A Multi-objective Swarm-Based Algorithm for the Prediction of Protein Structures

Leonardo de Lima Corrêa and Márcio Dorn(✉)

Institute of Informatics, Federal University of Rio Grande do Sul, Porto Alegre, Rio Grande do Sul, Brazil
{llcorrea,mdorn}@inf.ufrgs.br

Abstract. The protein structure prediction is one of the most challenging problems in Structural Bioinformatics. In this paper, we present some variations of the artificial bee colony algorithm to deal with the problem's multimodality and high-dimensionality by introducing multi-objective optimization and knowledge from experimental proteins through the use of protein contact maps. Obtained results regarding measures of structural similarity indicate that our approaches surpassed their previous ones, showing the real need to adapt the method to tackle the problem's complexities.

Keywords: Swarm intelligence · Multi-objective optimization · PSP

1 Introduction

The protein structure prediction (PSP) remains as one of the most challenging problems in Bioinformatics. Proteins are in all living systems and are responsible for a massive set of functions, participating in almost all cellular processes. Knowing the protein structure allows one to study biological processes more thoroughly. The PSP is classified as NP-hard problem in accord with the computational complexity theory [19], due to the multi-modal search space and high dimensionality, presenting an exponential growth of difficulty as the protein's size increases. Problem complexity relies on protein conformations' explosion, where a long amino acid (aa) chain can give rise to few conformations around a native state among numerous existing possibilities.

An extensive range of computational methods has been presented for the PSP problem. The existing methods can be classified into two major categories in accordance with the target protein characteristics [7,15]: (i) template-based modeling (TBM); and (ii) free-modeling (FM). So the first one encompasses aa sequences that have detectable evolutionary similarities to the experimentally determined ones, making it possible to identify similar structural models and ease prediction process. Differently, FM represents aa sequences which do not exhibit similarities to the experimentally determined proteins. Difficulty relies on

J. M. F. Rodrigues et al. (Eds.): ICCS 2019, LNCS 11538, pp. 101–115, 2019.
https://doi.org/10.1007/978-3-030-22744-9_8

the target modeling through *ab initio* methods which may incorporate protein structural information from databases. Methods under this classification generally represent hybrid approaches that use *aa* fragments combined to a purely *ab initio* strategy. *Ab initio* methods are based only on thermodynamic concepts and physicochemical properties of the folding process of proteins in nature.

It is well known that the energy function inaccuracies and the multi-modal search space are enough factors in expanding efforts to develop new strategies to obtain not only better structural results but insights about intrinsic and hidden problem properties. Multi-objective (MO) strategies aim to deal with optimization problems from different perspectives. Generally, complex problems present objective functions with several terms, many of them conflicting with each other, which, in turn, makes it hard to simultaneously optimize them properly [13]. Also, such problems may have specific properties that are not often considered in optimization processes, for reasons of simplicity or even inability to integrate them into the evaluation function when single-objective optimization [9]. In this sense, we adapted the Mod-ABC algorithm [5] to deal with the PSP by introducing MO strategies [9,13], in order to minimize the existing conflicts between energy function terms and reach an acceptable balance among them, and evaluate the MO algorithms in the face of a quite difficult problem. These new algorithms incorporated another experimentally determined protein structures' knowledge strategy besides the ones already integrated into the Mod-ABC. Encouraged by the latest CASP results [18], we modeled the information of contact maps (CMs) [12,18] as a term added to the energy function. CMs are predicted from analysis of correlated evolutionary mutations achieved from multiple sequence alignments. In this work, it was used as constraints in the algorithm calculation to support the heuristic, deal with the search space roughness and reduce its size. An assessment of CMs contribution to the solution quality was carried out regarding single and multi-objective optimization. Our major contribution in this work is the development and assessment of the ABC algorithm adaptation to work with MO strategies and also handle the information of CMs as constraints in optimization to reach better prediction results.

2 Problem Background

The methods described in this work are variants of the Mod-ABC algorithm [5]. All of them adopt the same computational protein representation and the Angle Probability List technique. Methods accept as input parameters the protein primary structure, its expected secondary structure (SS) and the generated CMs.

A. Protein Representation: From a structural perspective, a peptide is formed by two or more amino acids joined by a chemical bond known as a peptide bond. Larger peptides are known as polypeptides or proteins. So the proteins are represented by linear *aa* sequences, responsible for determining their conformations. The protein folding gives the protein-specific properties, which dictate its role in the cell. The amino acids found in proteins present all the same main structure, the backbone, and differ in the side chain structure. In

an *aa* chain, the peptide bond, known as Omega angle (C-N, ω), has a partial double bond character which does not allow the free molecule rotation around it. Conversely, the free molecule rotation is allowed over the bonds known as Phi (N-C_α) and Psi (C_α-C) dihedral angles, ranging under a continuous domain from $-180°$ to $+180°$. Such free rotation is mostly responsible for the 3-D structure assumed by the protein, whereas the amino acids' stable local arrangements define the SS. As the polypeptide backbone, side chains present dihedral angles too, known as Chi angles (χ). Their conformations contribute to the stabilization and packing of the protein structure. The Chi angles number in an *aa* is concerned to its type, varying from 0 to 4, ranging under a continuous domain from $-180°$ to $+180°$. Thereby, the protein's set of dihedral angles form its 3-D structure. In this paper, the protein structure was computationally represented by its dihedral angles as a way to reduce the use complexity of all-atom representation of the protein.

B. Objective Function: To assess the quality of a modeled protein structure, we adopted as fitness function the Rosetta energy function (all-atom high-resolution and minimization function) [17] provided by the PyRosetta toolkit https://www.rosettacommons.org. The Rosetta energy function considers more than 18 energy terms, most of them derived from knowledge-based potentials [17]. The function has terms based on Newtonian physics, inter-atomic electrostatic interactions and hydrogen bonding energies dependent on the orientation. According to the CASP experiments, Rosetta methods have reached one of the best results in the competition [15]. The final energy value of the Rosetta function ($E_{rosetta}$) is given by the sum of all weighted terms considered in the calculation. The terms' weights are defined based on the energy function *Talaris2014*, that is the standard Rosetta function used to assess *all-atom* protein structures. Additionally to the Rosetta terms, the solvent accessible surface area from the PyRosetta was included as a term ($SASA_{term}$) into the final energy function [5] with an atomic radius of 1.4 Å, to assist the 3-D structures packing given the difficulties presented by *Talaris2014* in such task. Also, to support the secondary structures formation, the SS term (Eq. 1) was added to the fitness function. The procedure gives: (i) a positive reinforcement to the energy function, adding a negative constant (-1000) to the sum of amino acids of the protein structure P, if the SS (zp_i) corresponding to the i-th amino acid (aa_i) is equal to the SS (zi_i) of the same *aa* informed as input to the method; or (ii) gives a negative reinforcement to the sum, adding a positive constant ($+1000$), when the SS of the corresponding amino acids are not the same. All protein amino acids are compared throughout the model evaluation. We used the DSSP method (https://swift.cmbi.umcn.nl/gv/dssp/) to assign the secondary structures. Finally, the terms previously described were integrated to the Rosetta function composing the evaluation function (E_{final}) (Eq. 3) used in this work.

$$SS_{term} = \sum_{aa \in P} V(aa_i, zp_i, zi_i) \tag{1}$$

$$V(aa, zp, zi) = \begin{cases} -const, & zp = zi \\ +const, & zp \neq zi \end{cases} \tag{2}$$

$$E_{final} = E_{rosetta} + SASA_{term} + SS_{term} \tag{3}$$

C. Amino Acids Conformational Preferences: The methods in this paper use the knowledge of experimental protein structures in the Protein Data Bank (PDB) (https://www.rcsb.org). The main benefit of using this information is to reduce the search space size and increase the effectiveness of the method. In the Mod-ABC [5], the authors incorporated the structural information of known protein templates to determine the conformational preferences of a target protein using the Angle Probability List (APL) strategy [4]. Such technique assigns the dihedral angles to the target amino acids by the conformational preferences analysis of such amino acids in experimental structures, regarding the secondary structures and the neighboring amino acids. To use it, according to the authors, they built histograms of $[-180°, 180°] \times [-180°, 180°]$ cells for each amino acid and SS, generating combinations up to 9 amino acids (1–9) and their secondary structures, and taking into account the reference *aa*'s neighborhood for combinations larger than 1. We note that the angle values are attributed only to the reference *aa*. Each histogram cell (i,j) has the number of times that a given *aa* (or combination of amino acids) presents a torsion angles pair ($i \leq \phi < i+1$, $j \leq \psi < j+1$) concerning a SS. The angle probability list was calculated for each histogram, representing the normalized frequency of each cell. APL was incorporated in the methods to create short combinations of amino acids aiming the use of high-quality individuals as a starting point and after a restarting function. A weighted random selection was employed to select the angle values from APL. It gives greater chances to the histograms' cells that present a higher relative frequency of occurrence. Furthermore, for a full APL description, we point out our web server NIAS-Server [4] created to investigate the amino acids conformational preferences.

D. Protein Contact Maps: The prediction of protein contact maps is based on the knowledge discovery from experimental protein structures data and tries to probabilistically determine which residues are in contact. There are several proposed contact map predictors in the literature [18]. Most of them explore strategies of machine learning, such as deep learning networks and support vector machines with classical biological features, like SS, solvent accessibility and sequence profile [2]. Ultimately, the incorporation of contact predictions from coevolution-based methods as additional features also significantly improved their performance [2,18]. In the last years, contact predictions were shown to be a valuable addition to the PSP methods [18]. As reported, improved contact methods can lead to improved FM model accuracy [1]. However, despite the improvement in the residue-residue contact prediction, its use in an efficient way into the PSP algorithms configures the major challenge [18]. Various factors determine the methods' performance, such as the number of contacts considered and how they are incorporated into the modeling. Hence, the most

suitable contact prediction technique and the number of contacts to consider are dependent on the PSP algorithm. As pointed out by the last CASP report [18], the use of size lists of $L/2$ contacts can improve the performance, reducing the false positives and taking into account the predicted residue contacts with higher probabilities of being in contact. L represents the target aa sequence length. By the prediction results carried out in the experiments, list sizes of $L/2$ seemed to be one of the best choices. In this paper, we used a reduced list of $L/2$ predicted contacts. The CMs were predicted by the MetaPSICOV predictor [10].

In a CM, two amino acids are close enough or in contact, if the distance between their Cβ side chain atoms, or Cα of backbone for Glycine, is less than or equal to a distance threshold, generally 8 Å. A term of distance constraint is generally used to get the information from CMs and to overcome some inaccuracies of the energy function [12]. In this paper, besides the terms of the fitness function described in Eq. 3, we proposed a scheme to employ the information of CMs in the problem as a new term in the energy function. This term was idealized based on an atom distance constraint function presented in the work of Kim et al. [12]. It was modified to follow the same idea of weighting used in the SS term (Eq. 1). The CM term is a function of the distances between the aa contained into the CMs, and it aims to positively reinforce the aa pairs that are within the contact bounds or to penalize the ones that are out of the threshold, according to Eq. 4.

$$CM_{term} = \sum_{i,j}^{CM_{pairs\,L/2}} = \begin{cases} p \times -c, & d(i,j) \leq ub \\ p \times -c \div 2, & ub < d(i,j) \leq ub + 2 \\ p \times +c, & d(i,j) > ub + 2 \end{cases} \tag{4}$$

where p denotes the probability of the residues are in contact, c is a constant, ub is a residue contact upper bound and $d(i,j)$ represents the Euclidean distance between a pair of amino acids in the predicted contact list. The MetaPSICOV considers the ub contact threshold of 8 Å, so in this paper, we adopted the same threshold of distance. For the constant c, we adopted $c = 1000$ to follow the reinforcement values defined in the SS term (Eq. 1). So for a target protein, the procedure goes through the $L/2$ aa pairs in the predicted CM, measuring the distances between these pairs regarding a given protein model. It gives (i) a positive reinforcement to the term summation, adding a negative constant $(-c)$ multiplied by the probability of the residues are being in contact, if the distance between them is less than or equal to the ub threshold; (ii) also a positive reinforcement to the term summation but considering the negative constant divided by 2 $(-c \div 2)$, if the distance between the amino acids is greater than the ub but does not exceed $ub + 2$ (tolerance threshold); and (iii) a negative reinforcement to the term summation, adding a positive constant $(+c)$ multiplied by the probability of the residues are being in contact, if the distance between the residues is greater than the threshold $ub + 2$.

E. Multi-objectivization: The multi-objectivization avoids conflicting terms compete to reach the best solutions and also favors the regarding of new properties about the problem, to aggregate information to the evaluation terms already considered in the optimization and better guide the search through new visions of the state space. In such approaches, the final optimization result encompasses a set of good solutions, called Pareto front (PF) [13]. PF represents a set of so-called non-dominated solutions. It comprises solutions where there is no possibility to improve one objective without disfavor another. Switching from one non-dominated solution to another will always result in a trade-off between objectives. The method's solutions can still be evaluated under different aspects, such as emerging features and unknown properties about the problem and input data.

In PSP, there are often conflicts between different terms of the energy function, as demonstrated by Cutello et al. [6]. The modeling of the energy function terms as independent objectives can provide a new exploration of the search space. Another interesting point is the possibility of inserting additional objectives containing information and constraints on the problem more naturally, avoiding the use of weighting coefficients, as it happens when inserted in traditional single objective approaches. Thus, the multi-objectivization of the prediction methods tends to ease the process of knowledge incorporation about the problem. An unconstrained MO optimization problem can be mathematically formulated as follows. Let $x = [x_1, x_2, ..., x_n]$ be a n-dimensional vector of decision variables, X be the search space (decision space) and Z be the objective space:

$$Minimize\ z = f(x) = [f_1(x), f_2(x), ..., f_m(x)], x \in X, z \in Z \tag{5}$$

where $m \geq 2$ is the number of objectives. Considering that during the optimization exists more than one single solution, the solutions are compared based on Pareto dominance, and the final answer is a set of non-dominate solutions (Pareto set). Let $M = [1, 2, ..., m]$ be the set of objectives, the Pareto set is defined according to the Eq. 6. A solution $x \in X$ dominates $y \in X$ ($x < y$) if and only if:

$$\forall i \in M : f_i(x) \leq f_i(y) \wedge \exists i \in M : f_i(x) < f_i(y), f_i(.) \in Z \tag{6}$$

To incorporate MO optimization in our algorithms and sort the solutions based on multiple objectives, we used the Pareto rank definition integrated into the evaluation function [16]. The Pareto rank of a solution measures the number of solutions that dominate it overall considered optimization objectives, regarding strict comparison ($<$), as shown in Eq. 6. So less the Pareto rank, less dominated is the solution. To order a set of solutions by Pareto rank as a minimization function, first the solutions are ordered from low to high Pareto rank and within this sorted order, those with the same Pareto rank are further ordered from low to high based on their energy values scored by an energy function.

3 Proposed Strategies

In this paper, we presented some ABC algorithm variations to tackle the PSP problem. We started from a previously proposed work, presented by Corrêa et al. [5], which has shown an ABC algorithm variation [11] implemented from suggested improvements in literature for the original ABC but never tested for the problem under study. It is called Mod-ABC and was designed to explore the specific properties of the problem. So the proposed algorithm variations were designed from the Mod-ABC based on an incremental development approach, in an attempt to improve the previously reached results. It was done by exploring additional features about the problem and adapting it to MO optimization to restrict the conformational space and overcome some energy function inaccuracies. In the following sections, Mod-ABC and the designed variations of it are presented.

A. Artificial Bee Colony Algorithm: ABC consists of a swarm intelligence based metaheuristic. It mimes the foraging process of honeybee swarms and is suitable for multi-numerical and multi-modal optimization [3,11]. Various works and ABC variations have been proposed indicating the algorithm competitiveness concerning other metaheuristics, such as genetic and differential evolution algorithms, particle swarm optimization and swarm-based algorithms [11]. It is said the key advantage of the heuristic is the use of a few control parameters [8]. In the ABC, the solution exploration and exploitation (refinement) are crucial optimization components. But the method has some inefficiencies, such as to perform well at the exploration but not so much at the solution refinement step [8]. This causes the heuristic's convergence slower and can be a problem on some occasions. To overcome it, improved ABC versions have been proposed in the literature. It was shown that these modified variations could be able to perform better than the original ABC [14]. Thus, the Mod-ABC assembles two proposed strategies for the algorithm. The first component, introduced in the work of Akay and Karaboga [3], concerns changes in the mechanisms that control the mutation frequency of variables of an individual and at the use of the most reasonable parameterization in the exploration ABC stage. The second one, presented by Zhu and Kwong [20], is related to the gbest-guided ABC (GABC). It uses the information regarding the best population's solution in the individual's mutation equation to improve the exploitation step. Authors of both methods pointed out that the ABC could be considered a promising metaheuristic regarding global and local optimization.

B. Mod-ABC Algorithm: In the ABC [3,11], each food source is a problem solution, and the solution quality is defined by the fitness value. Concerning the PSP, the food source means a possible solution for the protein under study and the quality of it is given by the energy value. The food sources are exploited by employed bees. Thus, the number of employed bees is the same number of food sources, i.e., the size of the population. The onlooker bees amount in the swarm is the same employed bees amount. Suppose that SN is the food sources amount (population's solutions), eb and ob the number of employed and onlooker bees,

respectively. So $SN = eb = ob$. The algorithm mimics the foraging behavior of honeybees regarding three steps: (*i*) in the employed bees' step (Algorithm 1, lines 3 to 10) each algorithm's solution represents a food source that is *updated* by a mutation procedure; (*ii*) in the onlooker bees' step (Algorithm 1, lines 18 to 27), *ob* individuals are randomly selected through the rank-based selection and the *update* procedure of the preceding stage is performed in the selected individuals; and (*iii*) in the scout bees' step (Algorithm 1, line 28) the most inactive population's individual is discarded and a new one is generated. An inactive individual is a solution that did not suffer improvements (fitness value) for a given number of generations. The update procedure (Algorithm 1, lines 5 and 21) used in the first two stages is responsible for generate a new individual from an existing one. So the generation of an individual $v_i = [v_{i1}, v_{i2}, ..., v_{in}]$ from the i-th individual $x_i = [x_{i1}, x_{i2}, ..., x_{in}]$, such that $x_i = v_i$, is described by (7).

$$v_{ij} = x_{ij} + \delta_{ij}(x_{ij} - x_{kj}) + \gamma_{ij}(y_j - x_{ij}), \tag{7}$$

where $i = [1, ..., SN]$, $j = [1, ..., n]$. SN represents the population size and n is the problem dimensionality. x_{ij} represents the j-th variable of individual x_i, v_{ij} is the new x_{ij} value, x_{kj} represents the j-th variable of the k-th population's individual ($k = [1, ..., SN]$) randomly chosen, and δ_{ij} means a random value in the continuous range $[-1, 1]$. The last term of 7 considers the population's best solution in the mutation operation. y_j denotes the j-th variable of the best individual and γ_{ij} represents a random value in the continuous range $[0, 1.5]$. Thus, the term presented by Zhu and Kwong [20] tries to guide the individual towards the population's best solution, increasing the algorithm convergence. Each variable j of the individual x_i is mutated regarding the control parameter MR (Algorithm 1, lines 4 and 20). Mod-ABC was set with $MR = 0.4$, according to the work of Akay and Karaboga [3]. So the update of a variable is done under the probability of 40%. The updating procedure concludes with a greedy selection between v_i and x_i (Algorithm 1, lines 8 and 24). Following the representation adopted in the paper (Sect. 2-A), each variable is an *aa* of the protein which has up to seven angles. Thus, the dihedral angles of the same variable are mutated in the same manner. To adjust the algorithm to the specific problem's characteristics, the Mod-ABC incorporates the function of *angle verification* (Algorithm 1, lines 6 and 22) into the updating procedure concerning the new generated values. The function verifies, at each angle mutation of the variable v_{ij}, if the newly generated value is in APL-1. It defines the *aa* conformational preferences regarding the variable v_{ij} and is used to avoid unfavorable state space regions or out of interval $[-180, 180]$. If the procedure verifies that the new value is not in the APL-1 or is out of the allowed interval, this value is discarded and the previous value is maintained. Lastly, in the scout bees' stage, if some population's individual did not suffer improvements over l generations, it is discarded and a new solution is included in the population (Algorithm 1, line 28). Suppose that l is the discarding threshold. We have used $l = 200$ according to Akay and Karaboga [3] and $SN = 300$ as population size, according to Corrêa et al. [5].

Irregular regions of proteins, such as coils and turns, are the hardest ones to predict because of the solvent exposure, configuring then structures with high flexibility level and low stability. Regarding the Mod-ABC, the algorithm focuses its search effort solely in such protein regions, excluding the more stable secondary structures, as β-sheets and α-helices, from the refinement process. Thus, the *updating* function (Algorithm 1, lines 5 and 21) is performed just in variables concerned the amino acids which present irregular secondary structures. To enhance the exploration aspect of the algorithm and increase the solutions diversity, as the updating of variables (Algorithm 1, lines 5 and 21) is constrained to the protein irregular secondary structures, the algorithm incorporates a crossover operation between two solutions of the population (Algorithm 1, line 14). The crossover was included between the first two Mod-ABC stages. The parents are selected through the rank-based strategy of selection (Algorithm 1, lines 12 and 13) and the operation is performed over the SS uniform crossover. The crossover concludes with a greedy selection between the generated solution and its parents (Algorithm 1, line 16). It is noteworthy that the Mod-ABC was implemented to assess in which way the knowledge-based strategies contribute to the algorithm performance facing a complex problem. The authors have shown by the obtained results that the method was able to outperform the ABC algorithm, corroborating the necessity of adapting the method to tackle the problem.

SS Uniform Crossover: From the proteins' structural preferences, it was created to support the secondary structures formation. The operator gives priority to the solutions that formed the appropriate arrangement concerning the SS input parameter. The crossover aims to maintain the similarity found so far between the solutions' secondary structures that are being optimized and the previously informed SS to create offspring with suitable secondary arrangements. Analogous to the uniform crossover, for each *aa* (specific positions of the angles in the vector solution), all the angles related to it are considered either from parent 1 or 2. The probability of 0.5 is used if both the secondary structures regarding the individuals' amino acids are equal or different from the previously informed SS. If only one of them is equal to the SS sequence parameter, the dihedral angles related to this amino acid are attributed to the offspring.

C. First Variation of the Mod-ABC Algorithm: The first variation of the Mod-ABC encompass modification just in the energy function used to assess the quality of a given protein structure. This version is called Mod-ABC-CM and incorporates the CM term (Eq. 4), already described in Sect. 2-D, into the final evaluation function. The CM term was designed to consider the information of protein contact maps in the PSP. The term was idealized in a way that penalizes violation of a predefined contact threshold regarding the distance of *aa* pairs in the CMs. In this sense, the CM term is added to the summation of all the terms already considered in the energy function (Rosetta energy function, SASA term, and SS term) (Sect. 2-B), forming then the final scoring function (Eq. 8) for the Mod-ABC-CM.

D. MO Versions of the Mod-ABC Algorithm

MO-ABC-1 Algorithm: The first MO version adapted from the Mod-ABC algorithm, called as MO-ABC-1, considers two objectives in the optimization process (bi-objective optimization). As first objective, the algorithm uses the final evaluation function (E_{final}) (Eq. 3) defined in Sect. 2-B. This scoring function is the summation result of three different terms, that is, Rosetta energy, SASA, and SS term. It is the fitness function used in the Mod-ABC algorithm. The second objective used in the MO-ABC-1 is the CM term (Eq. 4).

$$E_{finalCM} = E_{rosetta} + SASA_{term} + SS_{term} + CM_{term} \tag{8}$$

Algorithm 1. MO-ABC-1 algorithm's pseudocode.

Require: number of energy evaluations, primary and secondary *aa* sequence
Ensure: best individual found
initialize population using APL
while stop criteria not satisfied **do**
  for each *individual* in *population* **do** *//Employed bees's step*
    if $rand(0,1) \leq MR$ **then**
      update *individual* by (7)
      apply the **angle verification function**
      calculate the Pareto rank of *individual*
      apply a **greedy selection** between the new and old *individual*
    end if
  end for
  Sort *population* by Pareto rank and energy value (tiebreaker criterion)
  $bee_1 \leftarrow$ **select** an *individual* through rank-based selection *//Crossover step*
  $bee_2 \leftarrow$ **select** an *individual* through rank-based selection
  $bee_{offspring} \leftarrow$ **SSUniformCrossover**(bee_1, bee_2)
  calculate the Pareto rank of $bee_{offspring}$
  apply a **greedy selection** between $bee_{offspring}$ and its parents
  Sort *population* by Pareto rank and energy value (tiebreaker criterion)
  for $i \leftarrow 1 : ob$ **do** *//Onlooker bees's step*
    select an *individual* through rank-based selection
    if $rand(0,1) \leq MR$ **then**
      update *individual* by (7)
      apply the **angle verification function**
      calculate the Pareto rank of *individual*
      apply a **greedy selection** between the new and old *individual*
      Sort *population* by Pareto rank and energy value (tiebreaker criterion)
    end if
  end for
  Discard the most inactive individual *//Scout bees's step*
  Sort *population* by Pareto rank and energy value (tiebreaker criterion)
end while

One of the main reasons to consider the scoring function E_{final} as a unique objective besides the CM term is that SASA and SS terms tend to stabilize

during the optimization, as the population reaches some degree of convergence. Final solutions at the end of the process tend to present similar values for these terms, as can be seen in Table 1, regarding the average and standard deviation values for eight executions of the Mod-ABC algorithm for each listed target protein [5]. So it indicates that both of the terms are more necessary at the beginning of the optimization when the population is quite diversified. Both terms improve the search space exploration providing well-formed SS and more packing protein models. On the other hand, CMs were treated as a different objective as the contacts consider punctual atom distances in a more locally point of view, based on experimental protein knowledge, which can guide the search during the entire process making finer adjustments even when the algorithm reach some diversity degree. Another reason to categorize the objectives in this fashion was to assess the potential of the MO-Mod-ABC face a complex problem but including known and promising scoring potential. It is not so obvious how to organize terms of an energy function or include new ones into MO optimization for the PSP. However, it is indicated to keep the number of objectives small [16].

Algorithm 1 shows the MO-ABC-1 algorithm's pseudocode. The main difference of the MO-ABC-1 concerning its previous versions consists of the use of the Pareto rank strategy to compare and sort solutions during the optimization. The Pareto rank strategy, as well as how it is applied to sort the population's solution was already described in Sect. 2-E. The energy function employed as tiebreaker criterion when solutions present the same Pareto rank value was the final scoring function ($E_{finalCM}$) (Eq. 8) used in the Mod-ABC-CM.

MO-ABC-2 Algorithm: The MO-ABC-2 is the second MO version idealized from the Mod-ABC algorithm. It is basically the same MO-ABC-1 algorithm. However, it considers four objectives in the optimization process. The algorithm models each term of the final evaluation function (E_{final}) (Eq. 3), defined in Sect. 2-B, as different objectives. Thus, the first objective is the Rosetta energy function, the second is the SASA term, and the third is the SS term. The MO-ABC-2 also considers the CM term as a fourth objective during the optimization process. The energy function employed as tiebreaker criterion is the same used in the MO-ABC-1.

4 Computational Experiments

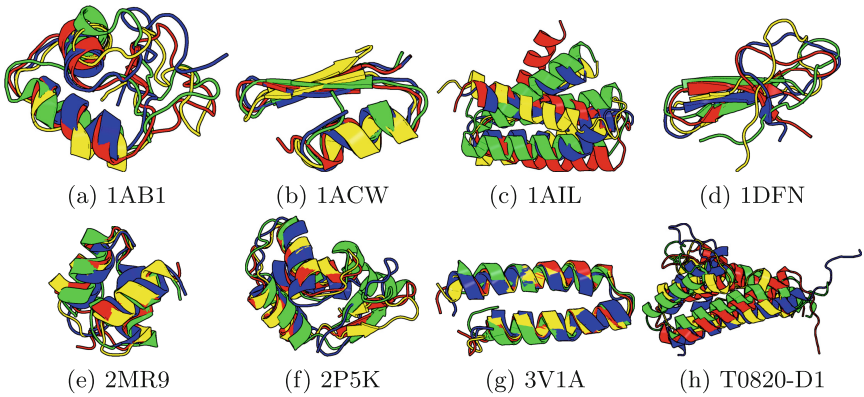
The described algorithms in this paper were run 8 times with a stop criterion of 10^6 calculations of energy per run on each target protein. We have used as case studies in our tests the *aa* sequences of 8 target proteins (Table 1) obtained from the PDB. To classify our algorithms concerning the most significant methods in the area, we have compared them to the Rosetta *ab initio* protocol [17]. Following the last CASP reports, Rosetta is one of the most relevant algorithms used to tackle the PSP problem [1,15]. Obtained results are presented in the next section.

Results and Discussion: For each case study, we have analyzed the best solutions among the performed executions, regarding the root-mean-square devia-

tion (RMSD, minimization measure) and the global distance total score test (GDT_TS, maximization measure) of the predicted structures in comparison with their corresponding experimental ones. Table 2 summarizes the obtained results of the Mod-ABC, Mod-ABC-CM, both MO Mod-ABC variations, and method of Rosetta applied to the target proteins.

Table 1. Target *aa* sequences. Average and standard deviation values for SASA and SS terms considering the best solutions of eight runs of the Mod-ABC algorithm [5] for each target protein.

Protein	Length	SS Content	SASA term		SS term	
			Avg.	Std.	Avg.	Std.
`1AB1` (Fig. 1a)	46	1 β-sheet/2 α-helices	3022.81	184.91	−42500.0	1936.49
`1ACW` (Fig. 1b)	29	1 β-sheet/1 α-helix	2168.96	79.74	−27000.0	0.0
`1AIL` (Fig. 1c)	70	3 α-helices	4512.33	116.53	−70000.0	0.0
`1DFN` (Fig. 1d)	30	1 β-sheet	2610.51	100.2	−24500.0	1322.88
`2MR9` (Fig. 1e)	44	3 α-helices	2698.88	93.02	−44000.0	0.0
`2P5K` (Fig. 1f)	64	1 β-sheet/3 α-helices	4581.71	282.31	−63000.0	0.0
`3V1A` (Fig. 1g)	48	2 α-helices	3329.42	97.05	−48000.0	0.0
`T0820-D1` (Fig. 1h)	90	3 α-helices	6304.73	291.76	−89750.0	661.44



(a) 1AB1 (b) 1ACW (c) 1AIL (d) 1DFN

(e) 2MR9 (f) 2P5K (g) 3V1A (h) T0820-D1

Fig. 1. Graphic representation of the experimental (red) and the predicted structures (lowest RMSD) for the Mod-ABC (green), MO-ABC-1 (blue) and Rosetta (yellow) (Color figure online).

According to the results summarized in the Table 2, we observe that the Mod-ABC-CM outperformed its previous version in almost all cases regarding lowest and average RMSD values, except for the 1ACW and T0820-D1. Similar results are noticeable analyzing the average and highest GDT_TS values,

where Mod-ABC-CM performed better than Mod-ABC in 5 of the eight targets. We strongly believe that Mod-ABC-CM surpassed Mod-ABC due to the use of experimental protein knowledge through the protein CMs incorporated into the fitness function. It reduced the size and complexity of the conformational space and eased the search process. These results reinforce the need to incorporate previous knowledge about the problem in the metaheuristics.

Regarding Table 2, we observe that the MO-ABC-1 reached better average RMSD values in 5 targets in comparison with the Mod-ABC-CM, and in 4 cases regarding lowest RMSD values. Related to the GDT_TS values, MO-ABC-1 outperformed Mod-ABC-CM in 6 targets for average results and 4 cases for highest ones. We should note the MO algorithm did not show great improvement when compared to its previous version. However, these results indicate that the MO strategy has great potential to be improved. It is observable that in this work we did not explore more sophisticated strategies to improve the multi-objectivation, and even though the algorithm was able to perform better in some cases. One of the reasons for that is the MO strategies capability to keep a set of non-dominated solutions over the PF. This sort of idea can increase the solutions' diversity by exploring different perspectives of the problem. It is observable that MO-ABC-1 in average presented better results than MO-ABC-2, corroborating that the arrangement of objectives also influences the search process.

Table 2. Methods simulation results. The **boldface** numbers represent the best results concerning RMSD and GDT_TS. The (*) denotes the best results between only Mod-ABC and its variations.

Method	RMSD (Å)							
	Lowest	**Avg. (std.)**	**Lowest**	**Avg. (std.)**	**Lowest**	**Avg. (std.)**	**Lowest**	**Avg. (std.)**
	1AB1		**1ACW**		**1AIL**		**1DFN**	
Mod-ABC	4.96	6.15 ± (1.43)	1.65	2.43* ± (0.69)	6.85	7.67 ± (0.55)	4.35	5.31 ± (0.55)
Mod-ABC-CM	4.31	**5.0*** ± (0.61)	1.85	3.01 ± (0.8)	3.9	6.77 ± (1.46)	3.55	4.5 ± (0.71)
MO-ABC-1	3.83*	5.09 ± (0.7)	**1.54***	2.8 ± (0.74)	**3.8***	**5.24*** ± (1.57)	**3.05***	**3.91*** ± (0.62)
MO-ABC-2	4.0	5.06 ± (0.67)	2.05	2.89 ± (0.82)	3.82	5.3 ± (1.33)	3.07	4.12 ± (0.7)
Rosetta	**3.45**	5.55 ± (1.02)	1.66	**2.11** ± (0.38)	6.85	9.45 ± (1.05)	3.63	5.29 ± (0.86)
Method	**2MR9**		**2P5K**		**3V1A**		**T0820-D1**	
Mod-ABC	2.32	4.12 ± (1.83)	6.81	10.3 ± (1.72)	1.74	2.53 ± (0.81)	**6.06***	11.82* ± (2.47)
Mod-ABC-CM	1.71*	2.29 ± (0.48)	2.69*	3.75* ± (0.94)	1.28*	**2.2*** ± (0.56)	9.66	14.39 ± (3.43)
MO-ABC-1	2.0	2.27* ± (0.28)	3.18	4.09 ± (0.65)	1.64	2.27 ± (0.47)	10.79	13.74 ± (2.63)
MO-ABC-2	1.84	2.44 ± (0.35)	3.76	4.42 ± (0.52)	1.73	2.35 ± (0.44)	9.86	13.3 ± (1.86)
Rosetta	**1.43**	**2.22** ± (0.69)	**1.57**	**2.29** ± (1.0)	**0.7**	2.51 ± (1.9)	7.34	**9.19** ± (1.7)
Method	**GDT_TS**							
	Highest	**Avg. (std.)**	**Highest**	**Avg. (std.)**	**Highest**	**Avg. (std.)**	**Highest**	**Avg. (std.)**
	1AB1		**1ACW**		**1AIL**		**1DFN**	
Mod-ABC	57.07	52.17 ± (2.94)	**77.72***	69.93* ± (5.34)	50.71	44.96 ± (2.52)	46.67	41.88 ± (3.43)
Mod-ABC-CM	63.04	59.1 ± (2.46)	72.41	65.41 ± (5.07)	60.36	47.9 ± (5.72)	**53.33***	46.98 ± (3.46)
MO-ABC-1	65.76	61.21 ± (2.41)	77.59	67.03 ± (5.97)	56.79	52.46 ± (5.95)	50.83	**48.13*** ± (2.11)
MO-ABC-2	**69.02***	**61.62*** ± (4.75)	74.14	67.24 ± (4.8)	**61.43***	**53.44*** ± (6.09)	50.0	46.35 ± (2.53)
Rosetta	62.5	56.45 ± (4.27)	77.59	**73.49** ± (3.33)	48.93	39.33 ± (5.36)	49.17	44.69 ± (2.6)
Method	**2MR9**		**2P5K**		**3V1A**		**T0820-D1**	
Mod-ABC	71.02	59.66 ± (6.86)	40.48	33.93 ± (3.02)	**55.73***	**53.97*** ± (1.1)	40.0	35.59 ± (2.6)
Mod-ABC-CM	79.55*	73.22* ± (4.72)	51.19*	47.02* ± (2.7)	55.2	52.66 ± (1.6)	36.94	34.13 ± (2.13)
MO-ABC-1	75.57	71.09 ± (3.13)	49.6	45.68 ± (2.18)	55.21	53.45 ± (1.04)	40.0	35.1 ± (4.04)
MO-ABC-2	76.7	71.45 ± (2.87)	45.63	44.44 ± (0.97)	54.17	52.67 ± (1.12)	40.28*	35.9* ± (2.94)
Rosetta	**83.52**	**73.79** ± (6.59)	**53.97**	**51.54** ± (1.85)	55.21	51.44 ± (4.63)	**45.28**	**39.62** ± (3.71)

Figure 1 shows the comparison between the 3-D topology of the models predicted by Mod-ABC (green), MO-ABC-1 (blue) and Rosetta (yellow) superimposed upon the experimentally determined structures (red). Analyzing the Table 2, we notice that Rosetta surpassed all of the other algorithms regarding the lowest and average RMSD values in 4 targets and related to the highest and average GDT_TS values in 3 and 4 cases, respectively. Although it is observable by visual inspection of Fig. 1 that the MO-ABC-1 and Rosetta reached overall target folding very similar to each other and comparable to the experimentally determined structures. Finally, such results denote the importance of adapting the metaheuristic to handle the specific complexities of the PSP problem.

5 Conclusion

In this paper, we proposed some variations of the artificial bee colony algorithm to deal with the protein structure prediction problem by introducing multi-objective strategies and exploration of knowledge from experimental proteins by the use of protein contact maps. The obtained results showed that our algorithms were able to find acceptable solutions concerning RMSD and GDT_TS structural measures and outperform their previous version in most of the cases, and also reached comparable solutions to the state of the art method of Rosetta regarding experimental protein structures. Besides that the obtained results are topologically similar to the experimentally determined structures, thus corroborating the proposed strategies' promising performance for the problem.

Acknowledgements. This work was supported by grants from FAPERGS [*16/2551-0000520-6*], MCT/CNPq [*311022/2015-4; 311611/2018-4*], CAPES-STIC AMSUD [*88887.135130/2017-01*] - Brazil, Alexander von Humboldt-Stiftung (AvH) [*BRA 1190826 HFST* CAPES-P] - Germany. This study was financed in part by CAPES - Finance Code 001.

References

1. Abriata, L.A., Tamò, G.E., Monastyrskyy, B., Kryshtafovych, A., Dal Peraro, M.: Assessment of hard target modeling in CASP12 reveals an emerging role of alignment-based contact prediction methods. Proteins: Struct. Funct. Bioinf. **86**, 97–112 (2018)
2. Adhikari, B., Hou, J., Cheng, J.: Protein contact prediction by integrating deep multiple sequence alignments, coevolution and machine learning. Proteins: Struct. Funct. Bioinf. **86**, 84–96 (2018)
3. Akay, B., Karaboga, D.: A modified artificial bee colony algorithm for real-parameter optimization. Inf. Sci. **192**, 120–142 (2012)
4. Borguesan, B., Inostroza, M., Dorn, M.: NIAS-server: neighbors influence of amino acids and secondary structures in proteins. J. Comput. Biol. **24**, 255–265 (2016)
5. Corrêa, L.D.L., Dorn, M.: A knowledge-based artificial bee colony algorithm for the 3-D protein structure prediction problem. In: 2018 IEEE Congress on Evolutionary Computation (CEC), pp. 1–8, July 2018

6. Cutello, V., Narzisi, G., Nicosia, G.: A multi-objective evolutionary approach to the protein structure prediction problem. J. R. Soc. Interface **3**(6), 139–151 (2006)
7. Dorn, M., e Silva, M.B., Buriol, L.S., Lamb, L.C.: Three-dimensional protein structure prediction: methods and computational strategies. Comput. Biol. Chem. **53**, 251–276 (2014)
8. Gao, W., Liu, S., Huang, L.: A global best artificial bee colony algorithm for global optimization. J. Comput. Appl. Math. **236**(11), 2741–2753 (2012)
9. Handl, J., Lovell, S.C., Knowles, J.: Investigations into the effect of multiobjectivization in protein structure prediction. In: Rudolph, G., Jansen, T., Beume, N., Lucas, S., Poloni, C. (eds.) PPSN 2008. LNCS, vol. 5199, pp. 702–711. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-87700-4_70
10. Jones, D.T., Singh, T., Kosciolek, T., Tetchner, S.: MetaPSICOV: combining coevolution methods for accurate prediction of contacts and long range hydrogen bonding in proteins. Bioinformatics **31**(7), 999–1006 (2014)
11. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J. Glob. Optim. **39**(3), 459–471 (2007)
12. Kim, D.E., DiMaio, F., Yu-Ruei Wang, R., Song, Y., Baker, D.: One contact for every twelve residues allows robust and accurate topology-level protein structure modeling. Proteins: Struct. Funct. Bioinf. **82**, 208–218 (2014)
13. Konak, A., Coit, D.W., Smith, A.E.: Multi-objective optimization using genetic algorithms: a tutorial. Reliab. Eng. Syst. Saf. **91**(9), 992–1007 (2006)
14. Li, G., Niu, P., Xiao, X.: Development and investigation of efficient artificial bee colony algorithm for numerical function optimization. Appl. Soft Comput. **12**(1), 320–332 (2012)
15. Moult, J., Fidelis, K., Kryshtafovych, A., Schwede, T., Tramontano, A.: Critical assessment of methods of protein structure prediction (CASP)-Round XII. Proteins: Struct. Funct. Bioinf. **86**, 7–15 (2018)
16. Olson, B., Shehu, A.: Multi-objective optimization techniques for conformational sampling in template-free protein structure prediction. In: International Conference on Bioinformatics and Computational Biology (2014)
17. Rohl, C.A., Strauss, C.E., Misura, K.M., Baker, D.: Protein structure prediction using Rosetta. Methods Enzymol. **383**, 66–93 (2004)
18. Schaarschmidt, J., Monastyrskyy, B., Kryshtafovych, A., Bonvin, A.M.: Assessment of contact predictions in CASP12: co-evolution and deep learning coming of age. Proteins: Struct. Funct. Bioinf. **86**, 51–66 (2018)
19. Unger, R., Moult, J.: Finding the lowest free energy conformation of a protein is an NP-hard problem. Bull. Math. Biol. **55**(6), 1183–1198 (1993)
20. Zhu, G., Kwong, S.: Gbest-guided artificial bee colony algorithm for numerical function optimization. Appl. Math. Comput. **217**(7), 3166–3173 (2010)