BIOINFORMATICS APPLICATIONS NOTE
Vol. 24 no. 17 2008, pages 1955–1956
doi:10.1093/bioinformatics/btn344

Structural bioinformatics

TRITON: a graphical tool for ligand-binding protein engineering

Martin Prokop[1], Jan Adam[1], Zdeněk Kříž[1], Michaela Wimmerová[1,2] and Jaroslav Koča[1,*]

[1]National Centre for Biomolecular Research and [2]Department of Biochemistry, Faculty of Science, Masaryk University, Kotlářská 2, 611 37 Brno, Czech Republic

Received on April 2, 2008; revised on July 2, 2008; accepted on July 3, 2008

Advance Access publication July 4, 2008

Associate Editor: Dmitrij Frishman

ABSTRACT

Summary: The new version of the TRITON program provides user-friendly graphical tools for modeling protein mutants using the external program MODELLER and for docking ligands into the mutants using the external program AutoDock. TRITON can now be used to design ligand-binding proteins, to study protein–ligand binding mechanisms or simply to dock any ligand to a protein.

Availability: Executable files of TRITON are available free of charge for academic users at http://ncbr.chemi.muni.cz/triton/

Contact: triton@chemi.muni.cz

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

Automated protein–ligand docking is an effective method for predicting the conformation of a ligand bound to a receptor and the structure of the complex (Sousa *et al*., 2006). Although the method is typically used for screening a library of potential inhibitors (Kitchen *et al*., 2004), it can also be useful in computational protein design (Kraemer-Pecore *et al*., 2001). A possible strategy in designing ligand-binding proteins is the combination of docking methods with computational site-directed mutagenesis. In this approach, the 3D structure of a wild-type protein is used as a template for modeling protein mutants with amino acid residue substitutions (deletions, insertions) at a defined site, typically at the binding site of the protein. Subsequently, ligands are docked into these protein mutants to investigate their binding properties such as ligand-binding modes and affinities. Protein mutants with predicted high binding affinity or desired ligand specificity can then be studied experimentally. *In silico* design can substantially reduce the amount of experimental work and it can assist in experimental protein design studies.

Since computational site-directed mutagenesis combined with subsequent modeling of mutant properties is accompanied by the processing of high amounts of input and output data for computational programs, development of user-friendly graphical software, which would automate these operations, is highly desirable.

2 METHODS

Our idea with TRITON software development was to create a user-friendly graphical tool that would automate and simplify utilization of computational software suitable for computational protein design. In the previous version of TRITON, we have implemented computational site-directed mutagenesis methodology to study enzymatic reactions (Prokop *et al*., 2000).

Here, we present a new version of TRITON which is focused on studies of the properties of protein–ligand complexes. The computational site-directed mutagenesis was transferred from the previous version and mutant protein structures are modeled by external software MODELLER (Sali and Blundell, 1993). Additionally, ligand-binding modes and affinities of these mutants are calculated by docking software AutoDock (Morris *et al*., 1998). Preparation of input data for these external programs and visualization of the output is performed by TRITON.

The program and the methodology were tested on docking monosaccharide ligands into PA-IIL lectin of *Pseudomonas aeruginosa* and its mutants S22A, S23A and G24N (see Supplementary Material for details).

3 IMPLEMENTATION

Program TRITON is a user oriented software with graphical interface that enables visualization of molecular structures, preparation of input files for computational software and analysis of output data. Computational data are organized in hierarchically structured projects. For each calculation, a separate project is created. Projects are displayed in the form of a tree list in the main window of the program (Supplementary Fig. 1) which enables fast access to input and output data. For user-friendly preparation of input data, TRITON offers wizards that lead the user step by step in the process of input structures, parameters and other data specifications. In the current version, four wizards are available: *Mutagenesis* for modeling mutants by MODELLER, *Docking* for protein–ligand docking by AutoDock, *Reaction* for calculation of reaction pathways by MOPAC and *Optimization* for optimization of structure geometry by MOPAC. Specific tools for analysis of output data of calculations are also implemented. Here, we will describe only mutagenesis, which has partially been improved from the previous version of TRITON, and docking, which is a new option not included in the previous version of TRITON.

3.1 Mutagenesis

The *Mutagenesis* wizard assists in specification of input structure of a protein wild-type in PDB format (which is used as a template for homology modeling by MODELLER). One-, two- or multiple-point mutations are possible by specification of residues to be mutated and the required substitutions. Also parameters for MODELLER have to be set. Multiple preconfigured versions of MODELLER can be used.

[*]To whom correspondence should be addressed.

Computations can be run directly from the graphical interface of TRITON on a local computer. For each mutant, a separate project is generated with related input data files. After finishing computations, input and output data are accessible from each project. They can be visualized using standard tools described below.

3.2 Protein–ligand docking

The *Docking* wizard is used for specification of input data for protein–ligand docking calculations. First, input structure of the receptor protein is specified in the wizard. Then superfluous molecules, e.g. crystallographic waters or unwanted ligands, can be removed. Hydrogen atoms have to be added to protein residues if they are not present in the input file. Next, partial atomic charges have to be set. Two types of charges are implemented: united atom charges (Weiner *et al*., 1984) are recommended for AutoDock version 3 and Gasteiger PEOE charges (Gasteiger and Marsili, 1980) should be used with AutoDock version 4. Subsequently, solvation parameters and atom types are set automatically, although they can also be modified manually. A docking box, which defines the area where ligand moves during the docking procedure, is also specified. Some sidechains of the binding site can be marked to be flexible if required. In the next part of the wizard, the user specifies a file containing the structure of the ligand. Hydrogens can be added to the ligand structure and atom types can also be specified. Then, active bonds, i.e. bonds to be rotated during the docking procedure, are marked. After completing the above specifications, the docking method and parameters for AutoDock are set. Calculation can be run automatically on the local computer or manually on a remote computer. After finishing the calculation, output data are displayed in the *Output Data* folder of the project (Supplementary Fig. 1). TRITON loads output structures into the main window and displays a dialog box where binding modes can be chosen from a list which is sorted by model or cluster number or by calculated binding energy. Visualization of affinity maps help to investigate which areas have high affinity of specified ligand atoms toward the receptor. Additionally, a graph depicting electrostatic interactions of individual ligand atoms with receptor residues can be generated. Structures of calculated protein–ligand complexes can be saved in PDB format. If a new calculation with different parameters is required, it is possible to use the 'project cloning' function. In this case, input structures and settings are copied to the new project from the existing user-specified project. The parameters and settings can then be modified in the wizard as required.

3.3 Graphical tools

Program TRITON offers the basic tools needed to manipulate 3D molecular structures. It can handle files in PDB and Mol2 formats as well as AutoDock input files (PDBQ, PDBQS, PDBQT) and MOPAC input and output files. Structures can be visualized as a 3D model in various representations (wire, stick, ball and stick, CPK) and colors. The source file from which the structure was loaded can be displayed in interactive form, i.e. with the possibility to click on atoms in the list provided. An interactive list of residues is provided for fast selection, coloring and other operations on the protein structure. Another tool enables adding polar and non-polar hydrogen atoms automatically to the standard protein residues or manually to atoms of non-standard residues. Measurement of distances and angles between atoms is also possible, as well as structure superposition, visualization of hydrogen bonds, working with alternate sidechain locations, preparation of simple animations, etc. Detailed description of TRITON features can be found in online manual: http://ncbr.chemi.muni.cz/triton/manual.pdf.

3.4 System requirements, availability

Program TRITON is written in C++ language and it uses an OpenGL interface for 3D rendering. It is ported to a Linux operating system and a version for MS Windows is under development. Executable files are available free of charge for academic users at http://ncbr.chemi.muni.cz/triton/

4 CONCLUSION

The new version of TRITON allows for protein mutant modeling and the docking ligand molecules into them. Implemented methodology can be used to design protein receptors with desired ligand binding affinity or specificity. It can also be used to investigate mechanisms of protein–ligand interactions and to assess the role of individual residues in the binding. Program TRITON substantially simplifies the work related to input data preparation and consequently decreases the probability of human errors. Thanks to its user-friendly interface, TRITON is helpful not only for computational chemists but also for experimentalists providing assistance and direction in preparation of laboratory work.

ACKNOWLEDGEMENTS

Funding: This work was funded by the Ministry of Education of the Czech Republic (MSM0021622413 to M.P. and J.K., LC06030 to Z.K.); Grant Agency of the Czech Republic (GA303/06/570 to M.W. and J.A.).

Conflict of Interest: none declared.

REFERENCES

Gasteiger,J. and Marsili,M. (1980) Iterative partial equalization of orbital electronegativity – a rapid access to atomic charges. *Tetrahedron*, **36**, 3219–3288.

Kitchen,D.B. *et al*. (2004) Docking and scoring in virtual screening for drug discovery. *Nat. Rev. Drug. Discov.*, **3**, 935–948.

Kraemer-Pecore,C.M. *et al*. (2001) Computational protein design. *Curr. Opin. Chem. Biol.*, **5**, 690–695.

Morris,G.M. *et al*. (1998) Automated docking using a Lamarckian genetic algorithm and an Empirical binding free energy function. *J. Comput. Chem.*, **19**, 1639–1662.

Prokop,M. *et al*. (2000) TRITON: in silico construction of protein mutants and prediction of their activities. *Bioinformatics*, **16**, 845–856.

Sali,A. and Blundell,T.L. (1993) Comparative protein modelling by satisfaction of spatial restraints. *J. Mol. Biol.*, **234**, 779–815.

Sousa,S.F. *et al*. (2006) Protein ligand docking: current status and future challenges. *Proteins*, **65**, 15–26.

Weiner,S.J. *et al*. (1984) A new force-field for molecular mechanical simulation of nucleic-acids and protein. *J. Am. Chem. Soc.*, **106**, 765–784.