A web-based docking interface for the bench chemist

- A good tool for communicating structure-based ligand design

*Anders Poulsen & Leon Leow, S*BIO Pte Ltd*

Introduction

As the number of x-ray structures of drugable targets increases, more bench chemists are venturing into the area of structure-based design. However, interpreting docking requires expertise in molecular modeling which is not always available to the scientist. Often the docking score alone is useless as a measure for how well a potential inhibitor will bind to the target protein, and a visual inspection of the docked poses is essential.

Our objective was to build a Web-based docking interface using freely available java applets and Schrödinger's Glide [1] as a docking engine. The Interface should be easy to use and tools for evaluating the docked poses should be readily accessible. The user should be able to access the results by a hyperlink, making it easy to present and communicate structure-based design.

Workflow

The infrastructure is built in Perl/CGI except, for the docking result viewer which is written in HTML and Java script. The input structure is attached as a mol-file and the user defined settings for ligand preparation and docking are passed from the input form to the CGI script. This script will make a unique job directory and copy files used for the job here. A Perl script is called that submits 4 jobs:

1. Ligand preparation using Schrödinger's Ligprep [1] with the user defined settings
2. Glide docking with the user defined settings
3. File conversion using Ligprep
4. Chopping up the SD-file into individual mol-files using a Perl script

The Perl script will wait for the jobs to finish and then load the docking result viewer. Figure 1 gives an overview of the workflow, the files, programs and applets used in the docking.

We use an Apache web-server running on an SGI Irix machine.

However, most web-servers running on Linux are compatible with our setup.

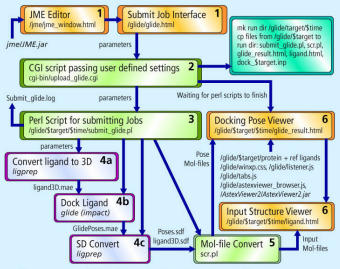

Figure 1: Workflow for a docking job. Orange boxes: HTML pages. Green boxes: Perl scripts. Purple boxes and italics: Programs and applets. Paths are relative to the htdocs root directory.

Setting up Targets

Each target is set up by a computational chemist. Protein structures are prepared as recommended by Schrödinger. Grid files are calculated including proper constraints like essential hydrogen bonds or metal interactions. A job file is saved for each combination of the user defined options: van der Waal scaling and penalise cis amide. The user will not be able to switch the constraints off, and the docking will fail if the constraints are not fulfilled.

Reference ligands are selected. These are usually the reference ligand(s) used in the enzymatic assay and co-crystallised ligands from publicly available protein structures. They are either docked into the target or superimposed onto the binding site.

Submitting Docking Jobs

The interface was designed to be as simple as possible while still enabling the user to maintain some control of the docking and ligand preparation by pull-down menus. The user must select a target and supply a valid structure. All other controls are optional.

The structure can be submitted as a mol-file or the user can draw the structure with the freely available JME Java applet [2] . As default the tautomerizer and ionizer functions of Ligprep are run to produce physiologically relevant tautomer and ionization states of the ligand, but the user has the option to switch these off, thereby decreasing the docking time. Figure 2 displays this page as well as the JME applet.

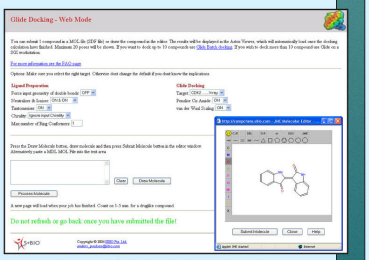

Figure 2: Web page for submitting the docking job. The user must supply a valid input structure and select a target. All other settings are optional.

Viewing the Docking Results

The docking result viewer is an HTML page displaying the AstexViewer [3] Java applet. The user can manipulate the viewer through controls on 4 separate tabs. Each control executes a function written in Java script. This page loads automatically when the docking job is finished with the "Poses" tab displayed as default.

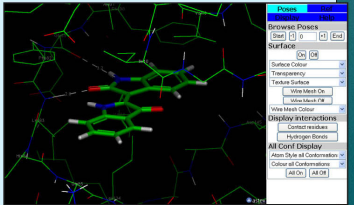

Figure 3: The docking result viewer displaying the "Poses" tab. Hydrogen bonds and contact residues can be displayed by the push of a button.

On this tab the controls for browsing through the docked poses is located on top. Below are other controls relating to the docked poses (Figure 3). The "Ref" tab contains controls for manipulating the target protein as well as reference ligands (Figure 4 & 5).

Evaluating Docked Poses

Our main objective was to write an interface which makes it easy to evaluate docking results. These controls are located on the "Poses" and "Ref" tabs. Hydrogen bonds and distances can be displayed for the active pose as well as contact residues (Figure 3). Displaying surfaces on the protein binding site as well as the pose is helpful in determining how complementary the ligand is to the receptor and how well it fills out the binding site (Figure 4). The surfaces can be displayed as solid, transparent or mesh.

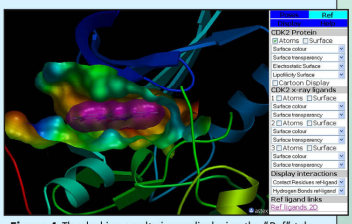

Figure 4: The docking result viewer displaying the "Ref" tab. Surfaces can be displayed by the push of a button

Medicinal chemists often think of protein-ligand interactions in terms of pharmacophores. A very useful feature of our docking viewer interface is that the user can display reference ligands containing these pharmacophore elements and compare the binding mode to that of the docked poses. This is shown in Figure 5 with the protein turned off. Displaying all reference ligands simultaneously is like superimposing the docked pose on a pharmacophore model.

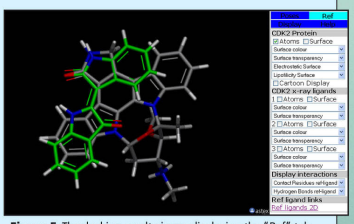

Figure 5: The docking result viewer displaying the "Ref" tab. The docked poses can be compared to the binding mode of reference ligands by the push of a button

The Display & Help Tabs

The "Display" tab contains controls for the AstexViewer applet as well as for more advanced graphics controls for the protein, docked poses and reference ligands. Since the graphics rendering of the applet is excellent, these are useful for displaying fancy graphics for screen-dumps.

In the "Help" tab information on shortcut keys and links to help files, log files and the Glide score are found. The output structures from Ligprep can also be opened in a separate window and displayed in the AstexViewer applet. The name of the job directory is displayed here as well as a link to the page itself. Figure 6 displays the "Help" tab with the Glide score and input structures windows open.

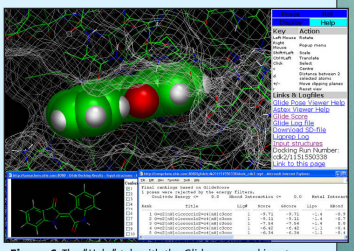

Figure 6: The "Help" tab with the Glide score and input structures windows open

Communicating Structure-Based Design

The docking viewer interface is copied to the unique job directory on the intranet server. The link to the docking viewer interface can be copied and pasted into any document such as a PowerPoint presentation or an e-mail. This makes it easy to share docking results with colleagues and present structure-based design at project meetings. The only requirement is access to S*BIO's intranet.

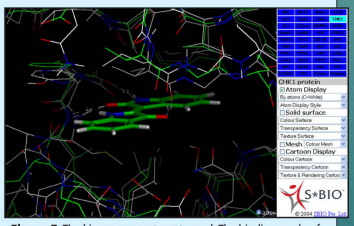

Figure 7: The kinase cancer target panel. The binding mode of one target can quickly be compared to other targets.

The docking job number, i.e. the job directory, can be used to access the docking results from other viewers. Figure 7 shows our cancer target panel. All cancer related kinase targets available from the Protein Data Bank were superimposed and poses docked into one target can be visualised in other targets. This is a fast way of determining potential targets of interest and accessing selectivity. The selectivity panel is opened simply by pasting the docking job number into a webpage which calls a Perl script. The script substitutes the path variable for the docked poses and writes out the viewer as a new webpage. We also built a viewer that lets the user compare docking results from several targets again just by pasting in the docking job numbers. These selectivity panels require the targets to be superimposed on a template before they are set up for docking.

Conclusion

We have built a Web-based docking interface that is intuitive and simple to use. The docking viewer contains tools that lets the user evaluate the docking results visually simply by the push of a button. The AstexViewer java applet is used because of its high quality graphics display, excellent functionality and its simple Java scripting language. We use the docking viewer as a tool for communicating structure-based design within the company. Docking results can be shared with anyone who has access to our intranet or integrated into PowerPoint presentations.

References

1. Schrödinger, 101 SW Main Street, Suite 1300, Portland, OR 97204, www.schrodinger.com.
2. Peter Ertl, Novartis Institutes for BioMedical Research, Basel, Switzerland, peter.ertl@pharma.novartis.com.
3. Astex Therapeutics Limited, 436 Cambridge Science Park, Milton Road, Cambridge CB4 0QA, United Kingdom, www.astex-therapeutics.com.