Structural characterization of protein-protein interactions with pyDockSAXS

Brian Jiménez-García,[1,2] Pau Bernadó,[3] Juan Fernández-Recio[1,4,5*]

[1]Barcelona Supercomputing Center (BSC), Barcelona, Spain

[2]Bijvoet Center for Biomolecular Research, Faculty of Science - Chemistry, Utrecht University, Utrecht, the Netherlands

[3]Centre de Biochimie Structurale, CNRS, INSERM, Université de Montpellier, Montpellier, France

[4]Institut de Biologia Molecular de Barcelona (IBMB), Consejo Superior de Investigaciones Científicas (CSIC), Barcelona, Spain

[5]Instituto de Ciencias de la Vid y del Vino (ICVV), Consejo Superior de Investigaciones Científicas (CSIC), Logroño, Spain

Running Head

Protein complex structure with pyDockSAXS

Abstract

Structural characterization of protein-protein interactions can provide essential details to understand biological functions at the molecular level, and to facilitate their manipulation for biotechnological and biomedical purposes. Unfortunately, the 3D structure is available for only a small fraction of all possible protein-protein interactions, due to the technical limitations of high-resolution structural determination methods. In this context, low-resolution structural techniques, such as Small-Angle X-ray Scattering (SAXS), can be combined with computational docking to provide structural models of protein-protein interactions at large scale. In this chapter, we describe the pyDockSAXS web server (https://life.bsc.es/pid/pydocksaxs), which uses pyDock docking and scoring to provide structural models that optimally satisfy the input SAXS data. This server, which is freely available to the scientific community, provides an automatic pipeline to model the structure of a protein-protein complex from SAXS data.

Key words

Protein-protein interactions, structural modeling, Small-Angle X-Ray Scattering (SAXS), computational docking, FTDock, CRYSOL, pyDock

1. Introduction

Protein-protein interactions are essential for the majority of biological processes in the cell. The high-resolution description of the 3D structure of these specific protein complexes can improve our understanding of the biological functions and facilitate rational intervention for biotechnological and biomedical purposes. Unfortunately, high-resolution structural data is available for only a tiny fraction of such complexes, due to the limitations of current structural biology methods. In this context, small-angle X-ray scattering (SAXS) has emerged as a powerful low-resolution methods for the characterization of biomolecules and macromolecular assemblies [1-5]. The structural information coded in a SAXS curve can be fully exploited when combined with computational approaches [6]. This combination is especially adequate for providing detailed models of protein-protein complexes [7]. One of the first methods for rigid-body modeling of SAXS data is SASREF [8], which uses simulated annealing to simultaneously fit structural models for protein-protein complexes to multiple SAXS/SANS profiles. However, relying on SAXS data alone is not sufficient to resolve degeneracy in the resulting models, since multiple docking orientations provide similar overall shapes and, therefore, show similar descriptions of the experimental data. This limitation can be overcome by using approaches with the capacity to discriminate between different arrangements. In that context, computational protein-protein docking strategies, which rely on the chemical nature of the interacting surface, turn out to be powerful tools to be combined with SAXS data and improve the quality of the resulting models. Computational docking methods can generate a large number of poses that are geometrically and energetically coherent [9]. The incorporation of SAXS experimental data can narrow the set of docking solutions that are consistent with such experimental data. The first reported method to implement and validate this strategy was pyDockSAXS

[10], and since then other methodologies appeared [11-16]. pyDockSAXS, when systematically tested on a standard protein-protein docking benchmark, showed twofold increase in the predictive success rates with respect to the individual approaches, energy-based docking or SAXS fitting alone. This method was implemented in a web server that provided an automatic pipeline for the structural characterization of protein-protein complexes with SAXS data [17]. In this chapter we will review the pyDockSAXS methodology, with detailed running instructions, example cases and advises for efficient application.

2. Materials

Our method is available as a web service, free accessible at https://life.bsc.es/pid/pydocksaxs. The web front-end acts as a proxy to the user, removing any complexity aroused from a local installation of the software. Via a user-friendly interface, the user is capable of uploading molecular structural information, in PDB format, and experimental SAXS data, compatible with the CRYSOL [18] software, to obtain a set of complex predictions consistent with the experimental data provided (if possible). Our protocol pyDockSAXS, described in [10], samples the rigid-body translational and rotational space in search for the best 10,000 protein complex conformations by means of FTDock [19] software, and then re-scores them by the pyDock scoring energy [20]. After this step, the capacity to describe the experimental SAXS curve is evaluated with the program CRYSOL [18]. A final score combining the agreement with the experimental SAXS curve and the protein docking scoring energy is calculated to filter out the best predictions.

The server runs on a cluster with reserved resources for this service. Allocated resources consist in a computation node composed of 16 cores (4 Intel Xeon E5620 Quad Core at 2.4 GHz) with 11 TB of total available disk space and 256 GB of physical memory. This configuration allows the user to compute docking predictions in a high-performance computing (HPC) environment.

3. Methods

3.1. Input Files

There are two possible modes of using pyDockSAXS server: default mode (in which docking models are generated for two given interacting proteins) and advanced mode (in which the user can provide a previously generated docking set).

3.1.1. Default Mode

In default mode, pyDockSAXS server requires three different files from the user:

a. **Receptor structure file**. This file is in plain-text PDB format and contains the information of the receptor protein structure. In order to avoid inaccurate results or software failure, the PDB structure must contain all the atomic information for every backbone and side-chain atom. Residues with incomplete backbone information are removed in the protocol, while incomplete side-chain atoms are rebuilt using SCWRL version 3.0 software [21]. Multi-chain PDB files are totally supported, while multi-model files are trimmed to the first model. Alternative atom positions are not considered in the protocol. An example of this file can be found online in the *Help* page, section *Sample Data*, file *1PPE_rec.pdb*, which are the coordinates of bovine β-trypsin extracted from the X-ray structure of its complex with CMTI-I (pdb code 1PPE). The use of the bound form in this

example is only for the purpose of clarity, but obviously in a real case, the coordinates (structure or model) of the individual input proteins will be used most of the times.

b. Ligand structure file. It is the same as the receptor structure file, but only containing the structure for the protein ligand. An example of this file can be found online in the *Help* page, section *Sample Data*, file *1PPE_lig.pdb*, corresponding to the coordinates of CMTI-I in complex with bovine β-trypsin (pdb code 1PPE). It is a common practice to define the receptor as the largest molecular partner in the complex, in terms of number of atoms and/or the maximum diameter of the minimum ellipsoid containing the protein, because it is usually faster for FFT-based methods to sample smaller mobile partners.

c. SAXS experimental data. A file containing SAXS experimental data compatible with CRYSOL software version 2.8 (Table 1). This data represents a scattering curve experimental data. The first line is always treated as a title. The following lines should contain momentum transfer, non-zero intensity and standard deviation in a free format (separated by blanks or commas). If standard deviations are not present, the errors are estimated automatically with the help of a polynomial smoothing procedure. An example of this file can be found online in the *Help* page, section *Sample Data*, file *1PPE_curve.dat*.

3.1.2. Advanced Mode

In expert mode (option stated as *advanced users*), an extra file is required:

d. Rigid-body docking set. This file represents a previous computed rigid-body docking by pyDock [20] software or pyDockWEB [22] web server. It is identified by the extension "*.rot*" and contains a set of 13 numerical columns separated by

spaces (Table 2). The first nine columns represent the Euler angles of the rotation matrix applied to the ligand structure to obtain the final complex pose. Columns 10 to 12 represent the translation vector of the ligand structure in respect of the origin of coordinates (0, 0, 0). The last column (13) is a numerical identifier of the docking pose. This docking set file is calculated by pyDock or pyDockWEB, and although it could be calculated by any other external docking program, this is strongly discouraged to avoid inconsistences (unless strictly checking that the format is correct). If this file has been generated in a previous protein-protein docking run or by the pyDockSAXS protocol, you can upload it to speed-up the calculations of the protocol. An example of this file can be found online in the *Help* page, section *Sample Data*, file *1PPE.rot*.

There is also the possibility to load sample data for the complex between *bovine beta-trypsin* and *CMTI-I* (PDB code 1PPE). The user has only to click on the "*Load sample data*" button and the sample files will appear in the same view (Figure 1).

Once the input files have been uploaded, then click on "*Continue*" button.

3.2. Using the pyDockSAXS protocol web server

3.2.1. Chain Selection

Once the user has provided the input files, a new view asks the user to select the chains for both receptor and ligand involved in the protein-protein docking prediction step (Figure 2). If no chain is selected for receptor nor for the ligand, an error will appear asking the user to select at least one chain per subunit. Figure 2 shows the available chains

for the receptor and ligand input files previously extracted from the structure of the complex between *bovine beta-trypsin* and *CMTI-I* (PDB code 1PPE), used as example.

3.2.2. CRYSOL Parameters Selection

After the selection, a new view for configuring additional parameters for CRYSOL [18] is displayed (Figure 3). Only after selecting the option "***For advanced users only**: use custom CRYSOL parameters"*, the *Constant subtraction* and the "*Angular units"* options will be enabled. Here we detail these two options:

a. Constant subtraction. This operation accounts for possible systematic errors due to mismatched buffers in the experimental data. This is a free parameter that is added to all intensities of the scattering profile to improve CRYSOL [18] fitting.

b. Angular units. By default, an attempt is made to estimate the unit scale of the SAXS curve. If angular units are explicitly selected, they will be used by the CRYSOL software and may incur in prediction failure. There are five available options:

- 1/Å, $s = 4\pi\sin(\theta)/\lambda$
- 1/nm, $s = 4\pi\sin(\theta)/\lambda$
- 1/Å, $s = 2\sin(\theta)/\lambda$
- 1/nm, $s = 2\sin(\theta)/\lambda$
- Automatic (by default)

Where s is the momentum transfer, 2θ is the scattering angle, and λ is the X-ray wavelength.

3.2.3. Data Submission

In the final step, a summary of the data provided by the user is displayed (Figure 4). This data is:

- Contact email (if provided). It would be used to notify the user after completion of the computation in the server.
- Receptor PDB structure file name and selected chains.
- Ligand PDB structure file name and selected chains.
- A plot of the scattering curve of the experimental data provided.

When clicking on "*Submit job*" button, the user will be redirected to the job results page, which displays the current status of the job and is automatically refreshed every five minutes.

3.3. Results page

The job has finished when the status in the job results page is set to "*calculated*". At this point, the page shows four basic blocks of data (Figure 5):

a. **Results files**. A compressed file in TAR-GZIP format containing all the results predicted is provided for downloading. Please refer to the *section 3.4* for more details.
b. **Table of predicted energies**. This table, which can be also downloaded in *PDF* format, shows the top 100 predictions as sorted by the pyDockSAXS score, with the different energies calculated by the protocol (Table 3). The order of each docking model is identified in the "*RANK*" column (from 1, the best one, to 100). For each conformation ("*Conf*" column), values for electrostatics ("*Ele*" column), desolvation ("*Desolv*" column) and Van der Waals ("*VDW*" column) energies calculated by pyDock are displayed. The column "pyDock" represents the

pyDock energy, which is calculated as the sum of "*Ele*", "*Desolv*" and 10% of the "*VDW*" column. "*Crysol*" column indicates the value of χ defining the goodness of fit to the SAXS data computed with CRYSOL 2.8. Finally, the "*pyDockSAXS*" column indicates the final score calculated by the protocol.

c. **SAXS curves for the top 10 predicted models compared to the input experimental curve**. Interactive plot where every fitted SAXS curve is in the same colour as the model represented in the 3D visualization section (Figure 5, panel 3). The number on the right of the colour identifies the conformation ("*Conf*" column in the predicted energies table).

d. **Top 10 predicted models in a 3D interactive visualization**. Receptor protein is fixed and displayed using Van der Waals spheres and white colour (Figure 5, panel 4). Ligand models are displayed in backbone-only mode and in different colours. Models can be activated or deactivated using the checkboxes below the 3D representation.

3.4. Output files

Output files are organized in four different folders: *input_data*, *pydock*, *top100* and *fit_top10_SAXS*. The tag "*xxx*" is a numerical identifier of the job in the pyDockSAXS web server:

a. ***input_data***. This folder contains the original data provided by the user.

b. ***pydock***. This folder contains the files generated by the pyDock software:

- *setup.log*: a description of how pyDock reads and parses the original PDB structures provided by the user.
- *project_[xxx]_rec.pdb*: receptor PDB structure after parsing in pyDock.

- *project_[xxx]_rec.pdb.amber*: AMBER94 [23] force field values for atoms in receptor PDB structure used by pyDock.
- *project_[xxx]_rec.pdb.H*: receptor PDB structure parsed by pyDock, including hydrogens.
- *project_[xxx]_lig.pdb*: ligand PDB structure parsed by pyDock.
- *project_[xxx]_lig.pdb.amber*: same as receptor.
- *project_[xxx]_lig.pdb.H*: same as receptor.
- *project_[xxx].rot*: rigid-body docking set generated from the FTDock [19] output.
- *project_[xxx].ftdock*: FTDock software output.
- *project_[xxx].ini*: pyDock initialization file.
- *project_[xxx].ene*: a table with a list of generated conformations scored and ranked by the pyDock energy.
- *project_[xxx].saxs*: a table with a list of *Chi-square* and the *radius of gyration* values for each generated conformation. When *Chi-square* is larger than 10, a 999.0 value is set in the file.
- *project_[xxx].ene.saxs*: a table with a list of generated conformations scored and ranked by the pyDockSAXS energy.

c. **top100**. This folder contains the top 100 structures scored by pyDockSAXS and their corresponding CRYSOL fitted curve. File names follow the pattern "*RankNumber_project_ProjectID_ConformationNumber.extension*", where "*RankNumber*" corresponds to the "*RANK*" column and "*ConformationNumber*" to the "*Conf*" column in the predicted energy table.

d. **fit_top10_SAXS**. A folder containing the top 10 fitted curves calculated by CRYSOL and in line with the *Chi-square* value. The file name format follows the

pattern "*RankNumber_ConformationNumber.fit*", where "*RankNumber*" corresponds to the "*RANK*" column and "*ConformationNumber*" to the "*Conf*" column in the predicted energy table.

4. Discussion

As a test exercise, we can compare the docking models provided by the pyDockSAXS server in the example case (bovine β-trypsin and CMTI-I) with the X-ray structure of the reference complex (PDB 1PPE). In order to evaluate the results, we can calculate standard measures in docking prediction assessment. One of the most popular measures is the ligand RMSD, which is the root mean standard deviation of the ligand atoms in the model with respect to those in the reference, after optimally superimposing the receptor molecules. Figure 6 shows in orange the model ranked 1 by pyDockSAXS for the example case in comparison with the reference complex in blue (receptor molecules from model and reference are superimposed). The ligand RMSD for this model is 2.4 Å, which indicates that the method has worked well in this case. Interestingly, the models with the best fitting to the SAXS curve and with best pyDock energy are further from the reference structure, which shows that the combination of energy-based and SAXS-based scoring is improving the predictive results as compared with the two individual approaches.

Of course, in a real case one does not have the reference complex to compare. The method provides a series of models that can be used to interpret or guide experimental results. There are some hints that can indicate reliability of the predictions, such as convergence of the best-scoring models towards the same structure, good pyDock and/or CRYSOL scoring values, consistency with other available experimental data, etc.

5. Notes

1. If a pre-computed rigid-body docking set is provided, it is mandatory to use the same input structures for receptor and ligand as in the previous docking prediction. This is crucial, as the protocol will not check if the docking set is compatible or not with the input structures.
2. Input PDB structures are one of the major sources of failure of this protocol. One of the reasons is the existing heterogeneity in the PDB file format. Despite the syntax of the PDB format is well defined, third-party software that users can apply to analyse, visualize or generate PDB files is very diverse and may interpret or modify the file format. In our protocol, only lines starting with the keyword "ATOM", containing only protein information, i.e. atom coordinates, type and information related to the standard 20 residues, are parsed. No water, cofactors, small molecules, DNA or RNA data are accepted by the protocol.
3. The second major source of failure of the protocol is the diverse set of errors and mistakes on the experimental SAXS data provided. Wrong identification of the units, wrong use of the constant subtraction or several initial lines in the data file (only the first line is identified by the CRYSOL software as a header or title) are typical examples. Despite many of these errors are controlled in the web server, unorthodox input formats can escape this sanitization step.
4. Predictions might not be compatible with the scattering data provided. In that case, the protocol is not capable of providing a good model for the assembly and thus the top predictions displayed in the results page (or the top predictions in the energy table) will show a wrong *CRYSOL* score (9999.0) to penalize them.
5. The user should note that SAXS contribution to the identification of correct docked models will strongly depend on the shape of the interacting proteins. In cases with anisotropic proteins (i.e. elongated or flattened), SAXS data will be

more discriminative. However, in cases with spherical proteins, the different docking models will yield similar SAXS curves and therefore the scoring will rely mostly on the pyDock energy.

Acknowledgements

This work was supported by the Spanish Ministry of Science (grant BIO2016-79930-R), the European Union H2020 programme (grant MuG 676566), and the Labex EpiGenMed, an "Investissements d'avenir" program (ANR-10-LABX-12-01). The CBS is a member of France-BioImaging (FBI) and the French Infrastructure for Integrated Structural Biology (FRISBI), two national infrastructures supported by the French National Research Agency (ANR-10-INSB-04-01 and ANR-10-INSB-05, respectively).

References

1. Koch MH, Vachette P, Svergun, DI (2003) Small-angle scattering: a view on the properties, structures and structural changes of biological macromolecules in solution. Q Rev Biophys 36:147-227.
2. Putnam CD, Hammel M, Hura GL, Tainer JA (2007) X-ray solution scattering (SAXS) combined with crystallography and computation: defining accurate macromolecular structures, conformations and assemblies in solution. Q Rev Biophys 40:191-285.
3. Jacques DA, Trewhella J (2010) Small-angle scattering for structural biology--expanding the frontier while avoiding the pitfalls. Protein Sci 19:642-657.

4. Cordeiro TN, Herranz-Trillo F, Urbanek A, Estaña A, Cortés J, Sibille N, Bernadó P (2017) Small-angle scattering studies of intrinsically disordered proteins and their complexes. Curr Opin Struct Biol 42:15-23.
5. Bernadó P, Shimizu N, Zaccai G, Kamikubo H, Sugiyama M (2018) Solution scattering approaches to dynamical ordering in biomolecular systems. Biochim Biophys Acta Gen Subj 1862:253-274.
6. Hub JS (2018) Interpreting solution X-ray scattering data using molecular simulations. Curr Opin Struct Biol 49:18-26.
7. Yang S (2014) Methods for SAXS-based structure determination of biomolecular complexes. Adv Mater 26:7902-7910.
8. Petoukhov MV, Svergun DI (2005) Global rigid body modeling of macromolecular complexes against small-angle scattering data. Biophys J 89:1237-1250.
9. Ritchie DW (2008) Recent progress and future directions in protein-protein docking. Curr Protein Pept Sci 9:1-15.
10. Pons C, D'Abramo M, Svergun DI, Orozco M, Bernado P, Fernandez-Recio J (2010) Structural characterization of protein-protein complexes by integrating computational docking with small-angle scattering data. J Mol Biol 403:217-230.
11. Schneidman-Duhovny D, Hammel M, Sali A (2011) Macromolecular docking restrained by a small angle X-ray scattering profile. J Struct Biol 173:461-471.
12. Schneidman-Duhovny D, Hammel M, Tainer JA, Sali A (2016) FoXS, FoXSDock and MultiFoXS: Single-state and multi-state structural modeling of proteins and their complexes based on SAXS profiles. Nucleic Acids Res 44(W1):W424-429.
13. Sønderby P, Rinnan Å, Madsen JJ, Harris P, Bukrinski JT, Peters GHJ (2017) Small-Angle X-ray Scattering Data in Combination with RosettaDock Improves the Docking Energy Landscape. J Chem Inf Model 57:2463-2475.

14. Schindler CEM, de Vries SJ, Sasse A, Zacharias M (2016) SAXS Data Alone can Generate High-Quality Models of Protein-Protein Complexes. Structure 24:1387-1397.

15. Schneidman-Duhovny D, Hammel M (2018) Modeling Structure and Dynamics of Protein Complexes with SAXS Profiles. Methods Mol Biol 1764:449-473.

16. Bonvin AMJJ, Karaca E (2013) On the usefulness of ion-mobility mass spectrometry and SAXS data in scoring docking decoys. Acta Cryst D69:683-694.

17. Jiménez-García B, Pons C, Svergun DI, Bernadó P, Fernández-Recio J (2015) pyDockSAXS: protein–protein complex structure by SAXS and computational docking. Nucleic Acids Res.43(W1): W356-6.

18. Svergun DI, Barberato C, Koch MHJ (1995) CRYSOL - a program to evaluate X-ray solution scattering of biological macromolecules from atomic coordinates. J. Appl Crystallogr 28:768–773.

19. Gabb HA, Jackson RM, Sternberg MJ (1997) Modelling protein docking using shape complementarity, electrostatics and biochemical information. J Mol Biol 272:106-20.

20. Cheng TM, Blundell TL, Fernandez-Recio J (2007) pyDock: electrostatics and desolvation for effective scoring of rigid-body protein-protein docking. Proteins 68:503-15.

21. Wang Q, Canutescu AA, Dunbrack RL Jr (2008) SCWRL and MolIDE: computer programs for side-chain conformation prediction and homology modeling. Nat Protoc 3:1832-47.

22. Jiménez-García B, Pons C, Fernández-Recio J (2013) pyDockWEB: a web server for rigid-body protein-protein docking using electrostatics and desolvation scoring. Bioinformatics 29:1698-9.

23. Cornell WD, Cieplak P, Bayly CI, Gould IR, Merz KM, Ferguson DM, Spellmeyer DC, Fox T, Caldwell JW and Kollman PA (1995) A Second Generation Force Field for the Simulation of Proteins, Nucleic Acids, and Organic Molecules. J Am Chem Soc 117:5179–5197.

Figures Captions

Figure 1. Automatic loading of the example data of a protein-protein complex (PDB code 1PPE) in the pyDockSAXS webserver. The user has to click on the "*Load sample data*" button (1) and the sample files will appear in the same page at the right side (2).

Figure 2. Chain selection page. The user is asked to select the chains for both receptor and ligand involved in the protein-protein docking prediction step.

Figure 3. Advanced mode for selecting CRYSOL-specific parameters. After selecting the option "***For advanced users only**: use custom CRYSOL parameters"* (1), the *Constant subtraction* (2) and the "*Angular units"* (3) options will be enabled.

Figure 4. Submission page. A summary of the main parameters appears before submitting the job to the server.

Figure 5. Results section. The page shows four basic blocks of information provided to the user by the pyDockSAXS webserver: Results files ready to be downloaded (1), a table with the predicted energies for the top 100 docking poses (2), SAXS curves for the top 10 predicted models compared to the input experimental curve (3), and a 3D graphics visualization of the top 10 predicted models (4).

Figure 6. Predicted models for the example case (bovine β-trypsin and CMTI-I). In orange ribbon is represented the position of the ligand in the docking model ranked 1 by pyDockSAXS. This can be compared with the complex reference (PDB code 1PPE) shown in blue ribbon (receptor in blue ribbon and white surface). This model has ligand

RMSD 2.4 Å with respect to the reference, after superimposing the corresponding receptor molecules.

Tables Captions

Table 1. Example of SAXS data file. Only the header and the first 5 rows are shown. The complete sample data of this table can be found at https://life.bsc.es/pid/pydocksaxs/static/data/1PPE_curve.dat.

Randomized data, RELERR= 2.00 %, file 1ppe_ref00.iMon Dec 18 14:30:51 2006		
0.5000E-02	0.1420E+08	0.2909E+06
0.7500E-02	0.1418E+08	0.2899E+06
0.1000E-01	0.1467E+08	0.2886E+06
0.1250E-01	0.1458E+08	0.2869E+06
0.1500E-01	0.1459E+08	0.2848E+06

Table 2. Example of .rot file describing a set of docking models. Only the first 10 rows are shown. The complete sample data for this file can be found at https://life.bsc.es/pid/pydocksaxs/static/data/1PPE.rot.

-0.956	0.004	-0.294	-0.277	0.325	0.905	0.099	0.946	-0.309	-2.306	-5.272	22.608	1
0.223	0.712	-0.665	-0.063	-0.671	-0.739	-0.973	0.207	-0.105	-7.227	10.897	24.014	2
-0.384	0.510	-0.769	0.791	-0.247	-0.559	-0.476	-0.824	-0.309	4.021	-8.084	28.232	3
0.207	-0.743	-0.636	-0.230	-0.669	0.707	-0.951	0.000	-0.309	30.032	-1.054	34.559	4
-0.780	0.504	-0.372	-0.175	-0.745	-0.644	-0.601	-0.437	0.669	18.081	-2.460	-10.433	5
-1.000	0.000	0.000	-0.000	-1.000	0.000	0.000	-0.000	1.000	3.318	1.055	38.074	6
0.420	-0.788	-0.450	-0.889	-0.258	-0.378	0.182	0.559	-0.809	8.239	-8.787	28.232	7
-0.763	0.336	0.552	-0.513	-0.834	-0.201	0.393	-0.437	0.809	33.547	-7.381	16.281	8
-0.541	0.393	-0.743	-0.588	-0.809	-0.000	-0.601	0.437	0.669	26.517	-12.302	6.439	9
-0.856	-0.143	-0.497	0.507	-0.038	-0.861	0.104	-0.989	0.105	2.615	17.224	7.142	10

Table 3. Example of results table. Only the header and the top 10 conformations and their respective energies are shown.

Conf	Ele	Desolv	VDW	pyDock	Crysol	pyDockSAXS	RANK
6367	-7.268	-8.294	57.23	-9.839	3.8	58.389	1
3057	-10.327	0.673	22.29	-7.425	3.604	59.02	2
16	-7.341	-9.325	76.99	-8.967	4.806	67.762	3
3536	5.175	-5.623	56.351	5.187	3.669	72.228	4
5503	-8.398	11.501	43.408	7.444	3.614	73.981	5
4722	-5.241	-10.226	53.554	-10.112	5.845	74.506	6
2636	-6.273	14.214	1.569	8.097	3.655	75.01	7
2253	-7.281	11.796	50.262	9.541	3.669	76.582	8
8008	-8.994	6.053	19.842	-0.957	5.026	77.509	9
2637	-8.721	-0.215	16.322	-7.304	5.999	78.421	10

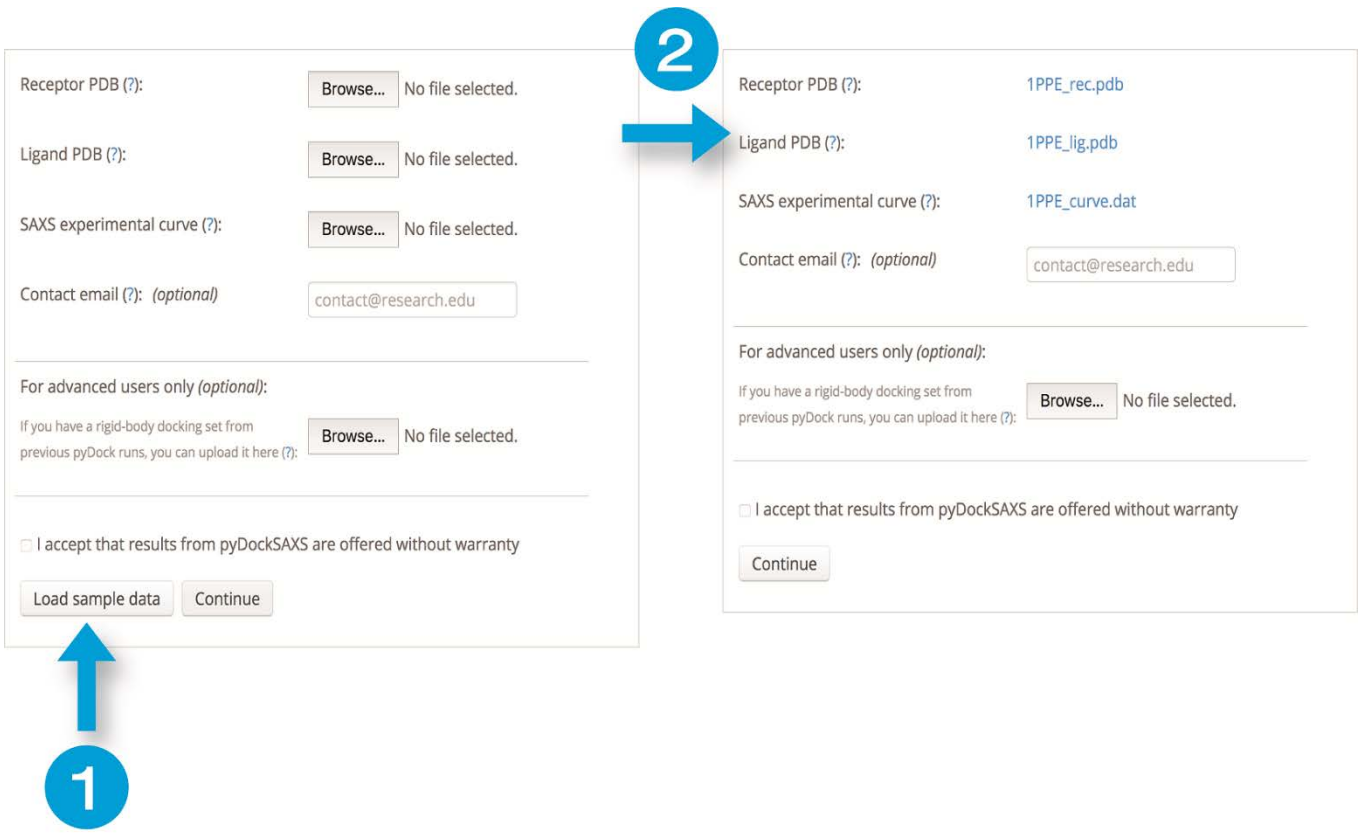

Receptor PDB (?):
Browse... No file selected.
Ligand PDB (?):
Browse... No file selected.
SAXS experimental curve (?):
Browse... No file selected.
Contact email (?): (optional)
contact@research.edu
For advanced users only (optional):
If you have a rigid-body docking set from previous pyDock runs, you can upload it here (?):
Browse... No file selected.
I accept that results from pyDockSAXS are offered without warranty
Load sample data
Continue
1
2
Receptor PDB (?):
1PPE_rec.pdb
Ligand PDB (?):
1PPE_lig.pdb
SAXS experimental curve (?):
1PPE_curve.dat
Contact email (?): (optional)
contact@research.edu
For advanced users only (optional):
If you have a rigid-body docking set from previous pyDock runs, you can upload it here (?):
Browse... No file selected.
I accept that results from pyDockSAXS are offered without warranty
Continue

Figure 1

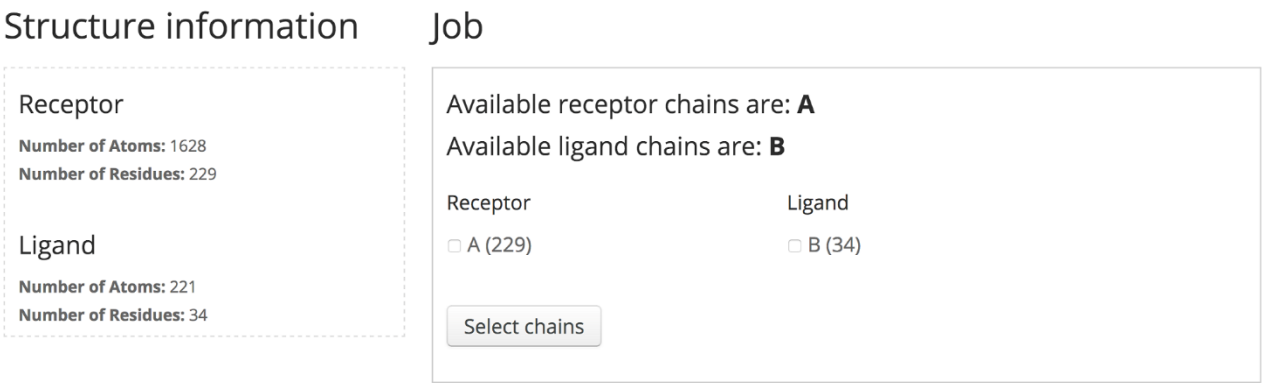

Structure information
Receptor
Number of Atoms: 1628
Number of Residues: 229
Ligand
Number of Atoms: 221
Number of Residues: 34
Job
Available receptor chains are: A
Available ligand chains are: B
Receptor
A (229)
Ligand
B (34)
Select chains

Figure 2

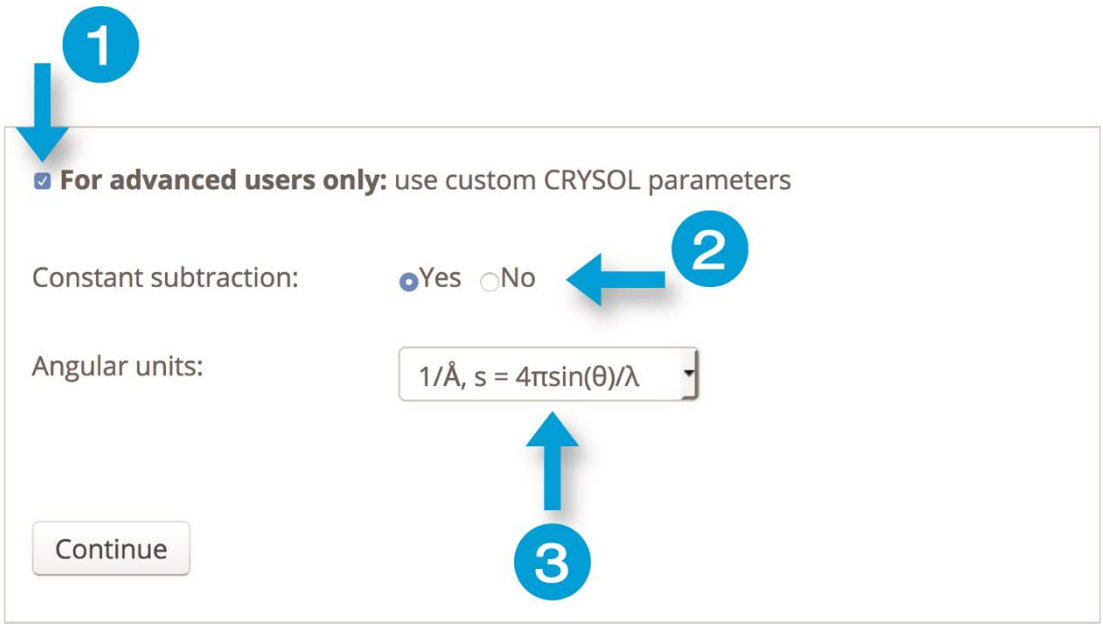
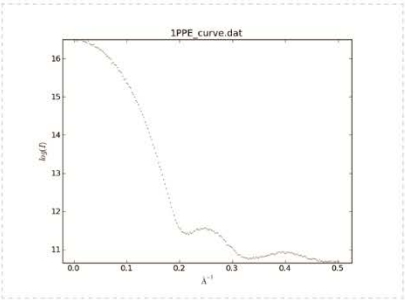

1
For advanced users only: use custom CRYSOL parameters
Constant subtraction:
Yes
No
2
Angular units:
1/Å, s = 4πsin(θ)/λ
3
Continue

Figure 3

SAXS data



Job

Contact email: Not provided

Receptor PDB file: 1PPE_rec.pdb
Selected chains in receptor: A

Ligand PDB file: 1PPE_lig.pdb
Selected chains in ligand: B

Custom crysol parameters:
Constant substraction: Yes
Angular units: 1/Å, s = 4πsin(θ)/λ

Submit job

Figure 4

Results:

The compressed results file includes the top 100 complex PDB structures predicted by pyDockSAXS and their corresponding CRYSOL fit curves. Please, refer to the help section for further details.

Download (compressed tar.gz file): 

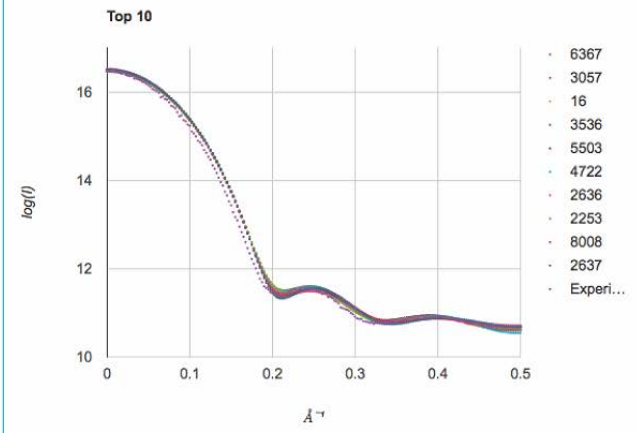
1

2

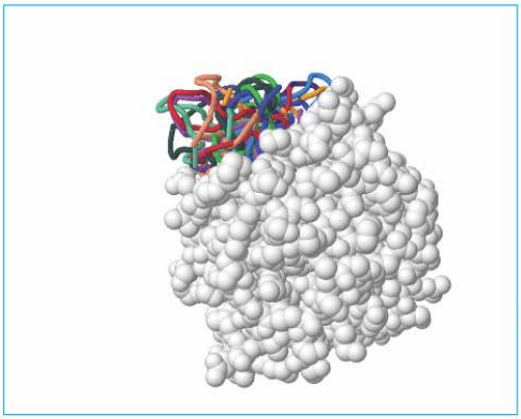
Download the table as a PDF file

Conf	Ele	Desolv	VDW	pyDock	Crysol	pyDockSAXS	RANK
6367	-7.268	-8.294	57.23	-9.839	3.8	58.389	1
3057	-10.327	0.673	22.29	-7.425	3.604	59.02	2
16	-7.341	-9.325	76.99	-8.967	4.806	67.762	3
3536	5.175	-5.623	56.351	5.187	3.669	72.228	4
5503	-8.398	11.501	43.408	7.444	3.614	73.981	5
4722	-5.241	-10.226	53.554	-10.112	5.845	74.506	6
2636	-6.273	14.214	1.569	8.097	3.655	75.01	7
2253	-7.281	11.796	50.262	9.541	3.669	76.582	8
8008	-8.994	6.053	19.842	-0.957	5.026	77.509	9
2637	-8.721	-0.215	16.322	-7.304	5.999	78.421	10
2634	-6.511	0.344	104.84	4.317	4.632	79.644	11
9144	1.303	7.433	12.219	9.959	4.337	82.848	12
5913	-8.111	11.122	60.6	9.071	4.566	83.86	13
995	-6.631	-9.649	98.605	-6.419	6.668	83.96	14
7332	-4.623	-1.674	27.115	-3.585	6.333	84.494	15
6908	-5.063	10.098	26.201	7.655	4.92	85.289	16
9725	-1.402	12.315	22.138	13.127	4.496	87.34	17
2618	-14.482	-6.41	47.451	-16.146	8.827	87.84	18
1952	-4.201	0.713	48.687	1.38	6.406	89.965	19
1245	-6.787	-9.588	47.038	-11.67	8.565	90.761	20
782	-13.767	-8.985	17.653	-20.987	10.201	90.8	21
184	-9.113	-9.458	48.245	-13.747	9.017	91.352	22
5498	-4.581	8.917	68.954	11.231	5.259	91.495	23
6861	-7.97	3.172	24.083	-2.389	7.424	92.976	24
7362	1.495	8.935	7.52	11.182	5.647	94.354	25
4125	-4.252	0.47	35.026	-0.279	7.355	94.641	26
5120	-13.792	-0.131	67.671	-7.156	8.476	94.741	27
1681	-8.853	-7.224	75.021	-8.575	8.917	95.94	28
6427	-3.273	10.159	42.6	11.146	5.908	96.218	29
380	-9.959	11.686	21.215	3.849	7.159	97.496	30
5474	-5.368	9.835	27.573	7.224	6.989	99.753	31
1694	-8.846	11.266	-10.124	1.407	8.241	101.882	32
3063	-10.463	13.695	32.292	6.461	7.68	103.456	33
2102	-6.466	-0.024	67.03	0.213	8.858	104.381	34
6868	-8.587	-0.519	45.38	-4.568	9.748	104.708	35
3060	-3.872	-1.167	35.894	-1.449	9.691	107.507	36
3067	-1.756	13.527	36.471	15.418	6.925	107.522	37
7911	-12.461	-4.449	31.255	-13.785	12.247	108.7	38
2106	-1.452	12.402	9.81	11.931	7.697	109.033	39
5094	-6.873	3.879	11.173	-1.877	10.554	111.827	40

3



Use mouse to zoom in or zoom out. Right click resets view.
Download as image file (including residuals plot)



4

JSmol

Model 1 Model 2 Model 3 Model 4 Model 5
Model 6 Model 7 Model 8 Model 9 Model 10

Figure 5

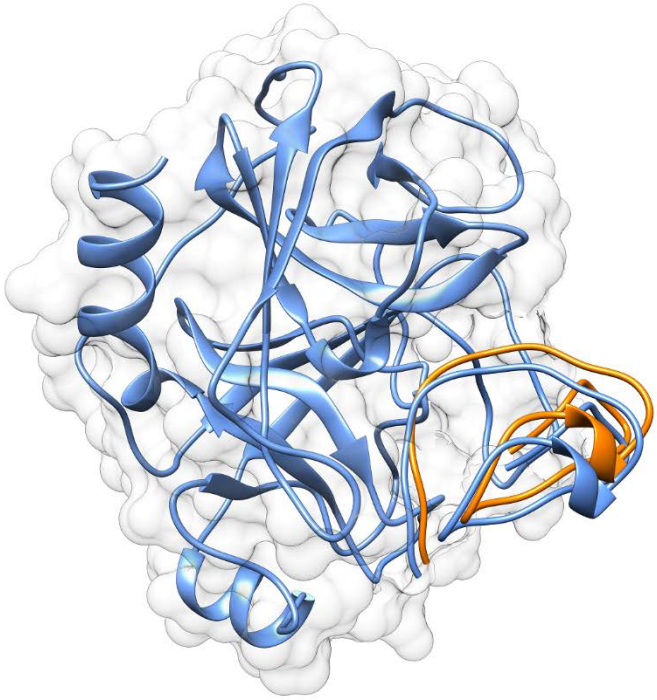


Figure 6