A topological approach to protein stability

Carlos Ronchi[*], Konstantin Mischaikow[†], Marcio Gameiro[*]
carloshvronchi@gmail.com, chronchi, chronchi.github.io
[*] ICMC - USP, [†] Rutgers University

Motivation

The protein folding problem is the question of how a protein's amino acid sequence dictates its three-dimensional atomic structure. This question is fundamental since it can lead to new developments in drug research and how to combat diseases. However, it is a tough challenge to develop new stable protein structures given an amino acid sequence. There have been some recent studies in protein design that address the development of new molecules with the aid of software, such as *Rosetta*, a macromolecular structure modeling tool. This approach has some downsides, and often, the generated proteins are not stable, thus making it very expensive to check the stability of every modeled molecule experimentally. It is necessary to develop new methods to predict if the given protein is stable or not, so one can speed up the development of macromolecular structures, and it also helps to save money in doing fewer experiments to verify the stability.

Topological Data Analysis

Recently, a lot of data has been generated and it is becoming tougher to analyse all the gathered information in a concise and fast way. Therefore, the use of appropriate tools to analyse every piece of data is fundamental. Topological data analysis is an applied branch of algebraic topology with several algorithms to extract information from data. One of these algorithms is **persistent homology**, a topological summarizer [3].

Given a set of points $\{ x_1, \ldots, x_n \}$, we can visualize its most important topological features through the persistence diagram, a multiset in $\mathbb{R}^2$ summarizing the persistnece of holes, voids and cavities found in data through time. Figure 1 shows an example starting from the points, going to the filtration and getting the persistence diagram.

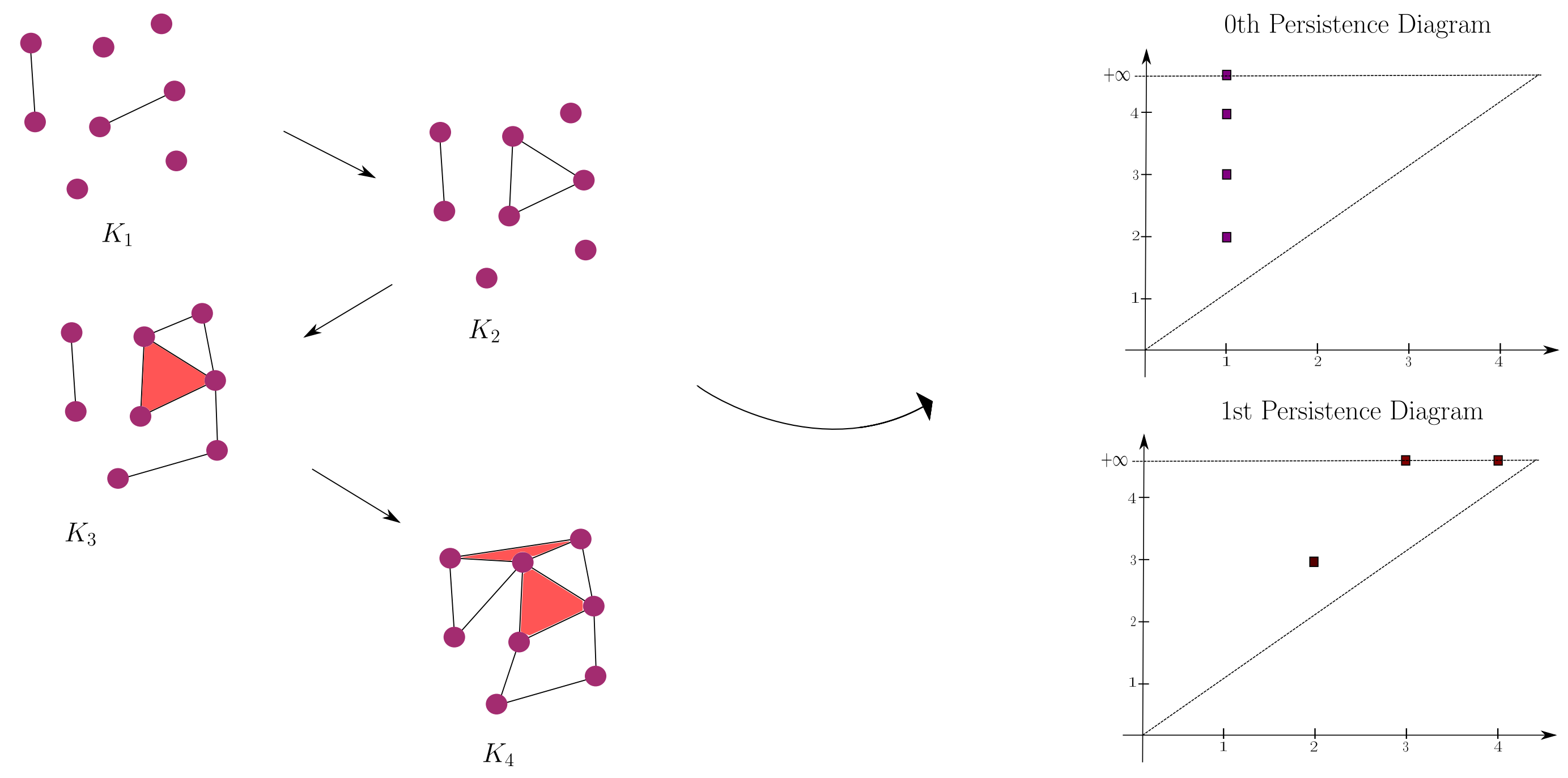


Figure 1: Steps to extract topological features from the data. The first step is to construct a filtration of the simplicial complex. Then we can select the cycles and represent their persistence in the persistence diagram.

Protein Stability

A protein is a 3D macromolecular structure obtained through a sequence of aminoacids and its folding. See Figure 2 for an scheme showing an example.

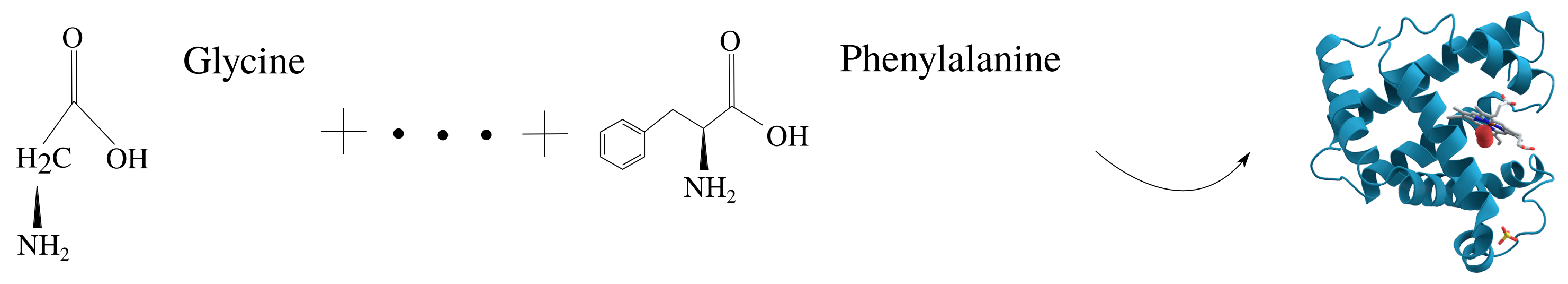
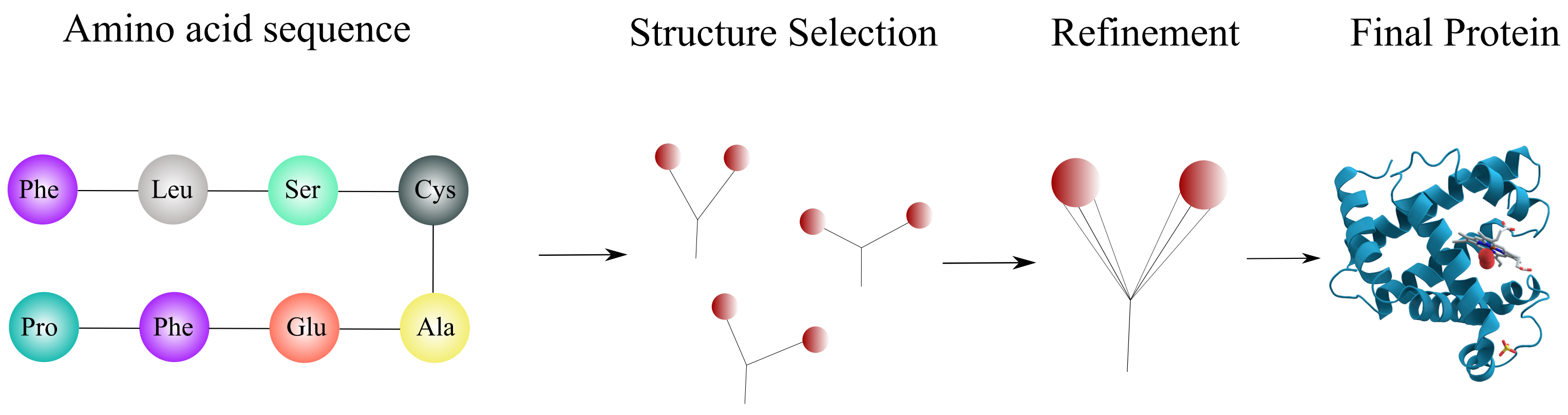


Figure 2: Folding from an aminoacid sequence to a protein.

In recent year, several people have been trying to study the fundamental problem of protein folding. We say that a protein is stable if it is hard to break using protease, enzymes that break peptide chains. It is crucial to analyse if a proposed molecule is stable or not. The problem is that it has been expensive to reproduce every molecule model in a laboratory. With that in mind, several computational approaches have been proposed [5], saving both time and money. The computational tools used and a pipeline of the process is shown in Figure 3.



Amino acid sequence

Structure Selection

Refinement

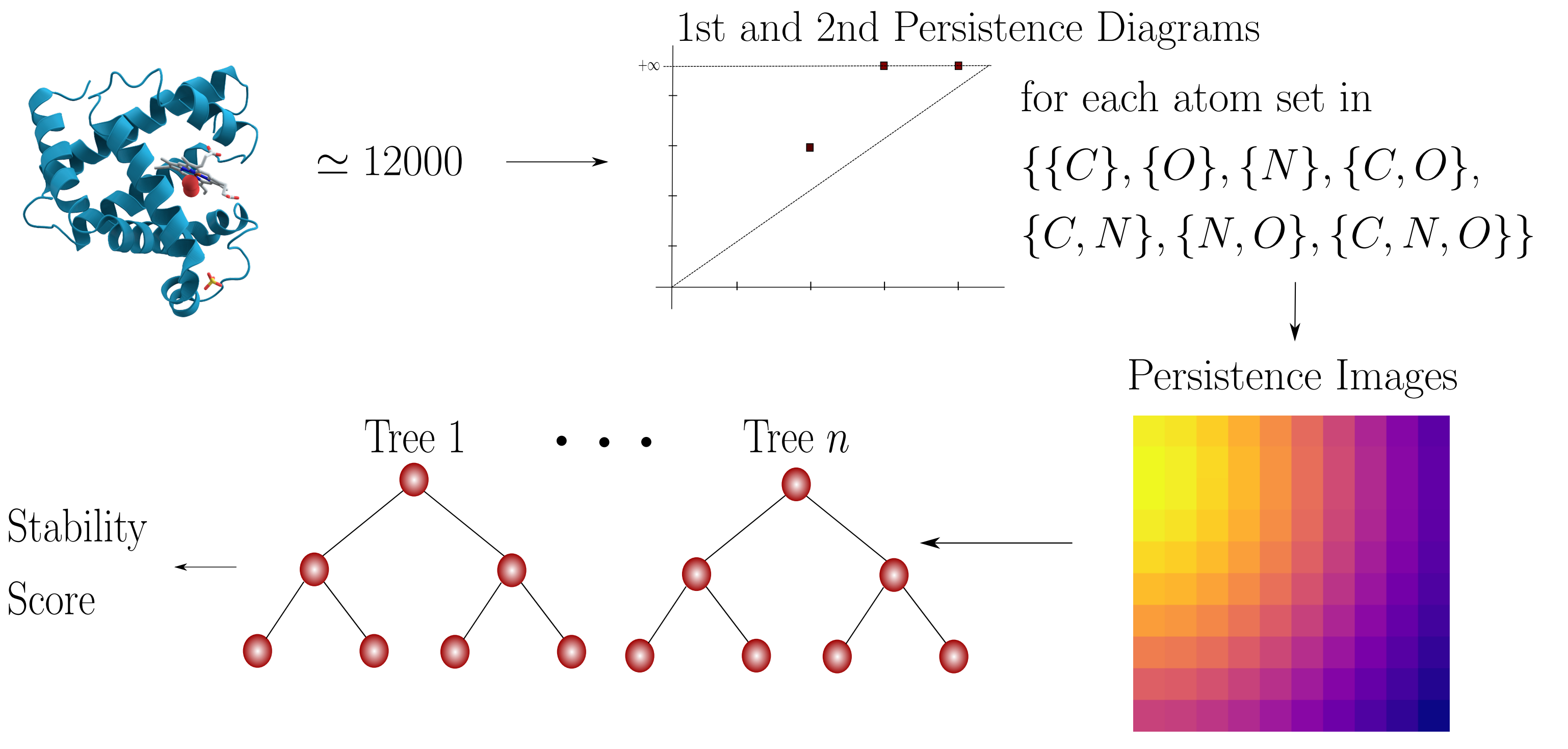
Final Protein

Figure 3: Steps in the protein designing when using computational methods.

How can we characterize the stability of a protein using persistent homology and predict its score?

Methodology

We predicted the stability score of a set of proteins. In order to do that we used persistence homology to extract several features from the molecules, vetctorized the respective persistence diagrams to persistence images [1] and used as input to several machine learning algorithms implemented in scikit-learn [4]. We calculated several persistence diagrams for each protein, where each PD corresponds to a subset of atoms in {{C}, {O}, {N}, {C,O}, {C,N}, {N,O}, {C,N,O}}. The pipeline is shown in Figure 4.



1st and 2nd Persistence Diagrams

$\simeq$ 12000

for each atom set in $\{\{C\},\{O\},\{N\},\{C,O\},$ $\{C,N\},\{N,O\},\{C,N,O\}\}$

Persistence Images

Tree 1 · · · Tree n

Stability Score

Figure 4 - Pipeline of the methodology.

Results

Model	RMSE	Percent Error (%)
Linear Regression	0.5046	13.69
Random Forest I	0.4877	13.24
Random Forest II	0.4874	13.23
GBoost Optimal	0.4770	12.95
Rocklin model[5]	0.419	11.381

Table 1 - Best results for persistence images with a 5 × 4 gridsize, spread 0.7. The last result used only 110 protein features and a random forest regressor.

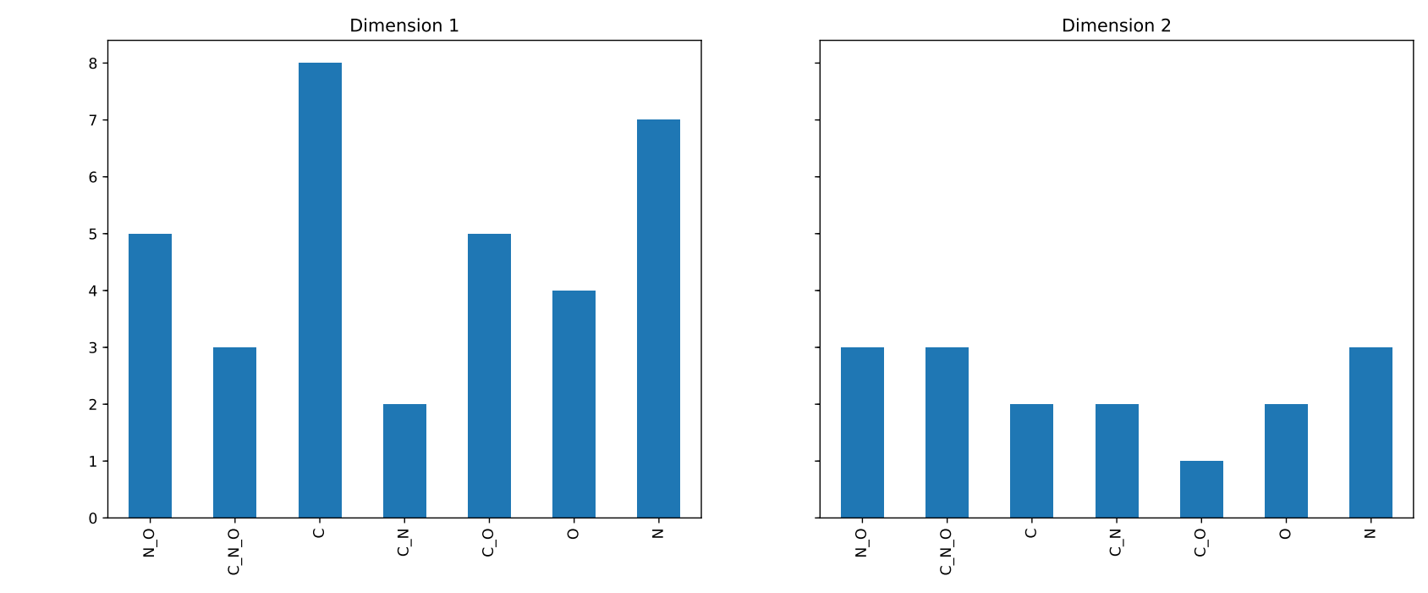


Figure 5 - Number of features per atom set. Persistence images calculated with only Carbon and Nitrogen atoms are the most frequent. According to [2] they represent hydrophobic and hydrophilic properties.

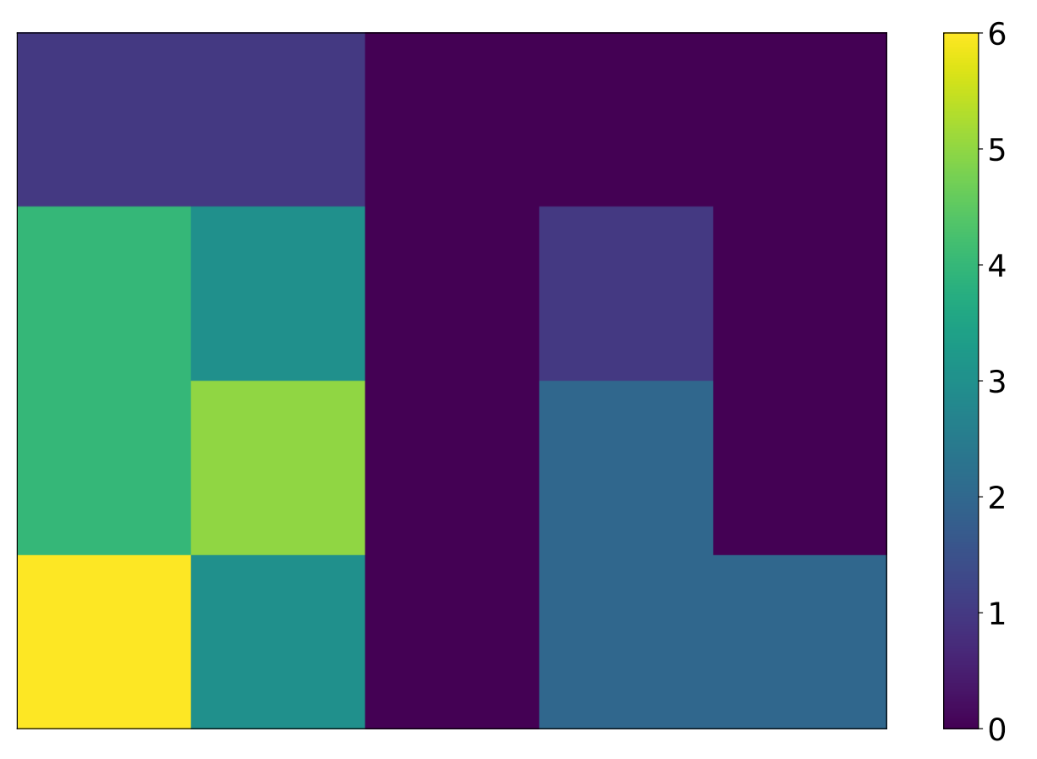


Figure 6 - Points in each bin in 1-dimensional persistence images

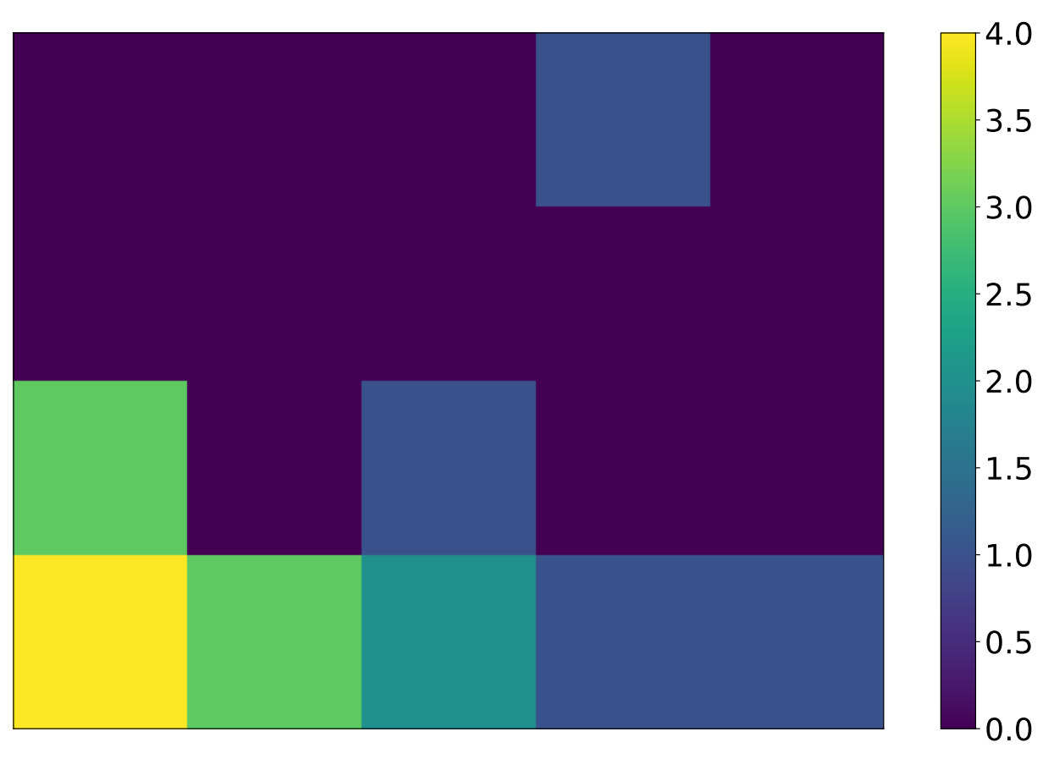


Figure 7 - Points in each bin in 2-dimensional persistence images

References

[1] Henry Adams et al. "Persistence Images: A Stable Vector Representation of Persistent Homology". In: *Journal of Machine Learning Research* 18.8 (2017), pp. 1–35.

[2] Zixuan Cang and Guo-Wei Wei. "Integration of element specific persistent homology and machine learning for protein-ligand binding affinity prediction". In: *International Journal for Numerical Methods in Biomedical Engineering* 34.2 (Aug. 2017). DOI: 10.1002/cnm.2914.

[3] Frédéric Chazal et al. *The Structure and Stability of Persistence Modules*. Springer International Publishing, 2016. DOI: 10.1007/978-3-319-42545-0.

[4] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[5] Gabriel J. Rocklin et al. "Global analysis of protein folding using massively parallel design, synthesis, and testing". In: *Science* 357.6347 (July 2017), pp. 168–175. DOI: 10.1126/science.aan0693.

Acknowledgments

We acknowledge FAPESP for the financial support in the projects #2017/14678-0 and #2018/20659-0 of student Carlos Ronchi. Also, the authors acknowledge the Office of Advanced Research Computing (OARC) at Rutgers, The State University of New Jersey for providing access to the Amarel cluster and associated research computing resources that have contributed to the results reported here. URL: http://oarc.rutgers.edu