An Efficient Solvent Accessible Surface Area calculation applied in Ab Initio Protein Structure Prediction

Daniel Bonetti[1], Horacio Pérez-Sánchez[2], and Alexandre Delbem[1]

[1]Computer Systems Department, University of Sao Paulo,
Institute of Mathematical Sciences and Computation, Avenida Trabalhador
Sao-carlense, 400, Sao Carlos, SP, Brazil
{dbonetti,acbd}@icmc.usp.br
[2]Bioinformatics and High Performance Computing Research Group (BIO-HPC)
Computer Science Department Universidad Católica San Antonio de Murcia
(UCAM), 30107 Guadalupe, Spain
hperez@ucam.edu

Abstract. Knowing the structure of proteins is an essential step in developing new medicines. This is a very time consuming and expensive process. Researchers from many different areas are trying to find new and efficient ways to discovery protein structures. This is known as the Protein Structure Prediction (PSP) problem and it is divided into experimental and *in silico* methods [3]. In this work, we use the *in silico* approach. The program is called ProtPred [4], which is an Evolutionary Algorithm (EA) that looks for feasible protein configurations in the search space from their amino acid sequences [5].

We know that the bottleneck of EA is the fitness function, which is used to evaluate solutions. The fitness function of ProtPred is composed by the sum of bonded and non-bonded energies. However, in this work, we used only van der Waals energy and Solvation energy, which also need to compute the Solvent Accessible Surface Area (SASA). We know that these two functions are time consuming processes, as they need to compute a pair-wise interaction among the atoms. Each of these energies has a specific contribution. For example, van der Waals energy tends to attract atoms at a certain distance and avoids atoms overlapping with each other in short distances, while the SASA maintains the compactness of structure.

The time needed for a single fitness function call with van der Waal and Solvation energy by ProtPred is relatively fast. In order to look for a promising solution, the fitness function must be called hundreds of thousands of times, even for a small protein, i. e the bottleneck of any EA for PSP is, in general, the fitness function. For this reason, we need to use fast and efficient van der Waals and Solvation energy. This will allow us to accelerate the whole ProtPred that enables it to predict more and larger protein configurations.

We demonstrated in a recent journal an efficient way of computing van der Waals energy, and we applied it to PSP using cell-list algorithm [1]. We also showed how to efficiently compute SASA using neighbor lists

with Graphic Processing Units (GPU) based method, called MURCIA [7]. Both works showed a complexity reduction from $O(n^2)$ to $O(n)$, rendering a significant time reduction.

In this research, we replaced the old $O(n^2)$ Solvation energy from ProtPred with the new SASA implementation provided by MURCIA. Both CPU and GPU implementations of MURCIA were used, allowing us to make time comparisons between the techniques. Both versions were able to produce the same SASA value. In order to compute the implicit interaction of the protein with solvent, we related the Accessibility Solvation Parameters (ASP) of Carbon, Nitrogen and Oxygen atoms [6] to their SASA calculated by MURCIA. The sum of ASP values times SASA gives the Solvation energy used in the fitness function of ProtPred together with van der Waals energy.

The experiments were performed in an Intel Xeon E5506 with a NVIDIA Tesla C2075. Three different times were measured: one for van der Waals energy, another for Solvation energy and the remaining time for the EA (initialization, population generation, composing new solutions etc.). It performed six runs of ProtPred with MURCIA in CPU, plus another six in GPU. Five proteins were chosen from PDB, ranging from 25 to about 1,000 residues, in order to evaluate the time and the prediction quality of the range of proteins. The convergence criterion adopted in EA was limited to one million evaluations.

The running time of Solvation energy produced better speedups for all sizes of proteins that we tested. The smallest protein (with 25 amino acids) had a calculated speedup of 2.1, and the largest protein (with 971 amino acids) had a speedup of 26.5. This is approximately 3% of the running time of Solvation energy in CPU. The running time of van der Waals energy and the remaining parts of EA, were kept constant in both versions in which Solvation energy was computed in CPU and GPU. This also shows that computing Solvation energy in GPU and CPU does not cause any noise in the remainder of the algorithm.

Also, it is possible to notice that the speedup line for Solvation energy in GPU(in relation to CPU version) grows linearly according to the number of atoms. This result is of interest, since we are now able to linearly increase the speedup using only one processor and one GPU, according to the number of atoms.

The smallest protein (with 25 residues) had the best RMSD, measured at 0.942, and the second best (with 50 residues) had a RMSD of 5.088. For proteins above 50, the RMSD was not as good as these small proteins. In order to work with non-small proteins, the parameters of ProtPred should be calibrated first. This is likely to be necessary to increase the number of evaluation functions. Initially, we ran the experiments for timing comparison purposes. Thus, from this point, it will be possible to properly calibrate the EA by using fast Solvation energy calculation.

Although van der Waals energy starts being the bottleneck for proteins above 1,000 atoms, the van der Waals would require approximately 34 hours of computation for a protein with 971 residues, against 4.5 hours of computation for Solvation energy in GPU. It had a speedup of 7.75, although the Solvation energy is slightly more complex compared

to the van der Waals energy. This work therefore demonstrates that it is possible to reduce the time of an EA applied to PSP, when using efficient energy function in fitness function. The main focus of this work was to show and evaluate the performance of an EA (called ProtPred), using a fast and efficient calculation of SASA made in GPU (called MURCIA). The results showed that the speedup of the efficient technique grows in a linear trajectory, according to the number of atoms.

The problem of computing the speedups in this work is that it requires the computation of both the GPU and CPU versions. As we can see, just the running time of Solvation energy in CPU for the largest proteins requires approximately 110 hours computing 1 million evaluations, while it only requires approximately 4.5 hours to compute the same values in the GPU. Thus, in order to compute the speedup, we spent a large amount of time computing the reference values, i.e., the values used to compose the numerator of the speedup. We also saw that van der Waals energy also turned into bottleneck for larger proteins. For a future occasion, we can also take advantage of the GPU algorithm of SASA and apply it to van der Waals energy. This would probably make a huge difference for non-small proteins as well. Finally, using a multi-objective EA would improve the solutions found by the searching process since there are two energy functions. These could be individual objectives, and treated as a multi-objective problem [2].

Keywords: Solvent Accessible Surface Area, Protein Structure Prediction, Ab initio, Graphic Processing Unit.

Acknowledgments

This work was partially supported by the computing facilities of Extremadura Research Centre for Advanced Technologies (CETA-CIEMAT), funded by the European Regional Development Fund (ERDF). CETA-CIEMAT belongs to CIEMAT and the Government of Spain. The authors also thankfully acknowledge the computer resources and the technical support provided by the Plataforma Andaluzad e Bioinformaťica of the University of Málaga. The authors also would like to acknowledge FAPESP (Brazilian research foundation) for the financial support given to this research.

References

1. D. R. Bonetti, A. C. Delbem, G. Travieso, and P. S. L. de Souza. Enhanced van der waals calculations in genetic algorithms for protein structure prediction. *Concurrency and Computation: Practice and Experience*, 25(15):2170–2186, 2013.
2. C. R. S. Brasil, A. C. B. Delbem, and F. L. B. da Silva. Multiobjective evolutionary algorithm with many tables for purely ab initio protein structure prediction. *Journal of Computational Chemistry*, 34(20):1719–1734, 2013.
3. J. M. Bujnicki. *Prediction of Protein Structures, Functions, and Interactions*. Wiley, 2009.

4. T. W. de Lima, R. A. Faccioli, P. H. R. Gabriel, A. C. B. Delbem, and I. N. da Silva. Evolutionary approach to protein structure prediction with hydrophobic interactions. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, GECCO '07, pages 425–425, New York, NY, USA, 2007. ACM.
5. J. H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. University of Michigan Press, 1975.
6. B. Lee and F. Richards. The interpretation of protein structures: Estimation of static accessibility. *Journal of Molecular Biology*, 55(3):379 – IN4, 1971.
7. Q. Zhang, J. Wang, G. D. Guerrero, J. M. Cecilia, J. M. Garca, Y. Li, H. Prez-Snchez, and T. Hou. Accelerated conformational entropy calculations using graphic processing units. *Journal of Chemical Information and Modeling*, 53(8):2057–2064, 2013.