Anu Bourgeois · Pavel Skums
Xiang Wan · Alex Zelikovsky (Eds.)

LNBI 9683

Bioinformatics Research and Applications

12th International Symposium, ISBRA 2016
Minsk, Belarus, June 5–8, 2016
Proceedings

Springer

A Multiagent *Ab Initio* Protein Structure Prediction Tool for Novices and Experts

Thiago Lipinski-Paes, Michele dos Santos da Silva Tanus, José Fernando Ruggiero Bachega, and Osmar Norberto de Souza(✉)

Laboratório de Bioinformática, Modelagem e Simulação de Biossistemas - LABIO, Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, Brazil
osmar.norberto@pucrs.br

Abstract. Proteins are vital to most biological processes by performing a variety of functions. Structure and function are intimately related, thus highlighting the importance of predicting a proteins 3-D conformation. We propose GMASTERS, a multiagent tool to address the protein structure prediction (PSP) problem. GMASTERS is a general-purpose *ab initio* graphical program based on cooperative agents that explore the protein conformational space using Monte Carlo and Simulated Annealing methods. The user can choose the abstraction level, energy function and force field to perform simulations. Because bioinformatics demands knowledge from diverse scientific fields, its tools are intrinsically complex. GMASTERS abstracts away some of this complexity while still allowing the user to learn and explore research hypotheses with the advantage of an embedded graphical interface. Although this abstraction comes at a cost, its performance is similar to state-of-the-art methods. Here, we describe GMASTERS and how to use it to explore the PSP problem.

Keywords: PSP Problem · Multiagent system · Monte Carlo · AB model

1 Introduction

Proteins are polymers of 20 different building blocks, called amino acids. These building blocks interact physicochemically resulting in a unique spatial conformation for each protein [1]. Due to advances in the Genome Project, there is a large number of protein sequences available in the GenBank [2]. Currently, there are about 82 million non-redundant protein sequences. However, in the Protein Data Bank or PDB [3], there are approximately 115,000 3-D structures of proteins. Eliminating redundancy by filtering very similar structures (SCOP), we get only 1,393 different folds or topologies. Under these circumstances, it is evident the huge gap between our competence to produce protein sequences and to determine 3-D structures of new proteins with yet unknown folds [4]. Computer Science, more specifically Structural Bioinformatics, has been a great ally on reducing this gap.

A. Bourgeois et al. (Eds.): ISBRA 2016, LNBI 9683, pp. 163–174, 2016.
DOI: 10.1007/978-3-319-38782-6_14

The Protein Structure Prediction (PSP) problem emerged in the 60's and even today its solution remains a major challenge to molecular biology [5]. Limitations of 3-D structure experimental determination techniques, such as X-ray diffraction crystallography and nuclear magnetic resonance, highlight the importance of computational methods to predict the structure of proteins. Advances in handling the PSP problem will allow us to predict the 3-D structure of proteins with relevant applications in the biopharmaceutical industry. It will also improve our understanding of proteins involved in vital processes, including diseases such as cancer [6]. Considering the difficulties faced by traditional approaches (*in vitro* and *in vivo* experiments) concerning biological systems, the use of computers becomes attractive, making possible to execute low-cost and faster *in silico* experiments. An application that involves PSP must consider the system's real time adaptability, i.e., parameters modifications such as thermal bath temperature. There is a clear need for *in virtuo* experiments: computer simulations susceptible to perturbations during execution. While the easy modification of parameters is a typical property of all computer simulations (*in silico* experiments), the easy modification of the experiment itself is a property of multiagent systems (MAS), resulting in *in virtuo* experiments [7,8].

Here, we present a general tool that allows addressing PSP according to the user needs. The user is free to choose (via a Graphical User Interface) both abstraction level and force field to guide the simulation. The agents are organized in hierarchical levels. Optimization is done by Monte Carlo/Simulated Annealing and the user can modify parameters and optimization method. GMASTERS can be obtained at labio.org.

2 Background

2.1 Proteins and the PSP Problem

If we take a deep look into living organisms and observe their cellular level functions we will notice these functions are carried out by a variety of proteins. Zooming in our body toward the cells, we will realize that each cell has its own copy of the genome. The genome is what gives the cell functionality. From a computational point of view, we could state that the genome is a string or sequence composed by four kinds of letters (A,C,T and G) referring to four kinds of nucleotides. Scanning the genome from the left to the right certain substrings are found (called genes). Each triplet of nucleotides is a code that can be parsed to one of the 20 amino acids. The concatenation of parsed triplet nucleotides (amino acids) generates a protein sequence. A protein linear sequence of amino acid residues is called its primary structure [9]. Figure 1 shows a typical representation of a protein, as well as the abstraction level used in this work.

The PSP problem is the problem of predicting the 3-D structure of a protein starting from its primary structure or amino acid sequence. The physical process by which a polypeptide folds into a functional protein is an old question (reviewed by Snow [10]) and continues to be one of the biggest challenges in current structural bioinformatics [5].

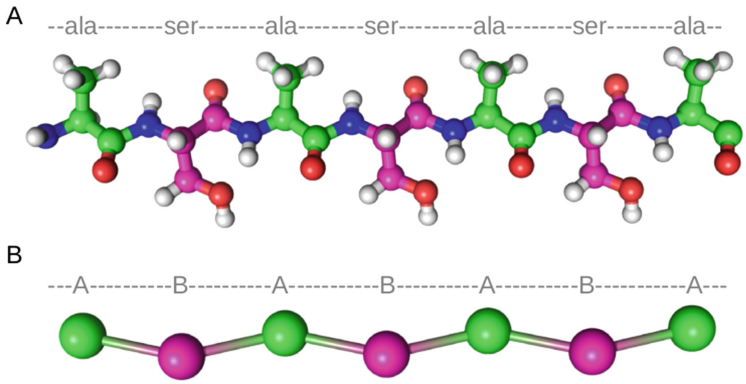

Fig. 1. (A) The representation of an extended hypothetical protein chain formed by alternated residues of alanines (hydrophobic) and serines (hydrophilic). (B) The representation of the abstraction level used in this work, where hydrophobic amino acids are marked as "A" while the hydrophilic amino acids are marked as "B".

The protein's 3-D structure is directly linked to its function. Determining its spatial conformation experimentally is expensive and time consuming. Bioinformatics has the important role of accelerating this knowledge discovery [11]. This paper's approach is based on Anfinsen's proposal which states that, at the environmental conditions (temperature, solvent concentration and composition) at which folding occurs, the native structure is a unique, stable and kinetically accessible minimum of the protein's free energy. However, finding this structure is not trivial and even simplified methods have NP-Complete complexity [12]. Figure 2 shows a hypothetical uni-dimensional energy function to illustrate the challenge of finding the lowest energy and achieving the native conformation.

Still regarding the inherent difficulty of the problem we can cite Levinthal's paradox [13], which states that for a 100-length chain there will be at least 2^{100} possible conformations (considering only two degrees of freedom), characterizing it as an intractable problem [14]. In the last five decades different algorithmic approaches have been tested and, although there has been progress, the problem remains unsolved even for small proteins. While the ultimate goal is to predict the 3-D or tertiary structure from the primary structure, the current knowledge and computing power is insufficient to handle a problem of such complexity [15].

2.2 Multiagent Systems

Multiagent systems (MAS) are part of the Artificial Intelligence field and refer to the modeling of autonomous agents in a common universe. MAS is a relatively new sub-field of Computer Science - it has been studied since about 1980 - and the field has gained widespread recognition around 1990 [16].

Agents are computational entities that interact with an environment and are goal-oriented, having a body and a location in time and space. An agent is capable of autonomous and flexible actions to reach its goals. According to

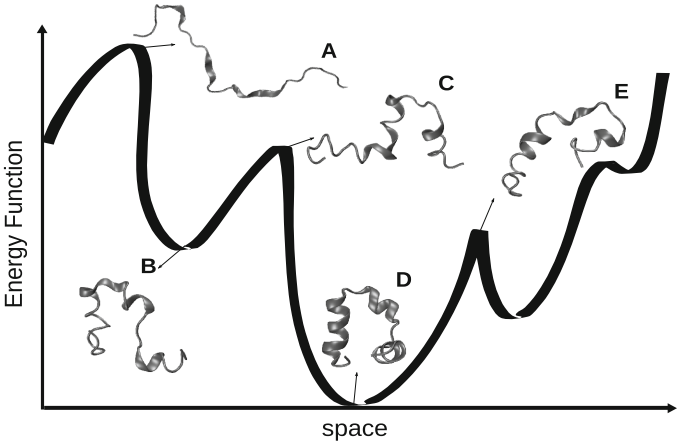

Fig. 2. Simplified protein structure prediction funnel. The figure shows an unidimensional component of a free energy hyper-surface where (A), (B), (C) and (E) represent local minima and (D) represents the global minimum (native structure).

Russel and Norvig [17], a rational agent is able to select the action that maximizes its performance, given the evidences accumulated by its perceptions and internal knowledge. According to Bradshaw [18], an agent is a software entity that works continuously and autonomously in a given environment. It should perceive and act in its environment in a flexible and intelligent way. It may learn from experience, communicate, cooperate with other agents and have the mobility required to satisfy its goals. The agents' autonomy means they have an existence independent of other agents, and have to achieve their own goals. Although there is no universally accepted definition of agent, some properties as autonomy, pro-activeness, reactivity and social ability are intrinsic of its behavior. A set of agents acting in an environment characterizes a MAS.

An agent needs an environment that can be of numerous types and complexities. The environment complexity is strongly linked to the agent's complexity.

Netlogo [19] is a very popular agent-based modeling tool and it is particularly suited for modeling complex systems that take time into account and where hundreds or thousands of agents can be programmed and interact independently, making possible to explore the connection between micro and macro-levels of behavioral patterns. One of the remarkable advantages of Netlogo is its embedded tools, and BehaviorSpace is one of them, offering the possibility of automatically perform a large set of experiments by changing parameters' values. Due to the BehaviorSpace capability, it is possible to explore more resourcefully the configuration space in PSP and tune parameters to improve results.

3 MASTERS

MASTERS [20] was built using Netlogo v5.0 (see Sect. 2.2 for details). A strong Netlogo's peculiarity is that it was developed for educational purposes, providing

a rich modeling environment which allows experimental coding. *Ab initio* methods require three elements [21]: (i) a method for searching the energy landscape, (ii) an energy function (iii) a geometrical representation of the protein chain. As a pure *ab initio* method, MASTERS must comprise these elements. How MASTERS addresses these items is shown hereafter.

Searching Agents. The Searching Agents have the mission of exploring the conformational space. Usually, one or more searching agents are associated to each amino acid in the protein, depending on the selected abstraction. The agents' position is expressed as Cartesian coordinates, resulting in movements that are local, i.e., they do not affect the position of other agents.

Director Agent. The Director Agent has total knowledge about the protein's current spatial conformation and coordinates searching agents, aiming at a more efficient spatial exploration. The Director agent has no representation in the Cartesian space. It is an agent that acts on the 3-D space from outside. Director Agents perform global moves on the searching agents. It is mandatory to have at least one Director Agent in the simulation.

Environment Agent. There is only one Environment Agent. Its role is to control the simulation flow, simulated annealing scheme, number of movements per time/temperature step, real time plots, and outputs.

3.1 Hierarchical Cooperation

This is related to (i). A core MASTERS concept is the agents hierarchical organization (Fig. 3). Higher-level agents have the role of coordinating the actions of lower-level agents [22]. In a bottom-up hierarchical order.

Searching and Director agents cooperate to find the conformation that better suits their goals. They are autonomous, not depending on each other to perform their moves. The agents are reactive to their environment: the Searching agents can see their neighborhood, whereas the Director agent has full information on the environment, being able to influence the simulation in a broader manner.

MASTERS' environment is treated as a box with dimensions delimited by the user. In a multiagent perspective it can be considered both accessible (for Director and Environment agents) and inaccessible (for Searching agents, who have limited access to information), non deterministic, due to its stochastic nature, dynamic and discrete.

3.2 Sampling Technique

Also related to (i), MASTERS' movements are controlled by Monte Carlo (MC). MC [23] is one of the most used energy landscape exploration techniques. It has a probabilistic nature and it is a method for generating different configurations of a particles system, i.e., points in space compatible with external conditions.

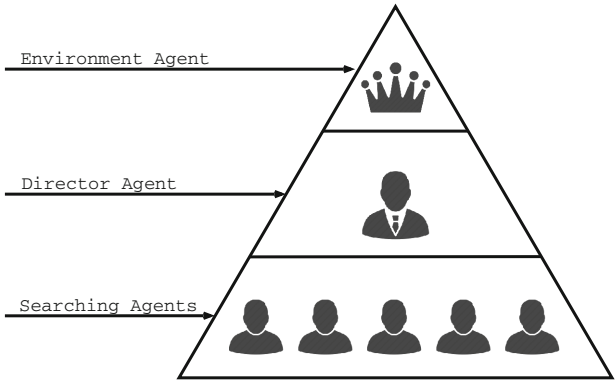

Fig. 3. MASTERS' hierarchy. The Environment Agent is responsible for the simulation flow. Searching agents explore the configuration space by performing move attempts. The Director agent is able to affect the position of all other agents in the environment.

Looking back to Fig. 2 we can pay attention to points (A) and (B). From there, MC defines two conformational states S_A and S_B, each one with its corresponding energy E_A and E_B. If $E_B < E_A$ the move is accepted. If $E_B > E_A$ there is still the possibility of accepting the move. However, in such cases, the probability of accepting a move from S_A to S_B follows the Eq. 1, where k is the Boltzmann constant and T the temperature. Once the system reaches (B) it is allowed to accept movements toward (C), hoping to someday find (D).

$$e^{-(EB-EA)kT} = e^{-(\Delta BA)kT} \tag{1}$$

Once an MC simulation is in progress it is necessary to constantly check its acceptance ratio, as it influences the agents' movements. Usually a ratio of 0.5 can be considered an optimal initial value for MC simulations involving PSP [24]. Since our method is based on different types of autonomous agents, each type of agent needs to have its acceptance ratio average assessed separately.

The MC method has drawbacks, though. In the PSP problem, the temperature of the system determines the size of energy barriers that could potentially be overcome. When dealing with temperatures that are too low, MC will not explore far from the minimum energy found, leading to local minima. Simulated Annealing (SA) is a simple MC modification that turns it into a global optimizer. At the beginning of the simulation the temperature is set high and fairly high-energy barriers are overcome. Then the system is gradually cooled, eventually being confined to a single energy. Due to the gradual cooling rate (logarithmic), the system ends up spending more time in low energy regions. This may increase the chances of finding the lowest energy state although there is no assurance [11]. Concerning SA performance, convergence is guaranteed only if the temperature is reduced to zero logarithmically. In MASTERS the temperature is gradually decreased according to Eq. 2, where $\alpha = 0.98$:

$$T_{k+1} = T_k * \alpha \tag{2}$$

Regarding the role of the agents on sampling, the simulation relies on accounting the number of movement attempts for each agent. To achieve minima at a given temperature, the system should explore the conformational space a large number of times. Counters are used to summarize the average number of move attempts per agent type (both Searching and Director Agents). Every time step is related to a specific temperature. The system will be stuck at each temperature until an average number of movement attempts has been attained. The simulation ends when the temperature reaches a value set by the user.

3.3 Choosing the Energy Function/Abstraction Level

Here the elements (ii) and (iii) are addressed. MASTERS currently incorporates MC/SA as sampling technique, not allowing the use of Molecular Dynamics or Genetic Algorithms. However, one of its particular characteristics is its orthogonality with respect to the simulation flow and the energy function/abstraction level used. The user must choose which energy function/force field to use and this is a primordial step within the framework's generality. MASTERS was not built exclusively to a particular energy function. The framework can be applied to a wide range of optimization problems that involve Cartesian coordinates (2-D or 3-D). Regardless of the problem, the energy function will directly affect the number of searching agents and their movements.

3.4 GMASTERS

The educational purpose is one of the main focuses of GMASTERS, providing a user-friendly interface where students that are not familiar with computer programming can explore the PSP problem in interactive ways. The MASTERS' version presented here includes a new graphical user interface named GMASTERS. The latter is an alternative to the old MASTERS' developed on the NetLogo environment. GMASTERS is written in the Python language and employs the GTK+ toolkit, which provides more sophisticated widgets and friendlier interface. We hope the interface to considerably assist users, something not commonly taken into consideration when simulating proteins. The results generated are plotted using the Matplotlib toolkit. For the visualization and 3D analysis of the obtained models GMASTERS connects with PyMOL [25] using a similar approach to GTKDynamo [26]. A typical session snapshot is shown in Fig. 4.

Creating Projects. In GMASTERS the user always works inside a given project. To create a new project the user provides information such as project directory, user name and protein sequence. Every project contains, beyond the information provided by the user, date and time of creation and the list of jobs.

Setup and Running Simulations. Once a project is created the user can setup and run Monte Carlo simulations inside GMASTERS. The user can set parameters such as box dimensions, temperature and maximum movement. Every new

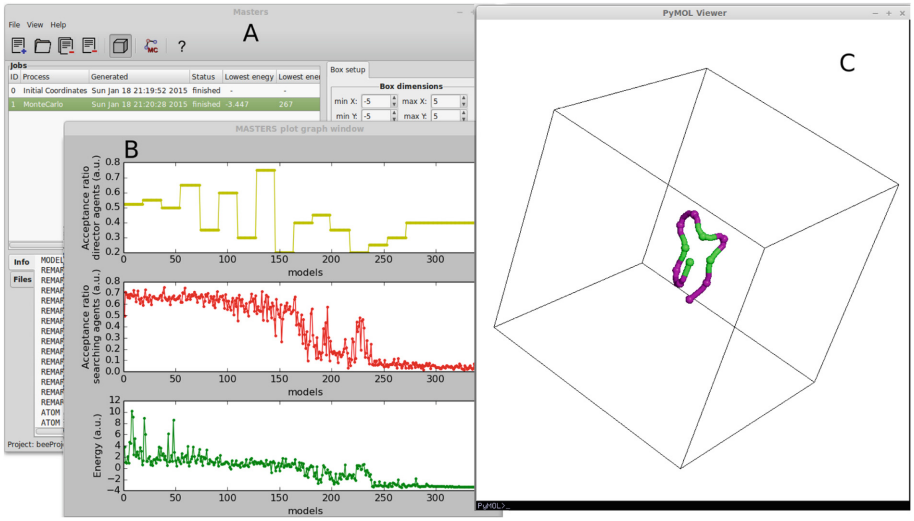

Fig. 4. GMASTER's GUI. (A) Main window, displaying simulation projects and parameters setup, (B) Plot window, showing energy and acceptance ratio, (C) PyMOL viewer, with the predicted conformation.

simulation result is a new item in the job list which provides time of creation, status, lowest obtained energy and the current step's energy. These data allow a preliminary comparison between different simulations and, at the same time, keep chronologically ordered steps performed by the user.

Data Analysis and Model Visualization. As a simulation proceeds its results are stored in a log file. GMASTERS is able to read and interpret these log files and display graphics of the relevant information, which can then be saved and manipulated by the user. Trajectories are also generated. These come in PDB format, and can hold a variety of information, including coordinates and energies.

4 Case Study

4.1 Geometry Representation and Energy Function

To examine GMASTERS' behavior and effectiveness, we adopted the same simplified model used in our last work [20], the AB Model (see Fig. 1).

The simplest and most conventional model among all applied abstractions used in PSP is the HP Model [27]. The HP model divides all 20 amino acids into two different groups, the hydrophobic (H) and the hydrophilic (P) ones. The amino acids are placed at an on-lattice grid, and the energy computation at each conformation takes into account only interactions between next-neighbored nonadjacent hydrophobic amino acids [27]. The energy of a conformation is the number of hydrophobic-hydrophobic contacts that are adjacent on the lattice,

but not adjacent on the string (sequence). The main idea is to force the establishment of a compact hydrophobic core as observed in real proteins [28]. Lattice models have proven to be useful tools for reasoning about the PSP problems complexity [27] and despite its high abstraction level, the PSP problem with HP models is still an NP-Complete challenge [29]. The AB model is a lattice model in which the amino acids are once again divided into two groups: hydrophobic amino acids are marked as A while the hydrophilic ones are marked as B.

Many authors have been using this model as starting point for PSP understanding [30–32]. The AB model, in comparison to the HP model, has the additional capability of collecting information about local interactions that might be significant for the local structure of protein chains. This allows finding compact, well-defined native structures that would not be found if these local interactions were neglected [33]. Unlike the HP model, the interactions considered in the AB model include both sequence independent local interactions and the sequence dependent Lennard-Jones term that supports the energy convergence to a hydrophobic core [32,34].

The AB off-lattice energy model is described by Eq. 3, where θ is the bend angle between the two bonds defined by three consecutive residues and r_{ij} is the distance between residues i and j [32]. The first sum, the backbone bending potential, calculates the bending angle energy of the protein chain. The double sum is the Lennard-Jones potential. It calculates the long-range interaction energy, which is attractive for pairs of the same amino acids (AA or BB) and repulsive for AB pairs. The residue specific prefactor C is given by the Eq. 4.

$$E = \sum_{i=2}^{n-1} \frac{1}{4}(1 - cos\theta) + \sum_{i=1}^{n-2} \sum_{j=i+2}^{n-1} [r_{ij}^{-12} - C(\xi_i, \xi_j) r_{ij}^{-6}] \tag{3}$$

$$C(\xi_i, \xi_j) = \left\{ \begin{array}{l} +1, \quad \xi_i \xi_j = A \\ +1/2, \, \xi_i \xi_j = B \\ -1/2, \, \xi_i \neq \xi_j \end{array} \right\} \tag{4}$$

4.2 Target Sequence

Although several works in the literature use Fibonacci sequences as targets for their simulation [32,34], we chose to work on real sequence targets. The aminoacids A, C, G, I, L, M, P and V were set to hydrophobic (class A) and D, E, F, H, K, N, Q, R, S, T, W and Y were set to hydrophilic (class B). As means of demonstration we chose the PDB ID 1AGT protein, a recurrent target [35].

4.3 Simulation Setup and Running

Once the target sequence (and abstraction level) is chosen, a simulation setup is required. Here it is possible to parametrize the MC/SA scheme, selecting values to initial temperature, temperature decrease ratio, movement amplitudes, etc.

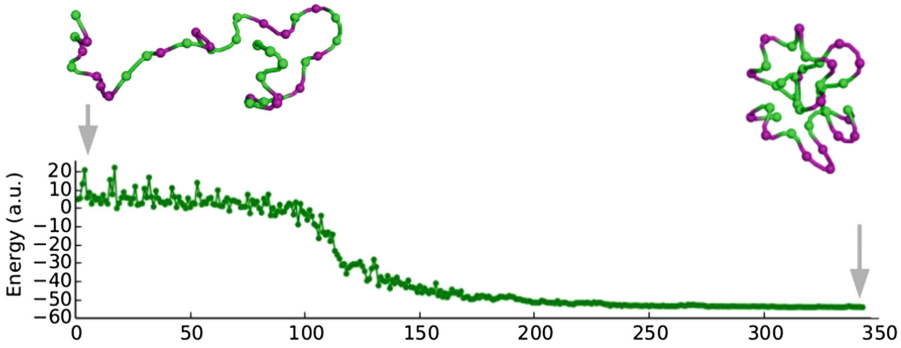

Fig. 5. The system starts with a high energy and the different kind of agents interact microscopically (MC/SA controlled moves) yielding a macroscopic behavior: the protein folding/energy convergence. The outputs are provided in a PyMOL canvas.

4.4 Data Analysis and Model Visualization

While the simulation runs it is possible to visualize the current energy profile and structure. Figure 5 shows that in the beginning of the simulation the protein is nearly unfolded and the first steps' energy is high. As the simulation progresses and the system cools down, lower levels of energy are reached and the protein starts to fold into more favorable conformations. If the system's energy doesn't start to decrease, it may be an indication to review the simulation setup parameters. In this case study, the simulation progresses normally and we can see the energy converging to lower levels. Furthermore it was observed an improvement in CPU time.

5 Conclusion and Future Works

This paper presented a tool for handling the protein structure prediction problem. More specifically GMASTERS, a GUI developed in GTK that runs over MASTERS core. The focus of the application is the user, making possible for students with different knowledge levels to learn about PSP and Monte Carlo and for experts to test their own methods.

To explore the new features available in GMASTERS we performed a case study with the protein whose PDB ID is 1AGT and, in addition to the new facilities provided, it was noticed a considerable improvement in CPU time.

We plan to evolve GMASTERS' architecture by parallelizing simulations, improving its overall performance. In addition, we intend to use different coarse grained abstraction levels and therefore different force fields.

Acknowledgements. This work was supported by grants CNPq-305984/20128 and FAPERGS-TO2054-2551/13-0 to ONS, a CNPq Research Fellow. TLP and MSS are supported by CAPES, JFRB by a CAPES/FAPERGS DOCFIX postdoctoral fellowship.

References

1. Lesk, A.M.: Introduction to Bioinformatics, 3rd edn. Oxford University Press, Oxford (2008)
2. Clark, K., Karsch-Mizrachi, I., Lipman, D.J., Ostell, J., Sayers, E.W.: Genbank. Nucleic Acids Res. **44**, D67–D72 (2016)
3. Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E.: The protein data bank. Nucleic Acids Res. **28**(1), 235–242 (2000)
4. Pavlopoulou, A., Michalopoulos, I.: State-of-the-art bioinformatics protein structure prediction tools. Int. J. Mol. Med. **28**(3), 295–310 (2011)
5. Dill, K., MacCallum, J.: The protein-folding problem, 50 years on. Science **338**(6110), 1042–1046 (2012)
6. Duan, Y., Kollman, P.A.: Computational protein folding: from lattice to all-atom. IBM Syst. J. **40**(297–309), 0018–8670 (2001)
7. Amigoni, F., Schiaffonati, V.: Multiagent-based simulation in biology. In: Magnani, L., Li, P. (eds.) Model-based Reasoning in Science, Technology, and Medicine, SCI, vol. 64, pp. 179–191. Springer, Heidelberg (2007)
8. Tisseau, J.: Virtual reality, in virtuo autonomy. Ph.D. thesis, University of Rennes 1 (2001)
9. Nelson, D., Cox, M.: Lehninger Principles of Biochemistry, 5th edn. W. H. Freeman and Company, New York (2008)
10. Snow, C.D., Sorin, E.J., Rhee, Y.M., Pande, V.S.: How well can simulation predict protein folding kinetics and thermodynamics? Annu. Rev. Biophys. **34**, 43–69 (2005). Annual Reviews, Palo Alto
11. Zvelebil, M., Baum, J.: Understanding Bioinformatics. Garland Science, US (2007)
12. Crescenzi, P., Goldman, D., Papadimitriou, C., Piccolboni, A., Yannakakis, M.: On the complexity of protein folding. J. Comput. Biol. **5**, 597–603 (1998)
13. Levinthal, C.: Are there pathways for protein folding? J. Med. Phys. **65**(1), 44–45 (1968)
14. Tramontano, A.: Integral and differential form of the protein folding problem. Phys. Life Rev. **1**(2), 103–127 (2004)
15. Helles, G.: A comparative study of the reported performance of ab initio protein structure prediction algorithms. J. Roy. Soc. Interface **5**(21), 387–396 (2008)
16. Wooldridge, M., Jennings, N.: Intelligent agents - theory and practice. Knowl. Eng. Rev. **10**(2), 115–152 (1995)
17. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach, 2nd edn. Prentice Hall, Englewood Cliffs (2003)
18. Bradshaw, J.M.: An introduction to software agents. In: Bradshaw, J.M. (ed.) Software Agents, pp. 3–46. AAAI Press / The MIT Press, Cambridge (1997)
19. Tisue, S., Wilensky, U.: Netlogo: a simple environment for modeling complexity (2004)
20. Lipinski-Paes, T., Norberto de Souza, O.: MASTERS: a general sequence-based multiagent system for protein tertiary structure prediction. Electron. Notes Theor. Comput. Sci. **306**, 45–59 (2014)
21. Osguthorpe, D.J.: Ab initio protein folding. Curr. Opin. Struct. Biol. **10**(2), 146–152 (2000)
22. Roli, A., Milano, M.: MAGMA: a multiagent architecture for metaheuristics. IEEE Trans. Syst. Man Cybern. Part B (Cybernetics) **34**, 925–941 (2004)

23. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equation of state calculations by fast computing machines. J. Chem. Phys. **21**(6), 1087–1092 (1953)
24. Allen, P., Tildesley, D.: Computer Simulation of Liquids. Oxford science publications, Clarendon Press (1987)
25. Schrödinger, L.L.C.: The PyMOL molecular graphics system, February 2016
26. Bachega, J.F.R., Timmers, L.F.S.M., Assirati, L., Bachega, L.R., Field, M.J., Wymore, T.: GTKDynamo: A PyMOL plug-in for QC/MM hybrid potential simulations. J. Comput. Chem. **34**(25), 2190–2196 (2013)
27. Dill, K.: Theory for the folding and stability of globular-proteins. Biochemistry **24**(6), 1501–1509 (1985)
28. William, E.H., Alantha, N.: Protein structure prediction with lattice models. Chapman & Hall/CRC Computer & Information Science Series, pp. 30-1–30-24. Chapman and Hall/CRC (2005)
29. Berger, B., Leighton, T.: Protein folding in the hydrophobic-hydrophilic (HP) model is NP-complete. J. Comput. Biol. **5**(1), 27–40 (1998)
30. Bachmann, M., Arkin, H., Janke, W.: Multicanonical study of coarse-grained off-lattice models for folding heteropolymers. Phys. Rev. E Stat. Nonlin Soft Matter Phys. **71**(3), 031906 (2005). doi:10.1103/PhysRevE.71.031906
31. Hsu, H.P., Mehra, V., Grassberger, P.: Structure optimization in an off-lattice protein model. Physical Review E **68**(3), 037703 (2003). doi:10.1103/PhysRevE.68.037703
32. Stillinger, F., Head-Gordon, T.: Collective aspects of protein-folding illustrated by a toy model. Phys. Rev. E **52**(3), 2872–2877 (1995)
33. Irback, A., Peterson, C., Potthast, F., Sommelius, O.: Local interactions and protein folding: a three-dimensional off-lattice approach. J. Chem. Phys. **107**(1), 273–282 (1997)
34. Stillinger, F., Head-Gordon, T., Hirshfeld, C.: Toy model for protein-folding. Phys. Rev. E **48**(2), 1469–1477 (1993)
35. Krezel, A.M., Kasibhatla, C., Hidalgo, P., Mackinnon, R., Wagner, G.: Solution structure of the potassium channel inhibitor agitoxin 2: Caliper for probing channel geometry. Protein Sci. **4**(8), 1478–1489 (1995)