Menger_Curvature : a MDAKit implementation to decipher the dynamics, curvatures and flexibilities of polymeric backbones at the residue level

Etienne Reboul [1*], **Jules Marien** [1*], **Chantal Prévost** [1], **Antoine Taly** [1¶], **and Sophie Sacquin-Mora** [1¶]

1 Laboratoire de Biochimie Théorique, Institut de Biologie Physico-Chimique, UPR9080, Université Paris-Cité, CNRS, France ROR ¶ Corresponding author * These authors contributed equally.

A more in-depth discussion is available in the Biorxiv preprint at https://www.biorxiv.org/content/10.1101/2025.04.04.647214v1

Background

New difficulties regarding the analysis of molecular dynamics (MD) simulations have emerged as MD becomes more and more involved in deciphering the behavior of highly flexible polymers such as Intrinsically Disordered Regions and Proteins (IDRs and IDPs). Characterization of their conformational ensemble (CoE) generally relies on global observables such as the radius of gyration. However, this global characterization cannot provide information on the local behavior of the backbone which participates in the IDP or IDR compactness and dynamics. Complex behaviors resulting from transient interactions and strong exposure to the solvent render powerless commonly used metrics such as the Root-Mean-Squared Fluctuations (RMSF). Past attempts to probe or to compare conformational ensembles have proven successful in extending the bioinformatic toolbox available for IDPs and IDRs, but often at the cost of computational speed and interpretability of the metrics (David & Jacobs, 2014; González-Delgado et al., 2022; Zhou et al., 2018). A successful attempt for a simpler flexibility metrics at the residue level was already made using Protein Blocks (PBs) by Barnoud et al with their Equivalent Number of PBs (Neq) (Barnoud et al., 2017). By statistically quantifying how many PBs are accessed by an amino acid along a simulation, the Neq can provide a value between 1 and 16 corresponding to low to high flexibility.

Statement of need

Although the PBs and Neq are adequate to describe the proteic backbone and its flexibility, they suffer from a few shortcomings. First, the PBs are specific to proteins and were designed from a statistical clustering of the Protein Data Bank, an ever-growing and ever-evolving repository of protein structures with a known biased towards folded proteins. Second, the main calculation tool PBxplore is not parallelized. Finally, PBs are specific to the proteic backbone and cannot be generalized to other polymers. We previously introduced an alternative metric, the Local Flexibilities (LFs), to successfully characterize the loss in flexibility induced by nP-collabs in multiphosphorylated peptides (Marien et al., 2024). They are based on a calculation of the Menger curvatures of the backbone at the residue level called in this context the Proteic Menger Curvatures (PMCs). Menger curvatures can also be used to obtain Local Curvatures (LCs) as the average curvature of a residue. To increase performances and ease-of-use of the PMCs, we propose a new implementation in the form of a MDAKit named Menger_Curvature, based on the widely-used MDAnalysis ecosystem. We first introduce formally the definition of a Menger curvature, before describing the package implementation.

Menger curvatures, Local FLexibilies (LFs) and Local Curvatures (LCs)

A visual schematics of the definition of the Menger curvature is available in Figure Figure 1. Consider a triangle. The circumcircle of this triangle (i.e. the circle passing through each of its summits) has a radius R, and its inverse is defined as the Menger curvature, named after mathematician Karl Menger (**?**). The curvature of a point in a polymer backbone can therefore be defined by selecting two other points of the backbone and calculating the resulting Menger curvature. We here focus on the context of proteins to demonstrate the use of such a metric. We define the PMC(n) by applying the Menger curvature to the triangle formed by the C_α of residue n and those of residues $n - s$ and $n + s$ where s is defined as the spacing. LC(n) is the average value of PMC(n) over all conformations, and LF(n) is the standard deviation over all conformations. The n subscript can be dropped for clarity and PMCs, LCs and LFs designate the contracted plural forms of the names. A Python module based on MDAnalysis (Michaud-Agrawal et al., 2011) was previously developed and can be accessed at : https://github.com/Jules-Marien/Articles/tree/main/_2024_Marien_nPcollabs/Demo_and_scripts_Proteic_Menger_Curvature. The aim of this implementation is to refine this code, parallelize it and implement it as a fully developed python package as an MDAKit (Alibay et al., 2023).

Further considerations on the choice of spacing s and the biological relevance of the PMCs, LFs and LCs can be accessed in the developed version of this paper (link above). We still provide a structural interpretation of the values of the PMCs in Figure Figure 2. An example notebook is provided at https://github.com/EtienneReboul/menger_curvature/blob/main/notebooks/jm01-tutorial.ipynb

Overview of the MDAKit implementation

The Menger_Curvature package contains the class MengerCurvature() which allows the calculation of the Menger curvatures of a single-chain polymer backbone with a spacing s. The function is compiled just-in-time thanks to the Numba package (**?**). If one wishes to run the calculation in parallel, the **.run_parallel()** function can be called. The number of workers can be defined with the **n_workers** variable in the call to **MengerCurvature**, otherwise the class will automatically set **n_workers** to the number of CPUs available on the machine minus 2. For simulation of F frames of a protein with N residues, the attribute results.curvature_array will have size (N − 2s,F), .local_curvatures and .local_flexibilities will have size N − 2s. A full example is available as a notebook in the documentation. The complete architecture of the resulting menger object is available in Figure Figure 3. A comparison of the performances between PBxplore and Menger_Curvature reveals that the latter outperforms the former by more than one order of magnitude, and is in the range of the RMSF (see Table 1).

	PMCs, LCs and LFs (serial)	PMCs, LCs and LFs (parallel)	PBs and Neq	RMSF
Tubulin	0.23 ± 0.00	0.19 ± 0.01	10.54 ± 0.17	1.21 ± 0.02
Tau protein	4.54 ± 0.50	2.84 ± 0.05	218.48 ± 1.17	22.96 ± 0.44

Table 1: Table comparing the computation time in seconds of the PMCs, Neq, and RMSF on a short tubulin trajectory (1000 frames, 450 Cas) and a long Tau protein trajectory (20000 frames, 441 Cas). The alignment time of all conformations on the reference was included in the runtime of RMSF as it is necessary for the calculation.

Menger_Curvature is available in the Python Package Index and on Github : https://github.com/EtienneReboul/menger_curvature

83 **Figures**

$$\text{Ca} \atop (n)$$

$$PMC = \frac{1}{R}$$
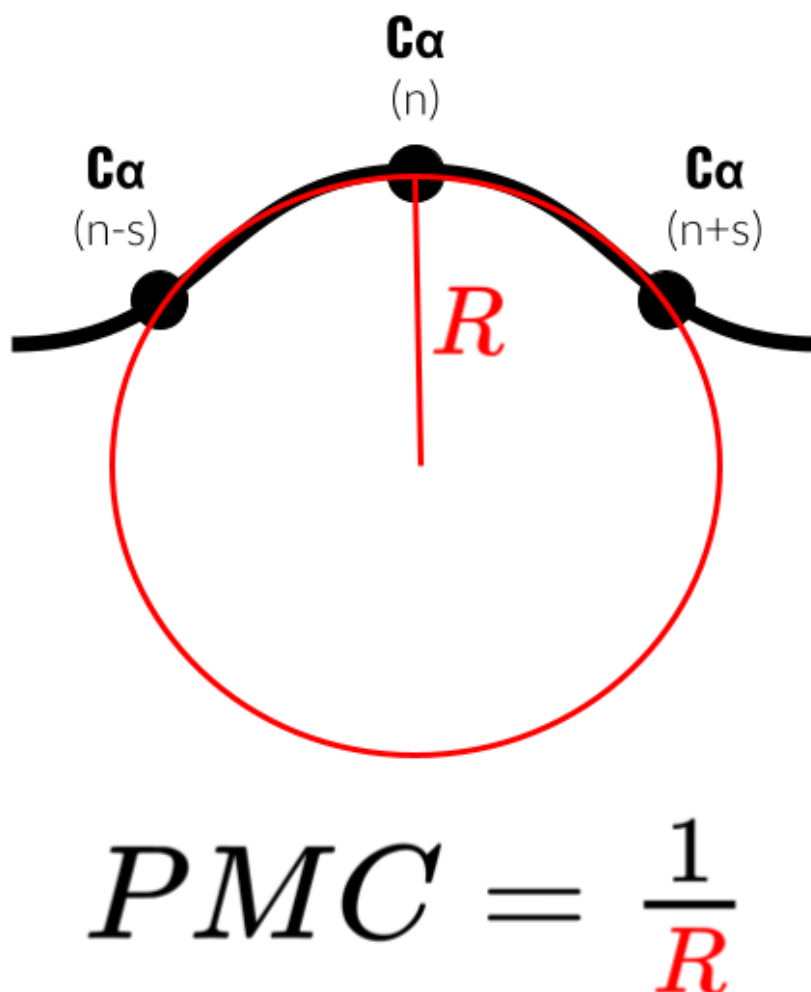
Figure 1: Schematic representation of the Proteic Menger Cuvarture (PMC) definition. The backbone of the polymer is in black, the C_αs are represented by spheres.
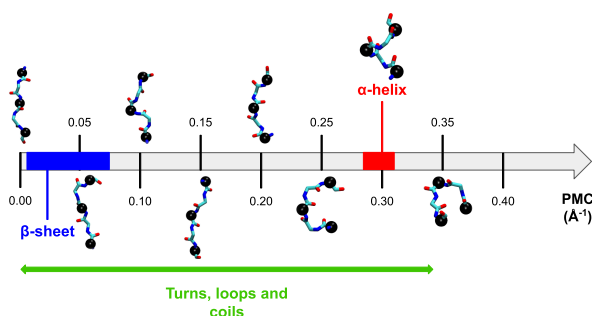
Figure 2: Range of PMC values and their associated structural elements. Backbone representations are extracted from the single chain tubulin simulation. Backbone is represented in licorice, C_αs involved in the PMC calculations are in black Van de Waals. Ranges for β-sheets and α-helices are based on the ranges of PMC values above 5% in Figure SI-1. These ranges are not meant to be statistically accurate but to provide the reader with a practical estimate of the values that can be expected.
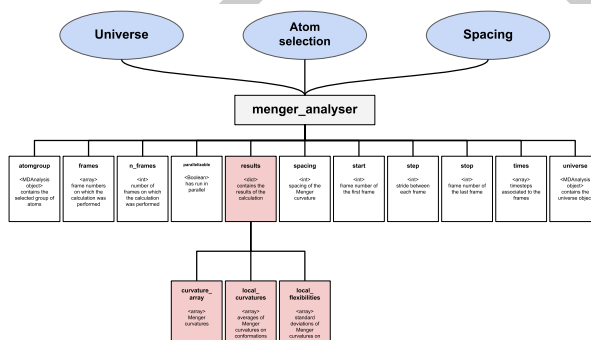
Figure 3: Architecture of the **menger_analyser** object obtained from the **MengerCurvature** class. Attribute names are in bold, their type is between chevrons followed by a brief description of their content. Required arguments are in blue, results are in red.

Acknowledgements

References

Alibay, I., Wang, L., Naughton, F., Kenney, I., Barnoud, J., Gowers, R., & Beckstein, O. (2023). *MDAKits: A framework for FAIR-compliant molecular simulation analysis*. 76–84. https://doi.org/10.25080/gerudo-f2bc6f59-00a

Barnoud, J., Santuz, H., Craveur, P., Joseph, A. P., Jallu, V., Brevern, A. G. de, & Poulain, P. (2017). PBxplore: A tool to analyze local protein structure and deformability with protein blocks. *PeerJ*, *5*, e4013. https://doi.org/10.7717/peerj.4013

David, C. C., & Jacobs, D. J. (2014). Principal component analysis: A method for determining the essential dynamics of proteins. *Methods in Molecular Biology (Clifton, N.J.)*, *1084*, 193–226. https://doi.org/10.1007/978-1-62703-658-0_11

González-Delgado, J., Sagar, A., Zanon, C., Lindorff-Larsen, K., Bernadó, P., Neuvial, P., & Cortés, J. (2022). *WASCO: A wasserstein-based statistical tool to compare conformational*

ensembles of intrinsically disordered proteins. bioRxiv. https://doi.org/10.1101/2022.12.01.518687

Marien, J., Prévost, C., & Sacquin-Mora, S. (2024). nP-collabs: Investigating counterion-mediated bridges in the multiply phosphorylated tau-R2 repeat. *Journal of Chemical Information and Modeling*. https://doi.org/10.1021/acs.jcim.4c00742

Michaud-Agrawal, N., Denning, E. J., Woolf, T. B., & Beckstein, O. (2011). MDAnalysis: A toolkit for the analysis of molecular dynamics simulations. *Journal of Computational Chemistry*, *32*(10), 2319–2327. https://doi.org/10.1002/jcc.21787

Zhou, H., Wang, F., & Tao, P. (2018). T-distributed stochastic neighbor embedding (t-SNE) method with the least information loss for macromolecular simulations. *Journal of Chemical Theory and Computation*, *14*(11), 5499–5510. https://doi.org/10.1021/acs.jctc.8b00652