FiberDock: a web server for flexible induced-fit backbone refinement in molecular docking

Efrat Mashiach¹, Ruth Nussinov²,³ and Haim J. Wolfson¹,*

¹Blavatnik School of Computer Science, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv 69978, Israel, ²Basic Science Program, SAIC-Frederick, Inc., Center for Cancer Research Nanobiology Program, NCI – Frederick, Frederick, MD 21702, USA and ³Department of Human Genetics and Molecular Medicine, Sackler Faculty of Medicine, Tel Aviv University, Tel Aviv 69978, Israel

Received January 29, 2010; Revised April 13, 2010; Accepted April 26, 2010

ABSTRACT

Protein–protein docking algorithms aim to predict the structure of a complex given the atomic structures of the proteins that assemble it. The docking procedure usually consists of two main steps: docking candidate generation and their refinement. The refinement stage aims to improve the accuracy of the candidate solutions and to identify near-native solutions among them. During protein–protein interaction, both side chains and backbone change their conformation. Refinement methods should model these conformational changes in order to obtain a more accurate model of the complex. Handling protein backbone flexibility is a major challenge for docking methodologies, since backbone flexibility adds a huge number of degrees of freedom to the search space. FiberDock is the first docking refinement web server, which accounts for both backbone and side-chain flexibility. Given a set of up to 100 potential docking candidates, FiberDock models the backbone and side-chain movements that occur during the interaction, refines the structures and scores them according to an energy function. The FiberDock web server is free and available with no login requirement at http://bioinfo3d.cs.tau.ac.il/FiberDock/.

INTRODUCTION

Most of the activities of living cells are performed by protein–protein interactions that form molecular complexes. Accurate modeling of the 3D structure of a complex assists in understanding its function in the cell. Additionally, atomic structures of molecular complexes are used in the field of drug design, permitting the design of small molecules that prevent or induce the formation of certain complexes. In some cases, the 3D structure of protein–protein complexes can be determined experimentally by X-ray crystallography or NMR spectroscopy. However, it is an extremely difficult and time-consuming task. Therefore, the ability to predict the structure of complexes by computational means is essential.

Protein–protein docking algorithms aim to predict the structure of a complex given the atomic structures of the proteins that assemble it. Due to protein flexibility, the structure of each individual protein (unbound conformation) is often rather different from its structure in the complex (bound conformation). Docking algorithms must therefore take the protein flexibility into account (1). This is currently the major challenge in the docking field. Protein flexibility, which includes both backbone and side-chains movements, adds a huge number of degrees of freedom to the search space, making it impossible for naive search algorithms to find the native structure of the complex. Thus, a two-stage docking protocol is often used: performing a fast soft rigid docking (rigid docking that allows a certain amount of steric clashes), followed by flexible refinement of the results. Applying a soft rigid-docking method on the unbound structures of two proteins often results in a near-native solution that is poorly ranked due to steric clashes and bad shape complementarity. The goal of the flexible refinement stage is to model the conformational changes that the proteins undergo, and thus to resolve the clashes and improve their shape complementarity. Re-scoring the refined solutions by a binding energy score significantly improves the ranking of near-native models. Obviously, the success of the flexible refinement stage strongly depends on the existence of a near native model in the initial rigid-docking solutions.

*To whom correspondence should be addressed. Tel: +972-3-640-8268; Fax: +972-3-640-6476; Email: wolfson@tau.ac.il

The publisher or recipient acknowledges right of the US Government to retain a non-exclusive, royalty-free license in and to any copyright covering the article.

© The Author(s) 2010. Published by Oxford University Press.

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/2.5), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Today, most docking refinement methods model only the side-chain flexibility and adjust the rigid-body orientations of the proteins. Modeling the backbone flexibility is considered to be a more difficult task that is addressed by only few, recently developed refinement methods (2–8).

There are many freely available web servers that deal with different aspects of the docking field. Rigid-body docking can be performed by PatchDock (9), ZDOCK (10), GRAMM-X (11), Hex (12) and SymmDock (9). ClusPro (13) filters, clusters and ranks docking solution candidates. The RosettaDock web server (14) performs local search in the vicinity of a single given input complex structure by optimizing rigid-body orientation and side-chain conformations. The NOMAD-Ref server (15) uses normal mode analysis to refine one of the molecules in a single-docking model. The FireDock web server (16), refines the rigid-body orientation and side-chain conformations of up to 1000 rigid-body solution candidates and re-scores the refined structures according to a binding energy function. The HADDOCK web server (17) performs experimental data-driven docking followed by a semi-flexible refinement.

In this article, a web server of a new flexible refinement method, called FiberDock, is presented. It is the first docking refinement web server that handles both backbone and side-chain flexibility and optimizes the relative rigid-body orientation of the proteins. Side-chain movements are modeled by a rotamer library and the backbone flexibility is modeled by an unlimited number of normal modes (18). Previous research has shown the importance of using high-frequency normal modes for modeling induced-fit conformational changes (19–21). While other, previously developed, refinement methods use only the first few normal modes, with the lowest frequency (2,3), FiberDock uses both low- and high-frequency modes. Hence, it is able to model both global and local conformational changes. The method was assessed on 20 test systems in which the receptor’s interface RMSD, varies in the range of 0.59–6.08 Å. The results showed that the method successfully models backbone movements that occur during molecular interactions, and that the inclusion of the backbone refinement stage improves both the accuracy and the ranking of near-native docking solution candidates (21).

The FiberDock server can refine up to 100 rigid-docking solution candidates. The user uploads or specifies codes of two PDB (Protein Data Bank (27)) files, receptor and ligand, and provides a list of up to 100 transformations. Each transformation, when applied on the ligand, produces a candidate docking solution. If no transformation file is uploaded the identity transformation is used. Alternatively, the user can upload a PDB file that contains the rigid-docking solutions as a set of models. The candidate solutions for FiberDock can be generated by any rigid-body docking methods favored by the user (such as PatchDock (9,28), ZDOCK (10,29), GRAMM-X (11), Hex (12), etc.). In addition, the user can choose whether...
The parameters of the backbone refinement stage include the number of lowest frequency normal modes that will be considered in the refinement. By specifying a small number (10 for example), the user restricts the backbone movements to be relatively global, whereas a high number of normal modes will allow the algorithm to use high-frequency modes, which describe local movements (if they correlate well with the chemical forces that the proteins apply on each other). In addition, the user can set the level of backbone flexibility. In order to prevent the backbone from over distorting, a penalty term is introduced into the backbone minimization step. The level of backbone flexibility determines the weight of this penalty term. The higher the level, the lower the weight. A value of 0.95 (the default value) was found to suit most of our test cases.

For the rigid-body optimization stage, the user can set the number of MC iterations. In general, increasing this value improves the search for a local minima in the vicinity of the ligand’s current position. However, according to our experience, the optimization usually converges after 50 iterations.

The complex type parameter (Default, Antibody-Antigen or Enzyme-Inhibitor), is used for adjusting the weights of the scoring function for a specific biological system. The parameter of atomic radius scale influences the extent of acceptable steric clashes in the final refined solutions. This parameter scales down the radius of the atoms, affecting the VdW terms that are used in all of the three refinement stages and the final calculated binding energy.

### Output

When the refinement is finished, a web page with the results is generated and a link to it is sent to the e-mail address specified by the user. This web page (Figure 2) contains a table in which each row corresponds to a single refined solution. Each row specifies the rank of the solution according to the binding energy value, its original number (according to the given transformation file), the global binding energy value and the values of four of the energy terms (Attractive VdW, repulsive VdW, ACE and hydrogen bonds). The table is sorted by the binding energy of the refined solution. The user can view the 3D structure of each refined complex in a Jmol applet window (33). The different structures can be viewed simultaneously, allowing the user to easily compare different models. The PDB files of the refined solutions can be downloaded, and so can the full results table that details the values of all the energy terms, for each solution. This table also specifies the linear combination of normal modes that generates the refined backbone conformation of the receptor and the ligand.

### CONCLUSIONS

Handling backbone flexibility is currently the main challenge in the docking field. In many cases, even a slight backbone movement prevents near-native rigid-docking solutions from being highly ranked, since these models will often contain steric clashes. Therefore, flexible refinement is needed in order to resolve these clashes by backbone and side-chain movements and a minimization of the rigid-body orientation. The FiberDock method was developed to meet this challenge. This new method mimics an induced fit-process. The backbone and side-chain movements are inferred from the vDW forces that the proteins apply on each other. The method models backbone movements by normal modes. It uses both low- and high-frequency modes and therefore is able to...
model both global and local conformational changes, such as opening of binding sites and loop movements.

In order to make this method available for the entire biological community, a clear and user-friendly web server was developed, which requires no previous knowledge in docking algorithms. This is the first web server for flexible docking refinement, which models both backbone and side-chain flexibility. It refines a single rigid-body docking solution in an average time of 14s. Therefore, it can be used for refining and re-ranking of up to 100 solutions in a reasonable time. The FiberDock software (for Linux users) can also be downloaded from the web site. The downloaded version does not restrict the amount of refined docking solutions. We believe that this server will be very useful to the biological community. It can help model new structures of protein–protein complexes and as such improve our understanding of protein functions in the living cell.

ACKNOWLEDGEMENTS

Molecular graphics images were produced using the UCSF Chimera package from the Resource for Biocomputing, Visualization, and Informatics at the University of California, San Francisco. The content of this publication does not necessarily reflect the views or policies of the Department of Health and Human Services, nor does mention of trade names, commercial products, or organizations imply endorsement by the US Government.

FUNDING

Adams fellowship of the Israel Academy of Sciences and Humanities (to E.M., in part); Israel Science Foundation (grant no. 1403/09, to H.J.W., in part); Hermann Minkowski Minerva Geometry Center; National Cancer Institute, National Institutes of Health (Federal funds, contract number HHSN261200800001E). Intramural Research Program of the NIH, National Cancer Institute, Center for Cancer Research (in part). Funding for open access charge: National Institutes of Health (contract number HHSN261200800001E).

Conflict of interest statement. None declared.

REFERENCES

1. Andrusier, N., Mashiach, E., Nussinov, R. and Wolfson, H.J. (2008) Principles of flexible protein-protein docking. *Proteins*, 73, 271–289.
2. Lindahl, E. and Delarue, M. (2005) Refinement of docked protein-ligand and protein-DNA structures using low frequency normal mode amplitude optimization. *Nucleic Acids Res.*, 33, 4496–4506.
3. May, A. and Zacharias, M. (2008) Energy minimization in low-frequency normal modes to efficiently allow for global flexibility during systematic protein-protein docking. *Proteins*, 70, 794–809.
4. May, A. and Zacharias, M. (2007) Protein-protein docking in CAPRI using ATTRACT to account for global and local flexibility. *Proteins*, 69, 774–780.
5. Wang, C., Bradley, P. and Baker, D. (2007) Protein-protein docking with backbone flexibility. *J. Mol. Biol.*, 373, 503–519.
6. Chaudhury, S., Sircar, A., Sivasubramanian, A., Berrondo, M. and Gray, J.J. (2007) Incorporating biochemical information and...
backbone flexibility in RosettaDock for CAPRI rounds 6-12. 

7. Fitzjohn,P.W. and Bates,P.A. (2003) Guided docking: first step to locate potential binding sites. Proteins, 52, 28–32.

8. Krol,M., Chalei,R.A., Tournier,A.L. and Bates,P.A. (2007) Implicit flexibility in protein docking: cross-docking and local refinement. Proteins, 69, 750–757.

9. Schneiderman-Duhovny,D., Inbar,Y., Nussinov,R. and Wolfson,H.J. (2005) PatchDock and SymmDock: servers for rigid and symmetric docking. Nucleic Acids Res., 33, W363–W367.

10. Chen,R., Li,L. and Weng,Z. (2003) ZDOCK: an initial-stage protein-docking algorithm. Proteins, 52, 80–87.

11. Tovchigrechko,A. and Vakser,I.A. (2006) GRAMM-X public web server for protein–protein docking. Nucleic Acids Res., 34, 310–314.

12. Ritchie,D.W. and Kemp,G.J.L. (2000) Protein docking using spherical polar Fourier correlations. Proteins, 39, 178–194.

13. Comeau,S.R., Gatchell,D.W., Vajda,S. and Camacho,C.J. (2004) Cluspro: a fully automated algorithm for protein-protein docking. Nucleic Acids Res., 32, W96–W99.

14. Lyskov,S. and Gray,J.J. (2008) The RosettaDock server for local protein-protein docking. Nucleic Acids Res., 36, W233–W238.

15. Lindahl,E., Azuara,C., Koehl,P. and Delarue,M. (2006) NOMAD-Ref: visualization, deformation and refinement of macromolecular structures based on all-atom normal mode analysis. Nucleic Acids Res., 34, W52–W56.

16. Mashiach,E., Schneiderman-Duhovny,D., Andrusier,N., Nussinov,R. and Wolfson,H.J. (2008) FireDock: a web server for fast interaction refinement in molecular docking. Nucleic Acids Res., 36, W229–W232.

17. Dominguez,C., Boelens,R. and Bonvin,A. (2003) HADDOCK: a protein-protein docking approach based on biochemical or biophysical information. J. Am. Chem. Soc., 125, 1731–1737.

18. Hinsen,K. (1998) Analysis of domain motions by approximate normal mode calculations. Proteins, 33, 417–429.

19. Petrone,P. and Pande,V.S. (2006) Can conformational change be described by only a few normal modes? Biophys. J., 90, 1583–1593.

20. Cavasotto,C.N., Koavas,J.A. and Abagyan,R.A. (2005) Representing receptor flexibility in ligand docking through relevant normal modes. J. Am. Chem. Soc., 127, 9632–9640.

Masliach,E., Nussinov.R. and Wolfson,H.J. (2009) FiberDock: flexible induced-fit backbone refinement in molecular docking. Proteins, 78, 1503–1519.

Andrusier,N., Nussinov,R. and Wolfson,H.J. (2007) FireDock: fast interaction refinement in molecular docking. Proteins, 69, 139–159.

Eriksson,O. (2001) Side chain-positioning as an integer programming problem. Lect. Notes Comput. Sci., 2149, 128–141.

Broyden,C.G. (1970) The convergence of a class of double-rank minimization algorithms. J. Inst. Math. Appl., 6, 76–90.

Fletcher,R. (1970) A new approach to variable metric algorithms. The Computer Journal, 13, 317–322.

Petersen,E.F., Goddard,T.D., Huang,C.C., Couch,G.S., Greenblatt,D.M., Meng,E.C. and Ferrin,T.E. (2004) UCSF Chimera - a visualization system for exploratory research and analysis. J. Comput. Chem., 25, 1605–1612.

Berman,H.M., Westbrook,J., Feng,Z., Gilliland,G., Bhat,T.N., Weissig,H., Shindyalov,I.N. and Bourne,P.E. (2000) The protein data bank. Nucleic Acids Res., 28, 235–242.

Duhovny,D., Nussinov,R. and Wolfson,H.J. (2002) Efficient unbound docking of rigid molecules. Lect. Notes Comput. Sci., 2452, 185–200.

Chen,R. and Weng,Z. (2002) Docking unbound proteins using shape complementarity, desolvation, and electrostatics. Proteins, 47, 281–294.

Smith,G.R., Sternberg,M.J.E. and Bates,P.A. (2005) The relationship between the flexibility of proteins and their conformational states on forming protein-protein complexes with application to protein-protein docking. J. Mol. Biol., 347, 1077–1101.

Rajamani,D., Thiel,S., Vajda,S. and Camacho,C.J. (2004) Anchor residues in protein-protein interactions. Proc. Natl Acad. Sci. USA, 101, 11287–11292.

Li,X., Keskin,O., Ma,B., Nussinov,R. and Liang,J. (2004) Protein-protein interactions: hot spots and structurally conserved residues often locate in complemented pockets that pre-organized in the unbound states: implications for docking. J. Mol. Biol., 344, 781–795.

Herraez,A. (2006) Biomolecules in the computer: Jmol to the rescue. Biochem. Mol. Biol. Educ., 34, 255–261.