Protein Flexibility Analysis: a Friendly Interface

Yang Li, Smith College, Northampton, MA
Faculty Adviser: Ileana Streinu, Computer Science Department, Smith College

Introduction

Motivation

- Proteins are flexible structures that exhibit internal motions.
- Protein flexibility analysis helps scientists understand protein function and facilitates new therapeutics for diseases.

About RigDyn

- A research software package for flexibility analysis of general structures, such as proteins.
- Uses rigidity theory to provide information on rigid regions of proteins.

Project Goal

Problems

- The first version of RigDyn was difficult to use for non-computer scientists.
- It was command-line driven and had to be invoked by manual editing of configuration file
- Analysis output was not easily comprehensible for human viewers.

Solution

- Design a user-friendly interface that integrates input configurations and analysis outputs.
- Maximize the usability of RigDyn for users at all levels

Interface Overview

- A web-based GUI (Graphical User Interface) primarily designed for protein flexibility analysis.
- Runs on a server, thus avoids the hassle of installation.
- Focuses on three aspects: the front-end, the viewer and the user interaction.

The Front-end

The front-end allows users to invoke RigDyn with a few clicks. Different inputting and running options are organized into expandable sections.

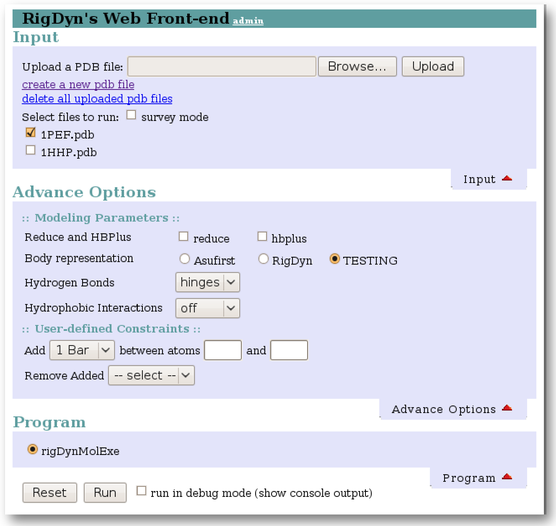

Fig 1. Screenshot of RigDyn Web Front-end (before running)

The Viewer

Numerical analysis outputs are transformed into graphical representations via Jmol, a 3D biomolecule viewer. The viewer is accompanied by interactive controls that can manipulate the visualization based on user's needs.

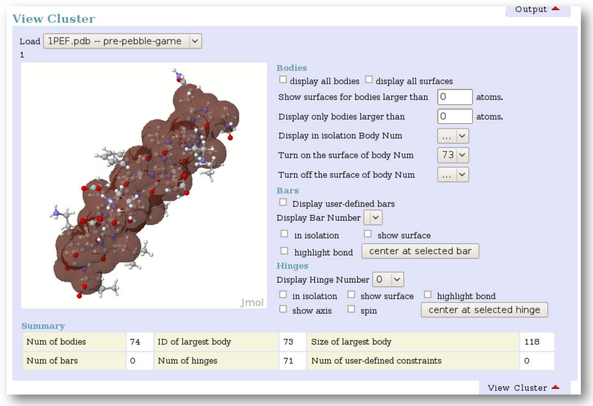

Fig 2. Screenshot of View Cluster section

Peptide f (PDB ID: 1PEF) displayed in Jmol Viewer. The largest rigid clusters is highlighted in brown. Interactive viewing options involving rigid clusters are shown on the right.

User Interaction

- Flexibility analysis can be easily re-run with adjusted configurations.
- Users may add constraints between a pair of atoms, and observe how rigidity structures change from previous results.

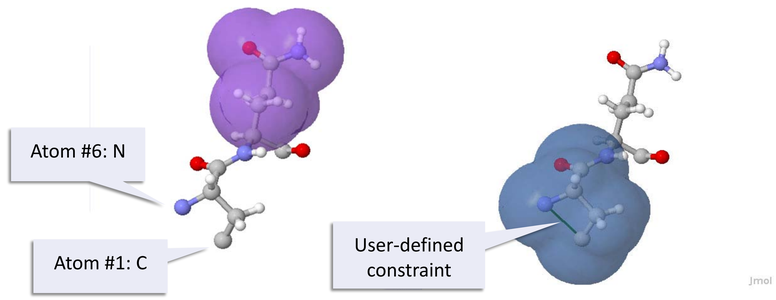

Fig 3. Example of a user-defined constraint

Left : A section of HIV-1 protease (PDB ID: 1HHP) that is missing a bond between Atom 1 (Carbon) and Atom 6 (Nitrogen), where a five-ring should have existed. The colored surface represents the largest rigid cluster.

Right: The same protein with the addition of a 5-bar constraint (shown as a green bar) between Atom 1 and Atom 6, resulting in a larger rigid cluster (shown in blue) around the ring.

Future Work

- Continue to enhance user-friendliness
- Test and verify the viewer with a larger data set.
- Stabilize its compatibility with different web browsers to prepare for actual release.

Acknowledgements

Thanks to Professor Ileana Streinu and graduate students Naomi Fox and Filip Jagodzinski who guided me through this project.

This work was initiated as part of the Smith Summer Research Fellows (SURF) Program at The Clark Science Center in Summer 2008, and continued as a NSF-funded research internship through the academic year of 2008-2009.