RiboXYZ User Manual

Artem Kushner, Khanh Dao Duc*

August 2022

Contents

*University of British Columbia, Department of Mathematics. Contact: kdd@math.ubc.ca

1 Introduction

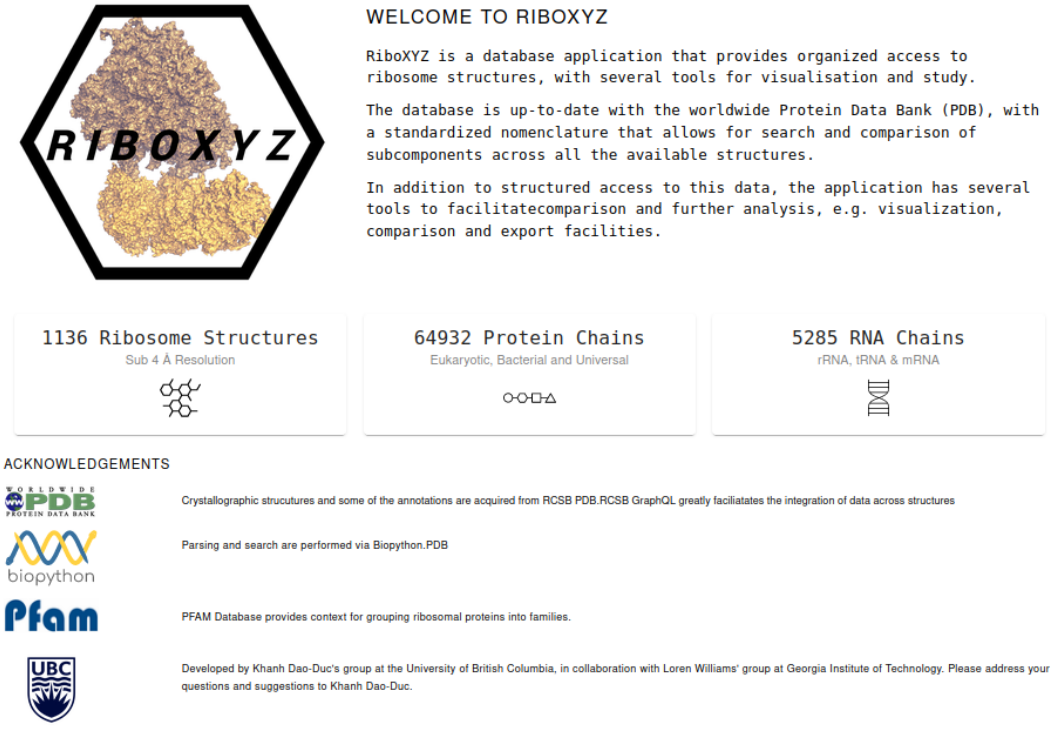

Figure 1: RiboXYZ: Home page

1.1 What is RiboXYZ?

RiboXYZ is a database application that provides organized access to ribosome structures, with several tools for visualisation and study. The database is up-to-date with the worldwide Protein Data Bank (PDB), with a standardized nomenclature that allows for search and comparison across all the available structures. Whole structures as well as individual components of the ribosome like proteins, RNA, ligandsare readily available for download. In addition to structured access to this data, the application has several tools to facilitate comparison and further analysis, e.g. visualization, comparison and export facilities.

1.2 General Infrastructure

The application is divided into two main sections: *Database* and *Tools*. A more detailed description of these sections and their content is provided in section 2.

1.2.1 Database

The *Database* section provides an interface to directly access all ribosome structures available in PDB and their primary components, with filters and multi-field search. The database is organized into four pages, as *Structures* (see 2.1.1), *Proteins* (2.1.2), *RNA* (2.1.3), and *Nomenclature* (2.1.4).

1.2.2 Tools

The *Tools* section provides multiple ways of visualizing the available data and running some analysis, via additional modules organized as, *3D visualization* (2.2.1), *3D superimposition* (2.2.2) and *Ligands/Binding sites* (2.2.3).

1.3 Development

The application is composed of three services: the **website** itself, the Django **server** and the Neo4j graph-**database**:

1. The user-facing website is a single-page application (SPA) that relies on the React framework for presentational components and the Redux library for state-management.
2. The Django web-server is the service that provides structural computation and exposes the API end-points (see section 2.3). This webserver has access to both the database and static files. The scientific Python ecosystem (numpy, scipy, Biopython) forms the backbone of most of the analyses available.
3. The database used is Neo4j and its schema definition language is Cypher, a graph-oriented variant of SQL.

All three parts of the application are deployed to a 8GB Linux node with 120 GB of storage on a public cloud.

Protocols: Communication between the front and back -ends takes place primarily over the http protocol whereas the server and the database communicate over the bolt protocol specific to Neo4j.

Other tools: The source data for the ribosomes and their `.mmcif` structure are obtained from RCSB's Web Service via the GraphQL endpoint. Further processing and analyses of these source data, especially the parsing of the `.mmcif` files is performed via BioPython's PDB module with gemmi structural biology project, and PyMOL's command line interface. Visualization capabilities on the front-end are provided by the Mol* viewer [1].

1.4 Database design, collection and update

We built a semantic database from ribosome model files, using a Neo4j graph-database as repository. The graph structure allows for rich exploration of relationships between ribosome structures and the information contained within the fields of the original model files. The schema of the database has been modelled closely after the RCSB PDB Data model.

Ribosome structures are initially obtained from RCSB PDB's Web Service as `.cif` files. Given the frequency of depositions to PDB, the database is updated (according to the protocol described below) every 2 weeks via a job scheduled on the server.

Before being being released to the application, each file is processed (i) and integrated into the semantic database structure (ii) as follows:

(i) (a) the `.cif` file is split into individual polymer strands (proteins and RNA) and non-polymer entities (ligands, ions, antibiotics and other small molecules), which are separately saved.

(b) If ligands are present in the structure, their binding sites are analyzed: a report is made, listing the residues present in the vicinity of a ligand, and the residue's parent chain if it happens to be a protein (using common protein nomenclature, cf. (ii)). This allows access to the binding site of ligands of the same class (e.g. GDP) across all the structures in the database.

(ii) The annotations provided by the authors themselves as well as external resources are obtained as a .json file from a given PDB deposition and constitute a semantic *profile* of the molecule. Ribosomal protein fields of each profile are augmented with the universal protein nomenclature [2] when possible, by cross-matching them against their corresponding PFAM entries [3]. Several other metrics are further computed that rely on the universal nomenclature and are appended to the nomenclature-augmented profiles. The profiles are then uploaded to the database. Inside the Neo4j-database ribosomal structures and their sub-components are nodes of the Neo4j semantic graph. The corresponding files are served statically and are available for selective or bulk download on the application website.

1.5 Requirements and files format

General access to the website only requires an internet connection. The current version of the website has been developed for and tested in Firefox 78.0.1 (64-bit) web-browser. We recommended to use the website with Mozilla Firefox. Other browsers (e.g. Safari) might have formatting differences.

Multiple file-formats are available for download. Some require specialized softwares to be open and processed. File formats include:

- .fasta is a text-based format for representing nucleotide or amino acid (protein) sequencesusing single-letter codes. It can be read with Biopython SeqIO.
- .mmcif is a macromolecular crystallographic information file that is the default format of the PDB. This format is used to communicate structural information by grouping positions of atoms in a molecule into residues, chains and larger structures. It can be read by a variety of bioinformatics softwares (e.g. PyMOL, ChimeraX, Mol*), as well as operated on programmatically (e.g. Biopython.PDB module and gemmi).
- .csv is a delimited text file that uses a comma to separate values and build tabular data.

2 General Features

The interface of RiboXYZ is divided into two main sections: the *Database* section, which allows for searching and getting access to the structures of the ribosomes and their components, and the *Tools* section, which provides additional modules for visualization and study.

- **Access to the Database and Tools pages:** To get access to the sections and the associated pages, open the menu by clicking on the *gear icon* displayed at the left side of the browser, as shown in Figure 2. This icon is available at any page of the website.

Figure 2: The gear icon opens the Main Menu

2.1 Database

The database section is mainly composed of three pages that provide separate access to whole structures, RNA's, and proteins. For each of these pages, the user can apply a set of filters to run a search over the database, obtain more detailed information on the structures and their components, and export the data, as described below. To facilitate the identification of all the components, the section also contains a specific page displaying the naming system used for ribosome proteins and RNA's.

2.1.1 Structures

The *Structures* page can be accessed from the website drawer menu or directly by using this link `https://ribosome.xyz/structs`. The page contains a main screen that displays models of ribosomes from the PDB, and a left tab menu for filtering. By default, structures are displayed on the main screen in order of PDB codenames, with specific pages associated with each of them (see Figure 3).

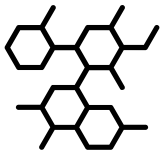

Figure 3: Structure tab in the left menu

- **Single structure details and components:** Each structure is represented by an *ID card* containing deposition details (e.g. organism, resolution, experimental method, publication info), and links to external resources (PDB, doi, EMDB). Clicking on each ID card displayed on the main screen opens a structure-specific page with the ID card on the left. The main panel lists structure components (also designated as chain units in the PDB database). Use the top tab to switch between the components type (Proteins, RNAS's or ligands), and click on a specific component to open the corresponding page in the other database sections (see 2.1.2, 2.1.3, 2.2.3).
- **Searching for structures:** The left tab includes various filtering categories (e.g. date of deposition, experimental method, resolution, ribosomal proteins present, species) and a keyword search tab. Upon choosing species, click on the + symbol next to the bacteria, eukarya or archaea categories to display the full list of species available for any kingdom. Editing the categories will directly update the main screen with filtered structures displayed.
- **Visualization and export:** Upon opening a single structure page, you can visualize it using the *Visualize* button on the left tab. This will open an interactive viewer containing the structure (see section 2.2.1). Alternately, you can use the dedicated *Visualization* page (see 2.2.1). The model file can be downloaded directly by clicking on the download icon of the structure's ID card.

2.1.2 Proteins

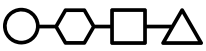

Figure 4: Proteins tab in the menu

The *Proteins* page can be accessed from the website drawer menu or directly by using this link `https://ribosome.xyz/rps`. The page contains a main screen that displays classes of ribosome proteins, organized into three columns, as *universal*, *eukaryotic*, and *bacterial* ribosome protein classes, according to Ban *et al.*'s nomenclature [2] (see also section 2.1.4). The left tab menu can be used for filtering and search. By default, proteins displayed are the ones belonging to the large ribosome subunit (LSU), and listed in alphabetical order for each column. To view proteins that belong to the SSU, toggle the LSU/SSU switch located also at the top of the left tab.

- **Protein class details:** Each class of ribosome protein is represented by a card, which contains the number of structures in which proteins of this class are present. PFAM annotations can be accessed by hovering over the question mark tool-tip. From each class, the user can get access to the *Parent Structures* or *Individual Chains* pages (by clicking on the corresponding buttons), described below.
- **Parent structures:** Clicking on the *Parent Structures* button yields access to the repository of structures that contain the protein class selected. This is equivalent to applying the *"Protein Present"* filter with the same protein class from the *Structures* page.
- **Individual chains:** Clicking on the *Individual Chains* button yields access to the repository of individual chains that belong to the protein class. Each individual chain presents a card that contains sequence information, description provided by the authors of the deposition, nomenclature name and a button giving access to the corresponding *Uniprot* resource. Each chain can be downloaded either as a structural model or an amino-acid sequence via the *"Download Structure (.cif)"* and the *"Download*

Sequence (.fasta)" buttons respectively. An individual chain's parent structure can be viewed and accessed directly by hovering over the oval next to the nomenclature name (showing the interactive structure card).

- **Visualization:** Individual chains can be visualized by clicking on the *Visualize* eye icon that opens a 3D viewer. Alternatively, you can visualize any single protein structure from the dedicated *Visualization* page (see 2.2.1).
- **Searching for specific proteins:** On the left tab, the keyword search bar can be used to navigate to a protein class of interest. The LSU/SSU button also filters proteins according to their belonging to the large (LSU) or small (SSU) ribosomal subunits. Once inside a specific class and *individual chains*, the chains displayed can further be filtered through on the left menu, using the same keyword and filtering tools as in the *Structures page*.
- **Exporting data:** Protein data can be exported by navigating to the protein class of interest and accessing the *Download* menu on the left. The user can choose to download a summary file of the selected chains, or a whole archive containing models of individual strands. In the summary file, proteins appear as rows in the spreadsheet and are uniquely identified by the id of their parent structure and their "strand id" as per PDB. Other details include PFAM and Uniprot accession codes, amino acid sequence. Chains can be downloaded individually or in bulk from the download dialogue menu on the left.

2.1.3 RNA

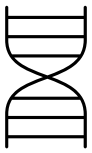

Figure 5: RNA tab in the left menu

The *RNA* page can be accessed from the website drawer menu or directly at this link `https://ribosome.xyz/rnas`. The page main screen displays a list of individual chains of one RNA class, with a left menu that allows for filtering and search.

- **RNA classes:** The user can use the buttons on top of the main screen to select a specific RNA class. The top buttons separate between ribosomal and non-ribosomal RNA's, with sub-tabs that correspond with classes (e.g. 5SrRNA, 5.8SrRNA etc. for ribosomal RNA, and mRNA/tRNA for non-ribosomal RNA). Click on the selected class to display a repository of individual chains present in the database that belong to the class.
- **Individual chains:** Each RNA strand within a class contains sequence information and description provided by the authors of the deposition. The parent structure can be accessed by hovering over the oval next to the nomenclature identifier.
- **Visualization:** Individual chains can be visualized by clicking on the *Visualize* eye icon that opens a 3D viewer. Alternatively, any individual chain can be visualized from the dedicated *Visualization* page (see 2.2.1).
- **Searching for specific individual chains:** Displayed chains displayed can further be filtered through on the left menu, using the same keyword and filtering tools as in the *Structures page*, and sorted by length, year, resolution and sequence length using buttons on top.
- **Exporting data:** Individual strands of interest can be downloaded directly as a `.cif` file via the *Download Structure* button or as a `.fasta` sequence via the *Download Sequence* button. Multiple RNA data can be exported by navigating to the class of interest and accessing the *Download* menu

present on the left under the filters.The user can choose to download a summary file of thestrands that have passed the applied filters, or a whole archive containing models of individual strands. In the summary file, RNA strands appear as rows in the spreadsheet and are uniquely identified by the id of their parent structure and their "strand id" as per PDB.

2.1.4 Nomenclature

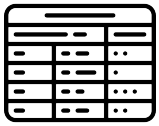

Figure 6: Nomenclature section in the left menu

To facilitate the identification and comparison of ribosome proteins and RNA's across structures, the names of the chain units present the original in cif files have been edited in our database, with a common naming system for all proteins and RNA's. This naming system is accessible at the *Nomenclature* page (accessible via the website drawer menu or directly at `https://ribosome.xyz/nomenclature`). The nomenclature can also be downloaded as a table by clicking on the Download button in the left tab.

By default, the main screen displays protein names in rows, ordered by their index number. Protein classes follow the nomenclature introduced in Ban *et al.* [2]. For each class, the main screen displays their alternative names and PFAM ID's. To switch from proteins to RNA's, use the toggle bar on top. In addition, the search bar can be used for filtering names. Using the toggle bar, one can also switch to a structure-specific page that provides for each structure the mapping between the protein chain structures ID from the original PDB file and the corresponding protein in the nomenclature.

2.2 Tools

The *Tools* section allows to visualize and compare the elements of the database. More or less complex visualization of the data is achieved via three modules, named *Visualization*, *Alignment* and *Ligands/binding sites*, which all integrate a 3D viewer [1].

2.2.1 Visualization

Figure 7: Visualization tool in the left menu

Single structures, ribosome proteins and RNA's can be directly selected from the database and visualized at the *Visualization* page, accessible in the window drawer menu or directly at https://ribosome.xyz/vis . The page displays a searching tool on the left tab menu, and a 3D viewer on the main screen.

- **Selecting an element of the database:** From the menu on the left, choose between structures, proteins or RNA to select which type of data to visualize. For each category, further filtering is available to select a specific class or parent structure. For structures, after selecting a specific structure (identified by their PDB ID number), the user has the option to highlight a specific chain (protein or RNA class) by using the drop-down menu next to the structure selection.
- **Visualizing the element:** Once chosen, the item is rendered in the 3D viewer and can be rotated and inspected. For more details on the viewer's features, see [1].

2.2.2 3D Superimposition

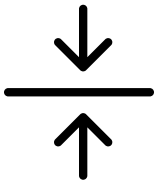

Figure 8: Superimposition tool in the left menu

The *3D Superimpose* tool allows the user to spatially align two chains from two structrures present in the database. The result can also be downloaded as a .pdb or .cif file. The tool can be accessed in the left tab menu or directly at the link www.ribosome.xyz/super. To obtain the superimposition of two chains, the tool executes the 3D superimposition function `super` from an open source version of PyMOL [**?**], to spatially align two chains from two structures of the database.

- **Selecting chains to compare**: Choose two structures and two chains within them from the selection fields on the left side of the page. Adjust the regions to align using the bar tool on the left, or enter the start and end residues to consider.
- **Visualizing the superimposition:** Clicking the *Align* button starts the Pymol's align function [**?**] to superimpose the chains. Once created, the superimposed chains are rendered in the 3D viewer and can be rotated and inspected. For more details on the Mol* viewer features, see [1].

  Exporting the alignment: By clicking on the *Download Aligned* button, the user can download the result as either a `pdb \ or \verb` cif file.

2.2.3 Ligands/Binding Sites

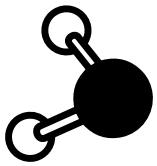

Figure 9: Ligands and Binding Sites tool in the left menu

Certain structures in the database contain components which are neither RNA nor ribosomal protein strands such as antibiotics, transfer and messenger RNA strands and elongation/initiation factors. We collectively refer to these non-ribosomal components as *ligands*. The *Ligands/Binding Sites* tool can be found in the left drawer-menu or directly at the link https://ribosome.xyz/bindingsites and allows the user to visualize the chemical neighborhood of a ligand inside a given structure. The tool also allows, given an extant ligand, predict its potential location in any other structure in which it doesn't exist. Export of residue-wise data for both the original and predicted ligands is made available.

- **Selecting and visualizing a binding site:** Use the drop-down menus on the left in the *Structure of Origin* section to select a structure with its ligand, and visualize it in the main screen by clicking the *Visualize Interface button*. The user is able to either select a ligand and then choose among the structures that contain it or select a structure and then one of the ligands it contains.
- **Download binding site details:** With a ligand-structure pair selected in the *Structure of Origin* section, it's possible to export the data as (.csv or .json) that constitutes the given binding site. Click on the *Download Binding Site* button to do so.

- **Predicting binding sites for other structures:** To predict a ligand's location, first select it from the *Original Structure* tab and choose the structure whose binding site will be used as a template. Select a *Prediction Target* from the *Prediction* section in the lower left. Once all three of *Ligand*, *Structure* and *Prediction Target* have been selected, the user can visualize, inspect and download the predicted binding site.

  - **Visualize Prediction:** To visualize the predicted binding site, the user must switch to the *Prediction* tab of the viewer (as opposed to the default *Original Structure* tab). Once the chosen prediction structure has loaded – click *Visualize Prediction*.
  - **Inspecting & Downloading Prediction:** To view the chains (proteins and RNA) involved in the prediction, click on *Inspect Prediction*. This will bring forward a dialogue window in which the corresponding chains in origin and target structures are aligned and the binding site's residues are highlighted.
  - **Download Prediction** To download a prediction's data click on the *Download Prediction* and select the desired format. A prediction's data includes a number of protein and RNA classes which are present in both the original and the target structure, an alignment of their DNA/RNA sequences and a list of residue ids, which are found within 5 Angstrom of any of the ligand's alpha-carbons.

2.3 API

Our application conforms to the server-client model with the frontend page at *https://ribosome.xyz* frontend being one of many possible clients to our database and API exposed at api.ribosome.xyz. It is outfited with basic Django/Swagger documentation and consists of a number of data-serving endpoints that accept various parameters like structure id, protein class name et cetera. Every endpoint is accessible via a basic HTTP request.

We welcome any development and experimental traffic from researches in the community but do not at present implement any rate-limiting. Hence we kindly ask to be mindful of data consumption and frugal where possible.

3 Tutorials

We provide a step-by-step example of usage for each of the three tools: *Visualization*, 3D Superimposition and *Ligands/Binding Sites*. Each tutorial is inspired by a given practical application of a tool and produces a Mol* model that can be captured and used as a figure of that application. Each tutorial assumes the home page of the database as the starting point and uses no external resources. These tutorials concern the tools aspect of the database, their function and the way to use them. The results of the visualizations created by these tools can serve as a ground for creating a more refined figure. We refer to Mol* viewer documentation for all the ways to manipulate the scene and models as well as to supplementary videos 1, 2 and 3 for a step-by-step guide to reproduce some of the figures in this work.

3.1 Visualization

We consider a scenario whereby the Peptidyl Transferase center of the ribosome needs to be found and highlighted.

1. From the home page navigate to the *Visualization* section by clicking on the gear icon and choosing it from the tools.

2. In the *Structures* tab field of the *Visualization* page now choose the structure of intreste, for exampl *4V7S*. This will initiate the download and rendering of the structure in the Mol* viewer.

3. Once the download is complete, choose the *23SrRNA* chain from Highlight Chain dropdown. This will highlight the chain in the molstar viewer and provide you with access to the residue range selection for

this chain below the structure description. There are multiple significant clusters of highly conserved residues that form the PTC, but for the sake of the example we will choose a range of 7 residues in the 23SrRNA chain: 2445 to 2452. Selecting this range should bring the your view to these residues.

4. Select these residues and create a separate object for this range of residues. Let's label the object "PTC".

5. To visually emphasize this particular range of residues, "PTC", against the rest of the structure we can change its representation to a *Gaussian Surface* and color to a brighter one like orange. The rest of the structure (object "Polymer" in the Molstar state) can be made white.

6. One can adjust the clipping parameter in the scene setting to 70 to bypass the part of the model between the camera and the "PTC". With optionally adding a *Label* representation to the "PTC" object, one is ready to find a desirable angle for capture.

3.2 3D Superimposition

This tutorial concerns the 3D alignment of version of the *uL4* protein associated with the constriction site of the ribosome exit tunnel. This protein is known to have an extended loop of a few residues in eukaryotes. Aligning the bacterial and eukaryotic versions in space will make this extension salient.

1. From the home page, navigate to the *3D Superimposition* section of the tools.

2. Select the eukaryotic 4UG0 structure for *Structure 1* drop-down field and uL4 protein in the *Chain 1* dropdown as well as bacterial ribosome 5AFI for the Structure 2 and protein *uL4* in *Chain 2*.

3. Considering that the limitations of the alignment algorithm, we can take advantage of the residue selection ranges on each of the two chain to make the alignment more specific to sought region. Select the range of 30-170 for the eukaryotic *uL4* and 5-80 for the bacterial version to clip out the rest of the chains. Press *Align*.

4. This will superimpose the corresponding regions of the bacterial and eukaryotic proteins in the viewer.

5. From here, the extended loop eukaryotic loop can be visually augmented by choosing a "Cartoon" representation with highter "Size Factor" and adding labels to the 8 residues that constitute the loop.

3.3 Ligands/Binding Sites

The purpose of this tutorial is to capture residues that constitute a streptomycin binding site in a bacterial structure given a eukaryotic structure initially. The prediction mechanism in this LigandsSites is crude but can be plausibly used to generate predictions on which further analyses may be based like a docking simulation. In this particular case we will attempt to predict a binding site for streptomycin in *m. smegmatis*, a bacterium, based on the existing binding site in a human mitochondrion. There is some utility in this synthetic data given that there are presently no resolved structures of streptomycin bound to *m. smegmatis*.

1. From the home page, navigate to the *Ligands/Binding Sites* tool in the dashboard on the left by clicking the gear icon.

2. *Ligand/Binding Sites* page contains two sections: **Original Structure** and **Prediction**. The first section accounts for visualizing and inspecting the extant binding site for a given ligand whereas the prediction section allows to choose a possible target for the predicting this ligand's binding residues in another structure. The tabs above the viewer correspond to these two sections and should be toggled to view the appropriate section.

3. Select *Streptomycin* in the *Ligands* dropdown and structure 6RW5 that has it bound in the *Binding Sites* dropdown. Wait until the structure is rendered in the viewer. Pressing Visualize Binding Site will highlight the residues in 5 Angstrom vicinity of streptomycin in this strucutre. This residue-wise dataset can be downloaded for further inspection by clicking *Download Bindign Site*.

4. Select the target in the structure of *m. smegmatis* in the Prediction Target dropdown in the *Prediction* section, for example 5ZEP. The Prediction tab on the top of the viewer should become active. Switch to it and wait until the 5ZEP structure is loaded.

5. One is able to inspect and download the resultant alignment between the chains of Origin and *Target* strucutures on which the prediction is based by clicking Inspect Prediction and *Downlaod Prediction* respectively.

6. Clicking on Visulize Prediction at this stage will highlight the residues that are implied to be the binding site for streptomycin in 5ZEP, the target structure.

3.4 Using curated structure for automated processing

To maximize the propagation of the universal naming system for ribosomal proteins, we embed our annotations directly into the mm-cif files as a regular .cif "loop" or table. As a result, it is possible to apply the same operation to the homolog chains (of the same protein class) across multiple structures. For example, given a set of RiboXYZ-annotated prokaryotic structures, one can obtain the approximate PTC coordinates for each of them by:

1. Looking up and extracting the chain with the relevant class name in RiboXYZ annotation table at the bottom of the provided `.cif` file. In this case it is *23SrRNA* for prokaryotes.

2. Further, given this set of single 23SrRNA strands, align each to a strand in which the PTC range is known, for example the 2445-2452 cluster in the 23SrRNA of *e. coli* PDB 5AFI. The nucleotide sequences are found in the corresponding field in the `.cif` file and can also be extracted with a number of parsers(`gemmi`, `Biopython`)

3. Given this set of alignments, which might now include skips because of the length difference – track the "aligned" position ids back to their original positions in the contiguous chains before the alignment.

4. This yields a set of 7 or less (some might have landed on skips during the alignment) residue positions corresponding to the 7 nucleotides of 2445-2452 "seed" range, but now in their respective structures.

5. One has only left to pick one of these residue positions and obtain the 3D coordinate for it, which is also recorded in the `.cif` file. These coordinate are ready to serve as the origin for further analyses.

This particular example is implemented as a Python3 script and is available in supplementary materials (Supplementary File 5) as well as on the **HowTo** page of RiboXYZ.

This can be useful in the scenario where one would like to initiate a cavity search algorithm in multiple structures simultaneously, but in way that's agnostic to their particular strand id's and residue numbers.

References

[1] Sehnal D, Bittrich S, Deshpande M, Svobodová R, Berka K, Bazgier V, et al. Mol* Viewer: modern web app for 3D visualization and analysis of large biomolecular structures. *Nucleic Acids Research* **49**, W431–W437 (2021).

[2] Ban N, Beckmann R, Cate JH, Dinman JD, Dragon F, Ellis SR, et al. A new system for naming ribosomal proteins. *Current opinion in structural biology* **24**, 165–169 (2014).

[3] El-Gebali S, Mistry J, Bateman A, Eddy SR, Luciani A, Potter SC, et al. The Pfam protein families database in 2019. *Nucleic acids research* **47**, D427–D432 (2019).