

Review

# Boosting the Full Potential of PyMOL with Structural Biology Plugins

Serena Rosignoli  and Alessandro Paiardini \* 

Department of Biochemical Sciences "A. Rossi Fanelli", Sapienza Università di Roma, 00185 Rome, Italy

\* Correspondence: [alessandro.paiardini@uniroma1.it](mailto:alessandro.paiardini@uniroma1.it)

**Abstract:** Over the past few decades, the number of available structural bioinformatics pipelines, libraries, plugins, web resources and software has increased exponentially and become accessible to the broad realm of life scientists. This expansion has shaped the field as a tangled network of methods, algorithms and user interfaces. In recent years PyMOL, widely used software for biomolecules visualization and analysis, has started to play a key role in providing an open platform for the successful implementation of expert knowledge into an easy-to-use molecular graphics tool. This review outlines the plugins and features that make PyMOL an eligible environment for supporting structural bioinformatics analyses.

**Keywords:** PyMOL; bioinformatics; plugin; molecular viewer; structural biology; sequence analysis; molecular docking; molecular dynamics; structure-function analysis; protein structure prediction; virtual screening



**Citation:** Rosignoli, S.; Paiardini, A. Boosting the Full Potential of PyMOL with Structural Biology Plugins. *Biomolecules* **2022**, *12*, 1764. <https://doi.org/10.3390/biom12121764>

Academic Editor: Vladimir N. Uversky

Received: 8 November 2022

Accepted: 24 November 2022

Published: 27 November 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The first version of the PyMOL graphical environment for modeling and visualization of molecules, which dates back to 2000, was developed by Warren Lyford DeLano and was at first distributed by DeLano Scientific LLC [1]. After the unexpected death of Warren DeLano in 2009, PyMOL was acquired by Schrödinger Inc. (NY, USA) [2], which continued to support the open-source vision of the original author, but also provided commercial support for maintenance and access to additional features. Currently, PyMOL is a cross-platform, open-source but proprietary software program, maintained, developed and supported by Schrödinger Inc., which also reserves licensing rights. The major changes since Schrödinger Inc. acquisition released with PyMOL version 2 (PyMOL 2), feature a new Graphical User Interface (GUI) that makes use of PyQt5 [3], and replaced Tcl/Tk [4]. PyMOL 2 packages and dependencies management are now on Anaconda [5], since version 2.3, PyMOL started to be developed and released in Python 3 [6]. Such novelties made it possible for PyMOL to be continuously up to date, arriving at the latest available version 2.5, which was released in October 2021. The use of efficient and high-performing graphical libraries, i.e., Open Graphics Library (OpenGL) [7] and the GL Shading Language (GLSL), underlies the advanced images and light renderings of PyMOL, which in turn permits deep control over the setup of final image appearance [8]. Indeed, PyMOL is widely exploited by the scientific community for creating high-quality images and videos that accurately depict the molecular structure being presented [9,10].

One of the main features of PyMOL is the ease for external developers to boost its functionality via scripts and plugins. In this way, PyMOL is programmatically accessible through its editable command-line and Application Programming Interface (API), which make PyMOL features easily exploitable by external software. As a consequence, sequence analysis, molecular docking, molecular dynamics, structure-function relationships analysis, protein structure prediction and virtual screening are all well-known structural bioinformatics approaches for which PyMOL plugins have been developed over the years [11,12]. In

addition to plugins, some useful tools are inherently implemented in PyMOL. For example, PyMOL handles molecular building and editing by providing a builder menu to draw small chemical compounds, peptides and nucleic-acids sequences. The builder menu comes with complete hydrogen and charge fixing tools and with 3D-model cleaning strategies. In the same context, PyMOL supports the possibility to mutate a protein residue or a nucleotide base with the ‘Mutagenesis Wizard’. The ‘Sculpting Wizard’ gives access to a real-time energy minimization function implemented in PyMOL, which is especially useful when dealing with chemical editing.

PyMOL is an ideal candidate for developing advanced Computational Protein Analysis (CPA) pipelines. Its two-pronged nature of molecular graphics viewer and programmatically accessible interface make it particularly suited to easily develop Python-based software tools for advanced investigations of biological macromolecules. A paper in 2017 outlined PyMOL plugins and features that are useful in computational drug design [12]. Some of these plugins are still actively maintained and are worth citing: CAVER 3.0, a tool for the analysis of transport pathways in dynamic protein structures [13]; PYROSETTA, an interactive interface to the powerful Rosetta molecular modeling suite [14]; Dynamics, which adds Gromacs-based molecular dynamics simulation features to PyMOL [15], and the APBS Electrostatics plugin, which implements the Adaptive Poisson–Boltzmann Solver [16] is suited for the analysis of solvation and electrostatics in protein structures. The latter, which has been reviewed in [12] as a standalone plugin, is now available along with PyMOL installation. In addition to those that were previously available, several other PyMOL plugins have been released in recent years. The main aim of this review is to provide an overview on the newly released PyMOL plugins, which are summarized in Table 1, and their scope in CPA. These are introduced by dividing the CPA realm into three categories: (i) Protein Sequences and Structures Analyses (PSSAs); (ii) Protein-Ligand Interactions (PLI); (iii) Protein Dynamics (PD).

**Table 1.** Summary of the available PyMOL plugins for CPA, released in the last years.

Name	Description	Release Date
DockingPie	A platform for molecular and consensus docking (PLI)	2022
PyMod	Environment for structural bioinformatics (PSSAs)	2021
pyProGA	Analysis of static protein residue networks (PSSAs)	2021
MPBuilder	Building and Refinement of Solubilized Membrane Proteins Against SAXS Data (PSSAs)	2021
PoseFilter	Filtering small molecule conformations ensemble (PLI)	2021
DRUGpy	Druggable hot spots identification (PLI)	2021
Geo-Measures	Analyses of protein structures ensemble (PD)	2020
Enlighten2	A platform for MD simulations (PD)	2020
ProBiS H <sub>2</sub> O MD	MD-based prediction of conserved water sites (PSSAs)	2020
iPBAVizu <sup>1</sup>	Protein structure superposition approach (PSSAs)	2019
DCA-MOL <sup>1</sup>	Analysis of Direct Evolutionary Couplings (PSSAs)	2019
pyMODE-TASK <sup>1</sup>	Environment for MD trajectories analyses (PD)	2018
Waterdock 2.0	Water placement prediction (PSSAs)	2017
ProBiS H <sub>2</sub> O	Conserved water sites identification (PSSAs)	2017

<sup>1</sup> Python 2 version of PyMOL is required ( $\leq 2.3$ ). PSSAs, Protein Sequences and Structures Analyses; PLI, Protein-Ligand Interactions (PLI); PD, Protein Dynamics.

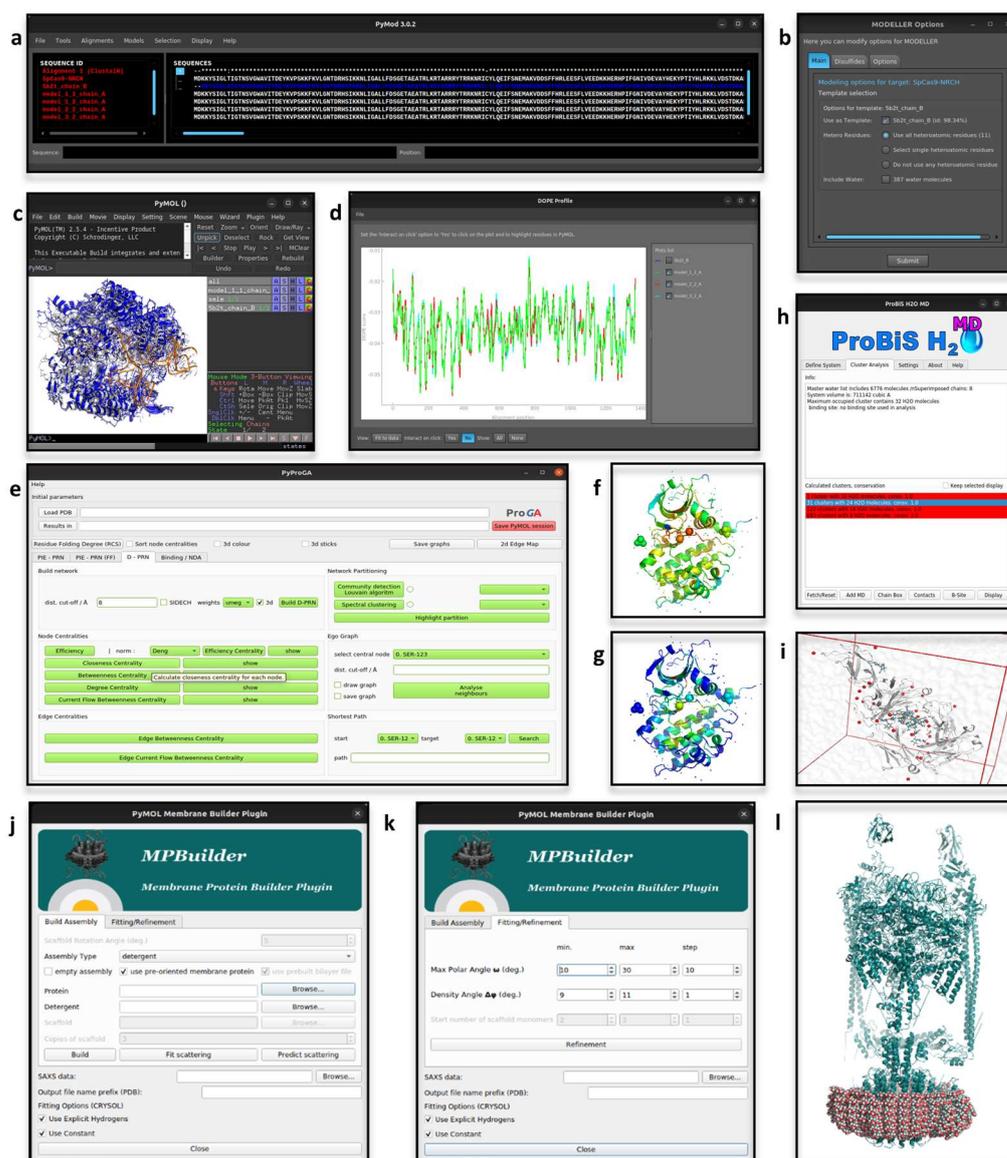
## 2. Protein Sequences and Structures Analyses (PSSAs)

PSSAs are crucial in a variety of biological research areas, particularly in the prediction and modeling of protein structures, which include similarity searches, alignments of sequences and structures, evolutionary and structural comparison, and homology modeling. PyMOL is widely used for PSSAs due to the visual and editing support that is provided for the sequences (i.e., protein, RNA and DNA sequences) of the loaded associated 3D structures.

### 2.1. PyMod

PSSAs include a wide range of often-combined methods such as similarity searches, alignments of sequences and structures, evolutionary comparisons and homology modeling. For example, the template-based prediction of protein 3D quaternary structures in complex with ligands and other heteroatoms requires: (i) databases search for homologous protein (s) with known 3D structure (s); (ii) multiple sequence and structure alignments tools, and (iii) algorithms for protein structure prediction and quality assessment. The PyMod plugin [17] provides a GUI-based and easy-to-use environment for PSSAs by implementing, in an integrated manner, a wide range of algorithms, e.g., (PSI-)Blast [18], MUSCLE [19], ClustalW [20], Clustal Omega [21], WebLogo 3 [22], ESPript 3.0 [23], CAMPO, SCR-Find [24], PsiPred [25] and MODELLER modules for 3D model building [26], sequence-structure alignment [27] and quality assessment [28].

The latest version of PyMod, i.e., PyMod 3, runs on Windows, MacOS, and Linux on both the open source and incentive PyMOL builds provided by Schrödinger Inc. Alignments, sequences or structures can be imported in PyMod, e.g., protein structures that are loaded in PyMOL. Any imported object is processed by PyMod to retrieve the associated information, e.g., heteroatoms and water molecules, and to display its sequence in the dedicated window. The latter provides the interface for the many PyMod functionalities, and is designed to visually support manual editing of sequences and alignments. The plugin so constituted is widely used for template-based protein structure predictions (Figure 1a–d), alignments, evolutionary conservation analyses, building of phylogenetic trees and database searches [29–32]. Protein structure analyses now often rely on AlphaFold (AF) [33] accurate predictions; nonetheless, some underlying dynamics features of proteins may be unveiled by comparing experimental and AF-predicted models. Such a hybrid approach was applied in a recent study on the human trans-3-hydroxy-L-proline dehydratase [34], in which the PyMod support for structural analyses, e.g., structure divergence plots, facilitated the comparisons between experimental and predicted 3D protein structures in different conformational states.



**Figure 1.** (a–d) Appearance of PyMod and PyMOL when running a template-based protein structure prediction. In this example, the ‘fasta’ file of the sequence to be modeled was directly opened in PyMod, while the template structure (PDB-ID: 5B2T [35]) was imported from PyMOL. The sequences of interest are visualized in the dedicated interactive window of PyMod, within which the analyses can be run (a). (b) PyMod window to set-up the parameters to run MODELLER [26] (e.g., how to consider heteroatoms, water molecules or disulfide bridges during calculations). (c) Visualization of the modeled structures in PyMOL workspace. (d) PyMod window for visualizing plots of the quality assessment [28]. (e) Appearance of the pyProGA window, from which the parameters for D-PRN analyses can be set-up. In this example, pyProGA was used to compute two different measurements on a protein (PDB-ID: 1OL5 [36]), which was directly loaded from PyMOL. (f,g) pyProGA processing of the proteins loaded in PyMOL according to the computed measures, which in this example were the ‘Degree Centrality’ (f) and ‘Betweenness Centrality’ (g). (h) Appearance of ProBiS H<sub>2</sub>O MD window for analyzing the results. Each identified cluster of conserved water molecules is identified in PyMOL as a red sphere (i). In this example the plugin was used to identify the conserved water molecules from molecular dynamics (MD) trajectories of a globular protein in water (the input files, a topology and a trajectory file, are provided by the ProBiS web-site (<http://insilab.org/probis-h2o-md>; accessed on 14 October 2022)). (j,k) Appearance of ‘Build Assembly’ and ‘Fitting/Refinement’ tabs of MPBuilder (Ubuntu Linux OS). (l) Visualization in PyMOL of MPBuilder output. The analysis reported here was carried out on the example files provided in MPBuilder development web-resource ([https://github.com/emblsaxs/MPBuilder/tree/main/test\\_cases](https://github.com/emblsaxs/MPBuilder/tree/main/test_cases); accessed on 22 November 2022).

## 2.2. *pyProGA*

Protein residue network (PRN) methods model proteins according to the graph theory [37] to obtain a reduction of the complexity of protein structures into more simple descriptors (residues as vertices/nodes and the interactions between them as edges) [38]. Analyses such as communication pathways, allosteric effects and networks of interactions can all be addressed by making use of PRN-based analyses [39–42]. The plugin Protein Graph Analyser (*pyProGA*) [43] offers a GUI for several PRN methods, i.e., centrality calculation, graph partitions evaluation, shortest path identification, ego graph, network differential analysis, binding energy computing, singular value decomposition technique and pair interaction energies analysis. Graphs solely computed from structural data (PDB file format) are distance-based networks (D-PRN). Otherwise, if an output from either the fragment molecular orbital [44] or an Amber calculation [45] (“*.prmtop*” file) is provided, the computed graphs are based on residue-pair interaction energies (PIE-PRN). The results can be inspected directly in the *pyProGA* window and/or in PyMOL (Figure 1e). The *pyProGA* window accommodates plot visualization, while PyMOL is used for color mapping the results on the analyzed proteins (Figure 1f,g). A PRN-based metric, the residue folding degree, has been proposed as a means to measure the propensity of the protein backbone to form secondary structures in Molecular Dynamics (MD) simulations [46,47]. The residue folding degree analysis relies on conformational state descriptors, namely subgraph centrality, which can be computed from the *pyProGA* plugin. Additional metrics computable by *pyProGA* are appropriate for the identification of residues mediating key interactions between proteins, as showed with the de novo synthesized TIM barrel protein (PDB-ID: 5BVL) [48], for which singular value decomposition (SVD) analysis was applied to identify protein-protein interactions between monomers [43].

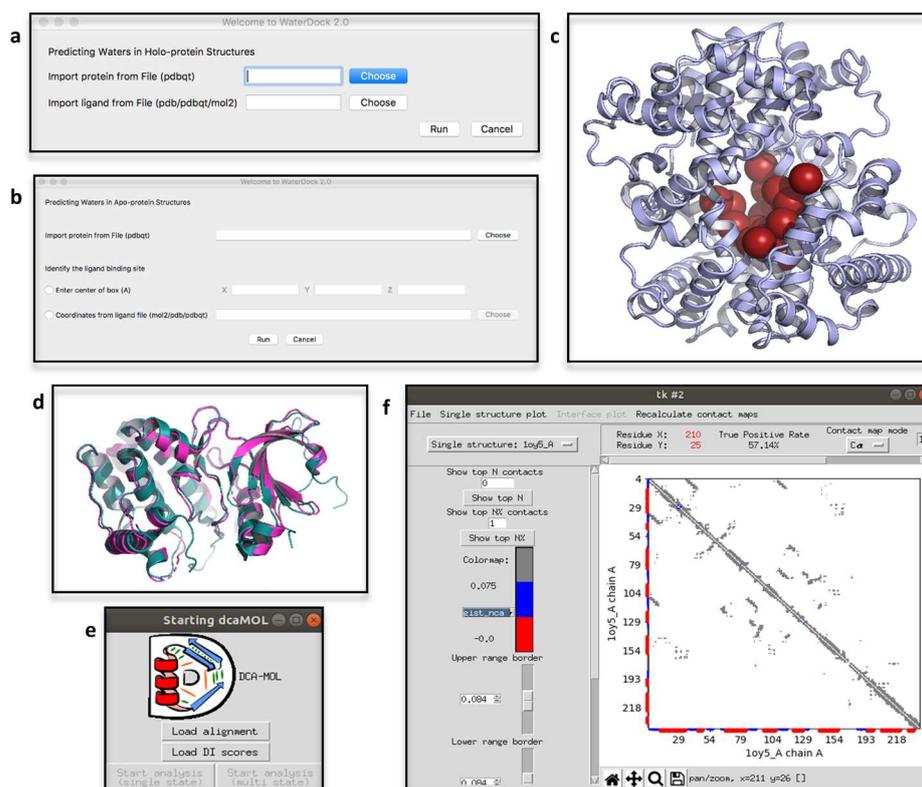
## 2.3. *MPBuilder*

Membrane protein 3D structures are hard to solve, due to their inherent association with lipidic macromolecules that makes the protein solubilization a crucial point [49,50]. Despite the challenges, many protocols have been developed for solving accurate 3D structures of membrane proteins, but are still cumbersome, expensive and time-consuming [51]. The Small-angle X-ray scattering (SAXS) technique is preferred because of less stringent sample preparation requirements. The output of a SAXS experiment needs to be interpreted according to computational modeling approaches [52,53]. A PyMOL plugin, *MPBuilder* [54], has been developed for assisting the modeling of membrane proteins on SAXS-derived data. Two tabs, for building and refinement (Figure 1j–k), guide the user through the analysis. The required inputs are a file storing SAXS data and the choice of an appropriate protein-lipidic assembly to guide the model building. *MPBuilder* greatly exploits PyMOL embedded functions, such as translation, rotation, atomic distance computation and atoms selection, as well as protein and lipidic visualization (Figure 1l). The implementation in the PyMOL framework makes it accessible for regular use, even by non-experts. For instance, *MPBuilder* was used for the interpretation of SAXS data of the phosphoethanolamine enzyme that is active at the lipidic interface, namely lipid A of pathogenic gram-negative bacteria as an antibiotic resistance mechanism [55].

## 2.4. *ProBiS H<sub>2</sub>O*, *ProBiS H<sub>2</sub>O MD* and *Waterdock 2.0*

Water solvation is well known for its influence in protein conformation, pharmacodynamics, rational drug design and protein structure prediction [56,57]. Given the relevance of water-mediated interactions, a PyMOL plugin called *ProBiS H<sub>2</sub>O* has been developed to infer conserved water binding sites by exploiting information derived from experimentally solved water-containing structures [58]. Compared to other computationally expensive methods, such as molecular dynamics-based methods, it provides an easy and fast way for conserved water site identification. The PDB-ID of the protein of interest is the unique mandatory input of *ProBiS H<sub>2</sub>O*, starting from which similar proteins are retrieved, while taking into consideration a user-defined percentage of sequence identity. Afterwards, local

superimposition, sampling and clustering of water molecules are carried out and a conservation score is computed for each cluster. In PyMOL, water molecules are visualized and colored according to the computed conservation score. Recently, a new version of ProBiS H<sub>2</sub>O was released, called ProBiS H<sub>2</sub>O MD [59]. To account for lesser known proteins, hence proteins lacking enough experimental data for using the standard ProBiS H<sub>2</sub>O pipeline, this new version of the plugin optimizes the algorithm extending experimental data with MD trajectories. The latter must be a file (i.e., a “.dcd” file) derived from MD simulations of the protein of interest in a water box. At the end of the analysis, from within the plugin window, alternative visualizations of the clusters can be chosen and the amino acids encompassing the chosen binding site can be visualized (Figure 1h,i). An interesting application of ProBiS H<sub>2</sub>O MD elucidated the influence of local hydration on the PP1-Src kinase ligand binding [60]. Alternatively, interacting water molecules can be inferred by making use of molecular docking engines, as it is done by Waterdock 2.0 [61]. This tool implements Autodock Vina and uses optimized parameters to dock water molecules in protein structures (Figure 2a–c). The influence of water-mediated interactions on protein–ligand complexes was recently investigated by making use of Waterdock 2.0 (in comparison with other software intended for the same aim), for estimating the binding free-energy of human acetylcholinesterase in a complex with tacrine at different solvation states [62].



**Figure 2.** Waterdock 2.0 GUI (mac OS version) for the analysis of the position of water binding sites on holo-(a) or apo-(b) protein structures. Waterdock was tested with inputs available on the plugin development web-resource ([https://github.com/bigginlab/WaterDock\\_pymol](https://github.com/bigginlab/WaterDock_pymol); accessed on 22 November 2022). (c) Waterdock rendering of the identified water molecules positions in PyMOL. (d) Example of PyMOL rendering of the pairwise structural alignment of two protein structures (i.e., Aurora-A kinase in two different conformations; PDB-ID: 1OL5 and 1OL6 [36]) obtained by iPBavizu. (e) Starting window of DCA-MOL, from which alignments and DI-scores can be imported. (f) Example of a contact map obtained from DCA-MOL. Input files available on the plugin development web-resource.

### 2.5. *iPBAvizu*

The functional and structural characterization of protein folding and conformational variability often employs structure superposition and comparison. PyMOL's inherent alignment method is sequence-based, i.e., it makes use of a sequence alignment followed by a structural superimposition. However, for an efficient structure superimposition, related proteins with very low sequence identity must rely on structure-based alignment algorithms such as DALI [63], CE [64], SSAP [65] and iPBA [66]. The iPBAVizu plugin [67] implements iPBA and simply requires a minimum of two 3D protein structures loaded in the PyMOL workspace to compute the structural alignment. The inferred sequence alignment is then visualized in the plugin window and the 3D structures are superimposed in PyMOL (Figure 2d). Many analyses [68,69] are now relying on iPBAVizu for efficient protein structure-based superimposition. For instance, iPBAVizu guided the structural comparison of the primary contributors to SARS-CoV-2 antigenicity for the in-silico design of a multi-epitope vaccine candidate [70].

### 2.6. *DCA-MOL*

Multiple Sequence Alignments (MSAs) have been used for decades to obtain structural, functional and evolutionary information on proteins [71,72]. MSA-based analyses have applications in structure prediction, protein conformation dynamics, analysis of folding pathways, identification of binding sites, inference of interaction interfaces and prediction of interacting partners [73–76]. Some methods that fall into the group of Direct Coupling Analyses (DCA) [77,78] are used to depict the inherent evolutionary information of MSAs of homologous proteins. Co-evolutionary couplings among residue pairs are quantified in DCA with a score, called 'Direct Information' (DI), which is computed for each pair of residues. However, to obtain a comprehensive understanding of such analyses, the results must be processed and mapped on 3D structures. The PyMOL plugin DCA-MOL [79] interactively visualizes coevolutionary residue-residue interactions in contact maps and 3D structures. By providing DI-containing files (Figure 2e), an MSA ".fasta" file and the 3D structure of the protein of interest, DCA-MOL maps the contacts onto the 3D structures loaded in PyMOL workspace and shows contacts and distances maps in its dedicated window (Figure 2f). The interactions are captured either in short (<8.0 Å) or long distances (>8.0 Å) and, when mapped in the same chain (intra-chain interactions), can support protein structure prediction analyses. However, since DCA data of concatenated MSAs can be interpreted, DCA-MOL is not limited to intra-chain analyses, but also inter-chain interactions can be analyzed. On the other hand, by providing more 3D structures of the same protein, the mapping of the analyzed coupling residues can be easily switched by DCA-MOL from one structure to another. In the case of multiple structures, provided by either a MD-trajectory or an experimental source, such visualization can help in rationalizing conformational variabilities. These case studies are deeply investigated in the tutorials provided by the developers (<https://dca-mol.cent.uw.edu.pl>; accessed on 14 October 2022).

## 3. Protein-Ligand Interactions

Protein-ligand interactions, prediction and analyses, comprise a set of widely used methods in Structure-Based Drug Discovery (SBDD), i.e., Molecular Docking (MDo) algorithms, re-scoring metrics, post-processing analyses and Virtual Screening (VS). Driven by the need to have in-hand and easy-to-access environments for SBDD, PyMOL has become a platform for assisting PLI analyses.

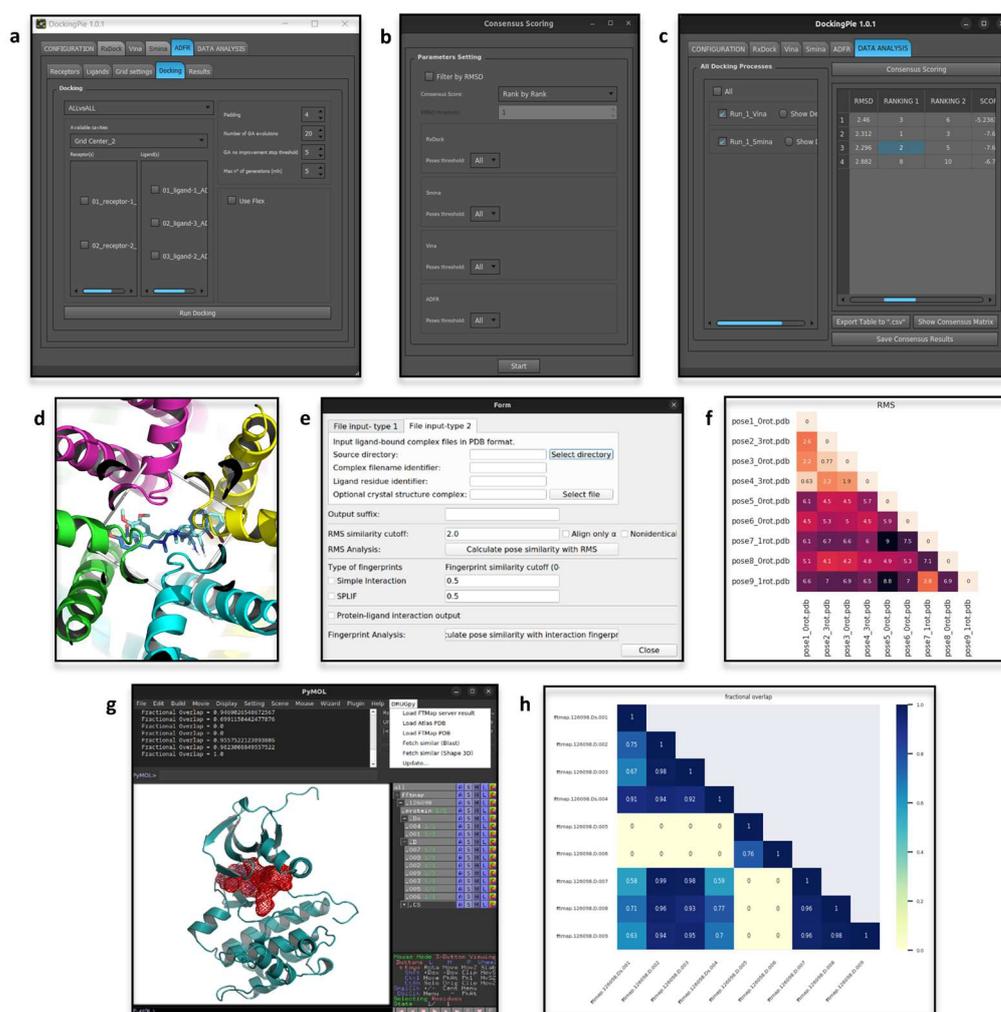
### 3.1. *DockingPie*

The widespread use of protein-ligand docking has led to the development of a wide range of search algorithms, scoring functions, procedures and post-processing analysis tools [80] that create diverse strategies and parameters to be set up by researchers. Although many tools are available for accomplishing each individual task, a lack of integration slows down and hampers the process. A newly developed PyMOL plugin, DockingPie [81],

addresses such limitations by providing an interoperable implementation of the many tools that are needed to carry out each step of a MDo process, from the preparation of input files to the analyses of the results. Currently, the docking engines supported by DockingPie are Smina [82], Vina [83], RxDock [84,85] and ADFR [86] (Figure 3a), as well as their peculiar protocols such as the assignment of side chain flexibility or the setup of pharmacophoric restraints. From within DockingPie, a user can: (i) compute RMSD; (ii) visualize the results by means of predicted affinity-based plots, and (iii) carry out consensus scoring analyses. The input required is any ligand or protein currently loaded in the PyMOL workspace and, for ease of use, DockingPie handles the installation of all external dependencies. How to use the many functionalities of DockingPie is described in different tutorials (available at <https://github.com/paiardin/DockingPie/wiki/Tutorials>; accessed on 14 October 2022). It has been demonstrated that combining the results of different docking programs with the application of a consensus scoring algorithm can increase the success rates in VS processes [87,88]. In this context, due to the implementation of different docking engines and consensus scoring metrics, DockingPie is a tool that lets the user collect and re-score the results in a single integrated environment (Figure 3b–d), which is particularly useful in VS campaigns.

### 3.2. DRUGpy

Hot-spots of interactions between proteins and ligands, namely protein regions with potential positive contribution to the ligand binding free energy, are usually inspected at the early stages of SBDD. The FTMap algorithm [89], which is available as a web server, probes the entire protein surface for detecting hot-spots of interactions, and returns the associated physico-chemical properties. However, the raw nature of FTMap output files prevents the addressing of biologically relevant conclusions. For these reasons, the DRUGpy plugin [90] has been developed to analyze FTMap output data (Figure 3g), and evaluate the level of druggability of the corresponding hot-spots of interaction. DRUGpy quickly locates and defines pockets that FTMap data predict would bind drug-like compounds with high or low affinity (druggable sites), and also displays how protein conformational flexibility affects the target's druggability. If a protein-ligand complex of an identified hot-spot is available, and is superimposed on the hot-spot, DRUGpy can compute fractional overlap to guide rational ligand design (Figure 3h). DRUGpy was used for the comparative analyses of two homologous enzymes, widely validated as drug targets, trypanothione reductase and glutathione reductase [90]. DRUGpy fractional overlap analysis of the druggable hot spots of both enzymes added significant physico-chemical considerations towards the design of species-specific inhibitors.



**Figure 3.** (a) DockingPie GUI (Windows OS) to set up an ‘all vs all’ MDo analysis with ADRF. (b) Appearance of the DockingPie windows (Ubuntu Linux OS) to run a consensus scoring analysis (c) and to inspect the results. In the latter, the re-scoring results are displayed in an interactive-table that can be clicked to directly visualize the molecules of interest in PyMOL (d). In this example, the re-scored poses were the results of two MDo runs carried out with Vina and Smina, from within DockingPie on a homo-tetrameric protein (PDB-ID: 5KMH [91]). (e) PoseFilter GUI to carry out either RMSD or interaction fingerprint comparison when a directory storing the objects to be analyzed as separated files (i.e., the protein and each conformation of the ligand) is provided. PoseFilter was used to analyze the MDo results obtained with DockingPie. (f) Heatmap of RMS values reporting the results of PoseFilter. (g) Visualization of the DRUGpy plugin and PyMOL workspace. The output data of a FTMMap analysis on Aurora-A protein (PDB-ID: 1OL5 [36]) were loaded in DRUGpy for the analysis. DRUGpy automatically shows the identified hot-spots of interaction in PyMOL and, when a protein-ligand complex is analyzed, computes the fractional overlap analysis (shown as a heatmap (h)).

### 3.3. PoseFilter

MDo analyses, especially when carried out on a large-scale, return a huge number of binding conformations to be analyzed. This is very common for flexible, ensemble and consensus docking protocols, making necessary the use of a tool for post-processing analyses of docked conformations. Indeed, some MDo engines directly implement a filtering tool for discarding redundant poses. However, the fact that a lot of tools do not consider symmetric poses leads to a misinterpretation of molecule orientations and similarities. To overcome this issue, PoseFilter [92] implements two methods for post-processing analyses of docked

complexes, i.e., root mean square deviations (RMSD) and interaction fingerprints, in a way that takes into account symmetric molecules. The correct interpretation of symmetric poses is a hurdle when ligands are docked in symmetrical binding sites, as usually occurs when a ligand binding pocket lies between adjacent monomers of oligomeric molecules. In analyzing these situations, PoseFilter gradually rotates the ligand and carries out an ‘all to all’ comparison, while discarding redundant poses. Either protein-ligand complexes or separated molecules can be provided (Figure 3e), and its use has been shown for (homo-) trimeric, dimeric and tetrameric proteins [92]. Since PyMOL is widely used for inspecting protein-ligand complexes, PoseFilter provides a quick and functional tool for analyzing MDo results (Figure 3f).

#### 4. Protein Dynamics

The dynamic behavior of biomolecules is preferably studied with visual support. Indeed, many algorithms for PD have been implemented in different molecular graphics environments. Whereas the length and size of MD trajectories can hamper storing and visualization, PyMOL was adapted to handle the parsing and rendering of big MD-data files. Such novelty has become a step towards the use of PyMOL either for carrying out simulations or for the post-processing of PD-derived data.

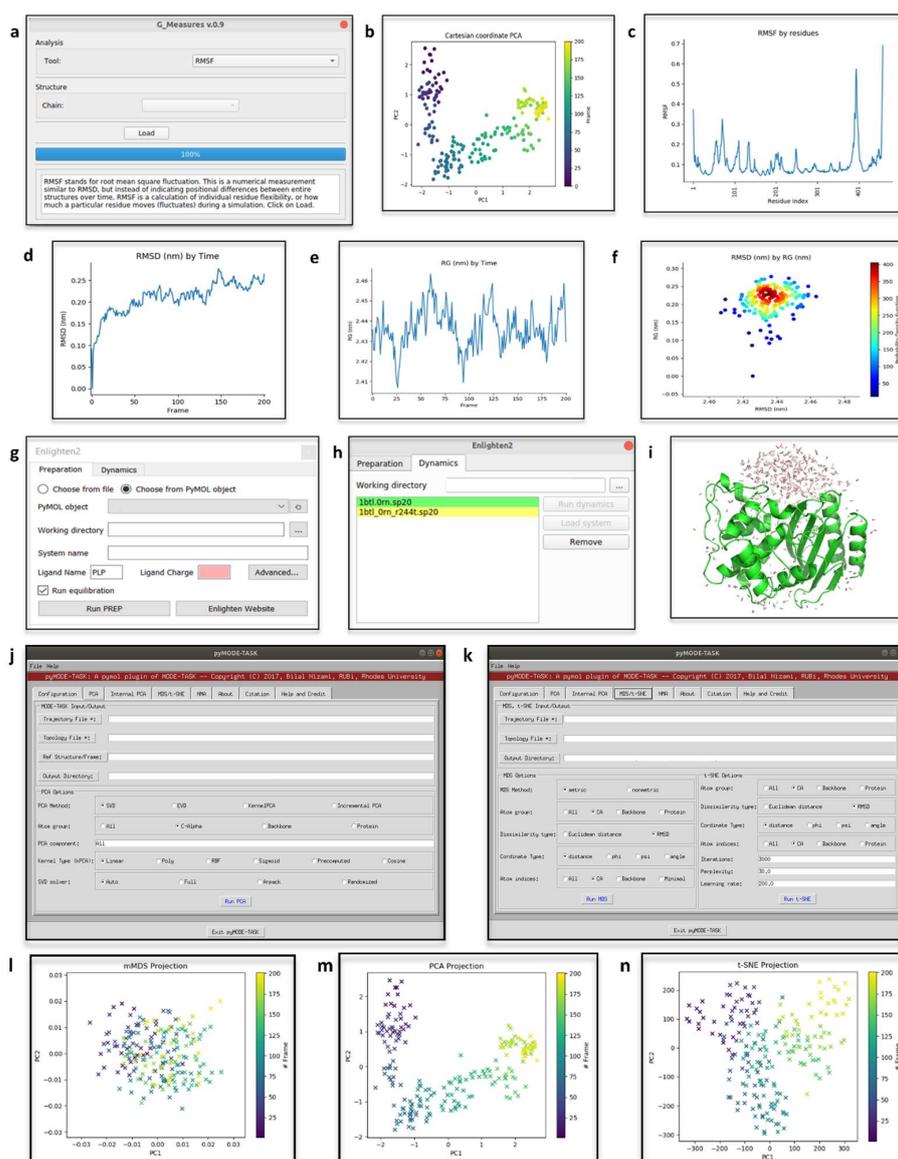
##### 4.1. Geo-Measures

The output of MD simulations is usually in the form of trajectory files that can be further interpreted, visualized and analyzed. Many python modules are available for molecular trajectory file interpretation, yet a free GUI for accessing such features is still missing. Geo-Measures [93] (Figure 4a) was born as a plugin for providing easy access to MDTraj [94], a python library of trajectory analyses modules. Given a topology and trajectory file (file formats supported by PyMOL are feasible), the following analyses can be carried out: (i) C $\alpha$  and dihedral angle calculations; (ii) C $\alpha$  triangle area; (iii) probability density function (PDF) of trajectory frequencies; (iv) root mean square deviation (RMSD) along trajectories; (v) radius of gyration (Rg); (vi) free energy landscape (FEL); (vii) principal component analysis (PCA); (viii) Ramachandran map; (ix) root mean square fluctuation (RMSF); (x) secondary structures definition; (xi) distance calculations, and (xii) trajectory paths visualization. Plots of the results can be visualized and exported as images from a dedicated window of Geo-Measures (Figure 4b–f) The authors provided practical examples of Geo-Measures use on hemoglobin and Ecto-5'-nucleotidase [93]. In a recent work on the mechanistic role of receptors-bound quercetin in hepatoprotection [95], the bound state stability, the compactness and conformational fluctuations were investigated analyzing the MD trajectories of the complex with GeoMeasures. In this application, Geo-Measures was pivotal to easily compute and plot the RMSD, Rg, RMSF and FEL. Moreover, Geo-Measures was the method of choice for analyzing more flexible and scattered trajectories, due to the support for PCA, which is usually applied in these cases [96].

##### 4.2. Enlighten2

Many environments for MD are available; however, the setup of programs for MD still requires detailed knowledge in the field. Enlighten2 [97], an easy-to-install interface to the Amber force-field [98], provides direct access to MD simulations. Protocols to set up MD simulations, which have already been tested and used in the previous version of Enlighten [99–102], together with the GUI, make MD accessible to all. Enlighten2 offers a GUI to several functionalities for molecule preparation (Figure 4g) and MD simulation set-up (Figure 4h). For instance, Enlighten2 implements Antechamber [103] and propka [104] to, respectively, compute ligands parametrization and to modify ligands and protein residues according to user-defined pH values. Furthermore, Enlighten2 supports the possibility to set up how to handle solvation and co-factors. When using Enlighten2, each step that modifies the protein's properties is promptly visualized in PyMOL. Following molecule preparation, a dedicated object is created in PyMOL to store and visualize the solvent cap

(Figure 4i). Moreover, the results are loaded in PyMOL as multiple separated objects. The interoperability of PyMOL and Enlighten2 is also emphasized in the tutorials provided on the Enlighten2 websites (<https://enlighten2.github.io/tutorial1>; accessed on 14 October 2022), where it is explained how to analyze mutant models of two enzyme-ligand complexes created with the PyMOL ‘Mutagenesis Wizard’. Providing easy access to MD simulations, also for non-expert users, it is now often a method of choice to support experimental data, as has been proposed for the engineering of glycan-binding proteins (GBP) [105].



**Figure 4.** (a) Geo-Measures GUI (Ubuntu Linux OS) to run a Root Mean Square Fluctuations (RMSF) analysis on the MD-trajectory loaded in PyMOL. (b–f) Plots of the results of the analyses carried out from within Geo-Measures. In this example, the analyzed data were MD simulations of the multiple endocrine neoplasia type 1 (MEN1) [106]. On such trajectories, PCA (b), RMSF (c), RMSD (d), Rg (e) and PDF between RMSD and Rg values (f), were computed. (g) Enlighten2 GUI (Windows OS) to set-up the parameters for molecule preparation. (h) Enlighten2 GUI (Ubuntu Linux OS) to run the MD simulation. (i) Enlighten2 rendering of the solvent cap in PyMOL. In this example, the tutorial provided on the Enlighten2 website for beta-lactamase TEM-1 (PDB-ID: 1BTL; [107]) was followed. (j,k) Appearance of pyMODE-TASK (Ubuntu Linux OS) tabs for the setup of PCA, MDS and t-SNE analyses. (l–n) Plots provided by pyMODE-TASK to show the results of the PCA, MDS and t-SNE analyses carried out on MEN1 MD simulations data.

### 4.3. *pyMODE-TASK*

The analysis of PD data, either derived from MD simulations or coarse-grained Elastic Network Models (ENMs), is often a complex task; hence dimensionality reduction algorithms have taken hold. Even though many algorithms for deciphering biological relevant conclusions from PD have been developed, their availability as standalone command-line tools, or integration in commercial software, limits a free and easy application. In this context, a recently released tool, MODE-TASK [108] offers an array of implements for analyzing PD-derived data. MODE-TASK can be exploited to carry out the following analyses: (i) PCA (i.e., standard, kernel or incremental PCA); (ii) Normal Mode Analysis (NMA); (iii) Multidimensional Scaling (MDS); (iv) t-Distributed Stochastics, and (v) Neighbor Embedding (t-SNE). Furthermore, MODE-TASK incorporates a new algorithm, namely a coarse graining technique, which was validated for being less computationally expensive than those previously available. PyMOL has been chosen for hosting the GUI of MODE-TASK as a plugin, known as *pyMODE-TASK* (Figure 4j,k). Such integration makes the analysis of proteic large-scale motions easily accessible to PyMOL users. The *pyMODE-TASK* site provides introductory tutorials ([https://pymode-task.readthedocs.io/en/latest/pyMODE-TASK\\_usage.html](https://pymode-task.readthedocs.io/en/latest/pyMODE-TASK_usage.html); accessed on 14 October 2022) on the use of this plugin, and its application in NMA and PCA of proteins (Figure 4l–n). PCA when applied to MD is pivotal to spot differences in the analysis of the same biological system in different states, as described for the destabilizing mutation of renin-angiotensinogen system [108], and the effect of phosphorylation on toll/interleukin-1 receptor domain-containing adapter protein (TIRAP)-mediated signaling [109]. In both works, *pyMODE-TASK* was exploited to compute and plot the principal components of the system.

## 5. Conclusions

It is clear that an interoperable use of software and tools can greatly facilitate the development of pipelines to approach CPA. Several algorithms with applications in this field have been developed accompanied by a command-line interface only, limiting their use to experts in computational biology. Moreover, given the computational capacity and precision, alongside existing ones, an increasing number of in-silico approaches will be available, placing the scientific community in front of a vast choice of alternatives. In this context, PyMOL is positioned as an integrative environment for bridging molecular visualization and CPA. Especially, PyMOL eligibility as a versatile platform, must be acknowledged in relation to accessing its framework through plugins. In addition, its most outstanding features, namely excellent image rendering, programmatic accessibility and inherent chemical editing tools, have contributed to making it an integral part of structural bioinformatics. This review highlights how the use of plugins that exploit and, at the same time, boost the use of PyMOL, can help create an environment to seamlessly carry out CPA.

**Author Contributions:** The authors contributed as follows: Conceptualization, A.P.; literature review and resources, S.R., A.P.; writing—original draft preparation, S.R.; writing—review and editing, A.P.; figure preparation and editing, S.R.; supervision, A.P.; project administration, A.P.; funding acquisition, A.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by AIRC (Italian Association for Cancer Research), grant number MFAG 20447 and by Sapienza University of Rome, Progetto di Ateneo, grant number RP12017275CED09F.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. DeLano, W.L. *The PyMOL Molecular Graphics System*; DeLano Scientific: San Carlos, CA, USA, 2002.
2. PyMOL. Available online: <http://www.pymol.org/pymol> (accessed on 14 October 2022).
3. Summerfield, M. *Rapid GUI Programming with Python and Qt: The Definitive Guide to PyQt Programming*; Prentice Hall: Upper Saddle River, NJ, USA, 2008; ISBN 9780132354189.
4. An Introduction to Tkinter. Available online: [www.Pythonware.Com/Library/Tkinter/Introduction/Index.Htm](http://www.Pythonware.Com/Library/Tkinter/Introduction/Index.Htm) (accessed on 14 October 2022).
5. Anaconda Software Distribution. Anaconda Documentation. Available online: <https://docs.anaconda.com/> (accessed on 14 October 2022).
6. Van Rossum, G.; Drake, F.L. *Python 3 Reference Manual*; CreateSpace: Scotts Valley, CA, USA, 2009; ISBN 1441412697.
7. Woo, M.; Neider, J.; Davis, T.; Shreiner, D. *OpenGL Programming Guide: The Official Guide to Learning OpenGL*, version 1.2; Addison-Wesley Longman Publishing Co. Inc.: Boston, MA, USA, 1999.
8. Mooers, B.H.M. Shortcuts for faster image creation in PyMOL. *Protein Sci.* **2020**, *29*, 268–276. [[CrossRef](#)] [[PubMed](#)]
9. Goodsell, D.S.; Jenkinson, J. Molecular Illustration in Research and Education: Past, Present, and Future. *J. Mol. Biol.* **2018**, *430*, 3969–3981. [[CrossRef](#)] [[PubMed](#)]
10. Martinez, X.; Krone, M.; Alharbi, N.; Rosem, A.S.; Laramee, R.S.; O'Donoghue, S.; Baaden, M.; Chavent, M. Molecular Graphics: Bridging Structural Biologists and Computer Scientists. *Structure* **2019**, *27*, 1617–1623. [[CrossRef](#)]
11. Lill, M.A.; Danielson, M.L. Computer-aided drug design platform using PyMOL. *J. Comput. Aided Mol. Des.* **2011**, *25*, 13–19. [[CrossRef](#)] [[PubMed](#)]
12. Yuan, S.; Chan, H.C.S.; Hu, Z. Using PyMOL as a platform for computational drug design. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2017**, *7*, e1298. [[CrossRef](#)]
13. Chovancova, E.; Pavelka, A.; Benes, P.; Strnad, O.; Brezovsky, J.; Kozlikova, B.; Gora, A.; Sustr, V.; Klvana, M.; Medek, P.; et al. CAVER 3.0: A tool for the analysis of transport pathways in dynamic protein structures. *PLoS Comput. Biol.* **2012**, *8*, e1002708. [[CrossRef](#)]
14. Chaudhury, S.; Lyskov, S.; Gray, J.J. PyRosetta: A script-based interface for implementing molecular modeling algorithms using Rosetta. *Bioinformatics* **2010**, *26*, 689–691. [[CrossRef](#)]
15. Makarewicz, T.; Kazmierkiewicz, R. Molecular dynamics simulation by GROMACS using GUI plugin for PyMOL. *J. Chem. Inf. Model.* **2013**, *53*, 1229–1234. [[CrossRef](#)]
16. Jurrus, E.; Engel, D.; Star, K.; Monson, K.; Brandi, J.; Felberg, L.E.; Brookes, D.H.; Wilson, L.; Chen, J.; Liles, K.; et al. Improvements to the APBS biomolecular solvation software suite. *Protein Sci.* **2018**, *27*, 112–128. [[CrossRef](#)]
17. Janson, G.; Paiardini, A. PyMod 3: A complete suite for structural bioinformatics in PyMOL. *Bioinformatics* **2021**, *37*, 1471–1472. [[CrossRef](#)]
18. Altschul, S.; Madden, T.; Schäffer, A.; Zhang, J.; Zhang, Z.; Miller, W.; Lipman, D. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.* **1997**, *25*, 3389–3402. [[CrossRef](#)]
19. Edgar, R.C. MUSCLE: Multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res.* **2004**, *32*, 1792–1797. [[CrossRef](#)]
20. Thompson, J.D.; Higgins, D.G.; Gibson, T.J. CLUSTAL W: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.* **1994**, *22*, 4673–4680. [[CrossRef](#)] [[PubMed](#)]
21. Sievers, F.; Wilm, A.; Dineen, D.; Gibson, T.J.; Karplus, K.; Li, W.; Lopez, R.; McWilliam, H.; Remmert, M.; Söding, J.; et al. Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. *Mol. Syst. Biol.* **2011**, *7*, 539. [[CrossRef](#)] [[PubMed](#)]
22. Crooks, G.E.; Hon, G.; Chandonia, J.M.; Brenner, S.E. WebLogo: A sequence logo generator. *Genome Res.* **2004**, *14*, 1188–1190. [[CrossRef](#)] [[PubMed](#)]
23. Robert, X.; Gouet, P. Deciphering key features in protein structures with the new ENDscript server. *Nucl. Acids Res.* **2014**, *42*, W320–W324. [[CrossRef](#)]
24. Paiardini, A.; Bossa, F.; Pascarella, S. CAMPO, SCR\_FIND and CHC\_FIND: A suite of web tools for computational structural biology. *Nucleic Acids Res.* **2005**, *33*, W50–W55. [[CrossRef](#)]
25. Jones, D.T. Protein secondary structure prediction based on position-specific scoring matrices. *J. Mol. Biol.* **1999**, *292*, 195–202. [[CrossRef](#)]
26. Webb, B.; Sali, A. Comparative Protein Structure Modeling Using MODELLER. *Curr. Protoc. Bioinform.* **2016**, *54*, 5.6.1–5.6.37. [[CrossRef](#)]
27. Madhusudhan, M.S.; Webb, B.M.; Marti-Renom, M.A.; Eswar, N.; Sali, A. Alignment of multiple protein structures based on sequence and structure features. *Protein Eng. Des. Sel.* **2009**, *22*, 569–574. [[CrossRef](#)]
28. Shen, M.Y.; Sali, A. Statistical potential for assessment and prediction of protein structures. *Protein Sci.* **2006**, *15*, 2507–2524. [[CrossRef](#)] [[PubMed](#)]
29. Rasool, S.; Veyron, S.; Soya, N.; Eldeeb, M.A.; Lukacs, G.L.; Fon, E.A.; Trempe, J.F. Mechanism of PINK1 activation by autophosphorylation and insights into assembly on the TOM complex. *Mol. Cell* **2022**, *82*, 44–59.e6. [[CrossRef](#)] [[PubMed](#)]

30. Hofrichter, M.; Kellner, H.; Herzog, R.; Karich, A.; Kiebish, J.; Scheibner, K.; Ullrich, R. Peroxide-Mediated Oxygenation of Organic Compounds by Fungal Peroxygenases. *Antioxidants* **2022**, *11*, 163. [[CrossRef](#)]
31. Hsin, K.T.; Hsieh, M.C.; Lee, Y.H.; Lin, K.C.; Cheng, Y.S. Insight into the Phylogeny and Binding Ability of WRKY Transcription Factors. *Int. J. Mol. Sci.* **2022**, *23*, 2895. [[CrossRef](#)] [[PubMed](#)]
32. Trabalzini, L.; Ercoli, J.; Trezza, A.; Schiavo, I.; Macri, G.; Moglia, A.; Spiga, O.; Finetti, F. Pharmacological and In Silico Analysis of Oat Avenanthramides as EGFR Inhibitors: Effects on EGF-Induced Lung Cancer Cell Growth and Migration. *Int. J. Mol. Sci.* **2022**, *23*, 8534. [[CrossRef](#)]
33. Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.; Potapenko, A.; et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583–589. [[CrossRef](#)]
34. Ferrario, E.; Miggiano, R.; Rizzi, M.; Ferraris, D.M. The integration of AlphaFold-predicted and crystal structures of human *trans*-3-hydroxy-l-proline dehydratase reveals a regulatory catalytic mechanism. *Comput. Struct. Biotechnol. J.* **2022**, *20*, 3874–3883. [[CrossRef](#)]
35. Hirano, S.; Nishimasu, H.; Ishitani, R.; Nureki, O. Structural Basis for the Altered PAM Specificities of Engineered CRISPR-Cas9. *Mol. Cell* **2016**, *61*, 886–894. [[CrossRef](#)]
36. Bayliss, R.; Sardon, T.; Vernos, I.; Conti, E. Structural basis of Aurora-A activation by TPX2 at the mitotic spindle. *Mol. Cell* **2003**, *12*, 851–862. [[CrossRef](#)]
37. Ghavasi, A.; Nicolini, C.; De Domenico, M. Statistical physics of complex information dynamics. *Phys. Rev. E* **2020**, *102*, 052304. [[CrossRef](#)]
38. Di Paola, L.; De Ruvo, M.; Paci, P.; Santoni, D.; Giuliani, A. Protein contact networks: An emerging paradigm in chemistry. *Chem. Rev.* **2013**, *113*, 1598–1613. [[CrossRef](#)] [[PubMed](#)]
39. Ribeiro, A.A.; Ortiz, V. Determination of Signaling Pathways in Proteins through Network Theory: Importance of the Topology. *J. Chem. Theory Comput.* **2014**, *10*, 1762–1769. [[CrossRef](#)] [[PubMed](#)]
40. Del Sol, A.; Fujihashi, H.; O’Meara, P. Topology of small-world networks of protein-protein complex structures. *Bioinformatics* **2005**, *21*, 1311–1315. [[CrossRef](#)] [[PubMed](#)]
41. Guarnera, E.; Tan, Z.W.; Zheng, Z.; Berezovsky, I.N. AlloSigMA: Allosteric signaling and mutation analysis server. *Bioinformatics* **2017**, *33*, 3996–3998. [[CrossRef](#)]
42. Higman, V.A.; Greene, L.H. Elucidation of conserved long-range interaction networks in proteins and their significance in determining protein topology. *Physica A* **2006**, *368*, 595–606. [[CrossRef](#)]
43. Sladek, V.; Yamamoto, Y.; Harada, R.; Shoji, M.; Shigeta, Y.; Sladek, V. pyProGA-A PyMOL plugin for protein residue network analysis. *PLoS ONE* **2021**, *16*, e0255167. [[CrossRef](#)]
44. Fedorov, D.G.; Nagata, T.; Kitaura, K. Exploring chemistry with the fragment molecular orbital method. *Phys. Chem. Chem. Phys.* **2012**, *14*, 7562–7577. [[CrossRef](#)]
45. Case, D.A.; Cheatham, T.E., 3rd; Darden, T.; Gohlke, H.; Luo, R.; Merz, K.M., Jr.; Onufriev, A.; Simmerling, C.; Wang, B.; Woods, R.J. The Amber biomolecular simulation programs. *J. Comput. Chem.* **2005**, *26*, 1668–1688. [[CrossRef](#)]
46. Sladek, V.; Harada, R.; Shigeta, Y. Protein Dynamics and the Folding Degree. *J. Chem. Inf. Model.* **2020**, *60*, 1559–1567. [[CrossRef](#)]
47. Sladek, V.; Harada, R.; Shigeta, Y. Residue Folding Degree-Relationship to Secondary Structure Categories and Use as Collective Variable. *Int. J. Mol. Sci.* **2021**, *22*, 13042. [[CrossRef](#)]
48. Huang, P.S.; Feldmeier, K.; Parmeggiani, F.; Velasco, D.A.F.; Höcker, B.; Baker, D. De novo design of a four-fold symmetric TIM-barrel protein with atomic-level accuracy. *Nat. Chem. Biol.* **2016**, *12*, 29–34. [[CrossRef](#)] [[PubMed](#)]
49. Moraes, I.; Evans, G.; Sanchez-Weatherby, J.; Newstead, S.; Stewart, P.D. Membrane protein structure determination—The next generation. *Biochim. Biophys. Acta* **2014**, *1838*, 78–87. [[CrossRef](#)] [[PubMed](#)]
50. Almeida, J.G.; Preto, A.J.; Koukos, P.I.; Bonvin, A.M.J.J.; Moreira, I.S. Membrane proteins structures: A review on computational modeling tools. *Biochim. Biophys. Acta Biomembr.* **2017**, *1859*, 2021–2039. [[CrossRef](#)] [[PubMed](#)]
51. Vinokumar, K.R.; Henderson, R. Structures of membrane proteins. *Q Rev. Biophys.* **2010**, *43*, 65–158. [[CrossRef](#)]
52. Schneidman-Duhovny, D.; Hammel, M. Modeling Structure and Dynamics of Protein Complexes with SAXS Profiles. *Methods Mol. Biol.* **2018**, *1764*, 449–473. [[CrossRef](#)]
53. Baranowski, M.; Pérez, J. Solution X-Ray Scattering for Membrane Proteins. *Methods Mol. Biol.* **2020**, *2168*, 177–197. [[CrossRef](#)]
54. Molodenskiy, D.S.; Svergun, D.I.; Mertens, H.D.T. MPBuilder: A PyMOL Plugin for Building and Refinement of Solubilized Membrane Proteins Against Small Angle X-ray Scattering Data. *J. Mol. Biol.* **2021**, *433*, 166888. [[CrossRef](#)]
55. Anandan, A.; Dunstan, N.W.; Ryan, T.M.; Mertens, H.D.T.; Lim, K.Y.L.; Evans, G.L.; Kahler, C.M.; Vrieland, A. Conformational flexibility of EptA driven by an interdomain helix provides insights for enzyme-substrate recognition. *IUCr* **2021**, *8*, 732–746. [[CrossRef](#)]
56. Spyarakis, F.; Ahmed, M.H.; Bayden, A.S.; Cozzini, P.; Mozzarelli, A.; Kellogg, G.E. The Roles of Water in the Protein Matrix: A Largely Untapped Resource for Drug Discovery. *J. Med. Chem.* **2017**, *60*, 6781–6827. [[CrossRef](#)]
57. Barillari, C.; Taylor, J.; Viner, R.; Essex, J.W. Classification of water molecules in protein binding sites. *J. Am. Chem. Soc.* **2007**, *129*, 2577–2587. [[CrossRef](#)]
58. Jukič, M.; Konc, J.; Gobec, S.; Janežič, D. Identification of Conserved Water Sites in Protein Structures for Drug Design. *J. Chem. Inf. Model.* **2017**, *57*, 3094–3103. [[CrossRef](#)] [[PubMed](#)]

59. Jukič, M.; Konc, J.; Janežič, D.; Bren, U. ProBiS H2O MD Approach for Identification of Conserved Water Sites in Protein Structures for Drug Design. *ACS Med. Chem. Lett.* **2020**, *11*, 877–882. [[CrossRef](#)] [[PubMed](#)]
60. Spitaleri, A.; Zia, S.R.; Di Micco, P.; Al-Lazikani, B.; Soler, M.A.; Rocchia, W. Tuning Local Hydration Enables a Deeper Understanding of Protein-Ligand Binding: The PP1-Src Kinase Case. *J. Phys. Chem. Lett.* **2021**, *12*, 49–58. [[CrossRef](#)] [[PubMed](#)]
61. Sridhar, A.; Ross, G.A.; Biggin, P.C. Waterdock 2.0: Water placement prediction for Holo-structures with a pymol plugin. *PLoS ONE* **2017**, *12*, e0172743. [[CrossRef](#)]
62. Dileep, K.V.; Ihara, K.; Mishima-Tsumagari, C.; Kukimoto-Niino, M.; Yonemochi, M.; Hanada, K.; Shirouzu, M.; Zhang, K. Crystal structure of human acetylcholinesterase in complex with tacrine: Implications for drug discovery. *Int. J. Biol. Macromol.* **2022**, *210*, 172–181. [[CrossRef](#)]
63. Holm, L.; Rosenström, P. Dali server: Conservation mapping in 3D. *Nucleic Acids Res.* **2010**, *38*, W545–W549. [[CrossRef](#)]
64. Shindyalov, I.N.; Bourne, P.E. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Eng.* **1998**, *11*, 739–747. [[CrossRef](#)]
65. Orengo, C.A.; Taylor, W.R. SSAP: Sequential structure alignment program for protein structure comparison. *Methods Enzymol.* **1996**, *266*, 617–635. [[CrossRef](#)]
66. Gelly, J.C.; Joseph, A.P.; Srinivasan, N.; de Brevern, A.G. iPBA: A tool for protein structure comparison using sequence alignment strategies. *Nucleic Acids Res.* **2011**, *39*, W18–W23. [[CrossRef](#)]
67. Faure, G.; Joseph, A.P.; Craveur, P.; Narwani, T.J.; Srinivasan, N.; Gelly, J.C.; Rebehmed, J.; de Brevern, A.G. iPBAvizu: A PyMOL plugin for an efficient 3D protein structure superimposition approach. *Source Code Biol. Med.* **2019**, *14*, 5. [[CrossRef](#)]
68. Bima, A.I.H.; Elsamanoudy, A.Z.; Alghamdi, K.S.; Shinawi, T.; Mujalli, A.; Kaipa, P.R.; Aljeaid, D.; Awan, Z.; Shaik, N.A.; Banaganapalli, B. Molecular profiling of melanocortin 4 receptor variants and agouti-related peptide interactions in morbid obese phenotype: A novel paradigm from molecular docking and dynamics simulations. *Biologia* **2022**, *77*, 1481–1496. [[CrossRef](#)]
69. Anies, S.; Jallu, V.; Diharce, J.; Narwani, T.J.; de Brevern, A.G. Analysis of Integrin  $\alpha$ IIb Subunit Dynamics Reveals Long-Range Effects of Missense Mutations on Calf Domains. *Int. J. Mol. Sci.* **2019**, *23*, 858. [[CrossRef](#)] [[PubMed](#)]
70. Rahman, M.S.; Hoque, M.N.; Islam, M.R.; Akter, S.; Rubayet Ul Alam, A.; Siddique, M.A.; Saha, O.; Rahaman, M.M.; Sultana, M.; Crandall, K.A.; et al. Epitope-based chimeric peptide vaccine design against S, M and E proteins of SARS-CoV-2, the etiologic agent of COVID-19 pandemic: An in silico approach. *PeerJ* **2020**, *8*, e9572. [[CrossRef](#)] [[PubMed](#)]
71. Chatzou, M.; Magis, C.; Chang, J.M.; Kemena, C.; Bussotti, G.; Erb, I.; Notredame, C. Multiple sequence alignment modeling: Methods and applications. *Brief Bioinform.* **2016**, *17*, 1009–1023. [[CrossRef](#)]
72. Edgar, R.C.; Batzoglou, S. Multiple sequence alignment. *Curr. Opin. Struct. Biol.* **2006**, *16*, 368–373. [[CrossRef](#)]
73. Weigt, M.; White, R.A.; Szurmant, H.; Hoch, J.A.; Hwa, T. Identification of direct residue contacts in protein-protein interaction by message passing. *Proc. Natl. Acad. Sci. USA* **2009**, *106*, 67–72. [[CrossRef](#)]
74. Thompson, J.D.; Linard, B.; Lecompte, O.; Poch, O. A comprehensive benchmark study of multiple sequence alignment methods: Current challenges and future perspectives. *PLoS ONE* **2011**, *6*, e18093. [[CrossRef](#)]
75. Hogeweg, P.; Hesper, B. The alignment of sets of sequences and the construction of phyletic trees: An integrated method. *J. Mol. Evol.* **1984**, *20*, 175–186. [[CrossRef](#)]
76. Michel, M.; Skwark, M.J.; Menendez Hurtado, D.; Ekeberg, M.; Elofsson, A. Predicting accurate contacts in thousands of Pfam domain families using PconsC3. *Bioinformatics* **2017**, *33*, 28592866. [[CrossRef](#)]
77. Morcos, F.; Pagnani, A.; Lunt, B.; Bertolino, A.; Marks, D.S.; Sander, C.; Zecchina, R.; Onuchic, J.N.; Hwa, T.; Weigt, M. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *J. Chem. Inf. Model.* **2011**, *108*, E1293–E1301. [[CrossRef](#)]
78. Morcos, F.; Jana, B.; Hwa, T.; Onuchic, J.N. Coevolutionary signals across protein lineages help capture multiple protein conformations. *Proc. Natl. Acad. Sci. USA* **2013**, *110*, 20533–20538. [[CrossRef](#)] [[PubMed](#)]
79. Jarmolinska, A.I.; Zhou, Q.; Sulkowska, J.I.; Morcos, F. DCA-MOL: A PyMOL Plugin To Analyze Direct Evolutionary Couplings. *J. Chem. Inf. Model.* **2019**, *59*, 625–629. [[CrossRef](#)] [[PubMed](#)]
80. Dos Santos, R.N.; Ferreira, L.G.; Andricopulo, A.D. Practices in Molecular Docking and Structure-Based Virtual Screening. *Methods Mol. Biol.* **2018**, *1762*, 31–50. [[CrossRef](#)] [[PubMed](#)]
81. Rosignoli, S.; Paiardini, A. DockingPie: A consensus docking plugin for PyMOL. *Bioinformatics* **2022**, *38*, 4233–4234. [[CrossRef](#)] [[PubMed](#)]
82. Koes, D.R.; Baumgartner, M.P.; Camacho, C.J. Lessons learned in empirical scoring with smina from the CSAR 2011 benchmarking exercise. *J. Chem. Inf. Model.* **2013**, *53*, 1893–1904. [[CrossRef](#)] [[PubMed](#)]
83. Eberhardt, J.; Santos-Martins, D.; Tillack, A.F.; Forli, S. AutoDock Vina 1.2.0: New Docking Methods, Expanded Force Field, and Python Bindings. *J. Chem. Inf. Model.* **2021**, *61*, 3891–3898. [[CrossRef](#)]
84. Ruiz-Carmona, S.; Alvarez-Garcia, D.; Foloppe, N.; Garmendia-Doval, A.B.; Juhos, S.; Schmidtke, P.; Barril, X.; Hubbard, R.E.; Morley, S.D. rDock: A fast, versatile and open source program for docking ligands to proteins and nucleic acids. *PLoS Comput. Biol.* **2014**, *10*, e1003571. [[CrossRef](#)]
85. Morley, S.D.; Afshar, M. Validation of an empirical RNA-ligand scoring function for fast flexible docking using Ribodock. *J. Comput. Aided Mol. Des.* **2004**, *18*, 189–208. [[CrossRef](#)]
86. Ravindranath, P.A.; Forli, S.; Goodsell, D.S.; Olson, A.J.; Sanner, M.F. AutoDockFR: Advances in Protein-Ligand Docking with Explicitly Specified Binding Site Flexibility. *J. PLoS Comput. Biol.* **2015**, *11*, e1004586. [[CrossRef](#)]

87. Houston, D.R.; Walkinshaw, M.D. Consensus docking: Improving the reliability of docking in a virtual screening context. *J. Chem. Inf. Model.* **2013**, *53*, 384–390. [[CrossRef](#)]
88. Palacio-Rodríguez, K.; Lans, I.; Cavasotto, C.N.; Cossio, P. Exponential consensus ranking improves the outcome in docking and receptor ensemble docking. *Sci. Rep.* **2019**, *9*, 5142. [[CrossRef](#)] [[PubMed](#)]
89. Brenke, R.; Kozakov, D.; Chuang, G.Y.; Beglov, D.; Hall, D.; Landon, M.R.; Mattos, C.; Vajda, S. Fragment-based identification of druggable ‘hot spots’ of proteins using Fourier domain correlation techniques. *Bioinformatics* **2009**, *25*, 621–627. [[CrossRef](#)] [[PubMed](#)]
90. Teixeira, O.; Lacerda, P.; Froes, T.Q.; Nonato, M.C.; Castilho, M.S. Druggable hot spots in trypanothione reductase: Novel insights and opportunities for drug discovery revealed by DRUGpy. *J. Comput. Aided Mol. Des.* **2021**, *35*, 871–882. [[CrossRef](#)] [[PubMed](#)]
91. Tang, L.; Gamal El-Din, T.M.; Swanson, T.M.; Pryde, D.C.; Scheuer, T.; Zheng, N.; Catterall, W.A. Structural basis for inhibition of a voltage-gated Ca<sup>2+</sup> channel by Ca<sup>2+</sup> antagonist drugs. *Nature* **2016**, *537*, 117–121. [[CrossRef](#)]
92. Williams, J.C.; Kalyanamoorthy, S. PoseFilter: A PyMOL Plugin for filtering and analyzing small molecule docking in symmetric binding sites. *Bioinformatics* **2021**, *37*, 3367–3368. [[CrossRef](#)]
93. Kagami, L.P.; das Neves, G.M.; Timmers, L.F.S.M.; Caceres, R.A.; Eifler-Lima, V.L. Geo-Measures: A PyMOL plugin for protein structure ensembles analysis. *J. Comput. Biol. Chem.* **2020**, *87*, 107322. [[CrossRef](#)]
94. McGibbon, R.T.; Beauchamp, K.A.; Harrigan, M.P.; Klein, C.; Swails, J.M.; Hernández, C.X.; Schwantes, C.R.; Wang, L.P.; Lane, T.J.; Pande, V.S. MDTraj: A Modern Open Library for the Analysis of Molecular Dynamics Trajectories. *Biophys. J.* **2015**, *109*, 1528–1532. [[CrossRef](#)]
95. Ghosh, A.; Sarmah, P.; Patel, H.; Mukerjee, N.; Mishra, R.; Alkahtani, S.; Varma, R.S.; Baishya, D. Nonlinear molecular dynamics of quercetin in *Gynocardia odorata* and *Diospyros malabarica* fruits: Its mechanistic role in hepatoprotection. *PLoS ONE* **2022**, *17*, e0263917. [[CrossRef](#)]
96. Mukerjee, N.; Das, A.; Maitra, S.; Ghosh, A.; Khan, P.; Alexiou, A.; Dey, A.; Baishya, D.; Ahmad, F.; Sachdeva, P.; et al. Dynamics of natural product Lupenone as a potential fusion inhibitor against the spike complex of novel Semliki Forest Virus. *PLoS ONE* **2022**, *17*, e0263853. [[CrossRef](#)]
97. Zinovjev, K.; van der Kamp, M.W. Enlighten2: Molecular dynamics simulations of protein-ligand systems made accessible. *Bioinformatics* **2020**, *36*, 5104–5106. [[CrossRef](#)]
98. Ponder, J.W.; Case, D.A. Force fields for protein simulations. *Adv. Protein Chem.* **2003**, *66*, 27–85. [[CrossRef](#)] [[PubMed](#)]
99. Rice, K.; Batul, K.; Whiteside, J.; Kelso, J.; Papinski, M.; Schmidt, E.; Pratasouskaya, A.; Wang, D.; Sullivan, R.; Bartlett, C.; et al. The predominance of nucleotidyl activation in bacterial phosphonate biosynthesis. *Nat. Commun.* **2019**, *10*, 3698. [[CrossRef](#)]
100. Wang, L.; Parnell, A.; Williams, C.; Bakar, N.A.; Challand, M.R.; van der Kamp, M.W.; Simpson, T.J.; Race, P.R.; Crump, M.P.; Willis, C.L. A Rieske oxygenase/epoxide hydrolase-catalysed reaction cascade creates oxygen heterocycles in mupirocin biosynthesis. *Nat. Catal.* **2018**, *1*, 968–976. [[CrossRef](#)]
101. Byrne, M.J.; Lees, N.R.; Han, L.C.; van der Kamp, M.W.; Mulholland, A.J.; Stach, J.E.; Willis, C.L.; Race, P.R. The Catalytic Mechanism of a Natural Diels-Alderase Revealed in Molecular Detail. *J. Am. Chem. Soc.* **2018**, *138*, 6095–6098. [[CrossRef](#)]
102. Drulyte, I.; Obajdin, J.; Trinh, C.H.; Kalverda, A.P.; van der Kamp, M.W.; Hemsworth, G.R.; Berry, A. Crystal structure of the putative cyclase IdmH from the indanomycin nonribosomal peptide synthase/polyketide synthase. *IUCr* **2019**, *6*, 1120–1133. [[CrossRef](#)] [[PubMed](#)]
103. Wang, J.; Wang, W.; Kollman, P.A.; Case, D.A. Automatic atom type and bond type perception in molecular mechanical calculations. *J. Mol. Graph. Model.* **2006**, *25*, 247–260. [[CrossRef](#)] [[PubMed](#)]
104. Olsson, M.H.; Søndergaard, C.R.; Rostkowski, M.; Jensen, J.H. PROPKA3: Consistent Treatment of Internal and Surface Residues in Empirical pKa Predictions. *J. Chem. Theory Comput.* **2011**, *7*, 525–537. [[CrossRef](#)]
105. Warkentin, R.; Kwan, D.H. Resources and Methods for Engineering “Designer” Glycan-Binding Proteins. *Molecules* **2021**, *26*, 380. [[CrossRef](#)]
106. Thakker, R.V. Multiple endocrine neoplasia type 1 (MEN1). *Best Pract. Res. Clin. Endocrinol. Metab.* **2010**, *24*, 355–370. [[CrossRef](#)]
107. Jelsch, C.; Mourey, L.; Masson, J.M.; Samama, J.P. Crystal structure of *Escherichia coli* TEM1 beta-lactamase at 1.8 Å resolution. *Proteins* **1993**, *16*, 364–383. [[CrossRef](#)]
108. Ross, C.; Nizami, B.; Glenister, M.; Sheik Amamuddy, O.; Atilgan, A.R.; Atilgan, C.; Tastan Bishop, Ö. MODE-TASK: Large-scale protein motion tools. *Bioinformatics* **2018**, *34*, 3759–3763. [[CrossRef](#)] [[PubMed](#)]
109. Rajpoot, S.; Kumar, A.; Zhang, K.Y.J.; Gan, S.H.; Baig, M.S. TIRAP-mediated activation of p38 MAPK in inflammatory signaling. *Sci. Rep.* **2022**, *12*, 5601. [[CrossRef](#)] [[PubMed](#)]