ARPHA Preprints

PENSOFT

Project Report

Author-formatted document posted on 05/12/2023

Published in a RIO article collection by decision of the collection editors.

DOI: https://doi.org/10.3897/arphapreprints.e116668

A Jupyter notebook to explore protein conformations: An output of PaNOSC SP8 case

Irina Safiulina, Miguel Angel Gonzalez, Paolo Mutti, Giuseppe La Rocca, Enol Fernández

A Jupyter notebook to explore protein conformations: An output of PaNOSC SP8 case

Authors: Safiulina, i., Gonzalez›, M., Mutti, P., La Rocca, G. Fernández, E.

Abstract

Small-angle scattering techniques are widely employed across scientific communities to elucidate the morphology, spatial distribution, and uniformity of particles within liquid solutions. Recent advancements and accelerated data acquisition capabilities have extended the utility of these techniques to probe the dynamic behavior of particles over time. Small-Angle X-ray Scattering (SAXS) and Small-Angle Neutron Scattering (SANS) emerge as powerful tools in this realm, enabling the investigation of phenomena such as the time-dependent release of genomes from phages, the comprehensive exploration of viral life cycles, and the assembly dynamics of macromolecular complexes. This tandem application of X-rays and neutrons, exemplified by the collaborative SANS-SAXS initiative between the European Synchrotron Radiation Facility (ESRF) and the Institut Laue-Langevin (ILL), yields complementary insights into complex infection pathways.

In this report, we present the outcomes of a software development project undertaken as part of Science Project 8: "PaNOSC Dynamics of Biological Processes" within the EOSC Future framework. The project's deliverables include a Jupyter Notebook and accompanying Python scripts, specifically sas_helper.py. The primary objective of Science Project 8 is to expand the accessibility of open Small-Angle Neutron Scattering (SANS) and Small-Angle X-ray Scattering (SAXS) data to a broader scientific community. This endeavor underscores the profound transformative potential of open-science data analysis within the European Open Science Cloud (EOSC) framework while advocating for the adoption of FAIR (Findable, Accessible, Interoperable, and Reusable) principles in scientific research.

Keywords

EOSC Future; Science Clusters; Science Projects; PaNOSC; Small angle neutron scattering (SANS); Small angle x-ray scattering (SAXS); Jupyter Notebook

Introduction

Small-Angle Scattering (SAS) techniques have emerged as invaluable instruments in the exploration of protein structure and dynamics within solution environments. These techniques furnish researchers with pivotal insights into the dimensions, conformational flexibility, and intermolecular interactions inherent to biological macromolecules.

At the heart of SAS methodologies lies the meticulous analysis of scattering patterns generated when a beam of X-rays or neutrons interacts with a sample. Through the interpretation of these intricate scattering profiles, researchers can unravel the intricacies of protein structures and other biomolecular entities. The moniker "small-angle" pertains to the scattering angles typically observed in SAS experiments, which manifest as deviations only slightly divergent from the incident beam direction. Within this regime of small angles, the intensity of scattered radiation serves as a treasure trove of information, unveiling the overarching morphology and size of the particles under scrutiny.

However, the journey from experimental data to meaningful insights encompasses several essential steps, including data reduction, modeling, and fitting. Notably, the data reduction process is a multifaceted challenge, contingent upon the nature of the radiation employed (X-rays or neutrons), the specific instrument configurations available within a facility, and even individual scientist preferences in data handling methods. Given the intricacies involved, automating this step comprehensively remains an elusive goal, particularly within the temporal constraints of a science project.

The subsequent phases of modeling and fitting, on the other hand, offer fertile ground for intervention and optimization. A plethora of commercial and open-source software tools have emerged to facilitate these crucial aspects of data analysis, encompassing notable solutions like ATSAS [Manalastas-Cantos et al. 2021], the IMP library [Schneidman- Duhovny et al. 2016], Inria library [ex. Hoffmann and Grudinin 2017, Grudinin et al. 2017] , SASView [SasView 2020], and more. In the pursuit of streamlining and democratizing the data analysis process, this project aspires to consolidate a selection of these software resources into a single, user-friendly hub. By doing so, both novice and experienced researchers can sidestep the need to install disparate software packages independently, sparing themselves the ordeal of mastering various command-line interfaces.

Moreover, the incorporation of Python libraries such as Matplotlib [Caswell et al. 2023] empowers users to effortlessly craft informative visualizations for scientific publications, reports, and presentations. In pursuit of these objectives, the project has chosen the Jupyter Notebook [Kluyver 2016] as an intuitive and accessible interface to facilitate data analysis endeavors. In the realm of modeling, the project capitalizes on open-source resources like the IMP [Schneidman-Duhovny et al. 2016] and Inria [Grudinin et al. 2017, Hoffmann and Grudinin 2017] libraries, offering comprehensive tools for data modeling and fitting.

The project builds also on the existence of two large, public and well-maintained databases: The Protein Data Bank (PDB) [Berman 2000], providing the structure of several hundreds of thousands of proteins and biomolecules, and the Small Angle Scattering Biological Data Bank [Kikhney et al. 2019], a repository of validated small angle scattering curves, fully corrected. But it should be stressed that it is not limited to those, and users can equally use protein structures and SAS data obtained from other sources, including their own simulations and/or experiments.

In sum, this project aspires to empower the scientific community by harnessing the capabilities of SAS techniques and consolidating essential software resources, thereby catalyzing breakthroughs in the study of protein dynamics and structure within solution environments.

Description

In this description section, we will provide an overview of the key components to be presented in this project. The section will be structured into three main subsections: "Methods", "Accessibility", and "Project Workflow".

In the "Methods" subsection, we will delve into the fundamentals of SAXS and SANS techniques. This segment will elucidate the distinctive features and inherent advantages of SAXS and SANS in the context of studying biological macromolecules and their dynamics in solution.

The "Accessibility" subsection will guide users through the process of gaining access to the project's routines and resources. It will commence with instructions on logging in using the UmbrellaID authentication system, followed by a detailed exploration of the platform hosted on the EGI infrastructure. Within this framework, we will provide explicit instructions for leveraging the Jupyter Notebook interface. Furthermore, we will outline the necessary steps for configuring the required bindings to access essential Python files and routines for seamless data analysis. This section will serve as a comprehensive guide for researchers, ensuring that they can readily engage with and harness the project's tools and resources.

In the "Project Workflow" subsection, we will present a systematic and versatile scientific pathway encapsulated within the SASHelper.ipynb Jupyter Notebook. This section will provide a structured overview of the key stages that users can navigate to achieve their research objectives.

Methods

SAXS and SANS Measurements on Protein and Macromolecules in Solutions

SAXS and SANS are instrumental techniques in elucidating the structural and dynamic characteristics of proteins and macromolecules within solution environments. These techniques provide a unique opportunity to explore not only the static properties but also the dynamic behaviors of these biomolecules. Unlike traditional methods like X-ray crystallography, which typically require crystallized samples and provide static snapshots of protein structures, SAXS and SANS enable the study of molecules in their native, solution- phase conditions. Consequently, researchers can observe different conformations of proteins, reflecting their inherent flexibility and dynamics.

In the following we give a very simplified description of the theory and practice of small angle scattering. Interested readers can check the abundant bibliography, e.g. [Feigin et al. 1987].

In a typical SAXS or SANS experiment, we will measure the fraction of the incident beam scattered by our sample in different directions. In the case of solutions, the scattering will be isotropic, so often one simply measures $I(2\theta)$, i.e. the intensity scattered as a function of the angle between the incident and output beams. After the appropriate data corrections, we obtain $I(Q)$, where Q is the momentum transfer:

$$Q = \frac{4\pi}{\lambda}\sin(\theta),$$

where λ is the wavelength of the incident probe. The measured intensity is related to the characteristics of the sample by the following expression:

$$I(Q) = NI_0(\Delta\rho)^2V^2P(Q)S(Q),$$

where N is the number of particles, I_0 Is the incident neutron or photon flux, $\Delta\rho$ is the contrast between the particle and the solvent, $P(Q)$ is the form factor of the particle describing its size and shape, and $S(Q)$ is the structure factor describing the organization of the particles. This latter term can be ignored in the case of a dilute system.

The main difference between SAXS and SANS comes from the $\Delta\rho$ term. While photons probe the electron density of the sample, neutrons are scattered by the nuclei. As a result, the interaction of x-rays and neutrons is characterized by different atomic scattering lengths and therefore different contrasts, providing distinct insights into molecular structures. For example, SAXS will be more sensitive to heavy elements, while SANS is able to differentiate between elements of close atomic numbers and even between isotopes. In particular, as hydrogen and deuterium have very different scattering lengths, it is possible to exploit H/D isotopic exchange to vary the contrast between the biomolecule and the solvent and even between parts of the macromolecule in order to obtain additional information.

On the experimental side, both the instrument resolution and statistical quality of the data need to be considered when modelling the data. The photon fluxes available in current synchrotron sources make that both issues can usually be neglected for SAXS data, but this is not the case for SANS data, so the data reduction procedure needs to be carefully performed and the final $I(Q)$ data produced should contain the correct information about the resolution and the uncertainty of each point.

Modeling Conformations

Broadly speaking, there are two very different approaches to model the structure of complex macromolecules in solution:

1. Fully ab initio, using a Reverse Monte Carlo approach where the macromolecule is modelled by a set of beads that are randomly moved until the calculated scattering curve matches the measured one [Svergun et al. 1998].

2. Starting from an atomic description of the macromolecule, often corresponding to the crystal structure.

Here we use the second approach. From the starting point provide by the crystal structure, there are several available methods to explore possible dynamical conformations of proteins and macromolecules:

NOLB (Non-Linear rigid Block NMA approach) introduces a conceptually simple and computationally efficient approach to non-linear normal mode analysis, capturing the intricate non-linear dynamics of proteins. It divides proteins into independent, rigid blocks to provide a more realistic representation of conformational space. This approach enables the study of collective protein motions beyond the linear harmonic motion assumption of traditional methods. NOLB offers conceptual simplicity, computational efficiency, and an accurate representation of essential protein dynamics.

RRT (Rapidly-exploring Random Tree Sampling) is an algorithm adapted for protein conformation calculations, efficiently exploring high-dimensional configuration spaces. It constructs a tree-like structure by iteratively expanding random conformations, enabling efficient sampling of diverse protein conformations within defined constraints. RRT provides efficient sampling, constraint satisfaction, and scalability for large proteins and complex systems.

Monte Carlo and molecular dynamics simulations are also powerful tools for modeling protein dynamics but are notably time-consuming and fall outside the scope of this project, which focuses on RRT and NOLB.

Fitting and Data Requirements

In the fitting process, one crucial aspect is the consideration of the hydration shell, typically represented in the form factor. Sometimes, a multistate modeling approach is necessary, enabling the simultaneous fitting of multiple experimental datasets against different structural models to explore protein conformational ensembles.

To conduct the analysis, users require at least one PDB file, which stores the atomic positions and the initial protein structure or conformation. Existing or published PDB files are available in the Protein Data Bank [Berman 2000] at https://www.rcsb.org.

Additionally, data files containing experimental measurements are indispensable, typically comprising columns for Q (scattering vector), Intensity, and error bars. For SANS data, a resolution column might also be included. Researchers seeking to replicate published results can access data files for SAXS and SANS measurements from the Small Angle Scattering Biological Data Bank [Kikhney et al. 2019]. These essential resources form the foundation for the analysis, enabling the exploration of protein structure and dynamics through SAXS and SANS techniques.

Accessibility

Accessing the resources and tools for Small-Angle X-ray Scattering (SAXS) and Small- Angle Neutron Scattering (SANS) data analysis is straightforward:

- Authentication:Begin by logging in through the Umbrella Authentication and Authorization Infrastructure (AAI) service, a secure single sign-on system designed for seamless access to research and academic resources, at https://replay.notebooks.egi.eu [EGI 2019].

- Binder Environment: Once authenticated, specify the computational environment using Binder, a web-based platform that simplifies the setup of necessary packages and software. Binder ensures that all required components are pre-installed, allowing immediate access to Jupyter Notebooks with predefined configurations.

- GitHub Repository: For our specific project, the Binder environment is hosted on the GitHub repository at https://github.com/isafiulina/sas_helper. This repository [Safiulina 2023] houses not only the essential software components but also example files, including the pivotal SASHelper.ipynbJupyter Notebook and the sas_helper.pymodule, both integral for data analysis and modeling.

- JupyterHub Access: Upon specifying the Binder environment, you will be directed to the JupyterHub interface. Here, you will find the indispensable SASHelper Jupyter Notebook, providing a detailed exposition of SAXS and SANS methods. The notebook elucidates the nuances between these techniques and imparts critical knowledge about their application. It also offers step-by-step instructions for utilizing the routines, enabling users to craft customized Jupyter Notebooks tailored to their specific research objectives.

- sas_helper Module: Within this environment, the sas_helper.py module assumes a pivotal role. It endows users with a spectrum of functionalities for protein structure analysis and visualization:

- Visualizing PDB Files: Leveraging the nglview [Rose et al. 2018] and pytraj [Roe and Cheatham 2013] modules, it facilitates dynamic and interactive visualization of Protein Data Bank (PDB) files, empowering users to explore protein structures with dexterity.
- SAXS and SANS Profile Calculation: The module performs the computation of SAXS and SANS profiles, enabling users to discern how varying parameters influence the intensity of scattering signals, thereby contributing to a deeper comprehension of the data.
- Fitting SAXS/SANS Data: Equipped with FoXS [Schneidman-Duhovny et al. 2016], Pepsi-SAXS [Grudinin et al. 2017], and Pepsi-SANS [Grudinin et al. 2017] tools, the module simplifies the intricate process of fitting experimental SAXS and SANS data. This fitting procedure enables users to draw meaningful insights by aligning experimental data with theoretical models.
- Multi-Model Fitting: For researchers seeking to undertake multi-model fitting, the module supports this endeavor through the multi_foxs [Schneidman-Duhovny et al. 2016] tool. It permits the simultaneous fitting of multiple experimental datasets, accommodating diverse structural models and facilitating the exploration of protein conformational ensembles.
- Protein Conformation Modeling: The sas_helper module seamlessly integrates the NOLB [Hoffmann and Grudinin 2017] and rrt_sample [Schneidman-Duhovny et al. 2016] techniques, offering efficient tools for modeling protein conformations. These tools facilitate the exploration of conformational space, the generation of ensembles, and dynamic studies.
- Simplifying Fitting Processes: By harnessing various libraries and modules, such as FoXS [Schneidman-Duhovny et al. 2016], Pepsi-SAXS [Grudinin et al. 2017], and Pepsi-SANS [Grudinin et al. 2017], the sas_helper module simplifies the often intricate process of fitting SAXS and SANS data. Researchers can perform data analysis and structural modeling without the need for separate software installations or familiarity with divergent interfaces.

In essence, the sas_helper module offers a user-friendly and comprehensive solution for scientists engaged in the analysis of protein structural data. It streamlines the analytical and visualization processes, granting researchers facile access to valuable insights concerning protein structure and dynamics.

Project workflow

The SASHelper.ipynb Jupyter Notebook serves as the gateway to a structured and comprehensive scientific workflow. Within this notebook, users encounter a wealth of scientific background information, command descriptions, and usage guidance. The schematic representation below [Fig. 1] illustrates the workflow's key stages, each designed to facilitate different research objectives. At the inception of the workflow, researchers require data. This data can take the form of original experimental data or be sourced from open databases. The Jupyter Notebook encompasses all the necessary packages to enable subsequent analysis. The versatility of this notebook extends to training purposes, serving as a guiding companion for newcomers. It meticulously leads users through every essential step while providing the necessaryscientific context. Moreover, it can replicate already published data, ensuring reproducibility and offering a valuable resource for researchers looking to validate existing results. Researchers can leverage this platform for their own data analysis, potentially leading to significant discoveries and eventual publication in scientific journals.

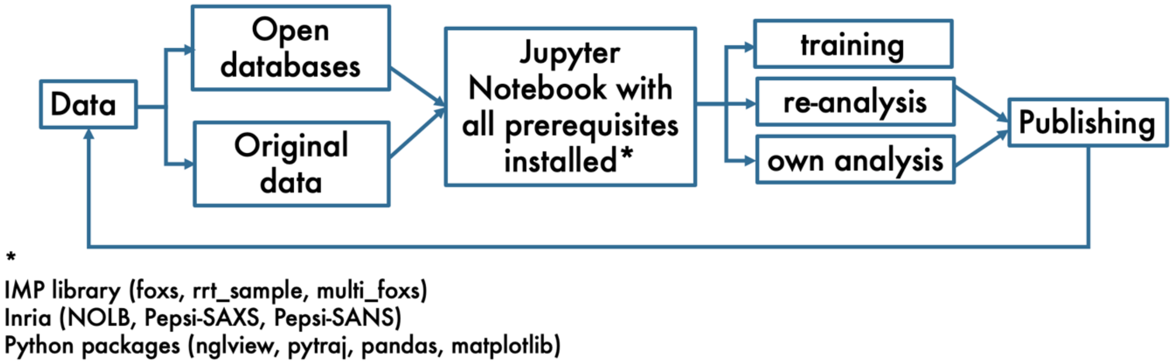
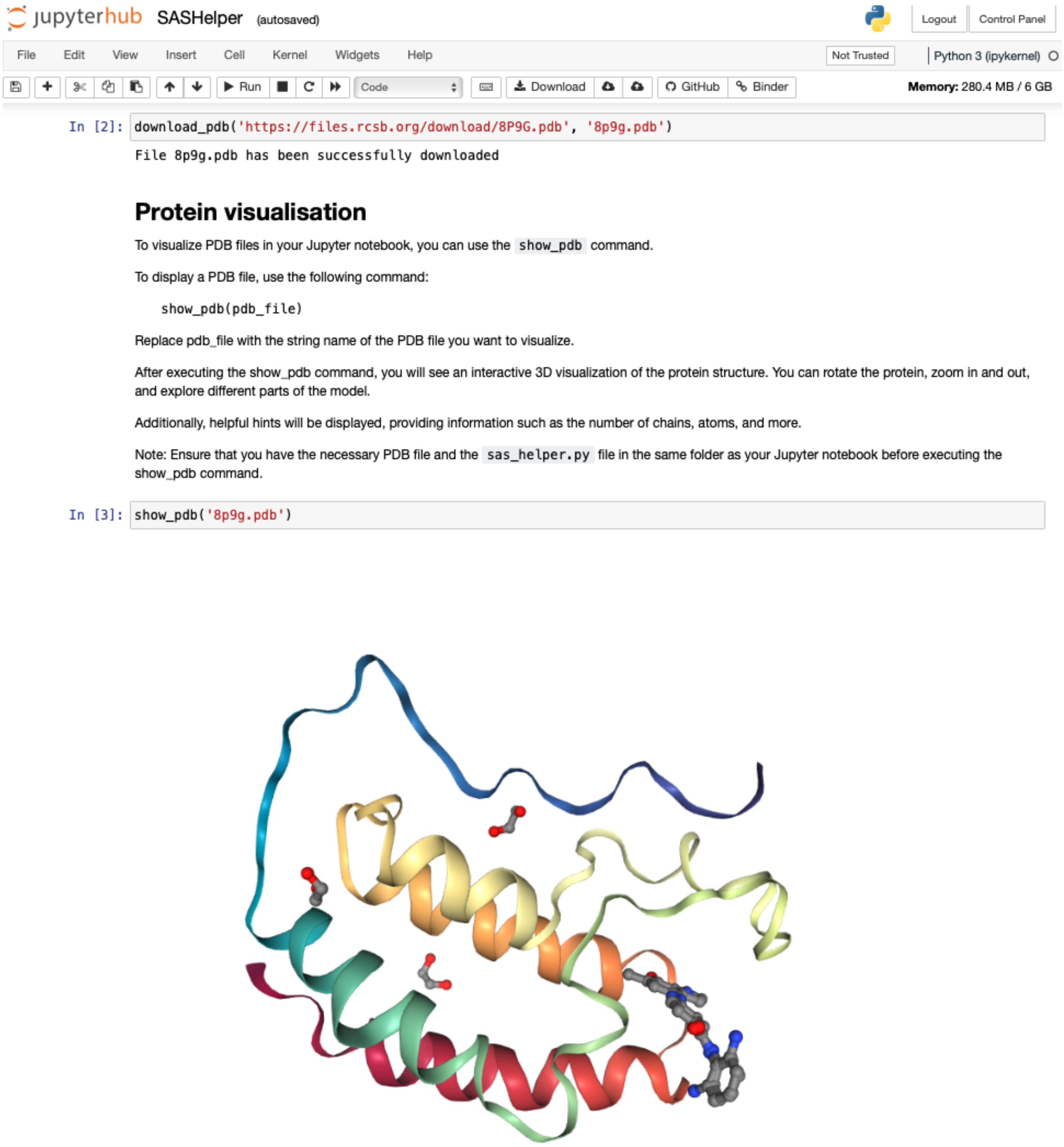

Data
Open databases
Original data
Jupyter Notebook with all prerequisites installed*
training
re-analysis
own analysis
Publishing
*
IMP library (foxs, rrt_sample, multi_foxs)
Inria (NOLB, Pepsi-SAXS, Pepsi-SANS)
Python packages (nglview, pytraj, pandas, matplotlib)

Fig.1. Schematic overview of the Project Workflow

The workflow initiates with the ability to download Protein Data Bank (PDB) files and visualize them with ease. Utilizing simple commands, users can manipulate the structure, rotate it, focus on specific atoms, and zoom in or out. This visualization capability is enhanced by the inclusion of the nglview [Rose et al. 2018] and pytraj [Roe and Cheatham 2013] packages, which come pre-installed in this Jupyter environment [Fig. 2].



```
In [2]: download_pdb('https://files.rcsb.org/download/8P9G.pdb', '8p9g.pdb')
        File 8p9g.pdb has been successfully downloaded
```

Protein visualisation

To visualize PDB files in your Jupyter notebook, you can use the `show_pdb` command.

To display a PDB file, use the following command:

```
show_pdb(pdb_file)
```

Replace pdb_file with the string name of the PDB file you want to visualize.

After executing the show_pdb command, you will see an interactive 3D visualization of the protein structure. You can rotate the protein, zoom in and out, and explore different parts of the model.

Additionally, helpful hints will be displayed, providing information such as the number of chains, atoms, and more.

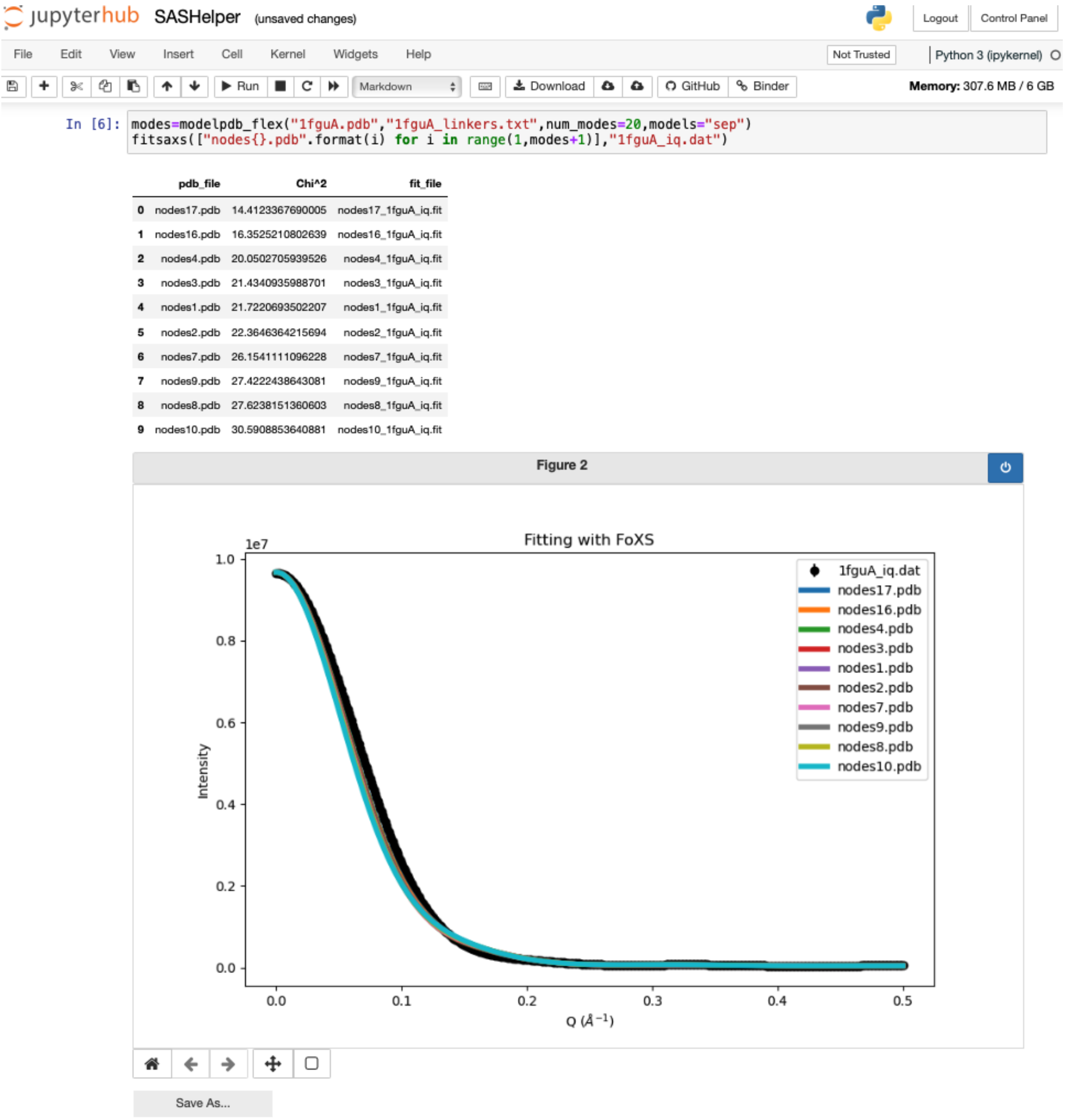
Note: Ensure that you have the necessary PDB file and the `sas_helper.py` file in the same folder as your Jupyter notebook before executing the show_pdb command.

```
In [3]: show_pdb('8p9g.pdb')
```

Fig.2. Example of Jupyter Notebook usage to obtain the PDB file and the following visualisation

The next step involves the modeling of protein or macromolecule structures, accomplished effortlessly with a single command. Users can then explore each conformation generated, whether through the NOLB [Hoffmann and Grudinin 2017] or rrt_sample [Schneidman- Duhovny et al. 2016] techniques. Animated conformations provide insights into structural dynamics, revealing rotations and transformations.

Subsequent to modeling, users can fit their data with the generated structural models. The notebook furnishes a comprehensive output [Fig. 3], including the top 10 χ^2 fits, contributing PDB files, and their weights in the case of multi-state modeling. Visualization of experimental data and fits is achieved through matplotlib [Caswell et al. 2023], offering customization options such as scale adjustment, range modification, title editing, legend management, and figure-saving capabilities.



```
In [6]: modes=modelpdb_flex("1fguA.pdb","1fguA_linkers.txt",num_modes=20,models="sep")
        fitsaxs(["nodes{}.pdb".format(i) for i in range(1,modes+1)],"1fguA_iq.dat")
```

	pdb_file	Chi^2	fit_file
0	nodes17.pdb	14.4123367690005	nodes17_1fguA_iq.fit
1	nodes16.pdb	16.3525210802639	nodes16_1fguA_iq.fit
2	nodes4.pdb	20.0502705939526	nodes4_1fguA_iq.fit
3	nodes3.pdb	21.4340935988701	nodes3_1fguA_iq.fit
4	nodes1.pdb	21.7220693502207	nodes1_1fguA_iq.fit
5	nodes2.pdb	22.3646364215694	nodes2_1fguA_iq.fit
6	nodes7.pdb	26.1541111096228	nodes7_1fguA_iq.fit
7	nodes9.pdb	27.4222438643081	nodes9_1fguA_iq.fit
8	nodes8.pdb	27.6238151360603	nodes8_1fguA_iq.fit
9	nodes10.pdb	30.5908853640881	nodes10_1fguA_iq.fit

Fig.3. Output of fitting with fitsaxs in case of a single PDB structure

This workflow simplifies complex tasks with straightforward commands. While understanding the parameters may require some initial exploration, the notebook streamlines data fitting and subsequent analysis. Users benefit from a unified environment where they don't need to install individual programs separately or seek documentation from disparate sources. Everything is thoughtfully prepared within this Jupyter Notebook for seamless utilization.

For advanced functionalities and a deeper exploration of the notebook's capabilities, users are encouraged to access the notebook itself [Safiulina 2023], where they can further enhance their understanding and harness the full potential of this comprehensive research tool.

Conclusions

In this project report, we have addressed the realm of Small-Angle Scattering techniques, specifically SAXS and SANS, and their profound impact on the study of protein dynamics and structure within solution environments. These techniques have enabled researchers to delve into the structural and dynamic characteristics of proteins and macromolecules, offering a unique perspective on their behaviors.

The report has also outlined the challenges and complexities associated with data analysis in SAS, emphasizing the importance of automating and streamlining this process. To address this need, the project has introduced the sas_helper module, housed in a user- friendly Jupyter Notebook environment, to facilitate data analysis, modeling, and fitting. By consolidating various software resources and integrating powerful tools, this project strives to empower researchers, both novice and seasoned, to delve into the intricate world of protein structural analysis.

Moreover, accessibility has been a key focus of this project, ensuring that researchers can readily engage with the tools and resources provided. The integration of the Umbrella Authentication and Authorization Infrastructure and Jupyter Notebook interface offers a seamless and straightforward approach to accessing and utilizing the sas_helper module.

In conclusion, this project represents a significant step towards democratizing SAS data analysis, promoting open science principles, and enhancing the accessibility of vital tools for the scientific community. By simplifying and centralizing the process of analyzing protein structure and dynamics, we hope to catalyze breakthroughs and foster collaboration in this exciting field of research.

Acknowledgements

PaNOSC has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852. ExPaNDS has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 857641. The EOSC Future project is co-funded by the European Union Horizon Programme call INFRAEOSC-03-2020 - Grant Agreement Number 101017536.

Funding program

EOSC Future project, WP 6.3, co-funded by the EU Horizon Programme call INFRAEOSC-03-2020 – Grant Agreement Number 101017536.

Conflicts of interest

The authors have declared that no competing interests exist.

References

- Berman HM (2000) The Protein Data Bank. Nucleic Acids Research 28 (1): 235-242. https://doi.org/10.1093/nar/28.1.235
- Caswell TA, Lee A, De Andrade ES, Droettboom M, Hoffmann T, Klymak J, Hunter J, Firing E, Stansby D, Varoquaux N, Nielsen JH, Root B, May R, Elson P, Seppänen J, Lee J, Dale D, Gustafsson O, Hannah ,, McDougall D, Straw A, Hobson P, Lucas G, Gohlke C, Vincent A, Yu TS, Ma E, Silvester S, Moad C, Kniazev N, et al. (2023) matplotlib/matplotlib: REL: v3.7.0rc1. Zenodo https://doi.org/10.5281/zenodo.7570264
- EGI (2019) https://replay.notebooks.egi.eu
- Feigin LA, Svergun DI, Taylor G (1987) General Principles of Small-Angle Diffraction. Structure Analysis by Small-Angle X-Ray and Neutron Scattering25-55. https://doi.org/ 10.1007/978-1-4757-6624-0_2
- Grudinin S, Garkavenko M, Kazennov A, et al. (2017) <i>Pepsi-SAXS</i>: an adaptive method for rapid and accurate computation of small-angle X-ray scattering profiles. Acta Crystallographica Section D Structural Biology 73 (5): 449-464. https://doi.org/10.1107/ s2059798317005745
- Hoffmann A, Grudinin S (2017) NOLB: Nonlinear Rigid Block Normal-Mode Analysis Method. Journal of Chemical Theory and Computation 13 (5): 2123-2134. https:// doi.org/10.1021/acs.jctc.7b00197
- Kikhney A, Borges C, Molodenskiy D, Jeffries C, Svergun D (2019) SASBDB: Towards an automatically curated and validated repository for biological scattering data. Protein Science 29 (1): 66-75. https://doi.org/10.1002/pro.3731
- Kluyver T, et al. (2016) Jupyter Notebooks – a publishing format for reproducible computational workflows. https://jupyter.org

- Manalastas-Cantos K, Konarev P, Hajizadeh N, Kikhney A, Petoukhov M, Molodenskiy D, Panjkovich A, Mertens HT, Gruzinov A, Borges C, Jeffries C, Svergun D, Franke D, et al. (2021) <i>ATSAS 3.0</i>: expanded functionality and new tools for small-angle scattering data analysis. Journal of Applied Crystallography 54 (1): 343‑355. https:// doi.org/10.1107/s1600576720013412
- Roe D, Cheatham T (2013) PTRAJ and CPPTRAJ: Software for Processing and Analysis of Molecular Dynamics Trajectory Data. Journal of Chemical Theory and Computation 9 (7): 3084‑3095. https://doi.org/10.1021/ct400341p
- Rose AS, Bradley AR, Valasatava Y, Duarte JM, Prlić A, Rose PW (2018) NGL viewer: web-based molecular graphics for large complexes. Bioinformatics 34 (21): 3755‑3758. https://doi.org/10.1093/bioinformatics/bty419
- Safiulina I (2023) Jupyter notebook for SAXS/SANS modelling and data analysis. https://github.com/isafiulina
- SasView (2020) http://www.sasview.org/
- Schneidman-Duhovny D, Hammel M, Tainer J, Sali A, et al. (2016) FoXS, FoXSDock and MultiFoXS: Single-state and multi-state structural modeling of proteins and their complexes based on SAXS profiles. Nucleic Acids Research 44 https://doi.org/10.1093/ nar/gkw389
- Svergun DI, Richard S, Koch MHJ, Sayers Z, Kuprin S, Zaccai G (1998) Protein hydration in solution: Experimental observation by x-ray and neutron scattering. Proceedings of the National Academy of Sciences 95 (5): 2267‑2272. https://doi.org/ 10.1073/pnas.95.5.2267