Universitat Autònoma de Barcelona

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

Progress Report

Student:
Arnau Solé Porta, 1630311

Tutor:
David Castells Rufas

March 2025

Contents

1 Introduction **2**

2 Objective **2**

3 Contextualization **2**
3.1 Usefulness 2
3.2 GaudiMM 3

4 Progress **5**
4.1 Dataset 8
4.2 Modifications 9

5 Methodology **9**
5.1 Planning 9

1 Introduction

The following report is based on the previous work submitted regarding the progress of the end of course work. The information has been updated and extended, and the objectives and methodology revised and confirmed. While there were no major errors on the previous report, there has been some rewording, as well as the addition of the Progress section.

2 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 their 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.

3 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 a biochemically feasible 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. Designing docking scoring functions, and in general designing any method used to evaluate 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.

3.1 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]

3.2 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 it uses to find binding sites. GaudiMM 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 aim. 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.

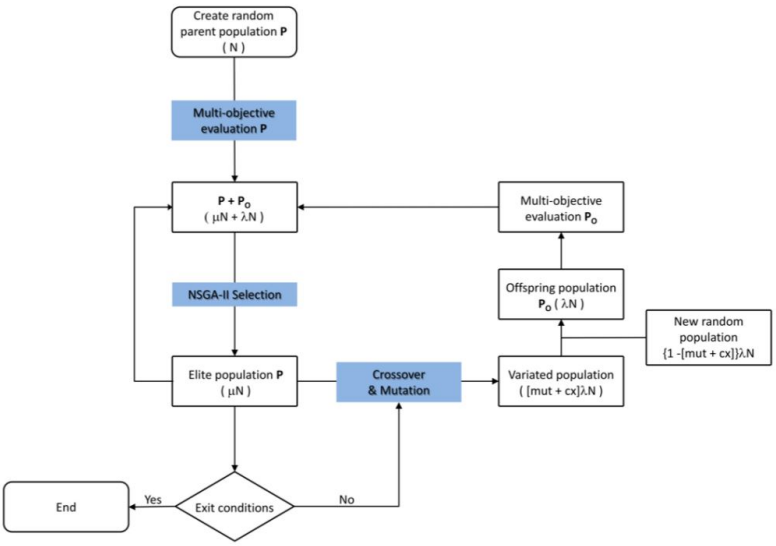

Figure 2: Image and note adapted from [3]. Workflow of a GaudiMM calculation. N is the number of individuals in the initial population P. μ is related to the number of individuals selected for the following generation. λ is related to the number of individuals produced at each generation as offspring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

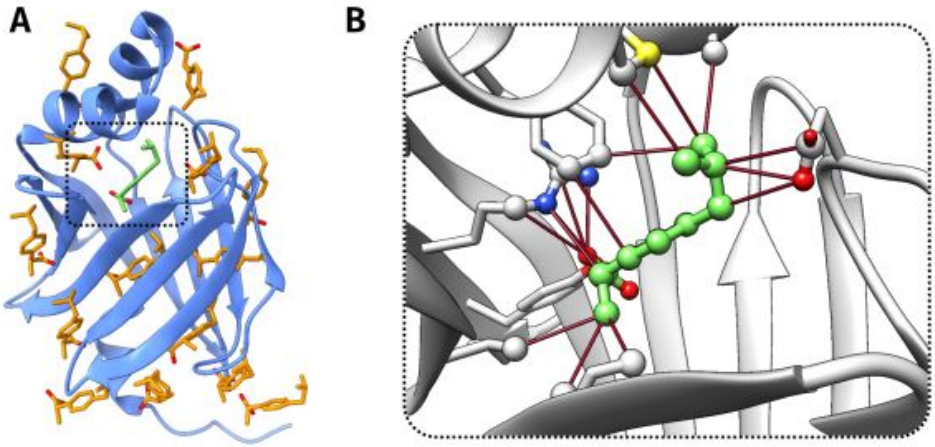

Figure 1: Binding example. Image and note adapted from [3]. Exploring the relative orientation of the ibuprofen molecule with respect to the FABP4 protein structure. (A) Multiple options are possible. In orange there are depicted several locations and orientations of the ibuprofen molecule. In green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

4 Progress

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 what protein binding is, the next step was to study the GaudiMM2 code, an alpha version of an update of GaudiMM that among other things, could end up including the deep learning-assisted convergence acceleration. Within this analysis, the focus has been on how does a normal execution of the evolutionary algorithm work. In order to be able to analyse it, the logs the program produces have been reformatted and saved, which has allowed the possibility to plot them. Four different visualizations have been designed: the evolution, both in normal and logarithmic scale, of the fitness score of all the individuals during an execution; the fitness values of the individuals at the end of the execution; the evolution of the values of the different genes during an execution; and a schema of the evolution of the location and rotation of each individual with regards to the protein during the execution. The evaluator used throughout all these testing has been Smina, a Vina-based docking scoring function. [7]

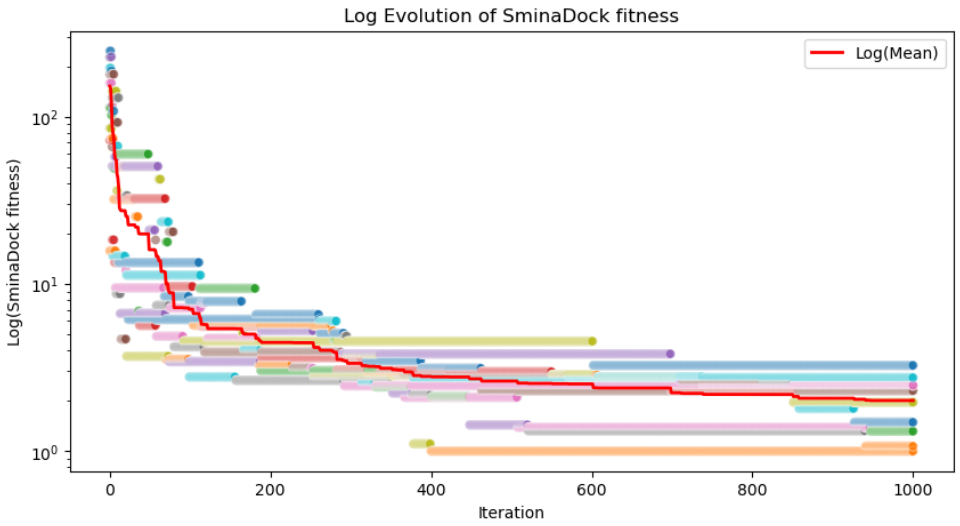

Figure 3: Plot of the evolution in log scale of the fitness value from an execution of the GaudiMM2 evolutionary algorithm. The execution had 10 individuals and 1001 iterations. Every individual has a different colour. Note how new individuals are created during the execution.

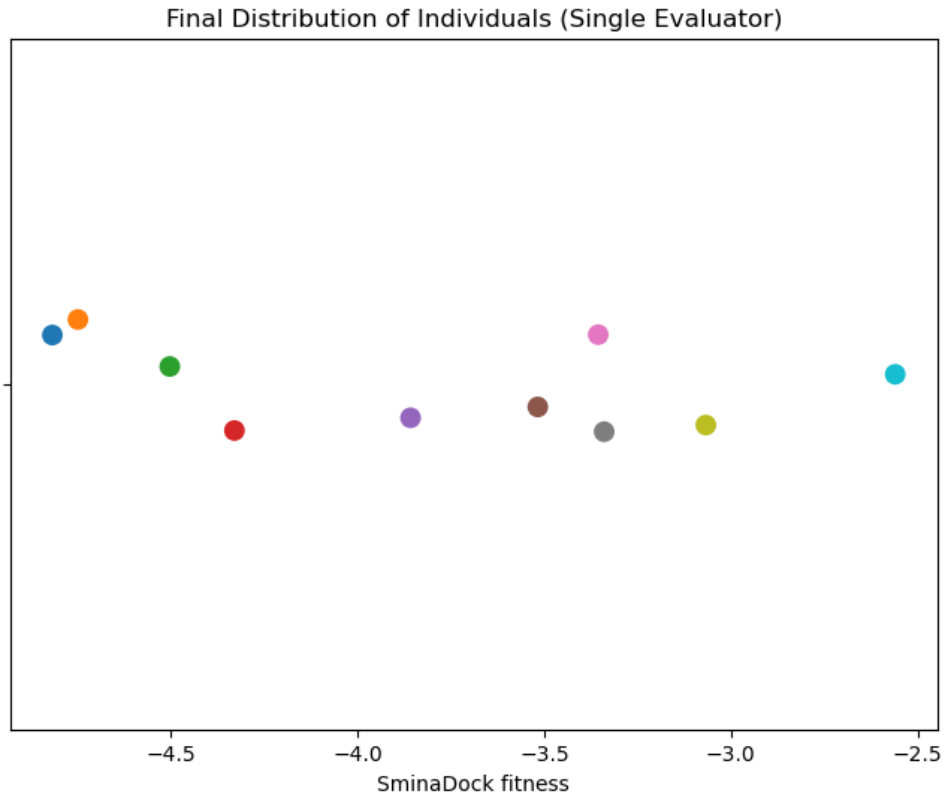

Figure 4: Plot of the final values after the execution of the GaudiMM2 evolutionary algorithm. The execution had 10 individuals and 1001 iterations.

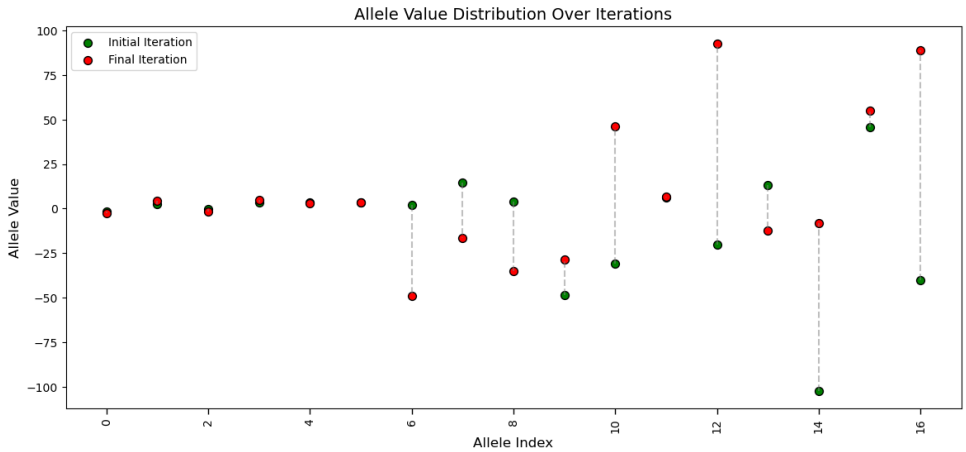

Figure 5: Every individual (molecule-protein position) has a set of genes (a gene can be any relevant feature, such as the position -x,y,z- in space or the rotation of the molecule) which is called allele. Plot of the evolution of the mean values of the alleles from an execution of the GaudiMM2 evolutionary algorithm. The execution had 10 individuals and 1001 iterations.

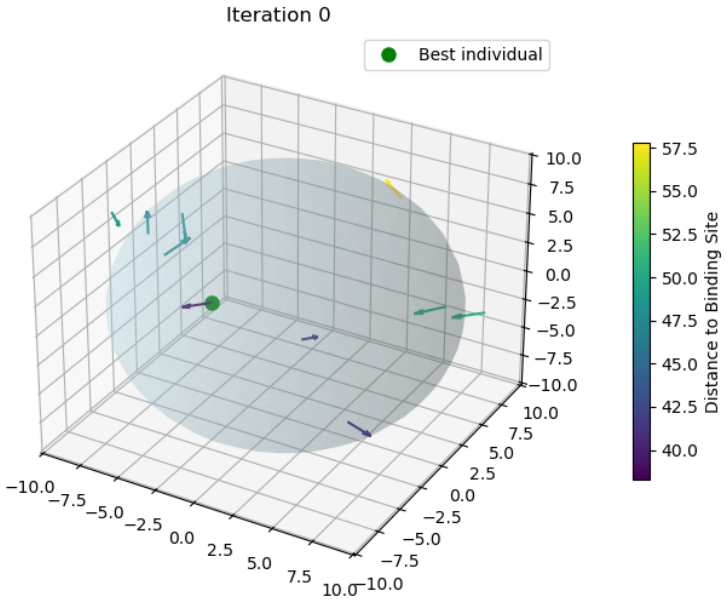

Figure 6: Plot of the spatial position and orientation of the individuals (represented as vectors) of an iteration from an execution of the GaudiMM2 evolutionary algorithm. The sphere represents the protein. The execution had 10 individuals and 1001 iterations.

4.1 Dataset

In order to develop the work, a dataset with 72 ligands (system consisting of a protein and a molecule binded) has been provided. From these, there are 5 ligands that have a defined target (optimal Smina fitness and location of the binding). These 5 are being used to perform executions and tests over the GaudiMM framework, since there is a clear evaluation of their performance. One of the insights obtained from analysing the dataset has been the fact that there is no way to generalize bindings. Proteins and molecules are terms that contain structures with an enormous diversity in composition and functionment, and while it is possible to generalise the quality of a binding site (which is done with fitness functions such as Smina [7]), it is not possible to generalise a strategy to find an acceptably good binding site (for example, the genetic algorithm that GaudiMM2 uses always begins from scratch, there are no pre-computed weights). It is true that there are groups of ligands that share common characteristics, but the generalist aim of GaudiMM2 makes it impossible to make use of them. The deep learning models designed in this work will have to be trained in a case-specific basis.

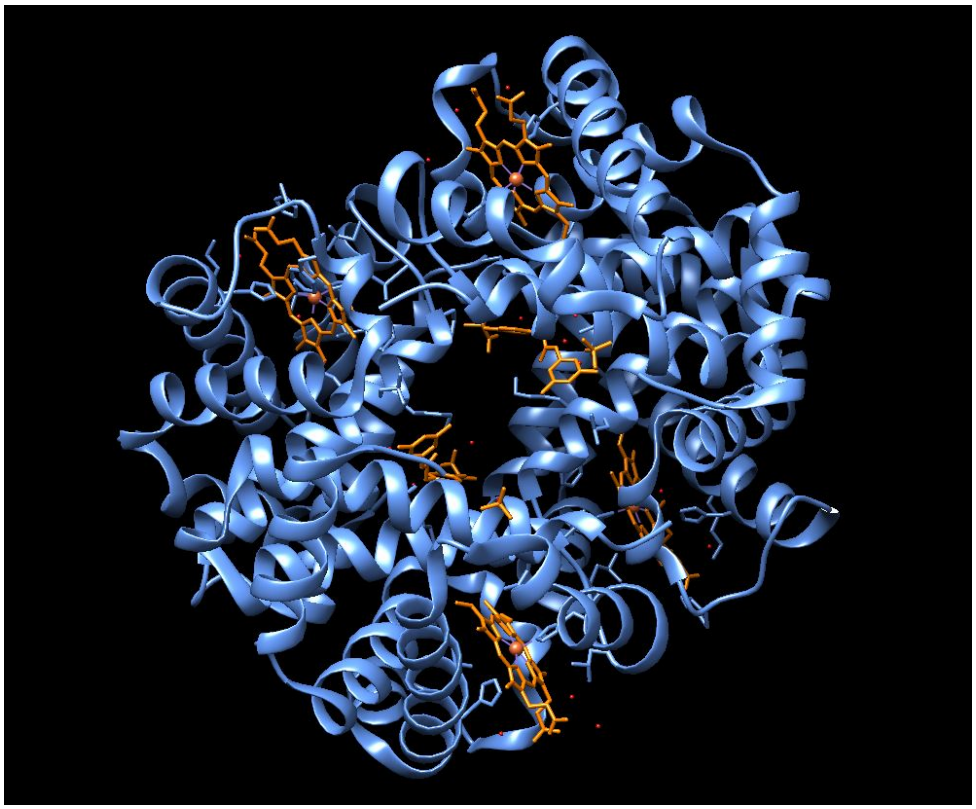

Figure 7: Plot in Chimera [8] of one of the 5 labelled ligands. It depicts various RSR-13 molecules (in orange) binding with Hemoglobin (protein that facilitates the transportation of oxygen in red blood cells, in blue). This union increases the oxygen delivery to tissues and its applications go from improving myocardial recovery to acting as a radiosensitizing agent in the treatment of brain tumours. [9] [10]

4.2 Modifications

The first modification to the code was the addition of a unique ID for every individual generated during an execution of GaudiMM2. This was achieved through the uuid4() function from the UUID Python module [11]. The motivation for this change was to increase the monitoring and logging of promising ligands. Finally, a new mutation function based on a deep learning model trained with policy gradient method is being developed. Since the weights of the model will be case-specific, there will be two versions: an online one that will train the model while the execution is running; and one where only inference will be done when the execution is running.

5 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.

5.1 Planning

The future plans are to finish the development of mutation and crossover (the two operations that modify individuals on the GaudiMM2 genetic algorithm) deep-learning based methods. Initially, they will be trained and then used on an execution of GaudiMM2. However, an online training is something that will also be explored. Finally, a sound testing environment that allows to assess the performance of the modifications to the genetic algorithm will be created.

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] Online DrugBank, "Aspirin (db01050)," 2025, accessed: 17 May 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

[8] Resource for Biocomputing, Visualization, and Informatics (RBVI), "Ucsf chimera: An extensive molecular modeling system," 2025, accessed: 17 May 2025. [Online]. Available: https://www.cgl.ucsf.edu/chimera/

[9] Rcsb Protein Data Bank, "Rcsb pdb - 1g9v: High resolution crystal structure of deoxy hemoglobin complexed with a potent allosteric effector." [Online]. Available: https://www.rcsb.org/structure/1G9V

[10] M. K. Safo, C. M. Moure, J. C. Burnett, G. S. Joshi, and D. J. Abraham, "High–resolution crystal structure of deoxy hemoglobin complexed with a potent allosteric effector," *Protein Science*, vol. 10, no. 5, p. 951–957, May 2001. [Online]. Available: https://doi.org/10.1110/ps.50601

[11] "Uuid objects according to rfc 4122." [Online]. Available: https://docs.python.org/3/library/uuid.html