

DynDom1D_Python: An Open-Source Python tool for the analysis of domain movements in proteins

Jien Lim¹, Hugh Millard¹, and Steven Hayward¹¶

¹ School of Computing Sciences, University of East Anglia, Norwich NR4 7TJ, U.K. ¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: ↗

Submitted: 30 July 2025

Published: unpublished

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#))

Summary

DynDom1D_Python is an open-source Python implementation based on the original DynDom Fortran program for the analysis of domain movements in proteins. DynDom1D works on single protein chains and can be used when two structures of the same protein are available representing a conformational change. If appropriate it describes the conformational change in terms of the relative rotations of quasi-rigid regions called “dynamic domains” by way of hinge axes and hinge-bending residues. It improves over the original standalone version in a number of ways with an input file that requires only the protein data bank (PDB) accession codes and the chain identifiers of the two structures. The output has also been updated to produce a PyMol script file for display and animation of the domain movement.

Statement of need

Protein function usually involves interaction with other molecules, this interaction often effecting a change in conformation. Proteins are macromolecules that usually comprise many tens of thousands of atoms and conformational change occurs over a vast spatial scale. On the global scale is the domain movement where each domain comprises a significant portion of the whole protein. Protein domain movements form an important class of movements in proteins, perhaps the most important, and they are often engaged in all aspects of protein function, e.g. an enzyme trapping of a small substrate in its interdomain cleft. In principle, understanding how domain movements are controlled during function may be useful for drug design, e.g. by designing a drug to bind to a hinge-bending region in order to disrupt a functional hinge-bending movement.

The original Fortran implementation of the DynDom program (Steven Hayward & Berendsen, 1998; S. Hayward & Lee, 2002) is available to download and run at the DynDom webserver (Lee et al., 2003) and to download from Collaborative Computational Project 4 (CCP4) (Agirre et al., 2023), where it has been used by protein X-ray crystallographers since 1999. Although versions of DynDom for the analysis of multimeric proteins (comprising multiple chains) have been released, namely DynDom3D (Poornam et al., 2009) and DynDom6D (Veevers & Hayward, 2019), it is the 1D version (1D indicating that it analyses a single protein chain) that remains the most popular. The need is apparent for an updated open-source version of DynDom1D with a less restrictive licence and output suitable for visualisation with current molecular graphics tools.

Method

The basic idea behind the DynDom approach and related approaches (Hinsen et al., 1999; Wriggers & Schulten, 1997) is that domains can be recognised as regions that move as quasi-

rigid bodies. DynDom1D reads two protein chain structures that represent a conformational change. These structures can be experimentally determined using methods such as X-ray crystallography, Nuclear Magnetic Resonance Spectroscopy, or cryo-electron microscopy. The two structures can also represent a set of atomic displacements derived from computational methods such as normal mode analysis or molecular dynamics simulation. DynDom is a coarse-graining method that aims to model, when appropriate, atomic displacements in terms of the relative screw movements of a small number of “dynamic domains”, their relative movement being facilitated and controlled by flexible interdomain bending regions. The process comprises the following steps:

1. Read in the two structures as specified by the PDB accession codes and chain identifiers.
2. Superpose the two structures using the main-chain atoms N, CA, and C.
3. Use a sliding window to generate overlapping main-chain segments.
4. Determine “rotation vector” of each main-chain segment plotting components as coordinates in the “rotation space”.
5. Use K-means clustering to determine clusters of rotation points indicating regions possibly rotating within a rigid body.
6. Use of connected set algorithm to determine if regions from each cluster are spatially connected.
7. Split regions from each cluster into dynamic domains (must have at least as many residues as the minimum domain size “domain”) if they are not spatially connected.
8. Check that that ratio of external to internal displacement of all dynamic domains that are connected through the main-chain is greater than the “ratio” threshold (default, 1.0).
9. If the ratio test is passed for all, determine screw axes.
10. Determine interdomain bending regions.
11. Assign mechanical hinges and percentage “closure”.
12. Output results including PDB files for display with PyMol (Schrödinger & DeLano, 2020) using a Pymol script file (.pml file).

Further details on the basic methodology can be found in the original DynDom publications (Steven Hayward & Berendsen, 1998; S. Hayward & Lee, 2002). This new open source release include improvements made in the development of DynDom3D and DynDom6D. Basic functions such as parsing PDB files and superposition are performed using GEMMI (Wojdyr, 2022) and clustering of rotation vectors uses scikit-learn’s (Pedregosa et al., 2011) kmeans clustering method. DynDom1D is the only tool that is able to assign interdomain bending regions. An important feature is its ability to assign “mechanical hinges”. These are defined as interdomain bending regions (more commonly, but more loosely referred to as “hinge-bending” regions) that are close to (within 5.5 Å) the interdomain screw axis (Steven Hayward, 1999) (more commonly, but more loosely referred to as the “hinge-axis”). In order to properly control a domain movement, two or more mechanical hinges are separated in space determining the direction and location of the screw axis. Often such arrangements create a “closure” motion whereby the interdomain cleft narrows or widens depending of the direction of rotation. DynDom1D_Python assigns mechanical hinges, and the percentage closure motion (0% closure means a pure (100%) twisting motion) and also gives a point on the screw axis and its direction in terms of a unit vector.

Example Result - Adenylate Kinase

Figure 1 shows the results of the domain movements that occur in the protein adenylate kinase upon binding of the small molecules ADP and AMP. It shows how the protein closes up upon these two molecules. Details regarding these movements are given in the “info” file where it states that the rotations are 53 degrees for the red moving domain, and 46 degrees for the yellow moving domain. Both motions are controlled via two separated sets of mechanical hinge residues and are closure motions (99% and 97%, respectively).

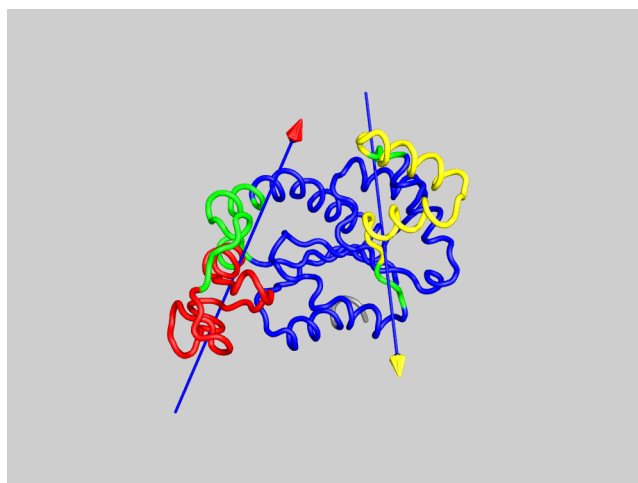


Figure 1: DynDom1D_Python result visualised using PyMol, for the movement between 4AKE(A) and 2ECK(B), the free and ADP,AMP bound structures of adenylate kinase. The “fixed” domain in the centre is coloured blue and the two “moving” domains are coloured red and yellow. The hinge bending regions are coloured green (DynDom always colours bending regions green) and the interdomain screw axes are depicted as arrows, with the shaft coloured the same as the fixed domain and the arrow head indicating the associated moving domain.

References

- Agirre, J., Atanasova, M., Bagdonas, H., Ballard, C. B., Basle, A., Beilsten-Edmands, J., Borges, R. J., Brown, D. G., Burgos-Marmol, J. J., Berrisford, J. M., Bond, P. S., Caballero, I., Catapano, L., Chojnowski, G., Cook, A. G., Cowtan, K. D., Croll, T. I., Debreczeni, J. E., Devenish, N. E., ... Yamashita, K. (2023). The CCP4 suite: Integrative software for macromolecular crystallography. *ACTA CRYSTALLOGRAPHICA SECTION D-STRUCTURAL BIOLOGY*, 79(6), 449–461. <https://doi.org/10.1107/S2059798323003595>
- Hayward, Steven. (1999). Structural principles governing domain motions in proteins. *Proteins: Structure, Function, and Bioinformatics*, 36(4), 425–435. [https://doi.org/10.1002/\(SICI\)1097-0134\(19990901\)36:4%3C425::AID-PROT6%3E3.0.CO;2-S](https://doi.org/10.1002/(SICI)1097-0134(19990901)36:4%3C425::AID-PROT6%3E3.0.CO;2-S)
- Hayward, Steven, & Berendsen, H. J. C. (1998). Systematic analysis of domain motions in proteins from conformational change: New results on citrate synthase and T4 lysozyme. *Proteins: Structure, Function, and Bioinformatics*, 30(2), 144–154. [https://doi.org/10.1002/\(SICI\)1097-0134\(19980201\)30:2%3C144::AID-PROT4%3E3.0.CO;2-N](https://doi.org/10.1002/(SICI)1097-0134(19980201)30:2%3C144::AID-PROT4%3E3.0.CO;2-N)
- Hayward, S., & Lee, R. (2002). Improvements in the analysis of domain motions in proteins from conformational change: DynDom version 1.50. *JOURNAL OF MOLECULAR GRAPHICS & MODELLING*, 21(3), 181–183. [https://doi.org/10.1016/S1093-3263\(02\)00140-7](https://doi.org/10.1016/S1093-3263(02)00140-7)
- Hinsen, K., Thomas, A., & Field, M. (1999). Analysis of domain motions in large proteins. *PROTEINS-STRUCTURE FUNCTION AND BIOINFORMATICS*, 34(3), 369–382. [https://doi.org/10.1002/\(SICI\)1097-0134\(19990215\)34:3%3C369::AID-PROT9%3E3.3.CO;2-6](https://doi.org/10.1002/(SICI)1097-0134(19990215)34:3%3C369::AID-PROT9%3E3.3.CO;2-6)
- Lee, R., Razaz, M., & Hayward, S. (2003). The DynDom database of protein domain motions. *BIOINFORMATICS*, 19(10), 1290–1291. <https://doi.org/10.1093/bioinformatics/btg137>
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., & others. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct), 2825–2830.
- Poornam, G. P., Matsumoto, A., Ishida, H., & Hayward, S. (2009). A method for the analysis of domain movements in large biomolecular complexes. *PROTEINS-STRUCTURE FUNCTION*

- 118 *AND BIOINFORMATICS*, 76(1), 201–212. <https://doi.org/10.1002/prot.22339>
- 119 Schrödinger, L., & DeLano, W. (2020). *PyMOL* (Version 2.4.0). <http://www.pymol.org/pymol>
- 120 Veevers, R., & Hayward, S. (2019). Methodological improvements for the analysis of domain
121 movements in large biomolecular complexes. *BIOPHYSICS AND PHYSICOBIOLOGY*, 16,
122 328–336. https://doi.org/10.2142/biophysico.16.0/_328
- 123 Wojdyr, M. (2022). GEMMI: A library for structural biology. *Journal of Open Source Software*,
124 7(73), 4200. <https://doi.org/10.21105/joss.04200>
- 125 Wriggers, W., & Schulten, K. (1997). Protein domain movements: Detection of rigid domains
126 and visualization of hinges in comparisons of atomic coordinates. *PROTEINS-STRUCTURE*
127 *FUNCTION AND BIOINFORMATICS*, 29(1), 1–14.

DRAFT