EasyModeller 4.0 : A new GUI to MODELLER

Background

Homology modeling has become a key component in structural bioinformatics for prediction of the three-dimensional structure of proteins from their sequences due to availability of huge amount of protein sequence data and the growing number of known structures. It becomes even more useful when constraints from X-ray diffraction or NMR are not yet available. Many tools have been developed for homology modeling out of which the most popular and widely used is MODELLER developed by Prof. A.Sali and co-workers. MODELLER stands apart from other packages due to its free availability, powerful features and reliable results. But most users find it a bit difficult to start with MODELLER as it is command line based and requires knowledge of basic Python scripting to use it efficiently. A freely available GUI for this great package would thus be very helpful to exploit the powers of this homology modeling package by everyone. EasyModeller is a frontend graphical interface to MODELLER developed using Perl/Tk, which can be used as a standalone tool in windows platform with MODELLER and Python preinstalled. The aim of this tool is to help inexperienced users as well as regular users to perform modeling, assessment, visualization, and optimization of protein models in a simple and straightforward way. EasyModeller can produce 3D structural models of proteins from sequence and given template(s) information using MODELLER in backend. A detailed help is provided for every step with robust error handling.

Previous Versions

EasyModeller 1.0 : A basic Windows based GUI to modeller with simple features
EasyModeller 2.0 : A windows based GUI with robust error handling and a much user friendly GUI
EasyModeller 2.1 : EasyModeller 2.0 with BioEdit support for Alignment Editing
EasyModeller 3.0 : A LINUX based GUI to Modeller introducing "on the fly" code editing

Publication Link : EasyModeller: A graphical interface to MODELLER

EasyModeller 4.0

EasyModeller 4.0 introduces a fresh new GUI for Homology Modeling using MODELLER in the backend and available for both Windows and Linux platform. This version has several new features integrating all the goodies of EasyModeller 3.0 which was only available for LINUX.

The highlighted features of this version are :

1. Tab based logical Modeling steps with extensive error handling
2. Allows to load unlimited number of templates
3. A colorful alignment viewer and also with an inbuilt alignment editor
4. On the fly MODELLER code editing (generated MODELLER scripts can be edited as per user's need and run from within the tool). This feature would be most useful for advanced MODELLER users.
5. Inbuilt DOPE profile Viewer, Ramachandran Plot viewer, Loop Modeling, Basic model optimization and dynamics for a selected Model.

Installation

Windows :

EasyModeller requires the following softwares to be installed in your system to run

1. Modeller (Freely available at salilab.org)
2. Compatible Python version (Freely available at www.python.org)

For older versions of Modeller use Python 2.5 or 2.6
For latest Modeller use Python 2.7
Python 3.x is not supported by EasyModeller

N.B:

1. If you have Python installed and was still asked with a popup box stating "Windows cannot find a software for test.py" then probably you have not associated the .py files to Python
2. Make sure that you have installed Modeller in the default location i.e, C:\\Program Files
3. Windows7 / Vista users may need to run the application by Right clicking on the application and select "Run as Administrator"

If you still face any errors you can always mail me at kuntal.bhusan@gmail.com

Linux

For this version of EasyModeller there are the following simple requisites :

1. A Linux operating system (Fedora, Ubuntu, etc.)
2. Python : Will be installed by default in all Linux systems.
3. Modeller : To be installed (preferably the latest version).
4. Rasmol / Pymol / Jmol : Any of the three PDB viewer should be in the path.

Download the appropriate Modeller from here (For Fedora and other Red Hat based Linux : Linux (32-bit RPM) and For Ubuntu and other Debian based OS : Linux (http://salilab.org/modeller/9.10/modeller_9.10-1_i386.deb) .

Obtain the Modeller key from Here

Installation for Linux (RPM) : Fedora

Install the RPM file by running the following command (either logging in as the root user, or by prepending the command with 'sudo'), replacing XXXX with the Modeller license key (and i386 with ia64 or x86_64 if necessary).

```
env KEY_MODELLER=XXXX rpm -Uvh modeller-9.10-1.i386.rpm
```

Installation for Linux (Debian) : Ubuntu

Install the package by running the following command, replacing XXXX with the Modeller license key (and i386 with x86_64 if you are using the 64-bit installer).

```
sudo env KEY_MODELLER=XXXX dpkg -i modeller_9.10-1_i386.deb
```

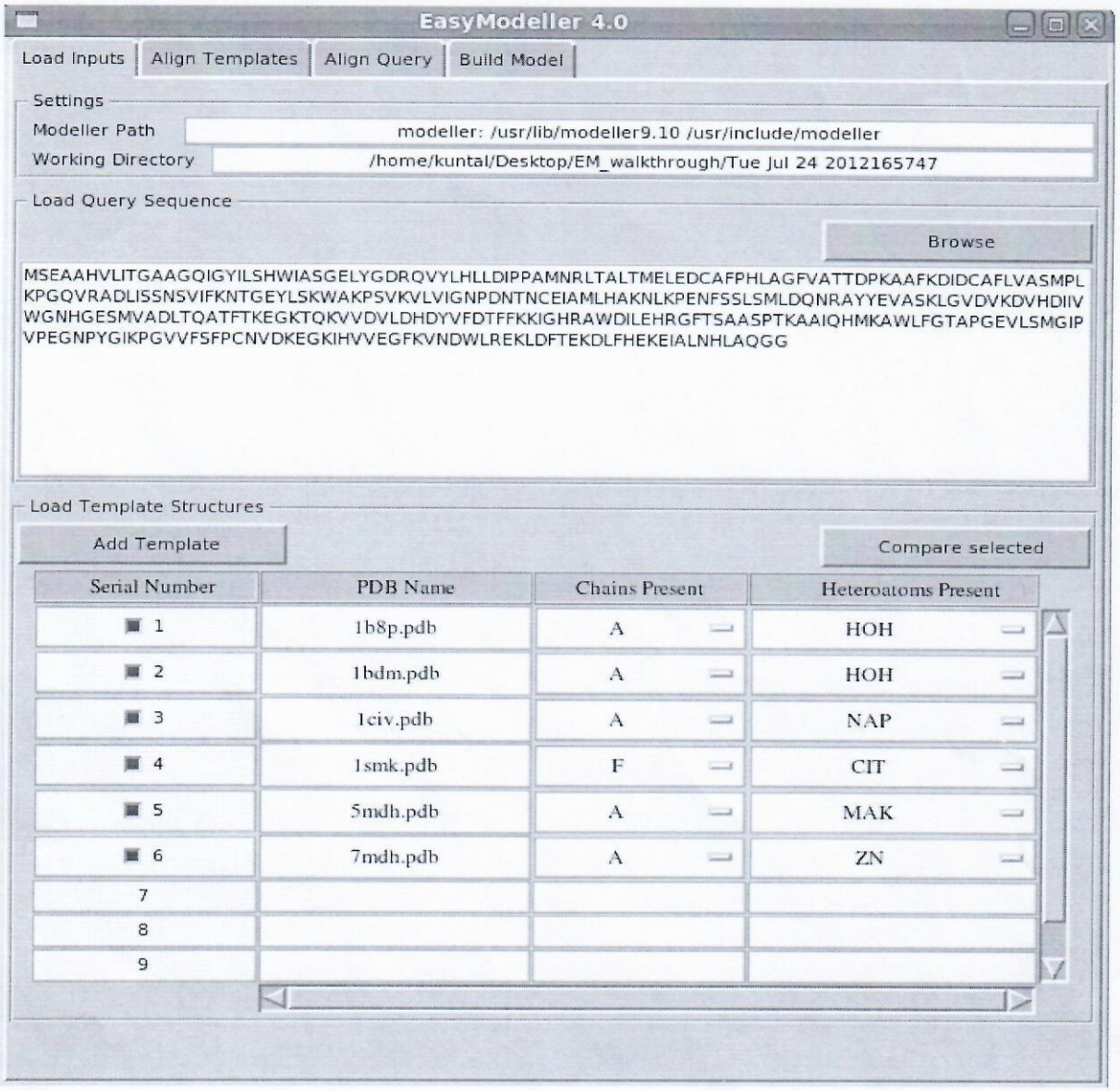
Step by Step Tutorial for using EasyModeller 4.0

Step 1 : Loading Query Sequence and its Templates

The Query sequence and the template can be easily loaded using the first tab titled "Load Inputs". The template sequence can either be loaded in a Fasta formatted file or can be pasted in the text box titled "Load Query Sequence". It should be kept in mind that the sequence should not contain any gaps.

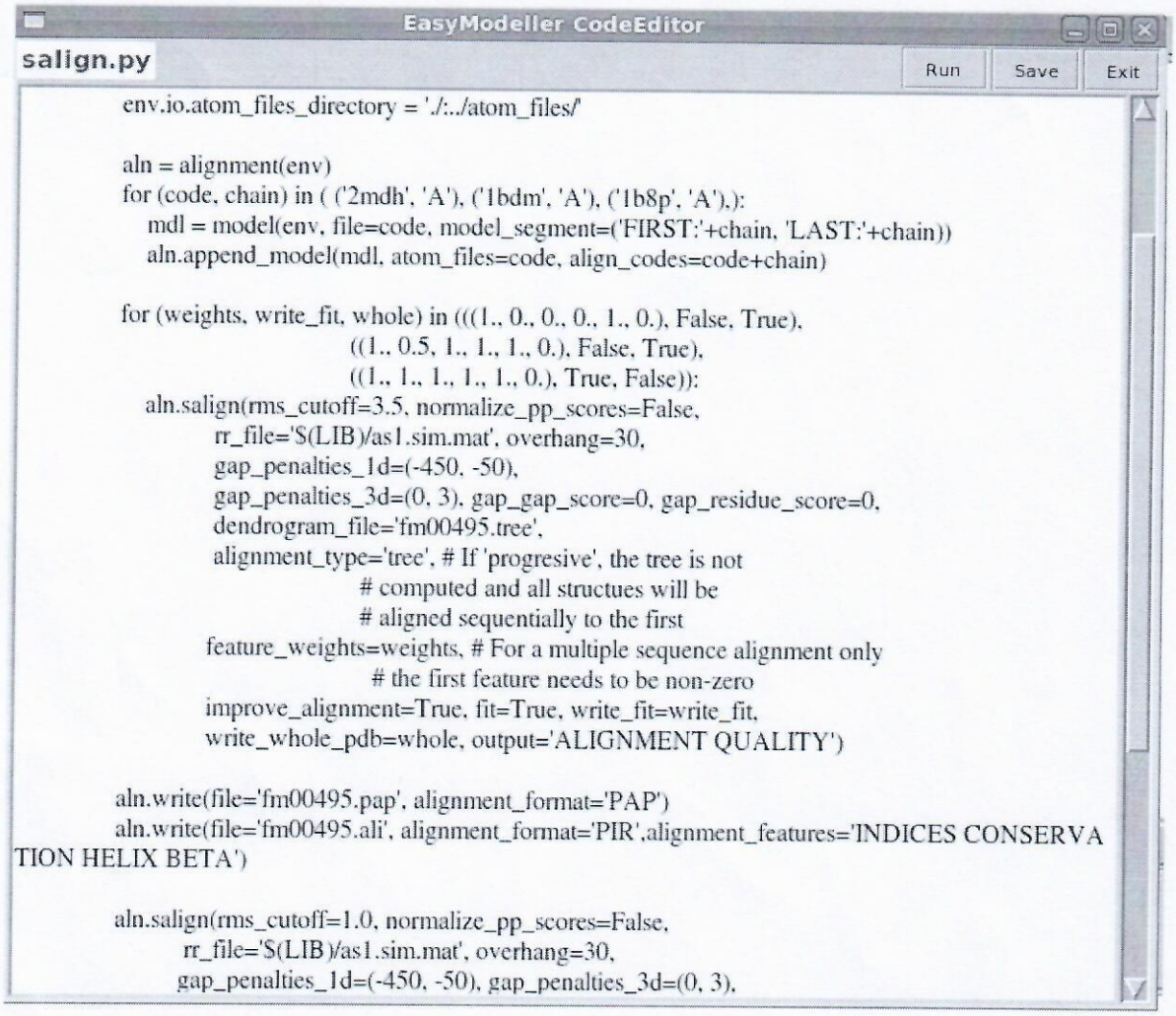
The appropriate templates for a query sequence (as obtained after performing a BLAST search against the PDB sequence database and choosing upon a set of templates by looking at the E-value of the search as well as the alignment length) can be loaded under "Load Template Structures" using the "Add Template" button.

A set of selected templates can also be compared using the "Compare selected" button after selecting the checkboxes corresponding to a sample.

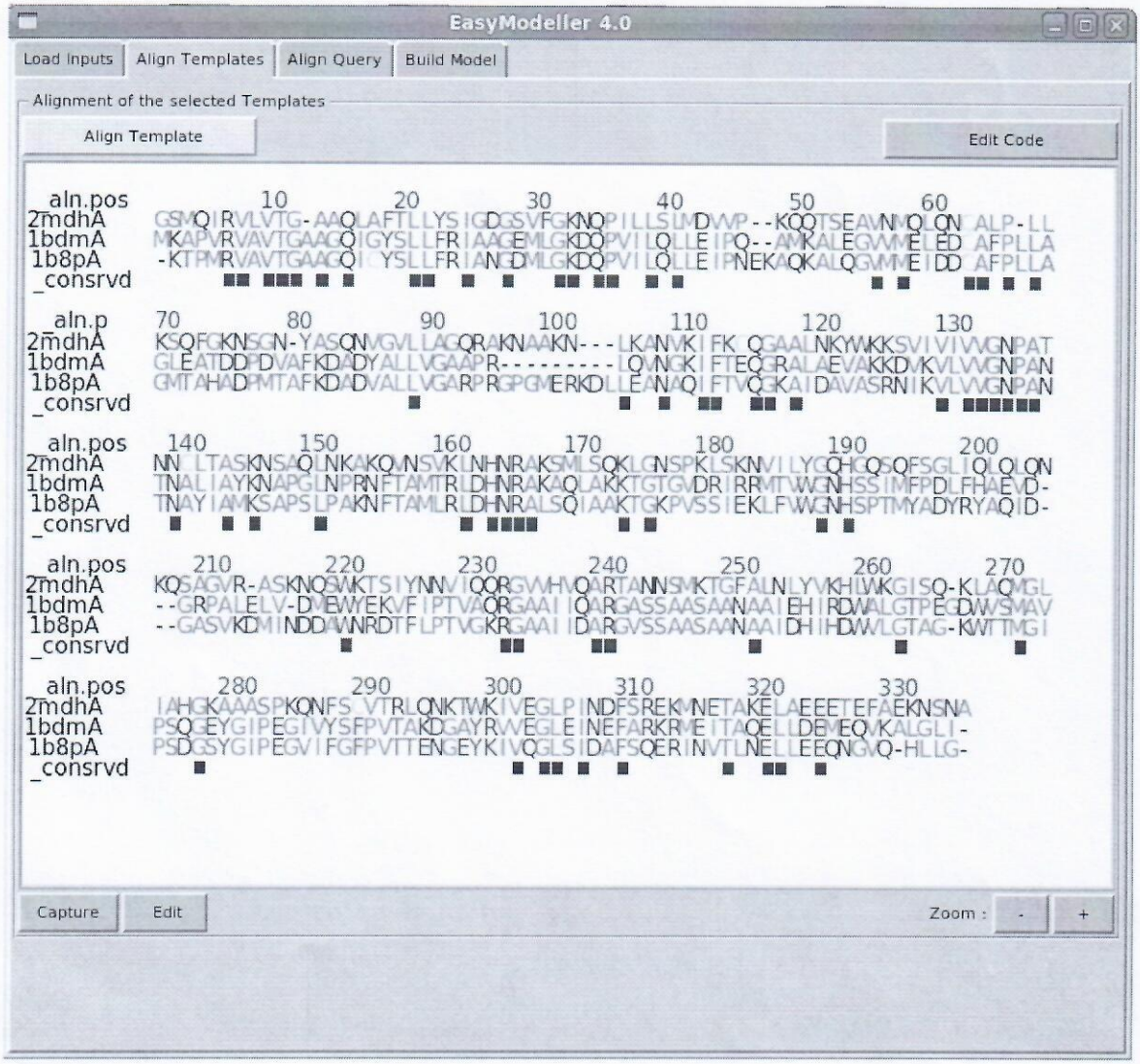


Step 2 : Aligning the templates

The selected templates can be aligned in the tab titled "Align templates" by selecting the "Align Template" button. This will implement the *salign* command of MODELLER and display the alignment in the canvas window. The amino acids in the alignment will be colored as per similarity and the conserved residues in the alignment will be marked by a red square below the alignment.

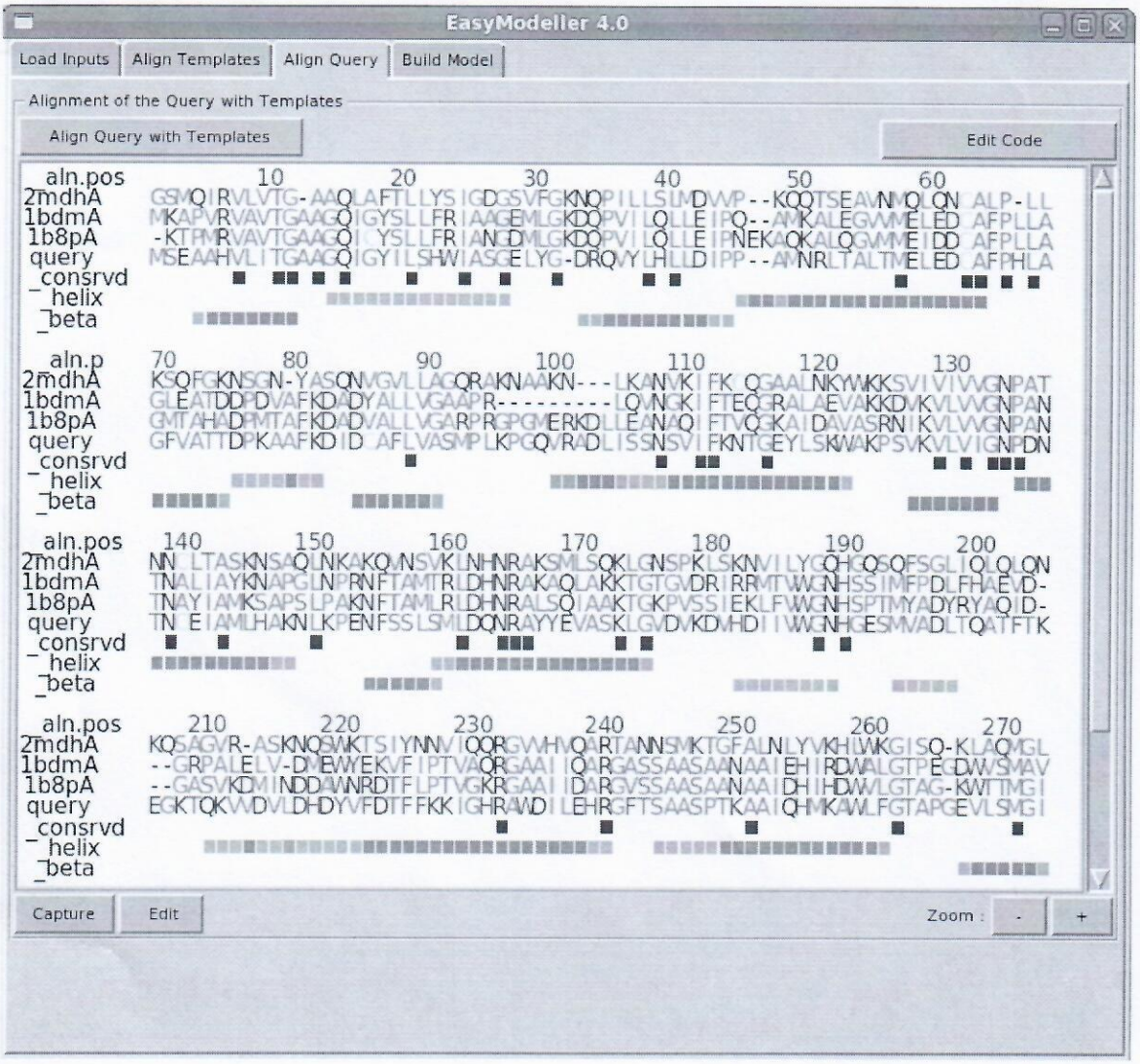


The code for the alignment can be seen by using the button "Edit Code". This will display a new window with the corresponding code which can be edited on the fly and the "Run" feature can be used to run the edited code. The generated alignment can be zoomed in and out and can also be saved as a postscript image file. The "Edit Alignement" feature can be used to edit the generated alignment.



Step 3 : Aligning the Query sequence with the Templates

The Query sequence can now be aligned to the templates in the tab titled "Align Query". The "Align Query with Templates" button can be used for performing the alignment. As discussed in Step 2, the MODELLER code for this step can also be edited and modified on the fly to generate user defined alignments. The alignment is displayed in the canvas window with the conserved residues in the alignment marked by red squares. The predicted secondary structures in the alignment namely alpha helices and beta sheets can be seen along with their occurrence probability indicated as a function of color such that a deeper shade of red indicates a higher confidence and a deeper shade of green represents a lower confidence. Ideally the alignment should be such that there should be no gaps present when there is a highly probable secondary structure predicted in that stretch of alignment.



Step 4 : Generating the Model

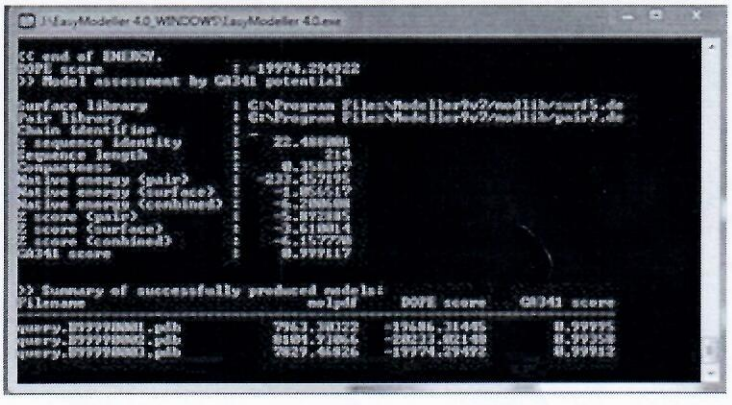
The next and final step is generating the model. This can be done from the "Build Model" tab using the feature "Generate Model". When this feature is selected the user is prompted to set the number of models to generate, whether to include the selected hetero-atom (only for single template modeling) and whether to perform an automatic loop modeling on the generated models. The Modeller output can be seen in the command prompt (or terminal window).The code for generating a model can be edited easily using the "Edit Code" feature and the "Run" option can be used to implement and run the new edited Modeller code from within the tool.

EasyModeller CodeEditor

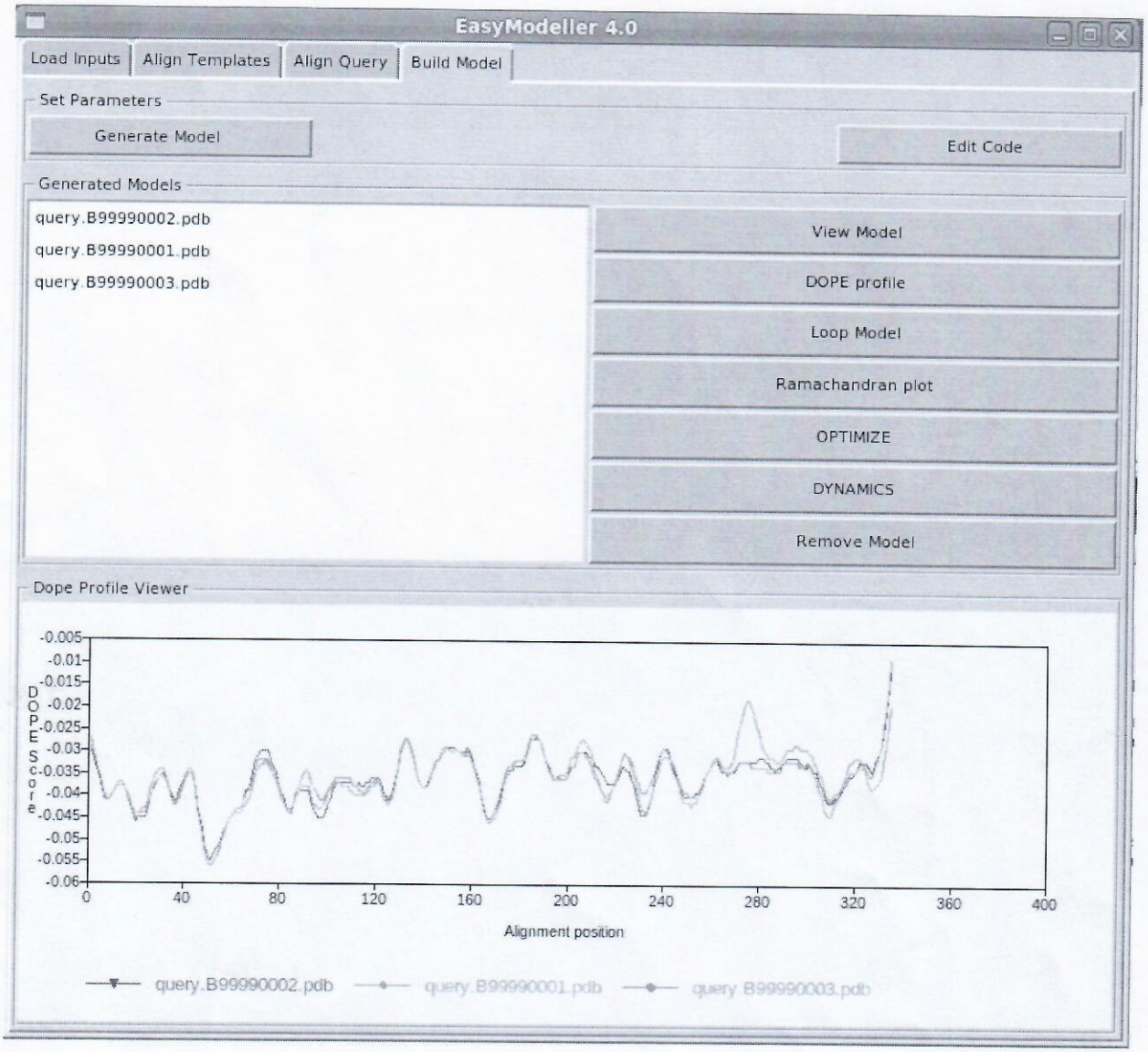
model-mult.py

Run Save Exit

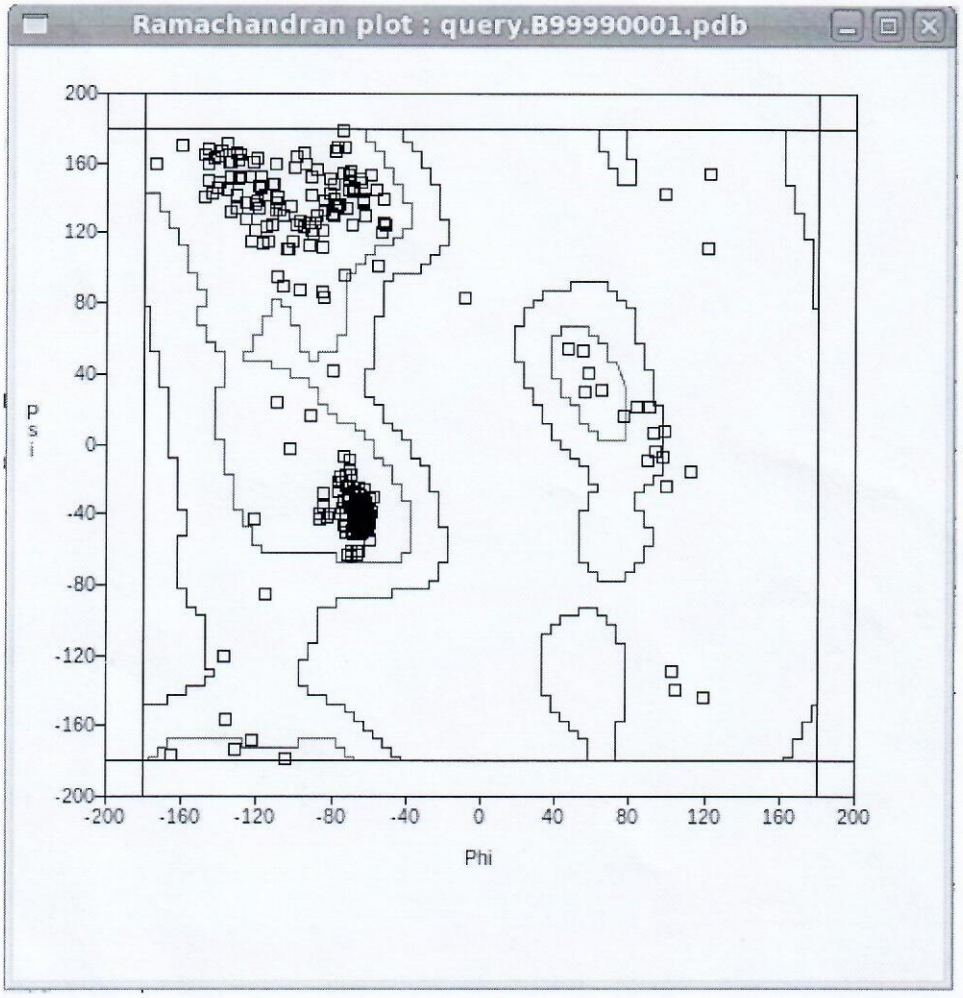
```
import sys
from modeller import *
from modeller.automodel import *

try:
        log.minimal()
        env = environ()
        a = automodel(env, alnfile='query-mult.ali',
                knowns=( ('2mdhA'), ('1bdmA'), ('1b8pA'),), sequence='query',assess_methods=
(assess.DOPE, assess.GA341))
        a.starting_model = 1
        a.ending_model = 3
        a.make()
except:
   FILE = open('run_status',"w")
   FILE.write(str(sys.exc_info()[0]))
   FILE.close()
   print "Unexpected error:", sys.exc_info()[0]
```



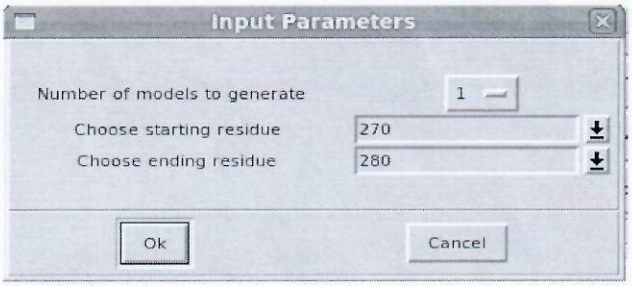
The generated models appear in the box "Generated Models" and can be manipulated using a suite of features. The DOPE profile viewer can be used to generate a DOPE profile plot of the generated models or a superimposed plot of the generated model with the template which can be chosen from a popup box that would allow to choose the structures for generating the plot. The generated plot can be zoomed using mouse selection.

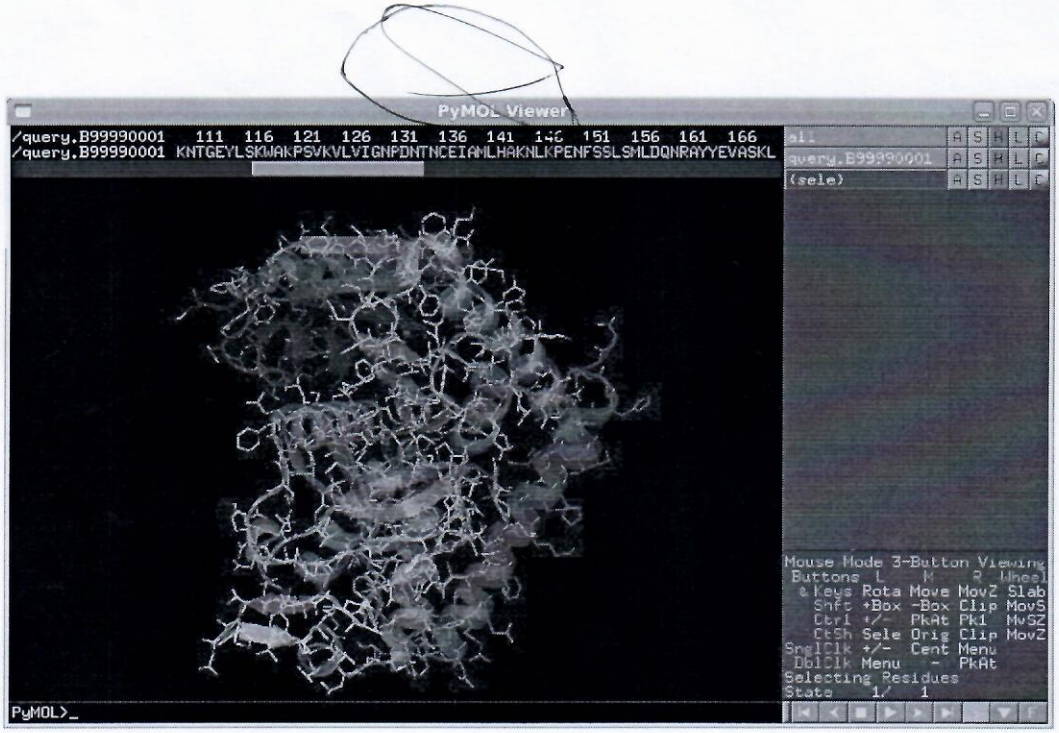


A Ramachandran plot can be generated for a model selected in the box. The plot can be zoomed using mouse selection and mouse over on a dot indicates its residue number and name. The red boundary forms the core region and the blue boundary forms the allowed region of the plot. A good model should have more than 90% residues in the allowed region.



The Loop model feature can be used to model a loop in the model by selecting the starting and ending residue of the loop. The generated models can also be optimized using the "Optimize" feature which allows the user to perform simple conjugate gradient and steepest descent based molecular optimization of the model. A simple molecular dynamics feature is also implemented for performing basic dynamics study on the models. The "View Model" feature can be used to view the 3d structure of the generated models using the default PDB viewer installed in the system.





Related works and citations using EasyModeller

Phylogenetic and in silico structural analysis of the Parkinson disease-related kinase PINK1.
Cardona F, Sánchez-Mut JV, Dopazo H, Pérez-Tur J.

Hum Mutat. 2011 Apr;32(4):369-78. doi: 10.1002/humu.21444.

PMID:21412950

Structural and mechanistic determinants of a novel site for noncompetitive inhibition of GluN2D-containing NMDA receptors.

Hansen KB, Traynelis SF.

J Neurosci. 2011 Mar 9;31(10):3650-61.

PMID:21389220

Phosphorylation Alters the Interaction of the Arabidopsis Phosphotransfer Protein AHP1 with Its Sensor Kinase ETR1

Scharein B, Groth G

PLoS ONE , 2011,6(9): e24173.

doi:10.1371/journal.pone.0024173

The NMR structure of stomagen reveals the basis of stomatal density regulation by plant peptide hormones.

Ohki S, Takeuchi M, Mori M.

Nat Commun. 2011 Oct 25;2:512. doi: 10.1038/ncomms1520.