UAB

Universitat Autònoma de Barcelona

Deep Learning-Assisted Evolutionary Algorithms: Exploring docking methods with GaudiMM2

Initial Report

Student:
Arnau Solé Porta, 1630311

Tutor:
David Castells Rufas

March 2025

Contents

1 Objective

The technical challenge faced in this end of course work revolves around the improvement of an evolutionary algorithm. It has been shown on some papers such as [1] and [2] that introducing deep learning techniques on the different steps of such algorithms can optimize its results. In the specific framework of this project, the intended improvement is to reduce the computation time of the algorithm and achieve faster convergences. Instead of randomizing the generation and mutation of new populations, the idea is to train models that guide new individuals towards those features most likely to improve the fitness score.

2 Contextualization

As implied in the previous section, there exists a working evolutionary algorithm to be improved. It is designed to find feasible binding sites between molecules and proteins. Protein binding is the process by which molecules attach to proteins, changing its composition and modifying its behaviour, and a binding site is one possible location of the molecule within the protein. Finding those sites is not a trivial task, since the exploration space, often sensitive to atom-level configurations, can be huge and difficult to evaluate.

Usually, the fitness of a certain binding site is evaluated through the resulting energy of the system (and thus the resulting stability) using a docking function. Docking scoring functions are especially difficult to design, but in general designing evaluating methods for binding sites requires a strong biological and chemical background, which is why it is outside of the scope of this work to optimize the performance of the algorithm.

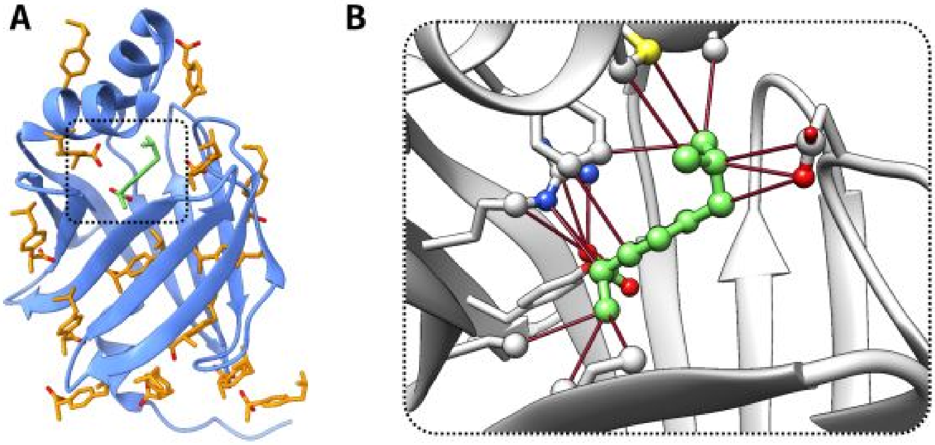

Figure 1: Binding example. Image and note adapted from [3]. Exploring the relative orientation of the ibuprofen molecule with respect to the FABP4 protein structure. (A) Multiple options are possible. In orange there are depicted several locations and orientations of the ibuprofen molecule. In green it is highlighted the chosen pose. (B) Detail of ibuprofen-FABP4 interactions that stabilize the chosen pose, highlighted in thin dark red sticks.

2.1 Usefulness

Putting molecules and proteins together has real-world uses and is widely applied in the pharmaceutical industry to design drugs. Especially benefitting use cases arise when proteins act as enzymes, which is one of their several biochemical functions. Enzymes are biological systems that act as catalysts by speeding up a specific chemical reaction. Modifying these reactions by modifying the enzymes is what ibuprofen, for example, does. The ibuprofen molecule binds itself to the COX-2 enzyme and inhibits its activity. COX-2 is involved in prostaglandin and thromboxane synthesis, which mediate inflammation, pain, fever, and swelling. Thus, the inhibition of COX-2 activity decreases the synthesis of prostaglandins and its effects. [4]

2.2 GaudiMM

The framework in which this work takes place is GaudiMM [5], a software designed to explore multidimensional molecular spaces. The aforementioned evolutionary algorithm is the one GaudiMM actually uses to find binding sites. It has two key aspects that differentiate it from other similar programs: in addition to a docking scoring function (it uses Vina-based [6] methods) the program also uses other geometry-based evaluators, such as distance or rotation between atoms, or the volume occupied by the bonded molecule. The use of several evaluators is explained by its second particularity: it is designed to be able to face different scenarios and have a general design. While other software are specialised in certain types of reactions for which they compute pre-calculations and optimize their code, GaudiMM cannot do this. That is the reason why trying to accelerate the convergence of its algorithm via deep learning, the objective of this work, is a meaningful idea.

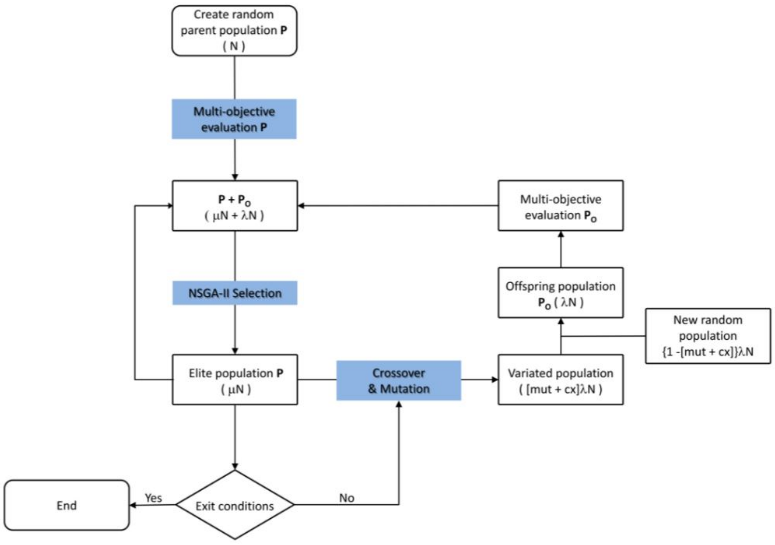
Create random parent population P (N)
Multi-objective evaluation P
P + P_o (μN + λN)
Multi-objective evaluation P_o
NSGA-II Selection
Offspring population P_o (λN)
New random population {1 -[mut + cx]}λN
Elite population P (μN)
Crossover & Mutation
Variated population ([mut + cx]λN)
End
Yes
Exit conditions
No

Figure 2: Image and note adapted from [3]. Workflow of a GaudiMM calculation. N is the number of individuals in the initial population P. μ is related to the number of individuals selected for the following generation. λ is related to the number of individuals produced at each generation as offspring (population P_0). The parameters mut and cx are the probabilities associated with mutation and crossover operators, respectively.

3 Methodology

Throughout the conducting of this project, a weekly meeting with the tutor has been done in order to guide the work. There is communication with José Emilio Sánchez Aparicio and Jean-Didier Maréchal, author and director of a GaudiMM-based PhD thesis respectively, whom proposed the idea of the project. This meeting schedules will continue during all the project.

3.1 Progress done

The first thing that had to be done was the study and analysis of all the biochemical context of the work, especially of the thesis [3]. After having understood the basics of how protein binding works, the next step was to study the GaudiMM2 code, an alpha version of an update of GaudiMM which if the result is positive, could include the deep learning-assisted convergence acceleration. Within this analysis, the focus has been on how does a normal execution of the objective algorithm work. In order to be able to analyse it, the logs the program produces have been reformatted and printed, which has allowed the possibility to plot them.

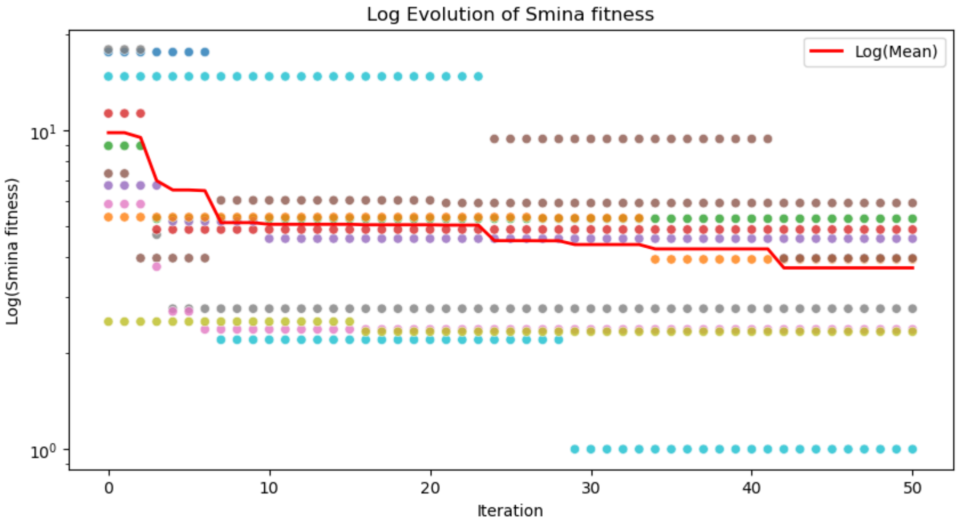

Figure 3: Plot of the evolution in log scale of the fitness value from an execution of the GaudiMM2 evolutionary algorithm. The execution had 10 individuals and 50 iterations. The evaluator used is Smina, a Vina-based docking scoring function. [7]

3.2 Planning

The immediate future plans are to analyse the functions that initialize, mutate and perform crossover over the population of the evolutionary algorithm to decide where would it be more beneficial to introduce a deep learning algorithm. In parallel to this, the scope of the deep learning algorithm has to be decided (one algorithm for every interaction, something more general...). Since this are key decisions to the project, further planning depends on them.

References

[1] E. Shem-Tov, M. Sipper, and A. Elyasaf, “Deep learning-based operators for evolutionary algorithms,” 2024. [Online]. Available: https://arxiv.org/abs/2407.10477

[2] Y. Song, Y. Wu, Y. Guo, R. Yan, P. N. Suganthan, Y. Zhang, W. Pedrycz, S. Das, R. Mallipeddi, and O. S. A. Q. Feng, “Reinforcement learning-assisted evolutionary algorithm: A survey and research opportunities,” 2024. [Online]. Available: https://arxiv.org/abs/2308.13420

[3] J. E. S. Aparicio, “Development and application of computational tools for the coupled exploration of chemical and biological spaces,” Ph.D. dissertation, Universitat Autònoma de Barcelona, 2022. [Online]. Available: https://uab-my.sharepoint.com/:b:/r/personal/2132358_uab_cat/Documents/Datos%20adjuntos/Contenido%20tesis%202.pdf?csf=1&web=1&e=IeyyW9

[4] O. DrugBank, “Aspirin (db01050),” 2025, accessed: 23 Mar. 2025. [Online]. Available: https://go.drugbank.com/drugs/DB01050

[5] J. Rodríguez-Guerra Pedregal, G. Sciortino, J. Guasp, M. Municoy, and J.-D. Maréchal, “Gaudimm: A modular multi-objective platform for molecular modeling,” *Journal of Computational Chemistry*, vol. 38, no. 24, pp. 2118–2126, 2017, code available at: https://github.com/insilichem/gaudi. [Online]. Available: http://dx.doi.org/10.1002/jcc.24847

[6] O. Trott and A. J. Olson, “Autodock vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading,” *Journal of Computational Chemistry*, vol. 31, no. 2, pp. 455–461, 2010, code available at: https://github.com/ccsb-scripps/AutoDock-Vina. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/jcc.21334

[7] D. R. Koes, M. P. Baumgartner, and C. J. Camacho, “Lessons learned in empirical scoring with smina from the csar 2011 benchmarking exercise,” *Journal of Chemical Information and Modeling*, vol. 53, no. 8, pp. 1893–1904, 2013, code available at: https://sourceforge.net/projects/smina/. [Online]. Available: https://doi.org/10.1021/ci300604z