dynJET2 User Manual

Elodie Laine

Laboratory of
Computational and
Quantitative Biology
elodie.laine@upmc.fr

Abstract

dynJET2 predicts interacting patches at the surface of proteins based either on protein sequence and structure analysis or on any pre-computed residue based property. Sequence analysis is used to derive evolutionary conservation levels and physico-chemical properties. Structure analysis is used to characterize the local geometry of the protein surface. The pre-computed property can be for example docking-based propensities to be found at an interface.

1 INSTALLATION

1.1 Download

The dynJET^2 package is available at `www.lcqb.upmc.fr/dynJET2`.

1.2 System requirements

dynJET^2 runs on Linux or Mac OS X. The program requires some external tools that should be installed:

- java6
- ClustalW v2.1, a tool for performing multiple alignment of nucleic acid and protein sequences. It is run with either Blosum62, Gonnet or HSDM matrix by automatic selection
  Thompson JD, Higgins DG, Gibson TJ. (1994) Nucleic Acids Res., 22:4673-4680
- Naccess v2.1.1, a program that calculates the accessible area of a molecule from a PDB (Protein Data Bank) format file
  Hubbard S. J., Thornton J. M. (1993), University College London
- PSI-BLAST from BLAST+ Toolkit (v2.2.27 or more recent). The program can be called on the server or locally. To reach similar outputs on the server and on a local machine, a local call to PSI-BLAST is coded in JET2, that uses the option -t 2 setting the composition-based score adjustment method conditioned on sequence properties.
  Altschul SF, Gish W, Miller W, Myers EW, Lipman D. (1990) J Mol Biol., 215, 403410

1.3 How to install dynJET^2

1. Unzip the dynJET^2.zip in the directory of your choice.
2. Set up dynJET^2 home directory in your .bashrc or .cshrc file:

```
export dynJET2_PATH = path_of_dynJET2_HOME_directory (bash syntax)
setenv dynJET2_PATH path_of_dynJET2_HOME_directory (csh syntax)
```

2 EXECUTION AND FUNCTIONALITIES

The dynJET^2 method requires as input a protein query sequence for which three-dimensional structural coordinates are available in the Protein Data Bank (PDB).

2.1 How to run dynJET^2

The command line to execute dynJET^2 is:

```
java -cp $dynJET2_PATH:$dynJET2_PATH/jet/extLibs/vecmath.jar jet.JET
```

dynJET^2 offers a number of functionalities whose choice is controlled by the option *-p*:

- **A** Compute the accessibility surface areas of the atoms and residues using Naccess
- **V** Compute the circular variances of the residues
- **J** Launch Joint Evolutionary Trees analysis to evaluate residues conservation levels
- **C** Run the clustering algorithm to define binding patches
- **G** Insert dynJET^2 results in the B-factors column of the input PDB file

Example of a dynJET^2 analysis:

1. Create a working directory and put the input PDB file in it
2. Copy the `default.conf` configuration file from the $dynJET2_PATH directory and edit it to modify the locations of the software and to set the parameters values you wish
3. Launch dynJET^2

```
java -cp $JET2_PATH:$JET2_PATH/jet/extLibs/vecmath.jar jet.JET -c <default.conf>
-i <struct.pdb> -o ‘pwd‘ -p AVJC -r local -a 3 -d chain
```

The arguments given are :

- the configuration file to be used (*-c*)
- the input PDB file (*-i*)
- the directory where output files should be stored (*-o*)
- the type(s) of analysis to run (*-p*)
- the mode for PSI-BLAST call (*-r*)
- the scoring scheme to predict patches (*-a*)
- the way the input PDB file should be treated (*-d*), either as a complex (complex) or by considering each chain individually (chain) – this option will not impact the calculation of evolutionary conservation levels but it will affect the calculation of solvent accessible surface area and circular variance

2.2 Access analysis

The access analysis (*-p* A) computes accessible surface areas using Naccess at the level of the atoms (`<struct>_atomAxs.res`) and of the residues (`<struct>_axs.res`). In the output file `<struct>_axs.res`, the two last columns give the per-residue asa absolute value (surfAxs) and relative value (percentSurfAxs). The third column (axs) enables to classify residues according to their asa:

- 1.0 for exposed residues, which display a relative asa above the Access res_cutoff (0.05 by default)
- 0.8 for buried residues whose absolute asa is yet not zero and which are neighbors to at least one residue with non-null absolute asa
- 0.5 for completely buried residues (absolute asa equals zero) which are neighbors to at least one residue with non-null absolute asa
- 0.3 for buried residues with non-null absolute asa and isolated (not neighbors to any residue with absolute asa above zero)
- 0.0 for residues that are both completely buried (absolute asa equals zero) and isolated (not neighbors to any residue with absolute asa above zero)

These values are also reported in the fourth column (axs) of dynJET^2 global output file `<struct>_jet.res`.

2.3 Circular variance analysis

Circular variance is a measure of the density of protein around an atom. It is expressed as one minus the resultant of the vectors defined from the atom considered to the other atoms of the protein. The circular variance analysis (*-p* V) computes the circular variance for all protein atoms, which is then averaged by residue. The resulting value is comprised between 0 and 1.
The calculation is performed by considering only atoms located in a sphere around the atom of interest. The obtained resolution of the local geometry of the protein surface depends on the radius of the sphere. Two different radii are employed, resulting in two output circular variance files:

1. `<struct>_cvlocal.res` for a fine resolution (radius of 12Å by default)
2. `<struct>_cv.res` for a coarse resolution (radius of 100Å by default)

The radius set for the coarse resolution can be modified in the configuration file (max_dist parameter). The circular variances computed with this radius are reported in the fifth column (cv) of dynJET^2 global output file `<struct>_jet.res`.

2.4 Joint Evolutionary Trees analysis

Conservation levels are computed by the JET method from phylogenetic trees constructed using sequences homologous to the query sequence and sampled by a Gibbs-like approach. They reflect the biological importance of the amino-acid residues. dynJET^2 is interfaced to external tools for performing this analysis: (*i*) PSI-BLAST to retrieve the homologous sequences, (*ii*) ClustalW to perform the multiple sequence alignments.

2.4.1 Retrieval of homologous sequences using PSI-BLAST

PSI-BLAST can be called on the server or locally. The choice between the two modes is controlled by the *-r* option (*server* or *local*). The retrieval of homologous sequences from BLAST database may take some time. It is possible to do it once for all, and then re-use the same PSI-BLAST output file over different runs of dynJET2. For this, the *-r* option should be used in *input* mode and the location of the PSI-BLAST output file should be given via the *-b* option:

```
java -cp $dynJET2_PATH:$dynJET2_PATH/jet/extLibs/vecmath.jar jet.JET -c <default.conf>
-i <struct.pdb> -o ‘pwd‘ -p AVJC -r input -b <dir_psiblast_files> -a 3 -d chain
```

2.4.2 Calculation of the conservation levels

Retrieved homologous sequences are appropriately filtered and sampled by dynJET^2. Then they are aligned by using ClustalW and phylogenetic trees are constructed. From every tree, dynJET^2 computes an evolutionary trace for each position in the query sequence. Traces are then averaged over the trees, to get statistically significant values. These values are reported in the trace column of dynJET^2 global output file `<struct>_jet.res`.

2.5 Clustering analysis

dynJET^2 detects putative binding patches at the surface of the protein based on three sequence- and structure-based residue descriptors: evolutionary information (traceMax), physico-chemical properties (PC) and local geometry (CV). Additionally, patches can be predicted by using any pre-computed residue-based property (for example, docking-inferred residue propensities to be found at an interface). The values for this property should be inserted in the B-factor column of the input PDB file.
dynJET^2 implements several scoring schemes whose choice is controlled by the option *-a*:

- **0** The most appropriate strategy is automatically determined by dynJET^2

- **1** Conservation levels are used for cluster seed detection and extension (deprecated)
- **2** Conservation levels and physico-chemical properties are used for cluster seed detection and extension (deprecated)
- **3** SC_{cons} *aka* SC1: conserved seeds are detected (traceMax), then extended with conserved and physico-chemically favorable layers (traceMax + PC) ; finally, an outer layer of physico-chemically favorable and protruding residues is added (PC + CV)
- **4** SC_{notLig} *aka* SC2: conserved and not too buried seeds are detected (traceMax + PC), then extended with conserved and not too buried layers (traceMax + PC) ; finally, an outer layer of physico-chemically favorable and protruding residues is added (PC + CV)
- **5** SC_{geom} *aka* SC3: physico-chemically favorable and protruding residues are used to form the seed, the extension and the outer layer (PC + CV)
- **6** SC_{dock} *aka* SC4: a pre-computed property is exclusively used to define the seed, the extension and the outer layer

The user can decide to run only the seed detection step, the seed detection and extension steps, or all three steps of dynJET^2 clustering procedure. S/he can do this by setting the parameter *layers* to 1, 2 or 3 (default value) in the configuration file. S/he can also choose to run only one round of the clustering algorithm (main clusters) or to complement the detected clusters by a second round of the algorithm that employs a complementary scoring strategy. This functionality is controlled by the parameter *complete* in the configuration file. The results of the clustering procedure are reported in the cluster and clusnumber columns of dynJET^2 global output file `<struct>_jet.res`.

2.6 Iterative mode

An iterative version of dynJET^2 (idynJET^2) is also proposed for large-scale predictions. idynJET^2 provides a list of consensus residues belonging to interaction patches and enables to explore the set of potentially interacting residues by varying the consensus threshold during iterations. Typically consensus residues detected in at least 2 iterations are considered as robust prediction. idynJET^2 is called using the *-n* option:

```
java -cp $dynJET2_PATH:$dynJET2_PATH/jet/extLibs/vecmath.jar jet.JET -c <default.conf>
-i <struct.pdb> -o ‘pwd‘ -p AVJC -r local -d chain -n 10
```

Conservation levels and cluster values computed over n iterations are reported in the trace_i, cluster_i and clusnumber_i columns, $i = 0..(n-1)$, of dynJET^2 global output file `<struct>_jet.res`. The maximum trace value over the different dynJET^2 runs and the occurrences of every residue in a predicted cluster are reported in the traceMax and clustersOccur column of `<struct>_jet.res`.

2.7 Visualization of the results

Any column of dynJET^2 global output file `<struct>_jet.res` can be inserted in the B-factor column of the input PDB file (*-p* G). This provides a simple means to visualize dynJET^2 results with a 3D visualization tool such as PyMoL, by simply coloring residues according to their B-factors.
The columns of interest in `<struct>_jet.res` are specified by using the option *-g*. Typically, one may want to visualize the maximum conservation levels per residue and the occurrences of residues in predicted clusters:

```
java -cp $dynJET2_PATH:$dynJET2_PATH/jet/extLibs/vecmath.jar jet.JET -c <default.conf>
-i <struct.pdb> -o ‘pwd‘ -p G -r input -d chain -n 10 -g traceMax,clustersOccur
```

3 PARAMETERS

dynJET^2 is a highly versatile tool that enables the user to tune all parameters depending on the biological question asked. Default values for most of the parameters are set in the `default.conf` file.

3.1 PSIBLAST sequence retrieval

- eValue= $1.0E-5$, psiblast maximum expected value threshold
- results= 5000, maximum number of retrieved sequences
- url= $http://www.ncbi.nlm.nih.gov/BLAST/Blast.cgi$, BlastQ server URL
- database: nr, database used
- matrix: blosum62, matrix used to fetch homologs
- gap_existence= 11, gap opening cost
- gap_extension= 1, gap extension cost
- max_iter= 3, number of iterations for the PSI-BLAST search

3.2 Sequence filtering

- min_identity= 0.2, minimum sequence identity
- max_identity= 0.98, maximum sequence identity
- length_cutoff= 0.8, minimum sequence length expressed in number of residues

3.3 Tree construction

- coverage= 0.95, trace levels are computed for a maximum of 95% of the residues of the reference sequence
- msaNumber= -1, number of alignments (trees), automatically computed by JET depending on the number of retrieved sequences if set to -1
- seqNumber= -1, number of sequences aligned for each tree, automatically computed by JET depending on the number of retrieved sequences if set to -1

3.4 Accessibility calculations

- probe_radius= 1.4, radius of the probe used for accessible surface calculation by Naccess
- res_cutoff= 0.05, minimum percentage accessible surface of a residu
- atom_cutoff= 0.01, minimum accessible surface of an atom

3.5 Interface determination

- cutoff= 0, minimum percentage accessible surface variation of an interface residu

3.6 Circular variance calculation

- max_dist= 100, cutoff distance to compute coarse-resolution circular variances

3.7 Clustering

- max_dist= 5.0, maximum distance between atoms to aggregate clusters
- analysis= 0, strategy used for the clustering procedure
- layers= 3, number of components (seed, extension, outer layer) of the clusters to detect
- complete= 1, primary predictions are complemented by using another scoring strategy