy$	-232.13 $\pm$ 12.47 (-259.61/-195.51)	-226.66 $\pm$ 21.44 (-276.83/-179.11)	-211.72 $\pm$ 20.79 (-270.26/-168.96)
Kabsch RMSD [Å]	10.22 $\pm$ 2.19 (6.92/23.51)	12.56 $\pm$ 2.96 (8.39/27.44)	20.16 $\pm$ 4.27 (12.14/32.39)
Avg t_p(s)	496.01	551.08	771.41

with different threshold values (for the sake of simplification, in these plots, only the results for three different threshold values are shown). Figure 5(a) shows that higher values of S_C are obtained using CMs with higher threshold values. Basically, it indicates that the approach obtained better structures using CMs with more *contacts*. On the other hand, Figure 5(b) shows that better values of S_{NC} are obtained for CMs with lower threshold. In Figure 5(c), lower RMSD values were obtained for higher threshold values. This indicates that the *contacts* of the structures obtained are more important than the *non-contacts* in the process of structure reconstruction. Overall, better results are obtained for CMs with higher threshold values.

Figures 6(a) and 6(b) show an example of the convergence plot for the CMs of the sequence 2gb1 ($N = 56$) and 3fxn ($N = 138$), respectively. In these figures each label indicates the identification of the species and its algorithm. The x-$axis$ shows the number of Ecological successions and the y-$axis$ represents the best-ever fitness value. Analyzing these plots it is observed that for the 56 amino-acids-long sequence the convergence is not accentuated in the direction of a stagnation point during the ecological successions. Thus, best solutions may be achieved increasing the number of ecological successions. For the 138 amino-acids-long sequences the convergence seems to be slow and attracted to a local minimum basin. It is observable that small improvements are achieved from half of the ecological successions forwards. These convergence plots indicate that, in order to improve the results, strategies for maintaining diversity inside populations are required as well as a method to detect and escape from the attraction basin regions of local minima. Also, these figures show some labels indicating which algorithm achieved the best solution at each ecological succession. Once a different algorithm updates the best solution, a new label is added. For example, for the 56 amino acids-long sequence a population with the PSO algorithm achieved the best solution until around succession 20, where the 17^{th} species found the best solution. From successions 21 to around 49 a population with the jDE/BBO algorithm (the 6^{th} species) achieved the best solution; from successions 319 to 345 a population with the ABC algorithm (the 14^{th} species) achieved the best solution, and from successions 370 to 2,000 different populations with the jDE/BBO algorithm achieved the best solution. Analysing these labels, it is possible to notice the coevolution between the different search strategies (ABC/PSO/DE/jDE-BBO) because they alternate in finding the best solutions. Possibly this is due to the peculiarity of each method in searching the space of solutions.

Figures 6 (c) shows an example of the evolution of the number of habitats for each ecological succession step for the CMs of the sequence 2gb1 ($N = 56$). It is observed that, at the beginning of the optimization process, with the populations widely dispersed in the search space, there is a large number of habitats. As the optimization process moves through the ecological successions, the populations tend to move through the search space converging to specific regions. As shown in this figure, the number of habitats decreases with the ecological succession cycles, indicating that the populations tend to converge to points close to each other. Overall, due to the high complexity of the problem, the populations are dispersed through the search space during all successions. This indicates that more ecological successions the pECO approach could lead to even better results.

A brief analysis of the load balancing of the pECO was done, based on the performance measures *speedup*, *efficiency* and *serial fraction* [1], that are a direct consequence of the balance between the processing load and the communication load between master and slaves processors. The "Versus panmixia" approach is used to evaluate the parallel implementation, using the sequential version of the algorithm as a reference. A sublinear speedup ($s_m < m$, where $m = Q = 40$) behavior could be clearly identified. Recall that a speedup higher than one suggests that the parallelization of the algorithm decreases the overall computational cost. Ideally, the speedup should be linear, but this is not possible in practice, since processors are not used only for processing, but also for other tasks such as for *message-passing* communication between them. It is also possible to observe that the speedup increases with the protein size. For instance, the speedups achieved were 11.98, 17.36, 23.84 and 27.87 for the 56, 99, 108 and 138 amino acid-long sequences. This is due to the relatively high time needed to transmit data between processes for small proteins, when compared with the processing load. Therefore, it is necessary to establish a load balance between the processing and communication loads between processes. Better speedups can be achieved for larger proteins.

The efficiency achieved were 0.3, 0.43, 0.59 and 0.69 for the 56, 99, 108 and 138 amino acid-long sequences. These values suggest that the processors are not fully used all the time. In fact, speedup and efficiency are a direct consequence of the balance between the processing load of the slaves and the communication load between master and slaves.

The serial fraction obtained are 0.06, 0.033, 0.017 and 0.011 for the 56, 99, 108 and 138 amino acid-long sequences. It indicates that the granularity of the parallel approach decreases when increasing the protein length. Thus, the approach is more efficient for larger protein sequences.

7. CONCLUSION

The reconstruction of protein structures from CMs is still an unsolved problem, which has been proved to be NP-hard. In this work, the performance of a parallel ecologically-inspired optimization algorithm (pECO) was analysed, under the task of reconstructing the structure of proteins from CMs, featuring the 3D-AB *off-lattice* model. Four population-based algorithms (ABC, PSO, DE, and jDE/BBO) were employed in an ecological heterogeneous model. It is possible to conclude that, in this problem case, these strategies are quite complementary, even during few successions. The convergence plots indicate that ABC and PSO algorithms are best suited for global search (initial ecological successions), whilst the DE and jDE/BBO algorithms are best suited for local search (final ecological successions). The results obtained suggest that a smooth convergence is achieved throughout the pECO simulations, avoiding the stagnation of the search. Overall, from the S_C and RMSD values, it is observed that better results (i.e. conformations with lower RMSD values) are obtained for CMs with larger threshold values, since the S_C increases and the RMSD decreases when increasing the threshold. Due to the high complexity of the problem, populations were dispersed in the search space during ecological successions. This indicates that even better results would be found by increasing the number of successions or through the diversification of evolutive behaviors of the computational ecosystem, by inserting other algorithms. For instance, local-search strategies could be used to improve the quality of the obtained solutions.

An important drawback is regarding the processing time for the simulations. It is clear that there is an increase of processing time as the length of the protein grows in all approaches presented in this work. This fact, by itself, strongly suggests that parallel processing is essential to allow us to obtain results in a reasonable processing time. Future research will address highly parallel approaches for dealing with the problem, such as the use of GPGPU (General Purpose Graphics Processing Units).

Another drawback is regarding the high number of user-defined parameters. It is important to recall that there is no specific procedure for adjusting running parameters of Evolutionary Algorithms for a given problem and that it represents one of the grand challenges of the Evolutionary Computation (EC) field. A strategy frequently used in the literature is setting a range for all important parameters of the algorithm and testing all possible combinations. Although self-adjustment of parameters tends to be more efficient than trial-and-error design and factorial experiments, this was not the focus of the present work. Although not optimal for any instances, the parameters used in the approaches presented in this work could be an initial reference to other researchers and they will be an important issue for future works.

In a broader sense, it is believed that the computational approach proposed in this work is promising for the research areas related to Evolutionary Computation and the Protein Folding Problem.

8. ACKNOWLEDGMENTS

Authors would like to thank the Brazilian National Research Council (CNPq) and the Fundação Araucária for the research grants to H.S. Lopes.

9. REFERENCES

[1] H. Alba. *Parallel Metaheuristics: A New Class of Algorithms*. Wiley-Interscience, New York, USA, 2005.

[2] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Molecular Biology of The Cell*. Garland Science, New York, USA, 2002.

[3] C. Anfinsen. Principles that govern the folding of protein chains. *Science*, 181(96):223–230, 1973.

[4] M. Begon, C. R. Townsend, and J. L. Harper. Ecology: from individuals to ecosystems. 2006.

[5] C. Benítez and H. Lopes. Ab-initio protein folding using molecular dynamics and a simplified off-lattice model. *Journal of Bionanoscience*, 7:391–402, 2013.

[6] C. Benítez, R. Stubs Parpinelli, and H. Lopes. A heterogeneous parallel ecologically-inspired approach applied to the 3D-AB off-lattice protein structure prediction problem. In *Computational Intelligence and 11th Brazilian Congress on Computational Intelligence (BRICS-CCI CBIC), 2013 BRICS Congress on*, number 1, pages 592–597, Piscataway, USA, 2013. IEEE.

[7] C. Benítez, W. Weinert, and H. Lopes. Gene expression programming for evolving two-dimensional cellular automata in a distributed environment. In D. Camacho, L. Braubach, S. Venticinque, and C. Badica, editors, *Intelligent Distributed Computing VIII*, volume 570 of *Studies in Computational Intelligence*, pages 107–117. Springer International Publishing, Heidelberg, 2015.

[8] R. Broglia and G. Tiana. Physical models for protein folding and drug design. In *Proceedings Idea-Finding Symposium*, pages 23–33, Frankfurt, Germany, 2003. Frankfurt Institute for Advanced Studies.

[9] J. Duarte, R. Sathyapriya, H. Stehr, I. Filippis, and M. Lappe. Optimal contact definition for reconstruction of contact maps. *BMC Bioinformatics*, 11(1):283, 2010.

[10] W. Gong, Z. Cai, and C. X. Ling. DE/BBO: a hybrid differential evolution with biogeography-based optimization for global numerical optimization. *Soft Computing*, 15(4):645–665, 2010.

[11] A. Irback, C. Peterson, and F. Potthast. Identification of amino acid sequences with good folding properties in an off-lattice model. *Physical Review E*, 55(1):860–867, 1997.

[12] W. Kabsch. A discussion of the solution of the best rotation to relate two sets of vectors. *Acta Crystallographica*, A34:827–828, 1978.

[13] D. Karaboga and B. Akay. A comparative study of artificial bee colony algorithm. *Applied Mathematics and Computation*, 214(1):108–132, 2009.

[14] R. MacCallum. Striped sheets and protein contact prediction. *Bioinformatics*, 20(1):I224–I231, 2004.

[15] M. Mann, R. Saunders, C. Smith, R. Backofen, and C. Deane. Producing high-accuracy lattice models from protein atomistic coordinates including side chains. *Advances in Bioinformatics*, 2012(1):148045, 2012.

[16] M. Matsumoto and T. Nishimura. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation*, 8(1):3–30, 1998.

[17] R. M. May et al. Theoretical ecology. principles and applications. *Theoretical ecology. Principles and applications.*, (Ed. 2), 1981.

[18] B. Niu, Y. Zhu, and X. He. Multi-population cooperative particle swarm optimization. In *Advances in Artificial Life*, volume 3630 of *Lecture Notes in Computer Science*, pages 874–883. Springer Berlin Heidelberg, 2005.

[19] R. S. Parpinelli and H. S. Lopes. A hierarchical clustering strategy to improve the biological plausibility of an ecology-based evolutionary algorithm. In *Advances in Artificial Intelligence - IBERAMIA 2012*, volume 7637 of *Lecture Notes in Computer Science*, pages 310–319. Springer Berlin Heidelberg, 2012.

[20] R. S. Parpinelli and H. S. Lopes. A computational ecosystem for optimization: review and perspectives for future research. *Memetic Computing*, 7(1):29–41, 2015.

[21] M. Punta and B. Rost. PROFcon: novel prediction of long-range contacts. *Bioinformatics*, 21(13):2960–2968, 2005.

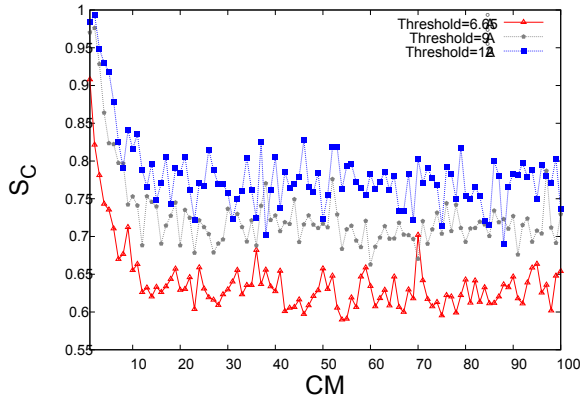
[22] D. Röthlisberger, O. Khersonsky, A. Wollacott, L. Jiang, J. DeChancie, J. Betker, J. Gallaher, E. Althoff, A. Zanghellini, O. Dym, S. Albeck, K. Houk, D. Tawfik, and D. Baker. Kemp elimination catalysts by computational enzyme design. *Nature*, 453(7192):190–195, 2008.

[23] F. Stillinger, T. Head-Gordon, and C. Hirshfeld. Toy model for protein folding. *Physical Review E*, 48(2):1469–1477, 1993.

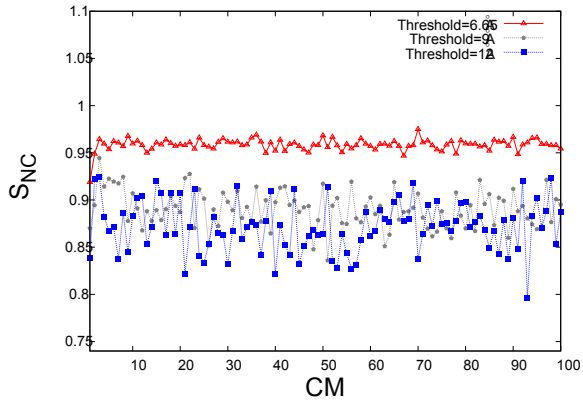
[24] C. Twomey, T. Stutzle, M. Dorigo, M. Manfrin, and M. Birattari. An analysis of communication policies for homogeneous multi-colony aco algorithms. *Information Sciences*, 180(12):2390–2404, 2010.

[25] M. Vassura, P. D. Lena, L. Margara, M. Mirto, G. Aloisio, P. Fariselli, and R. Casadio. Blurring contact maps of thousands of proteins: what we can learn by reconstructing 3D structure. *BioData Mining*, 4:1–15, 2011.

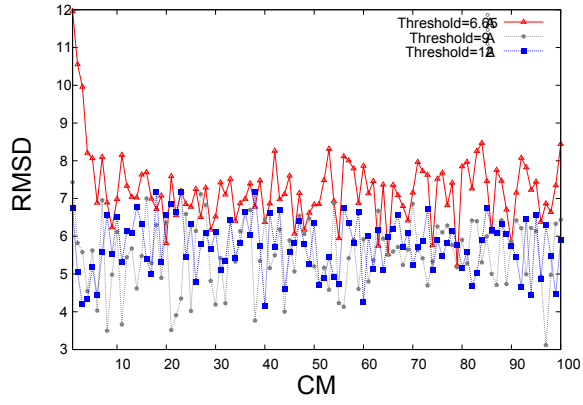
[26] D. Whitley, S. Rana, and R. B. Heckendorn. The island model genetic algorithm: On separability, population size and convergence. *Journal of Computing and Information Technology*, 7:33–48, 1999.



(a)

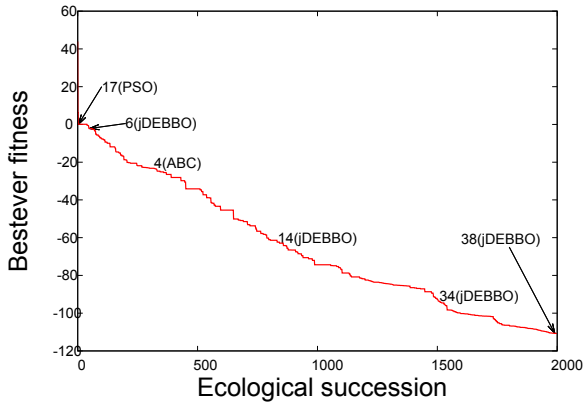


(b)

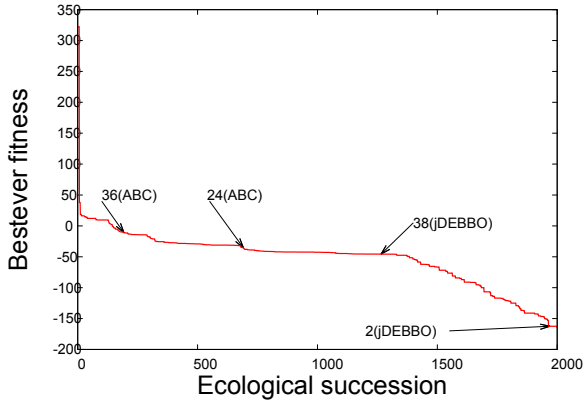


(c)

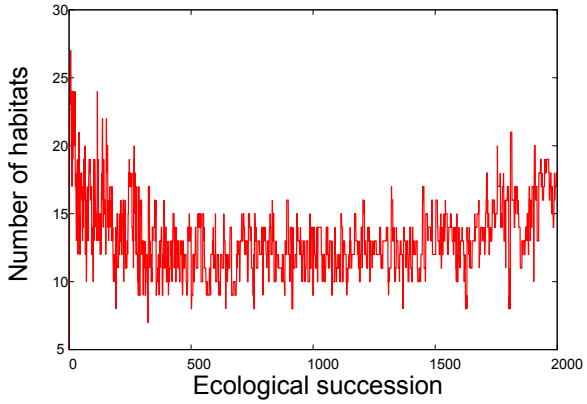
Figure 5: Best values of the metrics S_C (a), S_{NC} (b) and RMSD (c) for CMs with different threshold values



(a) Convergence for the 56 amino-acids-long sequence



(b) Convergence for the 138 amino-acids-long sequence



(c) Convergence for the 56 amino-acids-long sequence

Figure 6: Plots for the pECO convergence.