ISSN: 0975-8585

Research Journal of Pharmaceutical, Biological and Chemical Sciences

Development Of An Integrated Application For The Process Of Homology Modelling Using Net Framework.

Udayakumar M[a]*, Ragavan G[a], and Shrivatsan AD[b].

[a]School of Chemical and Biotechnology, SASTRA University India
[b]School of Computing,SASTRA University, India

ABSTRACT

One of the biggest goals in structural bioinformatics is the prediction of the three-dimensional structure of a protein from its one-dimensional protein sequence. The goal is to be able to determine the shape (known as a fold) that a given amino acid sequence will adopt. The problem is divided further based on whether the sequence will adopt a new fold or resemble an existing fold (template) in a protein structure database. Fold recognition is easy when the sequence in question has a high degree of sequence similarity to a sequence with known structure.If the two sequences share evolutionary ancestry, they are said to be homologous. For such sequence pairs we can build a structure for the query protein by choosing the structure of the known homologous sequence as a template. This process is knows as comparative modelling. Comparative Modelling is used when there exists a clear relationship between the sequence of a query protein (unknown structure) to that of a sequence of a known structure. MODELLER is most frequently used for homology or comparative protein structure modelling..We have developed a tool that helps the users in the process of homology modelling at its different levels namely data retrieval, sequence similarity search, sequence alignment, generation of the necessary input files, generation of models, validation of models, structural alignment and visualization of the generated models.There is a need to integrate the required tools and provide a rich GUI so that the overall process can be made easier with increased efficiency and less time consumption. By automating this process, errors that manually occur can be reduced to a great extent and the data transfer between the modules, the time taken for the completion of the process is very much reduced.The user interface was developed using Visual Basic .NET platform and SOAP web services was employed in the application. In this research article ,Biojava is used for structural superimposition of protein molecules, thus calculating the Root Mean Square Deviation (RMSD).

Keywords: VB.NET,Modeller, SOAP, WSDL, Biojava, JMOL,Homology Modelling

Corresponding author

INTRODUCTION

Homology modeling, also known as comparative modeling of protein refers to constructing an atomic-resolution model of the "*target*" protein from its amino acid sequence and an experimental three-dimensional structure of a related homologous protein (the "*template*"). Homology modeling relies on the identification of one or more known protein structures likely to resemble the structure of the query sequence, and on the production of an alignment that maps residues in the query sequence to residues in the template sequence. It has been shown that, that protein structures are more conserved than protein sequences amongst homologues, but sequences falling below a 20% sequence identity can have very different structure, and reach what is called in homology modelling, the twilight zone.[2]Evolutionarily related proteins have similar sequences and naturally occurring homologous proteins have similar protein structure. It has been shown that three-dimensional protein structure is evolutionarily more conserved than expected due to sequence conservation. [3]The sequence alignment and template structure are then used to produce a structural model of the target. Because protein structures are more conserved than DNA sequences, detectable levels of sequence similarity usually imply significant structural similarity. [4]Given a template and an alignment, the information contained therein must be used to generate a three-dimensional structural model of the target, represented as a set of Cartesian coordinates for each atom in the protein. Three major classes of model generation methods are Fragment assembly, Segment matching and Satisfaction of special restraints, followed by Loop Modelling, Model optimization, Model assessment and Structural comparison and Visualization.Homology modelling is presently the only accurate (as far available) and fast method for getting the protein 3D structure from its sequence. Other methods like *ab initio* modelling are resource intensive, computationally costly and are very difficult to implement. More generally, the inputs to the program are restraints on the spatial structure of the amino acid sequence(s) and ligands to be modelled. The output is a 3D structure that satisfies these restraints as well as possible. Restraints can in principle be derived from a number of different sources. These include related protein structures (comparative modelling), NMR experiments (NMR refinement), rules of secondary structure packing (combinatorial modelling), cross-linking experiments, fluorescence spectroscopy, image reconstruction in electron microscopy, site-directed mutagenesis, intuition, residue-residue and atom-atom potentials of mean force, *etc*. The restraints can operate on distances, angles, dihedral angles, pairs of dihedral angles and some other spatial features defined by atoms or pseudo atoms. A 3D model is obtained by optimization of a molecular probability density function (pdf). The molecular pdf for comparative modelling is optimized with the variable target function procedure in Cartesian space that employs methods of conjugate gradients and molecular dynamics with simulated annealing. MODELLER is a computer program that models three-dimensional structures of proteins and their assemblies by satisfaction of spatial restraints. MODELLER is most frequently used for homology or comparative protein structure modelling. The user provides an alignment of a sequence to be modelled with known related structures and MODELLER will automatically calculate a model with all non-hydrogen atoms. Many types of software are available for homology modelling of which the most famous and most common tool for homology modelling is MODELLER. Apart from this there are also commercially available softwares like Insight II, Discovery studio, Hyperchem, etc. But MODELLER stands apart from this because it is freely available. The GUI for Moldeller was developed but not from the sequence . Our tool will begins with the BLAST module and finally ends with the structure visualization.

MATERIALS AND METHODS

The tool was deployed using online web services provided by the European Bioinformatics Institute (EBI). These services are invoked using the Simple Object Access Protocol. A request is posted to the web server with the parameters given as input. The server processes the request posted by the client and returns the required results or returns an error or exception based on the parameters specified. After it is processed, the corresponding results are sent back to the client in the form of bytes using XML, and then the bytes are converted to string and written to the client's session directory. Once the results are completely written, they are processed to be displayed in a proper format that can help the user understand the output better.Also the data transfer among the modules can be automated.With the development of a tool, that can use the web services at the back-end, the burden on the user can be very much reduced, thus saving time and effort involved.The tool was developed using Microsoft Visual Studio, an Integrated Development Environment from Microsoft Corporation. Visual Basic .NET (VB.NET) is an object-oriented computer programming language that can be viewed as an evolution of Microsoft's Visual Basic(VB) which is generally implemented on the Microsoft .NET Framework. VB is an event-driven programming language and integrated development environment

(IDE). VB is also considered a relatively easy to learn and use programming language, because of its graphical development features and BASIC heritage.Jmol is an open-source Java viewer for chemical structures in 3D. Jmol returns a 3D representation of a molecule that may be used as a teaching tool, or for research e.g. in chemistry and biochemistry. It is free and open source software, written in Java and so it runs on Windows, Mac OS X, Linux and Unix systems. There is a standalone application and a development tool kit that can be integrated into other Java applications.Web services are typically application programming interfaces (API) or web APIs that can be accessed over a network, such as the Internet, and executed on a remote system hosting the requested services.The three most common styles of use of web services are RPC, SOA and REST.BioJava is an open-source project dedicated to providing a Java framework for processing biological data. BioJava provides many powerful modules that can help us to analyse and manipulate biological data. Here, in this project, Biojava is used for structural superimposition of protein molecules, thus calculating the Root Mean Square Deviation (RMSD).The methodology and the workflow is shown in Figure 1 .

Session creation

From Fig.1, the developed software begins its execution by choosing a directory for the current session. If the user does not want to use an existing directory, the tool creates a directory with the date and time as the name. And once a directory is chosen or created, a directory with name “source” is created and all the necessary script files to be used by the tool are written into the directory.

Data retrieval

In this module, the web service Dbfecth was used. Here, the GUI (Fig.2) is designed in such a way that the user simply has to select the database, keyword and the required format. The web service takes in the parameters specified by the user as input and posts a request to the web server and the corresponding data required is displayed to the user and the user is given the choice of saving the output obtained. If any of the parameters are wrong, the web service returns an error message to the user, so that the user can change the mistakes made in specifying the input.

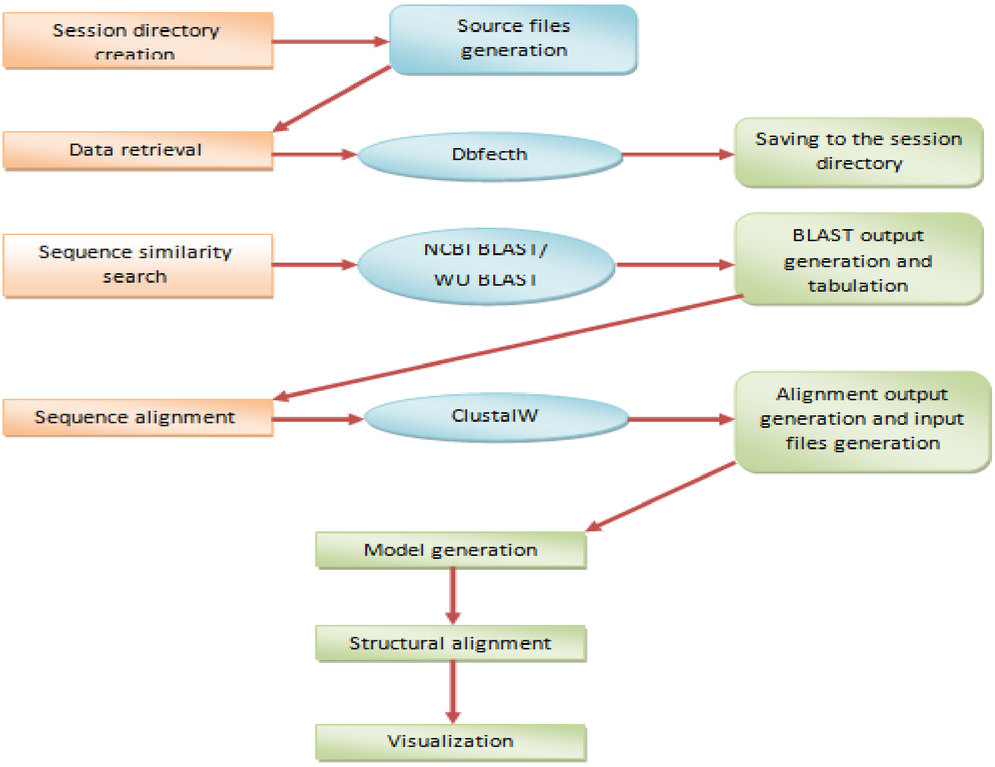

Session directory creation
Source files generation
Data retrieval
Dbfecth
Saving to the session directory
Sequence similarity search
NCBI BLAST/ WU BLAST
BLAST output generation and tabulation
Sequence alignment
ClustalW
Alignment output generation and input files generation
Model generation
Structural alignment
Visualization

Fig.1: Scheme of the tool .The arrows (red colour) show the possibility of data transfer between the different elements of the tool.

Sequence similarity search

In this module, the web services WUBLAST and NCBI BLAST are used. Here, the user simply has to input the sequence which is to be searched against the databases. The user is provided the option of choosing the desired search algorithm and the corresponding options are shown to the user for selecting the values of the required parameters which is shown in Fig.3. Once the values are specified for the parameters, a request is made to the web service with the chosen BLAST program. This web service, verifies the values and two outputs are sent back to the client: a text file and an XML file in the form of bytes. They are converted to strings, written to the session directory and the output in the XML file is tabulated with the help of the DataGridView component available as shown in Fig.4. And the BLAST results do not include any information regarding the resolution of the PDB id. But this software is made to retrieve the resolution for each hit.Once the results are tabulated, the check boxes are provided for each hit. The user simply has to mark the check box for the desired template and the corresponding PDB files are retrieved for the selected templates whih is shown in Fig. 5. Also the necessary input files are generated for the next module.

Sequence alignment

The sequences of the selected templates are programmatically retrieved and placed as inputs for sequence alignment which uses the CLUSTALW web service. Again, this module provides all the parameters available in the online server.Sequene alignment module is shown in Fig. 6. Once the values are selected, they are posted in the form of a request to the web service and on processing of the request, the web service returns 3 outputs: a text file, an alignment file(.aln) and a dendogram file(.dnd). The .aln file is then converted to the .ali format using the starting and ending residues of the selected templates.

Model generation

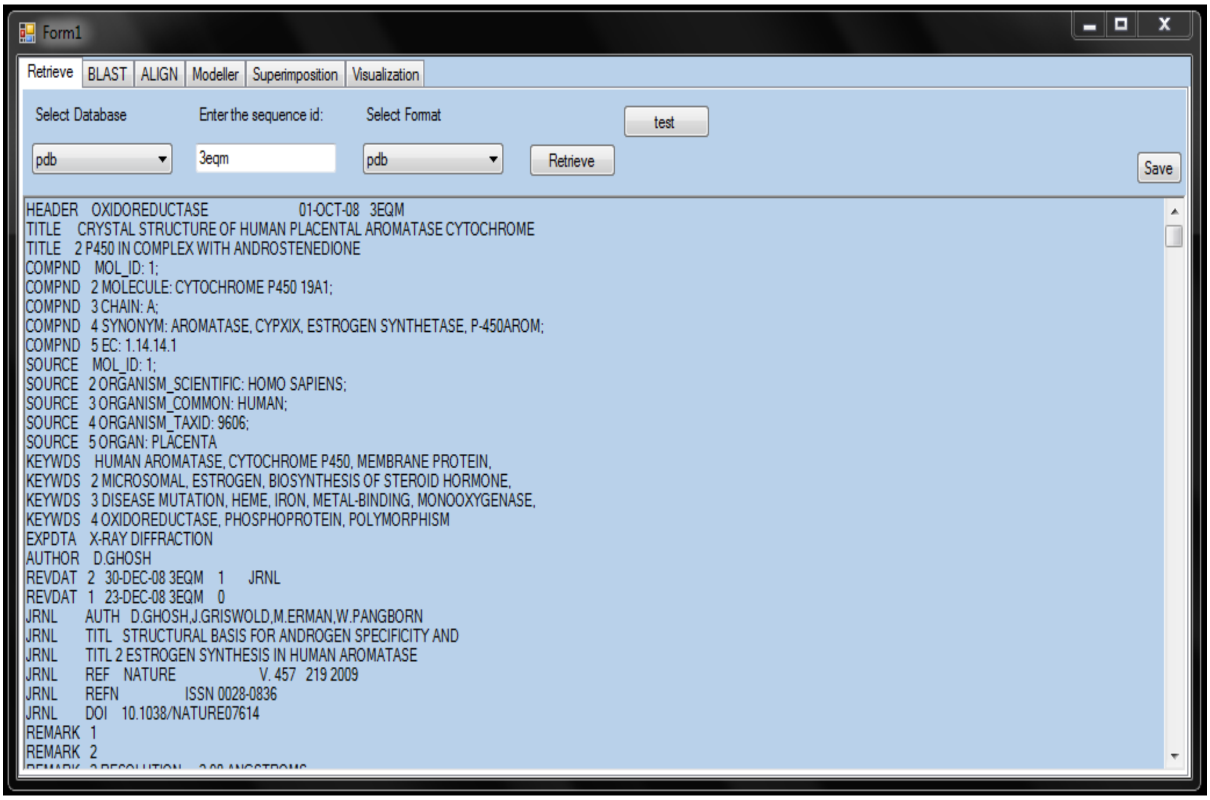
This is the module that actually generates the models for the target. Here, the user is provided the option of changing the number of models to be generated. Based on this value, the program generates the Python script necessary for the software to run.The Python scripts are nothing but text files, so they can be generated programmatically. After the generation of all the input files necessary, the commands for the execution of the Python scripts are run in the Windows Command Prompt in the background.Once the execution of the commands is over, the results are tabulated, thus showing the user the quality of models based on the assessment using DOPE Scores and Molecular pdf scores as shown in Fig.7.

Structural Alignment

The models, after being generated, can be subjected to superimposition with the templates. This is facilitated using the Structure package present in Biojava. Here, the user is given the option of selecting any two molecules and performing structural alignment.Once the molecules are selected, the tool generates a JAVA file containing the names of the molecules to be superimposed, compiles and executes this JAVA file and the results are written to a text file, which is then read using file handling techniques and the reaults are tabulated. Structural Alignment module is shown in Fig.9

Visualization

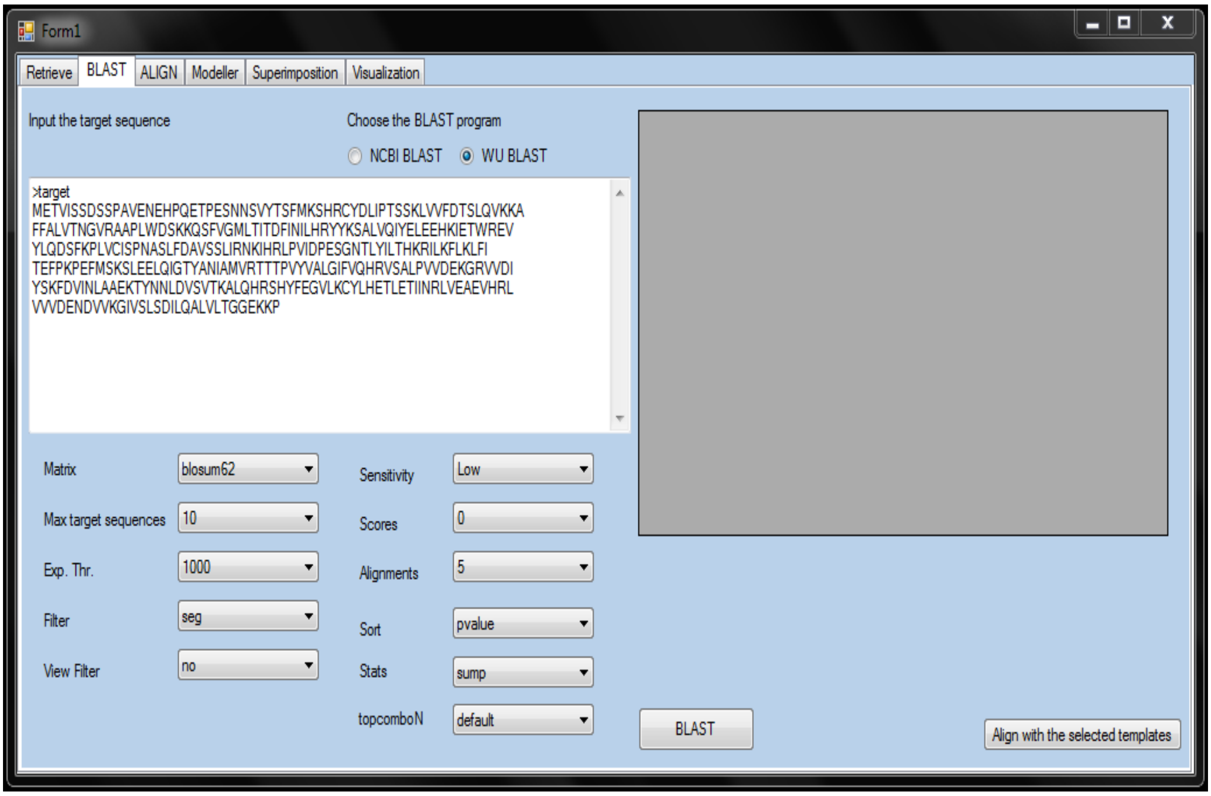
This module as shown in Fig.10 provides the user the facility of viewing the models generated. The user simply has to select the molecule to be visualized and the tool generates a Jmol script specifying the molecule to be visualized, embedded into a HTML file, which upon creation is loaded into the web browser present in the GUI.The tool uses Jmol to visualize the molecule, so all the features available in Jmol can be used in an efficient manner.



Select Database: pdb
Enter the sequence id: 3eqm
Select Format: pdb

```
HEADER    OXIDOREDUCTASE                          01-OCT-08   3EQM
TITLE     CRYSTAL STRUCTURE OF HUMAN PLACENTAL AROMATASE CYTOCHROME
TITLE    2 P450 IN COMPLEX WITH ANDROSTENEDIONE
COMPND    MOL_ID: 1;
COMPND   2 MOLECULE: CYTOCHROME P450 19A1;
COMPND   3 CHAIN: A;
COMPND   4 SYNONYM: AROMATASE, CYPXIX, ESTROGEN SYNTHETASE, P-450AROM;
COMPND   5 EC: 1.14.14.1
SOURCE    MOL_ID: 1;
SOURCE   2 ORGANISM_SCIENTIFIC: HOMO SAPIENS;
SOURCE   3 ORGANISM_COMMON: HUMAN;
SOURCE   4 ORGANISM_TAXID: 9606;
SOURCE   5 ORGAN: PLACENTA
KEYWDS    HUMAN AROMATASE, CYTOCHROME P450, MEMBRANE PROTEIN,
KEYWDS   2 MICROSOMAL, ESTROGEN, BIOSYNTHESIS OF STEROID HORMONE,
KEYWDS   3 DISEASE MUTATION, HEME, IRON, METAL-BINDING, MONOOXYGENASE,
KEYWDS   4 OXIDOREDUCTASE, PHOSPHOPROTEIN, POLYMORPHISM
EXPDTA    X-RAY DIFFRACTION
AUTHOR    D.GHOSH
REVDAT   2   30-DEC-08 3EQM    1       JRNL
REVDAT   1   23-DEC-08 3EQM    0
JRNL        AUTH   D.GHOSH,J.GRISWOLD,M.ERMAN,W.PANGBORN
JRNL        TITL   STRUCTURAL BASIS FOR ANDROGEN SPECIFICITY AND
JRNL        TITL 2 ESTROGEN SYNTHESIS IN HUMAN AROMATASE
JRNL        REF    NATURE                         V. 457   219 2009
JRNL        REFN                   ISSN 0028-0836
JRNL        DOI    10.1038/NATURE07614
REMARK   1
REMARK   2
```

Fig. 2 Data retrieval using the Dbfetch module web service.



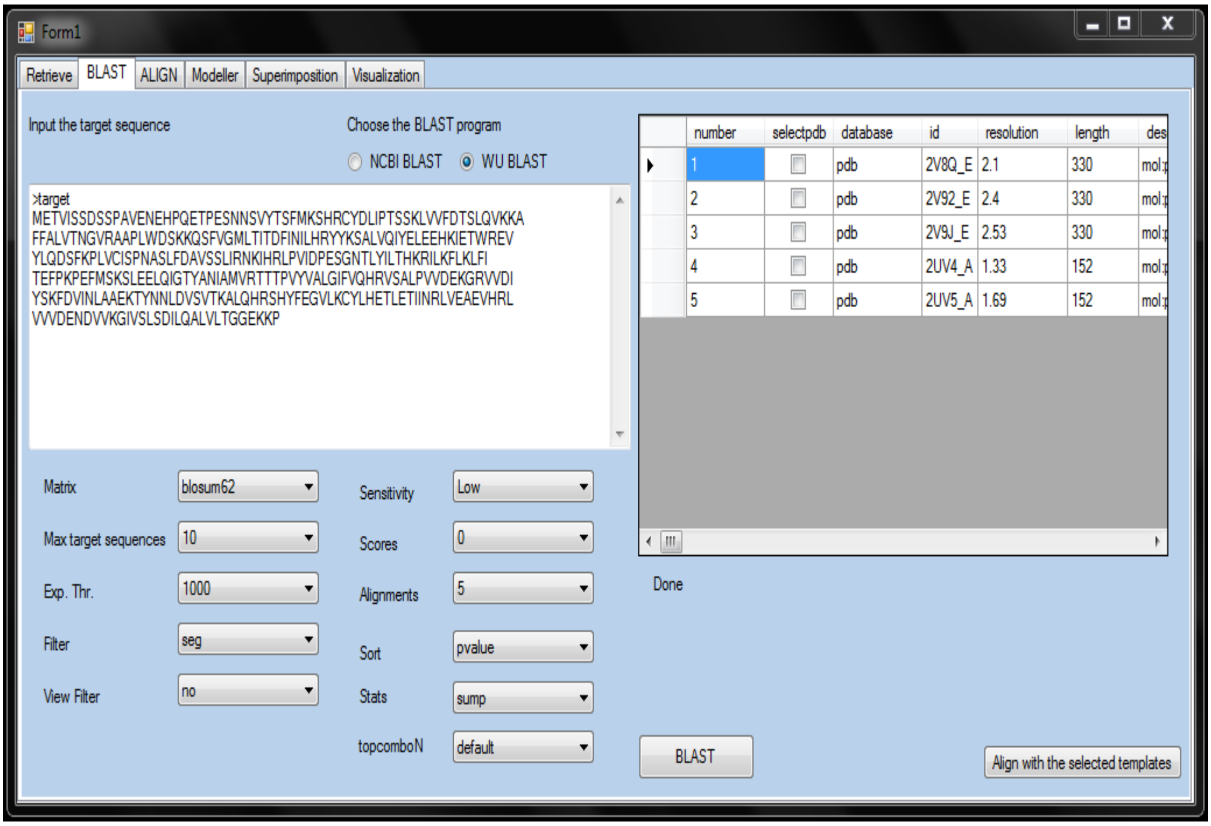
Input the target sequence

```
>target
METVISSDSSPAVENEHPQETPESNNSVYTSFMKSHRCYDLIPTSSKLVVFDTSLQVKKA
FFALVTNGVRAAPLWDSKKQSFVGMLTITDFINILHRYYKSALVQIYELEEHKIETWREV
YLQDSFKPLVCISPNASLFDAVSSLIRNKIHRLPVIDPESGNTLYILTHKRILKFLKLFI
TEFPKPEFMSKSLEELQIGTYANIAMVRTTTPVYVALGIFVQHRVSALPVVDEKGRVVDI
YSKFDVINLAAEKTYNNLDVSVTKALQHRSHYFEGVLKCYLHETLETIINRLVEAEVHRL
VVVDENDVVKGIVSLSDILQALVLTGGEKKP
```

Choose the BLAST program
○ NCBI BLAST ● WU BLAST

Matrix: blosum62
Max target sequences: 10
Exp. Thr.: 1000
Filter: seg
View Filter: no
Sensitivity: Low
Scores: 0
Alignments: 5
Sort: pvalue
Stats: sump
topcomboN: default

Fig. 3 BLAST front end in VB.NET application.



Input the target sequence

Choose the BLAST program

○ NCBI BLAST ● WU BLAST

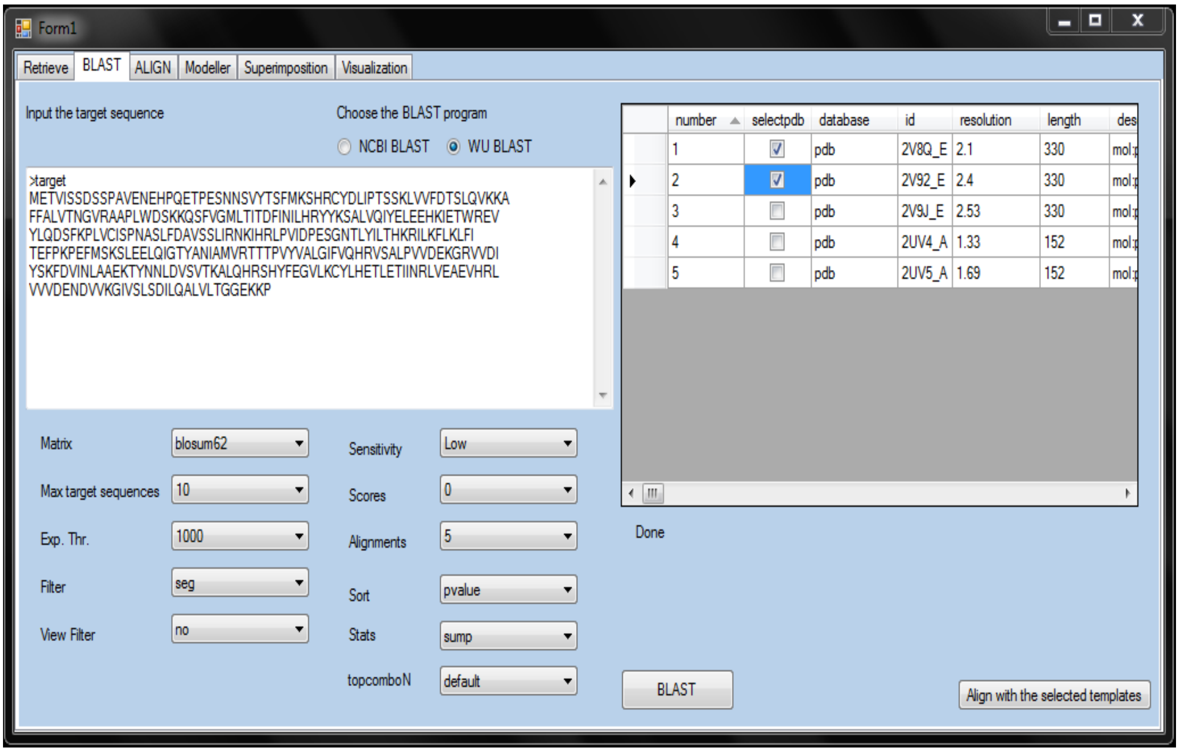
```
>target
METVISSDSSPAVENEHPQETPESNNSVYTSFMKSHRCYDLIPTSSKLVVFDTSLQVKKA
FFALVTNGVRAAPLWDSKKQSFVGMLTITDFINILHRYYKSALVQIYELEEHKIETWREV
YLQDSFKPLVCISPNASLFDAVSSLIRNKIHRLPVIDPESGNTLYILTHKRILKFLKLFI
TEFPKPEFMSKSLEELQIGTYANIAMVRTTTPVYVALGIFVQHRVSALPVVDEKGRVVDI
YSKFDVINLAAEKTYNNLDVSVTKALQHRSHYFEGVLKCYLHETLETIINRLVEAEVHRL
VVVDENDVVKGIVSLSDILQALVLTGGEKKP
```

Matrix	blosum62
Max target sequences	10
Exp. Thr.	1000
Filter	seg
View Filter	no
Sensitivity	Low
Scores	0
Alignments	5
Sort	pvalue
Stats	sump
topcomboN	default

number	selectpdb	database	id	resolution	length	des
1	□	pdb	2V8Q_E	2.1	330	mol:
2	□	pdb	2V92_E	2.4	330	mol:
3	□	pdb	2V9J_E	2.53	330	mol:
4	□	pdb	2UV4_A	1.33	152	mol:
5	□	pdb	2UV5_A	1.69	152	mol:

Done

Fig. 4 Tabulated of BLAST results after sequence similarity search



Input the target sequence

Choose the BLAST program

○ NCBI BLAST ● WU BLAST

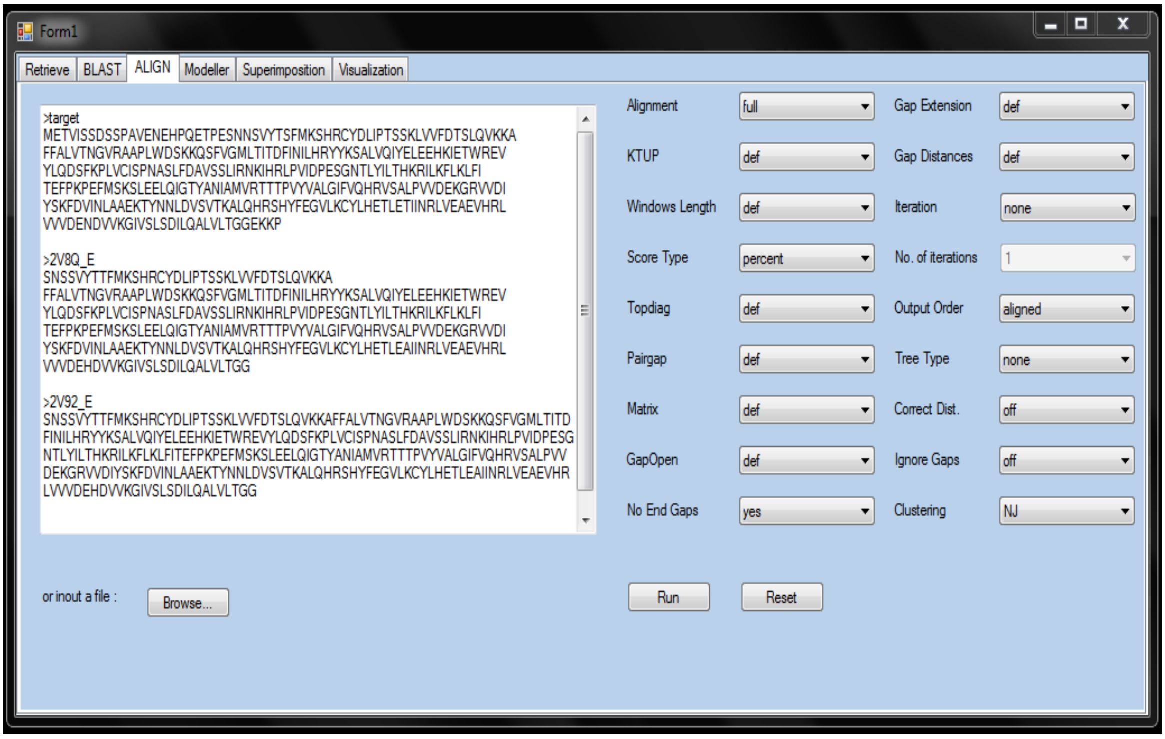
```
>target
METVISSDSSPAVENEHPQETPESNNSVYTSFMKSHRCYDLIPTSSKLVVFDTSLQVKKA
FFALVTNGVRAAPLWDSKKQSFVGMLTITDFINILHRYYKSALVQIYELEEHKIETWREV
YLQDSFKPLVCISPNASLFDAVSSLIRNKIHRLPVIDPESGNTLYILTHKRILKFLKLFI
TEFPKPEFMSKSLEELQIGTYANIAMVRTTTPVYVALGIFVQHRVSALPVVDEKGRVVDI
YSKFDVINLAAEKTYNNLDVSVTKALQHRSHYFEGVLKCYLHETLETIINRLVEAEVHRL
VVVDENDVVKGIVSLSDILQALVLTGGEKKP
```

Matrix	blosum62
Max target sequences	10
Exp. Thr.	1000
Filter	seg
View Filter	no
Sensitivity	Low
Scores	0
Alignments	5
Sort	pvalue
Stats	sump
topcomboN	default

number	selectpdb	database	id	resolution	length	des
1	☑	pdb	2V8Q_E	2.1	330	mol:
2	☑	pdb	2V92_E	2.4	330	mol:
3	□	pdb	2V9J_E	2.53	330	mol:
4	□	pdb	2UV4_A	1.33	152	mol:
5	□	pdb	2UV5_A	1.69	152	mol:

Done

Fig. 5 Selection of Template structure from PDB database

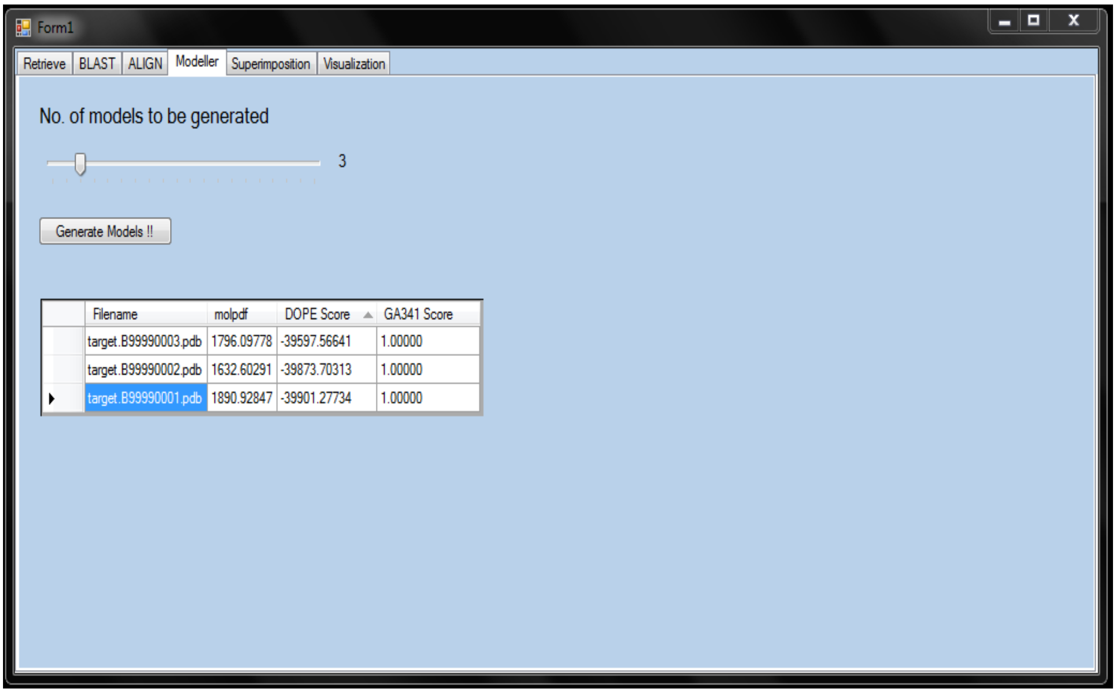


```
>target
METVISSDSSPAVENEHPQETPESNNSVYTSFMKSHRCYDLIPTSSKLVVFDTSLQVKKA
FFALVTNGVRAAPLWDSKKQSFVGMLTITDFINILHRYYKSALVQIYELEEHKIETWREV
YLQDSFKPLVCISPNASLFDAVSSLIRNKIHRLPVIDPESGNTLYILTHKRILKFLKLFI
TEFPKPEFMSKSLEELQIGTYANIAMVRTTTPVYVALGIFVQHRVSALPVVDEKGRVVDI
YSKFDVINLAAEKTYNNLDVSVTKALQHRSHYFEGVLKCYLHETLETIINRLVEAEVHRL
VVVDENDVVKGIVSLSDILQALVLTGGEKKP

>2V8Q_E
SNSSVYTTFMKSHRCYDLIPTSSKLVVFDTSLQVKKA
FFALVTNGVRAAPLWDSKKQSFVGMLTITDFINILHRYYKSALVQIYELEEHKIETWREV
YLQDSFKPLVCISPNASLFDAVSSLIRNKIHRLPVIDPESGNTLYILTHKRILKFLKLFI
TEFPKPEFMSKSLEELQIGTYANIAMVRTTTPVYVALGIFVQHRVSALPVVDEKGRVVDI
YSKFDVINLAAEKTYNNLDVSVTKALQHRSHYFEGVLKCYLHETLEAIINRLVEAEVHRL
VVVDEHDVVKGIVSLSDILQALVLTGG

>2V92_E
SNSSVYTTFMKSHRCYDLIPTSSKLVVFDTSLQVKKAFFALVTNGVRAAPLWDSKKQSFVGMLTITD
FINILHRYYKSALVQIYELEEHKIETWREVYLQDSFKPLVCISPNASLFDAVSSLIRNKIHRLPVIDPESG
NTLYILTHKRILKFLKLFITEFPKPEFMSKSLEELQIGTYANIAMVRTTTPVYVALGIFVQHRVSALPVV
DEKGRVVDIYSKFDVINLAAEKTYNNLDVSVTKALQHRSHYFEGVLKCYLHETLEAIINRLVEAEVHR
LVVVDEHDVVKGIVSLSDILQALVLTGG
```

or inout a file :

Alignment	full	Gap Extension	def
KTUP	def	Gap Distances	def
Windows Length	def	Iteration	none
Score Type	percent	No. of iterations	1
Topdiag	def	Output Order	aligned
Pairgap	def	Tree Type	none
Matrix	def	Correct Dist.	off
GapOpen	def	Ignore Gaps	off
No End Gaps	yes	Clustering	NJ

Fig. 6 Multiple Sequence Alignment between target and template structure

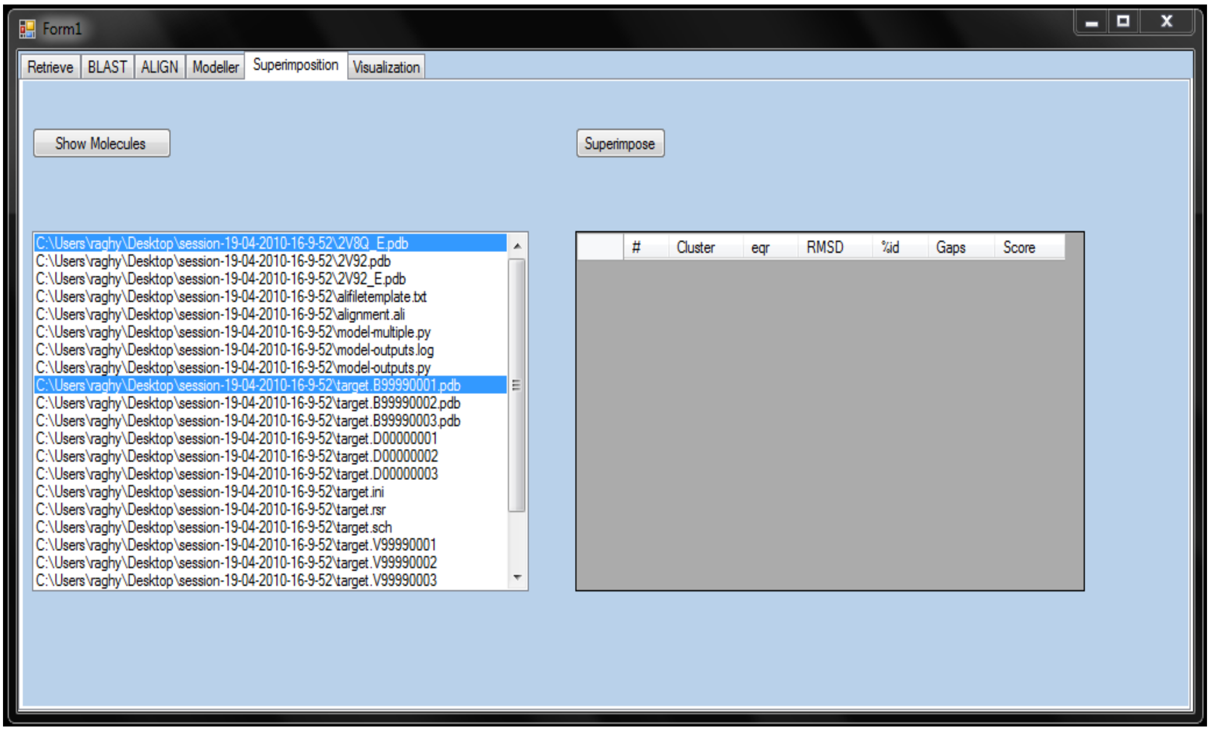


No. of models to be generated

3

Filename	molpdf	DOPE Score	GA341 Score
target.B99990003.pdb	1796.09778	-39597.56641	1.00000
target.B99990002.pdb	1632.60291	-39873.70313	1.00000
target.B99990001.pdb	1890.92847	-39901.27734	1.00000

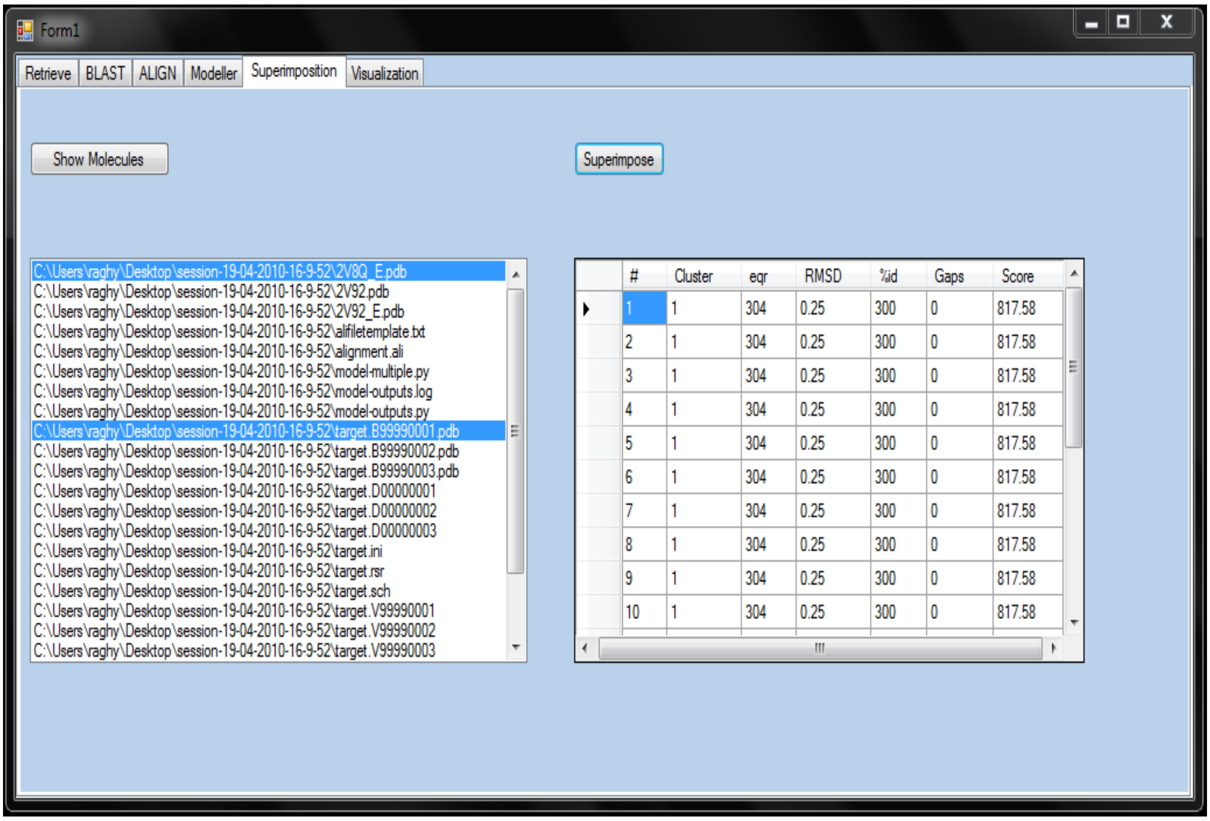
Fig. 7 Models generated and results are tabulated based on DOPE score



C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\2V8Q_E.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\2V92.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\2V92_E.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\alifiletemplate.txt
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\alignment.ali
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\model-multiple.py
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\model-outputs.log
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\model-outputs.py
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.B99990001.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.B99990002.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.B99990003.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.D00000001
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.D00000002
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.D00000003
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.ini
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.rsr
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.sch
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.V99990001
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.V99990002
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.V99990003

#	Cluster	eqr	RMSD	%id	Gaps	Score

Fig. 8 Molecules selected for superimposition



C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\2V8Q_E.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\2V92.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\2V92_E.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\alifiletemplate.txt
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\alignment.ali
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\model-multiple.py
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\model-outputs.log
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\model-outputs.py
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.B99990001.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.B99990002.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.B99990003.pdb
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.D00000001
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.D00000002
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.D00000003
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.ini
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.rsr
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.sch
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.V99990001
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.V99990002
C:\Users\raghy\Desktop\session-19-04-2010-16-9-52\target.V99990003

#	Cluster	eqr	RMSD	%id	Gaps	Score
1	1	304	0.25	300	0	817.58
2	1	304	0.25	300	0	817.58
3	1	304	0.25	300	0	817.58
4	1	304	0.25	300	0	817.58
5	1	304	0.25	300	0	817.58
6	1	304	0.25	300	0	817.58
7	1	304	0.25	300	0	817.58
8	1	304	0.25	300	0	817.58
9	1	304	0.25	300	0	817.58
10	1	304	0.25	300	0	817.58

Fig. 9 Superimposition results tabulated

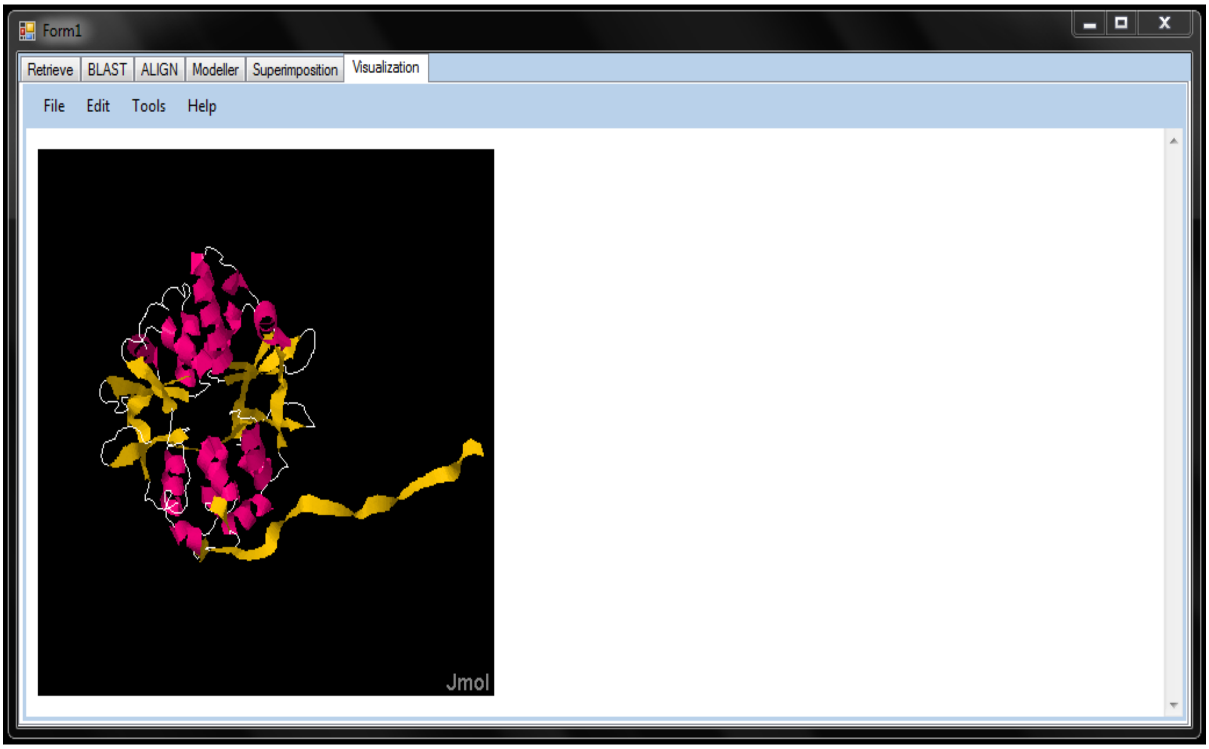


Fig. 10 Visualization of generated model using JMOL plugin

Sample Scripts:

The following code is used when the tool is started, to provide the user the option of creating a directory for a particular session. If no directory is selected, the tool creates a directory with the date and time of its creation. And in that directory, all the files required files for the tool are written.

```
Private Sub Form1_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
MsgBox("Select a directory for the session")
Dim opfol As New FolderBrowserDialog

        If opfol.ShowDialog = Windows.Forms.DialogResult.OK Then
        p = Path.GetFullPath(opfol.SelectedPath)
        Else
My.Computer.FileSystem.CreateDirectory(My.Computer.FileSystem.SpecialDirectories.Desktop & "\session-" & My.Computer.Clock.LocalTime.Date & "-" & My.Computer.Clock.LocalTime.Hour & "-" & My.Computer.Clock.LocalTime.Minute)
p=My.Computer.FileSystem.SpecialDirectories.Desktop&"\session-"& My.Computer.Clock.LocalTime
     End If
If My.Computer.FileSystem.DirectoryExists(p & "\source") Then
My.Computer.FileSystem.DeleteDirectory(p&"\source\", FileIO.DeleteDirectoryOption.DeleteAllContents)
End If
My.Computer.FileSystem.CreateDirectory(p & "\source\jmol\")
System.IO.File.WriteAllBytes(p & "\source\jmol\Jmol.jar", My.Resources.Jmol)
System.IO.File.WriteAllBytes(p&"\source\jmol\JmolApplet.jar",My.Resources.JmolApplet)
System.IO.File.WriteAllBytes(p&"\source\jmol\JmolApplet0.jar",My.Resources.JmolApplet0)
System.IO.File.WriteAllText(p & "\source\jmol\Jmoljs.js", My.Resources.Jmoljs)
If My.Computer.FileSystem.DirectoryExists(p & "\modeller") Then
My.Computer.FileSystem.DeleteDirectory(p&"\modeller\",FileIO.DeleteDirectoryOption.DeleteAllContents)
End If
My.Computer.FileSystem.CreateDirectory(p & "\source\modeller\")
System.IO.File.WriteAllBytes(p&"\source\modeller\model-multiple.py",My.Resources.model_multiple)
System.IO.File.WriteAllBytes(p& "\model-multiple.py", My.Resources.model_multiple)
System.IO.File.WriteAllBytes(p&"\source\modeller\model-outputs.py",My.Resources.model_outputs)
System.IO.File.WriteAllBytes(p & "\model-outputs.py", My.Resources.model_outputs)
System.IO.File.WriteAllText(p&"\source\modeller\modenv.bat",My.Resources.modenv)
End Sub
```

Script is used for data retrieval module from different databases.

```
Function retrieve()
Dim id As String = Nothing
id = ret_seqid.Text
Dim Dbfetch As wsdbfetch.WSDBFetchServerLegacyService = New wsdbfetch.WSDBFetchServerLegacyService()
Dim result, str1, str2 As String = Nothing
str1 = Me.ret_selectdb.SelectedItem.ToString.ToLower & ":" & ret_seqid.Text.ToLower
str2 = Me.ret_seqformat.SelectedItem.ToString.ToLower
Try
result = Dbfetch.fetchData(str1, str2, "")
ret_result.Text = result
Catch ex As Exception
MsgBox(ex.ToString)
MsgBox("Try Again !!", MsgBoxStyle.Exclamation)
ret_selectdb.SelectedItem = Nothing
ret_seqid.Text = Nothing
ret_seqformat.SelectedItem = Nothing
End Try
Return Nothing
End Function
```

This script is used in the execution of the BLAST program using the web service provided by EBI.

```
Private Sub blast_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles blast.Click
Dim wswublast As wswublast.WSWUBlastService=Newwswublast.WSWUBlastService
Dim parameters As New wswublast.inputParams()
parameters.matrix = wublastmatrix.SelectedItem.ToString
parameters.email = "vrraghy@gmail.com"
parameters.program = "blastp"
parameters.database = "pdb"
parameters.exp = wublastexpthr.SelectedItem.ToString
parameters.filter = wublastfilter.SelectedItem.ToString
parameters.scores = wublastscores.SelectedItem.ToString
parameters.numal = wublastalignments.SelectedItem.ToString
parameters.sort = wublastsort.SelectedItem.ToString
parameters.stats = wubl aststats.SelectedItem.ToString
parameters.topcombon = wublasttopcombon.SelectedItem.ToString
Dim seq() As wswublast.data = New wswublast.data(0) {}
seq(0).type = "sequence"
seq(0).content = blastinput.Text
Try
Dim jobId As String = wswublast.runWUBlast(parameters, seq)
If jobId <> Nothing Then
statuslabel.Text = "JobId : " & jobId
Dim status As String = wswublast.checkStatus(jobId)
statuslabel.Text = "Status : " & status
If status = "RUNNING" Then
Dim t As New Timer
Do
t.Start()
t.Interval = 500
```

```
status = wswublast.checkStatus(jobId)
Loop
End If
If status = "DONE" Then
statuslabel.Text = "Status : " & status
Dim resfile() As wswublast.WSFile = New wswublast.WSFile(1) {}
resfile = wswublast.getResults(jobId)
Dim x As Byte() = Nothing
Dim b64str As String = Nothing
Dim out(resfile.Length) As String
Dim filename As String = Nothing
Dim xmlfilename As String = Nothing
For i As Integer = 0 To resfile.Length - 1
x = wswublast.poll(jobId, resfile(i).type)
b64str = System.Text.Encoding.UTF8.GetString(x)
out(i) = Nothing
out(i) = b64str
filename = p & "\" & resfile(i).type & "." & resfile(i).ext
My.Computer.FileSystem.WriteAllText(filename, out(i), False)
If resfile(i).ext = "xml" Then
xmlfilename = filename
End If
Next
tabulate(xmlfilename, blastres)
End If
End If
Catch ex As Exception
MsgBox(ex.ToString)
MsgBox("Try Again !!", MsgBoxStyle.Exclamation)
End Try
End Sub
```

The following script is used to tabulate the BLAST results using the XML file generated by the above script.

```
Function tabulate(ByVal xmlfilename As String, ByVal bv As DataGridView)
Dim xfile As Xml.XmlDataDocument = New Xml.XmlDataDocument()
xfile.DataSet.ReadXml(xmlfilename)
hits = xfile.GetElementsByTagName("hits").ItemOf(0)
hit = hits.ChildNodes
For i As Integer = 0 To hit.Count - 1
For j As Integer = 0 To hit(i).Attributes.Count - 1
If bv.ColumnCount < hit(i).Attributes.Count Then
bv.Columns.Add(hit(i).Attributes(j).Name, hit(i).Attributes(j).Name)
End If
Next
alignments = hit(i).FirstChild
alignment = alignments.FirstChild
For k As Integer = 0 To alignment.ChildNodes.Count - 1
If bv.ColumnCount < (hit(i).Attributes.Count + alignment.ChildNodes.Count) Then
bv.Columns.Add(alignment.ChildNodes.ItemOf(k).Name, alignment.ChildNodes.ItemOf(k).Value)
If alignment.ChildNodes.ItemOf(k).Name = "matchSeq" Then
bv.Columns.Add(alignment.ChildNodes.ItemOf(k).Attributes("start").Name,
alignment.ChildNodes.ItemOf(k).Attributes("start").Name)
bv.Columns.Add(alignment.ChildNodes.ItemOf(k).Attributes("end").Name,
alignment.ChildNodes.ItemOf(k).Attributes("end").Name)
End If
End If
Next
bv.Rows.Add()
```

```
Next
For i As Integer = 0 To hit.Count - 1
For j As Integer = 0 To hit(i).Attributes.Count - 1
bv.Item(j, i).Value = hit(i).Attributes(j).Value
Next
alignments = hit(i).FirstChild
alignment = alignments.FirstChild
For k As Integer = 0 To alignment.ChildNodes.Count - 1
bv.Item(alignment.ChildNodes.ItemOf(k).Name,i).Value=alignment.ChildNodes.ItemOf(k).InnerText
If alignment.ChildNodes.ItemOf(k).Name = "matchSeq" Then
bv.Item(alignment.ChildNodes.ItemOf(k).Attributes("start").Name,i).Value=alignment.ChildNodes.ItemOf(k).Attributes("start").Value
bv.Item(alignment.ChildNodes.ItemOf(k).Attributes("end").Name,i).Value=alignment.ChildNodes.ItemOf(k).Attributes("end").Value
End If
Next
Next
bv.Columns("score").HeaderCell.SortGlyphDirection=Windows.Forms.SortOrder.Descending
Return True
End Function
```

The following script generate the alignment file(.ali) based on the selected templates.

```
Function align()
Dim wsclustalws As wsclustalw2.WSClustalW2Service = New wsclustalw2.WSClustalW2Service
Dim jobid As String = Nothing
Dim parameters As New wsclustalw2.inputParams
parameters.email = "vrraghy@gmail.com"
Dim seq() As wsclustalw2.data = New wsclustalw2.data(0) {}
seq(0) = New wsclustalw2.data()
parameters.output = "pir"
parameters.outorder = outorder
seq(0).type = "sequence"
seq(0).content = aligninput.Text
Try
jobid = wsclustalws.runClustalW2(parameters, seq)
If jobid <> Nothing Then
status = wsclustalws.checkStatus(jobid)
If status = "DONE" Then
Dim resfile() As wsclustalw2.WSFile = New wsclustalw2.WSFile(1) {}
resfile = wsclustalws.getResults(jobid)
For i As Integer = 0 To resfile.Length - 1
x = wsclustalws.poll(jobid, resfile(i).type)
b64str = System.Text.Encoding.UTF8.GetString(x)
out(i) = b64str
filename = p & "\" & resfile(i).type & "." & resfile(i).ext
My.Computer.FileSystem.WriteAllText(filename, out(i), False)
Next
While sr.EndOfStream = False
If sr.ReadLine().StartsWith(">P1") Then
count = count + 1
End If
End While
alnfiledata = My.Computer.FileSystem.ReadAllText(p & "\" & "toolaln.aln")
individuals = Nothing
individuals = alnfiledata.Split(">".ToCharArray)
alifilelines = Strings.Split(alifiletemplate, "@seperator@")
```

```
Dim towrite As String = Nothing
For i As Integer = 1 To count
individuals(i) = individuals(i).Replace("P1;", ">P1;")
lines = Strings.Split(individuals(i), vbNewLine)
lines = individuals(i).Split(vbNewLine.ToCharArray)
For j As Integer = 0 To alifilelines.Count - 1
If alifilelines(j).StartsWith(lines(0)) Then
lines(0) = alifilelines(j)
End If
Next
individuals(i) = Strings.Join(lines, vbNewLine)
towrite &= individuals(i) & vbNewLine
Next
towrite = "C; A sample alignment in the PIR format; used in tutorial" & vbNewLine & vbNewLine & towrite
My.Computer.FileSystem.WriteAllText(p & "\alignment.ali", towrite, False)
towrite = My.Computer.FileSystem.ReadAllText(p & "\model-outputs.py")
MsgBox(templates)
templates = templates.Substring(1)
MsgBox(templates)
towrite = towrite.Replace("templatecodes", templates)
fullalifile = towrite
TabControl1.SelectTab("modeller")
End If
End If
Catch ex As Exception
MsgBox(ex.ToString)
MsgBox("Try Again !!", MsgBoxStyle.Exclamation)
End Try
Return Nothing
End Function
```

Script used to generate the models by running the modeller commands in the background.

```
Private Sub genmodels_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles genmodels.Click
fullalifile = fullalifile.Replace("nummodels", n.ToString)
My.Computer.FileSystem.WriteAllText(p & "\model-outputs.py", fullalifile, False)
com1 = "cd " & p & "\source\modeller\ &&  modenv.bat"
com2 = "cd " & p & "\ && mod9v7 model-outputs.py"
fullcom = com1 & " && " & com2
Shell("cmd.exe /c " & dq & fullcom & dq, AppWinStyle.NormalFocus, True)
If System.IO.File.Exists(p & "\target.B99990001") Then
Dim content As String = My.Computer.FileSystem.ReadAllText(p & "\model-outputs.log")
lines=content.Split(vbNewLine.ToCharArray, StringSplitOptions.RemoveEmptyEntries)
For i = 0 To lines.Length - 1
If lines(i).StartsWith("Filename") Then
i = i + 2
Do While lines(i).StartsWith("target")
content &= lines(i) & vbNewLine
i = i + 1
Loop
Exit For
End If
Next
lines=content.Split(vbNewLine.ToCharArray, StringSplitOptions.RemoveEmptyEntries)
modeloutputstable.Columns.Add("filename", "Filename")
modeloutputstable.Columns.Add("molpdf", "molpdf")
modeloutputstable.Columns.Add("dope", "DOPE Score")
modeloutputstable.Columns.Add("ga", "GA341 Score")
For i = 0 To lines.Length - 1
modeloutputstable.Rows.Add()
linesplit = lines(i).Split(" ".ToCharArray, StringSplitOptions.RemoveEmptyEntries)
modeloutputstable.Item("filename", i).Value = linesplit(0)
modeloutputstable.Item("molpdf", i).Value = linesplit(1)
modeloutputstable.Item("dope", i).Value = linesplit(2)
modeloutputstable.Item("ga", i).Value = linesplit(3)
Next
End If
End Sub
```

CONCLUSION

We have developed a tool that helps the users in the process of homology modelling at its different levels namely data retrieval, sequence similarity search, sequence alignment, generation of the necessary input files, generation of models, validation of models, structural alignment and visualization of the generated models.The main aim was to reduce the human effort involved the overall process. Here, with the help of this tool, services can be used from the user’s desktop instead of visiting all the required websites. Thus the tool is found to reduce the time taken at every step.Since the input files are generated automatically, the probability of making errors is reduced by a greater extent. For generating the models, the tool runs the necessary commands in the background and gives the user the overall output in a better representation and finally the generated model is visualized with integrated JMOL plugin. We designed an efficient and user-friendly tool that can integrate the available tools and the overall process of homology modelling was made easier. Many modules are planned for future additions 1.Remaining modules available in MODELLER are to be integrated. 2.Inclusion of modules required for molecular dynamics. 3.Inclusion of a background worker to improve the efficiency.

REFERENCES

[1] http://www.salilab.org/modeller/manual/

[2] Chothia C and Lesk AM (1986). The relation between the divergence of sequence and structure in proteins. *EMBO J*5:823-6.

[3] Kaczanowski S and Zielenkiewicz P (2010). Why similar protein sequences encode similar three-dimensional structures? *Theoretical Chemistry Accounts* 125:543-50

[4] Marti-Renom MA, Stuart AC, Fiser A, Sanchez R, Melo F, Sali A. (2000). Comparative protein structure modeling of genes and genomes. *Annu Rev Biophys Biomol Struct* 29: 291-325.

[5] http://www.salilab.org/modeller/manual/node13.html

[6] R.C.G. Holland, T. Down, M. Pocock, A. Prlic 4, D. Huen, K. James, S. Foisy 6, A. Dräger, A. Yates, M. Heuer and M.J. Schreiber . BioJava: an Open-Source Framework for Bioinformatics.

[7] http://code.msdn.microsoft.com/

[8] http://www.ebi.ac.uk/Tools/webservices/services/dbfetch

[9] http://www.ebi.ac.uk/Tools/webservices/services/wublast

[10] http://www.ebi.ac.uk/Tools/webservices/services/ncbiblast

[11] http://www.ebi.ac.uk/Tools/webservices/services/clustalw2

[12] http://msdn.microsoft.com/en-us/library/sect4ck6.aspx

[13] http://msdn.microsoft.com/en-us/library/y4wf33f0.aspx

[14] http://www.w3schools.com/xml/default.asp

[15] www.w3schools.com/**soap**/default.asp