Title

Predicting Protein Secondary Structure Using Consensus Data Mining (CDM) Based on Empirical Statistics and Evolutionary Information

ISBN

978-1-4939-6404-8

Authors

Kandoi, Gaurav

Leelananda, Sumudu P

Jernigan, Robert L

et al.

Publication Date

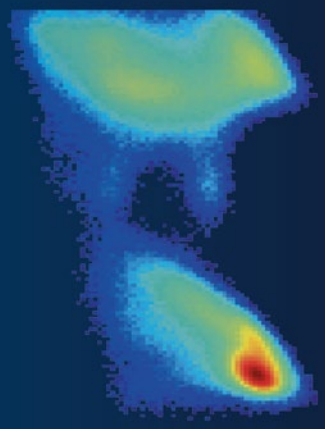
2017

DOI

10.1007/978-1-4939-6406-2_4

Methods in Molecular Biology 1484

Springer Protocols



Yaoqi Zhou
Andrzej Kloczkowski
Eshel Faraggi
Yuedong Yang *Editors*

Prediction of Protein Secondary Structure

Humana Press

METHODS IN MOLECULAR BIOLOGY

Series Editor
John M. Walker
School of Life and Medical Sciences
University of Hertfordshire
Hatfield, Hertfordshire, AL10 9AB, UK

For further volumes:
http://www.springer.com/series/7651

Prediction of Protein Secondary Structure

Edited by

Yaoqi Zhou

Institute for Glycomics and School of Information and Communication Technology, Griffith University, Southport, QLD, Australia

Andrzej Kloczkowski

Battelle Center for Mathematical Medicine, Nationwide Children's Hospital, Columbus, OH, USA; Department of Pediatrics, The Ohio State University College of Medicine, Columbus, OH, USA

Eshel Faraggi

Department of Biochemistry and Molecular Biology, Indiana University School of Medicine, Indianapolis, IN, USA; Research and Information Systems, LLC, Indianapolis, IN, USA

Yuedong Yang

Institute for Glycomics and School of Information and Communication Technology, Griffith University, Southport, QLD, Australia

Humana Press

Editors
Yaoqi Zhou
Institute for Glycomics and School of Information and Communication Technology
Griffith University
Southport, QLD, Australia

Eshel Faraggi
Department of Biochemistry and Molecular Biology
Indiana University School of Medicine
Indianapolis, IN, USA

Research and Information Systems, LLC
Indianapolis, IN, USA

Andrzej Kloczkowski
Battelle Center for Mathematical Medicine
Nationwide Children's Hospital
Columbus, OH, USA

Department of Pediatrics
The Ohio State University College of Medicine
Columbus, OH, USA

Yuedong Yang
Institute for Glycomics and School of Information and Communication Technology
Griffith University
Southport, QLD, Australia

ISSN 1064-3745 ISSN 1940-6029 (electronic)
Methods in Molecular Biology
ISBN 978-1-4939-6404-8 ISBN 978-1-4939-6406-2 (eBook)
DOI 10.1007/978-1-4939-6406-2

Library of Congress Control Number: 2016949704

Printed on acid-free paper

This Humana Press imprint is published by Springer Nature
The registered company is Springer Science+Business Media LLC New York

Preface

One of the most important challenges in molecular biology is to figure out how the one-dimensional (1D) sequence of amino acid residues in a protein at a physiological condition specifies its unique, functional three-dimensional (3D) structure. Despite more than 50 years of effort, reliable computational protein models with experimental resolution remain out of reach, except for homology models that are based on the structures of highly similar sequences. To overcome the challenge of prediction from 1D to 3D, many 1D to 1D methods have been developed as an intermediate step or a substitute for 1D to 3D prediction. These 1D quantities can be either structural or functional properties characterized by a one-dimensional vector along the protein sequence. One prominent example is protein secondary structure where protein backbone structure is annotated by a few states such as helices, sheets, or coils. Protein backbone structure can also be characterized by torsion angles. In addition to backbone structural properties, protein structures can be characterized by global structural properties: Properties that depend on interactions between multiple residues that are far apart in the sequence. One such example is the solvent accessible surface area, relevant to tertiary packing and function of proteins. More recently, predicting one-dimensional functional properties (functional sites in particular) from protein sequences has received increasing attention.

This book starts from secondary structure prediction based on sequence only (GOR, Chapters 1 and 2 and single helix prediction, Chapter 3), followed by secondary structure prediction based on evolution information (CDM, Chapter 4, SPINE-X, Chapter 5, and SPIDER2, Chapter 6). In addition to secondary structure, SPINE-X and SPIDER2 also predict solvent accessible surface areas and backbone torsion angles. The latter is reviewed in Chapter 7. Predicted secondary structures are utilized in model building (Chapters 8 and 9). Next, a few chapters focus on global structural properties (solvent accessibility in Chapter 10; intrinsically disordered regions in Chapters 11 and 12; and protein flexibility in Chapter 13). Functional properties are predicted in Chapter 14 (DNA/RNA-binding sites), Chapter 15 (RNA-binding residues), Chapter 16 (protein-binding sites), Chapter 17 (B-cell epitopes), Chapter 18 (phosphorylation sites), and Chapter 19 (post-translation modifications). Chapter 20 describes a tool for visualizing interior and protruding regions in proteins. These chapters represent a fraction of the excellent methods available in the literature. We hope that this collection will provide a guide to a few current state-of-the-art techniques that are useful for computational and experimental biologists.

Southport, QLD, Australia — *Yaoqi Zhou*
Columbus, OH, USA — *Andrzej Kloczkowski*
Indianapolis, IN, USA — *Eshel Faraggi*
Southport, QLD, Australia — *Yuedong Yang*

Contents

Contributors

Badri Adhikari • *Computer Science Department, University of Missouri, Columbia, MO, USA*

Debswapna Bhattacharya • *Computer Science Department, University of Missouri, Columbia, MO, USA*

Renzhi Cao • *Computer Science Department, University of Missouri, Columbia, MO, USA*

Jianlin Cheng • *Computer Science Department, University of Missouri, Columbia, MO, USA*

Abdollah Dehzangi • *Department of Psychiatry, Medical Research Center, University of Iowa, Iowa City, IA, USA*

Drena Dobbs • *Genetics, Development and Cell Biology Department, Iowa State University, Ames, IA, USA*

Yongchao Dou • *School of Biological Sciences, University of Nebraska–Lincoln, Lincoln, NE, USA*

Dániel Dudola • *Faculty of Information Technology and Bionics, Pázmány Péter Catholic University, Budapest, Hungary*

Yasser EL-Manzalawy • *College of Information Sciences and Technology, Pennsylvania State University, University Park, PA, USA*

Eshel Faraggi • *Department of Biochemistry and Molecular Biology, Indiana University School of Medicine, Indianapolis, IN, USA; Research and Information Systems, LLC, Indianapolis, IN, USA*

Jean Garnier • *IUPAB, International Council for Science (ICSU), Paris, France*

Zoltán Gáspári • *Faculty of Information Technology and Bionics, Pázmány Péter Catholic University, Budapest, Hungary*

M. Michael Gromiha • *Department of Biotechnology, Bhupat and Jyoti Mehta School of Biosciences, Indian Institute of Technology Madras, Chennai, Tamilnadu, India*

Rhys Heffernan • *Signal Processing Laboratory, School of Engineering, Griffith University, Brisbane, QLD, Australia*

Vasant G. Honavar • *College of Information Sciences and Technology, Pennsylvania State University, University Park, PA, USA*

Michal Jamroz • *Laboratory of Theory of Biopolymers, Faculty of Chemistry, University of Warsaw, Warsaw, Poland*

Robert L. Jernigan • *Bioinformatics and Computational Biology Program, Iowa State University, Ames, IA, USA; Department of Biochemistry, Biophysics, and Molecular Biology, Iowa State University, Ames, IA, USA*

János Juhász • *Faculty of Information Technology and Bionics, Pázmány Péter Catholic University, Budapest, Hungary*

Gaurav Kandoi • *Bioinformatics and Computational Biology Program, Iowa State University, Ames, IA, USA; Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA*

Marcin Kierczak • *Science for Life Laboratory, Department of Medical Biochemistry and Microbiology, Uppsala University, Uppsala, Sweden*
Daisuke Kihara • *Department of Biological Sciences, College of Science, Purdue University, West Lafayette, IN, USA; Department of Computer Science, College of Science, Purdue University, West Lafayette, IN, USA*
Andrzej Kloczkowski • *Battelle Center for Mathematical Medicine, Nationwide Children's Hospital, Columbus, OH, USA; Department of Pediatrics, The Ohio State University College of Medicine, Columbus, OH, USA*
Sebastian Kmiecik • *Faculty of Chemistry, University of Warsaw, Warsaw, Poland*
Andrzej Kolinski • *Faculty of Chemistry, University of Warsaw, Warsaw, Poland*
Maksim Kouza • *Faculty of Chemistry, University of Warsaw, Warsaw, Poland*
Lukasz Kurgan • *Department of Computer Science, Virginia Commonwealth University, Richmond, VA, USA*
Sumudu P. Leelananda • *Battelle Center for Mathematical Medicine, The Research Institute at Nationwide Children's Hospital, Columbus, OH, USA*
Zhixiu Li • *Translational Genomics Group, Institute of Health and Biomedical Innovation, Queensland University of Technology at Translational Research Institute, QLD, Australia*
Balázs Ligeti • *Faculty of Information Technology and Bionics, Pázmány Péter Catholic University, Budapest, Hungary*
James Lyons • *Signal Processing Laboratory, School of Engineering, Griffith University, Brisbane, QLD, Australia*
László Nyitray • *Department of Biochemistry, Eötvös Loránd University, Budapest, Hungary*
Kuldip Paliwal • *Signal Processing Laboratory, School of Engineering, Griffith University, Brisbane, QLD, Australia*
Zhenling Peng • *Center for Applied Mathematics, Tianjin University, Tianjin, China*
Lenna Peterson • *Department of Biological Sciences, College of Science, Purdue University, West Lafayette, IN, USA*
Dariusz Plewczynski • *Centre of New Technologies, University of Warsaw, Warsaw, Poland*
Sándor Pongor • *Faculty of Information Technology and Bionics, Pázmány Péter Catholic University, Budapest, Hungary*
Abdul Sattar • *Institute for Integrated and Intelligent Systems, Griffith University, Brisbane, QLD, Australia; National ICT Australia (NICTA), Brisbane, QLD, Australia*
Taner Z. Sen • *Bioinformatics and Computational Biology Program, Iowa State University, Ames, IA, USA; Department of Genetics, Development and Cell Biology, Iowa State University, Ames, IA, USA*
Alok Sharma • *Institute for Integrated and Intelligent Systems, Griffith University, Brisbane, QLD, Australia; School of Engineering and Physics, University of the South Pacific, Suva, Fiji*
Marcin Tatjewski • *Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland; Centre of New Technologies, University of Warsa, Warsaw, Poland*
Gábor Tóth • *Department of Medical and Biological Sciences, National Research, Development and Innovation Office, Budapest, Hungary*
Vladimir N. Uversky • *Department of Molecular Medicine and USF Health Byrd Alzheimer's Research Institute, Morsani College of Medicine, University of South Florida, Tampa, FL, USA; Institute for Biological Instrumentation, Russian Academy of Sciences, Moscow Region, Russian Federation; Laboratory of Structural Dynamics, Stability and*

Folding of Proteins, Institute of Cytology, Russian Academy of Sciences, Russian Federation

Roberto Vera • *Faculty of Information Technology and Bionics, Pázmány Péter Catholic University, Budapest, Hungary; National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, Bethesda, MD, USA*

Rasna R. Walia • *USDA-ARS, Ames, IA, USA*

Chen Wang • *Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada; Department of Computer Science, Virginia Commonwealth University, Richmond, VA, USA*

Jihua Wang • *Shandong Provincial Key Laboratory of Functional Macromolecular Biophysics, Dezhou University, Dezhou, Shandong, China*

Yuedong Yang • *Institute for Glycomics and School of Information and Communication Technology, Griffith University, Southport, QLD, Australia*

Bo Yao • *School of Biological Sciences, University of Nebraska–Lincoln, Lincoln, NE, USA*

K. Yugandhar • *Department of Biotechnology, Bhupat and Jyoti Mehta School of Biosciences, Indian Institute of Technology Madras, Chennai, Tamilnadu, India*

Chi Zhang • *School of Biological Sciences, University of Nebraska–Lincoln, Lincoln, NE, USA*

Tuo Zhang • *Department of Microbiology and Immunology, Weill Cornell Medical College, New York, NY, USA*

Yaoqi Zhou • *Institute for Glycomics and School of Information and Communication Technology, Griffith University, Southport, QLD, Australia*

Olav Zimmermann • *Jülich Supercomputing Centre (JSC), Institute for Advanced Simulation (IAS), Forschungszentrum Jülich GmbH, Jülich, Germany*

Chapter 1

Where the Name "GOR" Originates: A Story

Jean Garnier

Abstract

The GOR is a computer program to predict secondary structures in proteins when the amino acid sequence is known using the theory of information. The program GOR was named much later after publication in 1978 at the dawn of bioinformatics. The program is still distributed. Its development is an example of interplay between scientific friendships, pleasure of doing research, opportunities, new technologies, and lack of the intervention of any grant agencies.

Key words Information theory, GOR, Secondary structure prediction, Protein structure prediction

Among the four algorithms for protein secondary structure predictions originated in my laboratory through the years (GOR [1], SIMPA [2], COMBINE [3], and PREDANG [4]) one of them, GOR, received much attention due possibly to some circumstances that I think are worth examining in this series.

All started by meeting Roger Pain, Professor of Biochemistry at the University of Newcastle Upon Tyne, UK. At that time, in the sixties early seventies, my main research interest was to study the milk clotting by rennin (chymosin) in cheese making which turns out to result from the proteolysis of one single peptide bond in one of the three main components of the caseins. I was in particular investigating the secondary structures of the caseins mainly by optical rotatory dispersion and circular dichroism (CD) experiments, there were no X-ray data available at this time and not yet when writing this chapter. Roger, accustomed to visit France, came to my lab for enquiring on a recent apparatus of circular dichroism, developed by a French company, Jouan. Since then we kept a friendly relationship together.

Later, having attended a British Biophysical Society meeting in London, I heard a presentation by a young student, Barry Robson, about the use of the information theory to analyze the conformational properties of amino acids in proteins and it turned out that he was working in Roger's lab, in Newcastle. His talk

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_1,

attracted my interest having already experienced the recent Chou and Fasman [5] method.

In 1970 my INRA lab had moved a few miles away from Jouy en Josas to Orsay on the University of Paris-Sud campus (France). This coincided with a new project, the study of the pituitary glycoprotein hormones for the INRA Department of Animal Reproduction. These hormones have a common subunit, alpha, associated by non-covalent bonds to a different beta subunit according to their biological properties: TSH, FSH or LH and hCG. In order to get some information about these proteins I decided to apply the Chou and Fasman method to detect the regions of alpha helices and beta strands and compare with the information we could derive from the CD analysis, mostly when these chains undergo a conformational change. The experimental study could have been undertaken thanks to a generous gift of 10 mg of hCG from Robert Canfield of Columbia University, New York, it was a huge quantity for this kind of hormone! I proposed to one of my PhD student to predict the alpha chain and I will do the beta chains (90–120 amino acids each). Who made the mistake? I don't know but it turned out that both of us dealt with the same alpha chain and we did not get the same result of prediction. We checked who did wrong and concluded at the end that it was not one of us, but the un-complete description of the method: I started systematically the Chou and Fasman algorithm from the N terminal end of the peptide and the student at random in the sequence. No way in reading the article to find out how to do it one way or another. It was another example, unfortunately too frequent, when not enough information is given in the publication to reproduce the published results. Probably at that time I started to get the idea that only a method based on a distributed computer program will resolve this kind of ambiguity. Even later, at the occasion of meeting Chou at a CECAM workshop in Orsay in 1979 or by visiting Gerry Fasman in Boston, I could not clarify this point!

A short while after, Roger invited me to come to Newcastle as a visiting professor for 1 month, I accepted readily more so I was planning to talk to Barry about his recent development of the information theory he spoke about in London. This was finalized in March 1975. However when I arrived in Roger's lab, Barry had left to Manchester University for an associate professorship. Seeing my disappointment Roger proposed me to split my stay in two parts, half time in Newcastle, half time in Manchester. What a generous and a fair colleague, he has his share in the success of the GOR! During my stay with Barry in Manchester I became convinced that the information theory is well suited to handle the problem of secondary structure prediction, well scientifically founded and amenable to a computer program better than the manual and lengthy methods of Chou and Fasman or Lim [6], the two prediction methods available at that time. I proposed to Barry to write a computer program with the information values he already

have computed in Roger's lab [7, 8] however for that project he needed some kind of a sabbatical leave, I proposed him to get one in my lab in Orsay. Then finally I got the position from the university for 3 months and Barry came in my lab the last quarter of year 1976. Right upon his arrival we enlisted to the Orsay computing center, UNIVAC, a part of the Linear Accelerator Lab on the campus. Barry was the only programmer: I had no knowledge of Fortran programming, however after these 3 months I could manage to read any Fortran programs (version 7)! We worked together day and night at the computing center, mostly at night, the compilations were faster, it was an exiting period for both of us and a lot of fun for me. I could take Fortran lessons, I learned a lot about computers, compilations, how to check the robustness of a computer program and so on. We benefited of the help of talented particle physics engineers to look for bugs in our listings; at that time there were no screen editors and we were using punched cards, I have still a bunch of them in my office for the GOR program. The computing center was working on a self service basis, however my lab not being a CNRS lab, the computing time was not free for me, I did not take too much attention to it except months later I received the bill: half of my year lab budget!

Then we wrote the article describing the method. Before submitting the manuscript to J Mol Biol, Barry insisted that I sign first among the three authors, arguing that at that time that I had been so much involved in the algorithm and the writing that I deserved that position, and so I accepted without anticipating what will follow.

For several years this method, although published in a well-read journal, did not get any attention compare with what will happen later: the present number of citations is still 4331, (October 10, 2015, Google Scholar). Why this lag phase? Likely because in the late seventies and early eighties, biologists, biochemists were not using computers, the spread of using PCs will come later, or they were not having easy access to a computing center to run the program even if it was made available under request and that we published later a version in BASIC language [9]. In fact the GOR program was one of the very first tool of informatics for biologists, a field that will be called later "bioinformatics." Our predecessor in this field that I can see is Margaret O. Dayhoff who was editing since 1965 the "Atlas of Protein sequence and structure" made of a compilation on a computer of known amino acid sequences and periodically distributed to biologists or biochemists as a printed book.

The event that broke this silence was the sale in the late eighties, and still recently, of a package made of a series of computer programs by the Wisconsin University, the GCG package. It contained the GOR program listed under the name of Garnier's method, or "Garnier output file," although the complete reference was given somewhere in the description. Notice that I was never

asked by the Wisconsin team to provide them with the computer program. I suppose the authors of that package considered having indeed enough information from our J Mol Biol article, algorithm and information values, to write a computer program themselves without a request. I confess that I never checked if this program was giving the same results than the one we wrote in Orsay. However thanks to the wide diffusion of this package, the method became popular among the biologists but also the name of Garnier at the expense of the other two authors. Barry was seriously annoyed by that I think. It was rather unfair of not counting the contribution of the two others, Barry Robson and David Osguthorpe. Since then I appreciate the journals that request the authors to describe their own part in the published work. David, a Barry's student, computed and verified all the information values that we used in Orsay. To repair this injustice, we decided Barry and I to call the method "GOR" from the initials of the name of the three authors in the order of the signatures, and as we had already developed other versions, the one published in 1978 was taken as GOR I and the subsequent versions GOR II to GOR V at present. Barry on the other hand found that the word GOR close to "gore" has some strength in it.

Another remark, the last version of the Wisconsin package version 10.3 distributed in 2005 was still including the GOR I, 27 years after its publication, although the version V, more accurate, was already published [10–12]. Another package Emboss [13] is also distributing in our days the GOR I but it includes a warning that other methods give more accurate predictions however they keep GOR I because it is "simple to calculate on most workstations." I have always recommended students to not use computer programs as a black box but to acquire enough knowledge about them to ascertain that they are suited to their research problems.

References

1. Garnier J, Osguthorpe DJ, Robson B (1978) Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. J Mol Biol 120:97–120
2. Levin JM, Robson B, Garnier J (1986) An algorithm for secondary structure determination in proteins based on sequence similarity. FEBS Lett 205:303–308
3. Biou V, Gibrat JF, Levin JM, Robson B, Garnier J (1988) Secondary structure prediction: combination of three different methods. Protein Eng 2(3):185–191
4. Gibrat JF, Robson B, Garnier J (1991) Influence of the local amino acid sequence upon the zones of the torsional angles phi and psi adopted by residues in proteins. Biochemistry 30:1578–1586
5. Chou PY, Fasman GD (1974) Prediction of protein conformation. Biochemistry 13:222–245
6. Lim VI (1974) Algorithm for prediction of alph-helical and beta-structural regions in globular proteins. J Mol Biol 88:873–894
7. Robson B, Pain RH (1971) Analysis of the code relating sequence to conformation in proteins: possible implication for the mechanism of formation of helical regions. J Mol Biol 58:237–259
8. Robson B, Suzuki E (1976) Conformational properties of amino acid residues in globular proteins. J Mol Biol 107:327–356

9. Robson B, Garnier J (1986) Introduction to proteins and protein engineering. Elsevier, Amsterdam
10. Kloczkowski A, Ting K-L, Jernigan RL, Garnier J (2002) Protein secondary structure prediction based on the GOR algorithm incorporating multiple sequence alignment information. Polymer 43:441–449
11. Kloczkowski A, Ting K-L, Jernigan RL, Garnier J (2002) Combining the GOR V algorithm with evolutionary information for protein secondary structure prediction from amino acid sequence. Proteins 49:154–166
12. Sen TZ, Jernigan RL, Garnier J, Kloczkowski A (2005) GOR V server for protein secondary structure prediction. Bioinformatics 21:2787–2788, http://gor.bb.iastate.edu/
13. Emboss: http://emboss.sourceforge.net/apps/release/6.6/emboss/apps/garnier.html

Chapter 2

The GOR Method of Protein Secondary Structure Prediction and Its Application as a Protein Aggregation Prediction Tool

Maksim Kouza, Eshel Faraggi, Andrzej Kolinski, and Andrzej Kloczkowski

Abstract

The GOR method of protein secondary structure prediction is described. The original method was published by Garnier, Osguthorpe, and Robson in 1978 and was one of the first successful methods to predict protein secondary structure from amino acid sequence. The method is based on information theory, and an assumption that information function of a protein chain can be approximated by a sum of information from single residues and pairs of residues. The analysis of frequencies of occurrence of secondary structure for singlets and doublets of residues in a protein database enables prediction of secondary structure for new amino acid sequences. Because of these simple physical assumptions the GOR method has a conceptual advantage over other later developed methods such as PHD, PSIPRED, and others that are based on Machine Learning methods (like Neural Networks), give slightly better predictions, but have a "black box" nature. The GOR method has been continuously improved and modified for 30 years with the last GOR V version published in 2002, and the GOR V server developed in 2005. We discuss here the original GOR method and the GOR V program and the web server. Additionally we discuss new highly interesting and important applications of the GOR method to chameleon sequences in protein folding simulations, and for prediction of protein aggregation propensities. Our preliminary studies show that the GOR method is a promising and efficient alternative to other protein aggregation predicting tools. This shows that the GOR method despite being almost 40 years old is still important and has significant potential in application to new scientific problems.

Key words Secondary structure prediction, GOR, Information theory, Protein aggregation

1 Introduction

The prediction of protein structure from amino acid sequence is one of the most important problems in molecular biology. With large-scale genome sequencing of various organisms and individuals for personalized (precision) medicine that produces an enormous amount of amino acid sequence data, the problem became even more important. Although prediction of tertiary structure is one of the

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_2,

ultimate goals of protein science, the prediction of secondary structure from sequence is still a more feasible intermediate step in this direction. Furthermore, some knowledge of secondary structure can serve as an input for prediction. Instead of predicting the full three-dimensional structure, it is much easier to predict simplified aspects of structure, namely the key structural elements of the protein and the location of these elements not in the three-dimensional space but along the protein amino acid sequence. This reduces the complex three-dimensional problem to a much simpler one-dimensional problem. The fundamental elements of the secondary structure of proteins are alpha-helices, beta-sheets, coils, and turns. In 1983, Kabsch and Sander developed the classification of elements of secondary structure based mainly on hydrogen bonds between the backbone carbonyl and NH groups [1]. Their dictionary of secondary structure assignment Database of Secondary Structure in Proteins (DSSP) is widely used in protein science, although there are other alternative assignment methods, such as STRIDE [2]. According to the DSSP classification, there are eight elements of secondary structure assignment denoted by letters: H (alpha-helix), E (extended beta-strand), G (3_{10} helix), I (π-helix), B (bridge, a single residue beta-strand), T (beta-turn), S (bend), and C (coil). The eight-letter DSSP alphabet requires translation into the three-letter code. For instance, for the CASP (Critical Assessment of Structure Prediction) experiments, helices (H, G, and I) in the DSSP code are assigned the letter H in the three-letter secondary structure code, whereas strands (E) and bridges (B) in the DSSP code are translated into sheets (E) in the three-letter code. Other elements of the DSSP structure (T, S, C) are treated as coil (C). There are, however, other alternative ways to make these assignments.

1.1 The Original GOR Method

The GOR program is one of the first major methods proposed for protein secondary structure prediction from sequence. The original article (GOR I) was published by Garnier, Osguthorpe, and Robson in 1978, with the first letters of the authors' names forming the name of the method [3]. The method has been continuously improved and modified during the next 30 years. The first version (GOR I) used a small database of 26 proteins with about 4500 residues. The next version (GOR II) [4] used the enlarged database of 75 proteins containing 12,757 residues. Both versions predicted four conformations (H, E, C, and turns T) and were using singlet frequency. Starting with GOR III [5] the number of predicted conformations was reduced to three (H, E, and C). The GOR III method started to additionally use information about the frequencies of pairs (doublets) of residues within the window, based on the same database as the earlier version. The next version was named GOR IV [6] and it used 267 protein chains containing 63,566 residues and is still available as a web server at https://npsa-prabi.ibcp.fr/cgi-bin/npsa_automat.

pl?page=npsa_gor4.html. The latest version GOR V is using several improvements, including multiple sequence alignments and discussed in the next section [7].

Because of its simple assumptions, the GOR method has conceptual advantage over other later developed methods such as PHD [8], PSIPRED [5], SPINE-X [9], and others. While these secondary structure prediction tools rely on machine learning and typically are black boxes in terms of the principles leading to their predictions, as we briefly review below, the GOR method's reasoning for arriving at a particular prediction is clearly evident to the user. In some cases this clarity may be more significant than the slight loss in accuracy of the GOR algorithm.

The GOR algorithm is based on information theory combined with Bayesian statistics. One of the basic mathematical tools of information theory is the information function I(S,R):

$$I(S;R) = \log\left[P(S \mid R) / P(S)\right] \tag{1}$$

For the problem of protein secondary structure prediction, the information function is defined as the logarithm of the ratio of the conditional probability $P(S|R)$ of observing conformation S, [where S is one of the three states: helix (H), extended (E), or coil (C)] for residue R (where R is one of the 20 possible amino acids) and the probability $P(S)$ of the occurrence of conformation S. The information function $I(S;R)$ is computed from a database of proteins used in the program (267 proteins for GOR IV).

The conformational state of a given residue in the sequence depends not only on the type of the amino acid R but also on the neighboring residues along the chain within the sliding window. GOR IV used a window of 17 residues, that is, for a given residue, eight nearest neighboring residues on each side were analyzed.

According to information theory, the information function of a complex event can be decomposed into the sum of information of simpler events, generally:

$$I\left(\Delta S; R_1, R_2, \ldots., R_n\right) = I\left(\Delta S; R_1\right) + I\left(\Delta S; R_2 \mid R_1\right), + \ldots + I(\Delta S; R_n \mid R_1, .., R_{n-1}) \tag{2}$$

where the information difference is defined as:

$$I\left(\Delta S; R_1, R_2, \ldots., R_n\right) = I\left(S; R_1, R_2, \ldots., R_n\right) - I\left(n - S; R_1, R_2, \ldots., R_N\right) \tag{3}$$

Here, $n - S$ denotes all conformations different than S. The GOR IV method assumed also that the information function is a sum of information from single residues (singlets) and pairs of residues (doublets) within the window of width $2d + 1$ (i.e., $d = 8$, for the window of 17 residues):

$$\log\frac{P\left(S_j,\mathrm{LocSeq}\right)}{P\left(n-S_j,\mathrm{LocSeq}\right)}=\frac{1-2d}{2d+1}\sum_{m=-d}^{d}\log\frac{P\left(S_j;R_{j+m}\right)}{P\left(n-S_j;R_{j+m}\right)}+\frac{2}{2d+1}\sum_{n,m=-d}^{d}\log\frac{P\left(S_j;,R_{j+m};,R_{j+n}\right)}{P\left(n-S_j;R_{j+m},R_{j+n}\right)} \quad (4)$$

Here the first summation is over singlets and the second summation is over doublets within the window centered around the *j*-th residue. The pair frequencies of residues R_j and $R_{j+}m$ with R_j occurring in conformations S_j and n-S_j are calculated from the database. All 267 proteins in the GOR IV database have well-determined structures (with crystallographic resolution at least 2.5 Å). Using the frequencies calculated from the databases, the program could predict probabilities of conformational states for a new sequence. The accuracy of the prediction with the GOR IV program based on single sequences (without multiple alignments) tested on the database of 267 sequences with the rigorous jack-knife methodology was 64.4%.

The advantage of the GOR method over other methods is that it clearly identifies all factors that are included in the analysis and calculates probabilities of all three conformational states. Because the GOR IV algorithm is computationally fast, it is possible to perform the full jack-knife procedure: each time when the prediction for the given sequence (of 267 sequences) is done, the sequence is removed from the database and the spectrum of frequencies used for the prediction is recalculated without including the information about the query sequence.

1.2 The GOR V Method

Several changes to the GOR IV program to improve the accuracy of the secondary structure prediction were made in GOR V. The GOR V version of the program is available from http://gor.bb.iastate.edu/

Modifications and improvements incorporated into the GOR V version are listed below:

1. Enlarged database of sequences with known secondary structure was used. The GOR IV database of 267 sequences was replaced by a new database of 513 nonredundant domains containing 84,107 residues proposed by Cuff and Barton [10, 11].
2. Some parameters in the GOR algorithm were optimized to increase the accuracy of the prediction. The most important modification was the introduction of the decision constants in the final prediction of the conformational state. The GOR IV program had a tendency to overpredict the coil state (C) at the cost of the helical conformation (H), and to an even greatest extent at the cost of beta-strands (E). Decision parameters were therefore introduced to improve predictions.

The predicted probability of the coil (C) conformation must be greater by some critical margins than probability of either the (H) or (E) states to accept C as the winning conformation. The margin for the beta-strands is greater than for helices. The introduction of the decision constants significantly improves the predicted results by about 1.6%.

3. The GOR algorithm was modified to include the triplet statistics within the window. The previous versions of the program used only single residue statistics (GOR I–II) or the combination of the single residue and pair residue statistics within the window (GOR III–IV). Now the GOR algorithm calculates statistics of singlets, pairs, and triplets for the secondary structure prediction. The addition of the triplets improved the accuracy of the prediction by only 0.3%.
4. A resizable window was applied in the GOR program. The previous version of the program (GOR IV) was using the window having a fixed width of 17 residues, that is, with eight residues on both sides of the central one. The accuracy of the prediction is slightly better for the smaller window of the width of 13 residues. The Cuff and Barton database on nonredundant sequences of protein domains includes a significant number of short sequences, with many of them as short as 20–30 residues. The prediction of the secondary structure for such short sequences is very inaccurate, because of the artificial end effect of the window. Residues at the beginning or at the end of the sequence have neighbors only on one side of the window. To overcome this problem smaller windows are used for the prediction of the secondary structure of short sequences. For sequences up to 25 residues, the window size is seven residues; for sequences from 26 to 50 residues, the window size is nine residues; for sequences 51–100 residues, the window is 11 residues; and for all sequences longer than 100 residues, the window size is 13. The introduction of the resizable window allows to include all 513 nonredundant sequences in the prediction procedure.
5. Multiple sequence alignments were used for the secondary structure prediction. Multiple sequence alignments from the PSI-BLAST [12] program for each of the 513 nonredundant sequences from the database were used. The nr database which contains all known databases: all nonredundant GeneBank CDS translations + PDB + SwissProt + PIR + PRF was used, with the maximum number of five iterations in the BLAST computations. The number of alignments varied considerably depending on the sequence. For some sequences, the BLAST program produced more than 2000 alignments, whereas for some other sequences, only a few alignments. A small improvement in the prediction is obtained by removing the alignments

that are too similar to the query sequence. The best results are obtained by skipping all alignments that have identity greater than 97% to the query sequence. Besides the identity threshold, various methods of weighting of the alignments in the calculation of the accuracy of the prediction were used. The methodological procedure was based on the calculation of the matrices of the probabilities of various (H, E, and C) secondary structure elements $P_H(i, j)$, $P_E(i, j)$, and $P_C(i, j)$ for each j-th residue in the i-th alignment (with the inclusion of alignment gaps). The averages over alignments $<P_H(j)>$, $<P_E(j)>$, and $<P_C(j)>$ at the j-th position in the alignment were computed and used for the prediction of the secondary structure conformation for the j-th residue. The simplest method is to use the largest probability value max $\{<P_H(j)>, <P_E(j)>, <P_C(j)>\}$. We have modified this assignment procedure by introducing decision constants. The coil state is assigned only if the calculated probability of the coil conformation is greater than the probability of the other states (H, E) plus the imposed thresholds (0.15 for E and 0.075 for H). The value of the threshold for the beta-sheets is larger than for alpha-helices, because strands were more often erroneously predicted as coils.

All calculations for the translation of the eight-state DSSP assignments into the three secondary structure states H, E, and C are the same as these used by Frishman and Argos. This means that DSSP states H and E were translated to H and E in a three-state code, and all other letters of the DSSP code were translated to coil (C). Additionally, similar to Frishman and Argos [13], we treated helices shorter than five residues (HHHH or less) and sheets shorter than three residues (EE or E) like coils, assuming that they are most likely prediction errors. The Frishman and Argos assignment scheme is therefore highly compatible with the GOR program performance.

1.3 GOR V Web Server

We have created the GOR V web server for protein secondary structure prediction [14]. The GOR V algorithm combines information theory, Bayesian statistics, and evolutionary information. In its fifth version, the GOR method reached (with the full jackknife procedure) an accuracy of prediction Q_3 of 73.5%.

The GOR V server is based on the database of Cuff and Barton [11] of 513 sequentially nonredundant domains, which contains 84,107 residues. To ensure that such a set was representative of available proteins, nonredundancy was defined with stringent tests. The address of the GOR V web server is http://gor.bb.iastate.edu.

The GORV server works in the following manner. When the user provides the input sequence, the GORV server that was trained on 513 proteins calculates the helix, sheet, and coil probabilities at each residue position and makes an initial prediction based on the structural states having highest probabilities. After this initial prediction, heuristic rules are applied. These rules

include converting helices shorter than five residues and sheets shorter than two residues to coil and using decision parameters.

1.4 Input Data

The required input data includes (Fig. 1):

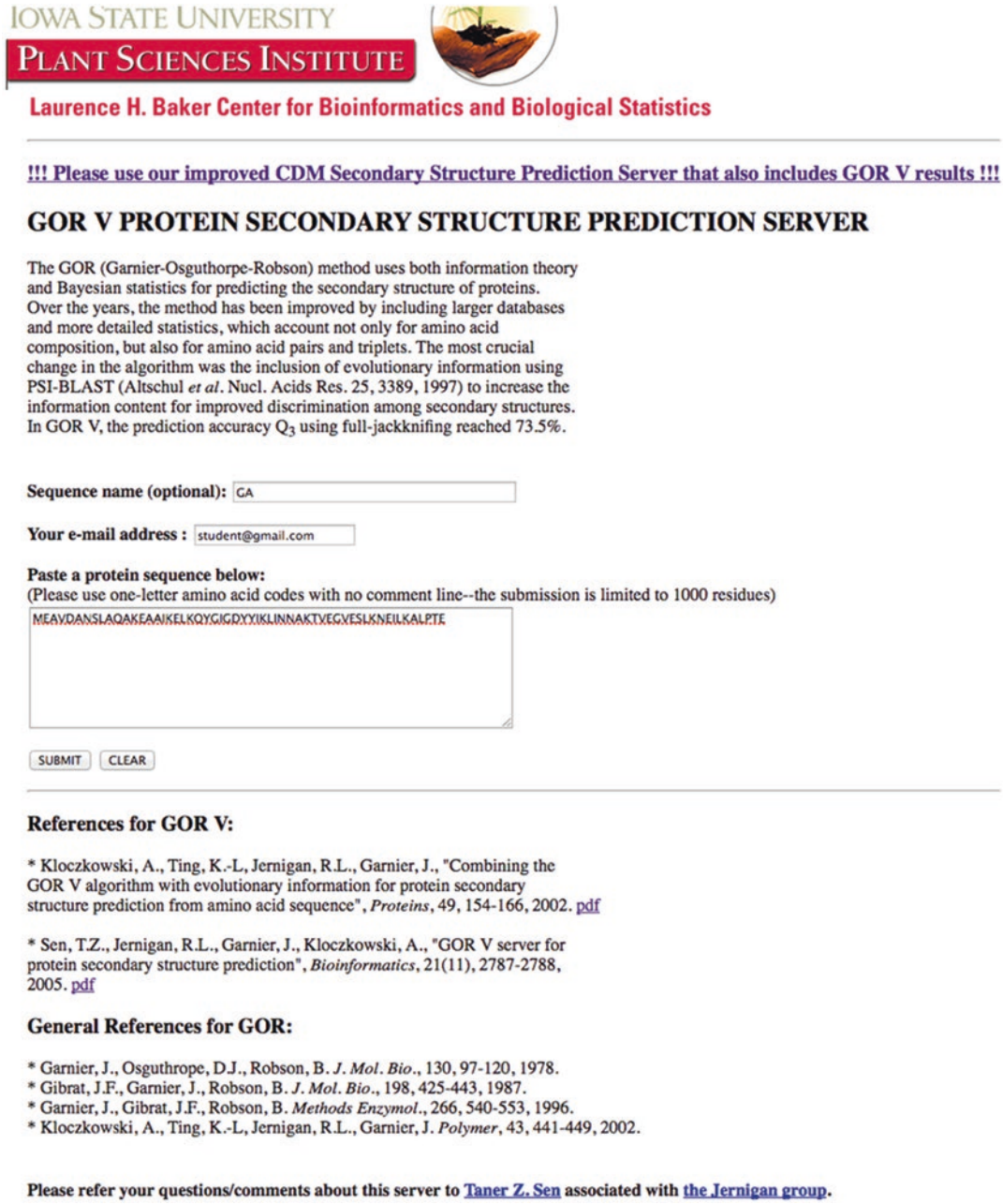
- Sequence name (optional). Name of the sequence or protein will appear in the Result page.
- User's e-mail address. An e-mail address to send the predicted information and notify of job completion.
- Protein sequence. The server accepts 20 single letter codes for standard amino acids, maximum 1000 amino acids in length.

For A domain of protein G (GA), the example protein shown in Fig. 1, the sequence can be accessed from the Protein Data Bank database (http://www.rcsb.org) or Uniprot database (http://www.uniprot.org/). The sequence of GA protein is deposited in PDB and Uniprot databases under identifiers 2FS1 and Q51918, respectively.

1.5 Output Data

As an output, the user receives the secondary structure prediction for the input sequence and the probabilities for each secondary state element at each position. The prediction results are shown in the web browser, which should stay open during the run, and are also sent to the e-mail address previously provided by the user. Any run-time error message will appear in the web browser, and if any problem arises, the user can contact the system administrator via the e-mail provided on the web page.

GOR V web server output data is shown in Fig. 2. The server provides the following: at the top of the output page a user might get either messages that the submission has been carried out successfully ("The e-mail address is accepted. The sequence is accepted. BLAST run is completed. GOR run is completed.") or error messages. If no submission errors have occurred, the secondary structure prediction is provided in a three-letter code: E-beta structure, H-helix, C-coil or loop. For the GA protein amino acid sequence, MEAV DANSLA[10]QAKEA AIKEL[20]KQYG IGDYYI[30]KLIN NAKTVE[40]GVESL KNEIL[50]KALPTE, we have obtained the following secondary structure prediction: CCCCHH HHHH[10]HHHHHH HHHH[20]HCCCCC CHHH[30]HHCCCCHHHH[40]HHHHHHHHHC[50]CCCCCC. Thus, GOR V web server predicted 3 alpha-helices, H1 [1, 5–18], H2 [25–29], and H3 [34–46] for GA protein which is in excellent agreement with the results of the DSSP algorithm [1] applied to the crystal structure (pdb code 2FS1). The structure predicted by DSSP is shown in Fig. 3. There are 3 alpha-helices: H1 [1, 5–20], H2 [24–31], H3 [36–48]. It is worth noting that more popular PSIPRED server fails to describe the entire range of complexity observed in GA folded structure. Namely, its prediction CHHH HHHHHH[10]HHHHH HHHHH[20]HHCCH HHHHH[30]HHHHHHHHHH[40]HHHH HHHHHH[50]HHCCCC suggests the presence of two helices, first


IOWA STATE UNIVERSITY
PLANT SCIENCES INSTITUTE

Laurence H. Baker Center for Bioinformatics and Biological Statistics

!!! Please use our improved CDM Secondary Structure Prediction Server that also includes GOR V results !!!

GOR V PROTEIN SECONDARY STRUCTURE PREDICTION SERVER

The GOR (Garnier-Osguthorpe-Robson) method uses both information theory and Bayesian statistics for predicting the secondary structure of proteins. Over the years, the method has been improved by including larger databases and more detailed statistics, which account not only for amino acid composition, but also for amino acid pairs and triplets. The most crucial change in the algorithm was the inclusion of evolutionary information using PSI-BLAST (Altschul *et al*. Nucl. Acids Res. 25, 3389, 1997) to increase the information content for improved discrimination among secondary structures. In GOR V, the prediction accuracy Q_3 using full-jackknifing reached 73.5%.

Sequence name (optional): GA

Your e-mail address : student@gmail.com

Paste a protein sequence below:
(Please use one-letter amino acid codes with no comment line--the submission is limited to 1000 residues)

MEAVDANSLAQAKEAAIKELKQYGIGDYYIKLINNAKTVEGVESLKNEILKALPTE

SUBMIT CLEAR

References for GOR V:

* Kloczkowski, A., Ting, K.-L, Jernigan, R.L., Garnier, J., "Combining the GOR V algorithm with evolutionary information for protein secondary structure prediction from amino acid sequence", *Proteins*, 49, 154-166, 2002. pdf

* Sen, T.Z., Jernigan, R.L., Garnier, J., Kloczkowski, A., "GOR V server for protein secondary structure prediction", *Bioinformatics*, 21(11), 2787-2788, 2005. pdf

General References for GOR:

* Garnier, J., Osguthrope, D.J., Robson, B. *J. Mol. Bio.*, 130, 97-120, 1978.
* Gibrat, J.F., Garnier, J., Robson, B. *J. Mol. Bio.*, 198, 425-443, 1987.
* Garnier, J., Gibrat, J.F., Robson, B. *Methods Enzymol.*, 266, 540-553, 1996.
* Kloczkowski, A., Ting, K.-L, Jernigan, R.L., Garnier, J. *Polymer*, 43, 441-449, 2002.

Please refer your questions/comments about this server to Taner Z. Sen associated with the Jernigan group.

Fig. 1 GOR V web server screenshot. Example input interface is presented for GA protein sequence

GOR is running, please wait...

GOR run is completed.

This is the secondary structure prediction:

CCCCHHHHHHHHHHHHHHHHHHHCCCCCCCHHHHHCCCCHHHHHHHHHHHHHHCCCCCCC

```
Column information:
1) Sequence index
2) Amino acid type
3) Helix probability
4) Sheet probability
5) Coil probability
6) GOR V prediction

1 M 0.063 0.126 0.811 C
2 E 0.069 0.147 0.784 C
3 A 0.126 0.210 0.664 C
4 V 0.253 0.280 0.467 C
5 D 0.410 0.291 0.299 H
6 A 0.635 0.200 0.165 H
7 N 0.625 0.186 0.189 H
8 S 0.681 0.163 0.156 H
9 L 0.715 0.150 0.136 H
10 A 0.724 0.137 0.139 H
11 Q 0.713 0.136 0.151 H
12 A 0.745 0.139 0.115 H
13 K 0.743 0.157 0.100 H
14 E 0.749 0.156 0.094 H
15 A 0.710 0.168 0.122 H
16 A 0.717 0.154 0.129 H
17 I 0.685 0.161 0.154 H
18 K 0.647 0.170 0.184 H
19 E 0.579 0.166 0.254 H
20 L 0.462 0.190 0.348 H
21 K 0.408 0.249 0.343 H
22 Q 0.265 0.263 0.472 C
23 Y 0.145 0.306 0.549 C
24 G 0.155 0.325 0.520 C
25 I 0.165 0.418 0.417 C
26 G 0.259 0.316 0.425 C
27 D 0.337 0.254 0.409 C
28 Y 0.444 0.289 0.267 H
29 Y 0.513 0.280 0.207 H
30 I 0.509 0.290 0.201 H
31 K 0.608 0.246 0.147 H
32 L 0.470 0.272 0.258 H
33 I 0.331 0.222 0.447 C
34 N 0.289 0.174 0.537 C
35 N 0.340 0.218 0.442 C
36 A 0.301 0.294 0.405 C
37 K 0.306 0.392 0.302 H
38 T 0.405 0.410 0.185 H
39 V 0.472 0.300 0.228 H
40 E 0.456 0.235 0.309 H
41 G 0.506 0.281 0.213 H
42 V 0.564 0.266 0.170 H
43 E 0.553 0.231 0.216 H
44 S 0.626 0.188 0.186 H
45 L 0.664 0.188 0.148 H
46 K 0.681 0.191 0.128 H
47 N 0.629 0.247 0.124 H
48 E 0.545 0.299 0.156 H
49 I 0.449 0.241 0.310 H
50 L 0.254 0.210 0.535 C
51 K 0.139 0.158 0.704 C
52 A 0.093 0.148 0.759 C
53 L 0.068 0.132 0.800 C
54 P 0.067 0.127 0.807 C
55 T 0.063 0.124 0.813 C
56 E 0.062 0.123 0.814 C
```

The prediction information is sent to your e-mail address. Thank you for using our service

Fig. 2 GOR V web server screenshot. Example output interface is presented for GA protein

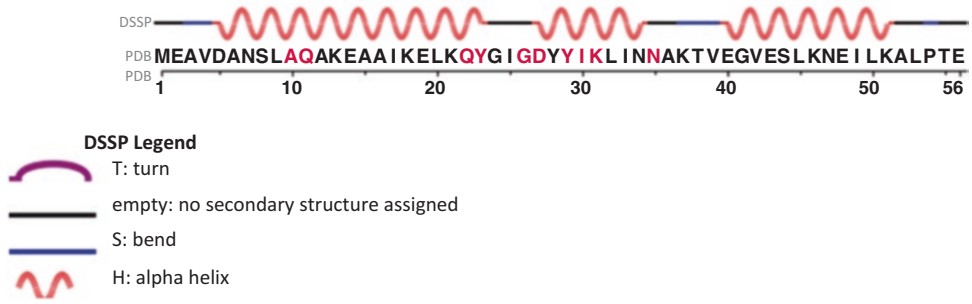


Fig. 3 Sequence chain view for GA protein (pdb code 2FSI) taken from Protein Data Bank (http://www.rcsb.org)

helix located from residue 2 to 22, H1 [1–19], and the second one located between residues 25 and 52, H2 [22–49]. Thus, for protein GA, the PSIPRED prediction is less accurate than the results obtained by the GOR server.

1.6 Hardware

Currently, the server is running Linux with 4.5GB RAM and 140GB memory. The program code is compiled using the Intel Fortran Compiler 8.0.034, and the web interface is established with a CGI script written using HTML and PERL. In order to use the GOR web server a user needs a personal computer workstation connected to the Internet. The GOR web server is compatible with most of popular web browsers like Google Chrome, Mozilla, or Safari.

1.7 Availability

The GOR web server is freely available at http://gor.bb.iastate.edu. Prediction time depends on the number of amino acids of the input sequence. For a sequence of ~100 amino acids, the secondary structure prediction takes ~30 s. The most time-consuming steps are PSI-BLAST alignments that in some cases, e.g., for many hits or slowly converging iterations, may take considerable time. Note, that the older version of the GOR server [6], GOR secondary structure prediction method version IV, is also available online at https://npsa-prabi.ibcp.fr/cgi-bin/npsa_automat.pl?page=npsa_gor4.html.

2 Methods

2.1 Secondary Structure Prediction with the GOR Model – Features and Applications

2.1.1 Combining Secondary Structure Prediction with Structure Prediction Servers to Reduce the Search Space

Secondary structure predictions of user submitted sequence obtained by GOR, PSIPRED, or similar algorithms might be used as additional input for structure prediction servers [15, 16] or as well as for the protein-peptide docking servers [17, 18]. Apart from protein sequence, those servers typically use some information about the predicted secondary structure of a protein or a peptide. For example, the CABS-fold web server provides tools for protein structure prediction not from sequence only (de novo modeling), but also using alternative templates [15]. If you get the secondary structure predicted for some residues of the sequence of interest, you can specify the secondary structure for structure

prediction web servers. By default, CABS-fold uses the PSIPRED method [19] prediction, but alternatively predictions by the GOR server can be used.

Although PSIPRED is probably the most commonly used secondary structure prediction method and, in general, it has been reported to be around 5% more accurate than the GOR method, we have shown above that the GOR-V secondary structure prediction for GA protein is in better agreement with DSSP. An interesting question is whether this observation is also valid for other proteins? To check this, we choose conceptually different protein to GA protein: the B domain of protein G (GB) that consists of one alpha-helix and four beta-strands. Moreover, if we compare predictions by PSIPRED and GOR methods for another well-studied protein, B domain of protein G (GB), we see slightly better performance of the GOR over PSIPRED method. Namely, for GB protein amino acid sequence, MTYKLILNGKTLKGETTTEAVDAATAEKVFKQYANDNGVD GEWTYDDATKTFTVTE, the following secondary structure predictions are obtained: CCCEEEECCC[10] CCCCCCCHHH[20] HHHHHHHHHH[30]HHHHCCCCCC[40]CEEEECCCCC[50]EEEE CC and CEEEE EEECE[10] ECCCCHH HHH[20]HCHHH HHHHH[30]HHHHHH CCCC[40]CCEEECCCCE[50]EEEEEC by using GOR and PSIPRED methods, respectively. Figure 4 shows the secondary structure for GB protein assigned by DSSP program. There are four beta-strands and one alpha-helix, S1 [2–8], S2 [1, 6, 11, 13–16], H1 [20–33], S3 [39–43], and S4 [49–52]. Both methods detect S1, S3, and S4 strands as well as alpha-helix observed in the native conformation of protein GB. It should be noted that both methods have shortcomings, e.g., H1 is predicted by GOR to start from the residue 18 and ends at the residue 34, while PSIPRED produces wrong prediction of beta-strand located from residue 9 to 10 and alpha-helix located between residues 16 and 21. As in protein the GA case, structure prediction for protein GB by the GOR-V server appears to be more accurate than by PSIPRED.

2.1.2 Modeling Chameleon Sequences of Proteins with the Help of GOR Method

An advantage of the GOR server is that it not only provides the secondary structure prediction for the input sequence but also offers the probabilities for each secondary state element at each position. This feature can be extremely useful for studying folding process of a protein by considering not only the single native state but also the complement structure(s), which might be observed with lower probabilities. A good example of such scenario is the mutants GA98 and GB98 with sequence identity of 98%, which were obtained by performing a set of mutation experiments starting from wild-type forms of proteins GA and GB [20, 21]. Note that the wild-type proteins GA and GB have no significant sequence homology and have different folds. The mutants GA98 and GB98

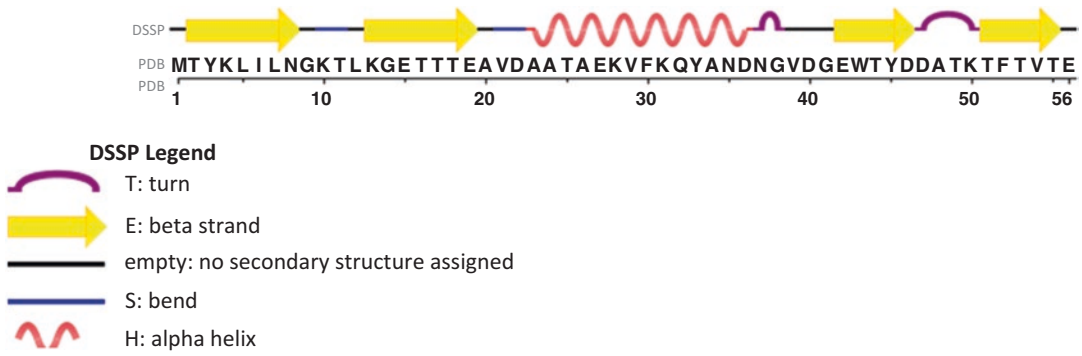


Fig. 4 Sequence chain view for the GB protein (pdb code 1PGB) taken from the Protein Data Bank (http://www.rcsb.org)

differ in one amino acid, but fold into different three-dimensional structures and perform different functions. GA98 shares the 3-alpha-helices fold of the parent protein GA, while GB98 shares the mixed alpha/beta fold of the parent protein GB. Interestingly, in the case of GA98, the competing structure (resembling the B domain of protein G instead of the A domain) is also observed experimentally with a certain, but low, probability [22, 23].

Figure 5 shows the probabilities to form alpha-helix and beta-strand as a function of residue position for GA (Fig. 5a), mutants GA30 and GA98 (Fig. 5b, c), GB (Fig. 5d), mutants GB30 and GB98 (Fig. 5e, f). In case of protein GB and its mutants, probabilities remain almost unchanged, while in case of protein GA and its mutants, we observe a switch between alpha fold of GA (and GA30) and alpha/beta fold of GA98. Experimentally in the case of GA98 protein, both alpha and alpha/beta folds were reported, whereas for GB98 only an alpha/beta fold of a parent GB protein [22].

Another example is the chameleon behavior of certain segments of protein sequence, which do not have a high preference for a particular conformation. The N-terminal fragment of the 49-residue protein CFr has been shown to fold into a helical structure, then unfold and finally refold into an extended beta-strand conformation [24, 25]. Incorporation of the GOR server predictions of both (alpha-helix and beta-strand) probabilities instead of only the most probable one might help to detect not only the native state, but also those observed with lower probabilities. We are now using a combination of the structure-based model [26, 27] with CABS software [28, 29] to test the effectiveness of this idea in ongoing simulations.

2.1.3 GOR Server as a Protein Aggregation Prediction Tool

The protein sequence determines its ability not only to fold but also to misfold and aggregate. Fibril formation resulting from protein misfolding and aggregation is a hallmark of several well-known neurodegenerative diseases such as Alzheimer's, type 2 diabetes, or Parkinson's diseases [30, 31]. The list of disorders linked to

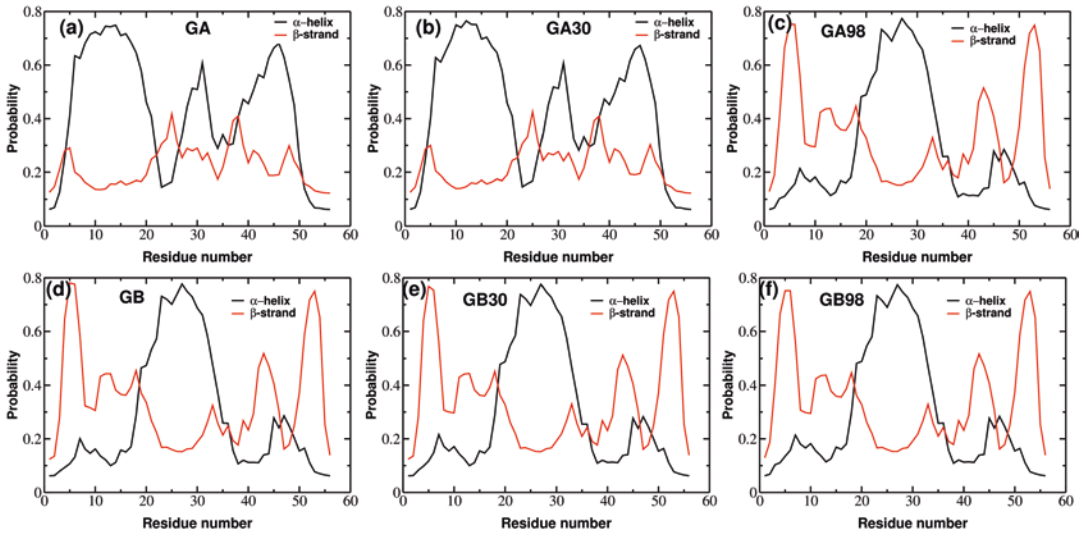


Fig. 5 Amino acid propensities for alpha-helices and beta-sheets as obtained by GOR method for protein GA (**a**) and GB (**d**) as well as their mutants GA30 (**b**), GA98 (**c**), GB30 (**e**), GB98 (**f**). In the case of protein GB and its mutants, probabilities remain almost unchanged (**d**–**f**), while in the case of protein GA and its mutants (**a**–**c**), we observe a switch from two alpha-helices (**a**, **b**) to three beta-strands (**c**)

protein misfolding continues to grow. For example, very recently the preeclampsia [32], a pregnancy-specific disorder, has been added to the list as it was shown to share pathophysiological features with recognized protein misfolding disorders. Although amyloid forming proteins and peptides exhibit no obvious sequence or structure homology, in many cases protein aggregates take the form of amyloid fibrils with high beta-sheet content. This suggests that the protein ability to misfold and aggregate can be described by general principles.

Recent theoretical and experimental results have indicated that protein aggregation rates depend on a number of factors such as the hydrophobicity of side chains [33, 34], preformed template fluctuations [35, 36], net charge [37], patterns of polar and non-polar residues [38], diverse secondary structure elements [39], high β-content [40], aromatic interactions [41], and the population of the fibril-prone conformation in the monomeric state [42, 43]. Many of those factors are used by bioinformatics predictive tools [44–48] to detect aggregation-prone fragments of proteins. Most of the predictive tools including TANGO [44], Aggrescan [45], Fold-amyloid [46], and Zyggregator [47] rely on a polypeptide chain sequence. For each sequence, those servers calculate own score and report aggregation-prone fragments of polypeptide sequence. Zyggregator predicts the aggregation-prone regions of polypeptide sequence based on a number of factors like hydrophobicity, charge, local stability, and the propensity to adopt alpha-helical or beta-sheet structures [47]. Aggrescan predictions are

based on an aggregation-propensity scale for natural amino acids derived from in vivo experiments and on the assumption that short and specific sequence stretches modulate protein aggregation [45].

In this work we test the GOR approach for the prediction of aggregation properties of protein structures. High beta-content in a monomeric state is one of the factors governing the fibril formation time [40]. Based on this, it is reasonable to assume that a high probability of amino acid to form beta-strand might correspond to its high aggregation propensity.

The accumulation of the beta-amyloid peptides found in two forms either 40 ($Abeta_{1\text{-}40}$) or 42 ($Abeta_{1\text{-}42}$) amino acids in human brains has been linked to Alzheimer's disease (AD) [31, 49]. Parkinson's disease, type 2 diabetes, and a disease known as dialysis-related amyloidosis are associated with the accumulation of amyloidogenic proteins: human alpha-synuclein, amylin, and beta-2 microglobulin respectively [50–52]. We choose these four proteins to investigate the effectiveness of the GOR method to predict aggregation-prone fragments of polypeptide sequence.

Figure 6 shows the propensity to form beta-strand as a function of residue index for typical amyloidogenic proteins, $Abeta_{1\text{-}40}$, human alpha-synuclein, amylin, and beta-2 microglobulin. The beta-sheet propensity profile of $Abeta_{1\text{-}40}$ has four peaks located at residue positions 4, 12, 17, and 32 (Fig. 6a). Note that two major peaks involving residues 17 and 32 correspond to the regions of high aggregation propensity determined experimentally (illustrated by red boundaries of squares). Namely, the central core of beta-amyloid involving residues 16–21 and the C-terminal fragment (residues 30–40).

The aggregation-prone interval from beta-sheet propensity profile can be identified as, $\Delta Ri = Ri_1 - Ri_2$, where Ri_1 and Ri_2 are the closest points to the half-maximum of the peak, Ri. We defined a fibril-prone conformation as one if beta-sheet probability exceeds the 40%. Using the above definitions we detected aggregation-prone regions for four typical amyloidogenic proteins, $abeta_{1\text{-}40}$, human alpha-synuclein, amylin, and beta-2 microglobulin (Table 1). Experimental results are also provided (Table 1 and Fig. 6) for comparison of GOR capability to predict aggregation-prone fragments of protein sequence.

In general, a peak in sheet propensity profiles can be interpreted as a sign of aggregation-prone fragment of protein sequence (except for the second peak for $Abeta_{1\text{-}40}$ and the first one for amylin). We believe that given the simplicity of the approach this agreement can be considered satisfactory and validates the use of the GOR method as an efficient alternative to other protein aggregation prediction tools.

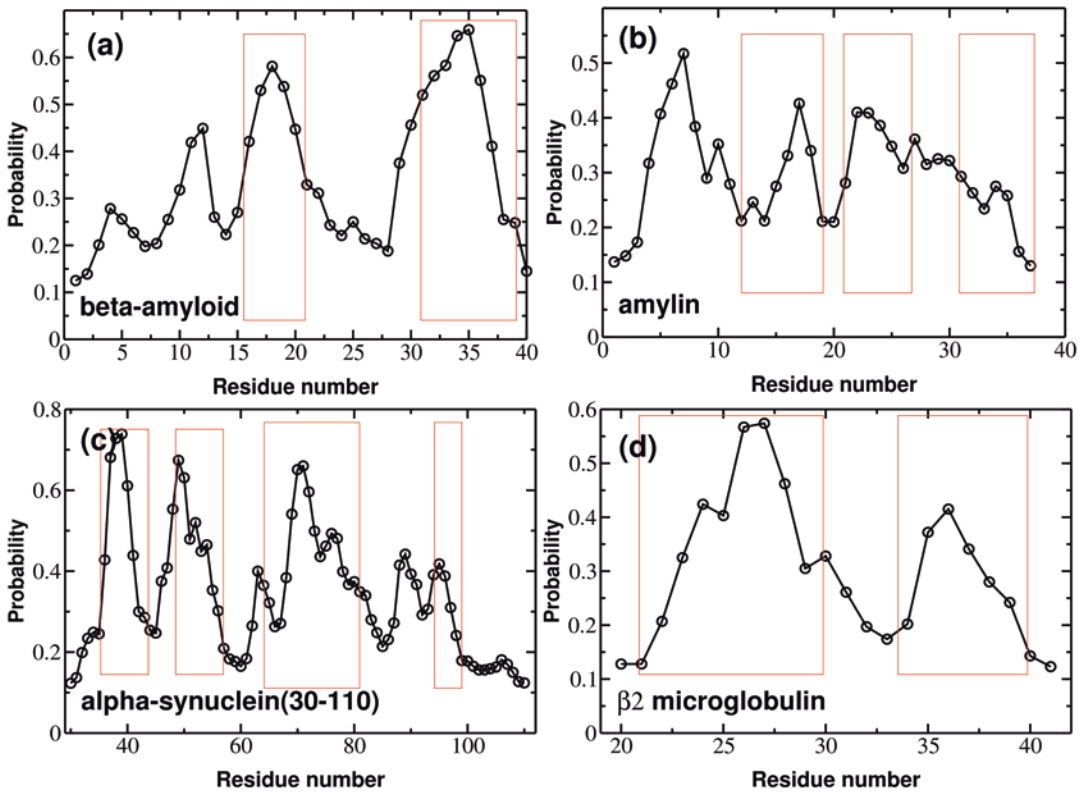


Fig. 6 Beta-sheet propensity profiles for $Abeta_{1\text{-}40}$ (**a**), amylin (**b**), alpha-synuclein (30–110) (**c**), beta-2 microglobulin [18–37] (**d**). *Red* boundaries of squares illustrate beta-strands of amyloid fibrils as determined by NMR experiment

Table 1
Comparison of GOR predictions with the regions adopting beta-strand configurations of different amyloid fibrils observed by NMR measurements

Protein	Experimentally known beta-strands of amyloid fibrils	GOR prediction
$Abeta_{1\text{-}40}$	– β_1: 16–21 β_2: 30–40	10–13 17–21 32–38
Alpha-synuclein (30–110)	β_1: 38–44 β_2: 49–58 β_3: 63–80 – β_4: 92–96	36–41 47–55 68–78 86–90 92–96
Amylin	– β_1: 12–17 β_2: 22–27 β_3: 31–37	4–8 14–17 21–31 –
Beta-2 microglobulin [17–38]	β_1: 21–30 β_2: 33–40	23–29 33–39

Acknowledgments

A. Kloczkowski would like to acknowledge support provided by start-up funds from The Research Institute of Nationwide Children's Hospital. This work was also supported by the Polish Ministry of Science and Higher Education Grant No. IP2012 016872 and "Mobilnosc Plus" No. DN/MOB/069/IV/2015; the National Science Center grant [MAESTRO 2014/14/A/ST6/00088].

References

1. Kabsch W, Sander C (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers 22:2577–2637
2. Frishman D, Argos P (1995) Knowledge-based protein secondary structure assignment. Proteins 23:566–579
3. Garnier J, Osguthorpe DJ, Robson B (1978) Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. J Mol Biol 120:97–120
4. Creighton TE (1990) Prediction of protein structure and the principles of protein conformation. Gerald D. Fasman, Ed. Plenum, New York, 1989. xiv, 798 pp., illus. $95, Science 247:1351–1352
5. Gibrat JF, Garnier J, Robson B (1987) Further developments of protein secondary structure prediction using information theory: new parameters and consideration of residue pairs. J Mol Biol 198:425–443
6. Garnier J, Gibrat JF, Robson B (1996) GOR method for predicting protein secondary structure from amino acid sequence. Meth Enzymol 266:540–553
7. Kloczkowski A, Ting KL, Jernigan RL, Garnier J (2002) Combining the GOR V algorithm with evolutionary information for protein secondary structure prediction from amino acid sequence. Proteins 49:154–166
8. Rost B, Sander C, Schneider R (1994) Phd—an automatic mail server for protein secondary structure prediction. Comput Appl Biosci 10:53–60
9. Faraggi E, Zhang T, Yang YD, Kurgan L, Zhou YQ (2012) SPINE X: improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles. J Comput Chem 33:259–267
10. Cuff JA, Barton GJ (1999) Evaluation and improvement of multiple sequence methods for protein secondary structure prediction. Proteins 34:508–519
11. Cuff JA, Barton GJ (2000) Application of multiple sequence alignment profiles to improve protein secondary structure prediction. Proteins 40:502–511
12. Altschul SF, Madden TL, Schaffer AA, Zhang JH, Zhang Z, Miller W, Lipman DJ (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25:3389–3402
13. Frishman D, Argos P (1997) Seventy-five percent accuracy in protein secondary structure prediction. Proteins 27:329–335
14. Sen TZ, Jernigan RL, Garnier J, Kloczkowski A (2005) GOR V server for protein secondary structure prediction. Bioinformatics 21:2787–2788
15. Blaszczyk M, Jamroz M, Kmiecik S, Kolinski A (2013) CABS-fold: server for the de novo and consensus-based prediction of protein structure. Nucleic Acids Res 41:W406–W411
16. Yang JY, Yan RX, Roy A, Xu D, Poisson J, Zhang Y (2015) The I-TASSER suite: protein structure and function prediction. Nat Methods 12:7–8
17. Kurcinski M, Jamroz M, Blaszczyk M, Kolinski A, Kmiecik S (2015) CABS-dock web server for the flexible docking of peptides to proteins without prior knowledge of the binding site. Nucleic Acids Res 43:W419–W424
18. Blaszczyk M, Kurcinski M, Kouza M, Wieteska L, Debinski A, Kolinski A, Kmiecik S (2016) Modeling of protein-peptide interactions using the CABS-dock web server for binding site search and flexible docking. Methods 93:72–83
19. Jones DT (1999) Protein secondary structure prediction based on position-specific scoring matrices. J Mol Biol 292:195–202
20. Alexander PA, He YA, Chen YH, Orban J, Bryan PN (2009) A minimal sequence code for switching protein structure and function. Proc Natl Acad Sci U S A 106:21149–21154
21. Alexander PA, He Y, Chen Y, Orban J, Bryan PN (2007) The design and characterization of

two proteins with 88% sequence identity but different structure and function. Proc Natl Acad Sci U S A 104:11963–11968

22. Bryan PN, Orban J (2010) Proteins that switch folds. Curr Opin Struct Biol 20:482–488
23. Kouza M, Hansmann UHE (2012) Folding simulations of the A and B domains of protein G. J Phys Chem B 116:6645–6653
24. Mohanty S, Meinke JH, Zimmermann O, Hansmann UHE (2008) Simulation of Top7-CFr: a transient helix extension guides folding. Proc Natl Acad Sci U S A 105:8004–8007
25. Gaye ML, Hardwick C, Kouza M, Hansmann UHE (2012) Chameleonicity and folding of the C-fragment of TOP7. Epl-Europhys Lett 97:68003
26. Whitford PC, Noel JK, Gosavi S, Schug A, Sanbonmatsu KY, Onuchic JN (2009) An all-atom structure-based potential for proteins: Bridging minimal models with all-atom empirical forcefields. Proteins 75:430–441
27. Clementi C, Nymeyer H, Onuchic JN (2000) Topological and energetic factors: what determines the structural details of the transition state ensemble and "en-route" intermediates for protein folding? An investigation for small globular proteins. J Mol Biol 298:937–953
28. Kolinski A (2004) Protein modeling and structure prediction with a reduced representation. Acta Biochim Pol 51:349–371
29. Wabik J, Kmiecik S, Gront D, Kouza M, Kolinski A (2013) Combining coarse-grained protein models with replica-exchange all-atom molecular dynamics. Int J Mol Sci 14:9893–9905
30. Chiti F, Dobson CM (2006) Protein misfolding, functional amyloid, and human disease. Annu Rev Biochem 75:333–366
31. Nasica-Labouze J, Nguyen PH, Sterpone F, Berthoumieu O, Buchete NV, Cote S, De Simone A, Doig AJ, Faller P, Garcia A, Laio A, Li MS, Melchionna S, Mousseau N, Mu YG, Paravastu A, Pasquali S, Rosenman DJ, Strodel B, Tarus B, Viles JH, Zhang T, Wang CY, Derreumaux P (2015) Amyloid beta protein and Alzheimer's disease: when computer simulations complement experimental studies. Chem Rev 115:3518–3563
32. Buhimschi IA, Nayeri UA, Zhao G, Shook LL, Pensalfini A, Funai EF, Bernstein IM, Glabe CG, Buhimschi CS (2014) Protein misfolding, congophilia, oligomerization, and defective amyloid processing in preeclampsia. Sci Transl Med 6:245ra292
33. Berhanu WM, Hansmann UHE (2012) Side-chain hydrophobicity and the stability of A beta(16-22) aggregates. Protein Sci 21:1837–1848
34. Otzen DE, Kristensen O, Oliveberg M (2000) Designed protein tetramer zipped together with a hydrophobic Alzheimer homology: a structural clue to amyloid assembly. Proc Natl Acad Sci U S A 97:9907–9912
35. Nguyen PH, Li MS, Stock G, Straub JE, Thirumalai D (2007) Monomer adds to preformed structured oligomers of A beta-peptides by a two-stage dock-lock mechanism. Proc Natl Acad Sci U S A 104:111–116
36. Kouza M, Co NT, Nguyen PH, Kolinski A, Li MS (2015) Preformed template fluctuations promote fibril formation: insights from lattice and all-atom models. J Chem Phys 142:145104
37. Chiti F, Calamai M, Taddei N, Stefani M, Ramponi G, Dobson CM (2002) Studies of the aggregation of mutant proteins in vitro provide insights into the genetics of amyloid diseases. Proc Natl Acad Sci U S A 99:16419–16426
38. West MW, Wang WX, Patterson J, Mancias JD, Beasley JR, Hecht MH (1999) De novo amyloid proteins from designed combinatorial libraries. Proc Natl Acad Sci U S A 96:11211–11216
39. Kallberg Y, Gustafsson M, Persson B, Thyberg J, Johansson J (2001) Prediction of amyloid fibril-forming proteins. J Biol Chem 276:12945–12950
40. Sgourakis NG, Yan YL, McCallum SA, Wang CY, Garcia AE (2007) The Alzheimer's peptides A beta 40 and 42 adopt distinct conformations in water: a combined MD/NMR study. J Mol Biol 368:1448–1457
41. Gazit E (2002) A possible role for pi-stacking in the self-assembly of amyloid fibrils. FASEB J 16:77–83
42. Chiti F, Stefani M, Taddei N, Ramponi G, Dobson CM (2003) Rationalization of the effects of mutations on peptide and protein aggregation rates. Nature 424:805–808
43. Nam HB, Kouza M, Hoang Z, Li MS (2010) Relationship between population of the fibril-prone conformation in the monomeric state and oligomer formation times of peptides: insights from all-atom simulations. J Chem Phys 132:165104
44. Fernandez-Escamilla AM, Rousseau F, Schymkowitz J, Serrano L (2004) Prediction of sequence-dependent and mutational effects on the aggregation of peptides and proteins. Nat Biotechnol 22:1302–1306
45. Castillo V, Grana-Montes R, Sabate R, Ventura S (2011) Prediction of the aggregation propensity of proteins from the primary sequence: aggregation properties of proteomes. Biotechnol J 6:674–685

46. Garbuzynskiy SO, Lobanov MY, Galzitskaya OV (2010) FoldAmyloid: a method of prediction of amyloidogenic regions from protein sequence. Bioinformatics 26:326–332
47. Tartaglia GG, Vendruscolo M (2008) The Zyggregator method for predicting protein aggregation propensities. Chem Soc Rev 37:1395–1401
48. Zambrano R, Jamroz M, Szczasiuk A, Pujols J, Kmiecik S, Ventura S (2015) AGGRESCAN3D (A3D): server for prediction of aggregation properties of protein structures. Nucleic Acids Res 43:W306–W313
49. Selkoe DJ (2004) Cell biology of protein misfolding: the examples of Alzheimer's and Parkinson's diseases. Nat Cell Biol 6:1054–1061
50. Polymeropoulos MH, Lavedan C, Leroy E, Ide SE, Dehejia A, Dutra A, Pike B, Root H, Rubenstein J, Boyer R, Stenroos ES, Chandrasekharappa S, Athanassiadou A, Papapetropoulos T, Johnson WG, Lazzarini AM, Duvoisin RC, DiIorio G, Golbe LI, Nussbaum RL (1997) Mutation in the alpha-synuclein gene identified in families with Parkinson's disease. Science 276:2045–2047
51. Kodali R, Wetzel R (2007) Polymorphism in the intermediates and products of amyloid assembly. Curr Opin Struct Biol 17:48–57
52. Floege J, Ketteler M (2001) beta(2)-microglobulin-derived amyloidosis: an update. Kidney Int 59:S164–S171

Chapter 3

Consensus Prediction of Charged Single Alpha-Helices with CSAHserver

Dániel Dudola, Gábor Tóth, László Nyitray, and Zoltán Gáspári

Abstract

Charged single alpha-helices (CSAHs) constitute a rare structural motif. CSAH is characterized by a high density of regularly alternating residues with positively and negatively charged side chains. Such segments exhibit unique structural properties; however, there are only a handful of proteins where its existence is experimentally verified. Therefore, establishing a pipeline that is capable of predicting the presence of CSAH segments with a low false positive rate is of considerable importance. Here we describe a consensus-based approach that relies on two conceptually different CSAH detection methods and a final filter based on the estimated helix-forming capabilities of the segments. This pipeline was shown to be capable of identifying previously uncharacterized CSAH segments that could be verified experimentally. The method is available as a web server at http://csahserver.itk.ppke.hu and also a downloadable standalone program suitable to scan larger sequence collections.

Key words Charged single alpha-helix, Charged residues, Consensus prediction, Ion pairs, Fourier transform, Helicity, Coiled coil

1 Introduction

The charged single alpha-helix (CSAH) is a relatively recently identified protein structural motif. Its main feature is its stability as a single helix in solution. CSAH segments are characterized by a high number of regularly alternating charged residues, mainly Glu, Arg, and Lys [1–3]. There are only a handful of such segments that are characterized experimentally; thus, their identification by prediction methods is of high importance. CSAH segments provide flexible extensions to proteins and are acting as lever arms in some myosins [4], spacers in caldesmon [5], and inner centromere protein [6] and presumably rulers in paraspeckle assembly [7]. CSAHs represent a rare structural motif; about 0.2% of human proteins were predicted to contain such a segment [8].

The basis of the stability of CSAH segments is not yet fully understood, although it is believed to be primarily of electrostatic

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_3,

nature. A recent detailed study highlighted that $K_i \rightarrow E_{i+4}$ pairs are favored over $E_i \rightarrow K_{i+4}$ salt bridges in synthetic single helical segments containing Glu and Lys residues [9].

CSAHs have been shown to be located in segments predicted to form coiled coils and/or disordered regions [1, 2, 6, 8]. The latter can be rationalized by the abundance of charged residues in CSAH segments and the former with their periodicity. CSAHs can be regarded as special, "single-stranded coiled-coils" and are in a possible evolutionary relationship with them [10, 11].

In characterizing structural features of protein sequences, it is important to obtain the best prediction as it will offer the most specific information about the protein. In our view, more specific predictions should have precedence over general ones. In our special case it is important to emphasize that coiled coil segments are often predicted as disordered ones and CSAHs as coiled coils and/or disordered ones. Thus, we suggest CSAH > coiled coil > disorder precedence when evaluating the results of structural predictions for a given sequence. In other words, a segment predicted to form a CSAH and also predicted to be disordered is most likely a CSAH, as the latter feature was predicted by a much more specialized method.

There are currently three described CSAH prediction methods: SCAN4CSAH and FT_CHARGE are integrated to a consensus method as described in this chapter, whereas Waggawagga offers an integrated analysis with coiled coil prediction [12]. As the number of experimentally verified CSAH segments is still very low, it is not yet possible to give a comprehensive analysis of the performance of these methods. Therefore, when designing CSAHserver, our primary aim was to yield a conservative estimate of potential CSAH segments and minimize the number of false positive hits.

2 Materials

Our consensus CSAH detection method is available as a web service at http://csahserver.itk.ppke.hu. The standalone version, offering more options for parametrization and with the capability of processing multiple sequences, can be downloaded from http://csahserver.itk.ppke.hu/csah/download/csahdetect.zip. The package contains three Perl executables and two parameter files along with an installer script INSTALL.PL and a README.TXT file, as well as sample input/output files. The program is designed to run under Linux but is expected to work on all architectures where Perl 5 can be installed. The only other external requirement is the Math:FFT Perl module, available from the CPAN archive (www.cpan.org). Normally the installer script will check the presence of the module and will guide the user through its installation. To install the programs, unpack the zip archive in a suitable directory

and invoke the installer with "perl INSTALL.PL" and follow the instructions (*see* also **Note 1**).

The installed package contains three Perl scripts:

1. A wrapper script csahdetect.pl which is designed to provide a single interface for the two CSAH prediction algorithms. This script handles the input FASTA file which might contain multiple protein sequences, passes them to scan4csah.pl and ft_charge.pl, reads their output, prepares their consensus, and filters out potential non-helical false positive hits.
2. The program scan4csah.pl predicts CSAHs using an optimized scoring scheme based on the presence of potential stabilizing and destabilizing interactions between charged residues in the sequence. It uses a file with precomputed extreme value distribution (EVD) parameters (scan4csah_evdtable.txt). This program can also be used without csahdetect.pl although invoking it through csahdetect.pl will yield a more readable simplified output.
3. The program ft_charge.pl implements a CSAH prediction method based on the regular periodicity in the pattern of charged residues. Similarly to scan4csah.pl, it uses a file with precomputed EVD parameters and can be invoked directly, although invoking it through csahdetect.pl will yield a more readable simplified output.

3 Methods

3.1 Input File

The input sequence can be in FASTA format or just a plain sequence. FASTA format sequences with headers conforming to the UniProt convention (e.g., >sp|Q9NQS7|INCE_HUMAN [free text]) are preferred. To apply the methods on a large number of protein sequences, e.g., a full proteome, we advise the use of the standalone version which is capable of handling FASTA files with multiple sequences.

3.2 The SCAN4CSAH Method

The SCAN4CSAH method is based on a scoring scheme developed to discriminate between stabilizing and destabilizing patterns of charged residues as listed in Table 1. The segment should contain a continuous run of charged residues with an uncharged segment of maximum 5 residues long as default. The scores are normalized with respect to the length of the segment analyzed.

The scoring scheme of SCAN4CSAH was optimized by an exhaustive grid search of the parameter space to yield high scores for known (or highly likely candidates of) CSAH segments at the time of its development (caldesmon, myosin IV and myosin X proteins). The scores were to fit to an extreme value distribution (EVD) curve to turn them into *P*-values [2]. The most important

Table 1

Stabilizing and destabilizing interactions considered in SCAN4CSAH

Description	Pattern(s)	Example(s)
Stabilizing patterns/modifiers		
Salt bridges in a helical turn	$i-i+3$, $i-i+4$	KxxE, RxxxE
Cooperative effect between residues with alternating charges	$i-i+4-i+8$, $i-i+4-i+7$, $i-i+3-i+7$ etc	RxxxExxxK, ExxKxxxE etc
Ion pairs with the acidic residue in the N-terminal position		ExxxK
Destabilizing patterns/modifiers		
Close opposite charges	$i-i+1$, $i-i+2$	EK, ExK
Repulsing interactions in a helical turn	$i-i+3$, $i-i+4$	ExxE, RxxxK

adjustable parameter of the method is the minimum length of the predicted CSAH segments, default (from version 2.0) is 30 residues (*see* **Note 2**).

3.3 The FT_CHARGE Method

The FT_CHARGE method relies on the identification of repeating charge patterns in protein sequences. In its default mode it first calculates the charge correlation function for a given sequence window as:

$$R(n) = \sum_{i=1}^{m-n} c(i)c(i+n)$$

where $c(i)$ is the charge of the residue at the ith position, m is the length of the segment examined, and n is the sequential distance between residues.

As a next step, Fourier transformation is applied to $R(n)$. The position of the maximum of the obtained frequency spectrum should be between 1/6 and 1/9 corresponding to approximately 2 turns in an alpha-helix [2]. The maximum value is converted to a P-score obtained from an EVD distribution fitted to maximum values (regardless of frequency) calculated from sets of 5000 sequences of different lengths (16, 32, 64, 128 residues) and amino acid compositions (composed of Ala, Glu, Lys residues with Glu and Lys contents varied at 10% steps). The main adjustable parameter is the segment length for FFT analysis, default is 32–64, meaning that all possible 32-residue and 64-residue segments will be analyzed (*see* **Note 3**). While this methodology is time-consuming, it ensures that the precise boundaries of putative CSAH segments can be found. The effect of the different window lengths on the predictions was analyzed and we found

that there are segments with relatively low charge density but regularly alternating charge pattern that can only be detected using longer (e.g., 64-residue) window. In contrast, there are also regions that exhibit the regular pattern characteristic for CSAHs only over a shorter stretch of residues that is only effectively recognized using the shorter window length [8].

3.4 Getting the Consensus and Omitting Sequences with Low Helicity

It should be noted that the two conceptually different methods predict a different number of CSAHs with FT_CHARGE being much more restrictive, recognizing less than 10% of the CSAHs predicted by SCAN4CSAH [8]. In an analysis of SwissProt 54.2, we found that previously uncharacterized predicted CSAH segments that obtained a high score by both methods indeed form single alpha-helices in solution as verified by CD spectroscopy [2]. However, as the number of experimentally verified CSAH segments is still very low and a comprehensive analysis of prediction methods is still not feasible, we aimed at a highly conservative approach to minimize the number of false positive hits. This is achieved by taking the consensus of the two methods and introducing a filter that rejects sequences that are not expected to be helical, e.g., segments with high proline content. This is important as neither of the methods checks for the amino acid composition of the segments, which was, on the other hand, an important design feature at the development of the methods to avoid any artificially introduced biases in charged amino acid types.

The filter is based on the Chou-Fasman helicity scale Pα [13] calculated for helices longer than 15 residues as assigned by DSSP (available from ftp://ftp.cmbi.ru.nl/pub/molbio/data/dssp/) in protein chains listed in PDB Select [14] (2012 October release, 25% list, sigma 3.0). Per-residue average Pα was calculated for these helices and the scores were converted to *P*-values by fitting an EVD curve. After inspection of CSAHs predicted in SwissProt (release 201508), the *P*-value threshold was set to 0.5, i.e., sequences with higher *P*-values are discarded as probable non-helical ones. We note that we deliberately refrained from using the currently best-performing secondary structure prediction methods as these are based on the identification of homologous sequences and CSAHs are of low complexity and probably also fast evolving, rendering such methods inappropriate.

3.5 Using the CSAHserver Web Service

The usage of the CSAHserver website is straightforward. Go to csahserver.itk.ppke.hu and either copy-paste your sequence into the sequence input field or upload a FASTA format file (Fig. 1a). If you use the default parameter settings as recommended, simply click on the "Predict CSAHs" button and you will obtain the results typically in a few seconds. Running with default parameters will yield a consensus-based conservative estimate of CSAH occurrences. For more detailed analysis of the input sequence,

Fig. 1 (**a**) Input form of CSAHserver 2.0 and (**b**) sample output for the inner centromere protein INCE_HUMAN (UniProt accession Q9NQS7)

the user can select one of the methods, SCAN4CSAH or FT_CHARGE and their parameters such as minimum length and window size (*see* **Notes 2** and **3**). The web server automatically sets the minimum length for the consensus CSAH segments to the lower of the SCAN4CSAH minimum length and FT_CHARGE lowest window length.

As output, the server displays the input sequence (in lowercase letters) with the CSAH regions highlighted (CSAH segment in bold capital letters with positively charged residues, Arg and Lys colored in blue, negatively charged ones, Glu and Asp colored red) and a tabular summary of the identified CSAHs (Fig. 1b). By default, consensus regions are shown that are detected both by SCAN4CSAH and FT_CHARGE, along with the corresponding segments predicted by these two methods (*see* **Note 4**).

In addition, a csv file containing the tabular output and two FASTA format files are available for download: one containing the input sequence with the CSAH regions masked (by "x" characters) and another with the sequence(s) of the identified CSAH region(s).

3.6 CSAH Prediction with the Locally Installed Standalone Version

The standalone version can be used for a FASTA formatted file with one or multiple protein sequences and offers more parametrization than the web service. Because ft_charge.pl runs much slower than scan4csah.pl, in the consensus mode scan4csah.pl is invoked first and ft_charge.pl is run only on sequences where scan4csah.pl predicted the presence of CSAHs (*see* **Note 5**). The script csahdetect.pl is also capable of invoking only one of the prediction programs or reading their precomputed output files (*see* **Note 6** for further options).

Normally, the script can be invoked as

csahdetect.pl --infasta = input.fst

to run with default parameters (consensus mode, minimum consensus CSAH length of 30 residues, nonhelicity filtering above $P = 0.5$).

The default output is a tab-separated CSAH table with each detected CSAH displayed in a separate line:

#Accession	Consensus	scan4csah	ft_charge
Q9NQS7	543–574	536–600	543–574
Q9NQS7	654–685	654–782	652–685

In addition, the two FASTA files, with the masked full and CSAH sequences, can be generated using the --maskedfasta and --csahfasta options. The masked fasta file contains only those sequences for which CSAHs were predicted.

4 Notes

1. After unpacking the zip archive, type "perl INSTALL.PL" from that directory. The install script will check the availability of Math::FFT and offer to install it using cpan (invoking "sudo cpan Math::FFT"); the user can skip this but will have to install Math::FFT manually later to use the FT_CHARGE method. The script will also ask where to put the scan4csah.pl and ft_charge.pl executables and the EVD parameter files, as well as the csahdetect.pl executable.
2. In the most recent version the default minimum CSAH length for SCAN4CSAH is set to 30 residues according to our analysis of paraspeckle proteins [7] where this setting yielded the most consistent results. Generally, we do not advise to use shorter lengths in order to avoid the possible increase of false positive hits.
3. Other possible options include a 16 residue-long window and its combination with longer windows. The use of multiple window sizes is based on our experience that some CSAHs might be missed with longer and others with shorter windows. The reason behind is that the charge periodicity is tested for the full window length and some sequences might not be regular enough over a longer span and others might have relatively low charge density (for a CSAH) and regularity is only evident in a longer segment.
4. In the consensus mode only those segments are shown that were predicted by both of the methods. Thus, regions detected by either SCAN4CSAH or FT_CHARGE only will not appear in the consensus-mode output in any form. To obtain this information, the web service can be run with only one of the methods chosen.
5. In practice, a filtered version of the input FASTA file is generated based on the SCAN4CSAH output and this is used as input for FT_CHARGE. While this reduces computing time (an ft_charge.pl run on the full SwissProt release 201508 containing 594008 sequences took ~4 days on a single Intel Core i5 2.4 GHz processor), it requires more disk space (comparable to the size of the original input FASTA file) which can be substantial when analyzing large proteomes or even full SwissProt. Thus, analysis of a large sequence databases with the present version requires extra care and some tricks like running scan4csah.pl first, generating the filtered file manually, splitting the filtered file and invoking ft_charge on them separately, then merging the outputs. Please also note that in such a case the unprocessed output files of SCAN4CSAH and FT_CHARGE can also be relatively large.

6. The most important additional option is --minconslen, with which the minimum consensus length can be set. In contrast to the web server, where this is set automatically, the standalone version uses 30 residues as default minimum consensus CSAH length and this can only be changed by explicitly setting the --minconslen option to a different value. Another important option is --helicalP; this can be used for nonhelicity filtering. The default value is 0.5; to turn this filtering off, this can be set to 1. Note that this is a "negative filter" to avoid false positives; thus, *P*-values lower than 0.5 might result in too stringent rejection of possible CSAH candidates. Example of a predicted CSAH segment with *P*-value 1.0, unlikely to form a helical structure based on its amino acid composition: >Q9LR47 254–316 RPSDYSRRPSDYSRRPSDYSRRPSDYSRRPSDSRPSDYSR PSDYYSRPSDYSRPSDFSRS SDD The csahdetect.pl script allows the parametrization of the invoked methods using the --scan4csah and --ft_charge options by providing a full (parametrized) command to be invoked for these methods. This is meant for expert usage. For example, it is possible for FT_CHARGE to set the step size between sequence windows. This results in faster scan at the expense of lower precision in CSAH boundaries. The default step size is 1; thus, each possible window of a given length is analyzed for each sequence.

   In addition, by setting the -f option for FT_CHARGE, it can be used to detect regularly alternating charge patterns in nonhelical sequence motifs as well.

   Detailed usage and a full list of options can be obtained by invoking scan4csah.pl --help and ft_charge.pl -h.

Acknowledgments

This work was supported by a grant from the Hungarian Scientific Research Fund (OTKA/NKFIH NF104198). Z.G. was also supported by a János Bolyai Research Fellowship from the Hungarian Academy of Sciences.

References

1. Knight PJ, Thirumurugan K, Xu Y, Wang F, Kalverda AP, Stafford WF III, Sellers JR, Peckham M (2005) The predicted coiled-coil domain of myosin 10 forms a novel elongated domain that lengthens the head. J Biol Chem 280:34702–34708
2. Süveges D, Gáspári Z, Tóth G, Nyitray L (2009) Charged single alpha-helix: a versatile protein structural motif. Proteins 74:905–916
3. Swanson CJ, Sivaramakrishnan S (2014) Harnessing the unique structural properties of isolated alpha-helices. J Biol Chem 289:25460–25467
4. Spudich SA, Sivaramakrishnan S (2010) Myosin VI: an innovative motor that challenged the swinging lever arm hypothesis. Nat Rev Mol Cell Biol 11:128–137

5. Huber PA (1997) Caldesmon. Int J Biochem Cell Biol 29:1047–1051
6. Samejima K, Platani M, Wolny M, Ogawa H, Vargiu G, Knight PJ, Peckham M, Earnshaw WC (2015) The inner centromere protein (INCENP) coil is a single -helix (SAH) domain that binds directly to microtubules and is important from chromosome passenger complex (CPC) localization and function in mitosis. J Biol Chem 290:21460–21472
7. Dobson L, Nyitray L, Gáspári Z (2015) A conserved charged single α-helix with a putative steric role in paraspeckle formation. RNA. doi:10.1261/rna.053058.115
8. Gáspári Z, Süveges D, Perczel A, Nyitray L, Tóth G (2012) Charged single alpha-helices in proteomes revealed by a consensus prediction approach. Biochim Biophys Acta 1824:637–646
9. Baker EG, Bartlett GJ, Crump MP, Sessions RB, Linden N, Faul CFJ, Woolfson DN (2015) Local and macroscopic electrostatic interactions in single α-helices. Nat Chem Biol 11:221–228
10. Szappanos B, Süveges D, Nyitray L, Perczel A, Gáspári Z (2010) Folded-unfolded cross-predictions and protein evolution: the case study of coiled-coils. FEBS Lett 584:1623–1627
11. Gáspári Z, Nyitray L (2011) Coiled coils as possible models for protein structure evolution. Biomol Concepts 2:199–210
12. Simm D, Hatje K, Kollmar M (2015) Waggawagga: comparative visualization of coiled-coil predictions and detection of stable single α-helices (SAH domains). Bioinformatics 31:767–769
13. Chou PY, Fasman GD (1977) Secondary structural prediction of proteins from their amino acid sequence. Trends Biochem Sci 2:128–131
14. Griep S, Hobohm U (2010) PDBselect 1992-2009 and PDBfilter-select. Nucleic Acids Res 38:D318–D319

Chapter 4

Predicting Protein Secondary Structure Using Consensus Data Mining (CDM) Based on Empirical Statistics and Evolutionary Information

Gaurav Kandoi, Sumudu P. Leelananda, Robert L. Jernigan, and Taner Z. Sen

Abstract

Predicting the secondary structure of a protein from its sequence still remains a challenging problem. The prediction accuracies remain around 80 %, and for very diverse methods. Using evolutionary information and machine learning algorithms in particular has had the most impact. In this chapter, we will first define secondary structures, then we will review the Consensus Data Mining (CDM) technique based on the robust GOR algorithm and Fragment Database Mining (FDM) approach. GOR V is an empirical method utilizing a sliding window approach to model the secondary structural elements of a protein by making use of generalized evolutionary information. FDM uses data mining from experimental structure fragments, and is able to successfully predict the secondary structure of a protein by combining experimentally determined structural fragments based on sequence similarities of the fragments. The CDM method combines predictions from GOR V and FDM in a hierarchical manner to produce consensus predictions for secondary structure. In other words, if sequence fragment are not available, then it uses GOR V to make the secondary structure prediction. The online server of CDM is available at http://gor.bb.iastate.edu/cdm/.

Key words Secondary structure, Protein structure prediction, GOR, Fragment database mining, Consensus data mining, Machine learning, Multiple sequence alignments

1 Introduction

One grand challenge problem in computational biology still stands—to use only amino acid sequences to predict the functions and structures of proteins. The challenge is still relevant, because despite having over 100,000 experimental structures in the Protein Data Bank (PDB), there are still many protein sequences of unknown structure and function, and genome sequencing continues to produce even more amino acid sequence data. New high resolution electron microscopy structure determination methods might improve upon this situation, but at this point in time we

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_4,

still have a major problem. The disparity between size of the experimental structure data and that of the sequence information emphasizes the urgency for developing better computational methods to predict protein structures. Because it is a one-dimensional problem, predicting secondary structures is a more feasible task than the ultimate goal of predicting full three-dimensional structures. Correctly predicting the secondary structure of proteins is crucial for several bioinformatics analyses, including predicting functions of proteins of unknown structures, discovering distant homologs, aligning structures [1], recognizing functional motifs, and analyzing whole genomes [2].

The most basic and most frequent secondary structural elements are α-helices, β-strands, and coils. A stringent definition is essential to define protein secondary structure elements from atomic coordinates deposited in the PDB. The dictionary of protein secondary structure (DSSP) developed by Kabsch and Sander [3] defines secondary structures consistently (an alternative is STRIDE [4] that also provides this information) based on geometrical features and hydrogen bonds. Secondary structure elements are classified into eight types with each represented by a single letter: E (extended β-strand), B (bridge, a single residue β-strand), H (α-helix), S (bend), I (π-helix), T (β-turn), G (3_{10} helix), and C (coil).

For the purpose of secondary structure prediction, eight types of structures may be too many and require too much detail. Therefore, some methods use reduced alphabets of four combined elements, i.e., extended (β-sheet) (E), coil (C), helix (H), and turn (T), or by combining turns with coil, only three. For example, in the Critical Assessment of Structure Prediction (CASP) [5] contest, the letter H is used for all three helices (H, I, and G); letter E is used for both strands (E) and bridge (D), and the remaining elements (S, T, and C) are considered as coil (C), though other ways of alphabet reduction have been used as well [4, 6–8]. Here we simply use the three states H, E, and C and combine the DSSP data in the way described above.

Protein secondary structure predictions have increased their prediction accuracy from around 60 % to about 80 % over several decades. Some of the earliest works in the field based on statistics derived from single amino acid residues includes those of Garnier et al. [9] (GOR I), Chou and Fasman [10], and Lim [11, 12]. Later, other methods were developed using nearest neighbor algorithms [6, 7, 13–17], neural networks [18–25], empirical statistics [9–12, 26, 27], hidden Markov models [28–33], and combined methods [34–39].

2 Methods

2.1 Information Theory and Bayesian Statistics

Information theory and Bayesian statistics constitute the foundation of the GOR [9] method, based on maximizing the information content represented by the information function $I(S, R)$. The basis for this method is the use of a function defined as the logarithm of the ratio of $P(S|R)$, the probability of observing the conformation S (in our case, helix (H), coil (C), or extended (E)) for the residue R (the type of amino acid) and $P(S)$, to the probability of occurrence of S:

$$I(S \mid R) = \log \frac{P(S \mid R)}{P(S)} \tag{1}$$

The information function can be computed from a database containing known protein secondary structure of sequences.

More specifically, given a protein sequence, the information function can be described as the information difference:

$$I(\Delta S; R_1, R_2, \ldots, R_n) = I(S; R_1, R_2, \ldots, R_n) - I(n - S; R_1, R_2, \ldots, R_N) \tag{2}$$

where n–S denotes the conformations other than S. For example, if S is C, then H and E represent $n - S$.

The total Information Content is defined as being the sum of much simpler events:

$$\begin{aligned} &I(\Delta S_{total}; R_1, R_2, \ldots, R_N) = I(\Delta S_1; R_1) + I(\Delta S_2; R_2 \mid R_1) \\ &+ \ldots + I(\Delta S_n; R_n \mid R_1, R_2, \ldots, R_{N-1}) \end{aligned} \tag{3}$$

where the indexes refer to the residue number with a total of N residues. If information from singlet and pairs of residues is used, then we obtain from algebraic operations:

$$\log \frac{P(S)}{P(n-S)} = \frac{1-2d}{2d+1} \sum_{m=-d}^{d} \log \frac{P(S; R_{j+m})}{P(n-S; R_{j+m})} + \frac{2}{2d+1} \sum_{n,m=-d}^{d} \log \frac{P(S; R_{j+m}, R_{j+n})}{P(n-S; R_{j+m}, R_{j+n})} \tag{4}$$

for the center residue in a given sliding window. Here, d refers to the number of residues on each side of the middle residue and $2d+1$ corresponds to total residues in the sliding window.

A sliding window of 17 residues (8 adjacent residues on each side of the jth residue) was used in the calculation of GOR versions I-IV [9, 27]. The sliding window was converted into a more dynamic form in the GOR V [26, 35], where a resizable window is used based on protein sequence length. The parameters of the information function can then be averaged over a set of experimentally determined secondary structures for the sequence.

2.2 Protein Secondary Structure Database

To estimate the parameters of the information function, a set of protein sequences with their known secondary structures is needed. For this, 513 nonredundant sequences (Cuff and Barton [40, 41]) (CB513) were used for GOR IV and V, as well as for CDM. This set contains information about the secondary structures for 84,107 residues.

2.3 Multiple Sequence Alignments

Several studies suggest that including evolutionary information in the prediction of secondary structures improves prediction accuracy [42]. Therefore, results from PSI-BLAST [43] in the form of multiple sequence alignments were used in GOR V [26, 35] to include evolutionary information. Up to five iterations of PSI-BLAST were run for CB513 sequences against the entire nr database. After assessing the performance of the method using various identity thresholds, it was found that using more diverse sequences, by removing highly identical alignments (>97% identity), improves the performance.

2.4 Data Mining

With the constant accumulation of protein structural data, data mining becomes more important and enables the development of faster and more efficient methods to extract biological knowledge. The concept of data mining is often combined with machine learning or artificial intelligence algorithms. GOR V [26, 35] uses data mining by integrating the results from PSI-BLAST, with Fragment Database Mining (FDM) [34], to obtain Consensus Data Mining (CDM) [36].

3 Consensus Data Mining (CDM)

3.1 GOR V

The method GOR V combines Bayesian statistics and information theory with multiple sequence alignments for protein secondary structure prediction. GOR calculates the probability of occurrence of secondary structure elements (H, C, and E) for every residue in the protein sequence. Since it considers only three possible structural elements, we have:

$$P_H + P_C + P_E = 1$$

where P_H, P_C, and P_E are the probabilities of elements H, C, and E structural states for a given residue.

GOR V predictions follow the steps below:

1. A multiple sequence alignment is produced using PSI-BLAST for the input protein sequence.
2. Using information theory as described above, the probabilities of elements H, C, and E at each residue position for the *j*th residue are calculated for the *i*th sequence in the multiple sequence alignment generated in **step 1**.
3. The results (P_H, P_C, and P_E) for all residues in the alignment are then stored in three matrices $P_H(i, j)$, $P_C(i, j)$, and $P_E(i, j)$

of size $n \times m$. Here, n refers to the number of alignments and m refers to the length of the alignment. $P_C(i, j)$ refers to the probability of coil in the ith multiple alignment sequence for the jth residue.

4. The probabilities $P_H(i, j)$, $P_C(i, j)$, and $P_E(i, j)$ at the jth residue in the multiple sequence alignments is summed over all alignments ($0 \le i \le n$). This sum is then divided by the number of sequences in the alignment (i.e., excluding those containing a gap and those with nonstandard residue or an unknown residue type) to obtain an average value. For residue j, the values of $P_H(j)$, $P_C(j)$, and $P_E(j)$ denote the probability of that structural element at position j.

Although usually the conformation at position j is predicted by the procedure above, there are a few exceptions to this. Very small heuristic constants have been added to prevent the overprediction of coil (C). Therefore, $P_C(j)$ will be greater by a margin from $P_H(j)$ (margin = 0.075) and $P_E(j)$ (margin = 0.15) for the jth residue to be predicted as a coil conformation. Helices (H) shorter than five residues are treated as coils and so are sheets (E) shorter than three residues (*see* **Note 4**).

GOR was at the time among the best methods for prediction of protein secondary structure: with an accuracy of 73.4%, GOR V [26, 35] is ~5% worse than other popular methods like PhD [44] and PSIPRED [20].

3.2 Fragment Database Mining (FDM)

A long-standing assumption in protein science is that similar sequences (usually over 50% sequence identity) will have a similar structure because sequences are less conserved than structures during evolution. Therefore, being able to harness the information content from similar sequences should aid protein secondary structure prediction. With this starting point, the FDM [34] method uses sequentially similar structure fragments for secondary structure prediction.

The database of 513 nonredundant domains (CB513) which was used in GOR V was also used for FDM. The reduced version of DSSP [3] is used to specify the three secondary structure elements (H, C, and E). BLAST is used to generate local sequence alignments using different substitution matrices (BLOSUM-45, -62, -80 and PAM-30, -70). Weights are then assigned to the matching segments in the form of an identity score and its power using:

$$\text{Identity score} = \text{id}^c$$

where c is a positive real number and id is the ratio of the number of exact matching residues to the total residues in the matching segment. The weights assigned according to the BLAST similarity or identity scores are then normalized for each position and a normalized score is calculated for each position for each of the three secondary structure states. This is done by calculating the normalized score for position i to be in state j:

$$s(j,i) = \frac{\sum w(j,i)}{\sum w(H,i) + \sum w(E,i) + \sum w(C,i)}$$

where $w(C, i)$ is the weight at position i in a coil in one matching segment and $w(E, i)$ and $w(H, i)$ are similarly defined.

The following parameters are used for assigning weights:

1. Substitution matrices: PAM-30, PAM-70, BLOSUM-45, BLOSUM-62, and BLOSUM-80.
2. Identity and Similarity cutoffs: Cutoff values of 60, 70, 80, 90, and 99 %.
3. Protein classification.
4. Protein Size.
5. Degree of solvent exposure of residues: buried (accessibility < 5), exposed (accessibility > 40), and intermediate for all others.

The best results are obtained using BLOSUM-45, power three for the identity score (id^3) and id cutoff of 0.99 as the weight assignment. When studying the effect of protein size, it was found that the prediction is more accurate for large proteins ($200 < \text{length} \leq 300$) and least accurate for very smalls ($\text{length} \leq 100$). The distinction of solvent exposure to residues has no significant impact on the prediction accuracy.

A major advantage of FDM [34] is that it uses structural templates for prediction of secondary structures. Thus, with the rise in the number of proteins in PDB, the performance of FDM [34] should increase and will likely outperform other methods not depending on the structural templates.

3.3 CDM

The GOR V [26, 35] and FDM [34] methods are built upon different principles. While GOR is an empirical statistical method, FDM [34] uses sequence similarity information to assign secondary structure. This complementarity provides an opportunity to combine these methods to take advantage of situations where one method outperforms the other method. With this assumption, GOR V [26, 35] is combined with FDM [34] to enhance their joint performance for protein secondary structure prediction.

When there is a protein in the PDB with a closely similar fragment sequence, the performance of FDM [34] is excellent. However, when there are divergent protein fragment sequences in PDB, FDM [34] performs poorly. By contrast, GOR V [26, 35] performs fairly accurately when the protein fragment sequence is less similar. This makes the two methods complementary to each other and the combined method has the advantage of utilizing the power of each of them. The only parameter that controls whether FDM [34] or GOR V [26, 35] is used for prediction is the sequence identity threshold. Conditional on the sequence identity score, GOR V [26, 35] (if the

score is below the sequence identity threshold) or FDM (if the score is above the sequence identity threshold) [34] is used to predict the secondary structure of the site. For the remaining sites, the complementary method is used.

In cases where there is a nearly perfect sequence identity, FDM [34] is used for predicting most of the residues. GOR V [26, 35] performs better without multiple sequence alignments (MSA) for the small portion of remaining residues than GOR V [26, 35] with MSA. CDM [36] has higher accuracy than GOR V [26, 35] irrespective of whether or not the MSA is included for an upper sequence identity limit of 90% or greater. Moreover, CDM [36] on average performs better than individual FDM [34] for the entire sequence regardless of the upper sequence identity limit.

3.4 Simple Algorithm for CDM

The CDM tool is available for public use on the web server: http://gor.bb.iastate.edu/cdm/. To improve prediction accuracy, the algorithm hierarchically integrates results from the GOR V and FDM methods. Briefly, the predictions are made as follows:

1. When a user submits a protein sequence, (*see* **Note 1–3, 5**), a Perl script runs PSI-BLAST (as input for GOR V) and PSI-BLAST (needed for FDM) against the nonredundant protein nr database.
2. GOR V and FDM are run independently to obtain separate secondary structure predictions.
3. Finally, the results from GOR V and FDM are combined to produce the consensus predictions based on the following procedure: first, predictions are made for sites with highly similar fragments (>55%) by FDM and then predictions from GOR V are used for the remaining sites.

4 Notes

1. The GOR V [26, 35] server can be accessed from: http://gor.bb.iastate.edu/ and CDM [36] can be found at: http://gor.bb.iastate.edu/cdm/.
2. The query sequence must be a single letter amino acid code.
3. The results from the CDM [36] are emailed to the user, and thus requires a valid email address.
4. Sheets that are shorter than three residues (E or EE) and helices shorter than five residues (H, HH, HHH, and HHHH) are predicted as coils.
5. Query sequences containing an unknown residue (X) are permitted.

5 Discussion

Protein structure prediction is a key goal of bioinformatics along with predicting the function of an amino acid sequence. The function of a protein is often guided by the correct folding of its three-dimensional structure which in turn depends on its sequence. Determining protein structures experimentally is a very time-consuming task even with the current state-of-the-art technologies. Therefore, there is a strong need for computational methods capable of predicting the correct folding of a protein into its two- and three-dimensional structures. Such methods provide us an opportunity to obtain insights into protein structure, evolution, and function beyond what is available experimentally.

Several different approaches have been used to successfully predict protein secondary structural elements. The heterogeneity in the prediction techniques allows us to combine them in order to achieve ensemble/hybrid methods with higher prediction accuracy than any of its individual components. With an increasing number of experimentally determined structures in the PDB, the accuracy of this method will improve. This method, based on abundant high-quality data, will be better able to capture the patterns underlying sequence-structure-function relationships and would be central in our understanding of biological systems.

Another opportunity for gains would be to select only sequences of similar size in the multiple sequence alignment. For a given size of structure there is a limit to the sizes of the secondary structure elements that should be approximately reflected in the structures of similar size. Often secondary structure elements are predicted to be longer than is possible for smaller sequences, and making this restriction on the sizes of the sequences in the alignments should partly correct for this effect of size.

References

1. Krissinel E, Henrick K (2004) Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr 60(Pt 12 Pt 1):2256–2268. doi:10.1107/S0907444904026460
2. Rost B (2001) Review: protein secondary structure prediction continues to rise. J Struct Biol 134(2–3):204–218. doi:10.1006/jsbi.2001.4336
3. Kabsch W, Sander C (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers 22(12):2577–2637. doi:10.1002/bip.360221211
4. Frishman D, Argos P (1995) Knowledge-based protein secondary structure assignment. Proteins 23(4):566–579. doi:10.1002/prot.340230412
5. Moult J, Pedersen JT, Judson R, Fidelis K (1995) A large-scale experiment to assess protein structure prediction methods. Proteins 23(3):ii–iv
6. Biou V, Gibrat JF, Levin JM, Robson B, Garnier J (1988) Secondary structure prediction: combination of three different methods. Protein Eng 2:185–191
7. Salamov AA, Solovyev VV (1995) Prediction of protein secondary structure by combining nearest-neighbor algorithms and multiple sequence alignments. J Mol Biol 247:11–15

8. Rost B, Sander C (2000) Third generation prediction of secondary structures. Methods Mol Biol 143:71–95
9. Garnier J, Osguthorpe DJ, Robson B (1978) Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. J Mol Biol 120:97–120
10. Chou PY, Fasman GD (1974) Prediction of protein conformation. Biochemistry 13:222–245
11. Lim VI (1974) Algorithms for prediction of alpha-helical and beta-structural regions in globular proteins. J Mol Biol 88:873–894
12. Lim VI (1974) Structural principles of the globular organization of protein chains. A stereochemical theory of globular protein secondary structure. J Mol Biol 88:857–872
13. Levin JM, Garnier J (1988) Improvements in a secondary structure prediction method based on a search for local sequence homologies and its use as a model building tool. Biochim Biophys Acta 955:283–295
14. Levin JM, Robson B, Garnier J (1986) An algorithm for secondary structure determination in proteins based on sequence similarity. FEBS Lett 205:303–308
15. Salamov AA, Solovyev VV (1997) Protein secondary structure prediction using local alignments. J Mol Biol 268:31–36
16. Salzberg S, Cost S (1992) Predicting protein secondary structure with a nearest-neighbor algorithm. J Mol Biol 227:371–374
17. Yi TM, Lander ES (1993) Protein secondary structure prediction using nearest-neighbor methods. J Mol Biol 232:1117–1129
18. Frishman D, Argos P (1997) Seventy-five percent accuracy in protein secondary structure prediction. Proteins 27:329–335
19. Holley LH, Karplus M (1989) Protein secondary structure prediction with a neural network. Proc Natl Acad Sci U S A 86:152–156
20. Jones DT (1999) Protein secondary structure prediction based on position-specific scoring matrices. J Mol Biol 292:195–202
21. Petersen TN, Lundegaard C, Nielsen M, Bohr H, Bohr J, Brunak S, Gippert GP, Lund O (2000) Prediction of protein secondary structure at 80% accuracy. Proteins 41:17–20
22. Qian N, Sejnowski TJ (1988) Predicting the secondary structure of globular proteins using neural network models. J Mol Biol 202:865–884
23. Rost B, Sander C (1993) Prediction of protein secondary structure at better than 70% accuracy. J Mol Biol 232:584–599
24. Rost B, Sander C, Schneider R (1994) PHD--an automatic mail server for protein secondary structure prediction. Comput Appl Biosci 10:53–60
25. Stolorz P, Lapedes A, Xia Y (1992) Predicting protein secondary structure using neural net and statistical methods. J Mol Biol 225:363–377
26. Kloczkowski A, Ting KL, Jernigan RL, Garnier J (2002) Combining the GOR V algorithm with evolutionary information for protein secondary structure prediction from amino acid sequence. Proteins 49:154–166
27. Kloczkowski A, Ting KL, Jernigan RL, Garnier J (2002) Protein secondary structure prediction based on the GOR algorithm incorporating multiple sequence alignment information. Polymer 43:441–449
28. Bystroff C, Thorsson V, Baker D (2000) HMMSTR: a hidden Markov model for local sequence-structure correlations in proteins. J Mol Biol 301:173–190
29. Söding J, Biegert A, Lupas AN (2005) The HHpred interactive server for protein homology detection and structure prediction. Nucleic Acids Res 33(suppl 2):W244–W248
30. Karplus K (2009) SAM-T08, HMM-based protein structure prediction. Nucleic Acids Res 37(suppl 2):W492–W497
31. Asai K, Hayamizu S, Handa KI (1993) Prediction of protein secondary structure by the hidden Markov model. Comput Appl Biosci 9(2):141–146
32. Li SC, Bu D, Xu J, Li M (2008) Fragment-HMM: a new approach to protein structure prediction. Protein Sci 17(11):1925–1934
33. Ding W, Dai D, Xie J, Zhang H, Zhang W, Xie H (2012) PRT-HMM: A novel hidden Markov model for protein secondary structure prediction. In Computer and information science (ICIS), 2012 IEEE/ACIS 11th international conference on. IEEE. pp 207–212
34. Cheng H, Sen TZ, Kloczkowski A, Margaritis D, Jernigan RL (2005) Prediction of protein secondary structure by mining structural fragment database. Polymer 46:4314–4321
35. Sen TZ, Jernigan RL, Garnier J, Kloczkowski A (2005) GOR V server for protein secondary structure prediction. Bioinformatics 21:2787–2788
36. Sen TZ, Cheng H, Kloczkowski A, Jernigan RL (2006) A consensus data mining secondary structure prediction by combining GOR V and fragment database mining. Protein Sci 15:2499–2506
37. Cheng H, Sen TZ, Jernigan RL, Kloczkowski A (2007) Consensus data mining (CDM) protein secondary structure prediction server: combining GOR V and fragment database mining (FDM). Bioinformatics 23:2628–2630

38. Faraggi E, Zhang T, Yang Y, Kurgan L, Zhou Y (2012) SPINE X: improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles. J Comput Chem 33(3):259–267
39. Heffernan R, Paliwal K, Lyons J, Dehzangi A, Sharma A, Wang J, Sattar A, Yang Y, Zhou Y (2015) Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning. Sci Rep 5:11476
40. Cuff JA, Barton GJ (1999) Evaluation and improvement of multiple sequence methods for protein secondary structure prediction. Proteins 34(4):508–519
41. Cuff JA, Barton GJ (2000) Application of multiple sequence alignment profiles to improve protein secondary structure prediction. Proteins 40(3):502–511
42. Simossis VA, Heringa J (2004) Integrating protein secondary structure prediction and multiple sequence alignment. Curr Protein Pept Sci 5(4):249–266
43. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25(17):3389–3402
44. Rost B (2003) Prediction in 1D: secondary structure, membrane helices, and accessibility. Methods Biochem Anal 44:559–587

Chapter 5

Accurate Prediction of One-Dimensional Protein Structure Features Using SPINE-X

Eshel Faraggi and Andrzej Kloczkowski

Abstract

Accurate prediction of protein secondary structure and other one-dimensional structure features is essential for accurate sequence alignment, three-dimensional structure modeling, and function prediction. SPINE-X is a software package to predict secondary structure as well as accessible surface area and dihedral angles ϕ and ψ. For secondary structure SPINE-X achieves an accuracy of between 81 and 84 % depending on the dataset and choice of tests. The Pearson correlation coefficient for accessible surface area prediction is 0.75 and the mean absolute error from the ϕ and ψ dihedral angles are 20° and 33°, respectively. The source code and a Linux executables for SPINE-X are available from Research and Information Systems at http://mamiris.com.

Key words Secondary structure, Dihedral angles, Accessible surface area, Protein structure

1 Introduction

Proteins are the biological machinery that carry out the instructions contained in the genetic code. As such, proteins are responsible on some level for all biological function. Proteins perform their functional duties by various interactions associated with their three-dimensional structure. Their three-dimensional structure is thought to be determined by boundary effects and their amino-acid sequence which is encoded in the genetic material of the organism [1]. While the genetic sequence, and hence the protein's amino-acid sequence, can be obtained using automated experimental procedures relatively cheaply, currently, the structure of proteins can only be experimentally obtained using labor-intensive and costly procedures [1, 2]. This makes protein structure prediction important.

Protein structures are typically categorized into four levels of increasing structural information. The first level in this categorization, sometimes called the primary structure, is the amino-acid sequence of the protein. The second level is the so-called secondary structure which involves a coarse grained view of the local structure of the protein along its amino-acid sequence. The third

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_5,

scale of structural information, the tertiary structure, is associated with the organization of secondary structure elements into single domain protein structure. The fourth level of structural classification is called the quaternary structure and is associated with the coalescence of tertiary structure elements into functioning biological components. This hierarchy enables proteins to bridge the size gap between individual atoms and biological components.

Besides secondary structure [3–22] other invariant one-dimensional properties that can be associated with individual residues exist. Most notably maybe is the accessible surface area [21, 23–38] that measures how much of a given residue can interact with the solvent. Another one-dimensional property are the dihedral angles [30, 39–44]. Notice that to be useful, we specifically insist that our one-dimensional structural properties be invariant under a coordinate transformation.

Prediction of protein structure usually employ a hierarchy as well. Secondary structure predictions are used to set initial conditions and act as constraints in three-dimensional prediction schemes [43, 45–50]. We have shown that substitution of dihedral angles for secondary structure constraints in no-homology modeling of three-dimensional structure results in a 100% improvement in the prediction accuracy [43]. Part of that approach of predicting dihedral angles uses secondary structure predictions as input features. Here we show how to use SPINE-X to predict secondary structures, accessible surface area, and the dihedral angles ϕ and ψ from the amino-acid sequence. Accurate prediction of protein secondary structure and other one-dimensional structure features is essential for accurate sequence alignment [51, 52], three-dimensional structure modeling [53–55], and function prediction [56, 57].

2 Materials and Methods

SPINE-X consists of six steps of iterative prediction of secondary structure, real-value residue solvent accessibility, and torsion angles. The first five steps predict real value torsion angles (both ϕ and ψ) [43]. It begins with generating the Position Specific Scoring Matrix (PSSM) using the PSIBLAST [43, 58] and seven representative physical parameters (PP) including a steric parameter (graph shape index), hydrophobicity, volume, polarizability, isoelectric point, helix probability, and sheet probability [59]. In the first step, a neural network is set up to predict secondary structure (SS0) employing PSSM and PP as input. The secondary structure was defined according to SKSP [60], a consensus assignment of four methods (STRIDE [61], KAKSI [62], SECSR [63], and P-SEA [64]), plus a further modification for those helical and sheet residues that are located in incorrect sheet or helical torsional angle regions, respectively (labeled as SKSP+) [22, 43].

In the second step, another neural network is built to predict residue solvent accessibility with PSSM, PP and predicted SS0 as input. These first two steps correspond to Real-SPINE 3.0 for real-value prediction of solvent accessibility [21] except that the predicted secondary structure is based on SKSP+. Then, predicted RSA and SS0 together with PSSM and PP are used to predict the torsion angles. The fourth step is to perform a new round of SKSP+ secondary structure prediction (SS1) based on previous predictions and with the PSSM and PP. Then a new round of torsion angle prediction is performed with previous predictions, PSSM, and PP.

The sixth and final step is a neural network that is trained to predict DSSP [65] assigned secondary structure using PSSM, PP, predicted values from the first five rounds. This step is useful when comparing with other methods that use the DSSP assignment. The eight-state DSSP assignments were grouped as follows: the 3-helix (G), alpha-helix (H), and pi-helix (I) into state H; beta-bridge (B) and extended-strand (E) into state E; and hydrogen-bonded-turn (T), bend (S), and other (_) into state C.

In each step, the general form of the neural networks is the same. It consists of two hidden layers with 101 hidden nodes. All weights were guided based on sequence separation [21]. A 21-residue window is employed. For a given 21-residue input window the target output is the one-dimensional property of the central residue in the window. Vacant locations in the windows around residues near the terminals of a protein were explicitly excluded from the training by limiting the range of the input window. We employed a bipolar activation function given by $f(x) = \tanh(\alpha x)$, with $\alpha = 0.2$, momentum of 0.4, and the back-propagation method with a learning rate of 0.001. These parameters were optimized in previous studies of torsion angles and solvent accessibility [21, 22, 42, 43].

Training and initial testing for all neural networks considered here were performed on the SPINE dataset of 2640 PDB [66] protein and on its subset of 2479 proteins with length less than 500 residues [21, 43]. The dataset of 2640 proteins was obtained from the protein sequence culling server PISCES [67, 68] with sequence identity less than 25%, X-ray resolution better than 3Å, and without unknown structural regions in early 2006 [17]. The subset of 2479 proteins was employed because we are interested to know if excluding long chains would lead to an improved secondary structure prediction as long chains will normally involve more nonlocal interactions. The final SPINE X server was built based on the subset of 2479 proteins.

The overall accuracy estimates for SPINE-X are as follows. For secondary structure the accuracy is between 81 and 84 % depending on the dataset and choice of tests. The Pearson correlation coefficient for accessible surface area prediction is 0.75 and the mean absolute error from the ϕ and ψ dihedral angles are 20° and 33°, respectively. Refer to the original publications [21, 22, 43] of these methods for a more detailed analysis of the accuracy of SPINE-X.

/tmp/1UBSB.spXout.ifort 62753 bytes

index	RR	SS	phi1	psi1	P_E	P_C	P_H	phi0	psi0	HSB	S_pk	S_SS	pk_phi	pk_psi	pkc_phi	pkc_psi
1	Q	E	-98.1	146.2	0.5081	0.4674	0.0246	-97.9	144.5	57.3	0.1797	0.7196	-5.0000	-5.0000	0.9963	0.9232
2	T	E	-110.3	140.2	0.8883	0.0832	0.0285	-103.2	138.7	43.2	0.0718	0.3765	-5.0000	-5.0000	0.9982	0.9235
3	I	E	-107.0	127.8	0.9072	0.0651	0.0238	-107.1	125.0	3.9	0.1722	0.3294	-5.0000	-5.0000	0.9987	0.9264
4	T	E	-103.0	129.9	0.9204	0.0426	0.0270	-111.6	127.5	16.7	0.4671	0.2582	-5.0000	-5.0000	0.9991	0.9278
5	Y	E	-111.6	127.0	0.9322	0.0406	0.0272	-111.5	126.7	3.5	0.3704	0.2648	-5.0000	-5.0000	0.9989	0.9282
6	K	E	-113.9	143.1	0.8822	0.0871	0.0306	-117.7	140.0	8.8	0.2215	0.3913	-5.0000	-5.0000	0.9946	0.9454
7	S	E	-108.9	152.3	0.8548	0.1155	0.0297	-112.6	153.7	6.7	0.3325	0.4440	-5.0000	-5.0000	0.9943	0.9368
8	H	C	-90.5	160.4	0.3388	0.6241	0.0370	-88.7	148.4	47.1	0.3633	0.7127	-5.0000	1.0000	0.9689	0.5101
9	Q	T	-82.5	-31.5	0.1475	0.3697	0.4828	-79.7	-27.0	78.6	0.8536	0.3327	-5.0000	-5.0000	0.9231	0.7373
10	T	C	-101.6	6.9	0.1553	0.3115	0.5332	-104.5	6.2	95.2	0.5779	0.3137	-5.0000	-5.0000	0.9452	0.7069
11	G	T	63.8	167.7	0.0390	0.1832	0.0768	66.2	165.4	19.8	0.6267	0.3883	5.0000	5.0000	0.3411	0.1602
12	P	H	-63.2	-25.4	0.0297	0.4517	0.5186	-66.5	-25.1	77.4	0.4873	0.7318	-5.0000	-5.0000	0.9955	0.8774
13	E	H	-73.7	-29.3	0.0245	0.3125	0.6630	-81.7	-29.4	63.5	0.5229	0.6616	-5.0000	-5.0000	0.9880	0.8850
14	L	H	-63.4	-38.3	0.0214	0.0704	0.9082	-61.8	-36.4	37.8	0.4041	0.3245	-5.0000	-5.0000	0.9957	0.9159
15	Q	H	-64.6	-34.7	0.0092	0.0459	0.9449	-64.1	-34.5	108.5	0.1892	0.2595	-5.0000	-5.0000	0.9971	0.9373
16	D	H	-63.4	-36.5	0.0132	0.0381	0.9487	-70.3	-38.1	44.9	0.2212	0.2331	-5.0000	-5.0000	0.9925	0.9610
17	V	H	-65.9	-41.9	0.0183	0.0328	0.9490	-65.0	-41.5	4.9	0.1915	0.2137	-5.0000	-5.0000	0.9991	0.9718
18	K	H	-58.7	-41.5	0.0183	0.0319	0.9498	-59.8	-41.5	100.8	0.2264	0.2112	-5.0000	-5.0000	0.9973	0.9780
19	Q	H	-62.1	-41.7	0.0200	0.0286	0.9514	-62.6	-42.6	100.4	0.1395	0.2070	-5.0000	-5.0000	0.9854	0.9804
20	T	H	-57.0	-45.3	0.0270	0.0274	0.9456	-58.9	-44.7	5.8	0.1648	0.2065	-5.0000	-5.0000	0.9958	0.9855
21	A	H	-58.2	-46.4	0.0216	0.0260	0.9524	-58.7	-45.6	0.0	0.0949	0.2040	-5.0000	-5.0000	0.9948	0.9875
22	A	H	-61.2	-41.1	0.0203	0.0276	0.9520	-60.9	-42.3	50.4	0.0718	0.2049	-5.0000	-5.0000	0.9966	0.9878
23	Q	H	-64.2	-43.9	0.0191	0.0284	0.9525	-63.0	-43.4	77.7	0.0733	0.2030	-5.0000	-5.0000	0.9962	0.9889
24	H	H	-64.6	-37.4	0.0202	0.0292	0.9506	-63.5	-37.4	7.1	0.1286	0.2095	-5.0000	-5.0000	0.9947	0.9867
25	A	H	-63.5	-37.9	0.0219	0.0358	0.9423	-64.7	-35.2	22.0	0.2717	0.2536	-5.0000	-5.0000	0.9905	0.9811
26	K	H	-61.5	-37.7	0.0210	0.0350	0.9440	-64.8	-40.7	152.1	0.0797	0.2330	-5.0000	-5.0000	0.9945	0.9810
27	R	H	-70.4	-22.9	0.0209	0.0375	0.9416	-71.5	-21.4	70.1	0.2266	0.2372	-5.0000	-5.0000	0.9966	0.9768
28	H	H	-126.7	5.5	0.0256	0.0457	0.8887	-126.3	-7.4	53.7	0.5743	0.3777	-5.0000	-5.0000	0.9965	0.8637

Fig. 1 An example of the output from SPINE-X. The columns numbered in parenthesis from left to right are: (1) The index of the residue in the protein chain supplied. (2) Residue type. (3) Assigned secondary structure prediction. (4,5) The final dihedral angle prediction. (6,7,8) Probability to be in sheet, coil, and helix secondary structure state, respectively. (9,10) Initial prediction of dihedral angles. (11) Accessible surface area prediction (12,13) The Shannon entropy for the dihedral angle peak prediction and secondary structure prediction. (14,15) The assigned peak for the ϕ and ψ dihedral angles. (16,17) The probability of peak assignment for the ϕ and ψ dihedral angles, respectively

In Fig. 1 we give an example of the output one gets from SPINE-X. Starting from the left the first column gives the index of the residue in the protein chain supplied. Column number two gives the residue type, while column number three gives the assigned secondary structure prediction. Columns four and five

give the final ϕ and ψ angle prediction, respectively, obtained in the last step of the iterative process. Columns six, seven, and eight give the predicted probability to be in sheet, coil, and helix secondary structure state, respectively. Columns nine and ten give the initial prediction of dihedral angles obtained in the first step of the iterative process. Column 11 gives the accessible surface area prediction. Columns 12 and 13 give the Shannon entropy for the dihedral angle peak prediction and secondary structure prediction, respectively. These are calculated from the predicted probabilities. Columns 14 and 15 give the assigned peak for the ϕ and ψ dihedral angles. These are calculated by assigning – 1 and 1 to the two peaks and summing the predictions from the five neural networks ensemble used. Columns 16 and 17 give the probability of peak assignment for the ϕ and ψ dihedral angles, respectively. In this case the probability of being in the peak assigned as – 1 is given.

3 Notes

1. **Source and Availability** SPINE-X was implemented in FORTRAN and is supplied in source code format. Running SPINE-X requires supplying a sequence file in the FASTA format, or if available, a PSSM matrix. The use of a PSSM matrix will significantly reduce the running time. A few wrappers then take this information and generate input features for the optimized iterative neural networks. Helpful documentation, executables, and sources are available from Research and Information Systems at http://mamiris.com.
2. **License** The license for SPINE-X is available from a file called "LICENSE" in its distribution directory. It allows for academic users the opportunity to use and modify it freely with proper citations while retaining some rights for commercial use.
3. **Installation** To run SPINE-X you must install it. The original program was compiled on a Linux system using Intel's ifort (freely available for academic use at Intel's website). The compile script given in the code directory assumes you have ifort or gfortran installed and that its location is in your path. Note: if you are using a Linux system, there is some chance that the compiled binaries supplied with this distribution will work on your system. Test cautiously, at your own risk while backing up anything important. To compile the source code, go to the code directory and issue the command *./compile*. If no errors were reported, you should be able to use spineX. Consult the script "compile" in the directory "code/" if errors did spring up. Further information is available in the "README" files that are packed with the distribution.

4. **blastpgp** You must have blastpgp installed on your system. You should set the environment variable spineXcodir pointing to the location of the spineX code, e.g., export spineXcodir=/path/. Depending on your local configuration you may also need to set spineXblast to point to the BLAST root directory.
5. **Test** The directory test has an example to see that your version of spineX is close to the official web-version. Minor differences will occur because you are probably using a different version of PsiBLAST but your results on the test protein chain 1URSA should not be more than about a percent different from the original predictions. The trend we observed is that the newer your version of PsiBLAST the better will be your predictions.
6. **Usage** To run SPINE-X, *spX.pl* is in your path, use *spX.pl idlistfile /prof/fastalocation* where *idlistfile* is a file containing the id name of chains to predict (e.g., id1) and */prof/fastalocation* is the path to the directory with the psiblast profiles with. mat suffix (e.g., id1.mat) or the fasta files with.fasta suffix (e.g., id1.fasta). Note, name conventions on prf/fasta files are strict. The program will specifically look for these names. For example, in the test directory *test/* one should use *../spX.pl list1./* to give predictions for the chains identified in *list1*. The output from SPINE-X is labeled as in *list1*. The indexes 0,1 associated with the phi and psi angles designate the iteration number. S_ refers to the information entropy calculated from the predicted probabilities of secondary structure. pk_ designates predicted peak assignments [43].
7. **Parallel Usage** If you have many files and several processors on the same machine to handle them, use the program spXbiglist.pl to automatically split the file list and run these parts in parallel. Type *./spXbiglist.pl* at the shell prompt to get help on using this program. Note: by default a *-a 4* option is passed to blastpgp for a regular run, this is unrelated to the use of *spXbiglist.pl*.
8. **DSSP Based Prediction** To obtain the SPINE-X prediction of secondary structure according to DSSP assignments use the program *phipsi_dssp.e*. Type *./phipsi_dssp.e* at the shell prompt to get help on using this program.

Acknowledgements

We gratefully acknowledge the financial support provided by the National Institutes of Health (NIH) through Grants R01GM072014 and R01GM073095 to Andrzej Kloczkowski. Support was also provided by Yaoqi Zhou through NIH Grant R01GM085003.

References

1. Creighton TE (1993) Proteins: structures and molecular properties. Macmillan, New York
2. Yang J, Yan R, Roy A, Xu D, Poisson J, Zhang Y (2015) The I-TASSER suite: protein structure and function prediction. Nat Methods 12(1):7–8
3. Garnier J, Osguthorpe D, Robson B (1978) Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. J Mol Biol 120(1):97–120
4. Gibrat JF, Garnier J, Robson B (1987) Further developments of protein secondary structure prediction using information theory: new parameters and consideration of residue pairs. J Mol Biol 198(3):425–443
5. Holley LH, Karplus M (1989) Protein secondary structure prediction with a neural network. Proc Natl Acad Sci 86(1):152–156
6. Kneller D, Cohen F, Langridge R (1990) Improvements in protein secondary structure prediction by an enhanced neural network. J Mol Biol 214(1):171–182
7. Sikorski A (1992) Prediction of protein secondary structure by neural networks: encoding short and long range patterns of amino acid packing*. Acta Biochim Pol 39(4)
8. Rost B, Sander C, et al (1993) Prediction of protein secondary structure at better than 70% accuracy. J Mol Biol 232(2):584–599
9. Rost B, Sander C (1993) Improved prediction of protein secondary structure by use of sequence profiles and neural networks. Proc Natl Acad Sci 90(16):7558–7562
10. Rost B, Sander C, Schneider R (1994) PHD-an automatic mail server for protein secondary structure prediction. Comput Appl Biosci 10(1):53–60
11. Garnier J, Gibrat JF, Robson B, et al (1996) GOR method for predicting protein secondary structure from amino acid sequence. Methods Enzymol 266:540
12. Cuff JA, Clamp ME, Siddiqui AS, Finlay M, Barton GJ (1998) JPred: a consensus secondary structure prediction server. Bioinformatics 14(10):892–893
13. Jones DT (1999) Protein secondary structure prediction based on position-specific scoring matrices. J Mol Biol 292:195–202
14. Cuff JA, Barton GJ (2000) Application of multiple sequence alignment profiles to improve protein secondary structure prediction. Proteins Struct Funct Bioinf 40(3):502–511
15. Hua S, Sun Z, et al (2001) A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach. J Mol Biol 308(2): 397–408
16. Rost B (2001) Review: protein secondary structure prediction continues to rise. J Struct Biol 134:204–218
17. Dor O, Zhou Y (2007) Achieving 80% ten-fold cross-validated accuracy for secondary structure prediction by large-scale training. Proteins Struct Funct Bioinf 66:838–845
18. Yoo PD, Zhou BB, Zomaya AY (2008) Machine learning techniques for protein secondary structure prediction: an overview and evaluation. Curr Bioinforma 3(2):74–86
19. Zhou Y, Faraggi E (2010) Prediction of one-dimensional structural properties of proteins by integrated neural networks. In: Introduction to protein structure prediction: methods and algorithms, pp 45–74
20. Mooney C, Vullo A, Pollastri G (2006) Protein structural motif prediction in multidimensional phi-psi space leads to improved secondary structure prediction. J Comput Biol 13: 1489–1502
21. Faraggi E, Xue B, Zhou Y (2009) Improving the prediction accuracy of residue solvent accessibility and real-value backbone torsion angles of proteins by guided-learning through a two-layer neural network. Proteins Struct Funct Bioinf 74(4):847–856
22. Faraggi E, Zhang T, Yang Y, Kurgan L, Zhou Y (2012) Spine x: improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles. J Comput Chem 33(3):259–267
23. Lee B, Richards FM (1971) The interpretation of protein structures: estimation of static accessibility. J Mol Biol 55(3):379–IN4
24. Chothia C (1974) Hydrophobic bonding and accessible surface area in proteins. Nature 248(5446):338–339
25. Chothia C (1976) The nature of the accessible and buried surfaces in proteins. J Mol Biol 105(1):1–12
26. Holbrook SR, Muskal SM, Kim SH (1990) Predicting surface exposure of amino acids from protein sequence. Protein Eng 3(8): 659–665
27. Rost B, Sander C (1994) Conservation and prediction of solvent accessibility in protein families. Proteins Struct Funct Genet 20(3): 216–226
28. Ahmad S, Gromiha MM, Sarai A (2003) Real value prediction of solvent accessibility from

amino acid sequence. Proteins Struct Funct Bioinf 50:629–635

29. Moret M, Zebende G (2007) Amino acid hydrophobicity and accessible surface area. Phys Rev E 75(1):011920
30. Dor O, Zhou Y (2007) Real-spine: an integrated system of neural networks for real-value prediction of protein structural properties. Proteins Struct Funct Bioinf 68(1):76–81
31. Durham E, Dorr B, Woetzel N, Staritzbichler R, Meiler J (2009) Solvent accessible surface area approximations for rapid and accurate protein structure prediction. J Mol Model 15(9): 1093–1108
32. Zhang H, Zhang T, Chen K, Shen S, Ruan J, Kurgan L (2009) On the relation between residue flexibility and local solvent accessibility in proteins. Proteins Struct Funct Bioinf 76(3): 617–636
33. Zhang T, Zhang H, Chen K, Ruan J, Shen S, Kurgan L (2010) Analysis and prediction of rna-binding residues using sequence, evolutionary conservation, and predicted secondary structure and solvent accessibility. Curr Protein Pept Sci 11(7):609–628
34. Gao J, Zhang T, Zhang H, Shen S, Ruan J, Kurgan L (2010) Accurate prediction of protein folding rates from sequence and sequence-derived residue flexibility and solvent accessibility. Proteins Struct Funct Bioinf 78(9):2114–2130
35. Nunez S, Venhorst J, Kruse CG (2010) Assessment of a novel scoring method based on solvent accessible surface area descriptors. J Chem Inf Model 50(4):480–486
36. Faraggi E, Zhang T, Yang Y, Kurgan L, Zhou Y (2012) Spine x: Improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles. J Comput Chem 33:259–267
37. Faraggi E, Zhou Y, Kloczkowski A (2014) Accurate single-sequence prediction of solvent accessible surface area using local and global features. Proteins Struct Funct Bioinf 82(11): 3170–3176
38. Wang C, Xi L, Li S, Liu H, Yao X (2012) A sequence-based computational model for the prediction of the solvent accessible surface area for α-helix and β-barrel transmembrane residues. J Comput Chem 33(1):11–17
39. Ramachandran GN, Sasisekharan V (1968) Conformation of polypeptides and proteins. Adv Protein Chem 23:283–437
40. Wood MJ, Hirst JD (2005) Protein secondary structure prediction with dihedral angles. Proteins Struct Funct Bioinf 59:476–481
41. Zimmermann O, Hansmann UHE (2006) Support vector machines for prediction of dihedral angle regions. Bioinformatics 22: 3009–3015
42. Xue B, Dor O, Faraggi E, Zhou Y (2008) Real value prediction of backbone torsion angles. Proteins Struct Funct Bioinf 72: 427–433
43. Faraggi E, Yang Y, Zhang S, Zhou Y (2009) Predicting continuous local structure and the effect of its substitution for secondary structure in fragment-free protein structure prediction. Structure 17:1515–1527
44. Kountouris P, Hirst JD (2009) Prediction of backbone dihedral angles and protein secondary structure using support vector machines. BMC Bioinf 10(1):437
45. Rost B (1995) TOPITS: threading one-dimensional predictions into three-dimensional structures. In: Third international conference on intelligent systems for molecular biology. AAAI Press, Menlo Park, CA, pp 314–321
46. Rost B, Sander C (1997) Protein fold recognition by prediction-based threading. J Mol Biol 270:471–480
47. Przybylski D, Rost B (2004) Improving fold recognition without folds. J Mol Biol 341: 255–269
48. Qiu J, Elber R (2006) SSALN: an alignment algorithm using structure-dependent substitution matrices and gap penalties learned from structurally aligned protein pairs. Proteins Struct Funct Bioinf 62:881–891
49. Cheng J, Baldi P (2006) A machine learning information retrieval approach to protein fold recognition. Bioinformatics 22:1456–1463
50. Liu S, Zhang C, Liang S, Zhou Y (2007) Fold recognition by concurrent use of solvent accessibility and residue depth. Proteins Struct Funct Bioinf 68:636–645
51. Huang YM, Bystroff C (2006) Improved pairwise alignments of proteins in the Twilight Zone using local structure predictions. Bioinformatics 22:413–422
52. Simossis V, Heringa J (2004) Integrating protein secondary structure prediction and multiple sequence alignment. Curr Protein Pept Sci 5(4):249–266
53. Zhang W, Liu S, Zhou Y (2008) SP^5: improving protein fold recognition by using predicted

torsion angles and profile-based gap penalty. PLoS One 3:e2325

54. Kihara D, Lu H, Kolinski A, Skolnick J (2001) Touchstone: an ab initio protein structure prediction method that uses threading-based tertiary restraints. Proc Natl Acad Sci 98(18): 10125–10130
55. Kolinski A et al (2004) Protein modeling and structure prediction with a reduced representation. Acta Biochim Pol 51:349–372
56. Liang S, Zhang C, Liu S, Zhou Y (2006) Protein binding site prediction using an empirical scoring function. Nucleic Acids Res 34(13): 3698–3707
57. Radivojac P, Clark WT, Oron TR, Schnoes AM, Wittkop T, Sokolov A, Graim K, Funk C, Verspoor K, Ben-Hur A, et al (2013) A large-scale evaluation of computational protein function prediction. Nat Methods 10(3): 221–227
58. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25:3389–3402
59. Meiler J, Muller M, Zeidler A, Schmaschke F (2001) Generation and evaluation of dimension reduced amino acid parameter representations by artificial neural networks. J Mol Model 7:360–369
60. Zhang W, Dunker AK, Zhou Y (2008) Assessing secondary-structure assignment of protein structures by using pairwise sequence-alignment benchmarks. Proteins Struct Funct Bioinf 71:61–67
61. Frishman D, Argos P (1995) Knowledge-based protein secondary structure assignment. Proteins Struct Funct Genet 23(4):566–579
62. Martin J, Letellier G, Marin A, Taly JF, De Brevern AG, Gibrat JF (2005) Protein secondary structure assignment revisited: a detailed analysis of different assignment methods. BMC Struct Biol 5(1):17
63. Fodje M, Al-Karadaghi S (2002) Occurrence, conformational features and amino acid propensities for the π-helix. Protein Eng 15(5):353–358
64. Labesse G, Colloc'h N, Pothier J, Mornon JP (1997) P-sea: a new efficient assignment of secondary structure from cα trace of proteins. Comput Appl Biosci 13(3):291–295
65. Kabsch W, Sander C (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers 22:2577–2637
66. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat T, Weissig H, Shindyalov IN, Bourne PE (2000) The protein data bank. Nucleic Acids Res 28(1):235–242
67. Wang G, Dunbrack RL (2003) PISCES: a protein sequence culling server. Bioinformatics 19(12):1589–1591
68. Wang G, Dunbrack RL (2005) PISCES: recent improvements to a PDB sequence culling server. Nucleic Acids Res 33(Suppl 2):W94–W98

Chapter 6

SPIDER2: A Package to Predict Secondary Structure, Accessible Surface Area, and Main-Chain Torsional Angles by Deep Neural Networks

Yuedong Yang, Rhys Heffernan, Kuldip Paliwal, James Lyons, Abdollah Dehzangi, Alok Sharma, Jihua Wang, Abdul Sattar, and Yaoqi Zhou

Abstract

Predicting one-dimensional structure properties has played an important role to improve prediction of protein three-dimensional structures and functions. The most commonly predicted properties are secondary structure and accessible surface area (ASA) representing local and nonlocal structural characteristics, respectively. Secondary structure prediction is further complemented by prediction of continuous main-chain torsional angles. Here we describe a newly developed method SPIDER2 that utilizes three iterations of deep learning neural networks to improve the prediction accuracy of several structural properties simultaneously. For an independent test set of 1199 proteins SPIDER2 achieves 82 % accuracy for secondary structure prediction, 0.76 for the correlation coefficient between predicted and actual solvent accessible surface area, 19° and 30° for mean absolute errors of backbone φ and ψ angles, respectively, and 8° and 32° for mean absolute errors of Cα-based θ and τ angles, respectively. The method provides state-of-the-art, all-in-one accurate prediction of local structure and solvent accessible surface area. The method is implemented, as a webserver along with a standalone package that are available in our website: http://sparks-lab.org.

Key words Secondary structure prediction, Solvent accessible surface area, Backbone torsion angles, Deep neural networks, C alpha-based angles

1 Introduction

With the rapid development of DNA sequencing techniques, there is a continuously increasing gap between the number of sequences available from genomic analysis and the number of structures and functions determined or annotated by expensive experimental techniques. It is highly desirable to develop theoretical methods to predict protein structures and functions from their one-dimensional

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_6,

sequences. However, methods for highly accurate prediction of protein three-dimensional structures (except homology modeling) are not yet available. This has significantly limited the ability to annotate protein functions based on their three-dimensional structures. As a result, predicted one-dimensional structural properties of proteins have often been utilized for predicting protein functions [1–4], their binding sites to other molecules [5–7], and other studies [8–11]. They have also been widely employed to improve protein structure prediction methods: both ab initio [12–14] and template-based techniques [15–18]. Thus any improvement in predicted one-dimensional structural properties will benefit protein structure and function modeling.

The most commonly predicted one-dimensional structural property of a protein is three-state secondary structure (helix, sheet, and coil). Secondary structure prediction accuracy without using homologous sequences in training has gradually been improved to above 81 % in recent years [19, 20], due to improved machine-learning algorithms, better features, and available larger training datasets.

An alternative to secondary structures is angle-based representation of backbone structure. Angle-based description such as torsion angles φ and ψ offers a continuous representation of local conformation [12], rather than discontinuous and somewhat arbitrary definition of three secondary-structure states. The advantage of angle-based representation leads to methods for predicting torsional angles φ and ψ [12, 21], and Cα-based angles [an angle between $C\alpha_{i-1} - C\alpha_i - C\alpha_{i+1}$ (θ) and a dihedral angle rotated about the $C\alpha_{i-1} - C\alpha^i$ bond (τ)] [22].

Another important one-dimensional structure property is solvent Accessible Surface Area (ASA) that measures exposure of amino acid residues of proteins to solvent, which is important for understanding and predicting protein structure, function, and interactions [23–26]. Earlier multistate prediction [23, 27, 28] has been gradually moved to continuous real value prediction [29–33].

In a recent study, we have developed SPIDER2, an iterative deep-learning neutral network, to predict all above-mentioned structural properties at the same time [34]. The iterative and cross-learning method achieved 82 % accuracy for secondary structure prediction, 0.76 for the correlation coefficient between predicted and actual solvent accessible surface area, 19° and 30° for mean absolute errors of backbone φ and ψ angles, respectively, and 8° and 32° for mean absolute errors of Cα-based θ and τ angles, respectively, for an independent test dataset of 1199 proteins. The resulting method provides state-of-the-art, all-in-one accurate prediction of local structure and solvent accessible surface area.

2 Algorithm

SPIDER2 server version was trained on a dataset of 5789 nonredundant (25 % cutoff), high resolution (<2.0 Å) structure by employing a three consecutive deep neural networks trained iteratively. In each iteration, we employed a deep neural network (DNN) consisting of three hidden layers with 150 hidden nodes in each layer. The weights were initialized by stacked sparse autoencoder [35] and then refined by standard back-propagation through fine-tuned supervised training [36, 37]. The learning rates for backward propagation were 1, 0.5, 0.2, and 0.05, respectively, with 30 epochs at each learning rate. The input layer for the DNN in the first iterative learning consists of 459 features (27 features per residue for a sliding window of 17 residues centered at the query residue). These 27 features include seven representative physical chemical properties parameters (steric parameter (graph shape index), hydrophobicity, volume, polarizability, isoelectric point, helix probability, and sheet probability properties of the amino acids), and 20 substitution probabilities obtained from 3 iterations searching by PSIBLAST [38]. All input features are normalized to the range of 0 to 1. For residues near the ends of a protein, the features of the amino acid residue at the current end of the protein were duplicated so that a full window could be used. Predicted outputs are 12 values of predicted probabilities for three secondary structure states, relative ASA, and sine and cosine of four angles θ, τ, ϕ, and ψ. The input layers for the DNN in the second and third iterative learning are 12 predicted values in the previous iteration plus 27 above-employed features per residue, that is, 663 features $[=(12+27)\times 17]$.

3 Web Server

The simplest way to use SPIDER2 is to submit a query sequence to our server at http://sparks-lab.org/yueyang/server/SPIDER2.

1. As shown in Fig. 1a, your protein sequence can be entered (or copy-pasted) in the FASTA format into the text area. Only one protein sequence is allowed each time. The sequence must contain 20 standard amino acids only. The first comment line in the FASTA format (“>” followed by the protein name) is employed to identify the name of the query protein. Without this line, the protein name will be set as “unknown” by default. The email address and target name in the webpage are optional. If you have a DNA/RNA sequence, you need first to convert them into a protein sequence (*see* **Note 1**).

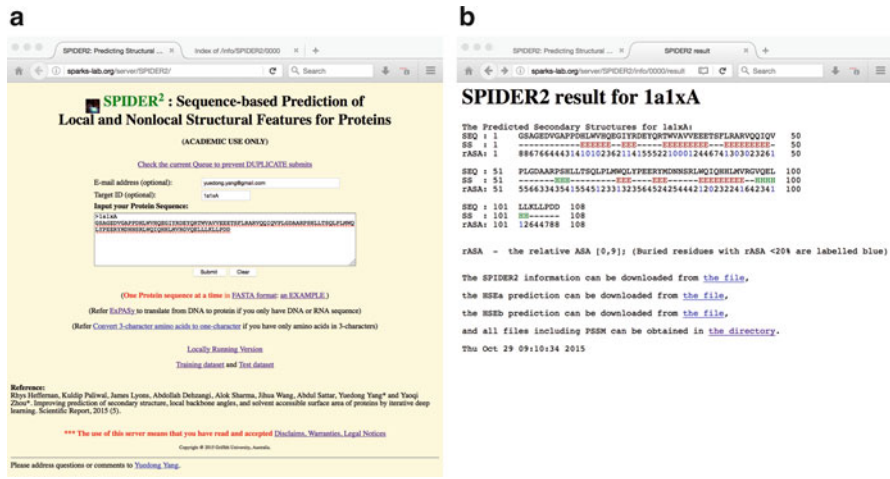


Fig. 1 The webserver input (**a**) and output screenshots (**b**) for example sequence"1a1xA.seq" by SPIDER2

2. By clicking the "submit" button, the job will be sent to a queue, and the webpage will be directed to a new page, where the "Click the link" points to a to-be-available result file. This webpage will be automatically refreshed every 60 s until the job is completed and the result is displayed on the web page.
3. Each prediction is usually completed within 10 min, but may take up to a few hours depending on how busy the server is and how long the protein chain is. If an email address is provided in submission, the link to the result webpage will be sent to the mailbox as soon as the prediction is finished. All prediction results are kept in the server for 1 month and automatically deleted afterwards.
4. If the users have their own Position Specific Substitution Matrix (PSSM) file for their query protein sequence, SPIDER2 prediction can be made by submitting the PSSM file to the server. Using an external PSSM file can skip the most time-consuming step of generating the evolution profile by PSI-BLAST, and the executive time reduce to a few seconds.
5. To save computing resources, please do not submit query sequences more than once. The status of your job can be found by clicking the link "Check the current Queue to prevent DUPLICATE submission" on the server webpage.
6. Figure 1b shows an example for the output webpage. Aligned lines started with "SEQ," "SS," and "rASA" represent query sequence, predicted secondary structure, and predicted relative accessible surface area, respectively. For SS, predicted coil, helix and sheet residues are represented by "–," red "E," and green "H," respectively. For rASA, the relative ASA is represented by 0–9 with "0" for up to 10% of its surface exposed and "9" for above 90% exposed. The residues of rASA less

than 20% (buried residues) are labeled in blue. Here, rASA is normalized by a residue-specific reference value (the ASA in the fully exposed state of a residue when connected by an ALA in each side). This output page does not contain predicted secondary structure probability, predicted angles, and actual real values of ASA. The complete prediction file "pro1.spd3" (*see* Subheading 4, **step 6** for explanation of the file) together with other intermediate files such as PSSM can be downloaded following the link in this output webpage.

4 Standalone Software

SPIDER2 is also available as a standalone software package. The program was designed to run in a Linux environment with python 2.7 and numpy version 1.4 or above. The input is a protein sequence in FASTA format, and outputs include predicted secondary structure, accessible surface area, main-chain torsional angles (phi/psi and theta/tau). The program can be installed in following steps.

1. Download the software package from our homepage with a shortcut link: http://sparks-lab.org/pmwiki/download/index.php?Download=yueyang/SPIDER2_local.tgz after entering your name and email address. This information will be used only for notification of future updates. You can fill in "none" if you prefer not to leave your information.
2. Unzip the package by command "tar zxvf SPIDER2_local.tgz" which creats the directory "SPIDER2_local" containing a "Readme" file and three subdirectories "dat," "ex," and "misc." The "dat" directory contains three npz files of trained parameters for three iterative neural networks, respectively, and the "misc" directory contains the program and auxiliary script files.
3. If BLAST or BLAST+ package is not installed in your computer, the software can be obtained from NCBI website. This program further requires correctly formatted nonredundant protein sequence databases, which can be downloaded from NCBI ftp://ftp.ncbi.nlm.nih.gov/blast/db (all files starting with "nr"). Until Oct 2015, the NR database contains a total of 40 files in 22GB before uncompressing. Alternatively, you can utilize a database by removing highly homology sequences, e.g., Uniref90 (*see* **Note 2**). This will speed up the calculation without making significant changes in prediction accuracy. This step can be skipped if you have prepared PSSM files (*see* **Note 3**).
4. SPIDER2 is called by the command "run_local.sh," followed by all sequence files in FASTA format. Here, one input file can contain a protein sequence only (*see* **Note 4**).

#	AA	SS	ASA	Phi	Psi	Theta(i-1=>i+1)	Tau(i-2=>i+1)	P(C)	P(E)	P(H)
1	G	C	63.4	-71.3	-166.0	112.3	-103.5	0.989	0.004	0.008
2	S	C	92.5	-84.7	72.1	107.8	-78.3	0.975	0.004	0.023
3	A	C	76.7	-80.7	32.2	104.6	109.9	0.920	0.016	0.053
4	G	C	52.6	79.7	-172.7	126.2	-92.0	0.912	0.059	0.028
5	E	C	129.5	-80.6	139.4	113.0	-52.6	0.890	0.080	0.021
6	D	C	92.6	-85.4	112.4	110.3	-100.7	0.898	0.064	0.016
7	V	C	64.7	-95.4	9.6	104.2	-179.9	0.907	0.055	0.022
8	G	C	33.9	80.7	-179.4	124.4	-127.1	0.907	0.067	0.028
9	A	C	56.5	-93.2	140.0	121.3	-59.6	0.990	0.008	0.004
10	P	C	53.5	-61.4	143.5	116.4	-80.1	0.994	0.004	0.003
11	P	C	24.1	-61.4	146.7	109.9	-102.5	0.976	0.015	0.010
12	D	C	59.0	-96.8	-13.5	101.9	-163.1	0.568	0.432	0.018
13	H	E	37.9	-141.7	140.3	131.4	26.3	0.045	0.955	0.001
14	L	E	4.1	-115.2	133.9	121.6	-155.4	0.091	0.910	0.007
15	W	E	33.5	-120.5	135.1	125.1	-162.8	0.017	0.983	0.003
16	V	E	3.4	-113.0	126.7	118.9	-154.5	0.014	0.983	0.002
17	H	E	40.0	-105.7	58.8	111.3	-170.3	0.103	0.873	0.016
18	Q	E	54.9	-137.5	142.4	129.6	69.2	0.393	0.611	0.004
19	E	C	120.2	-70.5	137.6	112.8	-98.1	0.917	0.061	0.019
20	G	C	18.7	111.5	85.9	113.4	-17.8	0.607	0.377	0.012
21	I	E	23.6	-110.1	140.1	123.5	135.2	0.280	0.699	0.008
22	Y	E	37.9	-118.7	137.3	125.7	-165.8	0.016	0.981	0.001
23	R	E	93.3	-117.8	136.9	124.3	-156.8	0.010	0.988	0.000
24	D	C	26.7	-76.4	165.6	118.7	-135.6	0.955	0.048	0.003
25	E	C	104.3	-66.0	-26.9	92.0	-95.4	0.954	0.016	0.023
26	Y	C	136.4	-94.5	4.7	98.2	52.3	0.996	0.001	0.002
27	Q	C	100.8	64.4	32.4	92.7	-112.7	0.966	0.022	0.009
28	R	C	65.9	-103.1	142.6	123.1	137.2	0.966	0.037	0.001
29	T	E	40.5	-102.2	135.7	119.5	-129.5	0.095	0.878	0.001
30	W	E	25.7	-123.0	143.6	129.4	-160.7	0.047	0.952	0.001

"1a1xA.spd3" 109 lines --0%-- 1,1 Top

Fig. 2 The partial prediction results by SPIDER2 for the example sequence "1a1xA.seq"

5. Results will be saved in an output file with extension "spd3." An example of output is shown in Fig. 2. The output file contains 11 columns that represent the residue index, residue type, predicted secondary structure type, ASA, φ, ψ, θ, τ, and probabilities as coil (C), sheet (E), and helix (H). The predicted secondary structure is the secondary structure type with the highest probability. The θ angle at residue index i is the angle between $C\alpha_{i-1} - C\alpha_i - C\alpha_{i+1}$, and τ is the dihedral angle formed by $C\alpha_{i-2} - C\alpha_{i-1} - C\alpha_i - C\alpha_{i+1}$. Three torsional angles φ, ψ, and τ range from −180 to 180°, and angle θ mostly ranges between 70 and 180°.
6. In addition, the package includes one program "pred_nopssm.py" that makes prediction without using the PSSM from PSI-BLAST. Instead, the profile is replaced by the BLOSUM62 substitution matrix. This replacement allows a fast calculation

at a lower accuracy (For example, secondary structure accuracy at 68.9%, compared to 81.8% by using PSI-BLAST profile). This may be useful for large-scale calculations in genome level. However, it should be noted that all parameters were not optimized for the evolution-profile free prediction, and the development of a specific predictor by using sequence only is in progress.

5 Notes

1. The query sequence must be a protein sequence in the FASTA format. The gene in the DNA/RNA sequence has to be converted to the sequence of amino acids first. This conversion can be made by using http://web.expasy.org/translate or any other tools. Nonstandard amino acids (e.g., X) must be removed, prior to the use of SPIDER2.
2. The package employs PSI-BLAST to generate PSSM generated by scanning NR database. Alternatively, you can employ the sequence database uniref90 that can be downloaded from ftp://ftp.uniprot.org/pub/databases/uniprot/uniref/uniref90/uniref90.fasta.gz. This database can be converted to BLAST-readable format by the command "gunzip -c uniref90.fasta.gz | ~/aspen/software/ncbi-blast-2.2.30+/bin/makeblastdb -in - -dbtype prot -parse_seqids -out uniref90 -title uniref90." This operation skips the step of unzipping the large database.
3. For users with their own PSSM files, they can obtain predictions by utilizing the script "pred_pssm.py" followed by PSSM file names. This command will skip running PSI-BLAST and prediction can be finished in a few seconds.
4. If your sequence file contains more than one protein sequence, you can use the script file "splitseq.py" to split your sequence files to many files, and each file will be named according to protein names in the FASTA file.

Acknowledgements

This work was supported in part by National Health and Medical Research Council (1059775) of Australia and Australian Research Council's Linkage Infrastructure, Equipment and Facilities funding scheme (project number LE150100161), the Taishan Scholars Program of Shandong province of China, National Natural Science Foundation of China (61540025) to Y.Z. and National Natural Science Foundation of China (61271378) to Y.Y. and J.W. We also gratefully acknowledge the support of the Griffith University eRe-

search Services Team and the use of the High Performance Computing Cluster "Gowonda" to complete this research. This research/project has also been undertaken with the aid of the research cloud resources provided by the Queensland Cyber Infrastructure Foundation (QCIF).

References

1. Lou W, Wang X, Chen F, Chen Y, Jiang B, Zhang H (2014) Sequence based prediction of DNA-binding proteins based on hybrid feature selection using random forest and Gaussian naïve Bayes. PLoS One 9(1)
2. Zhao H, Yang Y, Zhou Y (2011) Highly accurate and high-resolution function prediction of RNA binding proteins by fold recognition and binding affinity prediction. RNA Biol 8(6):988–996. doi:10.4161/rna.8.6.17813
3. Zhao H, Yang Y, von Itzstein M, Zhou Y (2014) Carbohydrate-binding protein identification by coupling structural similarity searching with binding affinity prediction. J Comput Chem 35(30):2177–2183
4. Zhao H, Wang J, Zhou Y, Yang Y (2014) Predicting DNA-binding proteins and binding residues by complex structure prediction and application to human proteome. PLoS One 9(5):e96694
5. Zhang T, Zhang H, Chen K, Ruan J, Shen S, Kurgan L (2010) Analysis and prediction of RNA-binding residues using sequence, evolutionary conservation, and predicted secondary structure and solvent accessibility. Curr Protein Peptide Sci 11(7):609–628
6. Zhang Z, Li Y, Lin B, Schroeder M, Huang B (2011) Identification of cavities on protein surface using multiple computational approaches for drug binding site prediction. Bioinformatics 27(15):2083–2088
7. Bradford JR, Westhead DR (2005) Improved prediction of protein–protein binding sites using a support vector machines approach. Bioinformatics 21(8):1487–1494
8. Folkman L, Yang Y, Li Z, Stantic B, Sattar A, Mort M, Cooper DN, Liu Y, Zhou Y (2015) DDIG-in: detecting disease-causing genetic variations due to frameshifting indels and nonsense mutations employing sequence and structural properties at nucleotide and protein levels. Bioinformatics 31(10):1599–1606
9. Zheng W, Zhang C, Hanlon M, Ruan J, Gao J (2014) An ensemble method for prediction of conformational B-cell epitopes from antigen sequences. Comput Biol Chem 49:51–58
10. Zhao H, Yang Y, Lin H, Zhang X, Mort M, Cooper DN, Liu Y, Zhou Y (2013) DDIG-in: discriminating between disease-associated and neutral non-frameshifting micro-indels, Genome Biology, 14, R43
11. Lyons J, Dehzangi A, Heffernan R, Yang Y, Zhou Y, Sharma A, Paliwal K (2015) Advancing the accuracy of protein fold recognition by utilizing profiles from Hidden Markov models, IEEE Transactions on NanoBioscience, 14, 761–772
12. Faraggi E, Yang Y, Zhang S, Zhou Y (2009) Predicting continuous local structure and the effect of its substitution for secondary structure in fragment-free protein structure prediction. Structure 17(11):1515–1527. doi:10.1016/j.str.2009.09.006
13. Bradley P, Chivian D, Meiler J, Misura KM, Rohl CA, Schief WR, Wedemeyer WJ, Schueler-Furman O, Murphy P, Schonbrun J, Strauss CE, Baker D (2003) Rosetta predictions in CASP5: successes, failures, and prospects for complete automation. Proteins 53(Suppl 6):457–468. doi:10.1002/prot.10552
14. Handl J, Knowles J, Vernon R, Baker D, Lovell SC (2012) The dual role of fragments in fragment-assembly methods for de novo protein structure prediction. Proteins 80(2):490–504
15. Yang Y, Faraggi E, Zhao H, Zhou Y (2011) Improving protein fold recognition and template-based modeling by employing probabilistic-based matching between predicted one-dimensional structural properties of query and corresponding native properties of templates. Bioinformatics 27(15):2076–2082. doi:10.1093/bioinformatics/btr350
16. Zhang Y (2009) I-TASSER: fully automated protein structure prediction in CASP8. Proteins 77(S9):100–113
17. Remmert M, Biegert A, Hauser A, Söding J (2011) HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. Nat Meth 9(2):173–175
18. Cheng J, Wang Z, Tegge AN, Eickholt J (2009) Prediction of global and local quality of CASP8 models by MULTICOM series. Proteins 77(S9):181–184
19. Faraggi E, Zhang T, Yang Y, Kurgan L, Zhou Y (2011) SPINE X: improving protein second-

ary structure prediction by multi-step learning coupled with prediction of solvent accessible surface area and backbone torsion angles. J Comput Chem 33:259–263

20. Yaseen A, Li YH (2014) Context-based features enhance protein secondary structure prediction accuracy. J Chem Inf Model 54(3):992–1002. doi:10.1021/Ci400647u
21. Wu S, Zhang Y (2008) ANGLOR: a composite machine-learning algorithm for protein backbone torsion angle prediction. PLoS One 3(10):e3400
22. Lyons J, Dehzangi A, Heffernan R, Sharma A, Paliwal K, Sattar A, Zhou Y, Yang Y (2014) Predicting backbone Calpha angles and dihedrals from protein sequences by stacked sparse auto-encoder deep neural network. J Comput Chem 35(28):2040–2046. doi:10.1002/jcc.23718
23. Rost B, Sander C (1994) Conservation and prediction of solvent accessibility in protein families. Proteins 20(3):216–226
24. Gilis D, Rooman M (1997) Predicting protein stability changes upon mutation using database-derived potentials: solvent accessibility determines the importance of local versus non-local interactions along the sequence. J Mol Biol 272(2):276–290
25. Tuncbag N, Gursoy A, Keskin O (2009) Identification of computational hot spots in protein interfaces: combining solvent accessibility and inter-residue potentials improves the accuracy. Bioinformatics 25(12):1513–1520
26. Lee B, Richards FM (1971) The interpretation of protein structures: estimation of static accessibility. J Mol Biol 55(3):379–400
27. Holbrook SR, Muskal SM, Kim SH (1990) Predicting surface exposure of amino acids from protein sequence. Protein Eng 3(8):659–665
28. Pollastri G, Baldi P, Fariselli P, Casadio R (2002) Prediction of coordination number and relative solvent accessibility in proteins. Proteins 47(2):142–153
29. Dor O, Zhou Y (2007) Real-SPINE: an integrated system of neural networks for real-value prediction of protein structural properties. Proteins 68(1):76–81
30. Garg A, Kaur H, Raghava GP (2005) Real value prediction of solvent accessibility in proteins using multiple sequence alignment and secondary structure. Proteins 61(2):318–324. doi:10.1002/prot.20630
31. Yuan Z, Huang B (2004) Prediction of protein accessible surface areas by support vector regression. Proteins 57(3):558–564. doi:10.1002/prot.20234
32. Ahmad S, Gromiha MM, Sarai A (2003) Real value prediction of solvent accessibility from amino acid sequence. Proteins 50(4):629–635. doi:10.1002/prot.10328
33. Adamczak R, Porollo A, Meller J (2004) Accurate prediction of solvent accessibility using neural networks-based regression. Proteins 56(4):753–767. doi:10.1002/prot.20176
34. Heffernan R, Paliwal K, Lyons J, Dehzangi A, Sharma A, Wang J, Sattar A, Yang Y, Zhou Y (2015) Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning. Sci Rep 5:11476
35. Bengio Y, Lamblin P, Popovici D, Larochelle H (2007) Greedy layer-wise training of deep networks. Adv Neural Inform Process Syst 19:153
36. Hinton GE (2007) Learning multiple a layers of representation. Trends Cogn Sci 11(10):428–434. doi:10.1016/J.Tics.2007.09.004
37. Bengio Y (2009) Learning deep architectures for AI. Found Trends Mach Learn 2(1):1–127
38. Altschul SF, Madden TL, Schaffer AA, Zhang JH, Zhang Z, Miller W, Lipman DJ (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25(17):3389–3402

Chapter 7

Backbone Dihedral Angle Prediction

Olav Zimmermann

Abstract

More than two decades of research have enabled dihedral angle predictions at an accuracy that makes them an interesting alternative or supplement to secondary structure prediction that provides detailed local structure information for every residue of a protein. The evolution of dihedral angle prediction methods is closely linked to advancements in machine learning and other relevant technologies. Consequently recent improvements in large-scale training of deep neural networks have led to the best method currently available, which achieves a mean absolute error of 19° for phi, and 30° for psi. This performance opens interesting perspectives for the application of dihedral angle prediction in the comparison, prediction, and design of protein structures.

Key words Dihedral angles, Torsional angles, Structure prediction, Machine learning

1 Introduction

For more than 50 years dihedral angles have served as compact descriptions of a protein backbone's local topology. Following the seminal analysis of Ramachandran et al. in 1963 [1] dihedral angles have been used to describe commonalities and differences between protein structures. It may therefore seem surprising that methods for the prediction of secondary structure predate those for dihedral angles by more than two decades and that dihedral angle prediction is still much less well known and applied.

In part this may be due to the fact that humans need a graphical representation to comprehend the topology of a protein. Scientists have got used to the aesthetically pleasing cartoon renderings of proteins with their easily recognizable α-helices and β-sheets and thereby tend to underestimate the importance of the large remainder flanking these elements. Hidden away in the large pseudo class "coil," however, is a plethora of recurring structure motifs that have evolved to shape the topology of the protein chains. This set of motifs is essential for protein folding and endows each protein with its individual set of static and dynamic features.

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_7,

Another reason for the relatively late appearance of methods to predict dihedral angles may be that prediction of secondary structure is a classification task, whereas dihedral angle prediction is a regression task which, in high dimensional spaces, is deemed to be more difficult than classification. This hypothesis is supported by the fact that early methods for dihedral prediction used classification approaches.

Finally many researchers may have used dihedral angle prediction algorithms without noticing as some of these methods are part of larger structure prediction pipelines.

After an introduction on Ramachandran statistics and measures for prediction performance, this mini review will try to summarize the evolution of different approaches to dihedral angle prediction, with a focus on the prediction of the "Ramachandran angles," phi (φ) and psi (ψ) from sequence information alone. In a discussion section the performance improvements will be linked to parallel advancements in several research and technology fields. Important aspects from a user perspective regarding availability and applications will be mentioned. Finally we will sketch a possible future of research in dihedral angle prediction with particular emphasis on open questions, new applications, and possible synergisms to simulation methods.

1.1 Definition of Backbone Dihedral Angles

Dihedral angles, also often called torsion angles, are defined with respect to three consecutive bonds connecting four atoms. By IUPAC convention [2] dihedral angles are positive for the clockwise difference from the bond preceding the bond that defines the dihedral angle to the bond following it.

The backbone conformation of each protein residue is determined by three dihedral angles: phi (φ), denoting the rotation around the N-Cα bond (atoms C′-N-C_α-C′), psi (ψ) denoting the rotation around the Cα-C′ bond (atoms N-C_α-C′-N), and omega (ω) denoting rotation around the C′-N peptide bond (atoms C_α-C′-N-C_α). Due to its partial double bond character the peptide bond C′-N connecting two residues is approximately planar. As the trans-configuration of the peptide bond with the two side chains spaced further apart is more stable, most ω-angles in proteins are close to 180°. A Proline following a residue, however, stabilizes the cis-configuration ($\omega \sim 0°$). Recent reevaluations of the PDB data hint that the number of cis peptide bonds in the experimentally known protein structures is currently underestimated [3].

1.2 Dihedral Angle Distributions

Taking into account that bond lengths and bond angles in proteins have only very limited flexibility, and that the peptide bond angle ω is largely fixed, it is evident that the two remaining dihedral angles φ and ψ contain the majority of information needed to reconstruct a protein backbone structure from its amino acid sequence. In 1963 Ramachandran and coworkers provided the first analysis of the dihedral angle distribution in proteins and calculated favored and disallowed regions based on steric

constraints and concluded that almost 3/4 of the φ–ψ angle space is unavailable to peptide backbone structures [1, 4]. They also noted the profoundly different dihedral angle distributions for Proline and Glycine (Fig. 1b, c). The informative 2d-plot of the φ-ψ angle space they used, the famous Ramachandran plot, has become a hallmark of protein structure analysis (Fig. 1).

Forty years later a much grown Protein Data Bank [5] allowed Lovell et al. to perform more accurate statistics. They derived surprisingly sharp bounds for the φ–ψ distributions and observed a markedly different distribution for amino acid residues followed by a Proline (Fig. 1c)] [6].

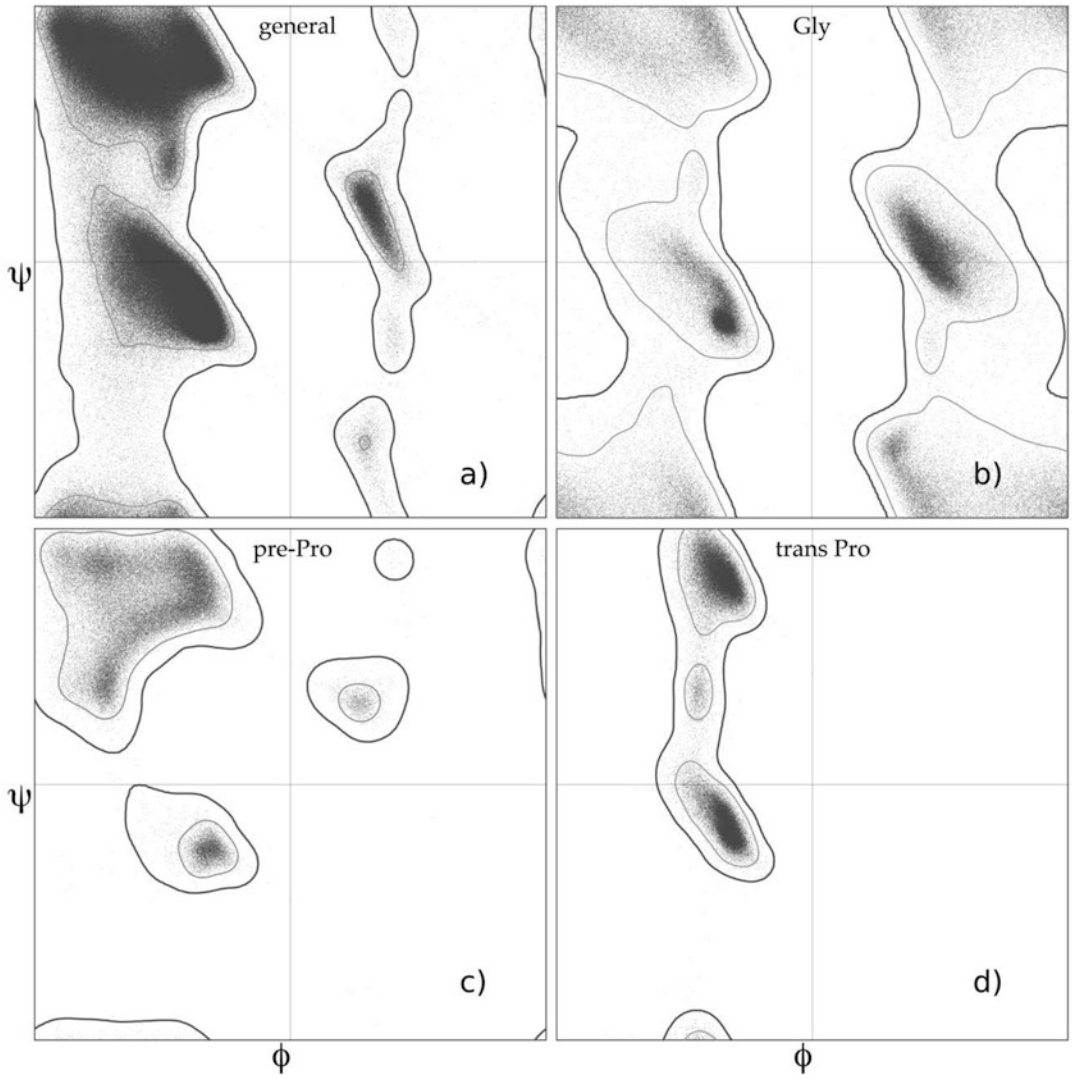


Fig. 1 Backbone dihedral angle distributions: (**a**) all residues except Pro/Gly, (**b**) Gly, (**c**) pre-Pro, (**d**) trans-Pro. from: "6 Ramachandran plots of wwPDB validation" by Dcrjsr—Own work. Licensed under CC BY 4.0 via Wikimedia Commons, edited: plots of cis-Pro and Ile/Val cut away

The φ/ψ distribution of the amino acid preceding a Proline deviates considerably from their typical distribution due to the restrictions imposed by the secondary amino group of Proline (Fig. 1c). While less pronounced than for Proline and Glycine all amino acids show individual differences in their backbone dihedral angle distributions. A very detailed account of these preferences can be found in [7] that also highlights the dependencies between individual subregions of the Ramachandran plot and secondary structure elements. The effects of the immediately adjacent residues in loops were quantified by Jordan's and Dunbrack's groups for all amino acids [8] whereas Kihara studied the impact of long-range contacts [9].

On average, i.e. on a sufficiently representative subset of the PDB, a good prediction algorithm should be able to reproduce the Ramachandran plots of Lovell et al. Ramachandran plots can therefore be used to assess dihedral angle prediction algorithms or backbone potentials for the presence of systematic errors or bias.

1.3 Correspondence to Local Chain Topology

Secondary structure elements feature regular hydrogen bond patterns which constrain their dihedral angles to characteristic values. While the dihedral angles in β-sheets and α-helices (also for the Polyproline II helix) repeat for each single residue, other nonrepetitive dihedral angle patterns are characteristic for the different types of β-turns [10–13] and other topological motifs [14–16]. These structure motifs are important for the topology of the protein chain but often remain uncharacterized in secondary structure analysis. Dihedral angle patterns thereby extend secondary structure descriptions by providing both enhanced coverage of the protein chain and more fine-grained structure characterization.

2 Prediction of Dihedral Angles

The backbone dihedral space available to a particular amino acid in a protein chain is restricted not only by steric constraints of the local backbone geometry as it had been the main finding of Ramachandran et al. but also by the remainder of the residue's three-dimensional neighborhood. Strong additional constraints are imposed by the formation of hydrogen bonds and the steric constraints due to side chain interactions between adjacent residues.

This indicates that dihedral angle preferences are mainly determined by the identity of the respective residue and its local sequence context. But this in reverse suggests that local sequence information can be used to infer the backbone dihedrals of a residue.

2.1 Performance Measures

Different methods have different prediction targets and depending on the prediction target different performance measures are used. Some methods are designed to predict the cluster region of the

Ramachandran plot that a residue maps while others predict the structural motif class, i.e., the closest cluster center in a multivariate dihedral space. These methods perform a classification and hence the predictions are either right or wrong. For binary prediction there are four possible outcomes.

Typical performance measures for binary classification are accuracy and Matthews correlation coefficient (MCC) [17], the latter being more robust in cases of class imbalance that are common for most dihedral angle region definitions.

$$\text{Accuracy} := (\text{TP}+\text{TN})/\text{N} = (\text{TP}+\text{TN})/(\text{TP}+\text{TN}+\text{FP}+\text{FN})$$

$$\text{MCC} := \frac{\text{TP}\times\text{TN}+\text{FP}\times\text{FN}}{\sqrt{(\text{TP}+\text{FP})\times(\text{TP}+\text{FN})\times(\text{TN}+\text{FP})\times(\text{TN}+\text{FN})}}$$

where TP, TN, FP, FN are defined in Table 1 and N is the total number of predictions.

For more than two classes accuracy can be easily extended as it is simply the fraction of all correct predictions and typically reported with an index indicating the number of classes, e.g., Q_3 is often reported as a performance measure for secondary structure classification into the three classes: helix, sheet, and coil. In dihedral angle prediction a further definition of Q_n in use is the fraction of predictions that are within n degree of the experimental value. This should more clearly be labeled as Q_{n° but the degree sign is often omitted.

Extension of MCC to multiclass settings is less straightforward [18] and has not yet been used to report performance of classifiers for dihedral angle regions or structural motifs. Some studies show the confusion matrix that tabulates the counts for all pairs of predicted and observed classes. This matrix allows identification of classes that often get mixed up by the respective prediction method.

Methods directly predicting the real value of a dihedral angle are regression methods. Typically the Pearson correlation coefficient (PCC) is used to evaluate linear regression and it has been used

Table 1
Possible outcomes of binary classification

		Observation	
		True	**False**
Prediction	True	True positive (TP)	False positive (FP)
	False	False negative (FN)	True negative (TN)

to report performance of dihedral angle regression. However, due to the cyclic nature of angles and the very uneven distribution of the angular values the mean absolute error (MAE) is a more robust measure of performance. It is defined as the average of the absolute difference between the experimentally observed angle x and the predicted angle y.

$$\mathrm{PCC} := \frac{\sum_{i=1}^{N}(x_i - \bar{x}) \times (y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2} \times \sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}}$$

where x-bar and y-bar are the mean of the observed and predicted angles, respectively, and N is the total number of angles.

$$\mathrm{MAE} := \frac{1}{N}\sum_{i=1}^{N} \min\left(|y_i - x_i|, |360° - (y_i - x_i)|\right)$$

where x and y are observed and predicted angles, respectively, and N is the total number of angles.

In addition many methods map their primary prediction target to secondary structure prediction to allow for comparison with dedicated secondary structure prediction methods. In these cases the respective publications may report typical performance measures for secondary structure prediction like Q_3 (helix, sheet, coil) or Segment overlap (SOV) [19, 20].

2.2 Prediction Methods

In the following we will try to sketch the evolution of dihedral prediction methods and highlight interesting ideas and findings that have been influential to later methods. Most methods are based on supervised machine learning algorithms, such as Hidden Markov Models (HMM) [21], Conditional Random Fields (CRF) [22], Support Vector Machines (SVM) [23], and Artificial Neural Networks (ANN) [24]. As doing justice to the importance of these methods would require a review of its own, we will forego to give introductions into these methods and instead refer to the numerous excellent introductions existing, e.g., [25].

The primary input for most dihedral prediction methods is a position-specific scoring matrix (PSSM) typically obtained by running PSI-BLAST [26] against a nonredundant sequence database. The PSSM represents the amino acid preferences of each residue position and reflects the averaged local environment. Thereby it has the potential to provide indirect information on influences of long-range interaction if the quality of the underlying multiple sequence alignment is good, and the variation of the local environment is not too large.

3 Dihedral Angle Region Prediction

Several early methods rephrased the dihedral angle inference as a classification problem [27–31]. Instead of predicting a particular point on the Ramachandran plot, they provided a prediction in which region of the φ–ψ space a residue's conformation would map. Depending on the number and size of the dihedral angle regions defined in this mapping, a correct prediction carries more or less information. For obvious reasons prediction of fewer classes is easier and accuracy is therefore likely to be higher for coarser classification schemes.

4 Structure Motif Prediction

As the torsion angle conformation of a residue is strongly dependent on the neighborhood of that residue, a logical extension to dihedral region prediction was the prediction of short sequences of dihedral angles also called structure motifs, structural alphabets, or shape strings. The I-Sites method appeared in 1998 when the PDB had grown to more than 7000 proteins, thereby providing sufficient data for the analysis of multivariate distributions of dihedral angles. Using a nonredundant set of 417 aligned sequence families with at least one known structure representative Bystroff and Baker performed iterative sequence profile searches with structure constraints and clustering to obtain 82 clusters of sequence profiles of 3–19 residues in length, the I-Sites library [15]. These clusters were grouped into 13 recurring structure motifs among them some that had not been described before. Prediction was performed by sequence profile comparison to the 82 cluster profiles. While the I-Sites method could provide more detailed prediction for loop regions, its performance for three-state prediction ($Q_3 = 64\%$) was inferior to dedicated secondary structure prediction methods such as PHD [32].

Subsequently Bystroff et al. merged the I-Sites library by representing the I-Sites library as a set of HMM models that could encode overlaps between the structure motifs in a more compact form and also could represent the neighbor correlations in the sequence-, dihedral angle-, and structure space [16]. This method called HMMSTR improved the accuracy of secondary structure prediction to $Q_3 = 74.3\%$, comparable to then available dedicated secondary structure prediction methods. Different clustering methods led to the development of a number of structural alphabets with motif lengths of 2–5 residues [33–35]. A comparison of their performance in fold recognition indicated that prediction of fine-grained structural alphabets is more informative than predicted three-state secondary structure [34]. Several methods for

prediction of structural alphabets followed [36–38], and the structure motif predictions were often mapped to provide predictions of secondary structure [37] or dihedral angles [38]. A recently developed fold recognition method based on LOCUSTRA was found to be more sensitive than HHsearch [39] in finding good templates for protein structure modeling [40].

5 Real Value Dihedral Angle Prediction

The first published performance value on a real value prediction of a dihedral angle was a by-product of dihedral angle enhanced secondary structure prediction in the DESTRUCT approach published by Wood and Hirst in 2005 [41] who reported a moderate Pearson's correlation coefficient (PCC) of 0.47 for ψ. The method was based on several generations of cascade-correlation networks that did not only use the PSSMs of 513 proteins but also added the output of a secondary structure prediction network to the input of the ψ-angle prediction network of the next generation and vice versa.

The first method dedicated to real value dihedral angle prediction was Real-SPINE by Zhou and coworkers [42] which was based on training a backpropagation neural network with 200 hidden units and a much larger training set (2640 chains) than used in DESTRUCT. Inputs were PSI-BLAST [26] generated PSSMs and SS predictions from SPINE. Real-SPINE like DESTRUCT provided predictions for ψ only. ψ is deemed more informative than φ due to the fact that α-helices and β-sheets share similar φ-values. They reported an improved PCC of 0.62; and a MAE of 0.15 = 54°, 9° lower than without predicted SS, thereby emphasizing the large effect of this additional input. Simply shifting the cut point that is used to map the circular ψ-angle space to a linear space from 0° (or +/− 180° as in other studies) to −100° and using a consensus of 5 ANN models lead to marked improvements. Adding also a predictor for φ they achieved an MAE of 38.2° for ψ, and 24.8° for φ [43]. A year later the group demonstrated that using a guiding technique for the NN weights and adding a second hidden layer could reduce the error even further (MAE = 36.4° for ψ, and 22.1° for φ) [44].

The Real-SPINE ANNs had also independently been trained to predict solvent accessibility but this information had not been used in the prediction of ψ. ANGLOR from Wu and Zhang in contrast used such information on predicted solvent accessibility in addition to the previous inputs [45]. Their study also compared the performance of SVM and ANNs for dihedral angle prediction on the same training input. As a third contender they used PSIPRED [46] by mapping the three predicted secondary structure classes to their average dihedral angle values. Interestingly

they found that for ψ the SVM-based predictor had a 10% lower MAE than the ANN predictor, while the opposite was true for φ. The performance of ANGLOR (MAE = 28.2° for φ, and 46.4° for ψ) compared favorably to the translated dihedral angles from PSIPRED (MAE = 30.4 ° for φ, and 49.6° for ψ) which constitutes a nontrivial baseline and a lower bound on the amount of dihedral angle information contained in secondary structure predictions. The same study found that predictor performance is correlated to the dihedral angle entropy which is higher for ψ than for φ and particularly high for the small amino acids Glycine and Asparagine that frequently occur in flexible loop regions.

Later Song et al. published an approach very similar but with two layers and additional inputs containing global sequence features [47]. Training the method on the same 500 proteins as ANGLOR their approach showed slightly better performance in particular for ψ (MAE = 27.8° for φ, and 44.6° for ψ).

Having observed that real value prediction of dihedral angles often led to angle conformations in the forbidden region of the Ramachandran plot, the Real-SPINE authors refined their approach by splitting the prediction process into two phases [48]. Using the fact that each angle has a two-peaked distribution a first ANN predicts which of the two peaks is closer. A second ANN then predicts the real value deviation from the peak center. This approach (SPINE X) already led to some improvement (MAE 35.2° for ψ) but a larger performance increase was achieved by adding a Conditional Random Field (CRF) [22]. The resulting method, called SPINE XI, used the predicted values for both φ and ψ as additional inputs to the sequence and the predicted secondary structure information, and could thereby learn the correlations between the dihedrals, leading to a MAE of 33.4° for ψ.

As a successor of their ANN-based method DESTRUCT, Kontouris and Hirst built their DISSpred method based on two sets of SVMs that classified secondary structure and dihedral angle regions respectively [49]. They trained the SVMs on the same 513 proteins they had used for training DESTRUCT and also used a similar iterative training approach where predicted secondary structure information from the previous SVM generation was used for training the dihedral angle region SVMs and vice versa. Two generations of such iterative training and clustering into seven dihedral regions proved to be optimal for dihedral angle prediction and lead to MAEs of 25.1° for φ and 38.5° for ψ.

Due to computational and algorithmic improvements, training of deep ANNs with several hidden layers has become feasible in recent years and deep learning approaches have surged in popularity. In 2014, Lyons et al. published SPIDER, which featured a deep auto-encoder network with three hidden layers of 150 nodes each and was trained on a record-size training set of 4590 proteins [50]. In contrast to previous methods this approach focused on

prediction of the angles between and around C_α–C_α pseudo bonds. Due to the fixed peptide bond these Cα-Cα pseudo bonds have a uniform length of approximately 3.8 Å. SPIDER predicts real values for the theta angle between the $C_{\alpha,}\, i_{-1}$–$C_{\alpha,}\, i$ and $C_{\alpha,}\, i$–$C_{\alpha,}\, i_{+1}$ pseudo bonds and the dihedral angle tau that measures rotation around the $C_{\alpha,}\, i$–$C_{\alpha,}\, i_{+1}$ pseudo bond. They obtained MAEs of 8.6° for θ and 33.6° for τ, which they compared to the values obtained from mapping θ and τ from the φ and ψ predictions of Spider X and found an improvement of approximately 10%. Substituting the deep learning architecture by an ANN with only one hidden layer had a small impact (MAE 8.8° for θ and 34.1° for τ), likewise the encoding of the angles as sine and cosine lowered MAE 2° compared to the two-stage approach of SPINE X that first predicts the distribution peak and then the deviation from that peak.

Zhou and coworkers recently improved SPIDER by iterative learning [51]. Using three separate deep ANNs for prediction of secondary structure, solvent accessibility, and dihedral angles the prediction output of all three networks is provided as input to each network of the following generation. With respect to secondary structure prediction, they find three generations to be optimal. This method, called SPIDER-2, simultaneously predicts the dihedral angles φ, ψ, and τ, the angle θ, as well as secondary structure and solvent accessibility. It reports the best prediction performance to date. MAEs are 19.2° for φ, 29.9° for ψ, 8.0° for θ, and 32.2° for τ. To put these values in perspective the authors compared the predictions for ψ and τ of SPIDER-2 to those from models of the CASP11 structure prediction contest. The MAE values for ψ were 14% lower and those for τ 10% lower than the respective best methods in CASP11 (Baker-Rosetta, Zhang-Server). Perhaps the most important result is that the secondary structure prediction of SPIDER-2 on independent test cases and a CASP-11 dataset was superior to the state-of-the-art secondary structure prediction programs PSIPRED [46] and SCORPION [52].

It is easy to foresee that the performance of structure prediction pipelines will improve by adopting methods like SPIDER-2, as well as future methods for dihedral angle prediction that build on the knowledge gathered in more than a decade of research in this area.

Table 2 lists those methods that predict real values of dihedral angles.

6 Discussion

6.1 Performance Comparison

There are many different ways to assess performance of dihedral angle prediction methods and hence meaningful performance comparisons between different algorithms are difficult. Cases where published performance values between two studies can be

Table 2
Methods predicting real value dihedral angles

Name	Year	Method	MAE φ/ψ	Availability	Reference
DESTRUCT	2005	NN	n.d./n.d.		[41]
Real-SPINE	2007	NN	n.d./54°		[42]
ANGLOR	2008	SVM, NN	28.2°/46.4°	Web, download	[45]
Real-SPINE 2.0	2008	NN	24.8°/38.2°		[43]
Real-SPINE 3.0	2009	NN	22.1°/36.4°		[44]
SPINE X	2009	NN	n.d./35.2°	Web, download	[48]
SPINE XI	2009	NN + CRF	n.d./33.4°	Web, download	[48]
DISSpred	2009	SVM	25.1°/38.5°	Web	[49]
TANGLE	2012	SVM	27.8°/44.6°	Web	[47]
SPIDER	2014	Deep NN	n.d./n.d.		[50]
SPIDER-2	2015	Deep NN	19.2°/29.9°	Web, download	[51]

directly compared are rare and in most cases are from the same authors. It is also important whether results are reported from cross validation or independent test sets.

Dedicated evaluation studies are the preferred way to judge performance but are rare as well. They need to apply the same performance measures using the same test datasets while ensuring that these have no overlap with the training datasets of the methods compared. Sometimes the latter condition is difficult to fulfill as not all methods list the PDB identifiers of their training sets.

The only such study we are aware of in the area of dihedral angle prediction is a recent performance comparison between TANGLOR, TANGLE, and SPINE X [53]. They concluded that the deep learning approach of SPINE X provides significantly more accurate predictions than the other methods. A part of this performance advantage is probably due to the large datasets that can be used for training deep ANNs. Such data set sizes are not feasible for training standard nonlinear SVMs. A study of Hovmöller and coworkers estimated the impact of larger training sets for their method to be on the order of 1 % improvement per doubling of the nonredundant training set size in both secondary structure prediction accuracy (Q_3) and eight-class structure motif prediction [54].

6.2 Determinants for Improved Prediction Methods

The prediction performance improvements that have been achieved in the last 15 years in the area of dihedral angle prediction reflect advancements in several areas of science and technology.

PDB: Much of the knowledge gained on local and nonlocal interactions of the peptide backbone atoms and its effect on

sequence-structure relationships would have been impossible without accurate statistical analysis of experimental protein structures. The continuous growth of the Protein Data Bank [5] was therefore of prime importance to generate accurate statistics of sequence-structure correlations but also for the availability of increasingly larger nonredundant datasets for the supervised training of the prediction models.

Machine learning: The foundation of most modern prediction methods is machine learning algorithms. Predictors of local structure therefore either employ some method dedicated to supervised sequential learning such as HMM or CRF, or they map the sequential information to a fixed size vector representation and use standard supervised learning methods such as SVM or ANN. While HMMs [21] provide a way to learn the joint probability distribution of features and labels and have been successfully employed for sequence labeling problems such as the prediction of local dihedral states [16], they lack the ability to learn correlations between nonadjacent labels. This problem was partially overcome by the introduction of CRFs [22] that in contrast to HMMs provide a model of the label probability conditioned on the previous label and the input features. Algorithmic improvements have made the global training of CRFs computationally less expensive [55].

Standard supervised learning algorithms assume independent and identically distributed samples which typically imply input feature vectors of a fixed size. Already in 1988 a secondary structure prediction method by Qian and Seijnowski [56] used a sliding window that mapped the protein sequence information to a fixed size vector and thereby enabled them to use ANNs, which recently had become popular [24]. The same technique is used in approaches that use SVMs [23]. Today SVMs and ANNs are the dominating machine learning approaches. For each of them many different varieties have been developed and both have their advantages and disadvantages. SVMs are global regularized optimizers and hence do neither suffer from local minima nor overfitting like standard ANNs. On the other hand the training time for ANNs is linear with respect to the number of training samples whereas SVMs with nonlinear kernels have higher computational complexity, and current parallel implementations provide only moderate scaling efficiency, thereby limiting the training set size that SVMs can handle. In fact most of the SVM-based methods described above have used not more than 80000 training samples in contrast to the ANN-based methods that can be trained on todays large nonredundant protein data sets with up to one million training samples [50, 51]. Furthermore ANNs can have arbitrarily complex output types, such as tuples of different dihedral angles combined with secondary structure categories and solvent accessibility [51] whereas SVMs have only recently ushered into the realm of structured output prediction [57]. The renaissance of ANNs, in particular deep

learning, can be largely attributed to recent algorithmic advances. Groundbreaking work by Hinton and coworkers has finally prevented overfitting and accelerated the training of deep ANNs to an extent that made large-scale deep learning practical [58, 59].

High performance computing: Despite highly tuned algorithms for efficient learning the training of complex predictive models using large training sets remains computationally extremely demanding. An exponential increase in the available compute power without incurring exponentially rising energy consumption has therefore been a third prerequisite to reach the prediction accuracy of today's dihedral prediction methods. As machine learning is increasingly used for large-scale commercial tasks such as voice or image recognition, several recent implementations make use of energy efficient hardware architectures such as graphics processing units (GPU) [60, 61], many-core processors [62, 63], or field programmable gate arrays (FPGA) [64, 65].

Research: The most important contribution to the improvements witnessed in the area of dihedral angle prediction has been the accumulation of many ideas, how to optimally exploit the information hidden in the sequence profile data, and how to tackle algorithmic shortcomings. Only few methods, e.g., HMMs and CRFs, have been designed to solve sequence labeling tasks such as predicting the sequence of dihedral angles of a peptide chain. Using a sliding window [56] mapping it became possible to harness the prediction performance of standard supervised learning methods for prediction of dihedral angles. This method however only learns from the sequence input, but ignores any independent information contained in the correlation of the dihedral angle labels. Furthermore, methods like SVM could only predict unary values, either a continuous dihedral angle or a discrete label, e.g., the letter of a structural alphabet, a dihedral region, or a secondary structure class. One key idea to learn the label correlations and approaching the original sequence labeling task more closely was iterative learning, also called recurrent sliding windows [29]. In this approach two or more predictors are trained successively using the output of the first predictor as additional input for the next. Iterative learning improved predictions not only by being able to learn local dihedral angle correlations but also allowed to generate different types of intermediate information such as solvent accessibility or secondary structure state that would provide complementary information to subsequent classifiers.

A further problem that had to be overcome was due to the cyclic nature of angles. As machine learning methods typically treat angles as linear dimensions, a small prediction error close to the implied cut in the circle, e.g., predicting ψ to be 178° instead of −178°, would be treated as an error of 356° instead of one of 4°. Various approaches have been tested to reduce the impact of this effect. Among them were shifting the ψ-angles by 100° and the

φ-angles by −10° to move the circle cut to regions of minimal probability density [43], splitting the original regression problem into identification of the distribution peak the dihedral angle would belong to, and the prediction of the signed distance to that peak [44]. The best performance so far has been achieved by mapping the dihedral angles to their sine and cosine values [51].

The synergies between advancements in these different areas were key to the development of dihedral angle prediction approaches that now have the potential to supersede classical secondary structure prediction.

6.3 Applications of Dihedral Angle Prediction

From a user perspective the most important aspects are probably in which contexts dihedral angle predictions are applicable and how to obtain dihedral angle predictions to apply them for own research tasks.

Dihedral predictions are not accurate enough to directly construct 3d models of entire protein chains [51]. Predicted dihedral angles will therefore more likely be used as a substitute or a supplement for predicted secondary structure in applications such as fold recognition, sequence-structure alignment, prediction of mutation effects, and 3d-modeling of proteins.

Predicted dihedral angles have the potential to substitute or complement secondary structure in various structure prediction tasks. One obvious way to apply dihedral angle predictions is to map them to secondary structure predictions. Only the very latest methods, however, provide secondary structure predictions that are superior to dedicated secondary structure prediction programs and more detailed information in particular for loop regions will be lost.

Secondary structure, dihedral angles, dihedral angle regions, or structure motifs all constitute mappings of a local 3d-conformation to a single letter or ordered tuple of numbers. The 3d-structure of a chain can thereby be reduced to a sequence of letters or values that is amenable to processing established for other sequences such as searching and alignment.

As dihedral patterns, i.e., structural motifs are capable of encoding topology-defining motifs such as beta-turns, their prediction has been successfully used in remote homology detection [34, 40, 66, 67]. Likewise inclusion of predicted dihedral angles has led to improved fold recognition approaches [68–70].

Structural alignment methods that are based on comparing dihedral angles rather than atomic distances [71, 72] can be directly used with predicted dihedral angles for improving target-template alignments in structure modeling.

It has also been found that substitution of predicted dihedral angles for predicted secondary structure improves fragment-free protein structure modeling [73].

For academic researchers several of the methods discussed above are available as web servers that take FASTA-formatted sequences as input and either return the results per email or allow for downloading them. None of the servers, however, provides programmatic access. Users that wish to integrate dihedral angle prediction methods into their own prediction or modeling pipelines will therefore need to install the respective software packages locally. At the time of writing at least ANGLOR, SPINE X/XI, and SPIDER-2 have been made available for downloading (c.f. Table 2).

Which of the current methods or future methods is optimal for a given task does not only depend on the reported prediction accuracy. Other considerations may be the computational effort needed for a prediction, the size of the models, how well the programs are documented, and how easily they can be integrated into existing pipelines to replace or supplement secondary structure predictions.

7 Future Directions

Although dihedral angle prediction has been very successful and augments secondary structure predictions in both accuracy and chain coverage, many open research questions remain.

Taking into account the effect of cis omega angles or nearby disulfide bonds on dihedral preferences may open up ways for further improvements in prediction accuracy.

First analysis reports also hint at different dihedral angle preferences in membrane environments [74]. Also the effect of residue modifications such as posttranslational modifications may provide new prediction targets. A large amount of research is also needed regarding the optimal way to exploit accurate dihedral predictions in terms of fold recognition, sequence-structure alignments, protein modeling, and other applications.

Despite their usual description in terms of static geometric features protein structures are highly dynamic and first methods for predicting the dihedral angle dynamics have been developed. Such prediction methods have the potential to enhance our method spectrum for predicting functional regions of proteins, to get insight into structural preferences of intrinsically disordered proteins and other aspects of folding. Research in this area is currently dominated by physics-based simulations and interesting developments may emerge from synergies between prediction and simulation approaches.

Several groups with a strong background in protein structure prediction have turned to the inverse problem, predicting sequences that adopt a given fold [75–77]. One requirement for protein design [78, 79] will be the accurate prediction of sequence-structure relationships for all parts of the desired structure, a task well suited for dihedral angle prediction [80].

References

1. Ramachandran GN, Ramakrishnan C, Sasisekharan V (1963) Stereochemistry of polypeptide chain configurations. J Mol Biol 7:95–99
2. Moss GP (2009) Basic terminology of stereochemistry (IUPAC Recommendations 1996). Pure Appl Chem 68:2193–2222
3. Touw WG, Joosten RP, Vriend G (2015) Detection of trans-cis flips and peptide-plane flips in protein structures. Acta Cryst D 71:1604–1614
4. Ramachandran GN, Sasisekharan V (1968) Conformation of polypeptides and proteins. Adv Protein Chem 23:284–438
5. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE (2000) The protein data bank. Nucleic Acids Res 28:235–242
6. Lovell SC, Davis IW, Adrendall WB, de Bakker PIW, Word JM, Prisant MG, Richardson JS, Richardson DC (2003) Structure validation by Cα geometry: ϕ, ψ and Cβ deviation. Proteins 50:437–450
7. Hovmöller S, Zhou T, Ohlson T (2002) Conformations of amino acids in proteins. Acta Cryst D58:768–776
8. Ting D, Wang GL, Shapovalov M, Mitra R, Jordan MI, Dunbrack RL (2010) Neighbor-dependent Ramachandran probability distributions of amino acids developed from a hierarchical Dirichlet process model. PLoS Comput Biol 6, e1000763
9. Kihara D (2005) The effect of long-range interactions on the secondary structure formation of proteins. Protein Sci 14:1955–1963
10. Venkatachalam CM (1968) Stereochemical criteria for polypeptides and proteins. V. Conformation of a system of three linked peptide units. Biopolymers 6:1425–1436
11. Lewis PN, Momany FA, Scheraga HA (1973) Chain reversals in proteins. Biochim Biophys Acta 303:211–229
12. Richardson JS (1981) The anatomy and taxonomy of protein structure. Adv Protein Chem 34:167–339
13. Hutchinson EG, Thornton JM (1994) A revised set of potentials for β-turn formation in proteins. Protein Sci 3:2207–2216
14. Milner-White EJ, Poet R (1987) Loops, bulges, turns and hairpins in proteins. Trends Biochem Sci 12:189–192
15. Bystroff C, Baker D (1998) Prediction of local structure in proteins using a library of sequence-structure motifs. J Mol Biol 281:565–577
16. Bystroff C, Thorsson V, Baker D (2000) HMMSTR: a hidden Markov model for local sequence-structure correlations in proteins. J Mol Biol 301:173–190
17. Matthews BW (1975) Comparison of the predicted and observed secondary structure of T4 phage lysozyme. Biochim Biophys Acta 405:442–451
18. Gorodkin J (2004) Comparing two K-category assignment by a K-category correlation coefficient. Comp Biol Chem 28:367–374
19. Rost B, Sander C, Schneider R (1994) Redefining the goals of protein secondary structure prediction. J Mol Biol 235:13–26
20. Zemla A, Venclovas C, Fidelis K, Rost B (1999) A modified definition of Sov, a segment-based measure for protein secondary structure prediction assessment. Proteins 34:220–223
21. Baum LE, Petrie T (1966) Statistical inference for probabilistic functions of finite state Markov chains. Ann Math Stat 37:1554–1563
22. Lafferty J, McCallum A, Pereira F (2001) Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of 18th international conference on machine learning. pp 282–289
23. Boser BE, Guyon IM, Vapnik VN (1992) A training algorithm for optimal margin classifiers. In: Haussler D (ed) 5th annual ACM workshop on COLT. pp 144–152
24. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323:533–536
25. Bishop CM (2006) Pattern recognition and machine learning (information science and statistics). Springer, New York
26. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25:3389–3402
27. Rooman MJ, Kocher JP, Wodak SJ (1991) Prediction of protein backbone conformation based on seven structure assignments: influence of local interactions. J Mol Biol 221:961–979
28. Gibrat JF, Robson B, Garnier J (1991) Influence of the local amino acid sequence upon the zones of the torsional angles φ and ψ adopted by residues in proteins. Biochemistry 30:1578–1586
29. Kuang R, Leslie CS, Yang AS (2004) Protein backbone angle prediction with machine learning approaches. Bioinformatics 20:1612–1621

30. Zimmermann O, Hansmann UH (2006) Support vector machines for prediction of dihedral angle regions. Bioinformatics 22:3009–3015
31. Zhang S, Jin S, Xue B (2013) Accurate prediction of protein dihedral angles through conditional random field. Front Biol 8(3):353–361
32. Rost B, Sander C, Schneider R (1994) PHD: an automatic mail server for protein secondary structure prediction. Comput Appl Biosci 10:53–60
33. de Brevern AG, Etchebest C, Hazout S (2000) Bayesian probabilistic approach for predicting backbone structures in terms of protein blocks. Proteins 41:271–287
34. Karchin R, Cline M, Mandel-Gutfreund Y, Karplus K (2003) Hidden Markov models that use predicted local structure for fold recognition: alphabets of backbone geometry. Proteins 51:504–514
35. Sims GE, Choi I, Kim S (2005) Protein conformational space in higher order ψ-φ maps. Proc Natl Acad Sci U S A 18:618–621
36. de Brevern AG, Etchebest C, Hazout S (2004) Local backbone structure prediction of proteins. In Silico Biol 4:31
37. Mooney C, Vullo A, Pollastri G (2006) Protein structural motif prediction in multidimensional φ- ψ space leads to improved secondary structure prediction. J Comput Biol 13:1489–1502
38. Zimmermann O, Hansmann UHE (2008) LOCUSTRA: accurate prediction of local protein structure using a two-layer support vector machine approach. J Chem Inf Model 48:1903–1908
39. Söding J (2005) Protein homology detection by HMM-HMM comparison. Bioinformatics 21:951–960
40. Ghouzam Y, Postic G, de Brevern AG, Gelly JC (2015) Improving protein fold recognition with hybrid profiles combining sequence and structure evolution. Bioinformatics:btv462
41. Wood MJ, Hirst JD (2005) Protein secondary structure prediction with dihedral angles. Proteins 59:476–481
42. Dor O, Zhou Y (2007) Real-SPINE: an integrated system of neural networks for real-value prediction of protein structural properties. Proteins 68:76–81
43. Xue B, Dor O, Faraggi E, Zhou Y (2008) Real-value prediction of backbone torsion angles. Proteins 72(1):427–433
44. Faraggi E, Xue B, Zhou Y (2009) Improving the prediction accuracy of residue solvent accessibility and real-value backbone torsion angles of proteins by guided-learning through a two-layer neural network. Proteins 74(4):847–856
45. Wu S, Zhang Y (2008) ANGLOR: a composite machine-learning algorithm for protein backbone torsion angle prediction. PLoS One 3(10), e3400
46. Jones DT (1999) Protein secondary structure prediction based on position-specific scoring matrices. J Mol Biol 292:195–202
47. Song J, Tan H, Wang M, Webb GI, Akutsu T (2012) TANGLE: two-level support vector regression approach for protein backbone torsion angle prediction from primary sequences. PLoS One 7:e30361
48. Faraggi E, Zhang T, Yang Y, Kurgan L, Zhou Y (2011) SPINE X: improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles. J Comput Chem 33(3):259–267
49. Kountouris P, Hirst JD (2009) Prediction of backbone dihedral angles and protein secondary structure using support vector machines. BMC Bioinformatics 10:437
50. Lyons J, Dehzangi A, Heffernan R, Sharma A, Paliwal K, Sattar A, Zhou Y, Yang Y (2014) Predicting backbone Cα angles and dihedrals from protein sequences by stacked sparse auto-encoder deep neural network. J Comput Chem 35:2040–2046
51. Heffernan R, Paliwal K, Lyons J, Dehzangi A, Sharma A, Wang J, Sattar A, Yang Y, Zhou Y (2015) Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning. Sci Rep 5:11476
52. Yaseen A, Li Y (2014) Context-based features enhance protein secondary structure prediction accuracy. J Chem Inf Model 54:992–1002
53. Singh H, Singh S, Raghava GPS (2014) Evaluation of protein dihedral angle prediction methods. PLoS One 9(8):e105667
54. Zhou T, Shu N, Hovmoller S (2010) A novel method for accurate one-dimensional protein structure prediction based on fragment matching. Bioinformatics 26:470–477
55. Chen M, Chen Y, Brent MR (2008) CRF-OPT: an efficient high-quality conditional random field solver. In: Proceedings of 23rd AAAI conference on artificial intelligence. pp 1018–1023
56. Qian N, Sejnowski TJ (1988) Predicting the secondary structure of globular proteins using neural network models. J Mol Biol 202:865–884
57. Tsochantaridis I, Joachims T, Hofmann T, Altun Y (2005) Large margin methods for structured and interdependent output variables. J Mach Learn Res 6:1453–1484

58. Nitish S, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15:1929–1958
59. Hinton GE, Osindero S, Teh YW (2006) A fast learning algorithm for deep belief nets. Neural Comput 18:1527–1554
60. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 1:1097–1105
61. Li Q, Salman R, Test E, Strack R, Kecman V (2011) GPUSVM: a comprehensive CUDA based support vector machine package. Cent Eur J Comput Sci 1:387–405
62. You Y, Fu H, Song SL, Randles A, Kerbyson D, Marquez A, Yang G, Hoisie A (2015) Scaling support vector machines on modern HPC platforms. J Parallel Distrib Comput 76:16–31
63. Viebke A, Pllana S (2015) The potential of the Intel Xeon Phi for supervised deep learning. arXiv:1506.09067
64. Rabieah MB, Bouganis CS (2015) FPGA based nonlinear support vector machine training using an ensemble learning. In: Proceedings of international conference field programmable logic and applications (FPL) 2015
65. Zhang C, Li P, Sun G, Guan Y, Xiao B, Cong J (2015) Optimizing FPGA-based accelerator design for deep convolutional neural networks. FPGA'2015
66. Huang YM, Bystroff C (2006) Improved pairwise alignments of proteins in the twilight zone using local structure predictions. Bioinformatics 22:413–422
67. Suresh V, Ganesan K, Parthasarathy S (2013) A protein block based fold recognition method for the annotation of twilight zone sequences. Protein Pept Lett 20:249–254
68. Wu S, Zhang Y (2008) MUSTER: improving protein sequence profile-profile alignments by using multiple sources of structure information. Proteins 72:547–556
69. Zhang W, Liu S, Zhou Y (2008) SP^5: improving protein fold recognition by using torsion angle profiles and profile-based gap penalty model. PLoS One 3(6):e2325
70. Yang Y, Faraggi E, Zhao H, Zhou Y (2011) Improving protein fold recognition and template-based modeling by employing probabilistic-based matching between predicted one-dimensional structural properties of query and corresponding native properties of templates. Bioinformatics 27:2076–2082
71. Miao X, Waddell PJ, Valafar H (2008) TALI: local alignment of protein structures using backbone torsion angles. J Bioinform Comput Biol 6:163–181
72. Jung S, Bae SE, Son HS (2011) Validity of protein structure alignment method based on backbone torsion angles. J Proteomics Bioinform 4:218–226
73. Faraggi E, Yang Y, Zhang S, Zhou Y (2009) Predicting continuous local structure and the effect of its substitution for secondary structure in fragment-free protein structure prediction. Structure 17:1515–1527
74. Saravanan KM, Krishnaswamy S (2015) Analysis of dihedral angle preferences for alanine and glycine residues in alpha and beta transmembrane regions. J Biomol Struct Dyn 33:552–562
75. Koga N, Tatsumi-Koga R, Liu G, Xiao R, Acton TB, Montelione GT, Baker D (2012) Principles for designing ideal protein structures. Nature 491:222–227
76. Mitra P, Shultis D, Zhang Y (2013) EvoDesign: de novo protein design based on structural and evolutionary profiles. Nucleic Acids Res 41(W):273–280
77. Bellows ML, Taylor MS, Cole PA, Shen L, Siliciano RF, Fung HK, Floudas CA (2010) Discovery of entry inhibitors for HIV-1 via a new de novo protein design framework. Biophys J 99:3445–3453
78. Khoury GA, Smadbeck J, Kieslich CA, Floudas CA (2014) Protein folding and de novo protein design for biotechnological applications. Trends Biotechnol 32:99–109
79. Woolfson DN, Bartlett GJ, Burton AJ, Heal JW, Niitsu A, Thomson AR, Wood CW (2015) De novo protein design: how do we expand into the universe of possible protein structures? Curr Opin Struct Biol 33:16–26
80. Li Z, Yang Y, Faraggi E, Zhan J, Zhou Y (2014) Direct prediction of profiles of sequences compatible to a protein structure by neural networks with fragment-based local and energy-based nonlocal profiles. Proteins 82:2565–2573

Chapter 8

One-Dimensional Structural Properties of Proteins in the Coarse-Grained CABS Model

Sebastian Kmiecik and Andrzej Kolinski

Abstract

Despite the significant increase in computational power, molecular modeling of protein structure using classical all-atom approaches remains inefficient, at least for most of the protein targets in the focus of biomedical research. Perhaps the most successful strategy to overcome the inefficiency problem is multiscale modeling to merge all-atom and coarse-grained models. This chapter describes a well-established CABS coarse-grained protein model. The CABS (C-Alpha, C-Beta, and Side chains) model assumes a 2–4 united-atom representation of amino acids, knowledge-based force field (derived from the statistical regularities seen in known protein sequences and structures) and efficient Monte Carlo sampling schemes (MC dynamics, MC replica-exchange, and combinations). A particular emphasis is given to the unique design of the CABS force-field, which is largely defined using one-dimensional structural properties of proteins, including protein secondary structure. This chapter also presents CABS-based modeling methods, including multiscale tools for de novo structure prediction, modeling of protein dynamics and prediction of protein–peptide complexes. CABS-based tools are freely available at http://biocomp.chem.uw.edu.pl/tools

Key words Protein modeling, Protein simulations, Force-field, Statistical potentials, Knowledge-based potentials

1 Introduction

In the last two or three decades, we have been witnessing incredible progress in experimental and theoretical molecular biology. Thanks to intensive experimental studies, especially genome projects, huge amounts of sequence data (primary structures of proteins, nucleic acids, and other biomacromolecules) are now available. The combination of new experimental techniques and theoretical tools for their interpretation also provides structural (three dimensional) data for many biological macromolecules. Nevertheless, experimentally determined structures remain unknown for an increasing fraction of known protein sequences (but also other biomacromolecules). The explanation of this growing gap is simple: sequencing is now easier, faster, and less expensive than structure determination.

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_8,

Deeper understanding of the molecular basis of life processes requires not only determination of structures of single biomacromolecules but also realistic pictures of their interaction with other biomacromolecules, mechanisms of assembly processes, and structural and dynamic properties of resulting complexes. Taking into consideration that we know experimental structures of only a fraction of monomeric proteins, and that the estimated number of possible protein dimers (oligomers) is an order of magnitude larger than the number of monomers, it becomes obvious that structural biology needs strong support from theoretical studies. Efficient methods for structure prediction, and modeling of dynamics and interaction are necessary. Many important problems of molecular biology can be studied using classical all-atom molecular dynamics (MD) methods. For very small and fast folding proteins it is now possible to simulate the entire folding process using superfast dedicated computers [1]. For larger systems it is still beyond capability of the available computing technology, and the time gap is huge. This is the main reason for the development of new molecular modeling tools that can handle large systems. Simplified coarse-grained, and thereby computationally very fast, models can be used for simulations of large biomacromolecules and/or for the modeling of long time processes [2, 3]. Useful coarse-grained models need to be of sufficient resolution, enabling reasonable connection with atomistic pictures [3, 4]. The high importance of such methods has been recognized a long time ago [5], resulting in the plethora of new molecular modeling tools. Recently, "*the development of multiscale models for complex chemical systems*," a pioneering work of Karplus, Levitt, and Warshel, was awarded the Nobel Prize in Chemistry for 2013.

Several very efficient coarse-grained protein models have been developed. Some of them, such as Rosetta [6, 7] or I-Tasser [8, 9], are targeted onto structure prediction, while others, such as CABS [10, 11] or UNRES [12, 13], are more universal, enabling not only structure modeling but also realistic simulations of the dynamic properties of protein systems.

The methods presented in this chapter are based on the CABS (C-Alpha, C-Beta, and Side chains) discrete representation of protein chains. Two quite fundamental features make CABS qualitatively different from other coarse grained models. The first one is that the coordinates of the model chains are restricted to discreet positions in a simple three dimensional lattice. Lattice spacing is small enough to ensure good resolution of chain representation, and large enough to make possible predefinition (and storage as integer numbers in large data tables) of all possible local conformations. This way, due to the simple computation of local moves and related energy changes, the Monte Carlo dynamics simulations are

much faster than it would be possible for otherwise equivalent continuous models. The second unique feature of CABS is its interaction scheme. The force field consists of knowledge-based statistical potentials derived from the regularities observed in the known protein structures. All interactions, especially those between side chains, are treated as context-dependent. This way complicated multi-body effects are encoded in pairwise potentials. The potentials describing the energies of side chain–side chain interactions depend on the secondary structure of the interacting fragments, their mutual orientations and on the distance (short distance contacts only are treated in the explicit fashion) between side chain centers. Such a context-dependent model of pairwise interactions, especially its dependence on secondary structure, encodes the averaged effects of many physical interactions, and all these interactions, including complex interactions with the solvent, are treated in an implicit fashion.

The coarse-graining level of the CABS model enables fast and realistic reconstruction of atom level structure representation, enabling efficient multiscale modeling of protein systems [4]. The CABS model has proven to be a good tool for the computational prediction of three-dimensional protein structures, including de novo and comparative modeling, studies of protein dynamics and folding pathways, and flexible docking.

This chapter is organized as follows. In the Subheading 2 we describe the CABS protein structure representation, its force field and the sampling method. Special attention is given to the context-dependent force field of the model, which is strongly dependent on the one-dimensional properties of protein chains, especially their secondary structure assignments.

In the Subheading 3, we list and briefly describe various protein modeling methods based on the CABS model with the emphasis on those utilizing sequence and secondary structure data only. These methods include publicly available modeling tools: CABS-fold: server for protein structure, including de novo modeling and comparative modeling using one or more structural analogs [14]; CABS-dock: server for the flexible docking of peptides to proteins using no knowledge about the binding site [15, 16]; and pyCABS: software package for the simulation and analysis of long-term protein dynamics of globular proteins [17]. In the Subheading 4, we present example performance of CABS-fold, CABS-dock and pyCABS, together with short descriptions of their input requirements and options.

Finally, the Subheading 5 provides several specific comments about the modeling results obtained using CABS-based methods, their further utilization, interpretation, or alternative modeling techniques that may enhance modeling accuracy.

2 Materials

2.1 CABS Model: Coarse Grained Representation of Protein Structure

The CABS model is a universal tool for the modeling of protein structure dynamics and protein molecular docking. The main chain of protein structure is represented by a chain of Cα-atoms and pseudo atoms representing the center of virtual Cα–Cα bonds (*see* Fig. 1a, b). The latter one is needed for the simplified definition of hydrogen bonds. Side chains are represented by Cβ atoms and pseudo-atoms representing the centers of the remaining portions (where applicable) of the amino acid side chains. The CABS Cα trace is placed onto a lattice network with 0.61 Å spacing. This lattice representation significantly speeds up the Monte Carlo sampling scheme when compared with continuous models of similar resolution. The lattice spacing of 0.61 Å enables a large set of allowed orientations of Cα–Cα virtual bonds (when a slight fluctuation of their length is allowed) and thereby eliminates any noticeable orientation biases that are present in simple lattice models. The average accuracy of the Cα-trace representation is about 0.35 Å, and slightly depends on the secondary structure patterns of the proteins studied (*see* Fig. 1c).

2.2 Force Field of the CABS Model with Secondary Structure Context-Dependent Statistical Potentials

The force field of CABS is constructed from knowledge-based statistical potentials, derived from the structural regularities (and their relation to the amino acid sequences) seen in protein structures collected in databases. A large representative set has been used for the derivation of all potentials. The weight of various potentials is properly tuned by optimizing the total energy of folded structure and other properties of the model, for instance secondary structure content at folded and unfolded structures of the proteins being modeled. The details of the force field of CABS models and the motivations for specific choices of their potentials have been described previously [10]. Here we outline the general ideas behind this force field, focusing on the crucial role of secondary structure assignments for the model and its force field.

Protein chain geometry in the CABS model is fully encoded by its Cα-trace, where positions of all Cα atoms are restricted to the points of the underlying cubic lattice grid. The planar angles between two subsequent Cα–Cα pseudo-bonds are restricted to values seen in protein structures. Sequence-independent and sequence-dependent potentials enforce distribution of this angle typical for the distribution seen in globular proteins. The angles of rotation of three consecutive Cα–Cα pseudo-bonds are similarly treated. This way, for instance, left-handed helix-like conformations are treated as unlike. The sequence dependence of the angular potentials is not straightforward, and it does not come from a specific identity of three or four residue fragments, but from the predicted secondary structure which depends on much

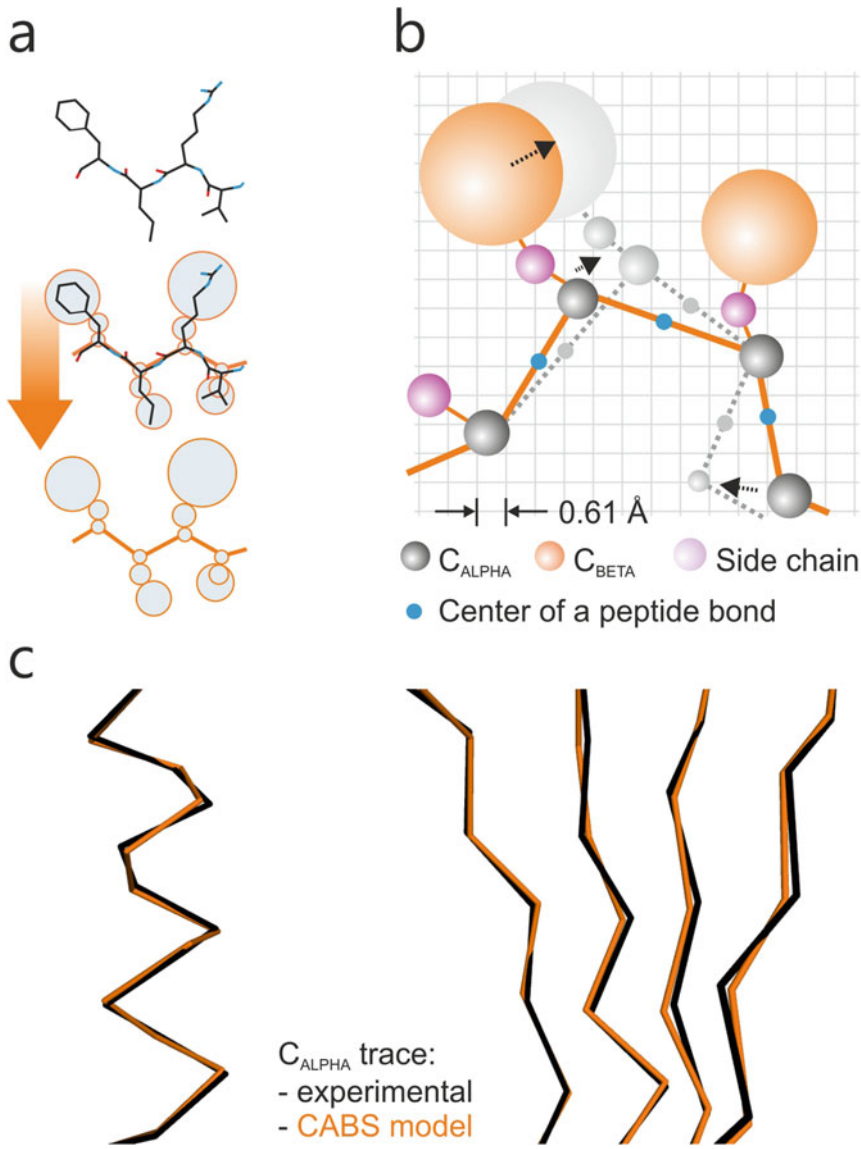


Fig. 1 Representation of a protein chain in the CABS model. (**a**) scheme showing conversion from all-atom to coarse-grained CABS representation, (**b**) details of CABS coarse-grained representation, (**c**) comparison of the C-alpha trace in experimental protein structure (*black* color) and after conversion to CABS representation (*orange*) presented here for an example helix and beta sheet secondary structure (experimentally derived C-alpha coordinates of both secondary structure motifs were taken from the 2GB1 PDB file)

longer protein fragments. This way complex multibody interactions are encoded in this simple potential. Positions of Cβ carbons (not restricted to the lattice) are defined by the positions of three consecutive Cα atoms for the Cβ bound to the central Cα. These positions depend on the planar angles between the Cα–Cα pseudo-bonds. Cα and Cα united atoms are treated as rigid bodies. Virtual united atoms, placed at the center of atom-Cα pseudo-bonds,

define the positions of the main chain hydrogen bonds in the form of attractive, orientation-dependent contact potentials of the same strength for all residues belonging to the same predefined secondary structure assignment. Other hydrogen bonds have the same geometry, but are considered weaker, with a smaller weight factor. The excluded volume spheres of united Cα, Cβ atoms and hydrogen bonds forming pseudo-atoms are slightly smaller in the CABS model than the distance of a corresponding strong repulsive interaction in real proteins. This is necessary to enable non-perfect dense packing in the native-like structure of the modeled proteins. Positions of centers of the remaining portion of amino acid side chains (where applicable) are also taken from tables defined for specific amino acids and local angles of the Cα-trace. Interactions between the centers of amino acid side chains are most important for the performance of the CABS model. Side chains are treated as soft excluded volume bodies at short distances, and interacting through contact potential at a longer distance. The width of contact distance is about 2 Å. The soft excluded volume of the side chains and the width of the contact range cover the potential problems with non-accurate representation of side chain conformations, especially for larger amino acids.

Side chain pairwise contact potentials are crucial for the performance of the CABS model. These statistical potentials are context dependent, and the strength of pairwise interactions depends on the mutual orientation of the interacting side chains and on the geometry of the nearest fragments of the main chain backbone. Here we discuss and present this potential for single domain globular proteins. It is important to note that the reference state in the derivation of CABS statistical potentials is a compact state of protein chains with a random sequence of the same composition as the protein of a given composition. Similar context-dependent potentials can be derived for interactions between globular proteins, transmembrane proteins, etc. It means that the CABS force field is not easily "transferable," it is rather "expandable" for an increasing range of modeled systems. We do not consider this a strong disadvantage of "knowledge-based" statistical potentials. "Transferability" of "physics-based" force fields for reduced models is also not trivial [18].

The idea of the context-dependent classification of side chain contacts is illustrated in Fig. 2 and numerical data are presented in Tables 2–10. The mutual orientation of the contacting side chains is divided into three ranges: near-antiparallel, intermediate, and near parallel (Fig. 2a). The local geometry of the main chain of a contacting residue is classified in the CABS force field into two classes: compact and expanded. This way the secondary structure prediction (or assignment) defines the specific energy of side chain interactions. For example: an antiparallel contact of two residues with a compact geometry of the corresponding elements of the main chain backbone usually means a helix-helix contact, while a parallel contact of side chains from two expanded elements of the

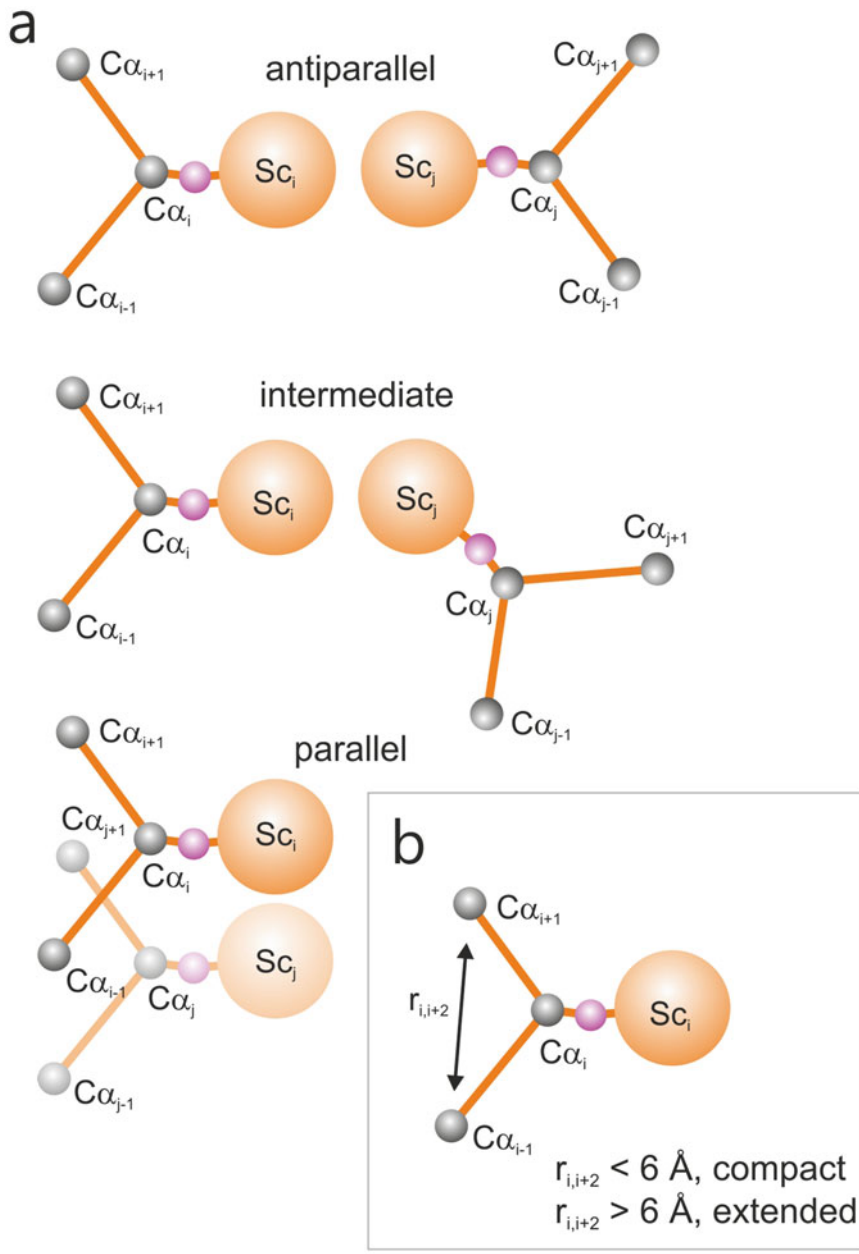


Fig. 2 Types of protein structure arrangements used in the definition of sequence-dependent pairwise potentials. (**a**) Three types of mutual orientations of the side chains (antiparallel, medium-intermediate, parallel). (**b**) Two types of main chain conformations (helical-compact and expanded-beta). Numerical values of the potentials are given in Tables 2–10

main chain usually comes from two adjoined beta strands. The context-dependent contact potentials of the CABS force field differ qualitatively from the other potentials (probably all of them) used in protein modeling. In the majority of these potentials the contact energy for two oppositely charged amino acids will suggest weak attractive interactions of their side chains. In the CABS force

field (for single-domain globular proteins) interactions of such pairs of residues are treated as strongly attractive for a parallel contact and strongly repulsive for an antiparallel contact of the side chains. Since the solvent in the CABS model force field is treated in a strictly implicit fashion (which is also the case for the majority of other statistical potentials) such orientation-dependent strength of interactions is not surprising. Charged residues are usually located on the surface of a protein globule, where they cannot form antiparallel contacts. If they are located (which is rather rare) more in the center of a globule, it is most likely that it is a binding site, where charged residues are on the surface of the binding site. Also in this case the parallel contact is more probable.

As discussed above, the predicted (or assigned) secondary structure in a three-state version (helix, beta, other) is crucial for CABS force field statistical potentials. This unique feature of the CABS coarse-grained modeling approach is a strength of the model, with very few drawbacks. The model has proved to be very efficient in de novo protein structure assembly simulations, comparative modeling support, modeling of protein dynamics and interactions with other biomolecules. In the last case the force field needs to be properly expanded, including for example contact potentials between side chains from two protein (peptide) chains. Due to the qualitative difference between CABS side chain contact potentials and other statistical potentials, we decided to attach its numerical data presented in nine tables (numbered from Tables 2–10). Two-digit accuracy is sufficient for most applications of this potential.

The contact potentials data (Tables 2–10) are potentially very useful not only in coarse-grained modeling (with model resolution similar to that assumed in the CABS model) but also as a source for definition/sorting of many other one-dimensional, two-dimensional, and three-dimensional protein features. For instance it is possible to use the numerical data of this potential for the classification of burial patterns of protein sequences. The potential can also be used in efficient threading algorithms, and in other structural bioinformatics methods. Additional comments on the meaning of the tables and accessibility in software packages are provided in **Note 1**.

2.3 Sampling Schemes

The Monte Carlo sampling scheme of CABS is a series of local, randomly selected, small conformational transitions onto the underlying lattice. The set of local changes of model chain coordinates includes single Cα moves (*see* Fig. 1b), moves of two Cα fragments, and rarely attempted small distance moves of longer fragments of the model chains. Chain ends are treated separately. Due to lattice discretization of the Cα coordinates (800 of allowed orientations of Cα–Cα pseudo-bond vectors) the possible local moves could be stored in large data tables and thereby local moves do not require any costly computations of trigonometric functions. Local moves require just simple random sorting of predefined

sequences of integer numbers. This way the discrete (restricted to a high coordination lattice) representation of chain conformations makes the CABS model computationally much faster in comparison to otherwise equivalent continuous coarse-grained models. The geometry of the main chain defines the positions of the side chain united atoms (not restricted to the lattice). A library of these positions is predefined by sorting and averaging PDB structures for all possible amino acid sequences of the central and two neighboring elements of the Cα–trace. All random moves are accepted according to the Metropolis criteria. Since the randomly selected moves mimic fast local conformational fluctuations of the modeled protein chains, their long series provides a realistic picture of the long time dynamics of modeled systems. CABS-based modeling schemes can use simple MC dynamics simulations at a given temperature, MC simulated annealing, and various versions of Replica Exchange (REMC) simulations. The CABS model (MC dynamics or REMC) could be easily combined with all-atom molecular dynamics. Several simple algorithms, classical and specifically targeted onto CABS representation, can be used for the fast and realistic reconstruction of atomistic representation, suitable for classical MD simulations (*see* **Note 2**). This way CABS can be used as a very efficient engine in multiscale protein modeling schemes. The basic structure of multiscale modeling procedures with CABS is illustrated in Fig. 3. Some helpful tools for the analysis of derived models and CABS trajectories are presented in **Notes 3** and **4**.

3 Methods

In the last several years, CABS coarse-grained protein models have become a key component in various multiscale modeling methods. Those methods generally follow a similar pipeline merging CABS simulations (usually the first modeling step) and all-atom modeling (final modeling steps), as presented in Fig. 3.

The CABS-based modeling methods have three application areas:

1. Protein structure prediction: homology modeling [14, 19–21], ab initio prediction of small proteins [14], or protein loops/fragments [22–24] (in ref. [23] also in combination with the classical Modeller tool [25]), modeling based on sparse experimental data [26].
2. Prediction of protein complexes: protein–peptide [15, 16, 27] and protein–protein [28, 29].
3. Efficient simulation of protein dynamics: protein folding mechanisms [4, 11, 30–34] and flexibility of globular proteins [35–38].

In all these applications, the CABS model serves as a highly efficient simulation engine that allows CABS-based methods to be much cheaper in terms of CPU time (in comparison to classical

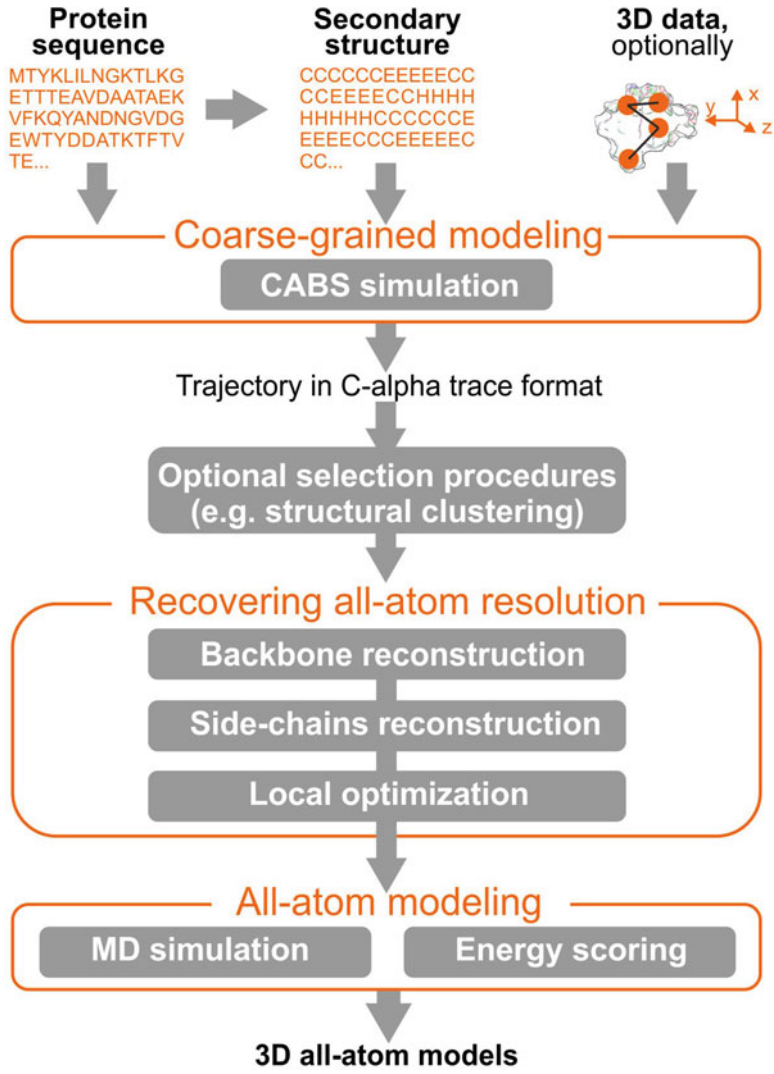


Fig. 3 Typical stages of the multiscale modeling scheme utilizing the CABS model. The modeling input includes one dimensional data (protein sequence and secondary structure) and, optionally, three-dimensional data (e.g., distance restraints from experiment or from evolutionary analysis). Secondary structure data are required in a three-letter code (C, coil; E, extended; H, helix). The modeling scheme consists of three major stages: (1) coarse-grained modeling with the CABS model, (2) several steps of reconstruction to all-atom representation, and (3) all-atom modeling procedures (e.g., simulation using all-atom MD or all-atom energy scoring)

modeling tools [35]), or to achieve sampling efficiency that exceeds other existing approaches. For example, the CABS-dock method for the molecular docking of peptides to proteins enables docking fully flexible peptides to flexible receptors without prior knowledge of the binding site [15, 16]. In practice, CABS-dock performs simulation of coupled folding and binding during which peptides have a possibility to explore the entire surface of a protein receptor. Presently, there are no other simulation methods enabling exploration of such a large conformational space in a reasonable time. In contrast to CABS-dock, other state-of-the-art protein–peptide docking methods are restricted to a specified binding site, or to very short peptides (2–4 amino acids, while CABS-dock has been successfully tested on a large set of peptides with 5–15 amino acids [15, 16]).

Table 1
Performance of the CABS-based modeling methods in ab initio prediction tasks (utilizing one dimensional data only: protein sequence and secondary structure)

Method and availability	Benchmark set	Performance summary
Prediction of protein structure or protein fragments		
CABS-fold server for the ab initio and consensus-based prediction of protein structure [14]. Available as a web server at: http://biocomp.chem.uw.edu.pl/CABSfold/	Methodology validated during CASP competitions as one of the leading approaches [19–21], applied to the ab initio modeling of large protein fragments or entire proteins (with or without 3D restraints)	Small proteins (up to 100 residues long) or peptides can be predicted with high accuracy (up to 2 Å) or medium accuracy (up to 5 Å) The CABS-fold server can also be used to predict protein loops (see the performance below)
Method(s) for predicting protein loops in globular proteins [23]	From 186 experimental protein structures, covering all the structural classes of proteins, internal loops of various length (from 4 to 25 residues) have been removed and treated as unknown	Performance was compared with two classical modeling tools: Modeller [25] and Rosetta [6]. Modeller performance was usually better for short loops, while CABS and Rosetta were more effective for longer loops (resolution of such models was usually on the level of 2–6 Å)
Prediction method for protein loops in GPCR membrane receptors [24]	From 13 experimental GPCR receptor structures, extracellular second loops (between 13 and 34 residues) have been removed and treated as unknown. The benchmark set is available at: http://biocomp.chem.uw.edu.pl/GPCR-loop-modeling/	Resolution of the best models obtained (among many others) was on the level of 2–6 Å, while the best scored models were on the level of 2–8 Å. Performance was comparable to that of other state-of-the-art methods [24]

(continued)

Table 1
(continued)

Method and availability	Benchmark set	Performance summary
Method for protein fragment reconstruction [22]	From 20 protein structures of various structural classes, protein fragments (from 10 to 29 residues) have been removed and treated as unknown	Resolution of the resulting models was on the level of 1.5 and 6 Å. Performance was compared with SICHO [53], Refiner [22], Swiss-model [54] and Modeller [25] methods. CABS, SICHO and Refiner performance was usually better than for Swiss-model and Modeller
Protein–peptide molecular docking and binding site prediction (using no knowledge about the peptide structure)		
CABS-dock method for molecular docking with no knowledge of the binding site [15, 16]. Available as a web server at: http://biocomp.chem.uw.edu.pl/CABSdock/	Benchmark set of nonredundant (<70% sequence identity with respect to the receptor protein) protein–peptide interactions (108 bound and 68 unbound receptors) with peptides of 5–15 amino acids [55]. The benchmark set is available at: http://biocomp.chem.uw.edu.pl/CABSdock/benchmark	For over 80% of bound and unbound cases high or medium accuracy models were obtained (high accuracy: peptide-RMSD<3 Å; medium accuracy: 3 Å ≤ peptide-RMSD ≤ 5.5 Å; where peptide-RMSD is the RMSD to the experimental peptide structure after superimposition of receptor molecules
Ab initio protocol for studying the folding and binding mechanism of intrinsically disordered peptides [27]	pKID-KIX protein complex (pKID is a 28 residue disordered peptide which folds upon binding to the KIX domain)	An ensemble of transient encounter complexes obtained in the simulations was in good agreement with experimental results
Prediction of protein folding mechanisms		
pyCABS protocols for efficient simulations of long-time protein dynamics [17]. Software package available at: http://biocomp.chem.uw.edu.pl/pycabs/	Tested in protein folding studies of small (up to 100 residues long) globular proteins [4, 11, 30–33]	The views of denatured ensembles of protein structures obtained in the simulations were in good agreement with the experimental measurements of protein folding [4, 11, 30–33]
Multiscale protocol merging efficient simulations with CABS and replica exchange all-atom MD [34]	β-Hairpin from the B1 domain of protein G (PDB code: 2GB1, residues 41–56)	Combination of CABS and all-atom MD simulations significantly accelerates system convergence (several times in comparison with all-atom MD starting from the extended chain conformation)

In Table 1, we list the CABS-based methods that enable protein structure modeling based on one-dimensional data only (sequence and secondary structure), together with their accessibility, references, benchmark information, and performance summary. For selected methods (CABS-fold [14], CABS-dock [15, 16], and pyCABS [17]), example case studies are presented in the next section.

Apart from the methods listed in Table 1, the CABS model has also been used in web server tools: CABS-flex server for the prediction of protein structure fluctuations [36, 37] and Aggrescan3D server for the prediction of protein aggregation properties and rational design [38] (Aggrescan3D uses the CABS-flex method for modeling the influence of conformational flexibility on aggregation properties). The major advantage of the CABS-flex method is its efficiency. It allows us to achieve similar results as with classical all-atom MD, but several thousand times faster [35].

4 Case Studies

4.1 Protein Structure Prediction Using the CABS-Fold Server

The CABS-fold server for protein structure prediction operates in two modeling modes: consensus modeling (based on structural templates) and de novo modeling (based only on sequence) [14]. In both modes, the secondary structure is an optional input (*see* **Note 5**): if the secondary structure is not provided, it is automatically predicted using the Psi-Pred method [39]. It is also possible to add distance restraints into the modeling process and to modify CABS simulation settings. These additional options can be accessed from the "Advanced options" input panel (*see* Fig. 4 presenting example CABS-fold screenshots).

CABS-fold performance and the benchmark summary are presented in Table 1. In Fig. 5, we present an example modeling result using the de novo modeling mode and the sequence of a small protein domain, yeast copper transporter CCC2A (72 residues). Protein sequence and secondary structure inputs are also provided in the figure. The CCC2A protein structure has been solved experimentally and has a beta–alpha–beta–beta–alpha–beta ferredoxin-like fold (PDB ID: 1fvq). Figure 5 shows a comparison of the experimental and CABS-fold predicted model with the same fold which differ in details of secondary structure packing. It is worth to mention that obtaining such a modeling result based on protein sequence only is not trivial and possible (in a reasonable computational time) only using a few coarse-grained based methods.

4.2 Protein–Peptide Docking Using the CABS-Dock Server

The CABS-dock server for modeling protein–peptide interactions [15, 16] enables efficient docking search of a peptide over

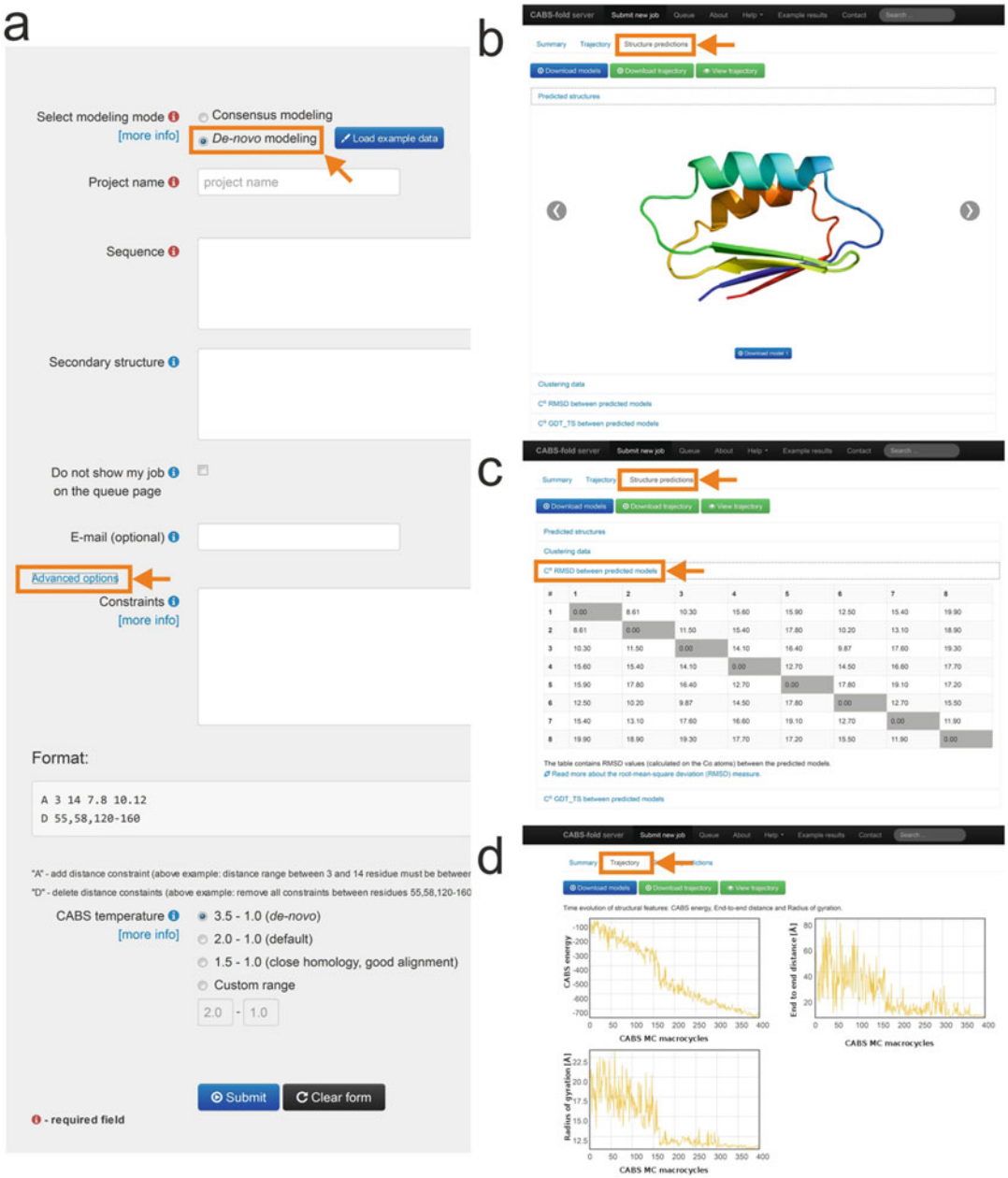


Fig. 4 Example screenshots from the CABS-fold server. (**a**) Main page input panel. Output panels presenting: (**b**) predicted models, (**c**) RMSD between the predicted models, (**d**) characteristics of the structure prediction trajectories. Selected/clicked options are marked with *orange rectangles* and *arrows*

Fig. 5 Example CABS-fold structure prediction result. The only input data: protein sequence and secondary structure (predicted from sequence by the Psi-pred method [ref]) are shown on the left. The experimental structure (*blue*) of a 72 residue protein (PDB ID: 1fvq) is superimposed on the CABS-fold predicted model (*orange*). In comparison to the experimental structure, the CABS-fold model has the same fold and RMSD value is 3.7 Å

the entire protein receptor structure. During CABS-dock docking, the peptide is simulated as fully flexible, while the protein receptor structure is also flexible but only to a small extent. As an input, the CABS-dock method uses information about the peptide sequence and structure of a protein receptor. The peptide secondary structure is an optional input (*see* **Note 5**; if not provided, the method uses the PsiPred tool [39] for secondary structure prediction). Other optional inputs include the possibility to assign high flexibility for selected receptor fragments, and to exclude selected receptor fragments from docking search (these are accessible from the optional input panel, see the CABS-dock screenshots in Fig. 6).

CABS-fold performance and the benchmark summary are presented in Table 1. In Fig. 7, we present an example modeling result obtained using the optional CABS-dock feature that allows for the significant flexibility of a selected receptor fragment. In the presented modeling case, assigning significant flexibility to the flexible loop

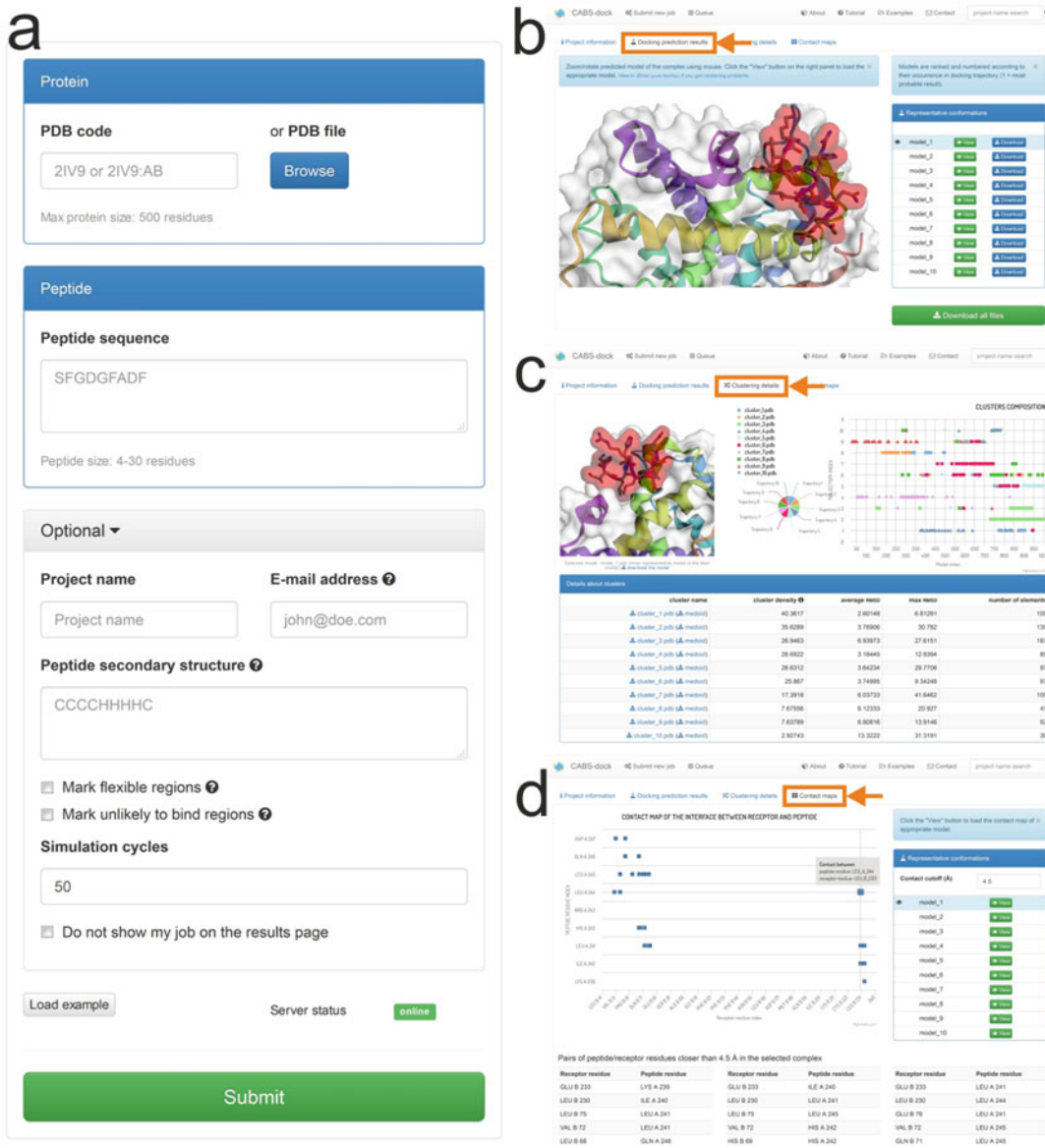


Fig. 6 Example screenshots from the CABS-dock server. (**a**) Main page input panel. Output panels presenting: (**b**) predicted models, (**c**) clustering results and analysis, (**d**) contact maps for predicted models. Selected/clicked options are marked with *orange rectangles* and *arrows*

(which partially blocks the binding site in the unbound input form) was crucial for obtaining a high resolution complex model.

4.3 Protein Dynamics Using the pyCABS Package

The pyCABS software package [17] is dedicated to performing long-time simulations of small globular proteins using the CABS model. The possible applications include de novo folding from a random structure (folding mechanisms), near-native dynamics, unfolding processes, and long-time dynamics of unfolded structures.

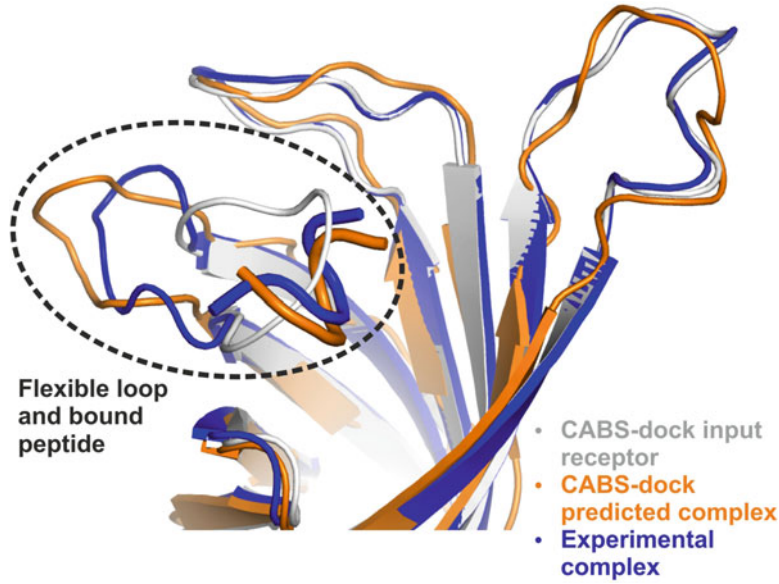


Fig. 7 Example result of CABS-dock protein–peptide docking using the option of significant flexibility for the selected receptor fragment. The figure shows comparison of the CABS-dock input structure in the peptide-unbound form (colored in *gray*, PDB ID: 2RTM) with a CABS-dock-predicted complex (in *orange*) and a peptide-bound experimental complex (in blue, PDB ID: 1KL3). RMSD between the predicted and experimental peptide structure is 2.03 Å. The flexible loop region (designated to be fully flexible during docking) is between residues 45 and 54

The package requires the protein sequence and its secondary structure (predicted or experimentally assigned, *see* **Note 5**) and starting structure(s): depending of the modeling goal, it can be a random structure, or a selected (e.g., native) structure.

pyCABS performance and the benchmark summary are presented in Table 1. In Fig. 8, we present an example modeling result from the simulation of folding of barnase globular protein. The simulation was performed in the de novo manner, i.e., using a random starting structure. The resulting picture of the folding mechanism matches well with the experimental data and has been described in detail in ref. [11] (the technical details for carrying out such a simulation using pyCABS are provided in ref. [17]).

5 Notes

1. Tables 2–10 are an integral part of the CABS (and pyCABS [17]) software package (stored in the "QUASI3S" text file). Each of these tables is labeled by a three-letter code (like PEE, PCC, and PCE) whose meaning is explained in Fig. 2. For

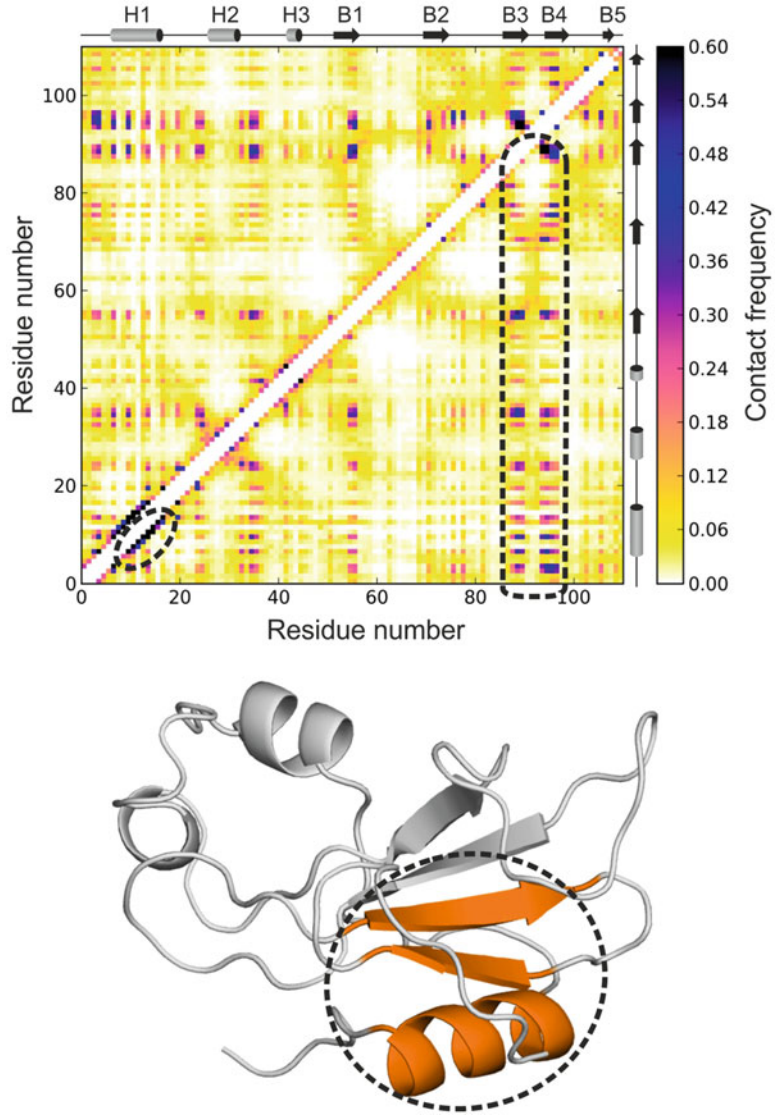


Fig. 8 Example result from simulations of long-term protein dynamics (from a fully denatured to a near-native state) using the CABS model and protein sequence only. A simulation contact map is presented showing the key step of barnase folding (PDB code: 1BNR). The presented key folding step is the formation of the nucleation site. The nucleation site is formed by the following elements of secondary structure: helix 1 and beta-strands: 3 (marked by *dashed lines* in the contact map and colored also in *orange* in the native barnase structure, shown below). The map colors indicate contact frequency (see the legend)

example, the PCE type of interactions occurs between amino acid chains forming a parallel contact (P), where the first contacting side chain (given in columns) is attached to a compact (C, most likely a helix) type of conformation and the second contacting side chain (given in rows) is attached to expanded (E, most likely beta-strand) conformation.

Table 2
Context pairwise contact potential for the PCC type of side chain interactions

PCC	GLY	ALA	SER	CYS	VAL	THR	ILE	PRO	MET	ASP	ASN	LEU	LYS	GLU	GLN	ARG	HIS	PHE	TYR	TRP
GLY	3.3	1.8	1.6	1.6	1.6	1.2	1.3	1.8	0.9	1.0	0.9	1.6	1.5	1.4	1.4	0.9	1.5	1.1	1.1	1.0
ALA	1.8	0.6	0.8	0.4	0.4	0.5	0.3	0.9	0.2	0.7	0.5	0.3	0.8	0.5	0.3	0.2	0.4	0.1	0.2	0.2
SER	1.6	0.8	0.7	0.9	0.3	0.4	0.2	0.9	0.3	0.2	0.2	0.5	0.3	0.2	0.0	−0.1	−0.1	0.1	0.0	0.0
CYS	1.6	0.4	0.9	−0.2	−0.4	0.3	−0.6	0.4	−0.6	0.9	0.4	−0.4	0.3	0.5	0.7	0.2	−0.4	−0.8	−0.6	−0.6
VAL	1.6	0.4	0.3	−0.4	−0.6	−0.3	−0.9	0.6	−0.8	0.7	−0.0	−0.9	0.1	0.2	−0.0	−0.3	−0.3	−1.0	−0.7	−1.1
THR	1.2	0.5	0.4	0.3	−0.3	−0.1	−0.5	0.7	−0.2	0.1	−0.2	−0.3	−0.1	−0.0	−0.4	−0.2	−0.2	−0.3	−0.4	−0.4
ILE	1.3	0.3	0.2	−0.6	−0.9	−0.5	−1.1	0.4	−0.9	0.5	0.1	−1.1	−0.1	−0.0	−0.2	-0.3	−0.3	−1.1	−0.7	−1.1
PRO	1.8	0.9	0.9	0.4	0.6	0.7	0.4	1.4	0.4	1.0	0.6	0.6	0.6	0.6	0.5	0.2	0.5	0.6	−0.0	−0.0
MET	0.9	0.2	0.3	−0.6	−0.8	−0.2	−0.9	0.4	−0.2	0.4	-0.1	−0.9	0.1	0.1	−0.3	−0.5	−0.7	−1.1	−0.9	−0.8
ASP	1.0	0.7	0.2	0.9	0.7	0.1	0.5	1.0	0.4	0.5	0.0	0.9	−0.6	0.5	−0.4	−0.7	−0.2	0.3	0.3	0.2
ASN	0.9	0.5	0.2	0.4	−0.0	−0.2	0.1	0.6	−0.1	0.0	0.1	0.2	−0.3	−0.1	−0.3	−0.3	−0.0	0.1	−0.1	−0.1
LEU	1.6	0.3	0.5	−0.4	−0.9	−0.3	−1.1	0.6	−0.9	0.9	0.2	−1.0	−0.0	0.2	−0.1	−0.3	−0.3	−1.1	−0.8	−0.9
LYS	1.5	0.8	0.3	0.3	0.1	−0.1	−0.1	0.6	0.1	−0.6	−0.3	−0.0	0.4	−0.9	−0.5	−0.1	0.1	−0.0	−0.4	0.0
GLU	1.4	0.5	0.2	0.5	0.2	−0.0	−0.0	0.6	0.1	0.5	−0.1	0.2	−0.9	0.2	−0.5	−1.0	−0.4	0.2	−0.1	−0.2
GLN	1.4	0.3	0.0	0.7	−0.0	−0.4	−0.2	0.5	−0.3	−0.4	−0.3	−0.1	−0.5	−0.5	−0.2	−0.5	−0.3	−0.1	−0.4	−0.7
ARG	0.9	0.2	−0.1	0.2	−0.3	−0.2	−0.3	0.2	−0.5	−0.7	−0.3	−0.3	−0.1	−1.0	−0.5	−0.2	−0.3	−0.3	−0.6	−0.5
HIS	1.5	0.4	−0.1	−0.4	−0.3	−0.2	−0.3	0.5	−0.7	−0.2	−0.0	−0.3	0.1	−0.4	−0.3	−0.3	−0.4	−0.5	−0.7	−0.8
PHE	1.1	0.1	0.1	−0.8	−1.0	−0.3	−1.1	0.6	−1.1	0.3	0.1	−1.1	−0.0	0.2	−0.1	−0.3	−0.5	−1.0	−1.0	−1.2
TYR	1.1	0.2	0.0	−0.6	−0.7	−0.4	−0.7	−0.0	−0.9	0.3	−0.1	−0.8	−0.4	−0.1	−0.4	−0.6	−0.7	−1.0	−0.5	−1.1
TRP	1.0	0.2	0.0	−0.6	−1.1	−0.4	−1.1	−0.0	−0.8	0.2	−0.1	−0.9	0.0	−0.2	−0.7	−0.5	−0.8	−1.2	−1.1	−1.1

The PCC type of interactions occurs between the amino acid chains forming a parallel contact (P), where both contacting side chains are attached to a compact (C, most likely a helix) type of conformation

Table 3
Context pairwise contact potential for the PEE type of side chain interactions

PEE	GLY	ALA	SER	CYS	VAL	THR	ILE	PRO	MET	ASP	ASN	LEU	LYS	GLU	GLN	ARG	HIS	PHE	TYR	TRP
GLY	2.6	1.3	0.9	0.8	0.8	0.8	0.6	1.4	0.5	0.5	0.8	0.5	1.0	0.8	0.6	0.6	0.3	0.0	0.1	−0.2
ALA	1.3	0.5	0.5	−0.0	−0.4	0.2	−0.6	0.6	−0.2	0.6	0.3	−0.6	0.5	0.6	0.3	0.1	0.1	−0.8	−0.6	−0.8
SER	0.9	0.5	0.0	0.1	0.1	−0.3	−0.1	0.6	−0.1	−0.3	−0.3	0.0	−0.0	−0.2	−0.3	−0.3	−0.6	−0.4	−0.5	−0.7
CYS	0.8	−0.0	0.1	−0.8	−0.9	−0.0	−0.9	0.2	−0.7	0.3	0.3	−0.9	0.1	0.4	−0.1	0.0	−0.6	−0.9	−0.7	−1.5
VAL	0.8	−0.4	0.1	−0.9	−1.3	−0.5	−1.5	0.2	−0.9	0.3	0.0	−1.4	−0.2	−0.1	−0.1	−0.5	−0.6	−1.3	−1.1	−1.2
THR	0.8	0.2	−0.3	−0.0	−0.5	−0.7	−0.6	0.3	−0.4	−0.2	−0.5	−0.4	−0.5	−0.6	−0.6	−0.7	−0.7	−0.5	−0.6	−0.6
ILE	0.6	−0.6	−0.1	−0.9	−1.5	−0.6	−1.6	0.1	−1.1	0.1	−0.0	−1.5	−0.2	−0.2	−0.2	−0.5	−0.5	−1.5	−1.1	−1.3
PRO	1.4	0.6	0.6	0.2	0.2	0.3	0.1	1.0	0.2	0.8	0.4	0.1	0.8	0.6	0.3	0.1	0.0	−0.3	−0.5	−0.5
MET	0.5	−0.2	−0.1	−0.7	−0.9	−0.4	−1.1	0.2	−0.4	0.2	−0.1	−1.1	−0.2	−0.2	−0.5	−0.5	−0.5	−1.3	−0.9	−1.2
ASP	0.5	0.6	−0.3	0.3	0.3	−0.2	0.1	0.8	0.2	0.4	−0.5	0.2	−0.6	0.1	−0.5	−1.0	−0.7	0.0	−0.4	−0.4
ASN	0.8	0.3	−0.3	0.3	0.0	−0.5	−0.0	0.4	−0.1	−0.5	−0.5	0.1	−0.3	−0.4	−0.6	−0.5	−0.4	−0.3	−0.6	−0.6
LEU	0.5	−0.6	0.0	−0.9	−1.4	−0.4	−1.5	0.1	−1.1	0.2	0.1	−1.3	−0.1	−0.2	−0.3	−0.4	−0.5	−1.4	−1.1	−1.2
LYS	1.0	0.5	−0.0	0.1	−0.2	−0.5	−0.2	0.8	−0.2	−0.6	−0.3	−0.1	0.3	−1.1	−0.6	−0.1	−0.3	−0.3	−0.8	−1.0
GLU	0.8	0.6	−0.2	0.4	−0.1	−0.6	−0.2	0.6	−0.2	0.1	−0.4	−0.2	−1.1	0.4	−0.3	−1.3	−0.8	−0.3	−0.8	−0.7
GLN	0.6	0.3	−0.3	−0.1	−0.1	−0.6	−0.2	0.3	−0.5	−0.5	−0.6	−0.3	−0.6	−0.3	−0.3	−0.8	−0.6	−0.6	−0.8	−1.0
ARG	0.6	0.1	−0.3	0.0	−0.5	−0.7	−0.5	0.1	−0.5	−1.0	−0.5	−0.4	−0.1	−1.3	−0.8	−0.4	−0.7	−0.7	−1.0	−0.8
HIS	0.3	0.1	−0.6	−0.6	−0.6	−0.7	−0.5	0.0	−0.5	−0.7	−0.4	−0.5	−0.3	−0.8	−0.6	−0.7	−0.6	−0.6	−1.0	−0.7
PHE	0.0	−0.8	−0.4	−0.9	−1.3	−0.5	−1.5	−0.3	−1.3	0.0	−0.3	−1.4	−0.3	−0.3	−0.6	−0.7	−0.6	−1.4	−1.2	−1.4
TYR	0.1	−0.6	−0.5	−0.7	−1.1	−0.6	−1.1	−0.5	−0.9	−0.4	−0.6	−1.1	−0.8	−0.8	−0.8	−1.0	−1.0	−1.2	−0.9	−1.1
TRP	−0.2	−0.8	−0.7	−1.5	−1.2	−0.6	−1.3	−0.5	−1.2	−0.4	−0.6	−1.2	−1.0	−0.7	−1.0	−0.8	−0.7	−1.4	−1.1	−0.9

The PEE type of interactions occurs between the amino acid chains forming a parallel contact (P), where both contacting side chains are attached to expanded (E, most likely beta-strand) type of conformation

Table 4

Context pairwise contact potential for the PCE and the PEC type of side chain interactions

PCE/PEC	GLY	ALA	SER	CYS	VAL	THR	ILE	PRO	MET	ASP	ASN	LEU	LYS	GLU	GLN	ARG	HIS	PHE	TYR	TRP
GLY	3.3	2.5	1.4	1.5	1.3	0.9	1.2	2.2	0.8	0.8	1.1	1.3	0.9	1.0	0.9	0.7	1.1	1.0	0.6	0.6
ALA	2.3	1.4	0.8	0.7	1.1	0.9	1.0	1.5	0.8	0.4	0.3	1.0	1.3	1.3	1.2	0.9	0.9	0.6	0.5	0.4
SER	1.3	0.8	0.4	0.6	1.1	0.5	1.0	0.5	0.9	−0.5	0.3	0.9	0.5	0.4	0.2	0.1	0.5	0.6	0.6	0.5
CYS	0.8	0.6	0.5	−1.2	0.2	0.3	−0.1	0.7	0.0	0.8	0.2	0.0	0.6	1.0	0.5	−0.1	−0.2	−0.3	−0.0	−0.8
VAL	1.6	0.6	0.6	0.3	0.1	0.5	−0.0	0.7	0.0	0.1	0.2	−0.2	0.5	0.2	0.4	0.3	0.2	−0.0	0.0	0.2
THR	1.3	0.9	0.3	0.8	0.8	0.3	0.6	0.7	0.5	−0.3	0.0	0.6	0.1	0.1	0.3	0.1	0.2	0.5	0.1	0.4
ILE	1.3	0.5	0.4	0.2	0.1	0.3	−0.2	0.6	−0.2	0.6	0.2	−0.3	0.7	0.5	0.3	0.5	0.5	−0.4	−0.1	−0.2
PRO	1.6	1.4	1.3	1.4	0.9	1.0	0.8	1.2	1.1	1.0	0.4	0.8	1.0	0.7	0.6	0.7	0.8	0.3	−0.1	−0.1
MET	1.0	0.4	0.5	0.0	0.3	0.8	0.0	0.5	−0.1	0.4	0.1	−0.0	0.6	1.0	0.3	0.2	−0.1	−0.2	0.0	−0.4
ASP	1.0	1.1	0.1	1.1	1.2	0.1	1.2	0.8	0.8	0.2	−0.2	1.0	−0.2	0.8	0.3	−0.5	−0.1	1.0	0.2	0.6
ASN	1.3	1.2	0.5	1.2	0.9	0.3	1.1	0.9	0.7	−0.2	−0.2	0.8	0.3	0.3	0.2	0.0	0.2	0.9	0.1	0.2
LEU	1.3	0.5	0.6	0.1	−0.0	0.6	−0.3	0.8	−0.0	0.6	0.4	−0.4	0.7	0.7	0.5	0.3	0.1	−0.3	−0.2	−0.4
LYS	1.8	1.9	0.6	1.6	1.3	0.4	1.3	1.0	0.8	−0.2	0.2	1.1	1.1	−0.1	0.5	0.7	0.5	0.8	0.4	0.4
GLU	1.3	1.3	0.1	1.1	1.6	0.2	1.3	0.7	0.9	0.5	0.1	1.3	−0.1	0.9	0.6	−0.3	0.2	1.0	0.7	0.4
GLN	1.0	1.1	0.4	1.3	1.4	0.5	0.8	0.7	0.6	0.1	0.3	0.8	0.4	0.5	0.3	0.1	0.6	0.6	0.5	−0.1
ARG	1.1	1.2	0.5	0.6	1.0	0.5	0.8	0.4	1.0	−0.5	0.1	0.7	0.5	−0.1	0.1	0.4	0.2	0.5	0.1	0.4
HIS	1.3	0.8	0.2	−0.1	0.6	0.2	0.5	0.3	0.6	0.0	−0.0	0.6	0.1	0.0	0.1	0.2	−0.1	0.1	−0.2	−0.4
PHE	1.0	0.2	0.2	−0.3	−0.2	0.1	−0.4	0.6	−0.5	0.3	0.0	−0.4	0.6	0.4	0.3	0.1	−0.1	−0.7	−0.4	−0.3
TYR	1.0	0.6	0.2	0.2	0.2	0.1	−0.0	−0.1	−0.2	0.0	−0.2	−0.1	0.2	0.2	0.3	−0.1	−0.4	−0.4	−0.1	−0.5
TRP	0.7	0.0	0.0	−0.1	−0.0	−0.2	−0.5	−0.4	−0.6	0.2	−0.1	−0.3	0.2	−0.2	−0.3	0.1	−0.3	−0.7	−0.4	−0.4

The PCE type of interactions occurs between amino acid chains forming a parallel contact (P), where the first contacting side chain (given in columns) is attached to the compact (C, most likely a helix) type of conformation and the second contacting side chain (given in rows) is attached to expanded (E, most likely beta-strand) conformation

Table 5
Context pairwise contact potential for the MCC type of side chain interactions

MCC	GLY	ALA	SER	CYS	VAL	THR	ILE	PRO	MET	ASP	ASN	LEU	LYS	GLU	GLN	ARG	HIS	PHE	TYR	TRP
GLY	2.0	1.4	1.1	0.7	1.0	1.0	0.7	1.2	0.5	0.9	0.9	0.7	1.0	1.3	0.8	0.5	1.0	0.4	0.4	0.2
ALA	1.4	0.3	0.4	0.0	−0.2	0.0	−0.3	0.5	−0.3	0.5	0.5	−0.3	0.4	0.3	0.3	0.1	0.4	−0.4	−0.4	−0.5
SER	1.1	0.4	0.6	0.4	0.1	0.2	0.2	0.5	0.2	0.3	0.4	0.1	0.2	0.5	0.1	0.1	0.1	−0.1	−0.1	−0.2
CYS	0.7	0.0	0.4	−1.3	−0.3	0.2	−0.5	0.3	−0.7	0.6	0.5	−0.5	0.5	0.5	0.8	0.1	−0.6	−0.9	−0.6	−0.9
VAL	1.0	−0.2	0.1	−0.3	−0.5	−0.1	−0.8	−0.0	−0.8	0.5	0.5	−0.8	0.5	0.2	0.1	0.1	−0.1	−0.8	−0.7	−1.1
THR	1.0	0.0	0.2	0.2	−0.1	0.2	−0.2	0.2	−0.3	0.3	0.0	−0.2	0.2	0.3	0.0	−0.0	0.0	−0.3	−0.3	−0.6
ILE	0.7	−0.3	0.2	−0.5	−0.8	−0.2	-0.8	0.1	−1.0	0.5	0.3	−1.0	0.2	0.1	0.0	−0.2	−0.2	−1.1	−0.9	−1.2
PRO	1.2	0.5	0.5	0.3	−0.0	0.2	0.1	0.7	0.0	0.6	0.5	0.2	0.5	0.7	0.3	−0.0	0.2	−0.3	−0.3	−0.3
MET	0.5	−0.3	0.2	−0.7	−0.8	−0.3	−1.0	0.0	−0.6	0.4	0.3	−0.9	0.2	0.3	−0.1	−0.3	−0.4	−1.1	−0.9	−1.2
ASP	0.9	0.5	0.3	0.6	0.5	0.3	0.5	0.6	0.4	0.5	0.2	0.6	−0.3	0.4	0.4	−0.6	−0.1	0.3	−0.2	−0.1
ASN	0.9	0.5	0.4	0.5	0.5	0.0	0.3	0.5	0.3	0.2	0.3	0.4	0.1	0.2	0.2	−0.1	0.1	0.1	−0.3	−0.4
LEU	0.7	−0.3	0.1	−0.5	−0.8	−0.2	−1.0	0.2	−0.9	0.6	0.4	−1.1	0.2	0.2	−0.1	−0.1	−0.2	−1.2	−0.9	−1.3
LYS	1.0	0.4	0.2	0.5	0.5	0.2	0.2	0.5	0.2	−0.3	0.1	0.2	0.9	−0.4	0.0	0.3	0.4	0.1	−0.3	−0.1
GLU	1.3	0.3	0.5	0.5	0.2	0.3	0.1	0.7	0.3	0.4	0.2	0.2	−0.4	0.4	0.0	−0.6	−0.2	0.0	−0.4	−0.2
GLN	0.8	0.3	0.1	0.8	0.1	0.0	0.0	0.3	−0.1	0.4	0.2	−0.1	0.0	0.0	0.1	−0.1	0.0	−0.1	−0.3	−0.5
ARG	0.5	0.1	0.1	0.1	0.1	−0.0	−0.2	−0.0	−0.3	−0.6	−0.1	−0.1	0.3	−0.6	−0.1	0.1	−0.0	−0.3	−0.6	−0.6
HIS	1.0	0.4	0.1	−0.6	−0.1	0.0	−0.2	0.2	−0.4	−0.1	0.1	−0.2	0.4	−0.2	0.0	−0.0	−0.0	−0.3	−0.3	−1.0
PHE	0.4	−0.4	−0.1	−0.9	−0.8	−0.3	−1.1	−0.3	−1.1	0.3	0.1	−1.2	0.1	0.0	−0.1	−0.3	−0.3	−1.2	−1.2	−1.5
TYR	0.4	−0.4	−0.1	−0.6	−0.7	−0.3	−0.9	−0.3	−0.9	−0.2	−0.3	−0.9	−0.3	−0.4	−0.3	−0.6	−0.3	−1.2	−0.7	−1.3
TRP	0.2	−0.5	−0.2	−0.9	−1.1	−0.6	−1.2	−0.3	−1.2	−0.1	−0.4	−1.3	−0.1	−0.2	−0.5	−0.6	−1.0	−1.5	−1.3	−1.2

The MCC type of interactions occurs between the amino acid chains forming a medium-intermediate contact (M), where both contacting side chains are attached to a compact (C, most likely a helix) type of conformation

Table 6
Context pairwise contact potential for the MEE type of side chain interactions

MEE	GLY	ALA	SER	CYS	VAL	THR	ILE	PRO	MET	ASP	ASN	LEU	LYS	GLU	GLN	ARG	HIS	PHE	TYR	TRP
GLY	1.5	1.2	0.8	0.2	0.6	0.5	0.6	0.8	0.3	0.7	0.3	0.5	0.9	0.7	0.7	0.3	0.1	−0.0	−0.3	−0.5
ALA	1.2	0.4	0.5	−0.3	−0.3	0.2	−0.4	0.3	−0.2	0.5	0.3	−0.2	0.5	0.4	0.4	0.1	0.1	−0.5	−0.4	−0.8
SER	0.8	0.5	0.4	0.0	0.1	0.1	0.1	0.4	0.2	0.1	−0.1	0.2	0.4	0.1	0.1	0.0	−0.2	−0.1	−0.3	−0.4
CYS	0.2	−0.3	0.0	−1.4	−0.5	−0.2	−0.5	−0.1	−0.5	0.3	0.2	−0.6	0.3	0.6	−0.0	−0.1	−0.5	−0.8	−0.8	−1.0
VAL	0.6	−0.3	0.1	−0.5	−0.4	0.1	−0.6	−0.1	−0.5	0.5	0.3	−0.7	0.2	0.4	0.0	−0.0	−0.1	−0.9	−0.7	−1.1
THR	0.5	0.2	0.1	−0.2	0.1	0.2	−0.0	0.1	0.1	−0.0	0.1	0.0	0.2	0.1	0.1	−0.2	−0.0	−0.2	−0.3	−0.4
ILE	0.6	−0.4	0.1	−0.5	−0.6	−0.0	−0.8	−0.2	−0.6	0.4	0.3	−0.8	0.3	0.4	0.1	0.0	−0.2	−1.0	−0.8	−1.1
PRO	0.8	0.3	0.4	−0.1	−0.1	0.1	−0.2	0.5	−0.3	0.6	0.2	−0.0	0.3	0.3	0.0	0.0	0.1	−0.3	−0.8	−0.9
MET	0.3	−0.2	0.2	−0.5	−0.5	0.1	−0.6	−0.3	−0.3	0.1	−0.2	−0.8	0.0	0.2	−0.1	−0.3	−0.2	−0.9	−0.7	−1.2
ASP	0.7	0.5	0.1	0.3	0.5	−0.0	0.4	0.6	0.1	0.3	−0.3	0.4	−0.3	0.4	0.1	−0.5	−0.5	0.1	−0.5	−0.4
ASN	0.3	0.3	−0.1	0.2	0.3	0.1	0.3	0.2	−0.2	−0.3	−0.1	0.2	0.2	0.0	0.2	−0.1	−0.2	−0.1	−0.4	−0.6
LEU	0.5	−0.2	0.2	−0.6	−0.7	0.0	−0.8	−0.0	−0.8	0.4	0.2	−0.8	0.2	0.4	0.0	−0.1	−0.1	−1.1	−0.8	−1.2
LYS	0.9	0.5	0.4	0.3	0.2	0.2	0.3	0.3	0.0	−0.3	0.2	0.2	0.8	−0.4	0.0	0.2	0.2	−0.1	−0.5	−0.6
GLU	0.7	0.4	0.1	0.6	0.4	0.1	0.4	0.3	0.2	0.4	0.0	0.4	−0.4	0.6	0.0	−0.7	−0.3	0.0	−0.4	−0.6
GLN	0.7	0.4	0.1	−0.0	0.0	0.1	0.1	0.0	−0.1	0.1	0.2	0.0	0.0	0.0	0.1	−0.3	−0.3	−0.4	−0.6	−0.7
ARG	0.3	0.1	0.0	−0.1	−0.0	−0.2	0.0	0.0	−0.3	−0.5	−0.1	−0.1	0.2	−0.7	−0.3	0.1	−0.2	−0.4	−0.8	−1.1
HIS	0.1	0.1	−0.2	−0.5	−0.1	−0.0	−0.2	0.1	−0.2	−0.5	−0.2	−0.1	0.2	−0.3	−0.3	−0.2	−0.7	−0.5	−0.8	−0.8
PHE	−0.0	−0.5	−0.1	−0.8	−0.9	−0.2	−1.0	−0.3	−0.9	0.1	−0.1	−1.1	−0.1	0.0	−0.4	−0.4	−0.5	−1.2	−1.1	−1.5
TYR	−0.3	−0.4	−0.3	−0.8	−0.7	−0.3	−0.8	−0.8	−0.7	−0.5	−0.4	−0.8	−0.5	−0.4	−0.6	−0.8	−0.8	−1.1	−0.6	−1.5
TRP	−0.5	−0.8	−0.4	−1.0	−1.1	−0.4	−1.1	−0.9	−1.2	−0.4	−0.6	−1.2	−0.6	−0.6	−0.7	−1.1	−0.8	−1.5	−1.5	−1.2

The MEE type of interactions occurs between the amino acid chains forming a medium-intermediate contact (M), where both contacting side chains are attached to expanded (E, most likely beta-strand) type of conformation

taner@iastate.edu

Table 7
Context pairwise contact potential for the MCE and the MEC type of side chain interactions

MCE/MEC	GLY	ALA	SER	CYS	VAL	THR	ILE	PRO	MET	ASP	ASN	LEU	LYS	GLU	GLN	ARG	HIS	PHE	TYR	TRP
GLY	2.0	1.3	1.0	0.9	1.3	0.9	1.1	1.3	0.9	0.6	0.6	0.9	1.2	1.4	0.9	0.6	1.0	0.5	0.5	0.7
ALA	1.5	0.9	0.5	0.5	0.4	0.4	0.2	0.8	0.1	0.5	0.6	0.1	1.0	1.1	1.0	0.7	0.7	−0.0	0.2	−0.1
SER	1.0	1.0	0.4	0.5	0.7	0.5	0.7	0.5	0.4	−0.1	0.1	0.6	0.5	0.5	0.7	0.4	0.1	0.2	0.1	0.2
CYS	0.6	0.2	0.2	−1.7	−0.2	0.1	−0.6	−0.0	−0.5	0.2	0.3	−0.4	0.4	0.7	0.1	−0.1	−0.5	−0.6	−0.3	−0.6
VAL	0.7	0.2	0.3	−0.1	−0.4	0.3	−0.5	0.0	−0.5	0.4	0.3	−0.7	0.7	0.5	0.3	0.2	0.1	−0.8	−0.4	−0.6
THR	0.8	0.5	0.2	0.3	0.2	0.3	0.2	0.1	0.0	−0.3	−0.1	−0.0	0.4	0.5	0.3	0.3	0.1	−0.1	0.0	−0.3
ILE	0.7	−0.1	0.5	−0.2	−0.6	−0.0	−1.0	−0.1	−0.5	0.5	0.2	−0.8	0.7	0.5	0.3	0.1	0.2	−0.9	−0.6	−0.8
PRO	1.0	0.7	0.7	0.3	0.4	0.8	0.3	0.5	0.3	0.6	0.2	0.4	0.7	0.7	0.5	0.1	0.1	0.1	−0.4	−0.6
MET	0.6	−0.0	0.5	−0.1	−0.5	0.1	−0.8	0.1	−0.4	0.3	−0.2	−0.7	0.8	1.0	0.2	0.1	0.0	−0.9	−0.8	−0.8
ASP	0.7	0.7	−0.2	0.5	0.7	−0.1	0.6	0.3	0.2	0.0	−0.1	0.2	−0.4	0.6	0.2	−0.7	−0.2	0.2	−0.3	−0.1
ASN	0.7	0.6	0.3	0.8	0.6	0.3	0.6	0.2	0.3	−0.3	−0.2	0.5	0.3	0.3	0.1	−0.2	0.2	0.2	−0.1	−0.5
LEU	0.9	0.0	0.5	−0.2	−0.6	0.2	−0.9	0.0	−0.7	0.7	0.5	−0.9	0.7	0.8	0.4	0.3	0.1	−1.0	−0.6	−0.8
LYS	1.2	1.0	0.7	0.8	0.9	0.6	0.7	0.9	0.5	−0.0	0.2	0.5	1.1	0.3	0.7	0.7	0.6	0.3	−0.1	0.1
GLU	1.0	1.0	−0.1	0.7	0.8	−0.2	0.6	0.4	0.1	0.4	0.0	0.4	−0.0	1.0	0.4	−0.3	0.1	0.2	0.1	−0.1
GLN	0.8	0.5	0.1	0.3	0.6	−0.2	0.4	0.3	−0.0	0.0	−0.1	0.1	0.3	0.4	0.4	−0.0	0.3	0.1	−0.3	−0.3
ARG	0.9	0.6	0.3	0.4	0.6	0.1	0.2	0.1	0.0	−0.3	0.1	0.2	0.7	−0.1	0.0	0.1	0.1	−0.1	−0.1	−0.3
HIS	0.6	0.5	−0.0	−0.7	0.2	−0.1	−0.0	0.1	−0.2	−0.5	0.1	−0.1	0.4	0.2	0.1	−0.1	−0.8	−0.4	−0.4	−0.7
PHE	0.4	−0.1	−0.0	−0.7	−0.7	−0.1	−1.0	−0.2	−1.1	0.1	−0.1	−1.0	0.4	0.2	−0.1	−0.2	−0.2	−1.2	−0.8	−1.2
TYR	0.2	0.2	−0.1	−0.2	−0.4	−0.2	−0.7	−0.6	−0.6	−0.4	−0.1	−0.7	−0.1	−0.0	0.1	−0.5	−0.6	−0.9	−0.8	−1.0
TRP	0.0	−0.6	−0.3	−0.6	−0.8	−0.3	−0.9	−0.9	−1.2	−0.0	−0.3	−1.2	−0.0	−0.0	−0.5	−0.7	−0.6	−1.3	−0.9	−1.1

The MCE type of interactions occurs between amino acid chains forming a medium-intermediate contact (M), where the first contacting side chain (given in columns) is attached to the compact (C, most likely a helix) type of conformation and the second contacting side chain (given in rows) is attached to expanded (E, most likely beta-strand) conformation

Table 8
Context pairwise contact potential for the ACC type of side chain interactions

ACC	GLY	ALA	SER	CYS	VAL	THR	ILE	PRO	MET	ASP	ASN	LEU	LYS	GLU	GLN	ARG	HIS	PHE	TYR	TRP
GLY	1.5	0.8	1.1	0.1	0.6	0.8	0.5	0.9	0.4	1.3	1.0	0.6	1.3	1.5	1.1	1.0	0.9	0.2	0.2	0.1
ALA	0.8	0.0	0.6	−0.4	−0.6	0.2	−0.7	0.3	−0.4	1.4	0.9	−0.5	0.9	1.1	0.8	0.6	0.3	−0.6	−0.6	−0.4
SER	1.1	0.6	0.8	0.1	0.1	0.8	−0.0	0.4	−0.0	1.1	0.8	0.1	1.0	1.4	1.0	0.7	0.7	0.1	0.1	−0.3
CYS	0.1	−0.4	0.1	−1.9	−0.8	−0.2	−0.5	−0.5	−0.9	0.9	0.5	−0.7	0.6	0.4	0.2	0.1	−0.4	−1.0	−0.8	−0.8
VAL	0.6	−0.6	0.1	−0.8	−0.8	−0.2	−1.3	−0.2	−0.8	0.7	0.4	−1.2	0.3	0.5	0.2	−0.2	−0.1	−1.2	−1.0	−1.1
THR	0.8	0.2	0.8	−0.2	−0.2	0.5	−0.4	0.2	−0.2	1.2	0.6	−0.3	1.3	0.9	0.7	0.3	0.1	−0.3	−0.3	−0.5
ILE	0.5	−0.7	−0.0	−0.5	−1.3	−0.4	−1.3	−0.2	−1.0	0.7	0.5	−1.3	0.2	0.3	0.0	−0.2	−0.4	−1.4	−1.2	−1.4
PRO	0.9	0.3	0.4	−0.5	−0.2	0.2	−0.2	0.6	−0.1	0.7	0.5	−0.2	0.7	0.7	0.5	0.2	0.1	−0.5	−0.5	−0.8
MET	0.4	−0.4	−0.0	−0.9	−0.8	−0.2	−1.0	−0.1	−1.1	0.8	0.3	−1.1	0.6	0.7	0.2	−0.1	−0.3	−1.4	−1.2	−1.3
ASP	1.3	1.4	1.1	0.9	0.7	1.2	0.7	0.7	0.8	2.1	1.3	0.9	0.8	2.0	1.4	0.3	0.3	0.4	−0.0	0.0
ASN	1.0	0.9	0.8	0.5	0.4	0.6	0.5	0.5	0.3	1.3	1.0	0.5	1.0	1.3	1.0	0.6	0.8	0.1	−0.2	−0.2
LEU	0.6	−0.5	0.1	−0.7	−1.2	−0.3	−1.3	−0.2	−1.1	0.9	0.5	−1.3	0.1	0.5	−0.0	−0.2	−0.3	−1.4	−1.2	−1.2
LYS	1.3	0.9	1.0	0.6	0.3	1.3	0.2	0.7	0.6	0.8	1.0	0.1	2.2	1.0	1.2	1.4	1.0	0.1	0.0	−0.2
GLU	1.5	1.1	1.4	0.4	0.5	0.9	0.3	0.7	0.7	2.0	1.3	0.5	1.0	1.8	1.3	0.5	0.2	0.3	0.1	−0.1
GLN	1.1	0.8	1.0	0.2	0.2	0.7	0.0	0.5	0.2	1.4	1.0	−0.0	1.2	1.3	0.9	0.7	0.9	−0.2	−0.0	−0.6
ARG	1.0	0.6	0.7	0.1	−0.2	0.3	−0.2	0.2	−0.1	0.3	0.6	−0.2	1.4	0.5	0.7	1.2	0.4	−0.4	−0.3	−0.5
HIS	0.9	0.3	0.7	−0.4	−0.1	0.1	−0.4	0.1	−0.3	0.3	0.8	−0.3	1.0	0.2	0.9	0.4	−0.3	−0.6	−0.6	−0.4
PHE	0.2	−0.6	0.1	−1.0	−1.2	−0.3	−1.4	−0.5	−1.4	0.4	0.1	−1.4	0.1	0.3	−0.2	−0.4	−0.6	−1.4	−1.4	−1.5
TYR	0.2	−0.6	0.1	−0.8	−1.0	−0.3	−1.2	−0.5	−1.2	−0.0	−0.2	−1.2	0.0	0.1	−0.0	−0.3	−0.6	−1.4	−0.9	−1.1
TRP	0.1	−0.4	−0.3	−0.8	−1.1	−0.5	−1.4	−0.8	−1.3	0.0	−0.2	−1.2	−0.2	−0.1	−0.6	−0.5	−0.4	−1.5	−1.1	−0.8

The ACC type of interactions occurs between the amino acid chains forming an antiparallel contact (A), where both contacting side chains are attached to a compact (C, most likely a helix) type of conformation

Table 9
Context pairwise contact potential for the AEE type of side chain interactions

AEE	GLY	ALA	SER	CYS	VAL	THR	ILE	PRO	MET	ASP	ASN	LEU	LYS	GLU	GLN	ARG	HIS	PHE	TYR	TRP
GLY	1.0	0.9	0.6	0.3	0.6	0.5	0.4	0.8	0.5	0.6	0.5	0.3	1.4	1.0	0.8	0.7	0.6	−0.2	0.1	0.2
ALA	0.9	0.9	0.9	0.3	−0.0	0.8	−0.3	0.4	−0.2	1.0	0.6	−0.2	0.9	1.2	0.4	0.7	0.5	−0.5	−0.3	−0.4
SER	0.6	0.9	1.0	0.3	0.5	0.7	0.4	0.5	0.6	0.8	0.5	0.3	1.0	0.9	0.6	0.4	0.4	0.1	−0.0	−0.4
CYS	0.3	0.3	0.3	−2.4	−0.1	0.1	−0.5	0.3	−0.5	0.4	0.1	−0.7	0.5	0.6	0.0	0.1	−0.4	−0.8	−0.6	−1.4
VAL	0.6	−0.0	0.5	−0.1	−0.5	0.4	−0.7	0.2	−0.6	1.1	0.8	−0.8	0.6	0.7	0.4	0.3	0.0	−1.0	−0.5	−1.3
THR	0.5	0.8	0.7	0.1	0.4	0.7	0.2	0.7	0.2	0.9	0.3	0.2	1.1	1.0	0.6	0.4	0.2	−0.1	−0.1	−0.5
ILE	0.4	−0.3	0.4	−0.5	−0.7	0.2	−0.9	0.1	−0.8	0.6	0.4	−1.1	0.4	0.7	0.2	0.2	0.1	−1.3	−0.9	−1.3
PRO	0.8	0.4	0.5	0.3	0.2	0.7	0.1	0.3	0.1	0.5	0.3	0.1	1.2	0.6	0.4	0.4	0.2	−0.2	−0.4	−0.9
MET	0.5	−0.2	0.6	−0.5	−0.6	0.2	−0.8	0.1	0.1	0.4	0.1	−0.8	0.5	0.3	0.4	0.3	−0.3	−1.0	−0.8	−0.8
ASP	0.6	1.0	0.8	0.4	1.1	0.9	0.6	0.5	0.4	1.2	0.4	0.7	0.5	1.2	0.5	0.3	−0.0	0.7	0.2	0.2
ASN	0.5	0.6	0.5	0.1	0.8	0.3	0.4	0.3	0.1	0.4	0.6	0.4	1.2	0.9	0.3	0.4	−0.3	0.2	0.1	−0.5
LEU	0.3	−0.2	0.3	−0.7	−0.8	0.2	−1.1	0.1	−0.8	0.7	0.4	−0.9	0.2	0.7	0.3	0.0	−0.2	−1.2	−1.1	−1.6
LYS	1.4	0.9	1.0	0.5	0.6	1.1	0.4	1.2	0.5	0.5	1.2	0.2	1.7	0.8	0.6	1.0	0.6	0.4	0.1	0.4
GLU	1.0	1.2	0.9	0.6	0.7	1.0	0.7	0.6	0.3	1.2	0.9	0.7	0.8	1.7	0.8	0.3	0.5	0.4	0.1	−0.2
GLN	0.8	0.4	0.6	0.0	0.4	0.6	0.2	0.4	0.4	0.5	0.3	0.3	0.6	0.8	1.2	0.6	0.6	−0.2	−0.1	−0.4
ARG	0.7	0.7	0.4	0.1	0.3	0.4	0.2	0.4	0.3	0.3	0.4	0.0	1.0	0.3	0.6	1.0	0.4	−0.2	−0.3	−0.4
HIS	0.6	0.5	0.4	−0.4	0.0	0.2	0.1	0.2	−0.3	−0.0	−0.3	−0.2	0.6	0.5	0.6	0.4	−0.2	−0.5	−0.4	−0.9
PHE	−0.2	−0.5	0.1	−0.8	−1.0	−0.1	−1.3	−0.2	−1.0	0.7	0.2	−1.2	0.4	0.4	−0.2	−0.2	−0.5	−1.2	−1.0	−1.5
TYR	0.1	−0.3	−0.0	−0.6	−0.5	−0.1	−0.9	−0.4	−0.8	0.2	0.1	−1.1	0.1	0.1	−0.1	−0.3	−0.4	−1.0	−0.7	−1.1
TRP	0.2	−0.4	−0.4	−1.4	−1.3	−0.5	−1.3	−0.9	−0.8	0.2	−0.5	−1.6	0.4	−0.2	−0.4	−0.4	−0.9	−1.5	−1.1	−0.8

The AEE type of interactions occurs between the amino acid chains forming an antiparallel contact (A), where both contacting side chains are attached to expanded (E, most likely beta-strand) type of conformation

Table 10
Context pairwise contact potential for the ACE and the AEC type of side chain interactions

ACE/AEC	GLY	ALA	SER	CYS	VAL	THR	ILE	PRO	MET	ASP	ASN	LEU	LYS	GLU	GLN	ARG	HIS	PHE	TYR	TRP
GLY	1.2	1.2	1.0	0.7	0.7	0.6	0.7	1.1	0.5	0.1	0.1	0.7	1.7	1.5	1.1	0.8	0.9	0.5	0.3	**0.6**
ALA	1.0	0.0	0.9	−0.2	−0.6	0.6	−0.8	0.6	−0.5	1.0	0.9	−0.5	1.2	1.3	0.8	0.6	0.3	−0.7	−0.6	**−0.6**
SER	1.1	0.9	0.7	0.3	0.2	0.8	0.3	0.7	0.3	0.7	0.6	0.1	0.9	1.3	0.6	0.4	0.4	−0.0	−0.2	**0.0**
CYS	0.3	−0.3	0.4	−2.6	−0.8	−0.2	−0.9	−0.4	−0.9	0.5	0.2	−0.9	0.1	1.1	0.4	−0.1	−0.5	−1.3	−1.1	**−1.4**
VAL	0.4	−0.3	0.4	−0.5	−1.1	0.2	−1.3	0.0	−0.7	0.6	0.5	−1.1	0.8	0.5	0.3	−0.0	−0.2	−1.3	−1.0	**−1.2**
THR	0.8	0.3	0.5	−0.3	−0.1	0.7	−0.4	0.3	−0.3	0.8	0.3	−0.3	0.8	0.8	0.6	0.4	−0.2	−0.5	−0.5	**−0.6**
ILE	0.4	−0.5	0.7	−0.4	−1.1	0.0	−1.5	−0.2	−1.0	0.8	0.4	−1.3	0.5	0.5	0.1	−0.1	−0.2	−1.4	−1.2	**−1.3**
PRO	1.0	0.7	0.7	0.3	0.3	0.7	0.2	0.4	0.3	1.1	0.6	0.2	1.2	0.8	0.5	0.5	−0.2	−0.0	−0.4	**−0.8**
MET	0.2	−0.7	0.4	−0.5	−0.8	−0.2	−1.2	−0.1	−0.9	0.5	−0.0	−1.2	0.9	1.0	0.8	−0.2	0.0	−1.4	−1.0	**−1.3**
ASP	1.3	0.8	0.8	0.9	1.0	1.2	0.4	1.0	0.9	1.6	0.7	0.8	0.5	1.9	1.3	0.3	0.3	0.5	−0.2	**0.0**
ASN	0.9	0.7	0.7	0.3	0.6	0.4	0.3	0.4	0.4	0.7	0.5	0.3	1.1	1.1	0.8	0.4	0.4	−0.0	0.1	**−0.2**
LEU	0.5	−0.5	0.3	−0.6	−1.2	0.1	−1.4	−0.2	−0.8	1.1	0.5	−1.3	0.5	0.7	0.3	−0.0	−0.4	−1.4	−1.2	**−1.2**
LYS	1.2	1.0	1.0	0.7	0.3	0.9	−0.0	0.8	0.1	0.7	1.1	0.1	1.5	0.9	0.8	1.7	0.9	−0.2	−0.1	**−0.3**
GLU	1.3	1.0	0.9	0.6	0.4	0.8	0.2	0.9	0.5	1.8	0.9	0.3	0.8	1.6	1.6	0.3	0.5	−0.0	−0.2	**0.1**
GLN	0.7	0.7	0.7	0.1	0.1	0.6	−0.2	0.3	0.3	0.9	0.5	−0.1	1.3	1.1	1.2	0.7	0.3	−0.3	−0.4	**−0.7**
ARG	1.0	0.6	0.7	−0.0	0.1	0.2	−0.2	0.2	0.1	0.0	0.6	−0.3	1.2	0.1	0.4	0.7	0.4	−0.5	−0.5	**−0.6**
HIS	0.9	−0.0	0.3	−0.9	−0.2	0.3	−0.2	0.2	−0.1	0.2	−0.1	−0.1	1.0	0.3	1.2	0.3	−0.4	−0.7	−0.7	**−0.1**
PHE	0.3	−0.5	0.1	−0.7	−1.2	−0.2	−1.4	−0.3	−1.1	0.4	0.5	−1.3	0.1	0.4	−0.2	−0.3	−0.6	−1.7	−1.2	**−1.4**
TYR	0.4	−0.1	0.5	−0.3	−0.7	−0.1	−0.9	−0.5	−1.0	−0.0	−0.0	−0.9	0.2	0.1	−0.5	−0.5	−0.4	−1.2	−0.8	**−1.1**
TRP	−0.1	−0.5	0.0	−0.9	−1.0	−0.3	−1.3	−1.0	−1.3	0.2	−0.3	−1.2	0.1	0.1	0.0	−0.7	−0.9	−1.5	−1.2	**−1.3**

The ACE type of interactions occurs between amino acid chains forming an antiparallel contact (A), where the first contacting side chain (given in columns) is attached to the compact (C, most likely a helix) type of conformation and the second contacting side chain (given in rows) is attached to expanded (E, most likely beta-strand) conformation

2. The basic output of the CABS model is a trajectory in Cα representation. The CABS coarse-grained trajectories, or selected trajectory models, can be reconstructed to all-atom representation. The major output of the CABS-based multiscale methods (like CABS-fold or CABS-dock servers) is a set of a few models in all-atom representation (automatically selected and reconstructed). These methods also provide hundreds (CABS-fold) or thousands (CABS-dock) of predicted models in Cα trajectories that may be useful in a more thorough analysis of the prediction results and reconstructed to all-atom resolution by the user.

   There are many strategies for the reconstruction from the Cα to all-atom format; however, the method chosen should be insensitive to small local distortions of the C-alpha distances present in CABS-generated models. Based on our experience, we can recommend the following reconstruction protocols:

   - ModRefiner package [40] for combined reconstruction and optimization (handles only monomeric protein chains, employed in the CABS-fold [14] server).
   - Modeller package [41] for combined reconstruction and optimization (employed in the CABS-dock server, details of the Modeller protocol are provided in ref. [16] and the CABS-dock online tutorial http://biocomp.chem.uw.edu.pl/CABSdock/tutorial).
   - Claessens et al. [42] or BBQ [43] approach for protein backbone reconstruction followed by the second rebuilding step (side chain reconstruction) using the SCWRL program [44].
   - The last two-step protocols require a third additional optimization step, which is more demanding when BBQ is used for backbone reconstruction [45]. We tested the performance of such reconstruction and fast optimization protocols in protein structure prediction [45] and protein dynamics [30] exercises. Optimization strategies have also been reviewed in ref. [46].

3. Reconstructed and optimized all-atom models can be assessed using specially designed scoring methods. An accurate scoring function that can discriminate near-native models, or docking poses, from a large set of alternative solutions is an important component of structure prediction methodologies [47–49].
4. CABS modeling trajectories can be additionally analyzed using external tools for the structural clustering and comparison of protein models, e.g., the ClusCo package [50] or hierarchical clustering within the Bioshell package [51]. Convenient analysis of protein models usually requires superimposition of the compared models, or entire trajectories; a useful tool for that is the Theseus package [52].

5. The accuracy of de novo structure prediction by CABS-fold or CABS-dock servers depends on the accuracy of the secondary structure input. Small errors in the predicted secondary structure do not impose any serious problems, but it is (on average) safer to use underestimated ranges of regular (helices and beta strands) secondary structure fragments than overestimated ranges (for instance prediction of a single long helix for the fragment that forms two differently oriented helices). Qualitative errors of secondary structure predictions, where helical fragments are predicted as beta strands (or vice versa), are dangerous for modeling results. Fortunately, this kind of errors is rare for good bioinformatics tools for secondary structure prediction and could be eliminated by rejecting more problematic predictions.

Acknowledgments

Funding for this work was provided by the National Science Center grant [MAESTRO 2014/14/A/ST6/0008] and by the Foundation for Polish Science TEAM project (TEAM/2011-7/6) cofinanced by the EU European Regional Development Fund operated within the Innovative Economy Operational Program.

References

1. Lindorff-Larsen K, Piana S, Dror RO, Shaw DE (2011) How fast-folding proteins fold. Science 334:517–520
2. Kmiecik S, Wabik J, Kolinski M, Kouza M, Kolinski A (2014) Coarse-grained modeling of protein dynamics. In: Computational methods to study the structure and dynamics of biomolecules and biomolecular processes, vol 1, Springer, Heidelberg, Berlin, pp 55–79
3. Kmiecik S, Gront D, Kolinski M, Wieteska L, Dawid AE, Kolinski A (2016) Coarse-grained protein models and their applications. Chem Rev. doi: 10.1021/acs.chemrev.6b00163
4. Kmiecik S, Jamroz M, Kolinski A (2011) Multiscale approach to protein folding dynamics. In: Multiscale approaches to protein modeling. Springer, New York, pp 281–293
5. Levitt M, Warshel A (1975) Computer simulation of protein folding. Nature 253:694–698
6. Rohl CA, Strauss CEM, Misura KMS, Baker D (2004) Protein structure prediction using Rosetta. In: Methods Enzymol, vol 383, Academic Press, pp 66–93
7. Mao B, Tejero R, Baker D, Montelione GT (2014) Protein NMR structures refined with Rosetta have higher accuracy relative to corresponding X-ray crystal structures. J Am Chem Soc 136:1893–1906
8. Roy A, Kucukural A, Zhang Y (2010) I-TASSER: a unified platform for automated protein structure and function prediction. Nat Protoc 5:725–738
9. Yang J, Yan R, Roy A, Xu D, Poisson J, Zhang Y (2015) The I-TASSER suite: protein structure and function prediction. Nat Methods 12:7–8
10. Kolinski A (2004) Protein modeling and structure prediction with a reduced representation. Acta Biochim Pol 51:349–371
11. Kmiecik S, Kolinski A (2007) Characterization of protein-folding pathways by reduced-space modeling. Proc Natl Acad Sci U S A 104: 12330–12335
12. Liwo A, Khalili M, Scheraga HA (2005) Ab initio simulations of protein-folding pathways by molecular dynamics with the united-residue

model of polypeptide chains. Proc Natl Acad Sci U S A 102:2362–2367

13. Liwo A, He Y, Scheraga HA (2011) Coarse-grained force field: general folding theory. Phys Chem Chem Phys 13:16890–16901
14. Blaszczyk M, Jamroz M, Kmiecik S, Kolinski A (2013) CABS-fold: server for the de novo and consensus-based prediction of protein structure. Nucleic Acids Res 41:W406–W411
15. Kurcinski M, Jamroz M, Blaszczyk M, Kolinski A, Kmiecik S (2015) CABS-dock web server for the flexible docking of peptides to proteins without prior knowledge of the binding site. Nucleic Acids Res 43:W419–W424
16. Blaszczyk M, Kurcinski M, Kouza M, Wieteska L, Debinski A, Kolinski A, Kmiecik S (2016) Modeling of protein–peptide interactions using the CABS-dock web server for binding site search and flexible docking. Methods 93:72–83
17. Jamroz M, Kolinski A, Kmiecik S (2014) Protocols for efficient simulations of long-time protein dynamics using coarse-grained CABS model. Methods Mol Biol 1137:235–250
18. Kar P, Feig M (2014) Recent advances in transferable coarse-grained modeling of proteins. Adv Protein Chem Struct Biol 96:143–180
19. Kolinski A, Bujnicki JM (2005) Generalized protein structure prediction based on combination of fold-recognition with de novo folding and evaluation of models. Proteins 61(Suppl 7):84–90
20. Skolnick J, Zhang Y, Arakaki AK, Kolinski A, Boniecki M, Szilagyi A, Kihara D (2003) TOUCHSTONE: a unified approach to protein structure prediction. Proteins 53(Suppl 6):469–479
21. Debe DA, Danzer JF, Goddard WA, Poleksic A (2006) STRUCTFAST: protein sequence remote homology detection and alignment using novel dynamic programming and profile-profile scoring. Proteins 64:960–967
22. Boniecki M, Rotkiewicz P, Skolnick J, Kolinski A (2003) Protein fragment reconstruction using various modeling techniques. J Comput Aided Mol Des 17:725–738
23. Jamroz M, Kolinski A (2010) Modeling of loops in proteins: a multi-method approach. BMC Struct Biol 10:5
24. Kmiecik S, Jamroz M, Kolinski M (2014) Structure prediction of the second extracellular loop in G-protein-coupled receptors. Biophys J 106:2408–2416
25. Sali A, Blundell TL (1993) Comparative protein modelling by satisfaction of spatial restraints. J Mol Biol 234:779–815
26. Latek D, Kolinski A (2011) CABS-NMR—de novo tool for rapid global fold determination from chemical shifts, residual dipolar couplings and sparse methyl-methyl NOEs. J Comput Chem 32:536–544
27. Kurcinski M, Kolinski A, Kmiecik S (2014) Mechanism of folding and binding of an intrinsically disordered protein as revealed by ab initio simulations. J Chem Theor Comput 10:2224–2231
28. Steczkiewicz K, Zimmermann MT, Kurcinski M, Lewis BA, Dobbs D, Kloczkowski A, Jernigan RL, Kolinski A, Ginalski K (2011) Human telomerase model shows the role of the TEN domain in advancing the double helix for the next polymerization step. Proc Natl Acad Sci U S A 108:9443–9448
29. Kurcinski M, Kolinski A (2007) Hierarchical modeling of protein interactions. J Mol Model 13:691–698
30. Kmiecik S, Gront D, Kouza M, Kolinski A (2012) From coarse-grained to atomic-level characterization of protein dynamics: transition state for the folding of B domain of protein A. J Phys Chem B 116:7026–7032
31. Kmiecik S, Kolinski A (2011) Simulation of chaperonin effect on protein folding: a shift from nucleation-condensation to framework mechanism. J Am Chem Soc 133:10283–10289
32. Kmiecik S, Kolinski A (2008) Folding pathway of the b1 domain of protein G explored by multiscale modeling. Biophys J 94:726–736
33. Kmiecik S, Kurcinski M, Rutkowska A, Gront D, Kolinski A (2006) Denatured proteins and early folding intermediates simulated in a reduced conformational space. Acta Biochim Pol 53:131–144
34. Wabik J, Kmiecik S, Gront D, Kouza M, Kolinski A (2013) Combining coarse-grained protein models with replica-exchange all-atom molecular dynamics. Int J Mol Sci 14:9893–9905
35. Jamroz M, Orozco M, Kolinski A, Kmiecik S (2013) Consistent view of protein fluctuations from all-atom molecular dynamics and coarse-grained dynamics with knowledge-based force-field. J Chem Theor Comput 9:119–125
36. Jamroz M, Kolinski A, Kmiecik S (2013) CABS-flex: server for fast simulation of protein structure fluctuations. Nucleic Acids Res 41:W427–W431
37. Jamroz M, Kolinski A, Kmiecik S (2014) CABS-flex predictions of protein flexibility compared with NMR ensembles. Bioinformatics 30:2150–2154
38. Zambrano R, Jamroz M, Szczasiuk A, Pujols J, Kmiecik S, Ventura S (2015) AGGRESCAN3D

(A3D): server for prediction of aggregation properties of protein structures. Nucleic Acids Res 43:W306–W313

39. McGuffin LJ, Bryson K, Jones DT (2000) The PSIPRED protein structure prediction server. Bioinformatics 16:404–405
40. Xu D, Zhang Y (2011) Improving the physical realism and structural accuracy of protein models by a two-step atomic-level energy minimization. Biophys J 101:2525–2534
41. Eswar N, Webb B, Marti-Renom MA, Madhusudhan MS, Eramian D, Shen M-Y, Pieper U, Sali A (2007) Comparative protein structure modeling using MODELLER. Curr Protoc Protein Sci 2:1–31
42. Claessens M, Van Cutsem E, Lasters I, Wodak S (1989) Modelling the polypeptide backbone with "spare parts" from known protein structures. Protein Eng 2:335–345
43. Gront D, Kmiecik S, Kolinski A (2007) Backbone building from quadrilaterals: a fast and accurate algorithm for protein backbone reconstruction from alpha carbon coordinates. J Comput Chem 28:1593–1597
44. Wang Q, Canutescu AA, Dunbrack RL Jr (2008) SCWRL and MolIDE: computer programs for side-chain conformation prediction and homology modeling. Nat Protoc 3:1832–1847
45. Kmiecik S, Gront D, Kolinski A (2007) Towards the high-resolution protein structure prediction. Fast refinement of reduced models with all-atom force field. BMC Struct Biol 7:43
46. Gront D, Kmiecik S, Blaszczyk M, Ekonomiuk D, Kolinski A (2012) Optimization of protein models. Wiley Interdiscipl Rev-Comput Mol Sci 2:479–493
47. Kim H, Kihara D (2015) Protein structure prediction using residue- and fragment-environment potentials in CASP11. Proteins. doi:10.1002/prot.24920
48. Zimmermann MT, Leelananda SP, Kloczkowski A, Jernigan RL (2012) Combining statistical potentials with dynamics-based entropies improves selection from protein decoys and docking poses. J Phys Chem B 116:6725–6731
49. Faraggi E, Kloczkowski A (2014) A global machine learning based scoring function for protein structure prediction. Proteins 82:752–759
50. Jamroz M, Kolinski A (2013) ClusCo: clustering and comparison of protein models. BMC Bioinformatics 14:62
51. Gront D, Kolinski A (2005) HCPM—program for hierarchical clustering of protein models. Bioinformatics 21:3179–3180
52. Theobald DL, Steindel PA (2012) Optimal simultaneous superpositioning of multiple structures with missing data. Bioinformatics 28:1972–1979
53. Kolinski A, Betancourt MR, Kihara D, Rotkiewicz P, Skolnick J (2001) Generalized comparative modeling (GENECOMP): a combination of sequence comparison, threading, and lattice modeling for protein structure prediction and refinement. Proteins 44:133–149
54. Schwede T, Diemand A, Guex N, Peitsch MC (2000) Protein structure computing in the genomic era. Res Microbiol 151:107–112
55. Raveh B, London N, Schueler-Furman O (2010) Sub-angstrom modeling of complexes between flexible peptides and globular proteins. Proteins 78:2029–2040

Chapter 9

Assessing Predicted Contacts for Building Protein Three-Dimensional Models

Badri Adhikari, Debswapna Bhattacharya, Renzhi Cao, and Jianlin Cheng

Abstract

Recent successes of contact-guided protein structure prediction methods have revived interest in solving the long-standing problem of ab initio protein structure prediction. With homology modeling failing for many protein sequences that do not have templates, contact-guided structure prediction has shown promise, and consequently, contact prediction has gained a lot of interest recently. Although a few dozen contact prediction tools are already currently available as web servers and downloadables, not enough research has been done towards using existing measures like precision and recall to evaluate these contacts with the goal of building three-dimensional models. Moreover, when we do not have a native structure for a set of predicted contacts, the only analysis we can perform is a simple contact map visualization of the predicted contacts. A wider and more rigorous assessment of the predicted contacts is needed, in order to build tertiary structure models. This chapter discusses instructions and protocols for using tools and applying techniques in order to assess predicted contacts for building three-dimensional models.

Key words Protein contact assessment, Contact-guided ab initio prediction

1 Introduction

In the last few years, prediction of protein residue contacts has shown improvement in the field of ab initio protein structure prediction [1–4]. Tertiary structure predictions can benefit from the use of predicted contacts for many reasons. One of the most crucial values of contact-guided protein structure prediction has to do with contact connection information that can give us a better look at the mechanism which causes proteins to fold. For successful ab initio modeling using contacts, the quality of predicted contacts is the most important consideration because for almost all proteins, accurate contact predictions result in correct folds. Since the field of contact prediction is still developing, the question of how the predicted contacts can be appropriately assessed so that we can use them to build three-dimensional models is still subject to discussion, debate and much more research. Given a set or sets of

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_9,

predicted contacts for a protein sequence, we are exploring novel and potentially transformative techniques to utilize these contacts for building tertiary structure models for proteins. Current techniques include visualization using contact maps, and evaluation using various measures like precision and coverage.

Those researchers exploring the task of building tertiary structure models like Rosetta [5], I-Tasser [6], and RBO-alph [7]—all have started to incorporate contacts to aid their methods. Those focusing on building 3D models primarily using predicted contacts have developed new methods like FRAGFOLD [2], EVFOLD [3], and CONFOLD [5]. For existing structure prediction systems like Rosetta and I-Tasser, a few predicted contacts can be used as additional information to guide the ab initio folding process. On the other hand, it is also important to have a decent number of contacts (for example, those ranging from L/2 to L, where L is the length of the protein) to guide the modeling process to predict protein folds to facilitate tools which build models from scratch, like EVFOLD and CONFOLD. This second group of modeling tools dedicated to building models from scratch is ideal for studying the quality of predicted contacts because they solely rely on contacts to build models, and the results are not biased by other prediction information such as the availability of good fragments.

Whether or not a native structure exists for a set of predicted contacts, a good way to evaluate the predicted contacts is to directly build three-dimensional models using them and observe the 3D models. In this chapter, we will discuss the protocols for using one such method, CONFOLD, available at http://protein.rnet.missouri.edu/confold/. We will also discuss the available tools and techniques for precision and coverage calculations, including improved contact map visualizations. For convenience, we have built a web server, CONASSESS, available at http://cactus.rnet.missouri.edu/conassess/.

2 Materials

When the true structure of a protein is known, there are widely used tools to evaluate predicted contacts. When no true structure exists, the only analysis we can perform is visualizations to check the proportion of contact types and ensure a good coverage. The three contact types—short, medium, and long-range—are defined using sequence minimum sequence separation of at least 6, 12, and 24 residues, respectively. For instance, contacts with residue sequence separation more than 11 and less than 24 are defined as medium range contacts. Among these three contact types, long-range contacts are the most important for folding purposes and are also the most difficult to predict [8, 9] (*see* **Note 2**). To help study the coverage of predicted contacts we introduce 1D visualization of the

contact coordination number, and to check the proportion of contact types, we discuss improved contact map visualization. When the 3D structure of the sequence is known we can simply calculate precision and coverage of a certain number of selected top contacts as a primary evaluation. Besides precision and coverage, other measures like spread [3], mean error, and *Xd* [10–12] are important to obtain a highly accurate three-dimensional fold of a protein. We will discuss these techniques in the following sections. All evaluation methods, including precision, *Xd*, coverage, and both 1D and 2D visualization techniques are implemented in our CONASSESS web server.

2.1 Contact Visualization

Visualizing three-dimensional information in lower dimensions is challenging, but as long as we are interested in a particular aspect of the data, simpler visualizations in lower dimensions can be easy and yet effective. A simple technique for 1D representation of predicted residue contacts is to assign numbers to each residue so that the numbers represent the number of contacts that the residue is involved in, also known as the coordination number. For a 1D visualization by showing a single character decimal number below the sequence, residues that are involved in less than nine contacts can be assigned numbers from 1 to 9, and the residues that are involved in more than nine contacts may be assigned a special character like "*." This visualization technique can show if contacts are clustered in a specific region or spread around evenly, and it is effective when we have fewer contacts to analyze, for example L/10, L/5, L/2, L, or even 2L contacts, where L is the length of the protein. In addition, it is also convenient to compare contacts predicted by multiple sources (*see* **Note 2**). An example of a 1D visualization is shown in Fig. 1. The limitation of this visualization technique is that it becomes ineffective when dealing with residues with too many of predicted contacts because all residues will be assigned the "*" character.

Two-dimensional visualization of contacts using contact maps with the help of tools like CMview [13] has been in existence for many decades in the field of proteomics (*see* **Note 4**). A slightly different version of the existing contact maps can help us differentiate long-range contacts from others, and also compare contacts from multiple sources, *see* Fig. 2. To separate the various contact types, different colors may be used for each of the three contact types. Furthermore,

```
         10        20        30        40        50
         |         |         |         |         |
DCCLGYTDRILHPKFIVGFTRQLANEGCDINAIIFHTKKKLSVCANPKQTWVKYIVRLLSK [sequence]
1----1-1----1-11--21-1----1----2212---------111-------1--1-1- [ccmpred.rr: 12 contacts touching 20 residues]
11---1------1--1--21---1--12--2--122-----1--1---------2----1- [pcons_confuzz.rr: 12 contacts touching 18 residues]
------------1--1--31-11---1---221-21--------11-1------2--1-1- [psicov.rr: 12 contacts touching 18 residues]
```

Fig. 1 An example 1D visualization of coordination numbers for predicted contacts. Top L/5 contacts predicted for the protein 1m8a, using three sources of predicted contacts (CCMPRED, PCONS-CONFUZZ, AND PSICOV), are compared in the lines below the sequence row. The numbers below each residue represent the number of contacts that the residue is involved with, such that every contact increases this number for two residues

for each contact prediction source, separate symbols may be used. This allows us to conveniently compare specific contact types of different sources such as long range contacts predicted by two sources. An example of such a contact map visualization is shown in Fig. 2.

2.2 Contact Evaluation

Precision and coverage are two of the most established methods for evaluating predicted protein contacts against a true structure. It is necessary to measure both precision and coverage because often they complement each other (*see* **Note 6**). If we evaluate just a few top predicted contacts and observe their high precision, it does not necessarily imply high coverage. Precision, as shown in Eq. 1, is calculated as the ratio of the number of correctly predicted contacts and the total number of predicted contacts. Coverage, however, may be calculated in three ways. The simplest technique, as shown in Eq. 2, is to calculate the number of predicted contacts divided by the total number of contacts in the native structure [10, 11, 14]. Coverage calculated in this way may result in a relatively smaller value because it is fairly difficult to precisely predict all of the (often redundant) neighboring contacts in the native structure.

Precision,

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{1}$$

where TP is true positive and FP is false positive.

Coverage,

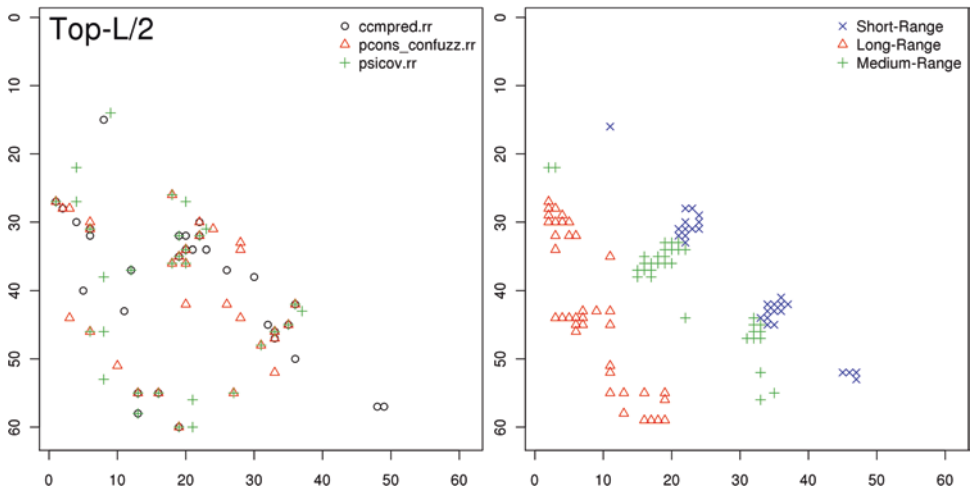


Fig. 2 Examples of contact map visualizations. *Top* L/2 contacts predicted for the protein 1m8a using three different contact prediction sources (*left*). Short-range, medium-range, and long-range true contacts in the native structure of the protein 1m8a are shown in different colors (*right*)

$$\text{Cov} = \frac{100 \times \text{TP}}{\text{no. of native contacts}} \tag{2}$$

$$\text{Mean Error} = \sum \frac{\text{error}}{\text{TP} + \text{FP}} \quad \begin{cases} \text{error} = d - T \text{ if } d > T \\ \text{error} = 0 \quad \text{if } d \le T \end{cases} \tag{3}$$

where d is the actual distance of a contact in a native structure, and T is the distance threshold of the predicted contact.

Distance distribution,

$$X_d = \sum_{i=1}^{15} \frac{Ppi - Pai}{15 \times di} \tag{4}$$

where *Ppi* is the fraction of predicted contacts in bin *i*, and *Pia*—the fraction of all residue pairs in bin *i*.

The second method of evaluating coverage is distance distribution: *Xd* [10–12], measures the weighted harmonic average difference between the distance distribution of predicted contacts and the all-pairs (Eq. 4). Fifteen distance bins cover the range from 0 to 60 Å. The 15 bins include ranges of distances from 0 to 4 Å, 4 to 8 Å, 8 to 12 Å, etc. This score estimates the deviation of the distribution of distances in the list of contacts from the distribution of distances in all pairs of residues in the protein (*see* **Note 5**). The Protein Structure Prediction Center sponsored by the US National Institute of General Medical Sciences (NIH/NIGMS) has been holding biannual meetings featuring preplanned Critical Assessment of protein Structure Prediction (CASP) experiments with specific goals and instructions since 1994. Their goal has been to assist in advancing the current state of the art in protein structure prediction by identifying annual progress and helping to determine where future effort should be most productively focused. CASP6 (2004) focused on precision and *Xd*, and the data from that experiment has been consistently used for contact evaluation in all the CASP competitions afterwards, including the CASP10 (2012) competition. Marks et al. introduced another method for calculating coverage by calculating the spread of contacts [3]. This is computed as the mean of the distances from every experimental (crystal structure) contact to the nearest predicted contact in the 2D contact map.

2.3 Building 3D Models Using Contacts

The emerging success of contact prediction methods demand more research towards building systems that build 3D models from contacts, and one such state-of-the-art method is CONFOLD [1], designed specifically for predicted contacts. The principal idea behind CONFOLD is to build models in two stages to detect self-conflicting contacts. In the first stage, all input contacts are used to build 3D models and the top ranking model in this stage is checked to find the contacts that are not satisfied with a looser definition of a contact. Then the unsatisfied contacts are ignored, in the second

stage, as the process of building models begins again. Besides removing self-conflicting contacts in the second stage, predicted strands that are close enough are paired to form beta-sheets in order to improve the accuracy and quality of the models. CONFOLD uses an algorithm known as "distance geometry simulated annealing protocol" implemented in a customized version of a well-established structure determination tool known as the CNS suite [15, 16].

For building 3D models using predicted contacts, the CONFOLD web server may be utilized. On a benchmark data set of 150 globular proteins, contacts predicted by PSICOV [17] were used as input to build 3D models using CONFOLD, to find the Pearson correlation coefficient between the precision of top L/2 contacts and the TM-score of the best models as 0.7. This high correlation suggests that the folding method of CONFOLD is primarily contact-guided, which is ideal for studying the folding information captured in predicted contacts. Unlike many other reconstruction tools, an important feature of CONFOLD is that it can accept secondary structure information (Helix and Strand predictions) along with beta sheet pairing information. This feature may be exploited by predicting secondary structure using a variety of tools in order to obtain a pool of different secondary structures, and then using them in conjunction with the predicted contacts. For building models, CONFOLD transforms the input contacts and secondary structures into restraints for guiding the modeling. In addition, the relative weights between contact restraints and secondary structure restraints can be adjusted, giving us more control over our model building experiments.

Besides CONFOLD, other reconstruction tools may be used for using contacts to build models. Fragment-based ab initio tools like Rosetta and FRAGFOLD [2] can improve their ab initio models using a just few residue contacts. Both ROSETTA and FRAGFOLD can be downloaded and run locally. The template modeling tool, Modeller [18], also accepts secondary structures and contacts as input restraints for building 3D models even though it is not well suited for ab initio modeling [1]. Reconstruction tools like FT-COMAR [19, 20] and Reconstruct [21] have shown state-of-the art performance with true contacts and can accept predicted contacts as input. However, they are not rigorously tested with predicted contacts.

3 Methods

To build 3D models for a given input sequence, we need to decide how many contacts or determine an appropriate maximum number of contacts to consider. When reconstructing using true contacts, we know that this number must be at least 8% of the native

contacts [22]. For predicted contacts, although current evaluations consider the top L/2, top L/5, and top L/10 [10, 11] (L being the length of the protein), the number of predicted contacts needed for reconstruction of a protein depends on many factors. These factors include (a) contact prediction method, (b) model building tool, (c) whether or not additional information is used for modeling, and also (d) the protein structure's reconstruct-ability. Some recent studies have considered a range of the number of contacts for building models [1–3] and the authors have suggested using up to top L contacts.

Once the number of contacts is decided, visualization techniques like 2D contact maps help to investigate the coverage and proportion of the three contact types (short-, medium-, and long-range). Upon visualization, if we observe that most of the contacts are clustered only around a specific region of the sequence, we can expect the coverage to be low. Similarly, visualization can also depict the proportion of the three contact types. For building 3D models, it is better to have a mixture of all the three contact types making sure that at least some long-range contacts are included. In addition, it may be important to observe the spread of only the long-range contact as they are considered the most important of the three. When multiple methods are used for contact prediction, visualizations also help to observe the overlaps in predicted sets of contacts. In the case that we have the true structure, however, the selected number of top predicted contacts needs to be evaluated by calculating precision and coverage. In addition, to check how much folding information is captured by the contacts, models may be built using CONFOLD. Below we present the steps for contact assessment.

1. Decide on a tool (or tools) for contact prediction. The results of searching for homologous sequences and templates may suggest whether a template-based method, a machine learning-based method like DNcon [14], NNcon [23], or SVMcon [24], a coevolution-based method like CCMpred [25], EPC-map [26], or FreeContact [27], or a hybrid contact prediction method like MetaPSICOV [28] or PconsC2 [29] is appropriate.
2. Determine the number of contacts for assessments. Typically, top L/10, top L/5, top L/2, or top L may be selected.
3. Visualize the predicted contacts using 1D and 2D methods, *see* Figs. 1 and 2. For a quick visualization submit the predicted contact to the CONASSESS web server. If contacts are predicted using multiple sources, the .RR files should be zipped into a single zip file and then be uploaded.
4. In case a native structure is available, calculate precision, coverage, *Xd*, and mean error using Eqs. 1 through 4 and the measure spread (*see* **Note 3**).
5. Build models using CONFOLD

(a) (Optional) Predict secondary structure for the input sequence. If the sequence does not have any homologous sequences, machine learning tools like SSPro [30] may be used. On the other hand, if many homologous sequences exist, sequence-based tools like Psi-blast based secondary structure prediction (PSIPRED) [31] may be considered.

(b) (Optional) Predict beta sheet pairing information using the predicted secondary structure prediction obtained in (a).

(c) Submit the input sequence, predicted contacts (obtained in **step 1**), secondary structure and beta pairing information (obtained from **steps 5a** and **5b** above) to the CONFOLD web server at http://protein.rnet.missouri.edu/confold/.

(d) Visualize the models by downloading the models from the link received in the e-mail.

4 Case Studies

One useful application of CONASSESS is to analyze predicted contacts when a native structure does not exist for the input sequence. As a case study, consider a 163 residue long CASP11 RR target T0763. The predicted contacts are available in a zip file preloaded in Set 5 of the pre-curated examples in the CONASSESS web server. Assuming that we do not have a native PDB, we may empty the "native pdb" text field. Once the job is submitted to CONASSESS, it calculates the number of long-range contacts and different numbers of top L/10 to top 2L contacts for each of the predicted contacts in the submitted set. The contact map of top L/10 contacts, shown in Fig. 3, shows the overlap in contacts predicted by the various contact prediction groups (or predictors). Upon observing the visualization of coordination numbers of the top L/10 contacts, we notice that the contacts predicted by most of the groups are well distributed over the sequence, but we may also notice some groups whose predicted contacts are clustered in 3 or 4 regions of the sequence. We may also guess that building models using such clustered contacts does not yield good models, and we may need to select the top L/5 or even the top L contacts from such predictors for model building purposes. In addition, if we plan to build models by combining the contacts predicted by all predictors, we may notice from the contact maps that the total number of contacts may be too many to efficiently work with if we select more than the top L/5 contacts.

Another important application of CONASSESS is to evaluate accuracy of predicted contacts against a known native

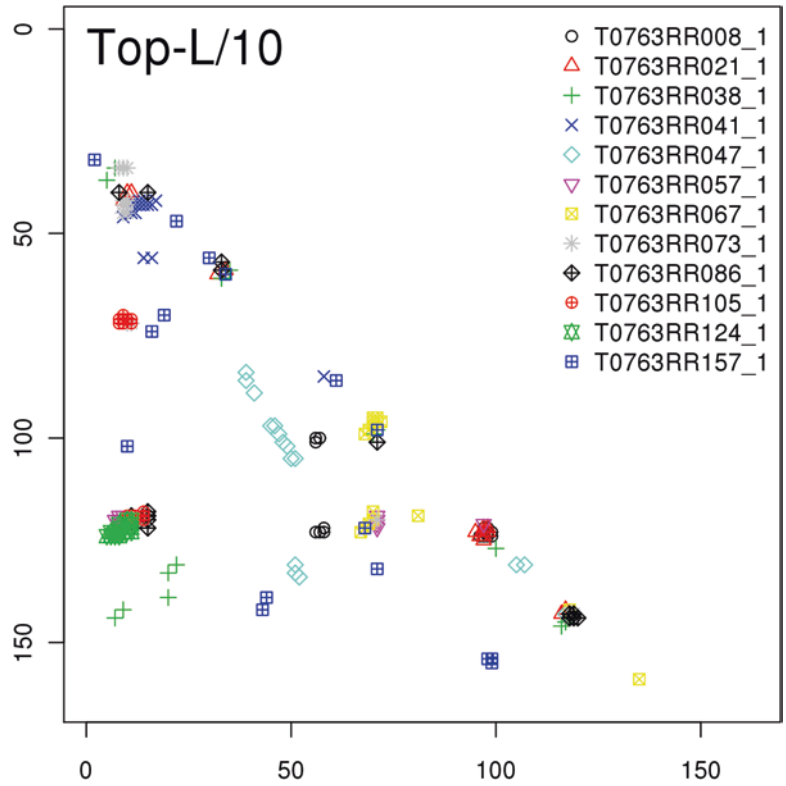


Fig. 3 An screenshot of CONASSESS server's output contact map for the top L/10 contacts predicted for the CASP11 RR target

structure from multiple, complementary perspectives. We may use contacts predicted from a diverse array of methods and readily compare them. As a second case study, let us consider a 145-residue protein (pdb id 1a3a) available in Set 2 of the examples in the CONASSESS web server. If we predict contacts from the sequence using three state-of-the-art approaches like CCMpred [25], PSICOV [17], and PconsC [32] for this protein in a pseudo-blind fashion, we can use CONASSESS web server to evaluate the accuracy of these predicted contacts using measures such as precision, mean error, coverage, distance distribution, and spread. We can then derive some interesting insights by simple visual inspections in addition to detailed, numerical data made available through CONASSESS web server in the form of tables. Fig. 4 shows a representative example for protein 1a3a. In this case the precision of the predicted contacts by PSICOV is higher compared to CCMpred or PconsC when less top ranked contacts are considered. However, CCMpred or PconsC tends to have higher precision of predicted contacts when more top ranked contacts are considered.

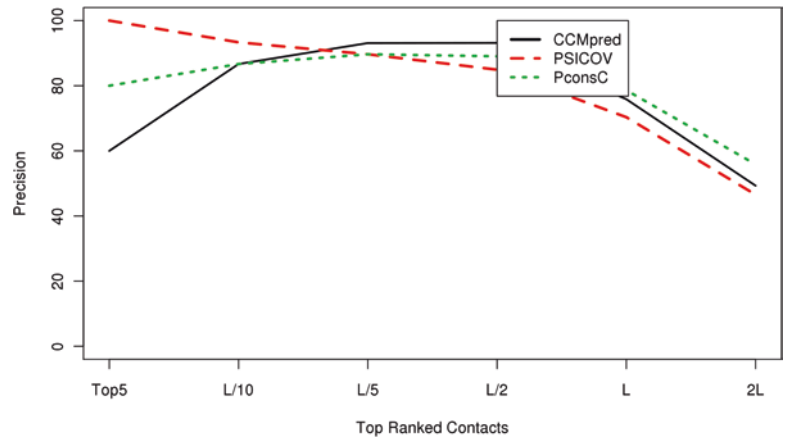


Fig. 4 Precision of predicted contact using three different methods at varied number of top ranked contacts for a representative protein (pdb id 1a3a)

5 Notes

1. Many contact prediction tools often predict many short-range contacts as the confident predictions ranked at the top. Many of these short-range contacts (contacts with small residue sequence separation, usually less than 6 residues) are not always useful if they are the only ones that are used for building models. In a set of top predicted contacts, if the proportion of short-range contacts is high compared to the proportion of long-range contacts, we may need to investigate more to find out if the 3D structure indeed has no (or too few) long-range contacts. The CONASSESS web server may be utilized to check the percentage of short-range contacts in a given set of predicted contacts.
2. Many contact prediction tools may predict contacts clustered in only one or two specific regions of the sequence/structure such as for beta-sheet proteins. Predicting secondary structure using existing tools and visualizing the coordination numbers using a simple 1D technique helps to identify this so that we are able to include more contacts to ensure good coverage.
3. Before contact assessment, make sure that the sequences of predicted contacts and the sequence of the native model are all same. Even if the sequences look similar, scan through the pdb file at least once to check (a) if the file has multiple models (b) if gaps appear in the residue numbering, (c) if residue insertions have been added, and (d) if alternate residues are being used.
4. Visual comparison of contact maps can be misleading. Two contact maps may look similar in contact maps, but the quantitative evaluations can be quite different.

5. The distance distribution score, *Xd*, can have negative values as well. This usually means that the quality of contacts is not good enough because values much higher than 0 usually refer to better contact predictions.
6. It is not surprising to observe high precision values with almost zero coverage for some predicted contacts. For instance, if we are evaluating the top five predicted contacts, and they all are correct, we will get a 100% precision score, but the coverage may be low because five contacts can be too few compared to the total number of contacts in the protein.

References

1. Adhikari B, Bhattacharya D, Cao R, Cheng J (2015) CONFOLD: residue-residue contact-guided ab initio protein folding. Proteins 83(8):1436–1449
2. Kosciolek T, Jones DT (2014) De novo structure prediction of globular proteins aided by sequence variation-derived contacts. PLoS One 9(3):e92197
3. Marks DS, Colwell LJ, Sheridan R, Hopf TA, Pagnani A, Zecchina R, Sander C (2011) Protein 3D structure computed from evolutionary sequence variation. PLoS One 6(12):e28766
4. Bhattacharya D, Cheng J (2015) De novo protein conformational sampling using a probabilistic graphical model. Sci Rep 5:16332
5. Leaver-Fay A, Tyka M, Lewis SM, Lange OF, Thompson J, Jacak R, Kaufman K, Renfrew PD, Smith CA, Sheffler W, Davis IW, Cooper S, Treuille A, Mandell DJ, Richter F, Ban YE, Fleishman SJ, Corn JE, Kim DE, Lyskov S, Berrondo M, Mentzer S, Popovic Z, Havranek JJ, Karanicolas J, Das R, Meiler J, Kortemme T, Gray JJ, Kuhlman B, Baker D, Bradley P (2011) ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules. Methods Enzymol 487:545–574. doi:10.1016/b978-0-12-381270-4.00019-6
6. Zhang Y (2008) I-TASSER server for protein 3D structure prediction. BMC Bioinformatics 9(1):40
7. Mabrouk M, Putz I, Werner T, Schneider M, Neeb M, Bartels P, Brock O (2015) RBO Aleph: leveraging novel information sources for protein structure prediction. Nucleic Acids Res 43(W1):W343–W348. doi:10.1093/nar/gkv357
8. Chen J, Zhang L, Jing L, Wang Y, Jiang Z, Zhao D (2003) Predicting protein structure from long-range contacts. Biophys Chem 105(1):11–21
9. Gromiha MM, Selvaraj S (1999) Importance of long-range interactions in protein folding. Biophys Chem 77(1):49–68
10. Monastyrskyy B, Fidelis K, Tramontano A, Kryshtafovych A (2011) Evaluation of residue–residue contact predictions in CASP9. Proteins 79(S10):119–125
11. Monastyrskyy B, D'Andrea D, Fidelis K, Tramontano A, Kryshtafovych A (2014) Evaluation of residue–residue contact prediction in CASP10. Proteins 82(S2):138–153
12. Ezkurdia I, Grana O, Izarzugaza JM, Tress ML (2009) Assessment of domain boundary predictions and the prediction of intramolecular contacts in CASP8. Proteins 77(S9):196–209
13. Vehlow C, Stehr H, Winkelmann M, Duarte JM, Petzold L, Dinse J, Lappe M (2011) CMView: interactive contact map visualization and analysis. Bioinformatics 27(11):1573–1574
14. Eickholt J, Cheng J (2012) Predicting protein residue–residue contacts using deep networks and boosting. Bioinformatics 28(23):3066–3072
15. Brunger AT (2007) Version 1.2 of the crystallography and NMR system. Nat Protoc 2(11):2728–2733
16. Brunger AT, Adams PD, Clore GM, DeLano WL, Gros P, Grosse-Kunstleve RW, Jiang J-S, Kuszewski J, Nilges M, Pannu NS (1998) Crystallography & NMR system: a new software suite for macromolecular structure determination. Acta Crystallogr D Biol Crystallogr 54(5):905–921
17. Jones DT, Buchan DW, Cozzetto D, Pontil M (2012) PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. Bioinformatics 28(2):184–190
18. Eswar N, Webb B, Marti-Renom MA, Madhusudhan M, Eramian D, Shen M, Pieper U, Sali A (2006) Comparative protein struc-

ture modeling using Modeller. Curr Protoc Bioinformatics 5.6.1:5.6.32

19. Vassura M, Margara L, Di Lena P, Medri F, Fariselli P, Casadio R (2008) FT-COMAR: fault tolerant three-dimensional structure reconstruction from protein contact maps. Bioinformatics 24(10):1313–1315
20. Di Lena P, Vassura M, Margara L, Fariselli P, Casadio R (2009) On the reconstruction of three-dimensional protein structures from contact maps. Algorithms 2(1):76–92
21. Duarte JM, Sathyapriya R, Stehr H, Filippis I, Lappe M (2010) Optimal contact definition for reconstruction of contact maps. BMC Bioinformatics 11(1):283
22. Sathyapriya R, Duarte JM, Stehr H, Filippis I, Lappe M (2009) Defining an essence of structure determining residue contacts in proteins. PLoS Comput Biol 5(12):e1000584
23. Tegge AN, Wang Z, Eickholt J, Cheng J (2009) NNcon: improved protein contact map prediction using 2D-recursive neural networks. Nucleic Acids Res 37(suppl 2):W515–W518
24. Cheng J, Baldi P (2007) Improved residue contact prediction using support vector machines and a large feature set. BMC Bioinformatics 8(1):113
25. Seemayer S, Gruber M, Söding J (2014) CCMpred—fast and precise prediction of protein residue–residue contacts from correlated mutations. Bioinformatics 30(21):3128–3130
26. Schneider M, Brock O (2014) Combining physicochemical and evolutionary information for protein contact prediction. PLoS One 9(10), 10.1371/journal.pone.0108438
27. Kaján L, Hopf TA, Marks DS, Rost B (2014) FreeContact: fast and free software for protein contact prediction from residue co-evolution. BMC Bioinformatics 15(1):85
28. Jones DT, Singh T, Kosciolek T, Tetchner S (2014) MetaPSICOV: combining coevolution methods for accurate prediction of contacts and long range hydrogen bonding in proteins. Bioinformatics. btu791
29. Skwark MJ, Raimondi D, Michel M, Elofsson A (2014) Improved contact predictions using the recognition of protein like contact patterns. PLoS Comput Biol 10(11):e1003889
30. Cheng J, Randall AZ, Sweredoski MJ, Baldi P (2005) SCRATCH: a protein structure and structural feature prediction server. Nucleic Acids Res 33(suppl 2):W72–W76
31. McGuffin LJ, Bryson K, Jones DT (2000) The PSIPRED protein structure prediction server. Bioinformatics 16(4):404–405
32. Skwark MJ, Abdel-Rehim A, Elofsson A (2013) PconsC: combination of direct information methods and alignments improves contact prediction. Bioinformatics 29(14):1815–1816. doi:10.1093/bioinformatics/btt259

Chapter 10

Fast and Accurate Accessible Surface Area Prediction Without a Sequence Profile

Eshel Faraggi, Maksim Kouza, Yaoqi Zhou, and Andrzej Kloczkowski

Abstract

A fast accessible surface area (ASA) predictor is presented. In this new approach no residue mutation profiles generated by multiple sequence alignments are used as inputs. Instead, we use only single sequence information and global features such as single-residue and two-residue compositions of the chain. The resulting predictor is both highly more efficient than sequence alignment based predictors and of comparable accuracy to them. Introduction of the global inputs significantly helps achieve this comparable accuracy. The predictor, termed ASAquick, is found to perform similarly well for so-called easy and hard cases indicating generalizability and possible usability for de-novo protein structure prediction. The source code and a Linux executables for ASAquick are available from Research and Information Systems at http://mamiris.com and from the Battelle Center for Mathematical Medicine at http://mathmed.org.

Key words Accessible surface area, Protein structure, Sequence only prediction

1 Introduction

Accessible surface area (ASA) has played a crucial role in understanding biological function and genome relationships. The reason is that on the molecular level interactions are mostly limited to regions in the protein that are accessible to solvent or other molecules. Hence, ASA serves as a good indicator whether a given genetic or protein sequence region can potentially interact with other molecules, i.e., be involved in protein interactions and hence participate in biological function. Specific targeting of such genetic regions promises great hope for the advancement of medicine and biological understanding. Predicting ASA for a given protein sequence helps in finding these regions.

The ASA plays an important role in the stability, aggregation, enzyme activity, and binding affinity of proteins. With the help of the ASA, protein stability can be investigated by analyzing the unfolding transition as monitored by difference in the ASA between the denatured and folded state of a protein [1]. Exposure of

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_10,

hydrophobic residues has been shown to facilitate the protein aggregation and fibril formation linked to different aggregation-related diseases such as Alzheimer's and Parkinson's diseases [2, 3], Menkes disease [4], and amyotrophic lateral sclerosis [5]. Based on experimental data [6] it was suggested that the ASA of a protein can be used to compute its solvation free energy [7]. The free energy of protein solvation has been shown to be strongly correlated with the ASA in a continuum approach [8, 9]. Simple linear relationship between solvation free energy and ASA has been used in developing fast implicit solvent models [10, 11].

When protein structure is available, all-atom models [12–15] and analytical methods [16, 17] have been used to compute ASA. However, the number of protein structures solved experimentally by X-ray crystallography and NMR and deposited in the Protein Data Bank [18] is about 114,000 (as of January of 2016), while the number of known protein sequences reached 80 millions in 2014 [19]. The number of protein sequences continues to grow with remarkably faster rate than the number of protein structures deposited in PDB. The widening gap between protein sequences and structures solved experimentally motivates the prediction of ASA from the protein sequence.

Marsh and Teichmann [20] studied the role of the relative solvent ASA in the monomeric state in predicting the magnitude of binding-induced conformational changes. The study indicated that relative solvent ASA of monomer protein is strongly correlated with its flexibility and conformational changes upon binding. It has been found that the larger the relative solvent ASA the larger its conformational changes upon binding and flexibility. On the other hand, peptide flexibility [21] and the population of fibril-prone conformation in the monomeric state [22, 23] have been shown to be one of the main factors governing fibril formation times. These results not only suggest that ASA plays key role in docking of proteins, but also might open new routes to understand the docking and aggregation of proteins just by looking at the solvent ASA of monomeric forms.

De novo prediction of the ASA of a protein is a challenging problem. Understanding the process by which the protein surface is predicted from its sequence has attracted attention of researchers for many years. Wolfgang Pauli, the 1945 Nobel Prize winner in Physics, claimed that "God made the bulk; the surface was invented by the devil" emphasizing that the behavior of atoms in the surface is very difficult to understand.

The ASA [7, 24–37] is defined as the surface area of a protein or residue that is accessible to a solvent and is given here in units of square Angstroms. First described by Lee and Richards [7], ASA is typically calculated using the 'rolling ball' algorithm [38, 39]. In this approach a computational ball is rolled on the surface of the protein coordinates and probes for which and how much of the

residues are accessible, i.e., in contact with the ball. Recently Klenin et al. introduced a fast analytical method to calculate the ASA using power diagrams [40]. Other efforts in characterizing the surface of proteins were also carried out [41–47]. Since the ASA describes the amount of surface a given residue has that is accessible to the solvent or for other intermolecular interactions, it is easy to understand why the ASA is important for recognizing functional sites along the chain [48]. Accessibility is a prerequisite for a residue to be involved in external interactions. Hence knowledge of the solvent exposed residues along the residue chain of the protein can facilitate various approaches associated with function prediction and targeted mutations.

The topic of ASA prediction has been well documented in many publications [24, 27, 28, 31, 37, 49–59]. Current, actively used methods for predicting the ASA use multiple sequence alignments. A computationally costly technique that grows slower by ever growing datasets of resolved genomes. The advent of modern computational power and automated processes enables significant amounts of patient or specimen specific genetic sequencing. Moreover, researchers and clinicians may be interested in the effect of varying the genetic sequence. Both of these interests mean that in many cases one would be interested in obtaining ASA for a large number of genetic sequences. Hence, the amount of time each prediction of ASA takes may be a crucial consideration. In addition, since most methods rely heavily on mutation profiles generated by multiple sequence alignments, and since some mutated sequences would generate identical profiles, algorithms using only single sequence information may have an advantage in some cases.

2 Materials and Methods

ASAquick [37] is such an algorithm. It was programmed in FORTRAN 90 and is constructed out of several subroutines that process the data, initialize the model, and train it. It is also capable of producing predictions from existing single models or producing ensemble predictions with expected deviations. Its execution is terminal based. It was built on Ubuntu Linux, under the BASH environment. It is a window-based predictor. We predict consecutively the ASA for all residues. We aim for a fast ASA predictor, and use sequence information alone. We represent the sequence in several ways. For a given residue we include a set number of its neighbors (window) and represent the residue types with a constant length vector according to the BLOSUM62 substitution matrix representation of residues and with physicochemical parameters [28, 31]. The first representation allows for some information on the mutations between residues while the second captures some of the chemical properties of the molecules involved.

Since we do not use multiple sequence alignment profiles there is less information sharing between sequentially similar chains. Hence, we have less of a problem of over-training. Potentially one may consider then that a more inclusive representation of the entire PDB [18] will facilitate learning from as many separate instances and will improve the overall prediction accuracy. This is indeed strongly suspected and is the topic of future work. For our case here, to maintain a reasonable comparison with previous methods we use a PISCES list [60, 61] of non-homologous protein chains with resolution better than 3 Å and sequence identity lower than 40%. Structure files were downloaded for these chains from the PDB. This non-redundant set was 14,361 proteins in size. A concern may arise here that we are using a higher sequence identity for clustering than is done in the training of most profile based predictors. One should realize that the common value of 25% sequence identity comes about in an attempt to eliminate redundancy in the training sets. Since proteins with sequence identity greater than 25% produce similar PSSMs. Hence, using more than one such sequence would mean a degenerate training example. However, ASAquick is trained at the sequence level. At this level, training examples are unique at 40% and even higher sequence identity.

For each residue in these chains we calculate the ASA using the DSSP program [62]. We then find the residue-type-dependent minimum and maximum values and use these to linearly normalize the ASA to get the relative solvent accessible area (RASA) between – 1 (completely buried) and 1 (completely exposed). This scale choice was motivated by the inherent bipolarity of the neural network we use; a decision that is based on our previous work in this field [31, 63]. When searching for the maximum ASA we arbitrarily ignore the largest 1% of values. This is done to allow for exceptional behavior and various mishaps. These RASA values are the target output for GENN, and can be transformed back to their corresponding ASA values. The final predictor reports back the ASA in Angstrom squared.

As mentioned previously, inputs for ASAquick were chosen with speed in mind. Hence, no alignment profiles were used. Instead, each residue type is represented by seven parameters characterizing their physicochemical properties, and by 20 parameters related to residue mutation probabilities taken from the BLOSUM62 matrix [64–66]. For a given residue we use a window of neighboring residues as inputs to capture the local sequential environment of the residue.

To make up for some of the information lost because of abandoning multiple alignments we also found the following global parameters to be useful: the length of the chain divided by 1000, the residue type composition of the whole chain (25 values), and the directional two-residue composition (625 values). We have 25 residue types because we allow for the various characters reported

by DSSP, these include atypical residues, unknowns and chain gaps in the structure file. The output and each of the local inputs is stored in a separate file in a directory named for that chain. The global inputs are all stored in a single file in the same directory. Refer to examples provided with the distribution of GENN [67] for further information.

The parameters of the neural network were optimized using the following approach. First we tested various values for the hidden layer size and settled on a size of 31 nodes per hidden layer as a balance between speed, generalizability, and over-training avoidance. We then tested the dependence of the accuracy on the input window size. We also used this data to analyze the benefit of each of the input parameters. Refer to the original ASAquick publication for justification of parameter choice.

The accuracy of ASAquick was tested on two different datasets. In the first test we randomly partition our non-redundant set with 14,361 proteins in two parts. The first contains 5000 proteins and is used to train different neural network instances with different inputs as described below. We also use a subset of this first set for setting aside an over-protection set. The second partition contains 9361 and we use this set only to test the accuracy of prediction. We have found that inclusions of more proteins in the training set was beneficial. We use the Pearson's correlation coefficient to measure the accuracy. It is important to note that these correlations are between the ASA and not RASA. Correlation between the RASA can be significantly different.

In general we found that the prediction accuracy peaks when using an average of about five neural networks, if global input features are used. We also ranked the weights according to their accuracy on the over fit protection set and progressively added the best neural networks to the ensemble. We found that in this case too an average of around five neural networks reaches a similar maximum accuracy. Note that using global features not only improves the prediction accuracy but also allows the accuracy to reach a peak after a relatively small number of networks (around five). In contrast, the predictor without global features seems to not peak even after 24 networks. Refer to the original publication of ASAquick [37] for a more complete discussion of these results.

Further testing was conducted on the CASP10 [68] set. We used the ASAquick server and the SPINE-X [35, 69] server to predict the ASA. SPINE-X has been repeatedly found to be among the top predictors of ASA available [33]. We find that ASAquick with global features achieves a correlation of 0.66, while SPINE-X achieves a correlation coefficient of 0.71. Among the proteins in the CASP competition one subset of proteins are known as "hard targets." For the hard targets in CASP10 we find that the ASAquick correlation for ASA prediction is 0.66, while for SPINE-X the correlation coefficient is 0.68. The difference between sequence-based

and profile-based predictions is almost completely erased when considering proteins with no experimentally solved structures of homologs.

To further test the accuracy of ASAquick and its usefulness for harder targets we randomly selected 500 protein chains from PDB chains clustered at 25 % sequence identity. This set of 500 proteins has approximately 112,000 residues. For each one of these proteins we calculated a prediction using ASAquick and using SPINE-X. For each protein we also ran a BLAST search against the PDB and calculated the product of the top five e-values excluding the query protein. This measure, the e-value product, allows us to quantify the amount of similar sequences with a structure deposited in the PDB. Those query proteins with many similar sequences in the PDB will have an e-value product much smaller than one, while those with few homologs will have a value of order one or greater. We found that for proteins with similar sequence deposited in the PDB, a consistent advantage for using SPINE-X and multiple alignments approaches over ASAquick is evident. However, as we move towards harder and harder targets, we found many cases where using a pure sequence approach is advantageous and on average both perform similarly [37]. It is interesting to note that the pure sequence approach seems more useful for improving the MAE than the correlations.

As we described at the onset, one reason for designing ASAquick was to drastically increase the speed of predictions. Indeed, analysis on the time it takes to generate a prediction was carried out. It was found that ASAquick takes less than a second to produce a prediction on a single Intel Xeon E5410 at 2.33 GHz processor. On the other hand, it was found that SPINE-X takes considerably more time. Since the major bottleneck for producing prediction in SPINE-X is the generation of the profile, by default SPINE-X attempts to use four processors to carry out this job. It was found that for a protein for which ASAquick needs less than a second and a single processor, SPINE-X needs over 2 min and four processors to complete the job. Hence, ASAquick reduces the time and resources necessary to get a prediction by almost three orders of magnitude.

3 Notes

1. Source and Availability ASAquick was implemented using GENN [67], a general neural network designed to train on ad-hoc data. GENN can take any numerical input/output problem and prepare a corresponding, non-memorizing, model representation. Data can be organized in files containing individual instances or a collection of ordered instances where each line is an individual input/output target. Running ASAquick requires supplying a

sequence file in the FASTA format. A few Perl wrappers then take this sequence and quantify them to use as input features for the set of optimized weights. Because of the modularity of GENN, modifications and additions to this approach can be easily implemented. GENN and ASAquick with helpful documentation and examples are available from Research and Information Systems at http://mamiris.com, from the Battelle Center for Mathematical Medicine at http://mathmed.org, and by contacting the authors.

2. License The license for both GENN and ASAquick is available from a file called "LICENSE" in their respective directory. It allows for academic users the opportunity to use and modify it freely with proper citations while retaining some rights for commercial use.

3. Installation To install ASAquick for full functionality one should install GENN by: (1) Download the tar gzipped archive file, (2) Uncompress it: *tar -xzf GENN+ASAquick.tgz*, (3) Go to the GENN directory: *cd GENN+ASAquick*, (4) And install: *./install*. If one wishes to only use ASAquick for several predictions, one may simply type *./ASAquick your-fasta-file*, however, you will need to copy the contents of the ASAquick bin directory to your main bin directory so these scripts are found during the ASAquick run. To install GENN you must have gfortran, sed, and bc. Type these at a BASH terminal prompt to see if you have them or how to get them. Further information is available in the "README" files that are packed with the distribution.

4. Usage To run ASAquick use *ASAquick your-file* where *your-file* is either a fasta file or a dsspget [67] file. dsspget is a personal system of recording the crystallographer index, amino acid type, 8-state-dssp-ss, 3-stat-dssp-ss, ASA, phi, psi from a given PDB file. You can make such a file from a PDB file using pdb2dsspget.pl, or from a fasta file using fasta2dsspget.pl. In the second case arbitrary values are used for the unknown structural values. Both programs are provided with the distribution and should be in your /bin/ and ready to use if the installation was completed successfully. For the usage of GENN, refer to the original publication [67].

Acknowledgements

We gratefully acknowledge the financial support provided by the ational Institutes of Health (NIH) through Grants R01GM072014 and R01GM073095 to Andrzej Kloczkowski, Grant R01GM085003 to Yaoqi Zhou, and the National Science Foundation through Grant NSF MCB 1071785. This work is also supported in part by the National Health and Medical Research Council (1059775) of Australia to Yaoqi Zhou. Maksim Kouza would like to acknowledge support from the Polish Ministry of

Science and Higher Education Grant No. IP2012 016872 and "Mobilnosc Plus" No. DN/MOB/069/IV/2015; National Science Center grant [MAESTRO 2014/14/A/ST6/00088].

References

1. Myers JK, Pace CN, Scholtz JM (1995) Denaturant m values and heat capacity changes: relation to changes in accessible surface areas of protein unfolding. Protein Sci 4(10):2138–2148
2. Dobson CM (1999) Protein misfolding, evolution and disease. Trends Biochem Sci 24(9):329–332
3. Soto C (2001) Protein misfolding and disease; protein refolding and therapy. FEBS Lett 498(2–3):204–207
4. Kouza M, Gowtham S, Seel M, Hansmann UHE (2010) A numerical investigation into possible mechanisms by that the A629P mutant of ATP7A causes Menkes disease. Phys Chem Chem Phys 12(37):11390–11397
5. Munch C, Bertolotti A (2010) Exposure of hydrophobic surfaces initiates aggregation of diverse ALS-causing superoxide dismutase-1 mutants. J Mol Biol 399(3):512–525
6. Nozaki Y, Tanford C (1971) The solubility of amino acids and two glycine peptides in aqueous ethanol and dioxane solutions: establishment of a hydrophobicity scale. J Biol Chem 246(7):2211–2217
7. Lee B, Richards FM (1971) The interpretation of protein structures: estimation of static accessibility. J Mol Biol 55(3):379–IN4
8. Richards FM (1977) Areas, volumes, packing, and protein structure. Annu Rev Biophys Bioeng 6(1):151–176
9. Eisenberg D, McLachlan AD (1986) Solvation energy in protein folding and binding. Nature 319:199–203
10. Lazaridis T, Karplus M (1999) Effective energy function for proteins in solution. Prot Struct Funct Genet 35:133–152
11. Ooi T, Oobatake M, Nemethy G, Scheraga HA (1987) Accessible surface areas as a measure of the thermodynamic parameters of hydration of peptides. Proc Natl Acad Sci USA 84(10):3086–3090
12. Humphrey W, Dalke A, Schulten K (1996) VMD - visual molecular dynamics. J Mol Graph 14:33–38
13. Van Der Spoel D, Lindahl E, Hess B, Groenhof G, Mark AE, Berendsen HJC (2005) GROMACS: fast, flexible, and free. J Comput Chem 26(16):1701–1718
14. Eisenmenger F, Hansmann UHE, Hayryan S, Hu CK (2005) An enhanced version of SMMP—open-source software package for simulation of proteins. Comput Phys Commun 174(5):422–429
15. Fraczkiewicz R, Braun W (1998) Exact and efficient analytical calculation of the accessible surface areas and their gradients for macromolecules. J Comput Chem 19:319–333.
16. Busa J Jr, Hayryan S, Wu MC, Busa J, Hu CK An enhanced version of SMMP - open-source software package for simulation of proteins. Comput Phys Commun 174:422
17. Wu MC, Li MS, Ma WJ, Kouza M, Hu CK (2011) Universal geometrical factor of protein conformations as a consequence of energy minimization. Europhys Lett 96:68005
18. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat T, Weissig H, Shindyalov IN, Bourne PE (2000) The protein data bank. Nucleic Acids Res 28(1):235–242
19. The UniProt Consortium (2015) UniProt: a hub for protein information. Nucleic Acids Res 43(D1):D204–D212
20. Marsh JA, Teichmann SA (2011) Relative solvent accessible surface area predicts protein conformational changes upon binding. Structure 19(6):859–867
21. Kouza M, Co NT, Nguyen PH, Kolinski A, Li MS (2015) Preformed template fluctuations promote fibril formation: insights from lattice and all-atom models. J Chem Phys 142(14):145104
22. Nam HB, Kouza M, Hoang Z, Li MS (2010) Relationship between population of the fibril-prone conformation in the monomeric state and oligomer formation times of peptides: insights from all-atom simulations. J Chem Phys 132(16):165104
23. Chiti F, Stefani M, Taddei N, Ramponi G, Dobson CM (2003) Rationalization of the effects of mutations on peptide and protein aggregation rates. Nature 424:805–808
24. Chothia C (1974) Hydrophobic bonding and accessible surface area in proteins. Nature 248(5446):338–339
25. Holbrook SR, Muskal SM, Kim SH (1990) Predicting surface exposure of amino acids from protein sequence. Protein Eng 3(8):659–665

26. Rost B, Sander C (1994) Conservation and prediction of solvent accessibility in protein families. Proteins Struct Funct Genet 20(3):216–226
27. Moret M, Zebende G (2007) Amino acid hydrophobicity and accessible surface area. Phys Rev E 75(1):011920
28. Dor O, Zhou Y (2007) Real-spine: an integrated system of neural networks for real-value prediction of protein structural properties. Proteins Struct Funct Bioinf 68(1):76–81
29. Durham E, Dorr B, Woetzel N, Staritzbichler R, Meiler J (2009) Solvent accessible surface area approximations for rapid and accurate protein structure prediction. J Mol Model 15(9):1093–1108
30. Zhang H, Zhang T, Chen K, Shen S, Ruan J, Kurgan L (2009) On the relation between residue flexibility and local solvent accessibility in proteins. Proteins Struct Funct Bioinf 76(3):617–636
31. Faraggi E, Xue B, Zhou Y (2009) Improving the prediction accuracy of residue solvent accessibility and real-value backbone torsion angles of proteins by guided-learning through a two-layer neural network. Proteins Struct Funct Bioinf 74(4):847–856
32. Zhang T, Zhang H, Chen K, Ruan J, Shen S, Kurgan L (2010) Analysis and prediction of rna-binding residues using sequence, evolutionary conservation, and predicted secondary structure and solvent accessibility. Curr Protein Pept Sci 11(7):609–628
33. Gao J, Zhang T, Zhang H, Shen S, Ruan J, Kurgan L (2010) Accurate prediction of protein folding rates from sequence and sequence-derived residue flexibility and solvent accessibility. Proteins Struct Funct Bioinf 78(9):2114–2130
34. Nunez S, Venhorst J, Kruse CG (2010) Assessment of a novel scoring method based on solvent accessible surface area descriptors. J Chem Inf Model 50(4):480–486
35. Faraggi E, Zhang T, Yang Y, Kurgan L, Zhou Y (2012) Spine x: improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles. J Comput Chem 33:259–267
36. Wang C, Xi L, Li S, Liu H, Yao X (2012) A sequence-based computational model for the prediction of the solvent accessible surface area for α-helix and β-barrel transmembrane residues. J Comput Chem 33(1):11–17
37. Faraggi E, Zhou Y, Kloczkowski A (2014) Accurate single-sequence prediction of solvent accessible surface area using local and global features. Proteins Struct Funct Bioinf 82(11):3170–3176
38. Shrake A, Rupley J (1973) Environment and exposure to solvent of protein atoms. Lysozyme and insulin. J Mol Biol 79(2):351–371
39. Hasel W, Hendrickson TF, Still WC (1988) A rapid approximation to the solvent accessible surface areas of atoms. Tetrahedron Comput Methodol 1(2):103–116
40. Klenin KV, Tristram F, Strunk T, Wenzel W (2011) Derivatives of molecular surface area and volume: simple and exact analytical formulas. J Comput Chem 32(12):2647–2653
41. Liang J, Edelsbrunner H, Fu P, Sudhakar PV, Subramaniam S (1998) Analytical shape computation of macromolecules: I. Molecular area and volume through alpha shape. Proteins Struct Funct Genet 33(1):1–17
42. Yan C, Dobbs D, Honavar V (2003) Identification of surface residues involved in protein-protein interaction—a support vector machine approach. In: Intelligent systems design and applications. Springer, Berlin, pp 53–62
43. Yan C, Dobbs D, Honavar V (2004) A two-stage classifier for identification of protein–protein interface residues. Bioinformatics 20(Suppl 1):i371–i378
44. Binkowski TA, Joachimiak A, Liang J (2005) Protein surface analysis for function annotation in high-throughput structural genomics pipeline. Protein Sci 14(12):2972–2981
45. Yan C, Wu F, Jernigan RL, Dobbs D, Honavar V (2008) Characterization of protein–protein interfaces. Protein J 27(1):59–70
46. Li B, Turuvekere S, Agrawal M, La D, Ramani K, Kihara D (2008) Characterization of local geometry of protein surfaces with the visibility criterion. Proteins Struct Funct Bioinf 71(2):670–683
47. Venkatraman V, Sael L, Kihara D (2009) Potential for protein surface shape analysis using spherical harmonics and 3d Zernike descriptors. Cell Biochem Biophys 54(1–3):23–32
48. Liang J, Woodward C, Edelsbrunner H (1998) Anatomy of protein pockets and cavities: measurement of binding site geometry and implications for ligand design. Protein Sci 7(9):1884–1897
49. Naderi-Manesh H, Sadeghi M, Arab S, Movahedi AAM (2001) Prediction of protein surface accessibility with information theory. Proteins Struct Funct Bioinf 42(4):452–459
50. Pollastri G, Baldi P, Fariselli P, Casadio R (2002) Prediction of coordination number and relative solvent accessibility in proteins. Proteins Struct Funct Bioinf 47(2):142–153

51. Ahmad S, Gromiha MM, Sarai A (2003) Real value prediction of solvent accessibility from amino acid sequence. Proteins Struct Funct Bioinf 50:629–635
52. Yuan Z, Huang B (2004) Prediction of protein accessible surface areas by support vector regression. Proteins Struct Funct Bioinf 57:558–564
53. Adamczak R, Porollo A, Meller J (2004) Accurate prediction of solvent accessibility using neural networks-based regression. Proteins Struct Funct Bioinf 56:753–767
54. Adamczak R, Porollo A, Meller J (2005) Combining prediction of secondary structure and solvent accessibility in proteins. Proteins Struct Funct Bioinf 59(3):467–475
55. Garg A, Kaur H, Raghava G (2005) Real value prediction of solvent accessibility in proteins using multiple sequence alignment and secondary structure. Proteins 61:318–324
56. Xu Z, Zhang C, Liu S, Zhou Y (2006) QBES: predicting real values of solvent accessibility from sequences by efficient, constrained energy optimization. Proteins Struct Funct Bioinf 63:961–966
57. Wang J, Lee H, Ahmad S (2005) Prediction and evolutionary information analysis of protein solvent accessibility using multiple linear regression. Proteins Struct Funct Bioinf 61:481–491
58. Pollastri G, Martin AJ, Mooney C, Vullo A (2007) Accurate prediction of protein secondary structure and solvent accessibility by consensus combiners of sequence and structure information. BMC Bioinf 8(1):201
59. Rost B (2009) Prediction of protein structure in 1d—secondary structure, membrane regions, and solvent accessibility. In: Structural bioinformatics, pp 679–714
60. Wang G, Dunbrack RL (2003) PISCES: a protein sequence culling server. Bioinformatics 19(12):1589–1591
61. Wang G, Dunbrack RL (2005) PISCES: recent improvements to a PDB sequence culling server. Nucleic Acids Res 33(Suppl 2):W94–W98
62. Kabsch W, Sander C (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers 22:2577–2637
63. Xue B, Dor O, Faraggi E, Zhou Y (2008) Real value prediction of backbone torsion angles. Proteins Struct Funct Bioinf 72:427–433
64. Henikoff S, Henikoff JG (1992) Amino acid substitution matrices from protein blocks. Proc Natl Acad Sci 89(22):10915–10919
65. Eddy SR et al (2004) Where did the blosum62 alignment score matrix come from? Nat Biotechnol 22(8):1035–1036
66. Styczynski MP, Jensen KL, Rigoutsos I, Stephanopoulos G (2008) Blosum62 miscalculations improve search performance. Nat Biotechnol 26(3):274–275
67. Faraggi E, Kloczkowski A (2014) GENN: a GEneral Neural Network for learning tabulated data with examples from protein structure prediction. In: Artificial neural networks: methods and applications. Methods in molecular biology
68. Moult J, Fidelis K, Kryshtafovych A, Tramontano A (2011) Critical assessment of methods of protein structure prediction (CASP) round IX. Proteins Struct Funct Bioinf 79(S10):1–5
69. Faraggi E, Yang Y, Zhang S, Zhou Y (2009) Predicting continuous local structure and the effect of its substitution for secondary structure in fragment-free protein structure prediction. Structure 17:1515–1527

Chapter 11

How to Predict Disorder in a Protein of Interest

Vladimir N. Uversky

Abstract

Currently available computational tools, which are many, provide a researcher with the multitude of options for prediction of intrinsic disorder in a protein of interest and for finding at least some of its disorder-based functions. This chapter provides a highly subjective guideline on how not to be lost in the "dark forest" of available tools for the analysis of intrinsic disorder. By no means it gives a unique pathway through this forest, but simply presents some of the tools the author uses in his everyday research.

Key words Intrinsically disordered protein, Protein function, Prediction, Posttranslational modification, Protein–protein interaction

1 Introduction

The existence of intrinsic disorder in proteins is not an alien idea anymore, although it contradicts the classical protein sequence–structure–function paradigm, where the "lock-and-key" model is used to explain how a protein can achieve its biological function via folding into a unique, highly structured state determined by its amino acid sequence [1]. Intrinsically disordered proteins (IDPs) or hybrid proteins containing ordered domains and intrinsically disordered regions (IDRs) are highly abundant in nature. In fact, all proteomes of organisms in all kingdoms of life and all viral proteomes analyzed so far have noticeable amounts of IDPs and IDPRs [2–8]. Furthermore, noticeably less than 30 % of the crystal structures in the Protein Data Bank (PDB) are completely devoid of disorder [9].

Unlike ordered proteins, whose 3D structure is relatively stable and whose Ramachandran angles vary only slightly around their equilibrium positions with occasional cooperative conformational switches, IDPs/IDRs, being biologically active, fail to form specific 3D structures and exist as highly dynamic structural ensembles, either at the secondary or at the tertiary level [5, 6, 10–15]. It is recognized now that IDPs/IDRs may contain

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_11,

collapsed disorder (where the intrinsic disorder is present in a molten globular form) and extended disorder (where intrinsic disorder is present in a form of random coil or pre-molten globule) under physiological conditions in vitro [5, 14, 16]. It was also pointed out that beside completely ordered and disordered regions, protein might contain regions of semi-disorder, i.e., fragments that have ~50% predicted probability to be disordered or ordered [17]. Curiously, such semi-disordered regions were shown to play key roles in protein aggregation and also to participate in protein–protein interactions where they undergo induced folding [17]. Based on the available structural data it was proposed that the heterogeneous spatiotemporal structure of IDPs/IDPRs can be described as a set of foldons, inducible foldons, semi-foldons, non-foldons, and unfoldons [18, 19]. The discovery of IDPs and IDRs, in which bioinformatics played a key role, has significantly broadened the understanding of protein functionality and revealed a new and unexpected role of dynamics, plasticity and flexibility in protein function.

In the laboratory, IDPs/IDRs can be identified by the variety of physicochemical methods elaborated to characterize protein structure and self-organization [14, 20–24]. These methods include missing electron density in X-ray crystallography maps [25]; NMR spectroscopy [14, 21, 26–28]; circular dichroism spectroscopy in the near-UV [29] and far-UV regions [12, 30–32]; optical rotatory dispersion spectroscopy (ORD) [12, 30]; Fourier transform infrared spectroscopy (FTIR) [12]; Raman spectroscopy and Raman optical activity [33]; fluorescent spectroscopy [22, 34]; gel-filtration, viscometry, small angle X-ray scattering (SAXS), small angle neutron scattering (SANS), sedimentation, and dynamic and static light scattering [22, 34, 35]; limited proteolysis [36–40]; aberrant mobility in SDS-gel electrophoresis [13, 41]; abnormal conformational stability [34, 42–45]; H/D exchange [22]; immunochemical methods [46, 47]; interaction with molecular chaperones [34]; electron microscopy or atomic force microscopy.

Importantly, the IDP field originated mostly due to the bioinformatics that was used to transform a set of anecdotal examples of structure-less biologically active proteins, which were originally considered to be intriguing exceptions within the protein realm, into a very promising branch of protein science and that clearly showed natural abundance of IDPs/IDRs. In fact, already at the early stage of the field, simple statistical comparisons of amino acid compositions and sequence complexity indicated that disordered and ordered regions are different to a significant degree. Based on the analysis of 150 ID segments and comparison of these segments with ordered proteins it has been suggested that the amino acids can be grouped into order promoting (C, F, I, L, F, N, V, W, Y), disorder promoting (A, E, G, K, P, Q, R, S), and neutral (D, H, M,

T) [5]. Several subsequent studies followed up this analysis using increasingly larger datasets [48–51]. In addition to the first-order statistics, recent studies also addressed higher-order patterns in amino acid sequence space and analyzed the space of various physicochemical properties [52], confirming the existence of several biases in IDP sequences. The mentioned sequence biases were exploited to develop a multitude of IDP/IDR predictors. Various computational tools for evaluating propensity of a given protein for intrinsic disorder were described in several reviews [53–59].

Another important side of computational analysis of intrinsic disorder is related to its applicability for finding of potential functional regions. Since short regions of predicted order embedded into the longer regions of predicted disorder were shown to correspond to binding sites that fold upon complex formation [60, 61], several specialized tools to find short regions that undergo disorder-to-order transitions on binding (known as Molecular Recognition Features, MoRFs) were developed [61–64]. Alternative and complementary models of MoRF-like interactions are the Short Linear Motif (SLiM) or Eukaryotic Linear Motif (ELM) based on sequence motifs that are recognized by peptide recognition domains [65]. A different approach is taken by the ANCHOR, which identifies segments of disordered regions that are likely to fold in conjunction with a globular binding partner [66, 67]. Also, one of the chapters in this book describes a novel computational method DisoRDPbind for high-throughput prediction of multiple functions of disordered regions that can be used to predict the RNA-, DNA-, and protein-binding residues located in IDRs in the input protein sequences [68].

Finally, it has been reported that sites of the enzyme-catalyzed posttranslational modifications, such as phosphorylation [69], acetylation, methylation, and ubiquitination [70], are commonly located within the IDRs. Based on these observations, corresponding computational tools were developed. For example, DisPhos (Disorder-enhanced Phosphorylation predictor) can efficiently find IDR-located phosphorylation sites with the accuracy of 76% for serine, 81% for threonine and 83% for tyrosine (83). Recently, a novel tool has been developed, which is a unified sequence-based predictor of 23 types of PTM sites [70].

This chapter represents an update of an article published in 2007 [71], with the major focus on the efficient analysis of the disorder status and disorder-based functionality in the individual protein. It is recommended to use described techniques for the analysis of any protein of interest, since it typically provides important information that can to better understand and interpret experimental data, to classify proteins and to understand their functionalities. Over the past few years, the disorder predictions aided in structural characterization of the retinal tetraspanin [72], nicotinic acetylcholine receptor [73], DBE [74], proapoptotic BH domain-containing

family of proteins [75], transcriptional corepressor CtBP [76], Notch signaling pathway proteins [77, 78], proteins associated with cancer [79] and cardiovascular disease [80], signaling proteins [79], transcription factors [81], PEST proteins [82], histones [83, 84], ribosomal proteins [85], various viral proteins [86–92], serine/arginine-rich splicing factors [93], partners of 14-3-3 proteins [94], nucleoporins [95], and many other proteins.

Currently, numerous computational tools are available for prediction of both the intrinsic disorder propensity of a protein of interest and for finding at least some of its disorder-based functions. Techniques represented in this chapter are the tools routinely utilized in my laboratory for the analysis of intrinsic disorder in various proteins. By no means, selection of these tools is related to their superiority over many other computational techniques. Being used on a daily basis, they are techniques of "convenience" and "habit". Therefore, presented below is a highly subjective guideline on how to get some useful structural and potential functional information about a query protein based on its amino acid sequence alone.

2 Materials

1. The UniProt database is described in ref. [96] and is available from http://web.expasy.org.
2. The database of experimentally characterized disordered proteins, DisProt, is available from http://www.disprot.org. This database is described in refs. [97, 98].
3. Composition profiler is available from http://www.cprofiler.org/ and is described in ref. [99].
4. PONDR® VLXT predictor is described in ref. [48] and is available from http://www.pondr.com/ and from http://www.disprot.org/metapredictor.php.
5. PONDR® VL3 is described in ref. [100] and is available from http://www.pondr.com/ and from http://www.disprot.org/metapredictor.php.
6. PONDR® VSL2 is described in refs. [101, 102] and is available from http://www.pondr.com/ and from http://www.disprot.org/metapredictor.php.
7. The metapredictor PONDR-FIT is described in ref. [103] and is available and from http://www.disprot.org/metapredictor.php.
8. CH-plot predictor is available from http://www.pondr.com/. The basic algorithm of this binary classifier is described in ref. [12].
9. Another binary disorder classifier, CDF analysis, is available from http://www.pondr.com/. This predictor is described in refs. [2, 104].

10. D^2P^2 is the database of disordered protein predictions. It is described in ref. [105] and is available from http://d2p2.pro/.
11. A database of intrinsically disordered and mobile proteins, MobiDB is described in refs. [106, 107] and is available from http://mobidb.bio.unipd.it/ and via the related link at UniProt (http://web.expasy.org/docs/swiss-prot_guideline.html).
12. ANCHOR and associated with it disorder predictor IUPred are available from http://anchor.enzim.hu/. ANCHOR is described in refs. [66, 67].
13. MoRFpred is described in ref. [64] and is available from http://biomine-ws.ece.ualberta.ca/MoRFpred/index.html.
14. DisPhos predictor also known as DEPP (*D*isorder *E*nhanced *P*hosphorylation *P*redictor) is described in ref. [69] and is available from http://www.pondr.com/.
15. ModPred is available from www.modpred.org and is described in ref. [70].
16. STRING is available from http://string-db.org and via the related link at UniProt (http://web.expasy.org/docs/swiss-prot_guideline.html). This database represents a Search Tool for the Retrieval of Interacting Genes, which provides information on both experimentally validated and predicted interactions of a query protein. It is described in ref. [108].

3 Methods

The methods outlined below describe the analysis of amino acid sequences using the intrinsic disorder knowledge to gain structural and functional information related to a protein of interest. Although numerous predictors of intrinsic disorder are currently available, this chapter focuses on utilization of PONDR® tools and two databases, D^2P^2 and MobiDB, as they cover wide range of potential applications of intrinsic disorder concept for structural and functional analysis of proteins. Obviously, this analysis could have been carried out with many other disorder predictors.

3.1 Analysis of Protein Amino Acid Composition

One of the specific features of an IDP or an IDR is the characteristic amino-acid compositional bias with low content of order-promoting residues (C, W, V, F, Y, L, I, and M) compensated by high content of disorder-promoting residues (Q, S, P, E, K, G, and A [5, 48, 109]. This means the ordered or intrinsically disorder nature of a given protein can be guessed based on a simple analysis of its amino acid composition biases using the fractional difference in amino acid approach [5]. Here, the fractional difference is calculated as $(f(r)-f_{\text{order}}(r))/f_{\text{order}}(r)$, where $r \in \{A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y\}$, $f(r)$ is the

frequency of residue r in a given protein set and $f_{order}(r)$ is the frequency of residue r in the reference set of globular proteins, and plotted for each amino acid. The plotting is done using the Composition profiler [99]. In the resulting graph, negative bars correspond to amino acids that are underrepresented in a given protein in comparison with the set of ordered proteins, whereas positive bars reflect the relative increase in the particular amino acid content in a query protein. Step-by-step protocol for the use of the Composition profiler is given below.

3.1.1 Retrieving Sequence Information from the UniProt Database

Start the UniProt database by typing http://web.expasy.org/docs/swiss-prot_guideline.html in the Internet browser and hit the "List of UniProtKB/Swiss-Prot (reviewed) entries" link located at the top of the front page. Use the following steps to download sequence information in FASTA format.

1. In the **Search** window (located at the top of the page), type the protein name after **reviewed:yes** and click Search.
2. On a Search in UniProt Knowledgebase page choose a protein of interest from the list of hits and click corresponding link (which will be located in the column entitled **Entry**).
3. On the left-hand side of the corresponding UniProtKB entry page, look for a blue bar containing link to Sequence and hit this link. In the section entitled **Sequence**, click FASTA link located within the light blue box.
4. Copy content of the page which includes a descriptive header related to your protein and a protein sequence. Keep this information as it will be used in the subsequent analysis. This can be done in Notepad or Microsoft Word. A separate document for each protein is recommended in which all the results of different analyses will be stored.

3.1.2 Applying Compositional Profiler Tool to Obtain Fractional Amino Acid Composition of a Query Protein

1. Start the Composition profiler by typing http://www.cprofiler.org/ in the Internet browser and hit Run Profiler link located at the top right corner of the front page.
2. Paste sequence of your protein in the **Query Sample** window located on the left side of the window. In the **Background Sample** window (also located on the left side of the window) chose Dataset and select PDB select 25 from the drop-down list. Find **Output Options** on the right side of the window, chose Ordering and select Flexibility (Vihinen) from the drop-down list. Click Draw Profile link located in a gray bar at the bottom of the **Output Options** section. The resulting page will contain a plot showing the fractional amino acid composition of a query protein and a table listing statistical parameters of this analysis.
3. If numerical values instead of plot are needed, **step 2** should be modified as follows. Paste sequence of your protein in the **Query**

Sample window located on the left side of the window. In the **Background Sample** window (also located on the left side of the window) chose Dataset and select PDB select 25 from the drop-down list. Find Output Options on the right side of the window and chose Output format, where TXT (raw data) should be selected from the drop-down list. Then, chose Ordering and select Flexibility (Vihinen) from the drop-down list. Click Draw Profile link located in a gray bar at the bottom of the Output Options section. The resulting page will now contain raw data in tabulated form, where first column represents single character residue name, second column shows calculated values of the fractional difference, and the third column gives errors.

4. To obtain compositional profile of typical IDPs (which is a recommended step in order to get reference plot), **step 2** should be modified as follows. In the Query Sample window located on the left side of the window chose Dataset and select DisProt 3.4 from the drop-down list. In the Background Sample window (also located on the left side of the window) chose Dataset and select PDB select 25 from the drop-down list. Find Output Options on the right side of the window, chose Ordering and select Flexibility (Vihinen) from the drop-down list. Click Draw Profile link located in a gray bar at the bottom of the Output Options section. The resulting page will contain a plot showing the fractional amino acid composition of typical disordered proteins and a table listing statistical parameters of this analysis.
5. If numerical values instead of plot are needed, **step 3** should be modified as follows. In the Query Sample window located on the left side of the window chose Dataset and select DisProt 3.4 from the drop-down list. In the Background Sample window (also located on the left side of the window) chose Dataset and select PDB select 25 from the drop-down list. Find Output Options on the right side of the window and chose Output format where TXT (raw data) should be selected from the drop-down list. Then, chose Ordering and select Flexibility (Vihinen) from the drop-down list. Click Draw Profile link located in a gray bar at the bottom of the Output Options section. The resulting page will now contain raw data in tabulated form, where first column represents single character residue name, second column shows calculated values of the fractional difference, and the third column gives errors.
6. To plot the compositional profile of a query protein in comparison with the corresponding profile of typical IDPs, use numerical data from **steps 3** and **5**. Although the order of residues you retrieved from Compositional profiler follows the Vihinen's flexibility scale, for better visual representation, residues should be ranged as follows C, W, I, Y, F, L, H, V, N, M,

R, T, D, G, A, K, Q, S, E, and P; i.e., from the most order-promoting at the left to the most disorder-promoting at the right (*see* Fig. 1).

Fig. 1 illustrates this approach by representing the relative amino acid composition of human RNA-binding protein FUS (FUS, UniProt ID: P35637, open bars) in comparison with the compositional profile of a set of typical IDPs available in the DisProt database [97] (black bars). This analysis clearly shows that FUS is enriched in major disorder-promoting residues and depleted in major order-promoting residues, thereby possessing amino acid composition close to that of typical IDPs.

3.2 Analyzing Disorder Propensity by PONDR® Tools

3.2.1 Entering Information to the PONDR® Site and Retrieving Results of Disorder Prediction

1. Go to the official PONDR® site by typing http://www.pondr.com/ in the Internet browser. Locate and click PREDICT DISORDER link at the top left corner of the major PONDR page. This will bring you to the PONDR® working page.
2. While on the PONDR® working page, select boxes corresponding to the desired Predictors (VLXT, VL3-BA, VSL2, CDF, and Charge-Hydropathy). When Charge-Hydropathy box is marked, two new boxes (**From:** and **To:**) will appear. Leave both empty. Put Protein name in the space provided (optional). Enter NCBI Accession Code or Protein Sequence (FASTA format or sequence only) in the corresponding boxes. Scroll down the page and check the box Raw Output at the Output Options section. Clicking Submit Query will bring you to the PONDR® results page.

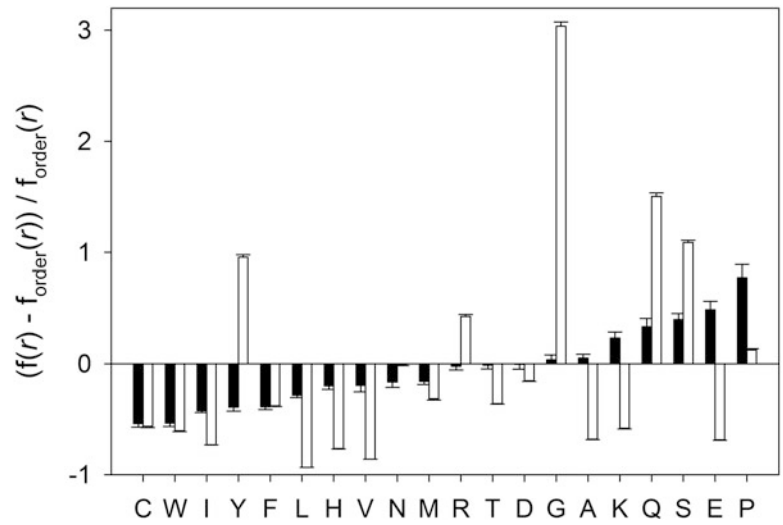


Fig. 1 Compositional profiling of an illustrative IDP, RNA-binding FUS protein (UniProt ID: P35637, *open bars*) in comparison with the compositional profile of typical ordered proteins. The compositional profile of typical intrinsically disordered proteins from the DisProt database is shown for comparison (*black bars*)

3. It is recommended to keep the content of the entire PONDR® results page. Figures can be used as illustrations. Statistics section provides useful information on the number of residues predicted to be disordered, overall percent of disordered residues, number of disordered regions, the length of the longest disordered region and the average prediction score. You will find here a list of regions predicted to be disordered. Raw output values can be used to plot the results for several proteins on one graph.

3.2.2 Retrieving Results of Disorder Prediction from the DisProt

1. Go to the official DisProt site by typing http://www.disprot.org/ in the Internet browser. Locate and click Disorder Predictors link at left side of the major DisProt page. This will bring you to the page containing a table with several disorder predictors. Locate and click Internal pointer to the PONDR-FIT meta-predictor and other PONDR methods link (which is the second line in the table). This will bring you to the Predict Disorder working page (you can also access it by typing http://www.disprot.org/metapredictor.php in the Internet browser.
2. While on the Predict Disorder working page, select boxes corresponding to the desired Predictors (VSL2B, VL3, VLXT, PONDR-FIT). Enter Protein Sequence in the corresponding box. At this page, several sequence formats are allowed, including FASTA, EMBL, and plain sequence format. These formats are described at the bottom of the page. Clicking Submit will bring you to the results page.
3. It is recommended to keep the content of the entire results page. Figure can be used as illustration. Raw output values can be found in links VSL2 DATA, VL3 DATA, VLXT DATA, and PONDR-FIT DATA located in the **Files used to produces this plot** section. Note that raw PONDR-FIT data contain both disorder scores (third column) and errors in disorder evaluation (fourth column), whereas the raw PONDR® VSL2 in addition to the disorder score include annotation of a given residue as ordered (O) or intrinsically disordered (D) shown in the fourth column.

3.2.3 Understanding the Results of the PONDR® Analyses

1. *PONDR® scores.* The PONDR® results page starts with the plot providing the distribution of PONDR® scores over the amino acid sequence. You will have three color lines there, red, blue, and pink, corresponding to the results for the PONDR® VLXT, PONDR® VL3-BA, and PONDR® VSL2 predictions, respectively. In the plot generated by DisProt, the results of the PONDR® VLXT, PONDR® VL3, PONDR® VSL2, and PONDR-FIT predictions are shown as gray, red, blue and green lines, respectively. Scores above the threshold of 0.5 correspond to the regions predicted to be disordered. Long disordered regions (with >30 consecutive residues predicted to be disordered) are indicated as thick black lines. Figure 2 represents illustrative PONDR® plot for the intrinsically disordered

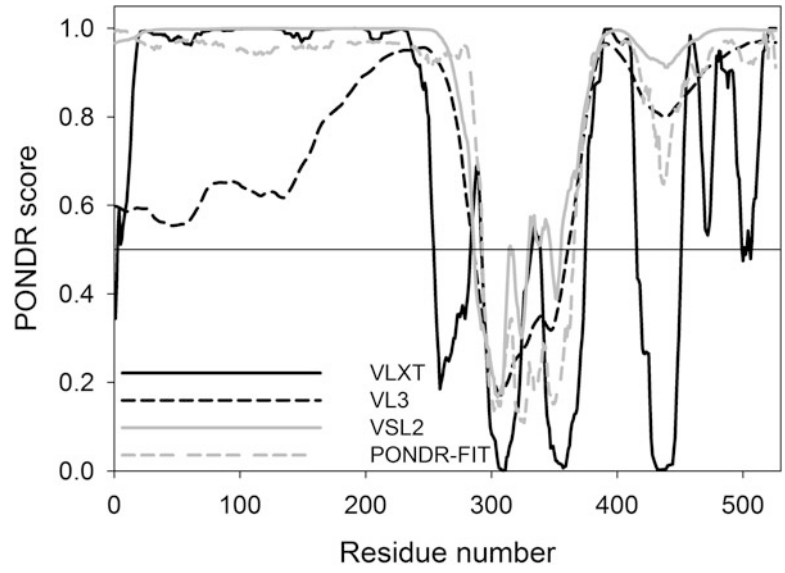


Fig. 2 Illustrative outputs of the per-residue PONDR® algorithms for FUS (UniProt ID: P35637). Results of the protein analysis by PONDR® VLXT (*black solid curve*), VL-3B (*black dashed curve*), VSL2 (*gray curve*) and PONDR-FIT (*gray dashed line*) are shown. A disorder threshold is indicated as a thin line (at score = 0.5) to show a boundary between disorder (>0.5) and order (<0.5)

protein FUS (UniProt ID: P35637) and clearly shows that the vast majorities of all three curves are located above the threshold, reflecting the fact that the FUS protein is highly disordered. Raw data of these analyses are at the end of the page in the **PREDICTOR VALUES** section.

2. *CDF analysis*. Second plot at the PONDR® data page represents the results of CDF analysis. Remember that CDF analysis summarizes the per-residue disorder predictions by plotting PONDR scores against their cumulative frequency, which allows ordered and disordered proteins to be distinguished based on the distribution of the corresponding prediction scores [2, 104]. In this case, the binary ordered-disordered classification of a whole protein is based on the positioning of the corresponding CDF curve (green curve at the screen) relative to the boundary line (thick black line with seven boundary points). Here, if curve is located below the majority of the boundary points, then entire protein is predicted to be mostly disordered, whereas if the CDF curve is above the majority of the boundary points, then the analyzed protein is mostly ordered. Raw data to reproduce this plot (results for your protein and boundary) are in the **CUMULATIVE DISTRIBUTION FUNCTION (CDF) OUTPUT** section.

3. *CH-plot analysis.* The last figure at the PONDR® results page shows the CH-plot [12]. This plot utilizes important observation that compact and highly disordered proteins plotted in the CH-space can be separated from each other to a significant degree by a linear boundary, with proteins located above this boundary being IDPs (red circles) while proteins located below the boundary line being compact (blue squares). The query protein is marked as a large green diamond. If this diamond is found above the boundary, then the protein is disordered, and if it is below the boundary, then the protein is compact. Raw data to build this plot (results for your protein, boundary as well as coordinates of sets of natively unfolded and ordered proteins) are in the **CHARGE-HYDROPATHY OUTPUT** section.
4. *Interpretation of PONDR® data* is rather straightforward. As pointed above, high PONDR® scores (above 0.5) for all three predictors (VLXT, VL3-BA, VSL2, PONDR-FIT) are characteristic of regions with high propensity to be disordered. Some peculiarities of the VLXT curve might correlate with protein functionality (see below). VL3-BA usually provides very smooth output, as it was trained on long regions of disorder and its raw predictions are averaged over an output window of length 31 to obtain the final prediction for a given position [100]. VL3-BA is useful for the accurate prediction of long disordered regions. VSL2 is one of the most accurate stand-alone predictor of intrinsic disorder in the PONDR® series. Its training set is 1,335 non-redundant protein sequences, containing 230 long disordered regions with 25,958 residues, 983 short disordered regions with 9,632 residues, and 354,169 ordered residues [101, 102]. Finally, PONDR-FIT is a metapredictor combining six individual predictors (PONDR® VLXT [48], PONDR® VSL2 [100], PONDR® VL3 [102], FoldIndex [110], IUPred [111], TopIDP [112]), which is moderately more accurate than each of the component predictors [103].
5. *Interpretation of CDF and CH-plot analyses* is straightforward too. It has been pointed out that sometimes these two analyses provide seemingly contradictory data, with CDF analysis predicting a much higher frequency of disorder in sequence databases than CH-plot discrimination [104, 113, 114]. The reasons for this discrepancy are outlined below (*see* **Note 1**). Differences in predictions by these two classifiers were suggested to be physically interpretable in terms of the degree of protein disorder. Proteins predicted to be disordered by both CH-plot and CDF (i.e., polypeptide chains with high net charge and low hydrophobicity) are likely to be in the extended disorder class. Proteins predicted to be disordered by CDF,

but ordered by CH-plot should have properties consistent with a dynamic, collapsed chain and are likely to be in the collapsed disorder class (i.e., molten globules), or be hybrid proteins with comparable content of ordered domains and IDRs. Rarely, proteins are predicted to be disordered by CH-plot, but ordered by the CDF analysis. This may represent structured proteins with an unusually high net charge; such proteins are likely to exhibit salt-sensitive structures. Finally, proteins predicted to be ordered by both algorithms are of course likely to be in the well-structured class [104].

3.3 Web-Based Means for the Visualization of Disorder Distribution in a Protein

3.3.1 Using D^2P^2

1. Go to the official D^2P^2 site by typing http://d2p2.pro/ in the Internet browser. Locate and click Search link located within the **Protein Search** section. This will bring you to the **Search** page. Enter sequence of a query protein in FASTA format to the **Sequences** box and click Find Proteins link, which will bring you to the **Results for your sequence search** page.
2. The result page contains a very useful picture that includes the results of the multi-tool analysis of the disorder status of the query protein, as well as some functional annotation. Note, while you are at this page, you can find useful information on location of disordered and functional regions and PTM sites by placing cursor over the corresponding part of the plot. It is recommended to save the resulting figure since it can be used as nice illustration (*see* Fig. 3).

3.3.2 Using MobiDB

1. Go to the official MobiDB site by typing http://mobidb.bio.unipd.it/ in the Internet browser. Type in the UniProt ID of your protein in the **UniProt Query** line and click Search link. This will bring you to the **Search** page. Locate your protein in the **List of Returned Results** and click the corresponding link containing UniProt ID of your protein. This will bring you to the results section, which contains a lot of information about the query protein.
2. Alternatively, you can get an access to MobiDB directly from the UniProt page corresponding to the protein of interest. At the left-hand side of the corresponding UniProtKB entry page, look for a blue bar containing link to Structure and click this link. In the section entitled **Structure**, locate MobiDB pointer and click Search link next to it. This will bring you to the same results section as above.
3. The result page contains a lot of very useful information about the query protein that includes structural information with corresponding PDB IDs (if available), the results of the multi-tool analysis of the disorder status of the query protein, as well as some functional annotation. Therefore, it is recommended to keep the content of the entire results page.

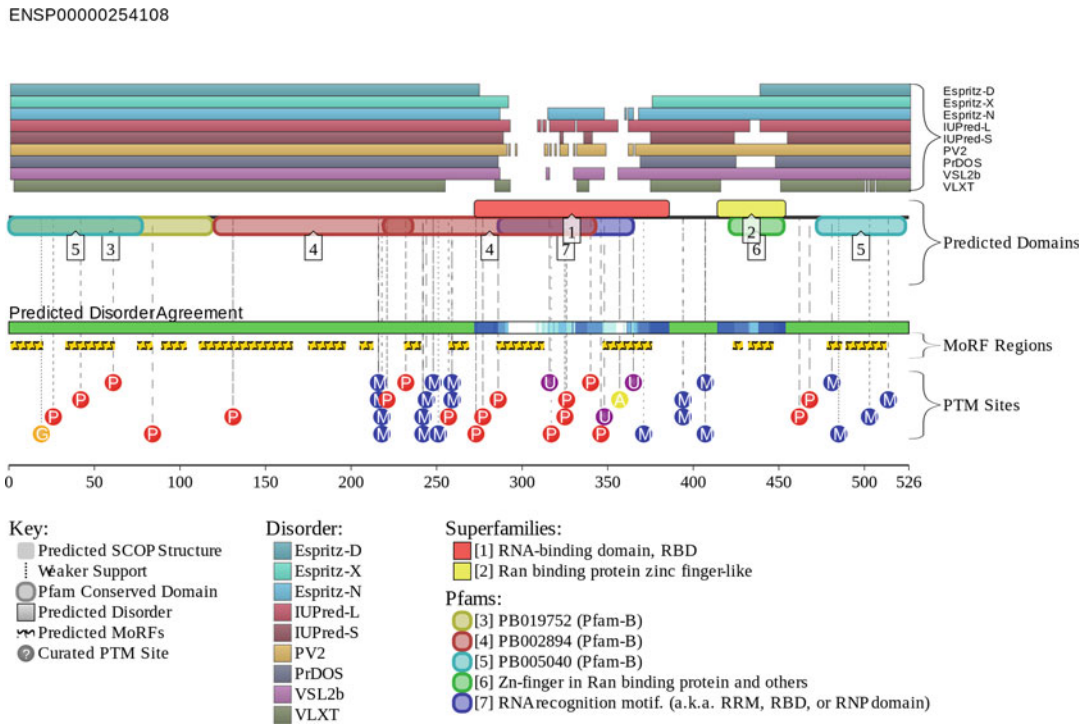


Fig. 3 Evaluation of the functional intrinsic disorder propensity of human FUS protein (UniProt ID: P35637) by D^2P^2 database (http://d2p2.pro/). In this plot, top nine-colored bars represent location of disordered regions predicted by different disorder predictors (Espritz-D, Espritz-N, Espritz-X, IUPred-L, IUPred-S, PV2, PrDOS, PONDR® VSL2b, and PONDR® VLXT, see keys for the corresponding color codes). *Green-and-white bar* in the middle of the plot shows the predicted disorder agreement between these nine predictors, with *green parts* corresponding to disordered regions by consensus. *Yellow bar* shows the location of the predicted disorder-based binding site (MoRF region), whereas *red circles* at the bottom of the plot show location of various PTM sites

3.3.3 Understanding Outputs of D^2P^2 and MobiDB

1. *Interpretation of D2P2 data* is rather straightforward. In a visually attractive form, this database provides an access to the pre-computed disorder predictions [105] using outputs of PONDR® VLXT [48], two flavors of IUPred [111], PONDR® VSL2B [101, 115], PrDOS [116], three flavors of ESpritz [117], and PV2 [105]. The visual console of D^2P^2 is further enhanced by providing information on the curated cites of various posttranslational modifications and on the location of predicted disorder-based potential binding sites. In the corresponding plots, top nine colored bars represent location of disordered regions predicted by different disorder predictors (Espritz-D, Espritz-N, Espritz-X, IUPred-L, IUPred-S, PV2, PrDOS, PONDR® VSL2b, and PONDR® VLXT, see keys for the corresponding color codes). Next two lines with colored and numbered bars show positions of predicted domains. Green-and-white bar in the middle of the plot shows the predicted disorder agreement between these nine predictors, with

green parts corresponding to disordered regions by consensus. Yellow bar shows the location of the predicted disorder-based binding site (MoRF region), whereas red, yellow, orange, blue, and violet circles at the bottom of the plots show locations of phosphorylation, acetylation, glycosylation, methylation, and ubuquitylation sites, respectively.

2. *Interpretation of the MobiDB data* is rather intuitive too. The page starts with the general **Sequence Annotations**, where locations of long disordered regions and structure/disorder information from all available sources (e.g., structural data from the PDB in form of NMR and X-ray structures (if available), and results of multi-tool disorder prediction) are shown. If several NMR (or X-ray) structures are available for a query protein, then data shown in this section will correspond to the consensus of all NMR (or X-ray) data. Numeric disorder scores are shown next to the corresponding lines. Next line shows location of Pfam domains. This is followed by the **Detailed Disorder Annotations** section, which contains multiple subsections showing results extracted from the individual PDB entries in a form of distribution of ordered and disordered regions. Consensus for all NMR or all X-ray structures is also shown. Each line is ended with the corresponding numeric score. MobiDB also generates consensus disorder scores based on the outputs of ten disorder predictors, such as ESpritz in its two flavors [117], IUPred in its two flavors [111], DisEMBL in two of its flavors [118], GlobPlot [119], PONDR® VSL2 [101, 115], and JRONN [120] in addition to showing results of the individual predictors. This is followed by the **Protein–Protein Interactions** section that contains **Known Structural Interactors (from PDB)** and **Known Experimental and Database Interactors (from STRING)** subsections. Here known and predicted binding partners are listed together with their corresponding disorder scores. The page is concluded with the **Detailed Sequence Annotations** section, where the **Consensus Table** and the **Prediction Table** shown numerically locations of disordered regions are located.

3.4 Intrinsic Disorder-Based Functional Analyses

Many IDPs and IDRs are known to undergo at least partial disorder-to-order transitions upon binding, which is crucial for their functions in recognition, regulation, and signaling [5, 11, 12, 18, 61, 62, 121–124]. Among these potential functional sites are short order-prone motifs within long disordered regions that are able to undergo disorder-to-order transition during the binding to a specific partner. These motifs are known as Molecular Recognition Feature (MoRF), and they can be identified computationally [61, 63].

3.4.1 Predicting Disorder-Based Binding Sites by ANCHOR

1. Go to the official ANCHOR site by typing http://anchor.enzim.hu/ in the Internet browser. Type in the UniProt ID of your protein in the **Enter SWISS-PROT/TrEMBL Identifier or Accession Number** line or deposit sequence of your protein in **Or Paste the Amino Acid Sequence** window and click SUBMIT link at the bottom of page. This will bring you to the **Prediction of Protein Binding Regions in Disordered Proteins** page.
2. The results page contains a self-explanatory plot with the results of ANCHOR analysis (*see* Fig. 4), a Table with localization of predicted disorder-based binding regions, and numerical raw data.

3.4.2 Predicting Disorder-Based Binding Sites by MoRFpred

1. Go to the official MoRFpred site by typing http://biomine-ws.ece.ualberta.ca/MoRFpred/index.html in the Internet browser. Deposit sequence of your protein (in FASTA format) in **(1) Enter protein sequence(s)** window. Enter your e-mail address in **(2) Provide your e-mail address** line and click Run MoRFpred! link in the **(3) Predict** line. This will bring you to the **MoRFpred processing page**, where the information will be provided on you position in queue and also a link will be given to the **Results page**. The results will be displayed there once your request will be completed and they will be stored on our server for at least 1 month. Please save this address for your reference. Information on the results will also be sent to your e-mail address.

3.4.3 Predicting Potential Phosphorylation Sites Using DisPhos

It has been shown that intrinsic disorder prediction might help increase the prediction accuracy of several protein post-translational modification (PTM) sites, including protein phosphorylation [69],

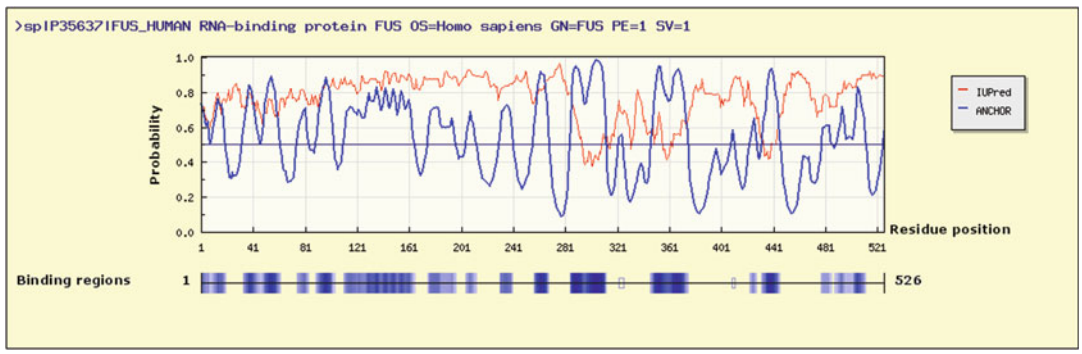


Fig. 4 Prediction of potential disorder-based interaction sites human FUS protein (UniProt ID: P35637) by ANCHOR. The plot provides the distribution of disorder propensity (evaluated by IUPred, *red line*) and distribution of ANCHOR scores (*blue line*). In IUPred plot, residues/regions with scores >0.5 are predicted to be disordered. In ANCHOR plot, residues/regions with scores >0.5 are predicted to correspond to the potential disorder-based binding sites. Bottom of plot represents binding regions as bars with different *shades of blue*, with darker color corresponding to higher ANCHOR scores. This bottom graph shows regions possessing ANCHOR scores >0.5

methylation [125], and many other PTMs. A brief protocol below describes utilization of DEPP (or DisPhos), which uses disorder information to improve the discrimination between phosphorylation and non-phosphorylation sites. The retrieved prediction score approximates the probability that the residue is phosphorylated. Only residues with a prediction score >0.5 (which) are considered to be phosphorylated. The step-by-step protocol of DEPP analysis is presented below.

1. Go to the PONDR® working page and click the DEPP Prediction button. This will bring you to the DEPP working page. While on this page, type Protein name in the space provided (optional) and enter NCBI Accession Code or Protein Sequence (FASTA format or sequence only) in the corresponding boxes. Scroll down the page and check the box Raw Output at the Output Options section. By clicking Submit Query button you will be forwarded to the DEPP results page.
2. The top of DEPP results page represents the plot providing the distribution of DEPP scores over the amino acid sequence. You will have three types of symbols corresponding to the Thr (green triangles), Ser (blue squares), and Tyr residues (red circles) predicted to be phosphorylated. Only residues possessing DEPP scores >0.5 will be shown.
3. Raw data related to this analysis are at the end of the page in the **PREDICTOR VALUES** section. The **DEPP NNP STATISTICS** section provides useful information on the number of phosphorylated serines, threonines, and tyrosines, together with the total number of these residues in a given protein and the relative phosphorylation efficiency. Once again, we recommend you to keep the content of the entire DEPP results page for the future use.

3.4.4 Predicting Potential Sites of 23 Different PTMs Using ModPred

Recently, a unified sequence-based predictor of 23 types of PTM sites was developed [70]. This tool represents a very useful instrument for guiding biological experiments and data interpretation [70].

1. Go to the official ModPred page by typing www.modpred.org in the Internet browser. This will bring you to the **ModPred: predictor of post-translational modification sites in proteins** page. Deposit protein sequence in **the Paste the protein sequence (one at a time)** box. Find and click the Check all link and then click the Predict link. This will bring you to the next page stating "**Predicting PTM sites … Please do not hit the 'Back' or 'Refresh' button**". When calculations will be done, you will be brought to the result page.
2. The results page has the **INPUT** section that provides sequence ID of your protein, its length, and the list of predicted PTMs. The **OUTPUT** section provides prediction results, where

sequence is color coded to show residues predicted to be modified with low confidence (red), medium confidence (yellow), and high confidence (green), as well as residues corresponding to multiple PTM sites (blue). Below this annotated sequence there is a table that lists all prediction results. This table can be downloaded as a tab-delimited file.

4 Notes

1. The difference in the disorder prediction by CDF analysis and CH-plot likely results from the fact that the CH-plot is a linear classifier that takes into account only two parameters of the particular sequence—charge and hydrophobicity [12], whereas the CDF analysis is dependent upon the output of the PONDR® VL-XT predictor, a nonlinear neural network classifier, which was trained to distinguish order and disorder based on a significantly larger feature space that explicitly includes net charge and hydropathy [2, 104]. Therefore, charge-hydropathy feature space can be considered as a subset of PONDR VL-XT feature space. By definition, CH-plot analysis is predisposed to discriminate proteins with substantial amounts of extended disorder (random coils and pre-molten globules) from proteins with globular conformations (molten globule-like and rigid well-structured proteins). On the other hand, PONDR-based CDF analysis may discriminate all types of disordered conformations, including molten globules, pre-molten globules, and coils from ordered proteins [104].

Acknowledgements

This work was supported in part by a grant from the Russian Science Foundation RSCF № 14-24-00131.

References

1. Fischer E (1894) Einfluss der configuration auf die wirkung der enzyme. Ber Dt Chem Ges 27:2985–2993
2. Dunker AK, Obradovic Z, Romero P, Garner EC, Brown CJ (2000) Intrinsic protein disorder in complete genomes. Genome Inform Ser Workshop Genome Inform 11:161–171
3. Uversky VN (2010) The mysterious unfoldome: structureless, underappreciated, yet vital part of any given proteome. J Biomed Biotechnol 2010:568068
4. Ward JJ, Sodhi JS, McGuffin LJ, Buxton BF, Jones DT (2004) Prediction and functional analysis of native disorder in proteins from the three kingdoms of life. J Mol Biol 337(3):635–645
5. Dunker AK, Lawson JD, Brown CJ, Williams RM, Romero P, Oh JS, Oldfield CJ, Campen AM, Ratliff CM, Hipps KW, Ausio J, Nissen MS, Reeves R, Kang C, Kissinger CR, Bailey RW, Griswold MD, Chiu W, Garner EC, Obradovic Z (2001) Intrinsically disordered protein. J Mol Graph Model 19(1):26–59
6. Uversky VN, Dunker AK (2010) Understanding protein non-folding. Biochim Biophys Acta 1804(6):1231–1264

7. Xue B, Dunker AK, Uversky VN (2012) Orderly order in protein intrinsic disorder distribution: disorder in 3500 proteomes from viruses and the three domains of life. J Biomol Struct Dyn 30(2):137–149
8. Peng Z, Yan J, Fan X, Mizianty MJ, Xue B, Wang K, Hu G, Uversky VN, Kurgan L (2015) Exceptionally abundant exceptions: comprehensive characterization of intrinsic disorder in all domains of life. Cell Mol Life Sci 72(1):137–151
9. Obradovic Z, Peng K, Vucetic S, Radivojac P, Brown CJ, Dunker AK (2003) Predicting intrinsic disorder from amino acid sequence. Proteins 53(S6):566–572
10. Dunker AK, Garner E, Guilliot S, Romero P, Albrecht K, Hart J, Obradovic Z, Kissinger C, Villafranca JE (1998) Protein disorder and the evolution of molecular recognition: theory, predictions and observations. Pac Symp Biocomput 473–484
11. Wright PE, Dyson HJ (1999) Intrinsically unstructured proteins: re-assessing the protein structure-function paradigm. J Mol Biol 293(2):321–331
12. Uversky VN, Gillespie JR, Fink AL (2000) Why are "natively unfolded" proteins unstructured under physiologic conditions? Proteins 41(3):415–427
13. Tompa P (2002) Intrinsically unstructured proteins. Trends Biochem Sci 27(10):527–533
14. Daughdrill GW, Pielak GJ, Uversky VN, Cortese MS, Dunker AK (2005) Natively disordered proteins. In: Buchner J, Kiefhaber T (eds) Handbook of protein folding. Wiley-VCH, Verlag GmbH & Co., KGaA, Weinheim, Germany, pp 271–353
15. Uversky VN, Dunker AK (2013) The case for intrinsically disordered proteins playing contributory roles in molecular recognition without a stable 3D structure. F1000 Biol Rep 5:1
16. Uversky VN (2003) Protein folding revisited. A polypeptide chain at the folding-misfolding-nonfolding cross-roads: which way to go? Cell Mol Life Sci 60(9):1852–1871
17. Zhang T, Faraggi E, Li Z, Zhou Y (2013) Intrinsically semi-disordered state and its role in induced folding and protein aggregation. Cell Biochem Biophys 67(3):1193–1205
18. Uversky VN (2013) Unusual biophysics of intrinsically disordered proteins. Biochim Biophys Acta 1834(5):932–951
19. Uversky VN (2013) A decade and a half of protein intrinsic disorder: biology still waits for physics. Protein Sci 22(6):693–724
20. Uversky VN (2002) Natively unfolded proteins: a point where biology waits for physics. Protein Sci 11(4):739–756
21. Bracken C, Iakoucheva LM, Romero PR, Dunker AK (2004) Combining prediction, computation and experiment for the characterization of protein disorder. Curr Opin Struct Biol 14(5):570–576
22. Receveur-Brechot V, Bourhis JM, Uversky VN, Canard B, Longhi S (2006) Assessing protein disorder and induced folding. Proteins 62(1):24–45
23. Uversky VN, Dunker AK (2012) Multiparametric analysis of intrinsically disordered proteins: looking at intrinsic disorder through compound eyes. Anal Chem 84(5):2096–2104
24. Uversky VN (2015) Biophysical methods to investigate intrinsically disordered proteins: avoiding an "Elephant and Blind Men" situation. Adv Exp Med Biol 870:215–260
25. Ringe D, Petsko GA (1986) Study of protein dynamics by X-ray diffraction. Methods Enzymol 131:389–433
26. Dyson HJ, Wright PE (2002) Insights into the structure and dynamics of unfolded proteins from nuclear magnetic resonance. Adv Protein Chem 62:311–340
27. Dyson HJ, Wright PE (2004) Unfolded proteins and protein folding studied by NMR. Chem Rev 104(8):3607–3622
28. Dyson HJ, Wright PE (2005) Elucidation of the protein folding landscape by NMR. Methods Enzymol 394:299–321
29. Fasman GD (1996) Circular dichroism and the conformational analysis of biomolecules. Plenem Press, New York
30. Adler AJ, Greenfield NJ, Fasman GD (1973) Circular dichroism and optical rotatory dispersion of proteins and polypeptides. Methods Enzymol 27:675–735
31. Provencher SW, Glockner J (1981) Estimation of globular protein secondary structure from circular dichroism. Biochemistry 20(1):33–37
32. Woody RW (1995) Circular dichroism. Methods Enzymol 246:34–71
33. Smyth E, Syme CD, Blanch EW, Hecht L, Vasak M, Barron LD (2001) Solution structure of native proteins with irregular folds from Raman optical activity. Biopolymers 58(2):138–151
34. Uversky VN (1999) A multiparametric approach to studies of self-organization of globular proteins. Biochemistry (Mosc) 64(3):250–266
35. Glatter O, Kratky O (1982) Small angle X-ray scattering. Academic, London
36. Markus G (1965) Protein substrate conformation and proteolysis. Proc Natl Acad Sci U S A 54:253–258

37. Mikhalyi E (1978) Application of proteolytic enzymes to protein structure studies. CRC Press, Boca Raton
38. Hubbard SJ, Eisenmenger F, Thornton JM (1994) Modeling studies of the change in conformation required for cleavage of limited proteolytic sites. Protein Sci 3:757–768
39. Fontana A, de Laureto PP, de Filippis V, Scaramella E, Zambonin M (1997) Probing the partly folded states of proteins by limited proteolysis. Fold Des 2:R17–R26
40. Fontana A, de Laureto PP, Spolaore B, Frare E, Picotti P, Zambonin M (2004) Probing protein structure by limited proteolysis. Acta Biochim Pol 51(2):299–321
41. Iakoucheva LM, Kimzey AL, Masselon CD, Smith RD, Dunker AK, Ackerman EJ (2001) Aberrant mobility phenomena of the DNA repair protein XPA. Protein Sci 10:1353–1362
42. Privalov PL (1979) Stability of proteins: small globular proteins. Adv Protein Chem 33:167–241
43. Ptitsyn O (1995) Molten globule and protein folding. Adv Protein Chem 47:83–229
44. Ptitsyn OB, Uversky VN (1994) The molten globule is a third thermodynamical state of protein molecules. FEBS Lett 341:15–18
45. Uversky VN, Ptitsyn OB (1996) All-or-none solvent-induced transitions between native, molten globule and unfolded states in globular proteins. Fold Des 1(2):117–122
46. Westhof E, Altschuh D, Moras D, Bloomer AC, Mondragon A, Klug A, Van Regenmortel MH (1984) Correlation between segmental mobility and the location of antigenic determinants in proteins. Nature 311(5982):123–126
47. Berzofsky JA (1985) Intrinsic and extrinsic factors in protein antigenic structure. Science 229(4717):932–940
48. Romero P, Obradovic Z, Li X, Garner EC, Brown CJ, Dunker AK (2001) Sequence complexity of disordered protein. Proteins 42:38–48
49. Wootton JC (1993) Statistic of local complexity in amino acid sequences and sequence databases. Comput Chem 17:149–163
50. Radivojac P, Obradovic Z, Smith DK, Zhu G, Vucetic S, Brown CJ, Lawson JD, Dunker AK (2004) Protein flexibility and intrinsic disorder. Protein Sci 13(1):71–80
51. Romero P, Obradovic Z, Kissinger CR, Villafranca JE, Dunker AK (1997) Identifying disordered regions in proteins from amino acid sequences. IEEE Int Conf Neural Netw 1:90–95
52. Lise S, Jones DT (2005) Sequence patterns associated with disordered regions in proteins. Proteins 58(1):144–150
53. Ferron F, Longhi S, Canard B, Karlin D (2006) A practical overview of protein disorder prediction methods. Proteins 65(1):1–14
54. Dosztanyi Z, Sandor M, Tompa P, Simon I (2007) Prediction of protein disorder at the domain level. Curr Protein Pept Sci 8(2):161–171
55. He B, Wang K, Liu Y, Xue B, Uversky VN, Dunker AK (2009) Predicting intrinsic disorder in proteins: an overview. Cell Res 19(8):929–949
56. Kurgan L, Disfani FM (2011) Structural protein descriptors in 1-dimension and their sequence-based predictions. Curr Protein Pept Sci 12(6):470–489
57. Cozzetto D, Jones DT (2013) The contribution of intrinsic disorder prediction to the elucidation of protein function. Curr Opin Struct Biol 23(3):467–472
58. Atkins JD, Boateng SY, Sorensen T, McGuffin LJ (2015) Disorder prediction methods, their applicability to different protein targets and their usefulness for guiding experimental studies. Int J Mol Sci 16(8):19040–19054
59. Varadi M, Vranken W, Guharoy M, Tompa P (2015) Computational approaches for inferring the functions of intrinsically disordered proteins. Front Mol Biosci 2:45
60. Garner E, Romero P, Dunker AK, Brown C, Obradovic Z (1999) Predicting binding regions within disordered proteins. Genome Inform 10:41–50
61. Oldfield CJ, Cheng Y, Cortese MS, Romero P, Uversky VN, Dunker AK (2005) Coupled folding and binding with alpha-helix-forming molecular recognition elements. Biochemistry 44(37):12454–12470
62. Mohan A, Oldfield CJ, Radivojac P, Vacic V, Cortese MS, Dunker AK, Uversky VN (2006) Analysis of molecular recognition features (MoRFs). J Mol Biol 362(5):1043–1059
63. Cheng Y, Oldfield CJ, Meng J, Romero P, Uversky VN, Dunker AK (2007) Mining alpha-helix-forming molecular recognition features with cross species sequence alignments. Biochemistry 46(47):13468–13477
64. Disfani FM, Hsu WL, Mizianty MJ, Oldfield CJ, Xue B, Dunker AK, Uversky VN, Kurgan L (2012) MoRFpred, a computational tool for sequence-based prediction and characterization of short disorder-to-order transitioning binding regions in proteins. Bioinformatics 28(12):i75–i83

65. Puntervoll P, Linding R, Gemund C, Chabanis-Davidson S, Mattingsdal M, Cameron S, Martin DM, Ausiello G, Brannetti B, Costantini A, Ferre F, Maselli V, Via A, Cesareni G, Diella F, Superti-Furga G, Wyrwicz L, Ramu C, McGuigan C, Gudavalli R, Letunic I, Bork P, Rychlewski L, Kuster B, Helmer-Citterich M, Hunter WN, Aasland R, Gibson TJ (2003) ELM server: a new resource for investigating short functional sites in modular eukaryotic proteins. Nucleic Acids Res 31(13):3625–3630
66. Dosztanyi Z, Meszaros B, Simon I (2009) ANCHOR: web server for predicting protein binding regions in disordered proteins. Bioinformatics 25(20):2745–2746
67. Dosztanyi Z, Csizmok V, Tompa P, Simon I (2005) The pairwise energy content estimated from amino acid composition discriminates between folded and intrinsically unstructured proteins. J Mol Biol 347(4):827–839
68. Peng Z, Wang C, Uversky VN, Kurgan L (2016) Prediction of disordered RNA, DNA, and protein binding regions using DisoRDPbind. In: Kloczkowski A, Zhou Y, Faraggi E, Yang Y (eds) Prediction of protein secondary structure and other one-dimensional structural properties, Methods in molecular biology. Springer, New York
69. Iakoucheva LM, Radivojac P, Brown CJ, O'Connor TR, Sikes JG, Obradovic Z, Dunker AK (2004) The importance of intrinsic disorder for protein phosphorylation. Nucleic Acids Res 32(3):1037–1049
70. Pejaver V, Hsu WL, Xin F, Dunker AK, Uversky VN, Radivojac P (2014) The structural and functional signatures of proteins that undergo multiple events of post-translational modification. Protein Sci 23(8):1077–1093
71. Uversky VN, Radivojac P, Iakoucheva LM, Obradovic Z, Dunker AK (2007) Prediction of intrinsic disorder and its use in functional proteomics. Methods Mol Biol 408:69–92
72. Ritter LM, Arakawa T, Goldberg AF (2005) Predicted and measured disorder in peripherin/rds, a retinal tetraspanin. Protein Pept Lett 12(7):677–686
73. Kukhtina V, Kottwitz D, Strauss H, Heise B, Chebotareva N, Tsetlin V, Hucho F (2006) Intracellular domain of nicotinic acetylcholine receptor: the importance of being unfolded. J Neurochem 97(Suppl 1):63–67
74. Yiu CP, Beavil RL, Chan HY (2006) Biophysical characterisation reveals structural disorder in the nucleolar protein, Dribble. Biochem Biophys Res Commun 343(1):311–318
75. Hinds MG, Smits C, Fredericks-Short R, Risk JM, Bailey M, Huang DC, Day CL (2007) Bim, Bad and Bmf: intrinsically unstructured BH3-only proteins that undergo a localized conformational change upon binding to prosurvival Bcl-2 targets. Cell Death Differ 14(1):128–136
76. Nardini M, Svergun D, Konarev PV, Spano S, Fasano M, Bracco C, Pesce A, Donadini A, Cericola C, Secundo F, Luini A, Corda D, Bolognesi M (2006) The C-terminal domain of the transcriptional corepressor CtBP is intrinsically unstructured. Protein Sci 15(5):1042–1050
77. Roy S, Schnell S, Radivojac P (2006) Unraveling the nature of the segmentation clock: intrinsic disorder of clock proteins and their interaction map. Comput Biol Chem 30(4):241–248
78. Popovic M, Coglievina M, Guarnaccia C, Verdone G, Esposito G, Pintar A, Pongor S (2006) Gene synthesis, expression, purification, and characterization of human Jagged-1 intracellular region. Protein Expr Purif 47(2):398–404
79. Iakoucheva LM, Brown CJ, Lawson JD, Obradovic Z, Dunker AK (2002) Intrinsic disorder in cell-signaling and cancer-associated proteins. J Mol Biol 323:573–584
80. Cheng Y, Le Gall T, Oldfield CJ, Dunker AK, Uversky VN (2006) Abundance of intrinsic disorder in proteins associated with cardiovascular disease. Biochemistry 45(35):10448–10460
81. Liu J, Perumal NB, Oldfield CJ, Su EW, Uversky VN, Dunker AK (2006) Intrinsic disorder in transcription factors. Biochemistry 45(22):6873–6888
82. Singh GP, Ganapathi M, Sandhu KS, Dash D (2006) Intrinsic unstructuredness and abundance of PEST motifs in eukaryotic proteomes. Proteins 62(2):309–315
83. Hansen JC, Lu X, Ross ED, Woody RW (2006) Intrinsic protein disorder, amino acid composition, and histone terminal domains. J Biol Chem 281(4):1853–1856
84. Peng Z, Mizianty MJ, Xue B, Kurgan L, Uversky VN (2012) More than just tails: intrinsic disorder in histone proteins. Mol Biosyst 8(7):1886–1901
85. Peng Z, Oldfield CJ, Xue B, Mizianty MJ, Dunker AK, Kurgan L, Uversky VN (2014) A creature with a hundred waggly tails: intrinsically disordered proteins in the ribosome. Cell Mol Life Sci 71(8):1477–1504
86. Xue B, Mizianty MJ, Kurgan L, Uversky VN (2012) Protein intrinsic disorder as a flexible

armor and a weapon of HIV-1. Cell Mol Life Sci 69(8):1211–1259

87. Fan X, Xue B, Dolan PT, LaCount DJ, Kurgan L, Uversky VN (2014) The intrinsic disorder status of the human hepatitis C virus proteome. Mol Biosyst 10(6):1345–1363
88. Xue B, Blocquel D, Habchi J, Uversky AV, Kurgan L, Uversky VN, Longhi S (2014) Structural disorder in viral proteins. Chem Rev 114(13):6880–6911
89. Meng F, Badierah RA, Almehdar HA, Redwan EM, Kurgan L, Uversky VN (2015) Unstructural biology of the dengue virus proteins. FEBS J 282(17):3368–3394
90. Goh GK, Dunker AK, Uversky VN (2015) Detection of links between Ebola nucleocapsid and virulence using disorder analysis. Mol Biosyst 11(8):2337–2344
91. Goh GK, Dunker AK, Uversky VN (2015) Shell disorder, immune evasion and transmission behaviors among human and animal retroviruses. Mol Biosyst 11(8):2312–2323
92. Dolan PT, Roth AP, Xue B, Sun R, Dunker AK, Uversky VN, LaCount DJ (2015) Intrinsic disorder mediates hepatitis C virus core-host cell protein interactions. Protein Sci 24(2):221–235
93. Haynes C, Iakoucheva LM (2006) Serine/arginine-rich splicing factors belong to a class of intrinsically disordered proteins. Nucleic Acids Res 34(1):305–312
94. Bustos DM, Iglesias AA (2006) Intrinsic disorder is a key characteristic in partners that bind 14-3-3 proteins. Proteins 63(1):35–42
95. Denning DP, Patel SS, Uversky V, Fink AL, Rexach M (2003) Disorder in the nuclear pore complex: the FG repeat regions of nucleoporins are natively unfolded. Proc Natl Acad Sci U S A 100(5):2450–2455
96. Boeckmann B, Bairoch A, Apweiler R, Blatter M-C, Estreicher A, Gasteiger E, Martin MJ, Michoud K, O'Donovan C, Phan I, Pilbout S, Schneider M (2003) The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. Nucleic Acids Res 31:365–370
97. Vucetic S, Obradovic Z, Vacic V, Radivojac P, Peng K, Iakoucheva LM, Cortese MS, Lawson JD, Brown CJ, Sikes JG, Newton CD, Dunker AK (2005) DisProt: a database of protein disorder. Bioinformatics 21(1):137–140
98. Sickmeier M, Hamilton JA, LeGall T, Vacic V, Cortese MS, Tantos A, Szabo B, Tompa P, Chen J, Uversky VN, Obradovic Z, Dunker AK (2007) DisProt: the database of disordered proteins. Nucleic Acids Res 35(Database issue):D786–D793
99. Vacic V, Uversky VN, Dunker AK, Lonardi S (2007) Composition profiler: a tool for discovery and visualization of amino acid composition differences. BMC Bioinformatics 8:211
100. Peng K, Vucetic S, Radivojac P, Brown CJ, Dunker AK, Obradovic Z (2005) Optimizing long intrinsic disorder predictors with protein evolutionary information. J Bioinform Comput Biol 3(1):35–60
101. Obradovic Z, Peng K, Vucetic S, Radivojac P, Dunker AK (2005) Exploiting heterogeneous sequence properties improves prediction of protein disorder. Proteins 61(Suppl 7):176–182
102. Peng K, Radivojac P, Vucetic S, Dunker AK, Obradovic Z (2006) Length-dependent prediction of protein intrinsic disorder. BMC Bioinformatics 7(1):208
103. Xue B, Dunbrack RL, Williams RW, Dunker AK, Uversky VN (2010) PONDR-FIT: a meta-predictor of intrinsically disordered amino acids. Biochim Biophys Acta 1804(4):996–1010
104. Oldfield CJ, Cheng Y, Cortese MS, Brown CJ, Uversky VN, Dunker AK (2005) Comparing and combining predictors of mostly disordered proteins. Biochemistry 44(6):1989–2000
105. Oates ME, Romero P, Ishida T, Ghalwash M, Mizianty MJ, Xue B, Dosztanyi Z, Uversky VN, Obradovic Z, Kurgan L, Dunker AK, Gough J (2013) D(2)P(2): database of disordered protein predictions. Nucleic Acids Res 41(Database issue):D508–D516
106. Di Domenico T, Walsh I, Martin AJ, Tosatto SC (2012) MobiDB: a comprehensive database of intrinsic protein disorder annotations. Bioinformatics 28(15):2080–2081
107. Potenza E, Domenico TD, Walsh I, Tosatto SC (2015) MobiDB 2.0: an improved database of intrinsically disordered and mobile proteins. Nucleic Acids Res 43(Database issue):D315–D320
108. Szklarczyk D, Franceschini A, Kuhn M, Simonovic M, Roth A, Minguez P, Doerks T, Stark M, Muller J, Bork P, Jensen L.J, von Mering C (2011) The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored. Nucleic Acids Res, 2011;39:D561–568
109. Vucetic S, Brown CJ, Dunker AK, Obradovic Z (2003) Flavors of protein disorder. Proteins 52:573–584
110. Prilusky J, Felder CE, Zeev-Ben-Mordehai T, Rydberg EH, Man O, Beckmann JS, Silman I, Sussman JL (2005) FoldIndex: a simple tool to predict whether a given protein

sequence is intrinsically unfolded. Bioinformatics 21(16):3435–3438

111. Dosztanyi Z, Csizmok V, Tompa P, Simon I (2005) IUPred: web server for the prediction of intrinsically unstructured regions of proteins based on estimated energy content. Bioinformatics 21(16):3433–3434
112. Campen A, Williams RM, Brown CJ, Meng J, Uversky VN, Dunker AK (2008) TOP-IDP-scale: a new amino acid scale measuring propensity for intrinsic disorder. Protein Pept Lett 15(9):956–963
113. Huang F, Oldfield C, Meng J, Hsu WL, Xue B, Uversky VN, Romero P, Dunker AK (2012) Subclassifying disordered proteins by the CH-CDF plot method. Pac Symp Biocomput 128–139
114. Huang F, Oldfield CJ, Xue B, Hsu WL, Meng J, Liu X, Shen L, Romero P, Uversky VN, Dunker A (2014) Improving protein order-disorder classification using charge-hydropathy plots. BMC Bioinformatics 15(Suppl 17):S4
115. Peng K, Radivojac P, Vucetic S, Dunker AK, Obradovic Z (2006) Length-dependent prediction of protein intrinsic disorder. BMC Bioinformatics 7
116. Ishida T, Kinoshita K (2007) PrDOS: prediction of disordered protein regions from amino acid sequence. Nucleic Acids Res 35(Web Server issue):W460–W464
117. Walsh I, Martin AJ, Di Domenico T, Tosatto SC (2012) ESpritz: accurate and fast prediction of protein disorder. Bioinformatics 28(4):503–509
118. Linding R, Jensen LJ, Diella F, Bork P, Gibson TJ, Russell RB (2003) Protein disorder prediction: implications for structural proteomics. Structure 11(11):1453–1459
119. Linding R, Russell RB, Neduva V, Gibson TJ (2003) GlobPlot: exploring protein sequences for globularity and disorder. Nucleic Acids Res 31(13):3701–3708
120. Yang ZR, Thomson R, McNeil P, Esnouf RM (2005) RONN: the bio-basis function neural network technique applied to the detection of natively disordered regions in proteins. Bioinformatics 21(16):3369–3376
121. Dyson HJ, Wright PE (2002) Coupling of folding and binding for unstructured proteins. Curr Opin Struct Biol 12(1):54–60
122. Dyson HJ, Wright PE (2005) Intrinsically unstructured proteins and their functions. Nat Rev Mol Cell Biol 6(3):197–208
123. Vacic V, Oldfield CJ, Mohan A, Radivojac P, Cortese MS, Uversky VN, Dunker AK (2007) Characterization of molecular recognition features, MoRFs, and their binding partners. J Proteome Res 6(6):2351–2366
124. Uversky VN (2013) Intrinsic disorder-based protein interactions and their modulators. Curr Pharm Des 19(23):4191–4213
125. Daily KM, Radivojac P, Dunker AK. Intrinsic disorder and protein modifications: building an SVM predictor for methylation. In: IEEE symposium on computational intelligence in bioinformatics and computational biology, CIBCB 2005, San Diego, CA, USA, November 2005, pp 475–481

Chapter 12

Intrinsic Disorder and Semi-disorder Prediction by SPINE-D

Tuo Zhang, Eshel Faraggi, Zhixiu Li, and Yaoqi Zhou

Abstract

Over the past decade, it has become evident that a large proportion of proteins contain intrinsically disordered regions, which play important roles in pivotal cellular functions. Many computational tools have been developed with the aim of identifying the level and location of disorder within a protein. In this chapter, we describe a neural network based technique called SPINE-D that employs a unique three-state design and can accurately capture disordered residues in both short and long disordered regions. SPINE-D was trained on a large database of 4229 non-redundant proteins, and yielded an AUC of 0.86 on a cross-validation test and 0.89 on an independent test. SPINE-D can also detect a semi-disordered state that is associated with induced folders and aggregation-prone regions in disordered proteins and weakly stable or locally unfolded regions in structured proteins. We implement an online web service and an offline stand-alone program for SPINE-D, they are freely available at http://sparks-lab.org/SPINE-D/. We then walk you through how to use the online and offline SPINE-D in making disorder predictions, and examine the disorder and semi-disorder prediction in a case study on the p53 protein.

Key words Intrinsically disorder, Semi-disorder, Prediction, Neural network, Protein induced folding, Protein aggregation, SPINE-D, P53

1 Introduction

Intrinsically disordered proteins (IDP) and intrinsically disordered regions (IDR) in proteins do not fold into stable three-dimensional structures under general physiological conditions. Although lacking specific structures IDPs and IDRs play crucial functional roles in many biological processes, such as transcriptional regulation, translation and cellular signal transduction [1–8]. Indeed, approximately a third of all eukaryotic proteins have been identified as including disordered regions greater than 30 residues, with 75% of mammalian signaling proteins being disordered [9]. IDPs and IDRs are also shown to be prevalent in various human diseases including cancer, cardiovascular disease, and genetic diseases [10–12], and are suggested as important targets for drug discovery [8, 13].

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_12,

Despite of their functional and theoretical importance, identifying IDPs and IDRs by experiments is not an easy task. Lack of electron density in X-ray crystallographic studies can imply disorder, however, this method only works for proteins of relatively short disordered regions. In order to characterize proteins with high levels of disorder, one needs to perform solution based methods, such as Nuclear Magnetic Resonance (NMR), circular dichroism (CD), and small angle X-ray scattering (SAXS) [8, 14–21]. These experimental techniques are usually costly and time-consuming, thus cannot be applied to large-scale protein analysis. This stimulates the development of high-throughput computational methods for predicting protein disorder.

Existing computational methods include amino acid propensity based methods [22–25], machine-learning-based methods [26–36], clustering-based methods [37, 38], template-based methods [39], and meta-approaches [40–44]. Given the fact that amino acid residues in short (≤30 residues) and long (>30 residues) disordered regions show different preferences in composition of amino acids, most of the existing methods implemented separate predictors for short and long disordered regions from which the predictions were combined to yield a balanced two-state prediction [45–48]. Here we describe a single neural-network-based method, SPINE-D, which makes an initial three-state (ordered residues and disordered residues in short and long disordered regions), rather than the commonly used two-state prediction of disorder (ordered residues and disordered residues) [26]. This unique design makes SPINE-D highly accurate in detecting residues in both short and long disordered regions.

SPINE-D takes a protein sequence as input and calculates a set of residue-level and window-level features. Those features are then fed into a two-hidden-layer neural network to yield a three-state prediction first and reduce it into a two-state prediction afterwards [26]. SPINE-D was trained on a large database of 4229 non-redundant protein chains, which was randomly divided into two subsets for cross-validation and independent tests, respectively. We further tested SPINE-D on a disorder benchmark dataset by comparing its prediction with 11 competing methods, where SPINE-D yields the highest area under the curve (AUC), the highest Mathews correlation coefficient (MCC) for residue-based prediction, and the lowest mean square error in predicting disorder contents [26]. Moreover, SPINE-D was officially assessed to be among the best performing methods in the 9th Critical Assessment of Structure Prediction techniques (CASP9) [49].

For a given protein sequence, SPINE-D can predict the probability of each amino-acid residue in the sequence to be disordered. This probability prediction appears to be physically meaningful. In particular, a semi-disordered state, which is about 50% being disordered or ordered, can capture protein regions that are semi-collapsed or

semi-structured with intermediate levels of predicted secondary structure and solvent accessibility. Further investigation indicates that some semi-disordered regions participate in induced folding and others play key roles in protein aggregation [50].

2 Materials and Methods

2.1 Methods

SPINE-D implements a two-hidden-layer neural network with an additional one-layer filter for smoothing the prediction. Each of the two hidden layers contains 51 hidden neurons and one bias, and the filter layer contains 11 hidden neurons. SPINE-D employs a hyperbolic activation function and a guided learning technique that assigns a lower weight for residues further apart in sequence distance within a pre-defined sliding window [51]. To reduce the fluctuations caused by the random selection of initial weights, we trained five independent predictors and the final prediction is based on the average result of the five predictors.

Given a protein sequence, an external program PSI-BLAST [52] is employed to generate position-specific substitution matrix (PSSM) by searching the query sequence with three iterations against the NCBI's non-redundant protein sequence database. The PSSM is then used as input to an in-house program, SPINE-X [53–55], to predict torsion angle fluctuation, secondary structure, and solvent accessibility of the query sequence. Next, SPINE-D calculates three sets of features: residue-level features, window-level features as well as one terminal tag. The residue-level features include representative physical parameters (7 dimensions); PSSM (20 dimensions); predicted secondary structure (3 dimensions); predicted solvent accessibility (1 dimension); and predicted torsion angle fluctuation (2 dimensions). A sliding window of size 21 is further employed to take into account the local information around the residue to be predicted. The window-level features include amino acid composition (20 dimensions); local compositional complexity (1 dimension); and predicted secondary structure content (3 dimensions). The window-level features are calculated based on the current residue plus 15 residues on either side. The terminal tag (1 dimension) indicates the relative position of a residue in the given sequence, i.e., five residues in N-terminus are encoded as -1.0, -0.8, -0.6, -0.4, -0.2 and five residues on C-terminus as 0.2, 0.4, 0.6, 0.8, 1.0 and the rest as 0.0. Overall, $(7+20+3+1+2)*21+(20+1+3)+1=718$ features per residue are collected and fed into a neural network to make a three-state prediction: ordered residue, residue in short disordered region or residue in long disordered region.

The three-state prediction is reduced to a two-state prediction by simply adding the probabilities in short and long disordered regions. A cutoff of 0.06 is chosen for assigning a binary prediction

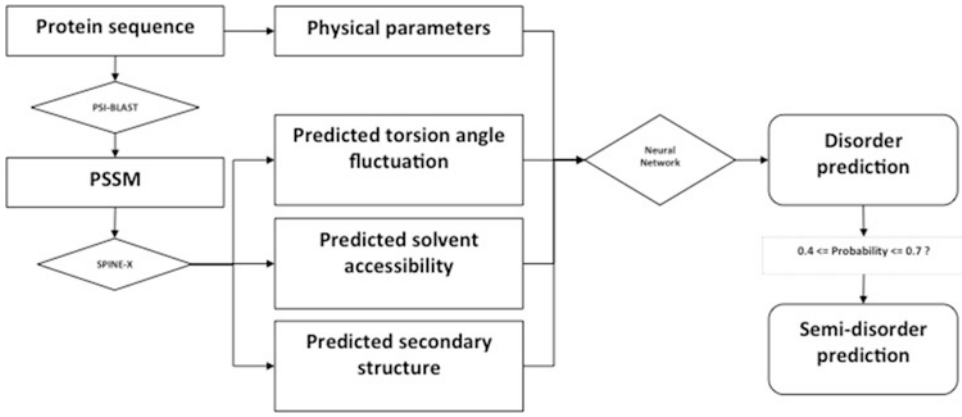


Fig. 1 Workflow chart of SPINE-D

(ordered or disordered). Then the predicted probability is rescaled to yield a more natural separation (0.5) between ordered and disordered states [50]. This is done by linearly transforming predicted probability from [0, 0.06] to [0, 0.5] and [0.06, 1] to [0.5, 1]. Finally, SPINE-D defines a semi-disordered state by selecting residues with rescaled probability predictions within range [0.4, 0.7]. A workflow chart of SPINE-D is shown in Fig. 1.

2.2 Databases

We built a large database of 4229 non-redundant protein chains with 1,036,634 (about 90%) ordered residues and 103,252 (about 10%) disordered residues. Four thousand one hundred and seventy five protein chains were retrieved from the protein databank [56] by carefully selecting X-ray structures with residues missing coordinates (recorded in REMARK465 section of a PDB file); the remaining 72 chains were fully disordered proteins obtained from the Disprot database [57]. We named this database DM4229.

We randomly selected 3000 chains (referred to as DM3000) from the DM4229 database to design our neural-network predictor and to perform a ten-fold cross-validation test. The remaining 1229 chains (referred to as DM1229) were used as an independent test set.

In addition, we prepared a benchmark database of 477 non-redundant protein chains and named it SL477. The SL477 database was built based on a refined Disprot database that tries to give reliable disorder and order contents [58]. We further selected a subset of 329 chains (referred to as SL329) from SL477 by removing chains with more than 25% sequence identity to that in the DM4229 database.

We further filtered the DM4229 database by removing chains from Disprot and chains that are similar to the benchmark SL477 set. The remaining 4080 X-ray structures (referred to as DX4080) were used to examine the dependence of our technique on training databases.

All databases are freely available on our webserver.

3 Webserver

An online web service for predicting protein disorder using SPINE-D is freely available at http://sparks-lab.org/SPINE-D/.

3.1 Webserver Input

The required input is either a protein sequence in FASTA format (*see* **Notes 1** and **2**) or a pre-generated PSI-BLAST profile (*see* **Notes 2** and **3**) for the query sequence. If the user chooses to input a protein sequence, the webserver will call PSI-BLAST to generate a PSI-BLAST profile for it. On the other hand, if the user uploads a PSI-BLAST profile, the webserver will skip the PSI-BLAST profile generation step and use the user-provided profile for the downstream prediction. It is up to the user to decide which file to input, however, using pre-calculated PSI-BLAST profile will significantly speed up the processing time (*see* **Note 3**).

The webserver also provides two optional inputs: e-mail address and target ID. In case an e-mail address is provided, the user will receive a notification e-mail including a hyperlink to the result page once the webserver completes processing the user's request. The webserver assigns a unique job ID for each prediction request it receives. The job ID is an integer that does not give any hints about what the query sequence is. The user has the option to enter a target ID, i.e., a name or a description for the query sequence, to help trace a prediction request. If provided, the target ID will be shown in the title of the notification e-mail. Although the e-mail address and target ID are optional, we do recommend that the user provides both information, because this will help the user quickly find the prediction result from a previously submitted request (*see* **Note 4**).

3.2 Submitting to the Webserver

Figure 2 displays the input page of our SPINE-D webserver. To submit a prediction request, please follow the four easy steps below:

1. Enter e-mail address (recommended although optional, *see* **Note 4**). If provided, a notification e-mail including a hyperlink to the result page will be sent to the user once the prediction is complete.
2. Enter target ID (recommended although optional, *see* **Note 4**). If provided, the target ID will be included in the title of the notification e-mail.
3. Enter the query protein sequence. The user has two options:
   (a) Upload a PSI-BLAST profile (*see* **Notes 2** and **3**) by clicking the "Choose File" button and choosing a file. Once finish uploading, the filename will be shown on the right of the "Choose File" button. An example of PSI-BLAST profile is available by clicking the "see example" link in case the user wants to know what a profile looks like.

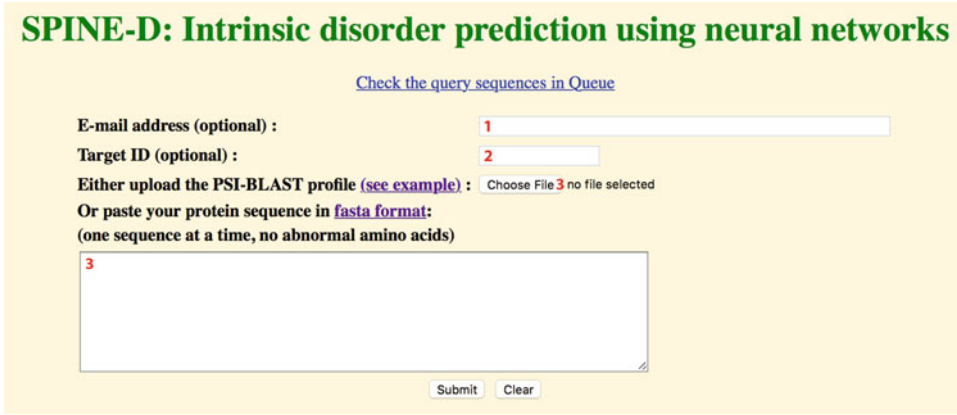


Fig. 2 Input page of SPINE-D webserver. Numbers 1–4 in *red* indicate the steps for submitting a query sequence to the webserver

(b) Copy and paste a protein sequence formatted in the FASTA format (*see* **Note 1** and **2**) into the text field. Please make sure that only one protein sequence is provided at a time and the sequence does not include any abnormal amino acid types (*see* **Note 1**). The "fasta format" link opens the Wikipedia page explaining the FASTA format.

4. Click "Submit" button to submit your request. In case you want to clear all entered information, simply click the "Clear" button and everything will be removed.

The webserver will perform a format check on the data submitted. If the data fail the format check, the webserver will print out an error message; otherwise it will start to make prediction. The user will be redirected to a waiting page (Fig. 3), which shows the unique job ID assigned to the query (number 1 in red, *see* **Note 4**), the hyperlink to the result page (number 2 in red) and the information the user has submitted (number 3 in red). This waiting page will automatically refresh every 10 s until the prediction is complete, then the user will be redirected to the result page. If an e-mail address is provided, a notification e-mail will be sent to the user. The prediction time usually takes a couple of minutes per protein sequence but may take longer time depending on the size of the query sequence and the load of our webserver.

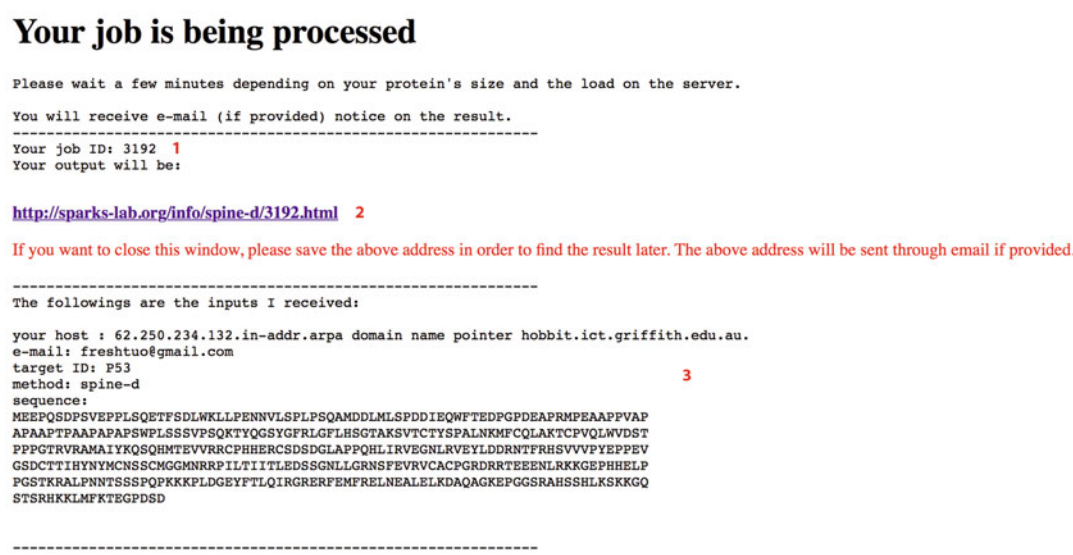
3.3 Webserver Output

Figure 4 shows a screenshot of an example webserver output page. The query protein sequence is arranged in multiple rows where each row represents one amino acid residue. The disorder prediction of each residue is shown in four columns (number 2 in red):

Column 1: index indicating the position of a residue in the query sequence.

Column 2: amino acid type of a residue.

Column 3: binary disorder prediction, where "D" indicates disordered and "O" indicates ordered.



Your job is being processed

Please wait a few minutes depending on your protein's size and the load on the server.

You will receive e-mail (if provided) notice on the result.

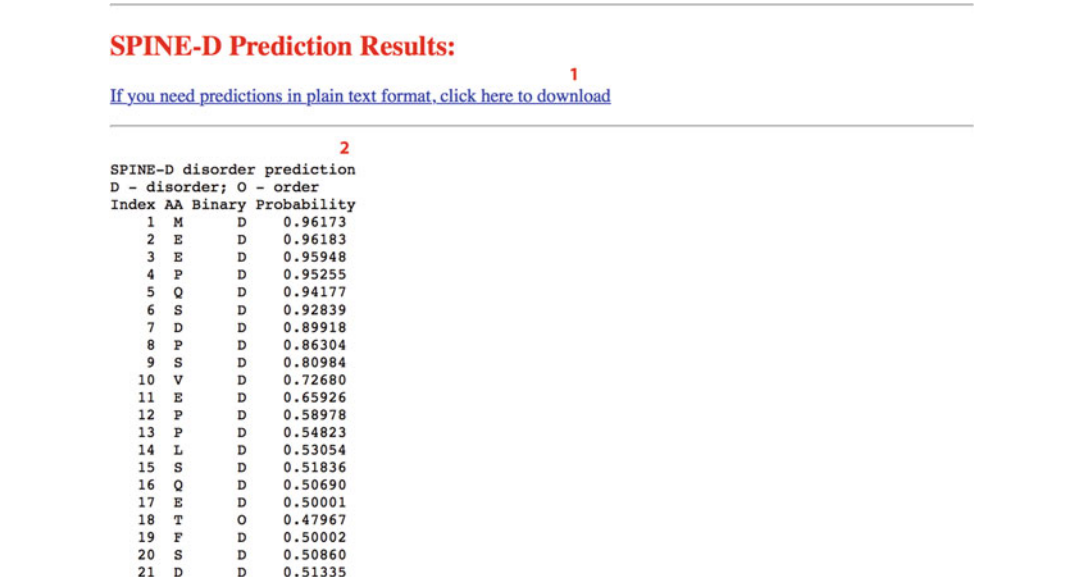
Your job ID: 3192 1
Your output will be:

http://sparks-lab.org/info/spine-d/3192.html 2

If you want to close this window, please save the above address in order to find the result later. The above address will be sent through email if provided.

The followings are the inputs I received:

your host : 62.250.234.132.in-addr.arpa domain name pointer hobbit.ict.griffith.edu.au.
e-mail: freshtuo@gmail.com
target ID: P53 3
method: spine-d
sequence:
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLPSQAMDDLMLSPDDIEQWFTEDPGPDEAPRMPEAAPPVAP
APAAPTPAAPAPAPSWPLSSSVPSQKTYQGSYGFRLGFLHSGTAKSVTCTYSPALNKMFCQLAKTCPVQLWVDST
PPPGTRVRAMAIYKQSQHMTEVVRRCPHHERCSDSDGLAPPQHLIRVEGNLRVEYLDDRNTFRHSVVVPYEPPEV
GSDCTTIHYNYMCNSSCMGGMNRRPILTIITLEDSSGNLLGRNSFEVRVCACPGRDRRTEEENLRKKGEPHHELP
PGSTKRALPNNTSSSPQPKKKPLDGEYFTLQIRGRERFEMFRELNEALELKDAQAGKEPGGSRAHSSHLKSKKGQ
STSRHKKLMFKTEGPDSD



SPINE-D Prediction Results:

1
If you need predictions in plain text format, click here to download

2
SPINE-D disorder prediction
D - disorder; O - order

Index	AA	Binary	Probability
1	M	D	0.96173
2	E	D	0.96183
3	E	D	0.95948
4	P	D	0.95255
5	Q	D	0.94177
6	S	D	0.92839
7	D	D	0.89918
8	P	D	0.86304
9	S	D	0.80984
10	V	D	0.72680
11	E	D	0.65926
12	P	D	0.58978
13	P	D	0.54823
14	L	D	0.53054
15	S	D	0.51836
16	Q	D	0.50690
17	E	D	0.50001
18	T	O	0.47967
19	F	D	0.50002
20	S	D	0.50860
21	D	D	0.51335

Fig. 4 Result page of SPINE-D webserver. Numbers 1–2 in *red* indicate the download link of the prediction result in plain text format, and the prediction result shown in the webpage

Column 4: predicted probability of a residue being disordered. A residue with a probability greater than 0.5 is assigned "D" (disordered) otherwise "O" (ordered).

A downloadable link (number 1 in red) is provided on top of the page in case the user needs a plain text format of the prediction shown in the output page.

4 Stand-Alone Program

A stand-alone program of SPINE-D is available on our web server page. The program can be run on a Linux based operating system.

4.1 Installation

The stand-alone program requires python 2.6+ and a Fortran compiler (either ifort or gfortran) (*see* **Note** 7). In addition, the user needs to install the following three programs:

1. PSI-BLAST with non-redundant protein sequences (NR) database.
   ftp://ftp.ncbi.nlm.nih.gov/blast/
   PSI-BLAST [52] is an external program for generating PSSM, which is an important input for SPINE-X and SPINE-D.
2. SPINE-X.
   http://sparks-lab.org/SPINE-X/
   SPINE-X [53–55] is an internal program for predicting secondary structure, solvent accessibility and torsion angle fluctuation.
3. IUpred.
   http://iupred.enzim.hu/
   IUpred [23] is an external program for generating features used in predicting torsion angle fluctuation [54].

To install the stand-alone program, please follow the instructions below:

1. tar zxvf spinedlocal_v2.0.tar.gz
   This command unzips the installation package into a folder named "SPINE-D-local."
2. cd SPINE-D-local/bin
3. Edit the "run_spine-d" file and:
   (a) Set the path for PSI-BLAST executable file.
   (b) Set the path for NR database.
   (c) Set the path for SPINE-X package.
   (d) Set the path for SPINE-D package.
   (e) Set the path of IUpred executable file.
   (f) Set the Fortran compiler by choosing either "ifort" or "gfortran."
   (g) Set the number of cores for running PSI-BLAST.
   (h) Set whether or not to keep intermediate prediction files.
4. Run a quick test to make sure SPINE-D is properly installed.
   (a) cd ../test/

(b) sh test_example

This shell script runs SPINE-D on an example protein sequence and generates a prediction file named "eg.spd" in the "../predout/" directory.

(c) diff ../predout/eg.spd eg.spd

This command compares the prediction file with a standard file. The two files should be identical if the user has successfully installed the SPINE-D. It should be noted that the result for the test sequence will be slightly different if a user employs a different PSSM resulted from a different sequence library.

4.2 Stand-Alone Program Input

The mandatory input file for the stand-alone program is a protein sequence in FASTA format. Optionally, the user can also provide a PSI-BLAST profile. If provided, the stand-alone program will skip the profile generation step, which will significantly reduce the prediction time (*see* **Note 3**).

The mandatory FASTA file and the optional PSI-BLAST profile must share the same filename while the filename extension for the FASTA file and the PSI-BLAST profile should be ".fasta" and ".mat," respectively (*see* **Note 5**).

4.3 Running Stand-Alone Program

Please follow two steps to run the stand-alone program:

1. cd to-where-you-install-SPINE-D/bin/

   This command changes the current working directory to where the user installs the stand-alone executable files.

2. ./run_spined input-file-path input-file-name

The stand-alone program will search in the "input-file-path" directory for the mandatory FASTA file "input-file-name.fasta" and the optional PSI-BLAST profile "input-file-name.mat." Depending on the input files provided, the program will decide whether or not to skip the PSI-BLAST profile generation step.

Please make sure provide a unique input-file-name that has not been used before (*see* **Note 6**).

The stand-alone SPINE-D package allows the user to make disorder prediction on a set of protein sequences using a simple shell script. Please *see* **Note 8** to learn how to set it up.

4.4 Stand-Alone Program Output

Once the prediction is complete, the result file can be found in the "where-you-install-SPINE-D/predout/" directory, and will be named "input-file-name.spd." The result file will be in plain text format, one amino acid residue per row where each row consists of three columns:

Column 1: amino acid type of a residue.

Column 2: binary disorder prediction, where "D" indicates disordered and "O" indicates ordered.

Column 3: predicted probability of a residue being disordered. A residue with a probability greater than 0.5 is assigned "D" (disordered) otherwise "O" (ordered).

5 Case Study

The p53 protein is a famous transcription factor known as the "guardian of the genome" due to its critical function in regulating cell division, repairing damaged DNA and preventing tumor formation. This protein locates in the nucleus of cells throughout the body, where it binds directly to DNA. When the DNA in a cell becomes damaged, p53 determines whether the DNA will be repaired or the damaged cell will self-destruct. If the DNA can be repaired, p53 activates other genes to fix the damage. If the DNA cannot be repaired, this protein prevents the cell from dividing and signals it to undergo apoptosis [59].

These complicated functions of the p53 protein are achieved by regulating a large signaling network. In fact, p53 is located at the center of this network and it transduces signals by physically interacting with a large number of proteins. p53's ability to bind to numerous protein partners is given by its intrinsically disordered domains. The p53 protein consists of three domains: the transactivation domain (p53TAD) at the N-terminal; the regulatory domain at the C-terminal; and the DNA binding domain in the middle. While the DNA binding domain is structured, the two terminal domains are characterized as being intrinsically disordered. These two disordered domains help mediate ~70% protein–protein interactions required by the p53 protein [60].

Among a variety of protein–protein interactions, the p53-Mdm2 interaction is of special interest due to its direct relation to the oncogenesis. The Mdm2 protein inactivates p53 by binding to its transactivation domain (p53TAD). This interaction prevents p53 from activating its target genes [60]. In its free form, p53TAD exists in equilibrium between disordered and partially helical conformations, whereas residues 19–25 form a stable amphipathic alpha-helix in the Mdm2 complex. This residual helicity forms the Mdm2 binding site and determines the binding affinity to Mdm2 in vitro and in cells. Conserved proline residues (Pro12, Pro13 and Pro27) outside the Mdm2 binding site preserve the defined levels of helicity, which is ultimately required for productive p53 signaling [61].

Figure 5 shows the predicted disordered probability at each residue position in the p53 protein (Uniprot ID: P04637) by using our SPINE-D webserver. The red dotted line indicates a separation between disordered and ordered residues, where points above the line are predicted as disordered and points underneath are predicted as ordered. The red bar on top of the figure labels 7 native

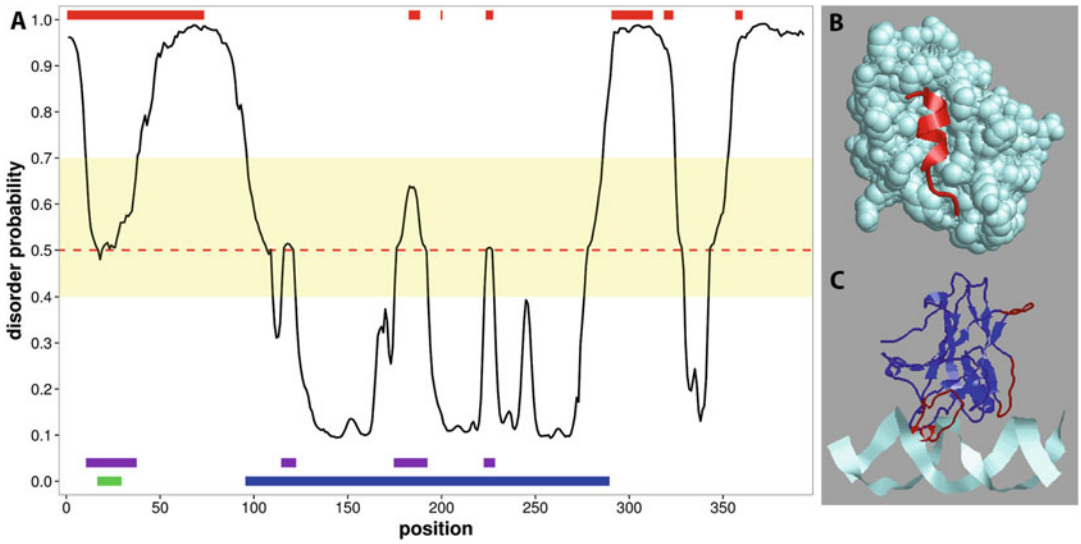


Fig. 5 Disorder and semi-disorder prediction on the p53 protein. (**a**) The *black line* indicates the predicted disorder probability at each residue position. The *red dotted* line separates the predicted disordered residues from the ordered residues. The *yellow* block highlights the semi-disordered residues of predicted disorder probability within [0.4, 0.7]. The red bars on top indicate the native disorder annotation in the Disprot database. The purple bars at the bottom indicate the predicted semi-disordered regions. The *green bar* indicates the Mdm2 binding site in the transactivation domain (p53TAD) of p53. The *blue bar* indicates the DNA-binding domain of p53. (**b**) The p53-Mdm2 complex (PDB ID: 1YCR). The Mdm2 protein is colored in cyan. The *red* helix indicates the semi-disordered Mdm2 binding site within the p53TAD. (**c**) The p53-DNA complex (PDB ID: 1TSR). The DNA is colored in *cyan*. The DNA-binding domain of p53 is colored in *blue*, in which the predicted semi-disordered regions are colored in *red*

disordered regions annotated in the Disprot database (Disprot ID: DP00086), including the entire p53TAD domain (residues 1–73) at N-terminus; 3 short disordered regions (residues 183–188, 200, 224–227) inside the DNA-binding domain; and 3 disordered regions (residues 291–312, 319–323, 357–360) in the C-terminal regulatory domain. The disorder prediction from SPINE-D is in agreement with the native disorder annotation. In particular, SPINE-D has successfully recovered the two terminal disordered domains as well as the structured domain. It captures eight out of the nine native disordered regions; the one missing region only contains a single amino acid (residue 200), which is too short to form an order–disorder–order transition.

SPINE-D can also predict a semi-disordered state with predicted probability in [0.4, 0.7]. The corresponding probability area in the p53 protein is highlighted in yellow in Fig. 5. Out of the 393 residues, 98 are predicted as being semi-disordered; they form four semi-disordered regions (labeled as purple bars) that undergo either disorder–order–disorder transitions or order–disorder–order transitions. One semi-disordered region (residues 11–37, colored in red in Fig. 5b) overlaps the Mdm2 binding site (labeled as a green bar) as

well as the conserved proline residues in flanking regions in the disordered p53TAD domain. This is in agreement with the finding that the binding site is semi-structured in its free form and folds into a alpha-helix upon binding to the Mdm2 protein [61], and that the proline residues are important in maintaining certain extent of disorder within p53TAD to temper affinity for Mdm2 [62]. The other three semi-disordered regions (residues 115–122, 175–192, 223–228, colored in red in Fig. 5c) lie in the DNA-binding domains (labeled as a blue bar). These semi-disordered regions play a role in providing flexibilities in a structured region that are needed for interacting with DNA molecules.

This case study demonstrates the quality of SPINE-D of predicting disordered residues in both short and long disordered regions, and the utility of using semi-disordered prediction in aid of identifying protein-induced folding regions in intrinsically disordered regions and locally unfolded regions in structured regions.

6 Notes

1. The input sequence must be a protein sequence in FASTA format, and should not include amino acids other than the 20 common amino acid types: ACDEFGHIKLMNPQRSTVWY.
2. If the user provides both the PSI-BLAST profile and the protein sequence, the webserver will pick the PSI-BLAST profile and skip the time-consuming step of PSI-BLAST profile generation.
3. The most time consuming step is the PSI-BLAST profile generation since it requires searching the query sequence against a very large non-redundant protein sequence database. Depending on the number of homologous sequences found in the database, the searching process might take a couple of minutes to half an hour. The rest steps, including feature generation, neural network calculation, prediction collecting and reformatting, are very fast, usually takes less than a minute.
4. The prediction result is stored on our webserver and can be accessed by a hyperlink (a http address plus a unique job ID). If possible, please provide both an e-mail address and a target ID for the query sequence when submitting a prediction request. At least we recommend that the user provides an e-mail address, otherwise the user will need to either write down the unique job ID assigned to the request or keep the web browser open for a while until the prediction result pops out. On the other hand, an e-mail address and a target ID can help the user quickly trace the prediction result at any time.
5. By default, the stand-alone program assumes that: (a) the filename extension of the input FASTA file and the input PSI-BLAST profile are ".fasta" and ".mat," respectively; (b) the two

input files are located in the same directory. If this is not the case, the user can modify the corresponding inputs (highlighted in red) in the last line of the "to-where-you-install-SPINE-D/bin/run_spine-d" file:

- Python run_spine-d.py input-FASTA-path/input-FASTA-filename .fasta input-profile-path/input-profile-filename .mat targetid

6. It is important to assign a unique filename for the input file. This is because the stand-alone SPINE-D is designed to not allow overwrite to existing files. In case it detects an identical existing filename, the program will terminate and output an error message.
7. Some codes in the stand-alone program are written in Fortran, and they will be compiled when the user runs SPINE-D for the first time. The program will print out some compiling messages only for the first time usage.
8. In case the user needs to run SPINE-D on a batch of protein sequences, the stand-alone SPINE-D package provides a shell script "batch.sh" in the "where-you-install-SPINE-D/bin" directory. The script requires the path of input files as well as a user-provided file that lists the filenames of all input files. Once they are provided, the user can simply type "sh batch.sh" and the script will take care the rest.

Acknowledgements

This study was financially supported by National Health and Medical Research Council (1059775 and 1083450) of Australia and Australian Research Council's Linkage Infrastructure, Equipment and Facilities funding scheme (project number LE150100161) to Y.Z. We also gratefully acknowledge the support of the Griffith University eResearch Services Team and the use of the High Performance Computing Cluster "Gowonda" to complete this research. This research/project has also been undertaken with the aid of the research cloud resources provided by the Queensland Cyber Infrastructure Foundation (QCIF).

References

1. Uversky VN, Oldfield CJ, Dunker AK (2005) Showing your ID: intrinsic disorder as an ID for recognition, regulation and cell signaling. J Mol Recognit 18(5):343–384. doi:10.1002/jmr.747
2. Liu J, Perumal NB, Oldfield CJ, Su EW, Uversky VN, Dunker AK (2006) Intrinsic disorder in transcription factors. Biochemistry 45(22):6873–6888. doi:10.1021/bi0602718
3. Galea CA, Wang Y, Sivakolundu SG, Kriwacki RW (2008) Regulation of cell division by intrinsically unstructured proteins: intrinsic flexibility, modularity, and signaling conduits. Biochemistry 47(29):7598–7609. doi:10.1021/bi8006803
4. Fuxreiter M, Tompa P, Simon I, Uversky VN, Hansen JC, Asturias FJ (2008) Malleable machines take shape in eukaryotic transcrip-

tional regulation. Nat Chem Biol 4(12):728–737. doi:10.1038/nchembio.127

5. Dunker AK, Cortese MS, Romero P, Iakoucheva LM, Uversky VN (2005) Flexible nets. The roles of intrinsic disorder in protein interaction networks. FEBS J 272(20):5129–5148. doi:10.1111/j.1742-4658.2005.04948.x
6. Wright PE, Dyson HJ (1999) Intrinsically unstructured proteins: re-assessing the protein structure-function paradigm. J Mol Biol 293(2):321–331. doi:10.1006/jmbi.1999.3110
7. Xie H, Vucetic S, Iakoucheva LM, Oldfield CJ, Dunker AK, Uversky VN, Obradovic Z (2007) Functional anthology of intrinsic disorder. 1. Biological processes and functions of proteins with long disordered regions. J Proteome Res 6(5):1882–1898. doi:10.1021/pr060392u
8. Habchi J, Tompa P, Longhi S, Uversky VN (2014) Introducing protein intrinsic disorder. Chem Rev 114(13):6561–6588. doi:10.1021/cr400514h
9. Ward JJ, Sodhi JS, McGuffin LJ, Buxton BF, Jones DT (2004) Prediction and functional analysis of native disorder in proteins from the three kingdoms of life. J Mol Biol 337(3):635–645. doi:10.1016/j.jmb.2004.02.002
10. Iakoucheva LM, Brown CJ, Lawson JD, Obradović Z, Dunker AK (2002) Intrinsic disorder in cell-signaling and cancer-associated proteins. J Mol Biol 323(3):573–584
11. Raychaudhuri S, Dey S, Bhattacharyya NP, Mukhopadhyay D (2009) The role of intrinsically unstructured proteins in neurodegenerative diseases. PLoS One 4(5):e5566. doi:10.1371/journal.pone.0005566
12. Uversky VN, Oldfield CJ, Dunker AK (2008) Intrinsically disordered proteins in human diseases: introducing the D2 concept. Annu Rev Biophys 37:215–246. doi:10.1146/annurev.biophys.37.032807.125924
13. Cheng Y, LeGall T, Oldfield CJ, Mueller JP, Van YY, Romero P, Cortese MS, Uversky VN, Dunker AK (2006) Rational drug design via intrinsically disordered protein. Trends Biotechnol 24(10):435–442. doi:10.1016/j.tibtech.2006.07.005
14. Eliezer D (2009) Biophysical characterization of intrinsically disordered proteins. Curr Opin Struct Biol 19(1):23–30. doi:10.1016/j.sbi.2008.12.004
15. Bernadó P, Svergun DI (2012) Structural analysis of intrinsically disordered proteins by small-angle X-ray scattering. Mol Biosyst 8(1):151–167. doi:10.1039/c1mb05275f
16. Kikhney AG, Svergun DI (2015) A practical guide to small angle X-ray scattering (SAXS) of flexible and intrinsically disordered proteins. FEBS Lett 589(19 Pt A):2570–2577. doi:10.1016/j.febslet.2015.08.027
17. Jensen MR, Ruigrok RW, Blackledge M (2013) Describing intrinsically disordered proteins at atomic resolution by NMR. Curr Opin Struct Biol 23(3):426–435. doi:10.1016/j.sbi.2013.02.007
18. Mittag T, Forman-Kay JD (2007) Atomic-level characterization of disordered protein ensembles. Curr Opin Struct Biol 17(1):3–14. doi:10.1016/j.sbi.2007.01.009
19. Receveur-Bréchot V, Bourhis JM, Uversky VN, Canard B, Longhi S (2006) Assessing protein disorder and induced folding. Proteins 62(1):24–45. doi:10.1002/prot.20750
20. Greenfield NJ (2006) Using circular dichroism spectra to estimate protein secondary structure. Nat Protoc 1(6):2876–2890. doi:10.1038/nprot.2006.202
21. Oldfield CJ, Dunker AK (2014) Intrinsically disordered proteins and intrinsically disordered protein regions. Annu Rev Biochem 83:553–584. doi:10.1146/annurev-biochem-072711-164947
22. Linding R, Russell RB, Neduva V, Gibson TJ (2003) GlobPlot: exploring protein sequences for globularity and disorder. Nucleic Acids Res 31(13):3701–3708
23. Dosztányi Z, Csizmok V, Tompa P, Simon I (2005) IUPred: web server for the prediction of intrinsically unstructured regions of proteins based on estimated energy content. Bioinformatics 21(16):3433–3434. doi:10.1093/bioinformatics/bti541
24. Prilusky J, Felder CE, Zeev-Ben-Mordehai T, Rydberg EH, Man O, Beckmann JS, Silman I, Sussman JL (2005) FoldIndex: a simple tool to predict whether a given protein sequence is intrinsically unfolded. Bioinformatics 21(16):3435–3438. doi:10.1093/bioinformatics/bti537
25. Schlessinger A, Punta M, Rost B (2007) Natively unstructured regions in proteins identified from contact predictions. Bioinformatics 23(18):2376–2384. doi:10.1093/bioinformatics/btm349
26. Zhang T, Faraggi E, Xue B, Dunker AK, Uversky VN, Zhou Y (2012) SPINE-D: accurate prediction of short and long disordered regions by a single neural-network based method. J Biomol Struct Dyn 29(4):799–813. doi:10.1080/073911012010525022
27. Ward JJ, McGuffin LJ, Bryson K, Buxton BF, Jones DT (2004) The DISOPRED server for the prediction of protein disorder. Bioinformatics 20(13):2138–2139. doi:10.1093/bioinformatics/bth195
28. Linding R, Jensen LJ, Diella F, Bork P, Gibson TJ, Russell RB (2003) Protein disorder prediction: implications for structural proteomics. Structure 11(11):1453–1459

29. Yang ZR, Thomson R, McNeil P, Esnouf RM (2005) RONN: the bio-basis function neural network technique applied to the detection of natively disordered regions in proteins. Bioinformatics 21(16):3369–3376. doi:10.1093/bioinformatics/bti534
30. Vullo A, Bortolami O, Pollastri G, Tosatto SC (2006) Spritz: a server for the prediction of intrinsically disordered regions in protein sequences using kernel machines. Nucleic Acids Res 34(Web Server issue):W164–W168. doi:10.1093/nar/gkl166
31. Romero P, Obradovic Z, Li X, Garner EC, Brown CJ, Dunker AK (2001) Sequence complexity of disordered protein. Proteins 42(1):38–48
32. Su CT, Chen CY, Hsu CM (2007) iPDA: integrated protein disorder analyzer. Nucleic Acids Res 35(Web Server issue):W465–W472. doi:10.1093/nar/gkm353
33. Hirose S, Shimizu K, Kanai S, Kuroda Y, Noguchi T (2007) POODLE-L: a two-level SVM prediction system for reliably predicting long disordered regions. Bioinformatics 23(16):2046–2053. doi:10.1093/bioinformatics/btm302
34. Yang JY, Yang MQ (2008) Predicting protein disorder by analyzing amino acid sequence. BMC Genomics 9(Suppl 2):S8. doi:10.1186/1471-2164-9-S2-S8
35. Schlessinger A, Liu J, Rost B (2007) Natively unstructured loops differ from other loops. PLoS Comput Biol 3(7):e140. doi:10.1371/journal.pcbi.0030140
36. Wang L, Sauer UH (2008) OnD-CRF: predicting order and disorder in proteins using [corrected] conditional random fields. Bioinformatics 24(11):1401–1402. doi:10.1093/bioinformatics/btn132
37. McGuffin LJ (2008) Intrinsic disorder prediction from the analysis of multiple protein fold recognition models. Bioinformatics 24(16):1798–1804. doi:10.1093/bioinformatics/btn326
38. McGuffin LJ, Atkins JD, Salehe BR, Shuid AN, Roche DB (2015) IntFOLD: an integrated server for modelling protein structures and functions from amino acid sequences. Nucleic Acids Res 43(W1):W169–W173. doi:10.1093/nar/gkv236
39. Ishida T, Kinoshita K (2007) PrDOS: prediction of disordered protein regions from amino acid sequence. Nucleic Acids Res 35(Web Server issue):W460–W464. doi:10.1093/nar/gkm363
40. Ishida T, Kinoshita K (2008) Prediction of disordered regions in proteins based on the meta approach. Bioinformatics 24(11):1344–1348. doi:10.1093/bioinformatics/btn195
41. Xue B, Dunbrack RL, Williams RW, Dunker AK, Uversky VN (2010) PONDR-FIT: a meta-predictor of intrinsically disordered amino acids. Biochim Biophys Acta 1804(4):996–1010. doi:10.1016/j.bbapap.2010.01.011
42. Schlessinger A, Punta M, Yachdav G, Kajan L, Rost B (2009) Improved disorder prediction by combination of orthogonal approaches. PLoS One 4(2):e4433. doi:10.1371/journal.pone.0004433
43. Mizianty MJ, Stach W, Chen K, Kedarisetti KD, Disfani FM, Kurgan L (2010) Improved sequence-based prediction of disordered regions with multilayer fusion of multiple information sources. Bioinformatics 26(18):i489–i496. doi:10.1093/bioinformatics/btq373
44. Deng X, Eickholt J, Cheng J (2009) PreDisorder: ab initio sequence-based prediction of protein disordered regions. BMC Bioinformatics 10:436. doi:10.1186/1471-2105-10-436
45. Vucetic S, Brown CJ, Dunker AK, Obradovic Z (2003) Flavors of protein disorder. Proteins 52(4):573–584. doi:10.1002/prot.10437
46. He B, Wang K, Liu Y, Xue B, Uversky VN, Dunker AK (2009) Predicting intrinsic disorder in proteins: an overview. Cell Res 19(8):929–949. doi:10.1038/cr.2009.87
47. Peng K, Radivojac P, Vucetic S, Dunker AK, Obradovic Z (2006) Length-dependent prediction of protein intrinsic disorder. BMC Bioinformatics 7:208. doi:10.1186/1471-2105-7-208
48. Radivojac P, Obradovic Z, Smith DK, Zhu G, Vucetic S, Brown CJ, Lawson JD, Dunker AK (2004) Protein flexibility and intrinsic disorder. Protein Sci 13(1):71–80. doi:10.1110/ps.03128904
49. Monastyrskyy B, Fidelis K, Moult J, Tramontano A, Kryshtafovych A (2011) Evaluation of disorder predictions in CASP9. Proteins 79(Suppl 10):107–118. doi:10.1002/prot.23161
50. Zhang T, Faraggi E, Li Z, Zhou Y (2013) Intrinsically semi-disordered state and its role in induced folding and protein aggregation. Cell Biochem Biophys 67(3):1193–1205. doi:10.1007/s12013-013-9638-0
51. Faraggi E, Xue B, Zhou Y (2009) Improving the prediction accuracy of residue solvent accessibility and real-value backbone torsion angles of proteins by guided-learning through a two-layer neural network. Proteins 74(4):847–856. doi:10.1002/prot.22193
52. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25(17):3389–3402

53. Faraggi E, Yang Y, Zhang S, Zhou Y (2009) Predicting continuous local structure and the effect of its substitution for secondary structure in fragment-free protein structure prediction. Structure 17(11):1515–1527. doi:10.1016/j.str.2009.09.006
54. Zhang T, Faraggi E, Zhou Y (2010) Fluctuations of backbone torsion angles obtained from NMR-determined structures and their prediction. Proteins 78(16):3353–3362. doi:10.1002/prot.22842
55. Faraggi E, Zhang T, Yang Y, Kurgan L, Zhou Y (2012) SPINE X: improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles. J Comput Chem 33(3):259–267. doi:10.1002/jcc.21968
56. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE (2000) The protein data bank. Nucleic Acids Res 28(1):235–242
57. Sickmeier M, Hamilton JA, LeGall T, Vacic V, Cortese MS, Tantos A, Szabo B, Tompa P, Chen J, Uversky VN, Obradovic Z, Dunker AK (2007) DisProt: the database of disordered proteins. Nucleic Acids Res 35(Database issue):D786–D793. doi:10.1093/nar/gkl893
58. Sirota FL, Ooi HS, Gattermayer T, Schneider G, Eisenhaber F, Maurer-Stroh S (2010) Parameterization of disorder predictors for large-scale applications requiring high specificity by using an extended benchmark dataset. BMC Genomics 11(Suppl 1):S15. doi:10.1186/1471-2164-11-S1-S15
59. Vousden KH, Lane DP (2007) p53 in health and disease. Nat Rev Mol Cell Biol 8(4):275–283. doi:10.1038/nrm2147
60. Uversky VN, Oldfield CJ, Midic U, Xie H, Xue B, Vucetic S, Iakoucheva LM, Obradovic Z, Dunker AK (2009) Unfoldomics of human diseases: linking protein intrinsic disorder with diseases. BMC Genomics 10(Suppl 1):S7. doi:10.1186/1471-2164-10-S1-S7
61. Borcherds W, Theillet FX, Katzer A, Finzel A, Mishall KM, Powell AT, Wu H, Manieri W, Dieterich C, Selenko P, Loewer A, Daughdrill GW (2014) Disorder and residual helicity alter p53-Mdm2 binding affinity and signaling in cells. Nat Chem Biol 10(12):1000–1002. doi:10.1038/nchembio.1668
62. Kriwacki RW (2014) Protein dynamics: tuning disorder propensity in p53. Nat Chem Biol 10(12):987–988. doi:10.1038/nchembio.1692

Chapter 13

Predicting Real-Valued Protein Residue Fluctuation Using FlexPred

Lenna Peterson, Michal Jamroz, Andrzej Kolinski, and Daisuke Kihara

Abstract

The conventional view of a protein structure as static provides only a limited picture. There is increasing evidence that protein dynamics are often vital to protein function including interaction with partners such as other proteins, nucleic acids, and small molecules. Considering flexibility is also important in applications such as computational protein docking and protein design. While residue flexibility is partially indicated by experimental measures such as the B-factor from X-ray crystallography and ensemble fluctuation from nuclear magnetic resonance (NMR) spectroscopy as well as computational molecular dynamics (MD) simulation, these techniques are resource-intensive. In this chapter, we describe the web server and stand-alone version of FlexPred, which rapidly predicts absolute per-residue fluctuation from a three-dimensional protein structure. On a set of 592 nonredundant structures, comparing the fluctuations predicted by FlexPred to the observed fluctuations in MD simulations showed an average correlation coefficient of 0.669 and an average root mean square error of 1.07 Å. FlexPred is available at http://kiharalab.org/flexPred/.

Key words Bioinformatics, Computational biology, Support vector machine, Support vector regression, Protein residue fluctuation, Protein flexibility, Protein conformational flexibility, Protein structure, Protein design, Molecular dynamics

1 Introduction

The function of many proteins is determined not only by their rigid three-dimensional (3D) structure but also by the flexibility of protein chains [1]. Protein flexibility can influence function by, for example, determining catalytic rates [2] and affecting ligand and protein interactions [3]. Knowledge of flexibility is also important for accurate protein design [4, 5] and computational protein/ligand docking [3, 6].

Despite the importance of chain flexibility, it is difficult to glean a full picture of a protein's flexibility via experimental methods. Information about atomic fluctuations is reflected in the B-factor in X-ray crystallography [7]; however, the fluctuation is only one component of this uncertainty convolved with other

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_13,

factors that cause errors in model building. In particular, B-factor tends to underestimate the motion of flexible regions [8]. Nuclear magnetic resonance (NMR) spectroscopy currently provides the most direct experimental evidence of flexibility; nevertheless, the accuracy depends on the experimental setup and the mathematical model used [9–12]. Cryogenic electron microscopy (cryo-EM) can detect heterogeneous conformational states [13] but not small conformational flexibility. Above all, experimental methods are costly in terms of time and resources and thus are not always applicable.

In order to augment experimental methods for determining protein flexibility, many computational approaches have been developed to model protein dynamics. Molecular dynamics (MD) simulations model the motion of all the atoms in a protein on the picosecond to microsecond timescale [14], from which per-residue fluctuation can be extracted. One approach to achieve faster results compared to MD is coarse-grained simulations [15–19]. Using trajectories from MD or coarse-grained simulations, normal mode analysis [20] can depict an overview of the motion that is easy to grasp. Alternatively, Gaussian network model (GNM) [21–24] uses a simplified model of protein structures to simulate protein motion. Many works have used GNM or related approaches to predict B-factor [21, 24–29]. Another work uses a mean-field model to predict fluctuations [30]. Although these physics-based computational methods provide a physical view of atom- or residue-level protein fluctuation, they are generally targeted toward computational biophysicists and may not be easy to use by experimental biologists. In addition to structure-based computational methods, there are methods that use sequence features to predict B-factor [31–33], relative motion [34], and two-state [35] or three-state [36] flexibility. These methods are applicable to a larger number of proteins since no structure is required, but with an inevitable decline in accuracy.

In this chapter, we describe the web server and stand-alone version of FlexPred, which rapidly predicts the absolute fluctuation of each residue in a protein structure. FlexPred is designed to predict the fluctuations exhibited by a protein during a 10 ns MD simulation (fast, comparatively small motions). Full details of the FlexPred method have been previously published [37]. We provide detailed instructions for using FlexPred and present example predictions for X-ray crystal structures of a monomer protein and a protein complex as well as a monomer structure determined using NMR. Both the web server and the stand-alone version can be found at http://kiharalab.org/flexPred.

2 Algorithm

FlexPred predicts the residue fluctuation observed during a 10 ns molecular dynamics (MD) simulation. MD fluctuation, the theoretical range of motion of the atoms in a protein structure, was computed as the root mean square distance between the Cα atom in the MD simulation and the Cα atom in the reference PDB file averaged across all time steps of the MD simulation [37]. The values produced by FlexPred can be used to estimate whether a specific portion of a protein chain is flexible and how flexible that region is.

FlexPred uses static features of a protein structure to predict MD residue fluctuation. The features tested were B-factor [7], residue distance from the protein center of mass [38, 39], residue contact number [40, 41], hydrophobic/hydrophilic [42] residue contact number, residue solvent accessible surface area [43, 44], residue depth [45], residue lower/upper half-sphere exposure [46], and secondary structure [43]. Details of each feature have been previously described [37]. FlexPred combines these features using the framework of support vector regression (SVR) implemented by LIBSVM [47]. It was trained on a non-redundant set of 592 molecular dynamics (MD) simulations from the Molecular Dynamics Extended Library (MoDEL) [48]. Almost all (96.11%) of the simulations were 10 ns in length and the rest were shorter. This is in the timescale of "fast" motions [14]; thus, FlexPred is not appropriate for predicting large movements such as domain-domain motion.

FlexPred predictions were evaluated using Pearson's correlation coefficient and the root mean square error (RMS) of the difference between the predicted fluctuation and the MD fluctuation. The highest single static feature correlation to real fluctuation was for residue contact number with a cutoff of 15 or 16 Å. This term had higher correlation to MD fluctuation than did B-factor. GNM prediction was also tested as a feature. While GNM alone had a higher correlation to MD fluctuation than did any static feature, including GNM in the feature set led to a consistent decrease in correlation coefficient [37]. Multiple combinations of features were tested and the highest correlation of 0.669 was observed with B-factor combined with residue contact number with cutoffs of 6, 8, 12, 16, 18, 20, and 22 Å (Feature set 15) [37]. The RMS of this feature combination was 1.07 Å [37]. This feature set effectively encodes information about 3D structures in a lower dimensional feature space.

3 Web Server

The web server takes a 3D protein structure in PDB format as input and predicts a fluctuation value for each residue of the protein. Figure 1 shows a screenshot of the web server input page.

3.1 Web Server Input

1. Go to the web server homepage: http://kiharalab.org/flexPred/.
2. Choose a PDB structure (*see* **Notes 1** and **2**):
   (a) To upload your own PDB file, click "Browse…" (Fig. 1 (1)).
   (b) To use a published PDB structure, enter the 4-character PDB code (e.g. 1bfg) into the PDB code box. (Fig. 1 (2)).
3. Choose the feature set (Fig. 1 (3); *see* **Note 3**):
   (a) "With B-factor": use this for X-ray structures.
   (b) "Without B-factor": use this for NMR structures and computational models.
4. Click "Predict Fluctuation" (Fig. 1 (4)).
5. The server will generally complete in 2–20 s.

3.2 Web Server Output

Figure 2 shows a screenshot of an example web server output page. The top of the page shows an image of the structure with high fluctuation residues colored in red and low fluctuation residues colored in blue (Fig. 2 (1)). The middle of the page shows a graph of the predicted fluctuation for each residue number (Fig. 2 (2)). Below the graph are links to download a comma separated value (CSV) file

Fig. 1 The FlexPred web server query submission page. (1) Upload a PDB file or (2) choose a published PDB structure by ID. (3) Choose the feature set with B-factor for X-ray structures or without B-factor for NMR structures. (4) Finally, click "Predict Fluctuation." (5) The software may also be downloaded

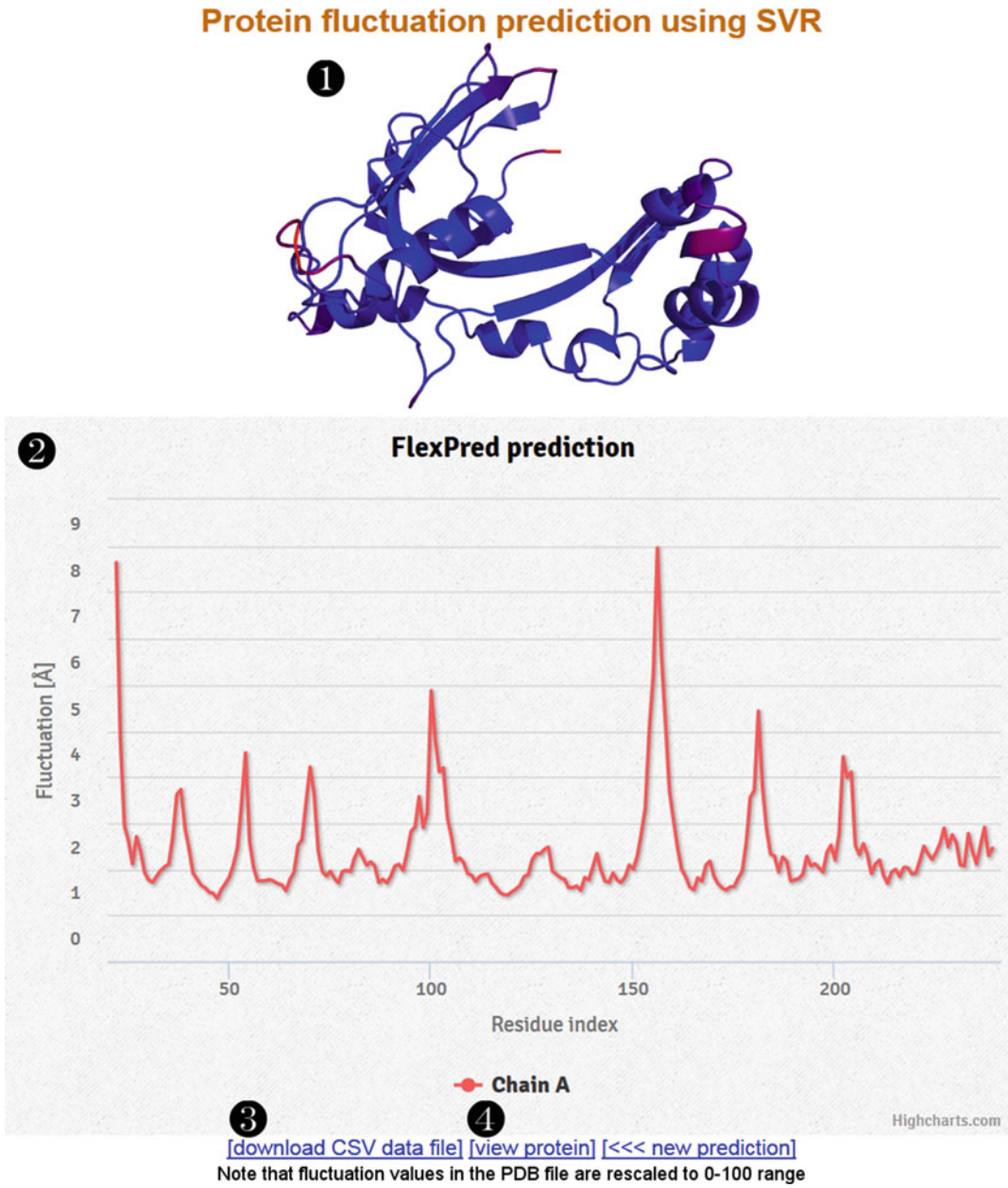


Fig. 2 The FlexPred output page. (1) The structure with high fluctuation shown in *red*. (2) The fluctuation of each residue. (3) Download link for the raw data file. (4) Download link for the PDB file with predicted fluctuation in the B-factor column

containing the predicted fluctuations for each residue (Fig. 2 (3)) and a PDB file with normalized predicted fluctuations in the B-factor column (Fig. 2 (4)). The CSV file can be conveniently viewed using spreadsheet software.

4 Stand-Alone Software

The stand-alone software requires Python and has only been tested on Linux. It takes a 3D protein structure in PDB format as input and predicts a fluctuation value for each residue of the protein.

4.1 Stand-Alone Input

1. Go to the web server homepage (http://kiharalab.org/flexPred/).
2. Under the heading "Download software," "click" "download file" (Fig. 1 (5)).
3. Expand the file using the command "tar -xvf flexPred.tar.xz."
4. Download libsvm from http://www.csie.ntu.edu.tw/~cjlin/libsvm/ and compile.
5. Edit "predictFluct.py" to set paths to libsvm and FlexPred.
6. The script has two positional arguments:
   (a) The first argument is the path to the PDB file (see Notes 1 and 2).
   (b) The second argument is the feature set. "NMR" uses the feature set without B-factor and "XRAY" uses the feature set with B-factor (*see* **Note 3**).
7. Try the example protein: "cd example; python ../predictFluct.py 1BFG.pdb XRAY" (*see* **Note 4**).
8. To run FlexPred on other proteins: "python predictFluct.py model_file.pdb [NMR|XRAY]."

4.2 Stand-Alone Output

The stand-alone version of FlexPred should complete within seconds. The stand-alone software produces a text file containing the predicted fluctuations for each residue.

5 Case Studies

We present example FlexPred predictions using three types of proteins: a monomer X-ray structure, a dimer X-ray structure, and a monomer NMR structure. The predictions for the X-ray structures use the "With B-factor" feature set while the prediction for the NMR structure uses the "Without B-factor" feature set. Figure 3 and Table 1 show that FlexPred predictions have moderate to high correlation to real fluctuation from MD simulations. On the test set in the previous paper, the average correlation coefficient was 0.669 and the average RMS was 1.07 Å [37]. The MD fluctuation is the RMS of the difference between each snapshot in the trajectory and the mean position of the trajectory.

The first example is the ssDNA binding protein gp32 (PDB ID 1gpc) (Fig. 3a, b). The highest core fluctuation is in a loop around

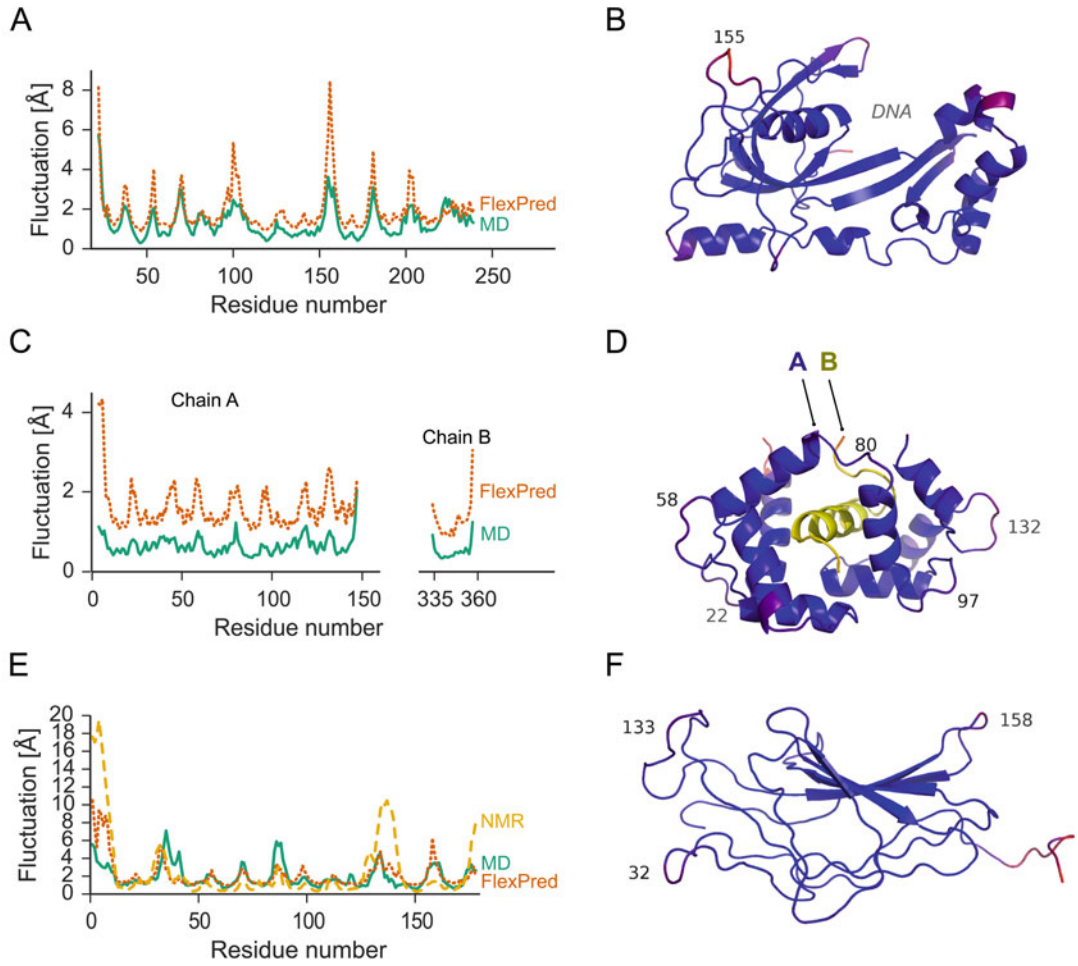


Fig. 3 Examples of FlexPred predictions. (**a**, **c**, **e**) fluctuation of each residue. *Green solid line*: MD fluctuation. *Orange dotted line*: Predicted fluctuation by FlexPred. *Yellow dashed line*: NMR fluctuation. (**b**, **d**, **f**) structures with high fluctuation shown in *red* and notable high fluctuation residues indicated with numbers. Proteins used: (**a**, **b**) single-stranded DNA binding protein gp32 from bacteriophage T4, residues 22–239 (PDB ID: 1gpc). (**c**, **d**) Calmodulin (chain A) in complex with a fragment of Ca(2+)/calmodulin-dependent kinase kinase (chain B) (PDB ID: 1iq5). Calmodulin, Chain A, is residues 4–147 and shown in *blue*. CaMKK, chain B, is residues 334–357 and shown in *yellow*. (**e**, **f**) human transcription factor NFATc, DNA-binding domain, residues 3–178 (PDB ID: 1nfa)

residue 155. ssDNA fits into a cleft sized to exclude dsDNA [49], where minimal flexibility was observed. High flexibility could allow dsDNA to bind. Overall, the predicted fluctuation agrees well with those in MD simulation. The prediction had a correlation coefficient of 0.839 and RMS of 0.83 Å to the MD simulation, better than the average correlation and RMS observed in the benchmark in the original paper [37] (Table 1).

The second example is the heterodimer of calmodulin (CaM) in complex with the CaM binding domain of CaM-dependent

Table 1
FlexPred prediction on two example X-ray protein structures

Structure	PCC	RMS (Å)
1gpcA	0.839	0.83
1iq5A	0.620	1.03
1iq5B	0.933	0.81
Prev. avg.	0.669	1.04

Pearson's correlation coefficient (PCC) (perfect correlation is 1, no correlation is 0, and perfect negative correlation is −1) and root mean square deviation (no deviation is 0) between FlexPred and MD residue fluctuations. Average values are from the X-ray dataset in the original paper [37]

kinase kinase (CaMKK) (PDB ID 1iq5) (Fig. 3c, d). The prediction overestimated fluctuations in comparison with the MD simulation, although the average correlation and RMS of the two chains were better than the average (Table 1). The high predicted flexibility of the calcium binding loops (at residues 22, 58, 97, and 132) may be because the features only consider the position of protein atoms, not the bound calcium ion. In the MD fluctuation, the linker between the two domains (residue 80) shows the highest core fluctuation. It was observed that binding of the CaMKK peptide causes a shift in the relative angle of the two domains [50]. However, while FlexPred does predict flexibility at the domain linker (residue 80), it is of similar magnitude to multiple other loops. The lower magnitude of the predicted linker flexibility could be due to the timescale of the domain-domain motion, which is likely longer than the 10 ns timescale of the training data. The accuracy for this complex is comparable to the accuracy for monomers (Table 1), indicating that while FlexPred was not trained using multimers, it is applicable to predict the flexibility of a protein-protein complex.

The third example is the NMR structure of the DNA binding domain of the transcription factor NFATc (PDB ID 1nfa) (Fig. 3e, f). Because NMR structures lack B-factor, the prediction used the "Without B-factor" setting. The predictions are slightly worse than the average computed on the previous test of NMR structures [37]. For this example, we also computed NMR ensemble fluctuation by computing the RMS of each model to the first model (Table 2). It is interesting that FlexPred shows higher correlation to NMR (0.85) than to MD (0.58). In the NMR ensemble, the highest fluctuation in the core region (residues 13–172 [51]) is in a loop around residue 133, where both MD and FlexPred match the location of the increased fluctuation but with much lower magnitude.

Table 2
FlexPred prediction on an example NMR protein structure

Pair	Structure	PCC	RMS (Å)
FlexPred-MD	1nfaA	0.584	1.37
	Prev. avg.	0.686	2.16
FlexPred-NMR	1nfaA	0.846	2.52
	Prev. avg.	0.739	1.81
MD-NMR	1nfaA	0.471	3.37
	Prev. avg.	0.651	2.42

Pearson's correlation coefficient (PCC) (perfect correlation is 1, no correlation is 0, and perfect negative correlation is −1) and root mean square deviation (no deviation is 0) of fluctuations between FlexPred and MD, between FlexPred and NMR, and between MD and NMR. Average values are from the NMR dataset in the original paper [37]

For NMR, the second highest core fluctuation is in the DNA binding loop around residue 32 [51], where FlexPred matches NMR almost perfectly while the peak from MD simulation is shifted. Both FlexPred and MD show much higher fluctuation than NMR for the loop around residue 158.

6 Conclusions

We outline the web server and stand-alone software for FlexPred, which predicts a real-valued absolute fluctuation for each residue of an input 3D protein structure. The web server is easy to use and quickly provides accurate prediction with intuitive visualization. It is useful for analyzing function of a protein from its structure and for artificial design of proteins.

7 Notes

1. If the PDB file contains multiple models (generally only in NMR structures and marked by lines such as "MODEL 1" and "MODEL 2"), only the first model will be considered.
2. If the PDB file contains multiple chains (indicated with different chain IDs, e.g., A, B), all chains will be used to compute contact maps and a separate prediction will be made for each chain. If the protein is a biological monomer but the PDB file contains crystal contacts, additional chains should be removed from the file before prediction for the most accurate results. Protein oligomeric state is often annotated in the "REMARK 350" section of a PDB file and can be predicted using the PISA server [52].

3. "With B-factor" uses feature set 15 from the original paper [37] (mean correlation coefficient 0.669, mean RMS 1.07 Å) and "Without B-factor" uses feature set 16 (mean correlation coefficient 0.660, mean RMS 1.09 Å). If "With B-factor" is selected but the B-factors in the file are all zero (0.0), "Without B-factor" will be substituted automatically.
4. The stand-alone version requires write access to the directory where the input PDB file is located.

Acknowledgements

This work was partly supported by the National Institute of General Medical Sciences of the National Institutes of Health (R01GM097528) and the National Science Foundation (IIS1319551, DBI1262189, IOS1127027).

References

1. Teilum K, Olsen JG, Kragelund BB (2009) Functional aspects of protein flexibility. Cell Mol Life Sci 66:2231–2247
2. Hammes GG, Benkovic SJ, Hammes-Schiffer S (2011) Flexibility, diversity, and cooperativity: pillars of enzyme catalysis. Biochemistry 50:10422–10430
3. Zacharias M (2010) Accounting for conformational changes during protein–protein docking. Curr Opin Struct Biol 20:180–186
4. Mandell DJ, Kortemme T (2009) Backbone flexibility in computational protein design. Curr Opin Biotechnol 20:420–428
5. Lassila JK (2010) Conformational diversity and computational enzyme design. Curr Opin Chem Biol 14:676–682
6. Lill MA (2011) Efficient incorporation of protein flexibility and dynamics into molecular docking simulations. Biochemistry 50:6157–6169
7. Debye P (1913) Interferenz von Röntgenstrahlen und Wärmebewegung. Ann Phys 348:49–92
8. Eastman P, Pellegrini M, Doniach S (1999) Protein flexibility in solution and in crystals. J Chem Phys 110:10141
9. Ishima R, Torchia DA (2000) Protein dynamics from NMR. Nat Struct Biol 7:740–743
10. Baldwin AJ, Kay LE (2009) NMR spectroscopy brings invisible protein states into focus. Nat Chem Biol 5:808–814
11. Nilges M, Habeck M, O'Donoghue SI, Rieping W (2006) Error distribution derived NOE distance restraints. Proteins 64:652–664
12. Chalaoux F-R, O'Donoghue SI, Nilges M (1999) Molecular dynamics and accuracy of NMR structures: effects of error bounds and data removal. Proteins 34:453–463
13. Wang Q, Matsui T, Domitrovic T, Zheng Y, Doerschuk PC, Johnson JE (2013) Dynamics in cryo EM reconstructions visualized with maximum-likelihood derived variance maps. J Struct Biol 181:195–206
14. Klepeis JL, Lindorff-Larsen K, Dror RO, Shaw DE (2009) Long-timescale molecular dynamics simulations of protein structure and function. Curr Opin Struct Biol 19:120–127
15. Liwo A, Oldziej S, Pincus MR, Wawak RJ, Rackovsky S, Scheraga HA (1997) A united-residue force field for off-lattice protein-structure simulations. I. Functional forms and parameters of long-range side-chain interaction potentials from protein crystal data. J Comput Chem 18:849–873
16. Kolinski A (2004) Protein modeling and structure prediction with a reduced representation. Acta Biochim Pol 51:349–371
17. Jamroz M, Kolinski A, Kmiecik S (2013) CABS-flex: server for fast simulation of protein structure fluctuations. Nucleic Acids Res 41:W427–W431
18. Jamroz M, Orozco M, Kolinski A, Kmiecik S (2013) Consistent view of protein fluctuations from all-atom molecular dynamics and coarse-grained dynamics with knowledge-based force-field. J Chem Theory Comput 9:119–125

19. Jamroz M, Kolinski A, Kmiecik S (2014) CABS-flex predictions of protein flexibility compared with NMR ensembles. Bioinformatics 30:2150–2154
20. Brooks B, Karplus M (1983) Harmonic dynamics of proteins: normal modes and fluctuations in bovine pancreatic trypsin inhibitor. Proc Natl Acad Sci U S A 80:6571–6575
21. Haliloglu T, Bahar I, Erman B (1997) Gaussian dynamics of folded proteins. Phys Rev Lett 79:3090–3093
22. Tirion MM (1996) Large amplitude elastic motions in proteins from a single-parameter, atomic analysis. Phys Rev Lett 77:1905–1908
23. Bahar I, Erman B, Haliloglu T, Jernigan RL (1997) Efficient characterization of collective motions and interresidue correlations in proteins by low-resolution simulations. Biochemistry 36:13512–13523
24. Yang L, Song G, Jernigan RL (2009) Protein elastic network models and the ranges of cooperativity. Proc Natl Acad Sci U S A 106:12347–12352
25. Kondrashov DA, Cui Q, Phillips GN Jr (2006) Optimization and evaluation of a coarse-grained model of protein motion using X-ray crystal data. Biophys J 91:2760–2767
26. Lin T-L, Song G (2010) Generalized spring tensor models for protein fluctuation dynamics and conformation changes. BMC Struct Biol 10:S3
27. Micheletti C, Carloni P, Maritan A (2004) Accurate and efficient description of protein vibrational dynamics: comparing molecular dynamics and Gaussian models. Proteins 55:635–645
28. Canino LS, Shen T, McCammon JA (2002) Changes in flexibility upon binding: application of the self-consistent pair contact probability method to protein-protein interactions. J Chem Phys 117:9927
29. Opron K, Xia K, Wei G-W (2015) Communication: capturing protein multiscale thermal fluctuations. J Chem Phys 142:211101
30. Pandey BP, Zhang C, Yuan XZ, Zi J, Zhou YQ (2005) Protein flexibility prediction by an all-atom mean-field statistical theory. Protein Sci 14:1772–1777
31. Zhang H, Kurgan L (2014) Improved prediction of residue flexibility by embedding optimized amino acid grouping into RSA-based linear models. Amino Acids 46:2665–2680
32. Chen P, Wang B, Wong H-S, Huang D-S (2007) Prediction of protein B-factors using multi-class bounded SVM. Protein Pept Lett 14:185–190
33. Schlessinger A, Yachdav G, Rost B (2006) PROFbval: predict flexible and rigid residues in proteins. Bioinformatics 22:891–893
34. Hirose S, Yokota K, Kuroda Y, Wako H, Endo S, Kanai S, Noguchi T (2010) Prediction of protein motions from amino acid sequence and its application to protein-protein interaction. BMC Struct Biol 10:20
35. Gu J, Gribskov M, Bourne PE (2006) Wiggle—predicting functionally flexible regions from primary sequence. PLoS Comput Biol 2:e90
36. de Brevern AG, Bornot A, Craveur P, Etchebest C, Gelly J-C (2012) PredyFlexy: flexibility and local structure prediction from sequence. Nucleic Acids Res 40:W317–W322
37. Jamroz M, Kolinski A, Kihara D (2012) Structural features that predict real-value fluctuations of globular proteins. Proteins 80:1425–1435
38. Shih C-H, Huang S-W, Yen S-C, Lai Y-L, Yu S-H, Hwang J-K (2007) A simple way to compute protein dynamics without a mechanical model. Proteins 68:34–38
39. Kloczkowski A, Jernigan RL, Wu Z, Song G, Yang L, Kolinski A, Pokarowski P (2009) Distance matrix-based approach to protein structure prediction. J Struct Funct Genomics 10:67–81
40. Lin C-P, Huang S-W, Lai Y-L, Yen S-C, Shih C-H, Lu C-H, Huang C-C, Hwang J-K (2008) Deriving protein dynamical properties from weighted protein contact number. Proteins 72:929–935
41. Halle B (2002) Flexibility and packing in proteins. Proc Natl Acad Sci U S A 99:1274–1279
42. Kyte J, Doolittle RF (1982) A simple method for displaying the hydropathic character of a protein. J Mol Biol 157:105–132
43. Kabsch W, Sander C (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers 22:2577–2637
44. Miller S, Janin J, Lesk AM, Chothia C (1987) Interior and surface of monomeric proteins. J Mol Biol 196:641–656
45. Chakravarty S, Varadarajan R (1999) Residue depth: a novel parameter for the analysis of protein structure and stability. Structure 7:723–732
46. Hamelryck T (2005) An amino acid has two sides: a new 2D measure provides a different view of solvent exposure. Proteins 59:38–48
47. Chang C-C, Lin C-J (2011) LIBSVM: a library for support vector machines. ACM Trans Intell Syst Technol 2:27:1–27:27

48. Meyer T, D'Abramo M, Hospital A et al (2010) MoDEL (molecular dynamics extended library): a database of atomistic molecular dynamics trajectories. Structure 18:1399–1409
49. Shamoo Y, Friedman AM, Parsons MR, Konigsberg WH, Steitz TA (1995) Crystal structure of a replication fork single-stranded DNA binding protein (T4 gp32) complexed to DNA. Nature 376:362–366
50. Kurokawa H, Osawa M, Kurihara H, Katayama N, Tokumitsu H, Swindells MB, Kainosho M, Ikura M (2001) Target-induced conformational adaptation of calmodulin revealed by the crystal structure of a complex with nematode Ca2+/calmodulin-dependent kinase kinase peptide. J Mol Biol 312:59–68
51. Wolfe SA, Zhou P, Dötsch V, Chen L, You A, Ho SN, Crabtree GR, Wagner G, Verdine GL (1997) Unusual Rel-like architecture in the DNA-binding domain of the transcription factor NFATc. Nature 385:172–176
52. Krissinel E, Henrick K (2007) Inference of macromolecular assemblies from crystalline state. J Mol Biol 372:774–797

Chapter 14

Prediction of Disordered RNA, DNA, and Protein Binding Regions Using DisoRDPbind

Zhenling Peng, Chen Wang, Vladimir N. Uversky, and Lukasz Kurgan

Abstract

Intrinsically disordered proteins and regions (IDPs and IDRs) are involved in a wide range of cellular functions and they often facilitate interactions with RNAs, DNAs, and proteins. Although many computational methods can predict IDPs and IDRs in protein sequences, only a few methods predict their functions and these functions primarily concern protein binding. We describe how to use the first computational method DisoRDPbind for high-throughput prediction of multiple functions of disordered regions. Our method predicts the RNA-, DNA-, and protein-binding residues located in IDRs in the input protein sequences. DisoRDPbind provides accurate predictions and is sufficiently fast to make predictions for full genomes. Our method is implemented as a user-friendly webserver that is freely available at http://biomine.ece.ualberta.ca/DisoRDPbind/. We overview our predictor, discuss how to run the webserver, and show how to interpret the corresponding results. We also demonstrate the utility of our method based on two case studies, human BRCA1 protein that binds various proteins and DNA, and yeast 60S ribosomal protein L4 that interacts with proteins and RNA.

Key words Intrinsic disorder, Prediction, Protein–protein interactions, Protein–DNA interactions, Protein–RNA interactions, DisoRDPbind

1 Introduction

Intrinsically disordered proteins and regions (IDPs and IDRs) lack a stable three-dimensional structure under physiological conditions in vitro and form an ensemble of structural conformations [1–3]. They participate in a wide range of cellular functions and are common in nature, particularly in eukaryotic species [3–6]. Many computational methods are available for the prediction of intrinsic disorder from protein sequences [7–13]. These predictors were used to estimate the amount of disorder in various species and domains of life and to characterize cellular functions of disorder [5, 14–20]. IDPs and IDRs were shown to be significantly involved in the protein–protein, protein–DNA, and protein–RNA interactions [5, 18, 20–27]; for convenience, here we

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_14,

utilize the terms disordered RNA, DNA, and protein-binding to denote the RNA-, DNA-, and protein-binding located in IDRs. Prediction of residues that bind proteins, RNAs, and DNAs has attracted strong research interest in the last decade [28–33]. However, these predictions address interactions annotated from crystal structures, which means that they are primarily focused on the structured (ordered) regions.

A number of studies that predict functions of IDPs and IDRs were also recently discussed [34]. A prediction of over one hundred Gene Ontology (GO) annotations associated with IDPs and IDRs was carried out by Khan *et al.* [35]. Moreover, several methods were developed for the prediction of disordered protein binding regions including alpha-MoRF-Pred [36], ANCHOR [37], MoRFpred [38], PepBindPred [39], MFSPSSMpred [40], DISOPRED3 [41], MoRFCHiBi [42], and fMoRFpred [43]. This implies that functions of IDRs and IDPs are predictable from protein sequences. Availability of hundreds of regions annotated as disordered RNA, DNA, and protein binding in the DisProt database [44] and the lack of methods that predict disordered RNA and DNA binding motivated the development of a new predictor DisoRDPbind [45]. This is the first method that predicts multiple functions mediated by IDPs and IDRs. DisoRDPbind obtains favorable predictive performance for these three types of disordered binding regions. It is also very fast and can be applied to predict full genomes in a matter of hours using its convenient webserver (the largest human genome can be predicted in about 2 days) [45]. The DisoRDPbind's webserver outputs three propensity scores for each input residue that quantify the likelihood for this residue to be involved in the disordered RNA, DNA, and protein binding. We overview architecture of our method and provide details on how to use the webserver and how to interpret the results. Finally, we use two case studies that involve analysis of RNA-, DNA-, and protein-binding proteins to illustrate how our method can be used to suggest localization of disordered RNA-, DNA-, and protein-binding regions in protein sequences.

2 Materials and Method

2.1 Datasets

We extracted 315, 114, and 36 proteins from the DisProt database [44] to develop three datasets: TRAINING, TEST114, and TEST36, respectively. Each dataset includes disordered regions that were annotated to bind RNAs, DNAs, and proteins. The TRAINING dataset was used for empirical design of DisoRDPbind while the other two datasets were used to assess its predictive performance and compare it against other methods. Proteins in TEST114 were collected to share <30 % sequence similarity with proteins in TRAINING to allow for assessment of

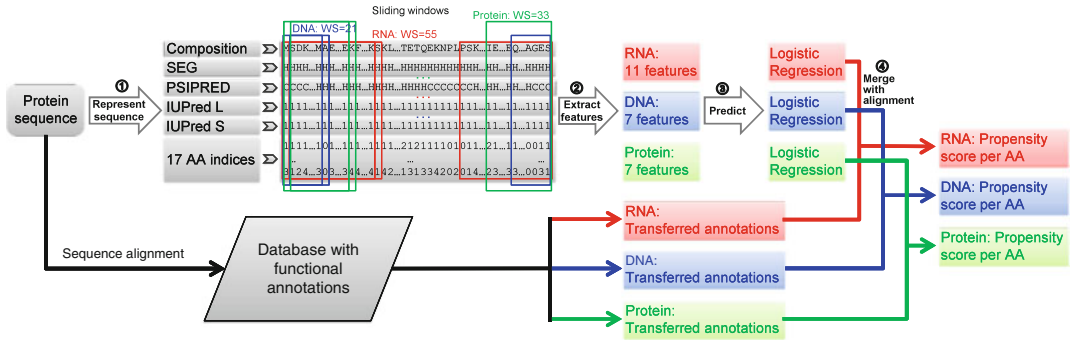


Fig. 1 The architecture of the DisoRDPbind method. The four layers are denoted by the corresponding numbers shown inside circles. We use term "composition" to denote the amino acid composition. The SEG algorithm is used to generate the sequence complexity and PSIPRED and IUPred L(S) are utilized to predict the profiles of secondary structure and intrinsic disorder, respectively. The "17 AA indices" denote the physiochemical properties of amino acids (AAs) including their hydrophobicity, net charge, and free energy

predictive performance on dissimilar proteins. The second test dataset, TEST36, includes new depositions to DisProt as compared with the proteins from TRAINING. These three datasets are available at http://biomine.ece.ualberta.ca/DisoRDPbind/. Reference [45] provides further details.

2.2 Architecture

Recently, we developed the DisoRDPbind method to predict the disordered RNA-, DNA-, and protein-binding residues in the input protein sequences. Our method is based on a runtime efficient four-layer design; *see* Fig. 1. First, we represent the input protein using several structural and functional properties. Second, these properties are used to represent each residue in the input protein chain using a vector of numeric descriptors/features. Third, these features are inputted into a predictive model. Fourth, the outputs of the predictive model are merged with an alignment-based prediction to derive the final result. Following we provide a more detailed explanation.

In layer 1, we represent each input protein sequence based on its amino acid (AA) composition, its sequence complexity generated by the SEG algorithm [46], intrinsic disorder predicted by IUPred (including IUPred L and IUPred S) [47], secondary structure predicted by PSIPRED [48], and 17 physiochemical properties of amino acids (AAs) including hydrophobicity, net charge, and free energy. In layer 2, we use this information to compute a vector of features for each residue in the input protein chain and each predicted function. We utilized sliding window with different window size (WS) to obtain the numerical features for different binding events where WS = 55, 21, and 33 for disordered RNA, DNA, and protein binding, respectively. We quantified the abovementioned putative sequence structural and functional characteristics, such as disorder, secondary structure, hydrophobicity, etc. in a window

centered over the predicted residue by computing their averages and content values. These values represent local (in the sequence) bias that contributes towards the prediction of the residue in the middle of the window. Since the many features considered in this layer are redundant and/or irrelevant to the predicted functions, we performed an empirical feature selection for each function using the TRAINING dataset. Consequently, we selected a small set of 11, 7, and 7 features for the prediction of disordered RNA, DNA, and protein binding, respectively. In layer 3, for each residue in the input protein sequence we pass a given selected set of features into a logistic regression model for the corresponding binding event. This means that three regression-based models are used to find the putative disordered RNA-, DNA-, and protein-binding residues. We picked this type of model based on its popularity, short runtime and the ability to output a real-valued propensity. In the last layer 4, we merge the regression-based predictions with functional annotations found through sequence similarity to generate the final predictions. We utilize BLAST [49] to align the input sequence against a database of functionally annotated proteins (TRAINING dataset) and then we transfer the functional annotations for each input residue that was aligned to a functional residue in a sufficiently similar annotated protein. The final predictions include three propensity scores for each residue in the input sequence that quantify its likelihood to be disordered RNA-, DNA-, and protein-binding residues, respectively; higher values of propensity correspond to a higher likelihood of binding. DisoRDPbind also provides a binary prediction for each function by using a threshold on a given putative propensity score; residues with propensities higher than the threshold are predicted as binding and the other residues are predicted as non-binding (*see* Subheading 2.6).

2.3 Predictive Quality and Runtime

The predictive quality of our method was assessed in the original manuscript [45]. DisoRDPbind was shown to secure the area under the ROC curve (AUC) values between 0.62 and 0.72, depending on the benchmark dataset used (TEST114 and TEST36) and the disorder function that was assessed. The TP-rate (fraction of correctly predicted binding residues) of DisoRDPbind computed at the FP-rate (fraction of incorrectly predicted non-binding residues) of 0.1 is 0.27, 0.25, and 0.24 for the prediction of the disordered DNA-, protein-, and RNA-binding residues, respectively. These are reasonable levels of TP-rate and AUC values, which were shown to be higher than the corresponding values of the closest alternatives (predictors of disordered protein-binding residues and predictors of ordered DNA- and RNA-binding residues) [45]. Interestingly, predictions from DisoRDPbind complement predictions from the predictors of structured DNA- and RNA-binding residues (they are characterized by low correlation <0.3), while as expected they are similar to the outputs of methods

that predict disordered protein-binding residues (correlation >0.5 with ANCHOR) [45]. Overall, these observations demonstrate that our method is relatively accurate and complements the other available methods.

Using a modern desktop (Intel i7-950 CPU at 3.06GHz with 24GB or RAM), the runtime of DisoRDPbind for a single protein is between 0.3 s and 1 min, depending on the chain length, and is characterized by a quadratic increase with the chain size [45]. An average size protein with about 200 residues is predicted in 1 s (*see* **Note 1**). This includes the combined runtime of the prediction of the three binding events. To put that into perspective, the runtime to predict the entire human proteome is just over 40 h on the abovementioned desktop computer, which means that DisoRDPbind can be used to predict full genomes.

2.4 Webserver

The webserver of DisoRDPbind was designed to be user-friendly and is freely available at http://biomine.ece.ualberta.ca/DisoRDPbind/. The end user only needs a modern web browser (Firefox, IE, and Chrome were tested) and internet connection to use the webserver.

The main (start) page of the webserver is for the submission of the user's query. It includes a text field where up to 5000 input protein sequences in FASTA format can be pasted and another text field for the e-mail of the user. For convenience, the server also provides an option to submit the input proteins in a FASTA-formatted file. The e-mail is required and is used to send notification when the predictions are completed. The notification provides a link to a summary page that explains the format of the outputs and the formatted text file with the predictions.

The DisoRDPbind method uses other programs to generate its inputs. Specifically, our method generates the disorder profiles utilizing IUPred [47], predicts the secondary structure with the fast version of PSIPRED (without using PSI-BLAST) [48], obtains the information about low complexity regions with the SEG algorithm [46], and transfers the functional annotations based on the alignment generated by BLAST [49]. These methods are used in a fully automated manner by the scripts that implement the webserver. Once the user provides the sequences and the e-mail and hits the "Run DisoRDPbind!" button, the results are generated without further interaction with the webserver.

2.5 Running DisoRDPbind

Three easy steps should be followed to use the DisoRDPbind webserver (the step numbers are highlighted in red in Fig. 2):

1. Copy and paste protein sequences formatted in the FASTA format into text field or upload FASTA-formatted file (an "Example" button may be used to see properly formatted example inputs) (*see* **Notes 2** and **3**).

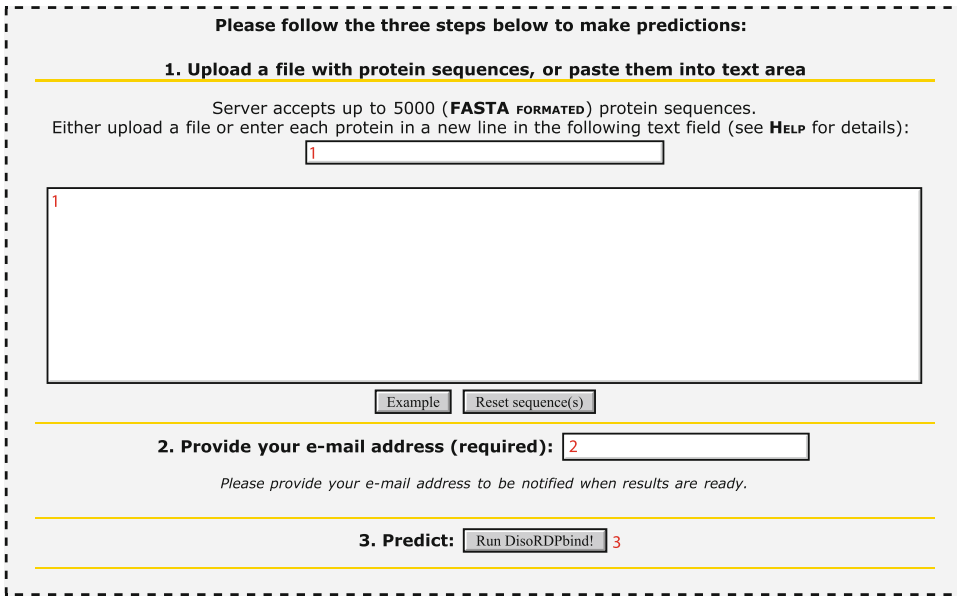


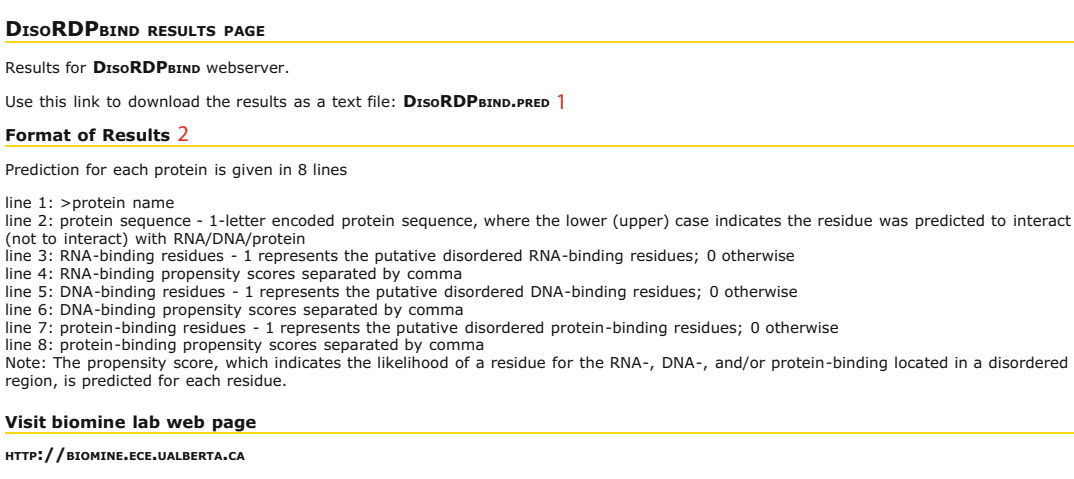
Fig. 2 Screenshot of DisoRDPbind input form on the main webserver page. The *red numbers* annotate the three steps that must be followed to run the predictions

2. Provide e-mail address (required, *see* **Note 4**). The notification e-mail, including the hyperlinks to the results page and the downloadable outputs, will send to the user once the predictions are completed.
3. Click "Run DisoRDPbind!" button to start the predictions.

Note that the webserver generates predictions of RNA, DNA, and protein binding at the same time for each input protein sequence. Once the "Run DisoRDPbind!" button is clicked, the user's web browser is redirected to another page that shows the current status of the prediction. The user's query is added to a queue of predictions on the biomine server (this server also implements a few other methods) and the position in the queue is shown and updated. The prediction is executed when the query reaches the first position in the queue. After the prediction is completed the user's web browser is automatically redirected to the page with the results and the notification e-mail with a link to this page is sent (*see* **Notes 4** and **5**). The prediction is completed and e-mail is sent even in the case when the user closes the web browser before the completion of the prediction.

2.6 Results Generated by DisoRDPbind

This webpage with the results includes a hyperlink to the downloadable text file (red number 1 in Fig. 3) and the description of the format of this file (red number 2 in Fig. 3). The text file, named DisoRDPbind.pred, is provided for download to the end user. This file includes the prediction of disordered RNA-, DNA-, and



DISORDPBIND RESULTS PAGE

Results for **DISORDPBIND** webserver.

Use this link to download the results as a text file: **DISORDPBIND.PRED** 1

Format of Results 2

Prediction for each protein is given in 8 lines

line 1: >protein name
line 2: protein sequence - 1-letter encoded protein sequence, where the lower (upper) case indicates the residue was predicted to interact (not to interact) with RNA/DNA/protein
line 3: RNA-binding residues - 1 represents the putative disordered RNA-binding residues; 0 otherwise
line 4: RNA-binding propensity scores separated by comma
line 5: DNA-binding residues - 1 represents the putative disordered DNA-binding residues; 0 otherwise
line 6: DNA-binding propensity scores separated by comma
line 7: protein-binding residues - 1 represents the putative disordered protein-binding residues; 0 otherwise
line 8: protein-binding propensity scores separated by comma
Note: The propensity score, which indicates the likelihood of a residue for the RNA-, DNA-, and/or protein-binding located in a disordered region, is predicted for each residue.

Visit biomine lab web page

HTTP://BIOMINE.ECE.UALBERTA.CA

Fig. 3 Screenshot of page with the results generated by DisoRDPbind. The *red numbers* indicate the two main parts of this page

protein-binding residues for all submitted protein sequences. For each of the three types of binding we provide a binary prediction (1 for putative binding residues and 0 for putative non-binding residues) and a real-valued propensity (higher values indicates higher likelihood for binding) for each input residue. The results are organized in eight lines per protein where six lines provide prediction for the entire input sequence and two lines lists the residues from the input sequence and its name:

- The first line lists the protein name (as provided in the user's input).
- The second line is the protein sequence where each letter identifies a residue and where a lower (upper) case indicates the residue was predicted to interact (not to interact) with RNA, DNA, or protein. This is based on the binary prediction across the three types of binding.
- The third/fifth/seventh line provides the putative binary prediction of the RNA-binding/DNA-binding/protein-binding residues (*see* **Note 6**).
- The fourth/sixth/eighth line provides the putative propensity for the RNA binding/DNA binding/protein binding for each input residue. The values of the propensities are separated by commas, they range between 0 (lowest propensity) and 1 (highest propensity), and they are provided with the precision of 3 digits after the decimal point.

The notification e-mail includes the hyperlinks to the page with the results (red number 1 in Fig. 4) and to the downloadable outputs (red number 2 in Fig. 4). The first hyperlink leads the user directly to the "DisoRDPbind Results Page" (Fig. 3). We also provide a unique job ID at the top of the e-mail. This ID

Predictions for DisoRDPbind job id: 20151005020757 are ready.

You can find the results for this job at:
http://biomine-ws.ece.ualberta.ca/webresults/DisoRDPbind/20151005020757/results.html 1
The text file can be found here:
http://biomine-ws.ece.ualberta.ca/webresults/DisoRDPbind/20151005020757/DisoRDPbind.pred 2

Upon the usage the users are requested to use the following citations:
Peng Z., Kurgan L. High-throughput prediction of RNA, DNA and protein binding regions mediated by intrinsic disorder. Nucleic Acids Res. doi:10.1093/nar/gkv585, 2015.

The webserver can be found here:
http://biomine.ece.ualberta.ca/DisoRDPbind/

Thank you for using our webserver,
Biomine group

Fig. 4 Screenshot of the notification e-mail. The *red numbers* indicate the two main parts of this e-mail

can be used to trace a given prediction query. In case if the user encounters problems then (s)he should simply reply to the e-mail with a description of what is wrong making sure that the job ID is included.

3 Case Studies

3.1 Case 1: BRCA1

BRCA1 is the breast cancer type 1 susceptibility protein which is known to play a number of important roles in controlling the development of breast cancer. The *BRCA1* gene expression is dependent on the cell cycle, and the G1–S and the G2–M transition checkpoints are controlled by the BRCA1 protein [50]. However, major functions of BRCA1 are related to the repair of chromosomal damage and to the error-free repair of DNA double-strand breaks [51]. In the norm, BRCA1 is involved in repair of the damaged DNA, or, if the DNA damage cannot be repaired, it initiates the cell destruction. The mutation-induced decrease or loss of the BRCA1 functions results in the accumulation of the damaged DNA, increasing the probability of the development of breast cancer [51]. Of the 1863 amino acids of BRCA1, only ~20% terminally located residues are involved in the formation of structured domains (residues 1–169 and 1646–1863 are folded into the RNG and tandem BRCT domains, respectively), whereas a long central region (residues 170–1645) is mostly disordered, acting as a scaffold that determines the exceptional binding promiscuity of BRCA1 [52]. Among known interacting partners of the central region of BRCA1 are several proteins involved in regulation of various biological processes. They include c-Myc, which is a proto-oncogene that is implicated in tumorigenesis, embryonic

development and apoptosis, which binds to BRCA1 at residues 173–303 and 433–511 [53]; retinoblastoma protein (pRB) that is a tumor suppressor protein dysfunctional in several tumors interacts with residues 304–394 of BRCA1 [54]; p53, which is known to acts as the guardian of the genome and a tumor suppressor [55] and binds BRCA1 at residues 224–500 [56]; Rad50, which forms a complex with Mre11 and p95/nibrin that acts in meiotic recombination, homologous recombination, nonhomologous end joining, the DNA damage response, and telomere maintenance [57], and that binds BRCA1 at residues 341–748 [58]; Rad51, which is a member of a protein family that mediates DNA strand–exchange functions related to normal recombination [59], and which interacts with the residues 758–1064 of BRCA1 [60]; FANCA, a member of the proteins related to Fanconi anemia that form a nuclear complex [61], which binds BRCA1 at residues 740–1083 [62]; whereas JunB, a transcription factor involved in regulation of the gene activity following the primary growth factor response interacts with BRCA1 at residues 1343–1440 [63]. Finally, residues 452–1079 of human BRCA1 are known to interact with DNA [64].

Results of the DisoRDPbind analysis of human BRCA1 (UniProt ID: P38398) are shown in Fig. 5a. We observe that the putative propensities clearly illustrate that this protein has a number of identifiable disordered protein- and DNA-binding sites that are located in the intrinsically disordered region of this protein. The entire long central region is predicted as protein binding, which is in agreement with the annotated native binding sites that are discussed in the previous paragraph. The known DNA-binding region, which is shown as a blue horizontal line at the bottom of Fig. 5a also lines up with the higher values of the predicted propensities for the DNA binding; we note that the DisoRDPbind webserver predicts residues with the propensities for DNA binding ≥0.245 as DNA binding (*see* **Note 6**), and such residues are fairly abundant in the native DNA-binding region.

3.2 Case 2: Yeast 60S Ribosomal Protein L4

Every living cell contains ribosomes, which are ancient ribonucleoprotein complexes serving as molecular machines for protein biosynthesis. Ribosomes are large (with the molecular mass of at least 2.5 MDa) macromolecular complexes composed of one or more ribosomal RNA molecules and a variety of proteins. Being the major force in the cellular protein production, these highly specialized machines have two major components known as the small and the large ribosomal subunits. These components have different roles in protein biosynthesis, with the small ribosomal subunit being responsible for "reading" the mRNA and with the large ribosomal subunit catalyzing the peptide bond formation. Although overall function and organization of ribosomes is similar between different organisms, prokaryotic and eukaryotic ribosomes have significant differences. For example, in prokaryotes, ribosomes are composed of ~65% of rRNA and 35% of

ribosomal proteins, whereas in eukaryotic ribosomes, the rRNA–protein ratio is close to 1. Furthermore, in prokaryotic ribosomes, small (30S) subunit includes 16S rRNA and 21 ribosomal proteins, whereas large (50S) subunit contains 5S and 23S rRNA molecules and 31 proteins [65]. In the 80S eukaryotic ribosome, the small 40S subunit contains 18S rRNA and 33 proteins, and the large 60S subunit is composed of 3 rRNA molecules (5S, 28S, and 5.8S) and 46 proteins [66]. Proteins derived from the small and large ribosomal subunits are named S1, S2, S3, … and L1, L2, L3, …, respectively. Their high conservation during evolution suggests that they have critical roles in ribosome biogenesis or functions of the mature ribosome. The ribosomal proteins are known to be enriched in intrinsic disorder [18] which is why they are relevant for our case study.

Since ribosomal proteins are abundant in every cell, and since they can interact with nucleic acids and other proteins, these RNA-binding proteins are known to be recruited to carry out many extra-ribosomal or auxiliary functions, i.e., they serve as moonlighting proteins [67–70]. It has been pointed out that the ribosomal proteins might have over 30 extra-ribosomal functions including regulation of the gene-specific control of transcription, transcript-specific translational control, and surveillance of ribosome synthesis and they could be involved in induction of cell-cycle arrest or apoptosis and in regulation of normal development and cancer [67, 69, 70]. One of the characteristic examples of such moonlighting ribosomal proteins is given by the protein L4. The L4 protein is annotated to have 24% of disordered residues in the MobiDB database [71] which are localized in the several regions including residues 1–11, 52–91, 189–196, 300–313, and at the C-terminus starting at the residue 341. This is also in agreement with the D^2P^2 database [72] that lists residues 1–12, 72–81, 193-–94, 306–311, and 347–351 as disordered. This protein is known to both inhibit [73] and attenuate [74] the translation of the S10 operon, which, in *E.coli*, encodes eleven different ribosomal proteins, one of which is L4 itself [75]. Also, L4 can bind to RNase E (which is a part of the degradosome that plays an important role in mRNA turnover as well as in the processing and decay of noncoding RNAs), modulate activity of this crucial nuclease and thereby regulate mRNA composition in response to stress [76]. Curiously, eukaryotic L4 seems to be also engaged in the extra-ribosomal functions. In fact, recently it has been pointed out that this protein plays an important role in the ribosome biogenesis, since the deletion of the universally conserved internal loop of yeast L4 resulted in severe impairment of the growth and reduction of the levels of large ribosomal subunits [77], and since the eukaryote-specific acidic C-terminal extension (residues 265–362) is involved in several distinct interactions with the 60S surface needed for the hierarchical ribosome assembly [78]. Therefore, the internal loop (~60 residues) is known to be involved in

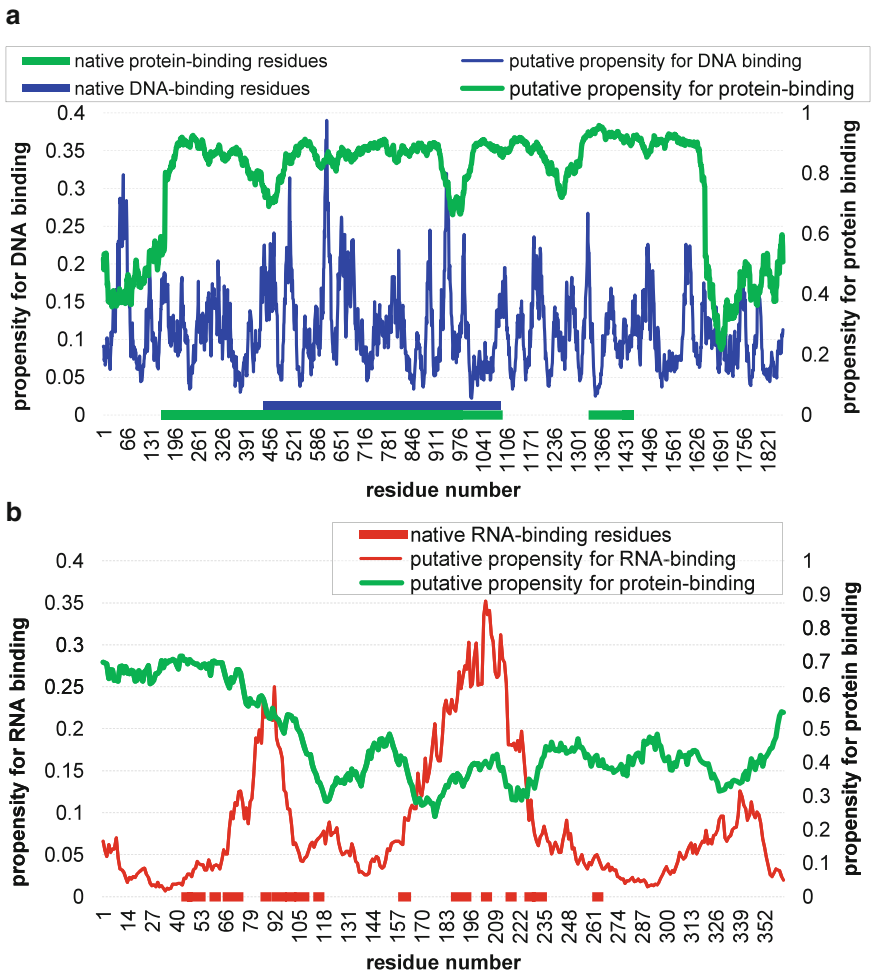


Fig. 5 Predictions generated by DisoRDPbind for human BRCA1 protein (UniProt ID: P38398) (panel **a**) and yeast 60S ribosomal protein L4 (UniProt ID: P10664) (panel **b**). The putative propensities for DNA binding in panel **a** and RNA binding in panel **b** are shown using *blue* and *red line*, respectively; and putative propensities for protein binding are shown using *green lines*. The native annotations are shown using *horizontal lines* that lie on the *x*-axis

interaction with the chaperone Acl4 involved in the assembly of the mature ribosome and later binds to the cognate nascent rRNA site [78]. In fact, in mature ribosome, the aforementioned loop (residues 46–111) protrudes from the globular folded core of L4 and deeply projects into the 25S rRNA core, lining a peptide exit tunnel of the mature ribosome [78].

Figure 5b represents results of the DisoRDPbind-based analysis of the interactions of yeast 60S ribosomal protein L4 (UniProt ID: P10664) with RNA (red lines) and proteins (green lines). We also annotate the native RNA-binding regions (red horizontal line at the bottom of Fig. 5b) which were collected from the protein–ligand binding database BioLiP [79]; they are in agreement with

the discussion in the above paragraph. We observe that the two large peaks in the putative propensities for disordered RNA binding align with the localization of the native RNA-binding regions; the DisoRDPbind webserver predicts residues with the propensities for RNA binding ≥0.151 as RNA binding (*see* **Note 6**). Also, both MobiDB and D^2P^2 suggest that these regions are intrinsically disordered. Although the predicted propensities for the protein binding are below a cut-off value (the DisoRDPbind webserver predicts residues with the propensities for protein binding ≥0.799 as protein binding; *see* **Note 6**), the N-terminus is predicted with relatively high values that suggest potential for disordered protein binding.

Overall, we conclude that both case studies demonstrate that our webserver generates predictions that provide useful clues to find native disordered DNA-, RNA-, and protein-binding regions.

4 Notes

1. The runtime in milliseconds for a given sequence with n residues can be estimated using the following formula, time = $0.0077*n^2 + 0.9028*n + 301.06$. This formula was estimated based on empirical data discussed in [45]. Given $n = 200$, time = 789.6 [milliseconds] = 0.79 [seconds]. Given $n = 1000$, time = 8903.7 [milliseconds] = 8.9 [second]. This formula can be used to estimate a total runtime for a large set of proteins since predictions on the webserver are run serially.
2. Server accepts between 1 and 5000 protein sequences. The user must submit their sequence(s) in FASTA format to guarantee they will receive the correct response from DisoRDPbind webserver. This format is described at https://en.wikipedia.org/wiki/FASTA_format
3. Due to a limitation of one of the methods that is used to generate DisoRDPbind sequence features (i.e., secondary structure profile predicted by PSIPRED), the webserver cannot process very long (>10,000 residues) protein chains.
4. Although DisoRDPbind can predict an average size protein with about 200 residues within 1 s, it may take hours to process the prediction for thousands of (up to 5000) protein sequences. Keeping the web browser window open this long could be prohibitive. Therefore, we require the user to provide an e-mail address where (s)he will be notified when the results are available and how to access these results.
5. User should store the link to the results for future reference. We store the results of the prediction for at least 3 months under the provided link. Although the same link that is shown in the web browser window is sent via e-mail, we advise users to copy the

link from the web browser. This is in case if an invalid e-mail address was entered and thus no e-mail will reach the user.

6. The binary prediction is generated from the predicted propensity scores using a threshold, i.e., residues with the propensity higher than the threshold are assigned with the binary value 1 and the remaining residues are assigned with 0. These thresholds equal 0.245, 0.151, and 0.799 for the predictions of the disordered DNA, RNA, and protein binding, respectively. They correspond to the FP-rate (fraction of incorrectly predicted non-binding residues) of 0.1 that was estimated using the TRAINING dataset. This means that the user should expect that among the predicted binding residues there are about 10% of the non-binding residues.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (Grant No. 11501407) and the China National 863 High-Tech Program (Grant No. 2015AA020101) to Z.P. and by the Discovery grant from the Natural Sciences and Engineering Research Council of Canada to L.K. that was used to fund C.W.

References

1. Keith Dunker A, Madan Babu M, Barbar E, Blackledge M, Bondos SE, Dosztányi Z, Jane Dyson H, Forman-Kay J, Fuxreiter M, Gsponer J, Han K-H, Jones DT, Longhi S, Metallo SJ, Nishikawa K, Nussinov R, Obradovic Z, Pappu RV, Rost B, Selenko P, Subramaniam V, Sussman JL, Tompa P, Uversky VN (2013) What's in a name? Why these proteins are intrinsically disordered. Intrinsically Disordered Proteins 1(1):e24157
2. Guharoy M, Pauwels K, Tompa P (2015) SnapShot: intrinsic structural disorder. Cell 161(5):1230. doi:10.1016/j.cell.2015.05.024, e1231
3. Habchi J, Tompa P, Longhi S, Uversky VN (2014) Introducing protein intrinsic disorder. Chem Rev 114(13):6561–6588. doi:10.1021/cr400514h
4. van der Lee R, Buljan M, Lang B, Weatheritt RJ, Daughdrill GW, Dunker AK, Fuxreiter M, Gough J, Gsponer J, Jones DT, Kim PM, Kriwacki RW, Oldfield CJ, Pappu RV, Tompa P, Uversky VN, Wright PE, Babu MM (2014) Classification of intrinsically disordered regions and proteins. Chem Rev 114(13):6589–6631. doi:10.1021/cr400525m
5. Peng Z, Yan J, Fan X, Mizianty MJ, Xue B, Wang K, Hu G, Uversky VN, Kurgan L (2015) Exceptionally abundant exceptions: comprehensive characterization of intrinsic disorder in all domains of life. Cell Mol Life Sci 72(1):137–151. doi:10.1007/s00018-014-1661-9
6. Yan J, Mizianty MJ, Filipow PL, Uversky VN, Kurgan L (2013) RAPID: fast and accurate sequence-based prediction of intrinsic disorder content on proteomic scale. Biochim Biophys Acta 1834(8):1671–1680. doi:10.1016/j.bbapap.2013.05.022
7. Atkins JD, Boateng SY, Sorensen T, McGuffin LJ (2015) Disorder prediction methods, their applicability to different protein targets and their usefulness for guiding experimental studies. Int J Mol Sci 16(8):19040–19054. doi:10.3390/ijms160819040
8. Bhowmick P, Guharoy M, Tompa P (2015) Bioinformatics approaches for predicting disordered protein motifs. Adv Exp Med Biol 870:291–318. doi:10.1007/978-3-319-20164-1_9
9. Deng X, Eickholt J, Cheng J (2012) A comprehensive overview of computational protein disorder prediction methods. Mol Biosyst 8(1):114–121. doi:10.1039/c1mb05207a

10. Dosztanyi Z, Meszaros B, Simon I (2010) Bioinformatical approaches to characterize intrinsically disordered/unstructured proteins. Brief Bioinform 11(2):225–243. doi:10.1093/bib/bbp061bbp061 [pii]
11. He B, Wang K, Liu Y, Xue B, Uversky VN, Dunker AK (2009) Predicting intrinsic disorder in proteins: an overview. Cell Res 19(8):929–949. doi:10.1038/cr.2009.87
12. Monastyrskyy B, Kryshtafovych A, Moult J, Tramontano A, Fidelis K (2014) Assessment of protein disorder region predictions in CASP10. Proteins 82(Suppl 2):127–137. doi:10.1002/prot.24391
13. Peng ZL, Kurgan L (2012) Comprehensive comparative assessment of in-silico predictors of disordered regions. Curr Protein Pept Sci 13(1):6–18
14. Galea CA, High AA, Obenauer JC, Mishra A, Park CG, Punta M, Schllessinger A, Ma J, Rost B, Slaughter CA, Kriwacki RW (2009) Large-scale analysis of thermostable, mammalian proteins provides insights into the intrinsically disordered proteome. J Proteome Res 8(1):211–226. doi:10.1021/pr800308v
15. Tompa P, Dosztanyi Z, Simon I (2006) Prevalent structural disorder in E-coli and S-cerevisiae proteomes. J Proteome Res 5(8):1996–2000. doi:10.1021/Pr0600881
16. Ward JJ, Sodhi JS, McGuffin LJ, Buxton BF, Jones DT (2004) Prediction and functional analysis of native disorder in proteins from the three kingdoms of life. J Mol Biol 337(3):635–645,10.1016/j.jmb.2004.02.002S0022283604001482 [pii]
17. Xue B, Dunker AK, Uversky VN (2012) Orderly order in protein intrinsic disorder distribution: disorder in 3500 proteomes from viruses and the three domains of life. J Biomol Struct Dyn 30(2):137–149. doi:10.1080/07391102.2012.675145
18. Peng Z, Oldfield CJ, Xue B, Mizianty MJ, Dunker AK, Kurgan L, Uversky VN (2014) A creature with a hundred waggly tails: intrinsically disordered proteins in the ribosome. Cell Mol Life Sci 71(8):1477–1504. doi:10.1007/s00018-013-1446-6
19. Peng Z, Xue B, Kurgan L, Uversky VN (2013) Resilience of death: intrinsic disorder in proteins involved in the programmed cell death. Cell Death Differ 20(9):1257–1267. doi:10.1038/cdd.2013.65
20. Peng Z, Mizianty MJ, Xue B, Kurgan L, Uversky VN (2012) More than just tails: intrinsic disorder in histone proteins. Mol Biosyst 8(7):1886–1901. doi:10.1039/c2mb25102g
21. Chen JW, Romero P, Uversky VN, Dunker AK (2006) Conservation of intrinsic disorder in protein domains and families: II. Functions of conserved disorder. J Proteome Res 5(4):888–898. doi:10.1021/Pr060049p
22. Cumberworth A, Lamour G, Babu MM, Gsponer J (2013) Promiscuity as a functional trait: intrinsically disordered regions as central players of interactomes. Biochem J 454:361–369. doi:10.1042/Bj20130545
23. Dyson HJ (2012) Roles of intrinsic disorder in protein-nucleic acid interactions. Mol Biosyst 8(1):97–104. doi:10.1039/C1mb05258f
24. Fuxreiter M, Toth-Petroczy A, Kraut DA, Matouschek AT, Lim RYH, Xue B, Kurgan L, Uversky VN (2014) Disordered proteinaceous machines. Chem Rev 114(13):6806–6843. doi:10.1021/Cr4007329
25. Haynes C, Oldfield CJ, Ji F, Klitgord N, Cusick ME, Radivojac P, Uversky VN, Vidal M, Iakoucheva LM (2006) Intrinsic disorder is a common feature of hub proteins from four eukaryotic interactomes. PLoS Comput Biol 2(8):890–901, doi: ARTN e100 DOI 10.1371/journal.pcbi.0020100
26. Tompa P, Csermely P (2004) The role of structural disorder in the function of RNA and protein chaperones. FASEB J 18(11):1169–1175. doi:10.1096/fj.04-1584rev
27. Wu Z, Hu G, Yang J, Peng Z, Uversky VN, Kurgan L (2015) In various protein complexes, disordered protomers have large per-residue surface areas and area of protein-, DNA- and RNA-binding interfaces. FEBS Lett 589(19 Pt A):2561–2569. doi:10.1016/j.febslet.2015.08.014
28. Shen J, Zhang J, Luo X, Zhu W, Yu K, Chen K, Li Y, Jiang H (2007) Predicting protein-protein interactions based only on sequences information. Proc Natl Acad Sci U S A 104(11):4337–4341. doi:10.1073/pnas.0607879104, 0607879104 [pii]
29. Zhang QC, Petrey D, Deng L, Qiang L, Sin Y, Thu CA, Bisikirska B, Lefebvre C, Accili D, Hunter T, Maniatis T, Califano A, Honig B (2013) Structure-based prediction of protein-protein interactions on a genome-wide scale (vol 490, pg 556, 2012). Nature 495(7439):127. doi:10.1038/Nature11977
30. Puton T, Kozlowski L, Tuszynska I, Rother K, Bujnicki JM (2012) Computational methods for prediction of protein-RNA interactions. J Struct Biol 179(3):261–268. doi:10.1016/j.jsb.2011.10.001
31. Zhao HY, Yang YD, Zhou YQ (2013) Prediction of RNA binding proteins comes of

age from low resolution to high resolution. Mol Biosyst 9(10):2417–2425. doi:10.1039/C3mb70167k

32. Kauffman C, Karypis G (2012) Computational tools for protein-DNA interactions. Wiley Interdiscipl Rev-Data Mining and Knowl Discov 2(1):14–28. doi:10.1002/Widm.48
33. Gromiha MM, Nagarajan R (2013) Computational approaches for predicting the binding sites and understanding the recognition mechanism of protein-DNA complexes. Protein-Nucleic Acids Interact 91:65–99. doi:10.1016/B978-0-12-411637-5.00003-2
34. Varadi M, Vranken W, Guharoy M, Tompa P (2015) Computational approaches for inferring the functions of intrinsically disordered proteins. Front Mol Biosci 2:45. doi:10.3389/fmolb.2015.00045
35. Sharma A, Dehzangi A, Lyons J, Imoto S, Miyano S, Nakai K, Patil A (2014) Evaluation of sequence features from intrinsically disordered regions for the estimation of protein function. PLoS One 9(2). doi: ARTN e89890 DOI 10.1371/journal.pone.0089890
36. Cheng Y, Oldfield CJ, Meng J, Romero P, Uversky VN, Dunker AK (2007) Mining alpha-helix-forming molecular recognition features with cross species sequence alignments. Biochemistry 46(47):13468–13477. doi:10.1021/bi7012273
37. Meszaros B, Simon I, Dosztanyi Z (2009) Prediction of protein binding regions in disordered proteins. PLoS Comput Biol 5(5):e1000376. doi:10.1371/journal.pcbi.1000376
38. Disfani FM, Hsu WL, Mizianty MJ, Oldfield CJ, Xue B, Dunker AK, Uversky VN, Kurgan L (2012) MoRFpred, a computational tool for sequence-based prediction and characterization of short disorder-to-order transitioning binding regions in proteins. Bioinformatics 28(12):i75–i83. doi:10.1093/bioinformatics/bts209
39. Khan W, Duffy F, Pollastri G, Shields DC, Mooney C (2013) Predicting binding within disordered protein regions to structurally characterised peptide-binding domains. PLoS One 8(9). doi: ARTN e72838 DOI 10.1371/journal.pone.0072838
40. Fang C, Noguchi T, Tominaga D, Yamana H (2013) MFSPSSMpred: identifying short disorder-to-order binding regions in disordered proteins based on contextual local evolutionary conservation. BMC Bioinformatics 14:300. doi:10.1186/1471-2105-14-300
41. Jones DT, Cozzetto D (2014) DISOPRED3: precise disordered region predictions with annotated protein-binding activity. Bioinformatics. doi:10.1093/bioinformatics/btu744
42. Malhis N, Gsponer J (2015) Computational identification of MoRFs in protein sequences. Bioinformatics 31(11):1738–1744. doi:10.1093/bioinformatics/btv060
43. Yan J, Dunker AK, Uversky VN, Kurgan L (2016) Molecular recognition features (MoRFs) in three domains of life. Mol Biosyst 12:697–710
44. Sickmeier M, Hamilton JA, LeGall T, Vacic V, Cortese MS, Tantos A, Szabo B, Tompa P, Chen J, Uversky VN, Obradovic Z, Dunker AK (2007) DisProt: the database of disordered proteins. Nucleic Acids Res 35(Database issue):D786–D793. doi:10.1093/nar/gkl893
45. Peng Z, Kurgan L (2015) High-throughput prediction of RNA, DNA and protein binding regions mediated by intrinsic disorder. Nucleic Acids Res. 43(18):e121. doi:10.1093/nar/gkv585
46. Wootton JC, Federhen S (1993) Statistics of local complexity in amino-acid-sequences and sequence databases. Comput Chem 17(2):149–163. doi:10.1016/0097-8485(93)85006-X
47. Dosztanyi Z, Csizmok V, Tompa P, Simon I (2005) IUPred: web server for the prediction of intrinsically unstructured regions of proteins based on estimated energy content. Bioinformatics 21(16):3433–3434. doi:10.1093/bioinformatics/bti541
48. McGuffin LJ, Bryson K, Jones DT (2000) The PSIPRED protein structure prediction server. Bioinformatics 16(4):404–405
49. Altschul SF, Madden TL, Schaffer AA, Zhang JH, Zhang Z, Miller W, Lipman DJ (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25(17):3389–3402. doi:10.1093/nar/25.17.3389
50. Vaughn JP, Davis PL, Jarboe MD, Huper G, Evans AC, Wiseman RW, Berchuck A, Iglehart JD, Futreal PA, Marks JR (1996) BRCA1 expression is induced before DNA synthesis in both normal and tumor-derived breast cells. Cell Growth Differ 7(6):711–715
51. Friedenson B (2007) The BRCA1/2 pathway prevents hematologic cancers in addition to breast and ovarian cancers. BMC Cancer 7:152. doi:10.1186/1471-2407-7-152, 1471-2407-7-152 [pii]
52. Mark WY, Liao JC, Lu Y, Ayed A, Laister R, Szymczyna B, Chakrabartty A, Arrowsmith CH (2005) Characterization of segments from the central region of BRCA1: an intrinsically disordered scaffold for multiple protein-protein and protein-DNA interactions?

J Mol Biol 345(2):275–287. doi:10.1016/j.jmb.2004.10.045

53. Wang Q, Zhang H, Kajino K, Greene MI (1998) BRCA1 binds c-Myc and inhibits its transcriptional and transforming activity in cells. Oncogene 17(15):1939–1948. doi:10.1038/sj.onc.1202403
54. Aprelikova ON, Fang BS, Meissner EG, Cotter S, Campbell M, Kuthiala A, Bessho M, Jensen RA, Liu ET (1999) BRCA1-associated growth arrest is RB-dependent. Proc Natl Acad Sci U S A 96(21):11866–11871
55. Lane DP (1992) p53, guardian of the genome. Nature 358(6381):15–16
56. Zhang H, Somasundaram K, Peng Y, Tian H, Bi D, Weber BL, El-Deiry WS (1998) BRCA1 physically associates with p53 and stimulates its transcriptional activity. Oncogene 16(13):1713–1721. doi:10.1038/sj.onc.1201932
57. Haber JE (1998) The many interfaces of Mre11. Cell 95(5):583–586, doi:S0092-8674(00)81626-8 [pii]
58. Zhong Q, Chen CF, Li S, Chen Y, Wang CC, Xiao J, Chen PL, Sharp ZD, Lee WH (1999) Association of BRCA1 with the hRad50-hMre11-p95 complex and the DNA damage response. Science 285(5428):747–750, doi:7719 [pii]
59. Baumann P, Benson FE, West SC (1996) Human Rad51 protein promotes ATP-dependent homologous pairing and strand transfer reactions in vitro. Cell 87(4):757–766, doi:S0092-8674(00)81394-X [pii]
60. Scully R, Chen J, Plug A, Xiao Y, Weaver D, Feunteun J, Ashley T, Livingston DM (1997) Association of BRCA1 with Rad51 in mitotic and meiotic cells. Cell 88(2):265–275, doi:S0092-8674(00)81847-4 [pii]
61. Garcia-Higuera I, Kuang Y, Naf D, Wasik J, D'Andrea AD (1999) Fanconi anemia proteins FANCA, FANCC, and FANCG/XRCC9 interact in a functional nuclear complex. Mol Cell Biol 19(7):4866–4873
62. Folias A, Matkovic M, Bruun D, Reid S, Hejna J, Grompe M, D'Andrea A, Moses R (2002) BRCA1 interacts directly with the Fanconi anemia protein FANCA. Hum Mol Genet 11(21):2591–2597
63. Hu YF, Li R (2002) JunB potentiates function of BRCA1 activation domain 1 (AD1) through a coiled-coil-mediated interaction. Genes Dev 16(12):1509–1517. doi:10.1101/gad.995502
64. Paull TT, Cortez D, Bowers B, Elledge SJ, Gellert M (2001) Direct DNA binding by Brca1. Proc Natl Acad Sci U S A 98(11):6086–6091. doi:10.1073/pnas.111125998, 111125998 [pii]
65. Cate JH, Yusupov MM, Yusupova GZ, Earnest TN, Noller HF (1999) X-ray crystal structures of 70S ribosome functional complexes. Science 285(5436):2095–2104, doi:7861 [pii]
66. Ben-Shem A, Garreau de Loubresse N, Melnikov S, Jenner L, Yusupova G, Yusupov M (2011) The structure of the eukaryotic ribosome at 3.0 A resolution. Science 334(6062):1524–1529. doi:10.1126/science.1212642, science.1212642 [pii]
67. Wool IG (1996) Extraribosomal functions of ribosomal proteins. Trends Biochem Sci 21(5):164–165, doi:S0968-0004(96)20011-8 [pii]
68. Weisberg RA (2008) Transcription by moonlight: structural basis of an extraribosomal activity of ribosomal protein S10. Mol Cell 32(6):747–748. doi:10.1016/j.molcel.2008.12.010, S1097-2765(08)00851-4 [pii]
69. Lindstrom MS (2009) Emerging functions of ribosomal proteins in gene-specific transcription and translation. Biochem Biophys Res Commun 379(2):167–170. doi:10.1016/j.bbrc.2008.12.083, S0006-291X(08)02492-3 [pii]
70. Warner JR, McIntosh KB (2009) How common are extraribosomal functions of ribosomal proteins? Mol Cell 34(1):3–11. doi:10.1016/j.molcel.2009.03.006, S1097-2765(09)00177-4 [pii]
71. Potenza E, Di Domenico T, Walsh I, Tosatto SC (2015) MobiDB 2.0: an improved database of intrinsically disordered and mobile proteins. Nucleic Acids Res 43(Database issue):D315–D320. doi:10.1093/nar/gku982
72. Oates ME, Romero P, Ishida T, Ghalwash M, Mizianty MJ, Xue B, Dosztanyi Z, Uversky VN, Obradovic Z, Kurgan L, Dunker AK, Gough J (2013) D(2)P(2): database of disordered protein predictions. Nucleic Acids Res 41(Database issue):D508–D516. doi:10.1093/nar/gks1226
73. Gaal T, Bartlett MS, Ross W, Turnbough CL Jr, Gourse RL (1997) Transcription regulation by initiating NTP concentration: rRNA synthesis in bacteria. Science 278(5346):2092–2097
74. Zengel JM, Lindahl L (1994) Diverse mechanisms for regulating ribosomal protein synthesis in Escherichia coli. Prog Nucleic Acid Res Mol Biol 47:331–370
75. Mikhaylina AO, Kostareva OS, Sarskikh AV, Fedorov RV, Piendl W, Garber MB, Tishchenko SV (2014) Investigation of the regulatory function of archaeal ribosomal protein L4. Biochemistry (Mosc) 79(1):69–76. doi:10.1134/S0006297914010106, BCM79010087 [pii]

76. Singh D, Chang SJ, Lin PH, Averina OV, Kaberdin VR, Lin-Chao S (2009) Regulation of ribonuclease E activity by the L4 ribosomal protein of Escherichia coli. Proc Natl Acad Sci U S A 106(3):864–869. doi:10.1073/pnas.0810205106, 0810205106 [pii]
77. Gamalinda M, Woolford JL Jr (2014) Deletion of L4 domains reveals insights into the importance of ribosomal protein extensions in eukaryotic ribosome assembly. RNA 20(11):1725–1731. doi:10.1261/rna.046649.114, rna.046649.114 [pii]
78. Stelter P, Huber FM, Kunze R, Flemming D, Hoelz A, Hurt E (2015) Coordinated ribosomal L4 protein assembly into the pre-ribosome is regulated by its eukaryote-specific extension. Mol Cell 58(5):854–862. doi:10.1016/j.molcel.2015.03.029, S1097-2765(15)00220-8 [pii]
79. Yang J, Roy A, Zhang Y (2013) BioLiP: a semi-manually curated database for biologically relevant ligand-protein interactions. Nucleic Acids Res 41(Database issue):D1096–D1103. doi:10.1093/nar/gks966

Chapter 15

Sequence-Based Prediction of RNA-Binding Residues in Proteins

Rasna R. Walia, Yasser EL-Manzalawy, Vasant G. Honavar, and Drena Dobbs

Abstract

Identifying individual residues in the interfaces of protein–RNA complexes is important for understanding the molecular determinants of protein–RNA recognition and has many potential applications. Recent technical advances have led to several high-throughput experimental methods for identifying partners in protein–RNA complexes, but determining RNA-binding residues in proteins is still expensive and time-consuming. This chapter focuses on available computational methods for identifying which amino acids in an RNA-binding protein participate directly in contacting RNA. Step-by-step protocols for using three different web-based servers to predict RNA-binding residues are described. In addition, currently available web servers and software tools for predicting RNA-binding sites, as well as databases that contain valuable information about known protein–RNA complexes, RNA-binding motifs in proteins, and protein-binding recognition sites in RNA are provided. We emphasize sequence-based methods that can reliably identify interfacial residues without the requirement for structural information regarding either the RNA-binding protein or its RNA partner.

Key words Protein–RNA interfaces, Binding site prediction, Machine learning, RNA-binding proteins (RBPs), Ribonucleoprotein particles (RNPs), Homology-based prediction, RNABindRPlus, SNBRFinder, PS-PRIP, FastRNABindR

1 Introduction

RNA-binding proteins (RBPs) are key regulators of cellular and developmental processes [1], playing pivotal roles in the posttranscriptional splicing and localization of mRNAs [2–5], mediating the activities of noncoding RNAs (ncRNAs) [6, 7] and even "moonlighting" as metabolic enzymes [8, 9] and promoting phase transitions to generate stress granules inside cells [10]. Defects in RBPs and ribonucleoprotein particles (RNPs) have been linked to immunological disorders [11], cancer [12, 13], and neurodegenerative diseases in humans [5, 14]. Still, even though the human genome encodes more than 1500 different RNA-binding proteins [15,

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_15,

16]—at least as many RBPs as DNA-binding transcription factors [17]—our understanding of the cellular roles of RBPs, how they recognize their targets, and how they are regulated has lagged far behind our understanding of transcription factors. Recent exciting developments have begun to close this gap, providing proteome-wide catalogs and databases of RNA-binding proteins, "RNA interactomes" or "RBPomes" [18–21], an impressive compendium of RNA recognition sites [22], detailed views of the architecture and dynamics of important RNP complexes and RNA viruses, e.g., refs. [23, 24], and substantial progress in engineering RBPs with customized functions and high specificity for desired RNA targets [25, 26].

RNA-binding proteins are often modular, and many well-characterized RBPs contain one or more conserved RNA-binding domains or motifs [1, 27]. The RNA recognition motif (RRM), for example, is one of the most abundant structural motifs in vertebrate proteins, and is found in ~2% of all human proteins [25]. Other abundant RNA-binding domains and motifs include the KH, dsRBD, DEAD-Box, PUF, SAM, and ZnF domains [1, 27], all which have conserved structures and can be recognized in the primary sequences of proteins (*see* Subheading 3.1, **step 6** below). However, only ~50% of the mRNA-binding proteins identified by "interactome capture" in HeLa cells contain a characterized RNA-binding domain [19]. Also, many RBPs bind RNA through intrinsically disordered regions (IDRs), which are thought to promote formation of extended interaction interfaces and contribute to the generation of higher order assemblies and the formation of RNA granules [28, 29]. Finally, a survey of available structures for protein–RNA complexes revealed that the majority of amino acids in the protein–RNA interface are not part of a characterized RNA-binding motif [30] and the presence of an RNA-binding signature does not conclusively identify the specific amino acids involved in RNA recognition and binding.

The most definitive way to identify RNA-binding residues (i.e., residues that directly contact RNA) (*see* **Note 1**) is to extract them from a high-resolution experimental structure of a protein–RNA complex. Three-dimensional structures are available for only a small fraction of the known protein–RNA complexes [31]. As of December 16, 2015, the number of solved structures in the Protein Data Bank (PDB) for protein–RNA complexes is only 1661 out of 114,402 total structures, and ~40% of the RNA-containing structures in the PDB correspond to ribosomes. Protein–RNA complexes can be very difficult to crystallize and many are too large for structure determination using NMR spectroscopy [32, 33]. Fortunately, recent advances in NMR [34], cryo-electron microscopy [35], and small-angle X-ray scattering (SAXS) [36] offer tremendous promise for providing structural details for RNPs that have been recalcitrant to experimental structure determination. At present, in the absence of a 3D structure, several types of

experiments can be used to identify RNA-binding residues that are required for function (e.g., site-specific mutagenesis) or residues that are either required for high affinity binding or are located in close proximity to RNA in protein–RNA complexes, either in vivo or in vitro (e.g., co-immunoprecipitation assays, cross-linking mass spectrometry, yeast 3-hybrid assays, footprinting, and electrophoretic shift assays (reviewed in refs. [1, 27, 38]).

The development of high-throughput CHIP and RNASeq-based methods that employ a combination of in vivo cross-linking and immunoprecipitation (e.g., RIP-Chip, HITS-CLIP, PAR-CLIP, iCLIP, and CRAC) has made it possible to identify RNAs bound by specific proteins on a genome-wide scale (reviewed in refs. [1, 39, 40]). Along with these advances, several powerful integrated biochemical/bioinformatics approaches can identify both the target RNAs and the specific ribonucleotides recognized by the RNA-binding proteins [41–43]. In contrast, at present, there are no truly high-throughput experimental approaches for identifying interfacial residues in the protein component of a protein–RNA complex, although CLAMP [44] and other cross-linking and combined cross-linking mass spectrometry methods can identify interfacial residues in both the protein and RNA [37, 45, 46]. Despite all of these impressive advances, the cost and effort involved in the experimental determination of protein–RNA complex structures and/or identifying specific RNA-binding residues in proteins, has created a need for reliable computational approaches that can predict the most likely RNA-binding residues in proteins.

Computational approaches to predicting protein–RNA interfaces have been the topic of several recent reviews and benchmark comparisons [31, 47–50]. These approaches can be broadly classified into sequence- and structure-based methods [31, 47]. Sequence-based methods use sequence-derived features (such as amino acid identity or physicochemical properties) of a target residue and its sequence neighbors to make predictions. Structure-based methods use structure-derived features (such as solvent-accessible surface area or secondary structure) of a target residue and its sequence or structural neighbors to make predictions. Both sequence-based and structure-based approaches could, in theory, take advantage of recognizable RNA-binding motifs in RBPs and protein-binding motifs in their RNA targets. But, although hundreds of RNA-binding domains, motifs and signatures are annotated in the **InterPro** resource [51], at present there is no comprehensive database focused specifically on RNA-binding motifs in proteins (*see* **Note 2**). For protein-binding motifs in RNA, there is a valuable compendium of "RNA-binding motifs" (i.e., RNA motifs recognized by RBPs) [22] and excellent databases of RNA sequence motifs and binding specificities [41, 43], which provide experimentally determined recognition sites in RNA for a large number of RBPs. Also, one of the protocols provided

here, **PS-PRIP** (*see* Subheading 3.3) employs a dataset of interfacial sequence motifs from RBPs and their targets to predict RNA-binding residues *and* protein-binding residues in the RNA component of specific protein–RNA complexes [52].

Recent benchmark comparisons of software and servers for predicting RNA-binding residues in proteins [31, 47] have demonstrated that the performance of methods that require only sequence information is often superior to that of methods that require structural information. One reason for this is that the best sequence-based methods encode sequences using PSSMs (Position-Specific Scoring Matrices) (*see* **Note 3**), which capture powerful evolutionary information from large multiple alignments of homologous sequences. In considering potential RNA-binding residues in a specific protein of interest, however, the user is strongly encouraged to take advantage of any available structural information, especially in evaluating the validity of predictions. For example, in most cases, RNA-binding residues are located on the solvent-exposed surface of the protein. Any predicted RNA-binding residues that are buried in the three-dimensional structure of a protein should be viewed with suspicion, although buried interfacial residues in "unbound" protein structures can become exposed due to conformational changes in the protein that occur upon RNA binding [28, 53–55].

Another way in which structural information can be exploited to accurately identify potential RNA-binding residues is illustrated in the so-called "homology-based" approaches. Homology-based approaches take advantage of the observation that RNA-binding residues are often conserved across homologous proteins [56, 57]. Thus, if a "bound" structure is available for a close sequence homolog of the query protein, the RNA-binding residues of the query protein can be inferred, based on their alignment with the known RNA-binding residues in the homologous sequence. When applicable, homology-based approaches provide the most reliable computational predictions of RNA-binding sites, but they have an important limitation: if no homologs with experimentally determined bound structures are available for the query protein, no predictions can be generated. This limitation can be overcome by combining a homology-based method, with a machine learning-based method, which can return predictions for every residue in any protein. This is the strategy employed by **RNABindRPlus** (*see* Subheading 3.2), which combines a PSSM-based Support Vector Machine (SVM) with a homology-based method to generate highly reliable predictions [57], and by **SNBRFinder** (*see* Subheading 3.3), which combines an SVM classifier that uses sequence profiles, residue conservation scores, physicochemical properties and interface propensities, with a homology-based method that uses profile hidden Markov models (HMMs) to search for the homologs [58].

The major goal of the chapter is to provide a step-by-step protocol for predicting RNA-binding residues in proteins, with a focus on machine learning and homology-based methods. In keeping with the theme of this volume, the methods outlined here are sequence-based; they do not require structural information regarding the protein of interest. We also provide a brief guide to accessing and utilizing state-of-the-art computational methods, web servers and databases that provide information about interfaces in protein–RNA complexes and/or predictions of RNA-binding residues in proteins. For additional information, the reader is referred to two excellent reviews: a recent review by Si et al. [50], which includes a comprehensive table of available sequence, structure and docking based methods; and a review by Tuszynska et al. [59], which focuses on structural docking-based approaches which are not considered here.

In this chapter, we focus on currently available web-based computational tools for interface prediction, i.e., predicting which specific amino acid residues in an RNA-binding protein are involved in recognition of and binding to RNA. A few tools are also capable of predicting the converse, i.e., which ribonucleotides in the bound RNA directly contact the protein of interest (e.g., [52, 60, 61)]. Software and servers for partner prediction, i.e., predicting which RNA(s) bind to a specific protein of interest (or *vice versa*) in a protein–RNA complex or a protein–RNA interaction network, are not described here, but have been reviewed elsewhere [62–65]. Tools for predicting whether or not a query protein is likely to bind RNA are also available (e.g., Tartaglia [39, 66, 67)]. but are not considered here.

The protocol involves two major steps (illustrated in Fig. 1):

Step 1: Determine whether experimental data regarding RNA-binding residues in the query RNA-binding protein (or putative RNA-binding protein) are already available. This step is described in Subheading 3.1, which outlines strategies for exploiting available online databases and servers (provided in Table 1 below) that provide structural data regarding protein–RNA complexes, or focus on RNA-binding proteins, RNA-binding motifs, or protein–RNA interactions.

Step 2: If known RNA-binding residues cannot be identified using available resources, or if the user wishes to identify additional potential interfacial residues, use one (or, preferably, all three) of the following web-based tools for predicting RNA-binding residues in protein–RNA complexes:

- **RNABindRPlus** (*see* Subheading 3.2)—a hybrid machine learning/sequence homology-based approach developed by our group [57] which requires only sequence information for the protein(s) of interest. The accuracy of this and similar sequence-based methods from other groups is generally greater than that obtained using structure-based methods.

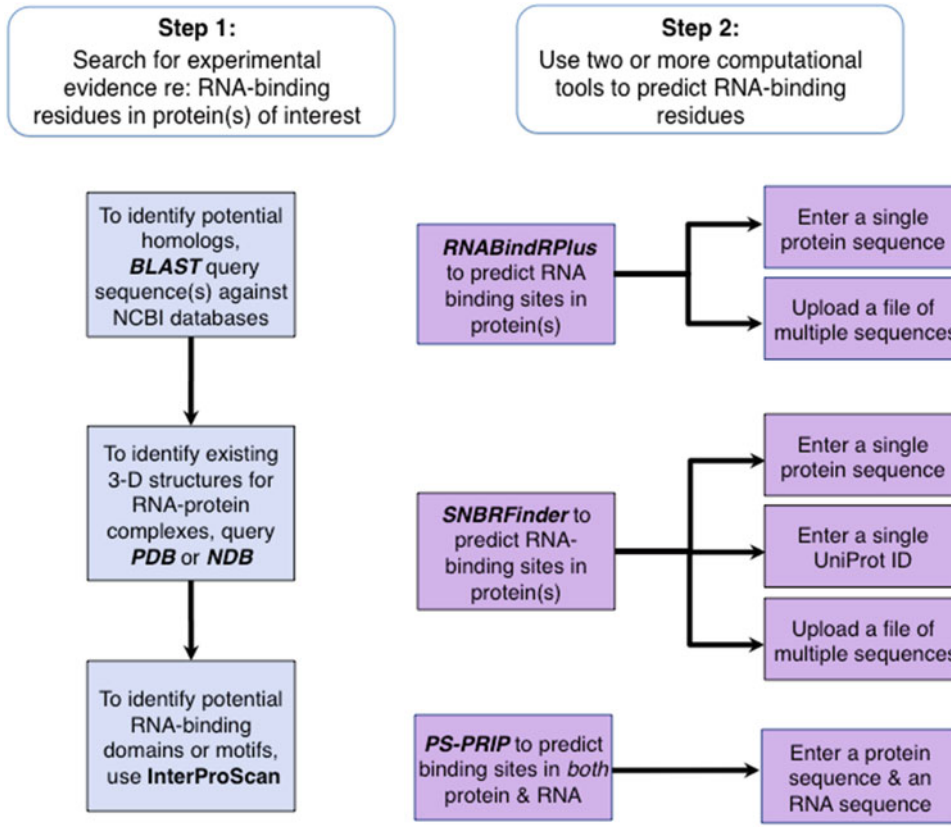


Fig. 1 Flowchart for identifying potential RNA-binding residues in proteins

- **SNBRFinder** (*see* Subheading 3.3)—a method developed by Yang et al. [58], which can predict either RNA- or DNA-binding residues in proteins by combining a machine learning method with a template (homology)-based method. The key differences between SNBRFinder and RNABindRPlus are: (a) inputs to the SVM classifier in SNBRFinder include sequence profiles and other sequence descriptors such as residue conservation scores, physicochemical properties, and interface propensities, whereas the only inputs to the SVM for RNABindRPlus are sequence PSSMs; (b) SNBRFinder uses profile hidden Markov models to find remote homologs for the query protein, whereas RNABindRPlus uses BLAST searches.
- **PS-PRIP** (*see* Subheading 3.4)—a new motif-based method developed by our group [52], which can predict interfacial residues in both the protein and the RNA components of a protein–RNA complex and can provide "partner-specific" predictions.

Table 1
Databases of protein–RNA complexes and resources for analyzing interfaces and motifs in protein–RNA complexes

Database	Description	Reference
Databases of structures of RNA–protein complexes		
PDB (Protein Data Bank)	www.pdb.org This is a database of 3D macromolecular structures—protein–protein, protein–DNA, protein–RNA, and protein–ligand structures solved using X-ray crystallography, cryo-EM, NMR, and others	[68]
NDB (Nucleic Acid Database)	http://ndbserver.rutgers.edu/ This is a database of three-dimensional structural information for nucleic acids	[69]
PDBSum	https://www.ebi.ac.uk/thornton-srv/databases/cgi-bin/pdbsum/GetPage.pl?pdbcode=index.html A pictorial database of PDB structures that provides access to interfacial residues in known structures	[70, 71]
Resources for analyzing interfaces and RNA-binding motifs in RNA		
BIPA (Biological Interaction Database for Protein–Nucleic Acid)	http://mordred.bioc.cam.ac.uk/bipa BIPA provides a list of protein–RNA (and protein–DNA) complexes from the PDB and displays RNA binding residues within the linear primary sequence of a chosen protein, or within a multiple sequence alignment of related RNA binding proteins (BIPA has not been updated since 2009 and is not fully functional at present)	[72]
InterPro & InterProScan	http://www.ebi.ac.uk/interpro/ InterPro classifies protein sequences into families using information from ten different databases; InterProScan identifies functional and/or conserved domains, motifs, and other important sites in protein sequences	[51, 73]
NPIDB (Nucleic Acid-Protein Interaction Database)	http://npidb.belozersky.msu.ru/ A database for extracting biologically meaningful characteristics of protein–RNA and protein–DNA complexes	[74]
DBBP (DataBase of Binding Pairs in protein–nucleic acid interactions)	http://bclab.inha.ac.kr/dbbp A database that provides structural data for hydrogen bonding interactions between proteins and nucleic acids	[75]
PRIDB (Protein RNA interface database)	http://pridb.gdcb.iastate.edu A database of protein–RNA complexes from the PDB, with tools for identifying and visualizing interfacial residues in both the protein and RNA sequences and structures. (PRIDB has not been updated since 2013 and is under remediation)	[76]

(continued)

Table 1
(continued)

Database	Description	Reference
RsiteDB	http://bioinfo3d.cs.tau.ac.il/RsiteDB/ This database stores information about the protein binding pockets that interact with single-stranded RNA nucleotide bases	[77]
ProNIT	http://www.abren.net/pronit/ A database of thermodynamic interaction data (binding constants, free energy change, and so on) between proteins and nucleic acids	[78]
RNA CoSSMos	http://cossmos.slu.edu/ A tool that provides information on secondary structural motifs such as bulges and hairpin loops of 3D protein–nucleic acid structures	[79]
RNA 3D Hub	http://rna.bgsu.edu/rna3dhub/ A suite of tools including the RNA Structure Atlas and RNA 3D Motif Atlas. These provide information about RNA 3D motifs	[80]
RNA Bricks	http://iimcb.genesilico.pl/rnabricks A database that provides information about recurrent RNA 3D motifs and their interactions, extracted from experimentally determined structures of RNA and RNA-protein complexes	[81]
Databases of recognition sites/protein-binding motifs in RNA		
CISBP-RNA	http://cisbp-rna.ccbr.utoronto.ca/ A database of inferred sequence binding preferences of RNA-binding proteins	[22]
RBPDB	http://rbpdb.ccbr.utoronto.ca/ A database of manually curated RNA-binding sites collected from literature	[41]

We encourage users to submit their proteins of interest to all three web servers described in this protocol because the underlying algorithms and datasets used for training and evaluating performance are different in each case, and the methods have different strengths and weaknesses. Even though all three methods have been shown to provide highly reliable predictions on benchmark datasets, it is not possible to guarantee an accurate prediction for any specific RNA-binding protein with any of these methods.

2 Materials

2.1 Databases of Experimentally Validated Protein–RNA Complexes and Resources for Analyzing Interfaces

Before using computational methods to *predict* RNA-binding residues, the user should first search for existing experimental data regarding interfacial residues in the specific RNA-binding protein(s) of interest, both in published literature and in relevant specialized databases. The "gold standard" for identifying RNA-binding residues in proteins is analysis of a high resolution three-dimensional structure of the protein bound to its cognate RNA, i.e., a "bound" structure of the complex containing the protein bound to RNA. The Protein Data Bank (PDB) [68] and the Nucleic Acid Database (NDB) [69] are two comprehensive databases of experimentally determined structures, from which residue and atomic-level information regarding the interfaces in macromolecular complexes can be extracted. Table 1 provides URLs for these two primary databases, followed by an alphabetical listing of several databases that contain valuable information about protein–RNA complexes and their interfacial residues, either derived from structures in the PDB/NDB or from other types of experiments. A suggested strategy for utilizing selected resources from this list is provided in Subheading 3.1 below.

2.2 Servers and Software for Predicting Interfaces in Protein–RNA Complexes

There are more than 20 published approaches for predicting RNA-binding residues in proteins (for a recent compilation, see [50]), and a few methods are capable of predicting interfacial residues in both the protein and the RNA components of a protein–RNA complex (e.g., [52, 82]). Subheadings 3.2–3.5 below focus on three methods (RNABindRPlus, SNBRFinder, PS-PRIP) that are freely available on web-based servers and have been shown to perform well on benchmark datasets. Table 2 lists these and several additional methods. Please note that not all of these are currently available as web-based servers.

2.3 The RNABindRPlus Server

RNABindRPlus [57] is a purely sequence-based method for predicting RNA-binding residues in putative RNA-binding proteins. It uses logistic regression to combine predictions from HomPRIP, a sequence homology-based method, with predictions from SVMOpt, an optimized Support Vector Machine (SVM) classifier. The SVM classifier utilizes sequence-based PSSMs as features. HomPRIP makes highly accurate predictions of RNA-binding residues when homologs (with solved structures) of the query protein can be found, but a major drawback is that no predictions are returned when no such homologs can be found. Additionally, HomPRIP cannot return predictions for parts of the query protein sequence that are not aligned with its homologs. This limitation of HomPRIP is overcome by combining it with a machine learning-based method, SVMOpt, which returns predictions for every residue in any protein sequence.

Table 2
Servers and software for predicting RNA-binding sites in proteins

Method	Description	Reference
BindN	http://bioinfo.ggc.org/bindn/ An SVM classifier that uses hydrophobicity, side chain pKa, molecular mass, and PSSMs for predicting RNA-binding residues; it can also predict DNA-binding residues	[83]
BindN+	http://bioinfo.gcc.org/bindn An updated version of BindN, that uses an SVM classifier based on PSSMs and several other descriptors of evolutionary information; it can also predict DNA-binding residues	[84]
catRAPID signature	http://s.tartaglialab.com/grant_submission/signature Predicts both RNA-binding and protein-binding residues in RNPs based on physicochemical features instead of sequence similarity searches	[82]
DR_bind1	http://drbind.limlab.ibms.sinica.edu.tw/ Predicts RNA-binding residues in proteins using information derived from 3D structure	
DRNA	http://sparks-lab.org/yueyang/DFIRE/dRNA-DB-service.php Predicts RNA-binding proteins and RNA-binding sites based on similarity to known structures	[85]
KYG	http://cib.cf.ocha.ac.jp/KYG Uses a set of scores based on the RNA-binding propensity of individual and pairs of surface residues of the protein, used alone or in combination with position-specific multiple sequence profiles	[86]
Meta predictor	http://iimcb.genesilico.pl/meta2/ A predictor that combines the output of PiRaNhA, PPRInt, and BindN+ to make predictions of RNA-binding residues using a weighted mean. (Not available as of March 2014)	[31]
NAPS	http://prediction.bioengr.uci.edu A modified C4.5 decision tree algorithm that uses amino acid identity, residue charge, and PSSMs to predict residues involved in DNA- or RNA-binding. (Not available as of March 2014)	[87]
OPRA	Uses path energy scores calculated using interface propensity scores weighted by the accessible surface area of a residue to predict RNA-binding sites. Available from the authors upon request	[88]
PPRInt	http://www.imtech.res.in/raghava/pprint/ An SVM classifier trained on PSSM profiles to predict RNA-binding residues	[89]
PS-PRIP	http://pridb.gdcb.iastate.edu/PSPRIP/ A partner-specific method for predicting RNA-binding residues in proteins and protein-binding residues in RNAs using sequence motifs extracted from interfacial regions in RNA-protein complexes	[52]
PRBR	http://www.cbi.seu.edu.cn/PRBR/ An enriched random forest classifier trained on predicted secondary structure, a combination of PSSMs with physic-chemical properties, a polarity-charge correlation, and a hydrophobicity correlation	[90]

(continued)

Table 2
(continued)

Method	Description	Reference
PRIP	Uses an SVM classifier and a combination of PSSM profiles, solvent accessible surface area, betweenness centrality, and retention coefficient as input features. Not accessible via a web server, but results can be obtained via correspondence with the author	[91]
RBScore	http://ahsoka.u-strasbg.fr/rbscore/ Utilizes a score derived from physicochemical and evolutionary features, integrating a residue neighboring network approach; it predicts both DNA- and RNA-binding residues in proteins	[92]
RISP	http://grc.seu.edu.cn/RISP An SVM-based method that uses evolutionary information in terms of PSSMs (Not available as of March 2014)	[93]
RNABindR	http://bindr.gdcb.iastate.edu/RNABindR/ A Naïve Bayes classifier that uses the amino acid sequence identity to predict RNA-binding residues in proteins (no longer maintained)	[94]
RNABindR v2.0	http://ailab1.ist.psu.edu/RNABindR/ An SVM classifier that uses sequence PSSMs to predict RNA-binding residues in proteins	[47]
RNABindRPlus	http://ailab1.ist.psu.edu/RNABindRPlus/ A predictor that combines an optimized SVM classifier with a sequence homology-based method to predict RNA-binding residues in proteins	[57]
RNApin	http://www.imtech.res.in/raghava/rnapin/ An SVM classifier that predicts protein-interacting nucleotides (PINs) in RNA	[61]
SNBRFinder	http://ibi.hzau.edu.cn/SNBRFinder/ A sequence-based hybrid predictor that combines a feature-based predictor and a template-based predictor to predict nucleic-acid binding residues in proteins	[95]
SPOT-Seq-RNA	http://sparks-lab.org/yueyang/server/SPOT-Seq-RNA/ A template-based technique for predicting RBPs, RNA-binding residues and complex structures	[95]

RNABindRPlus was trained on the RB198 dataset, and tested on two different datasets, RB44 and RB111. On a subset of proteins for which homologs with experimentally determined interfaces could be reliably identified, HomPRIP outperformed all other methods, achieving an MCC of 0.63 on RB44 and 0.83 on RB111. RNABindRPlus was able to predict RNA-binding residues of all proteins in both test sets, achieving an MCC of 0.55 on RB44 and 0.37 on RB111, and outperforming all other methods, including structure-based methods (e.g., KYG [86]).

2.4 The SNBRFinder Server

SNBRFinder is a sequence-based predictor that combines predictions from a Support Vector Machine (SVM) classifier, SNBRFinder^{F}, with predictions from a template-based classifier, SNBRFinder^{T}.

SNBRFinder^{F} utilizes a sliding window of the target residues and five neighboring residues on each side to represent the sequential environment. The features used as inputs to the classifier include the sequence profile, residue conservation scores, predicted structural features, physicochemical properties, interface propensity, sequential position, and two global features, sequence length and the global amino acid composition.

SNBRFinder^{T} is a template-based method, i.e., a method that utilizes sequence or structural alignments to retrieve homologs/templates of a query protein and then infer binding residue information for the query protein. SNBRFinder^{T} uses the HHblits program [96] to identify homologs of the query protein. HHblits represents both the query and database sequences using profile hidden Markov models (HMM), and then compares the two to identify homologs of the query protein. For each query and homolog pair, a probability score is output for evaluating the similarity between the aligned HMMs. The higher the score is, the better the alignment is and vice versa. Specifically, a residue in the query protein is predicted to be RNA-binding with a probability score of 1 if it is matched with a binding residue in the homolog, otherwise the residue is predicted to be non RNA-binding with a probability score of 0.

On the RB44 [31] dataset, SNBRFinder had an MCC of 0.48, whereas RNABindRPlus had an MCC of 0.49. In terms of AUC values, SNBRFinder and RNABindRPlus achieved very similar results, with both getting 0.84.

2.5 The PS-PRIP Server

PS-PRIP [52] is a motif-based method that predicts interfacial residues for both the RNA and protein components of protein–RNA complexes in a partner-specific manner (*see* **Note 4**). PS-PRIP requires as input the sequences of both the RNA-binding protein and its putative bound RNA(s). Although no structural information is required, PS-PRIP exploits the co-occurrence of specific pairs of short protein and RNA sequence motifs (5 amino acids long and 5 ribonucleotides long) from a database of motifs extracted from interfaces in known protein–RNA complexes from the PDB. On an independent dataset of 327 RNA-protein pairs, PS-PRIP obtained a sensitivity of 0.64, precision of 0.80, and MCC of 0.59 compared to RNABindRPlus with values of 0.88, 0.76, and 0.71, respectively. In addition to providing predicted RNA-binding residues in proteins, PS-PRIP makes predictions of protein-binding residues in RNAs, although with much lower accuracy. Other methods designed to predict protein-binding residues in RNA have been published recently (e.g., [61, 82]).

3 Methods

3.1 Searching Existing Literature and Databases for Relevant Experimental Data

Currently, all computational methods for predicting RNA-binding residues in proteins return only *predicted* interfacial residues, even when the actual interfaces are known from experimental data. Thus, before using software to predict potential RNA-binding residues, the user should search published literature and existing databases for experimentally identified interactions involving the protein of interest (*see* **Note 5**). If the query protein is newly discovered or has no known function, the user should first search for potential homologs using a BLAST search. As outlined below, both the original query sequence and its homologs can be used to search databases of known protein–RNA interactions, such as those listed in Table 1.

1. If the query protein sequence corresponds to an "unknown" or novel protein, run the sequence through **NCBI's BLAST server**, available at http://blast.ncbi.nlm.nih.gov/Blast.cgi [97, 98] or use similar genomics resources elsewhere (e.g., http://www.ebi.ac.uk/Tools/sss/). BLAST (Basic Local Alignment Search Tool) finds highly similar sequences in the NCBI or ENSEMBL databases (*see* **Note 6**). A good starting point for most protein sequence searches is SMARTBLAST, available here: http://blast.ncbi.nlm.nih.gov/smartblast/ (*see* **Note 7**). If the query sequence itself is not available in one of the NCBI or ENSEMBL databases, potential homologs identified by BLAST can be used as the query for subsequent searches in the databases listed in **steps 2–6** below (*see* **Note 8**).
2. If the query protein has been previously identified and/or analyzed, a search using the **NCBI "Protein"** tool may quickly reveal previously annotated RNA-binding domains or motifs and links to experimentally determined structures. Enter the name of the protein (or name of a potential homolog, identified in **step 1**) into the box provided here: (http://www.ncbi.nlm.nih.gov/protein/). In the list of "Items" returned, click on the protein name from the appropriate organism to access the full GenBank protein entry. Then, examine information on the right side of the GenBank protein page; for example, if a high resolution structure is available, it will appear under the "Protein 3D Structure" header. Under the "Related Information" header, click on "Conserved Domains (Concise)" or "Conserved Domains (Full)" to access any annotated RNA-binding domains (or other conserved domains) identified in the protein sequence. The "Conserved Domains" results page also provides links to available three-dimensional structure(s) similar to that of the query protein, if available. Other links on this page can lead to additional information regarding potential RNA-binding domains in the protein of interest (*see* **Note 9**).

3. In every case, the user should search the **Protein Data Bank (PDB),** available at www.rcsb.org [68] for any available structures of protein–RNA complexes that contain the protein of interest. The PDB contains over 1600 three-dimensional structures of protein–RNA complexes determined using experiments such as X-ray crystallography, nuclear magnetic resonance (NMR) imaging, and cryo-electron microscopy. The PDB has a powerful search engine that allows the database to be queried in a variety of ways, e.g., by protein (or RNA) name, sequence, or GO terms. The PDB also provides excellent structure visualization tools as well as links to valuable third-party resources for visualizing and analyzing the structures of macromolecules (*see* **Note 10**).
4. Similarly, the **Nucleic Acid Database (NDB**), available at http://ndbserver.rutgers.edu [69], is another valuable resource that focuses on experimentally determined three-dimensional structures of nucleic acids, including both protein–RNA and protein–DNA complexes. The NDB contains only a subset of structures in the PDB, making it easier for the user to focus on structures that contain RNA. Also, the NDB provides convenient access to a wide variety of tools and software specifically designed for analyzing RNA sequences and structures (*see* **Note 11**).
5. If it is possible to identify a structure for the query protein–RNA complex (or a homologous complex) in one of the previous steps, the user can quickly obtain a graphical representation of the protein–RNA interface, using **PDBSum** [70, 71] available at: https://www.ebi.ac.uk/thornton-srv/databases/cgi-bin/pdbsum/GetPage.pl?pdbcode=index.html. Enter the 4-letter PDB code in the box provided and click "Find." At the top of the PDBSum entry page that appears, click on the "DNA/RNA" link to access a page listing all of the nucleic acid chains in the complex. Then click on "**NUCPLOT**" to visualize the ribonucleotides that are contacted by individual amino acids, as well as additional information (backbone *vs.* phosphate group contacts, hydrogen bonding, etc.). Another way to identify the RNA-binding amino acids is to click on the "Protein" link at the top of the page to reveal a diagrammatic representation of the protein sequence, in which Residue Contacts to DNA/RNA are labeled. Tools for visualizing, analyzing and manipulating the structure are provided by both the PDB and NDB (*see* **Notes 10** and **11**). *See* Table 1 for additional tools that provide detailed information about the interfacial residues (e.g., NPIDB [74], DBBP [75]).
6. If no structure for the query protein–RNA complex can be identified, the user can **search for known RNA-binding domains or motifs in the protein sequence.** Typically, only a few of the amino acids in well-characterized RNA-binding

domains or motifs (e.g., the RNA Recognition Motif (RMM), which is ~90 amino acids) are actually "interfacial residues" involved in contacting RNA (*see* **Note 1**). But, if the query protein does contain such a conserved domain or motif, homologous structures are likely available and can indicate which amino acids are directly involved in recognizing and binding RNA. The EMBL-EBI's **InterPro** [51] is a valuable comprehensive resource that includes more than 10 databases of protein structural and functional motifs, and an integrated tool, **InterProScan** [73], which can be used to identify all known motifs, including RNA-binding motifs, in a protein of interest. Access InterPro here: http://www.ebi.ac.uk/interpro/ and enter the query protein sequence in the text box. Within a few minutes, a "Results" page will appear, providing a graphical summary of all domains, motifs and signatures identified, with links to additional information about each.

7. For many RNA-binding proteins, recognition motifs (i.e., the specific RNA sequences bound by the RBP) are now known [1, 22, 99]. Several valuable databases and tools are available if the user wishes to identify known or potential recognition sites in the RNA component of a specific protein–RNA complex. Databases of experimentally defined RNA sequence motifs that are bound by RBPs include: CISBP-RNA [22], RBPDB, [41], and RBPMotif [43]. Databases of RNA structural motifs, e.g., BRICKS [81] and the RNA 3D Motif Atlas [80], are also available, but these have not yet been systematically annotated regarding their protein-binding activities. Also, a valuable tool for mapping binding sites for RBPs within the genomes of several model organisms is RBPMap [100], which is available at: http://rbpmap.technion.ac.il.

3.2 Using RNABindRPlus to Predict RNA-Binding Residues in Proteins

The **RNABindRPlus** method implements a combination of a machine learning method (**SVMOpt**) and a sequence homology-based method (**HomPRIP**) to predict RNA-binding residues in proteins [57] (*see* Subheading 2.3). Given a single protein sequence (or a file of multiple protein sequences), RNABindRPlus can predict which amino acid residues are mostly likely to bind RNA. Run times can be slow when large numbers of protein sequences are submitted in a single job (*see* **Note 12**). A faster version of the server is under development (*see* **Note 13**).

1. Access the **RNABindRPlus** web server at: http://ailab1.ist.psu.edu/RNABindRPlus/.
2. **To predict RNA-binding residues in a single putative RNA-binding protein**: Enter the protein sequence in FASTA format (*see* **Note 14**) in the text box provided on the homepage.

3. **To predict RNA-binding residues for multiple putative RNA-binding proteins**: In this case, the user has two options: (a) Enter the protein sequences in FASTA format in the text box provided; or (b) upload a FASTA formatted file of protein sequences by clicking the "Choose file" button on the homepage.
4. Provide an email address where results should be sent. Computing the results requires approximately 10 min per protein sequence submitted to RNABindRPlus (*see* **Notes 12** and **13**).
5. The user has the option of excluding highly similar proteins from the homolog list, at the desired sequence identity level by selecting the check box at the bottom of the submission page. To obtain the most reliable predictions, leave this option blank (*see* **Note 15**).
6. Once all submission fields have been filled, click on the "Submit" button. The user will receive an email confirming that the job is currently running. RNABindRPlus results will be returned to the user by email.
7. Figure 2 shows results returned by RNABindRPlus for the S5 protein from the 30S ribosomal subunit of *T. thermophilus*, which corresponds to protein chain E, in PDB structure 1HNX). Figure 2a shows the *Results Summary* email, which contains several links that can be clicked to display selected portions of the results. Figure 2b (*Interface Prediction Results)* displays predictions from three different methods: HomPRIP (homology-based method), SVMOpt (optimized SVM) and RNABindRPlus (which combines predictions from HomPRIP and SVMOpt). The first section of output for each method (e.g., Prediction from HomPRIP), is a list of the predictions for each residue, where "1" corresponds to predicted interfacial residues (i.e., RNA-binding) and "0" corresponds to predicted non-interfacial residues. The second section of output (e.g., "Predicted score from HomPRIP") gives the probability score for each residue (where a probability of ≥ 0.5 means the residue is an interface residue, otherwise it is a non-interface residue). Figure 2c (*Homologs of the query protein*) displays a list of homologous proteins identified by HomPRIP, the homology-based component of RNABindRPlus, along with their corresponding interface conservation scores (IC_scores) (*see* **Note 16**). These are the homologous proteins used for inferring RNA-binding residues in the query protein using HomPRIP. Figure 2d (*All potential homologs in the PDB*) shows only a portion of the output providing information about all potential homologs found in the PDB for the query protein.

3.3 Using SNBRFinder to Predict RNA-Binding Residues in Proteins

SNBRFinder is a sequence-based hybrid predictor that combines predictions from a Support Vector Machine method, SNBRFinder^{F}, with predictions from a template-based method, SNBRFinder^{T} [58] (*see* Subheading 2.4). The inputs to the SVM method include

a

RNABindRPlus Prediction Results for 1HNX

Inbox x

rnabindr_plus@iastate.edu Dec 14 (4 days ago)

to me

Thank you for using RNABindRPlus.

The results for your job *1HNX* are now available.

The interface prediction results can be downloaded here.

The homologs of your query protein(s) and their corresponding IC_scores can be downloaded here.

All potential homologs that exist in the Protein Data Bank (PDB) protein-RNA complexes and their sequence similarity to the query proteins can be downloaded here.

.

If you have questions regarding RNABindRPlus, please visit the corresponding web page(s) or write to rnabindr_plus@iastate.edu.

b

RNABindRPlus: Honavar (PSU) & Dobbs (ISU) Laboratories

```
    #Input sequence length: 162
#Number of binding residues predicted by HomPRIP: 50
#Number of binding residues predicted by SVM: 52
#Number of binding residues predicted by RNABindRPlus: 50
>1HNX_E
sequence:
M,P,E,T,D,F,E,E,K,M,I,L,I,R,R,T,A,R,M,Q,A,G,G,R,R,F,R,F,G,A,L,V,V,V,G,D,R,Q,G,R,V,G,L,G,F,G,K,A,P,E,V,P,L,A,
V,Q,K,A,G,Y,Y,A,R,R,N,M,V,E,V,P,L,Q,N,G,T,I,P,H,E,I,E,V,E,F,G,A,S,K,I,V,L,K,P,A,A,P,G,T,G,V,I,A,G,A,V,P,R,A,
I,L,E,L,A,G,V,T,D,I,L,T,K,E,L,G,S,R,N,P,I,N,I,A,Y,A,T,M,E,A,L,R,Q,L,R,T,K,A,D,V,E,R,L,R,K,G,E,A,H,A,Q,A,Q,G
Prediction from HomPRIP:
?,?,?,?,0,0,0,0,0,0,0,0,1,0,1,1,1,1,1,1,1,1,1,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,1,0,0,0,1,0,
0,0,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,0,0,1,0,0,1,1,1,1,1,0,0,1,0,
0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,0,1,1,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,?,?,?,?,?,?,?
Predicted score from HomPRIP:
?,?,?,?,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,1.00,0.00,1.00,1.00,1.00,1.00,1.00,1.00,1.00,1.00,1.00,
1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,1.00,0.0
0,1.00,1.00,1.00,0.00,0.00,0.00,0.50,0.00,0.00,0.00,1.00,0.00,0.00,1.00,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0
.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.50,1.00,1.00,1.00,1.00,0.50,0.00
,0.50,0.00,1.00,1.00,1.00,1.00,0.00,0.00,1.00,0.00,0.00,1.00,1.00,1.00,1.00,0.00,0.00,1.00,0.00,0.00,0.00,0.
00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,1.00,1.00,1.00,1.00,1.00,1.00,1.00,1.00,1.00,0.25,1.00,1.00,0.00,0.00,
1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.0
0,0.00,?,?,?,?,?,?,?
Prediction from SVM:
0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,1,1,1,0,0,0,1,0,
0,0,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,1,0,1,1,1,1,1,0,1,0,0,1,1,1,0,0,0,1,0,
0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0
Predicted score from SVM:
0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.01,0.01,0.01,1.00,1.00,1.00,1.00,1.00,1.00,1.00,1.00,1.0
0,1.00,1.00,1.00,0.13,1.00,0.31,0.05,0.04,0.05,0.01,0.03,0.01,0.03,0.02,0.32,0.43,0.08,0.84,0.10,0.08,0.04,0
.12,1.00,0.12,1.00,1.00,0.99,0.02,0.00,0.06,1.00,0.07,0.03,0.07,1.00,0.01,0.00,0.91,1.00,0.00,0.01,0.01,0.00
,0.01,0.00,0.01,0.00,0.01,0.00,0.01,0.01,0.06,0.13,0.02,0.03,0.07,0.02,0.00,0.00,0.02,0.99,0.97,0.99,0.98,0.
97,0.04,0.01,0.98,0.00,1.00,0.99,0.96,0.98,0.99,0.01,1.00,0.01,0.02,1.00,1.00,1.00,0.09,0.01,0.01,1.00,0.00,
0.00,0.00,0.01,0.01,0.00,0.01,0.00,0.01,0.10,0.01,1.00,1.00,1.00,1.00,0.91,1.00,1.00,1.00,1.00,1.00,1.0
0,0.04,0.01,1.00,0.05,0.03,0.00,0.01,0.01,0.00,0.01,0.01,0.00,0.01,0.01,0.00,0.00,0.01,0.01,0.01,0.01,0.07,0
.48,0.06,0.55,0.02,0.00,0.04,0.00,0.00,0.00,0.00,0.00
Prediction from RNABindRPlus:
0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,1,0,0,0,1,0,
0,0,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,1,0,1,0,0,1,0,1,1,1,0,0,0,1,0,
0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Predicted score from RNABindRPlus:
0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.98,0.53,0.98,0.98,0.98,0.98,0.98,0.98,0.9
8,0.98,0.98,0.98,0.01,0.98,0.03,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.03,0.05,0.01,0.32,0.01,0.01,0.01,0
.01,0.98,0.01,0.98,0.98,0.98,0.01,0.01,0.01,0.88,0.01,0.01,0.01,0.98,0.01,0.01,0.96,0.98,0.01,0.01,0.01,0.01
,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.15,0.01,0.87,0.97,0.98,0.97,0.
97,0.03,0.01,0.86,0.01,0.98,0.98,0.97,0.97,0.52,0.01,0.98,0.01,0.01,0.98,0.98,0.98,0.22,0.01,0.01,0.98,0.01,
0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.98,0.98,0.98,0.98,0.98,0.96,0.98,0.98,0.98,0.74,0.98,0.9
8,0.01,0.01,0.98,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0
.06,0.01,0.09,0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01
```

Fig. 2 (**a**) RNABindRPlus results notification email obtained for the *T. thermophilus* S5 protein. (**b**) RNABindRPlus prediction results for the *T. thermophilus* S5 protein. Results are also returned for the two individual components of RNABindRPlus, HomPRIP and SVMOpt. For each method, under the header "Prediction from," the predicted RNA-binding residues are represented by a string of 1's and 0's, where "1" and "0" correspond to predicted RNA-binding and non-RNA binding residues, respectively. See text for additional details.

c

```
Homologs of: 1HNX_E
3pyuE   0.86
3pynE   0.86
3pyqE   0.86
3pysE   0.86
```

d

```
#num_residue1: the length of the seq.
#num_residue2: the number of resi that have (non)int information.
#>QUERY PDBID + CHAINID
#HOMOLOG-PDBID+CHAINID  num_residue1    num_residue2    num_int Bit_score       Evalue  Positive_Score
IdentityScore   alignment_length        aligLen_Query   aligLen_Homolog
>1HNX_E 162
3knjE   162     150     50      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
3uxsE   162     148     50      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
3i9bH   162     154     49      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
3uyfH   162     151     51      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
3i8gH   162     151     54      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
4g5mH   162     151     51      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
3uydH   162     151     46      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
2v48E   162     150     52      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
2uxbE   162     150     50      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
1fjgE   162     150     47      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
3ohyE   162     150     50      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
2wh3E   162     150     49      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
3uxtE   162     148     51      322     4e-115  100     100     162     0.993827160493827       0.993827160493827
```

Fig. 2 (continued) (**c**) List of homologs and IC scores obtained by RNABindRPlus. These are the homologs used by HomPRIP for making the homology-based predictions. (**d**) List of all potential homologs with structures in the PDB for *T. thermophilus* S5 protein identified by RNABindRPlus. num_residue1 (e.g., 162) denotes the number of amino acids in the query protein; num_residue2 shows the number of amino acids (e.g., 150) in the homolog of the query protein (e.g., 3KNJ, chain E); num_int is the number of binding residues (e.g., 50) in the homolog of the query protein; Bit_score (e.g., 322) gives an indication of the quality of the alignment between the query protein and its homolog—the higher the score, the better the alignment; Evalue is the number of hits expected by chance when searching the database of homologous proteins—the lower the Evalue, the more significant a match to a database sequence is; Positive_Score gives an indication of how many amino acids in the query protein were at least similar to the amino acid sequences found in the database; IdentityScore gives an indication of how many exact matches the query protein had with amino acid sequences in the database; alignment_length is an indication of the number of residues in the query protein aligned with homologs from the database; aligLen_Query is the alignment_length divided by the length of the query protein; aligLen_Homolog is the alignment_length divided by the length of the homolog of the query protein

sequence profiles and other sequence descriptors, such as residue conservation scores, physicochemical properties, and interface propensities. SNBRFinder^{T} uses profile hidden Markov models to find remote homologs of the query protein sequence, but the basic methodology used for building the classifier is similar to that used in RNABindRPlus.

1. Access the SNBRFinder web server at http://ibi.hzau.edu.cn/SNBRFinder/index.php.
2. Use the radio buttons provided to choose one of three different options for submitting a protein sequence: (a) enter the amino acid sequence in FASTA format; (b) upload a protein sequence file by clicking on "Browse File"; or (c) input UniProt IDs for retrieval (*see* **Note 17**).
3. The user has the option of filtering out proteins homologous to the query protein sequence by specifying a sequence identity threshold. By default, the method excludes homologous templates that share ≥30% sequence identity. To obtain the most reliable predictions, leave this option blank (*see* **Note 18**).

4. Because SNBRFinder can predict either RNA- or DNA-binding residues in proteins, the user should select the binding nucleic acid type (RNA) from a drop-down list. By default, the selection is "DNA."
5. Before clicking on the "submit" button, the user can optionally enter an email address. After the job is submitted, a webpage showing the job id and indicating that the job is running should appear. This page also includes the URL where prediction results will be posted, after they become available. If an email address was provided, the URL will also be included in the email. Typically, results are returned to users after about 15 min.
6. Figure 3 shows results returned by SNBRFinder for the S5 protein from the 30S ribosomal subunit of *T. thermophilus*, which corresponds to protein chain E, in PDB structure 1HNX. Figure 3a shows a summary of the results, in which the query sequence is

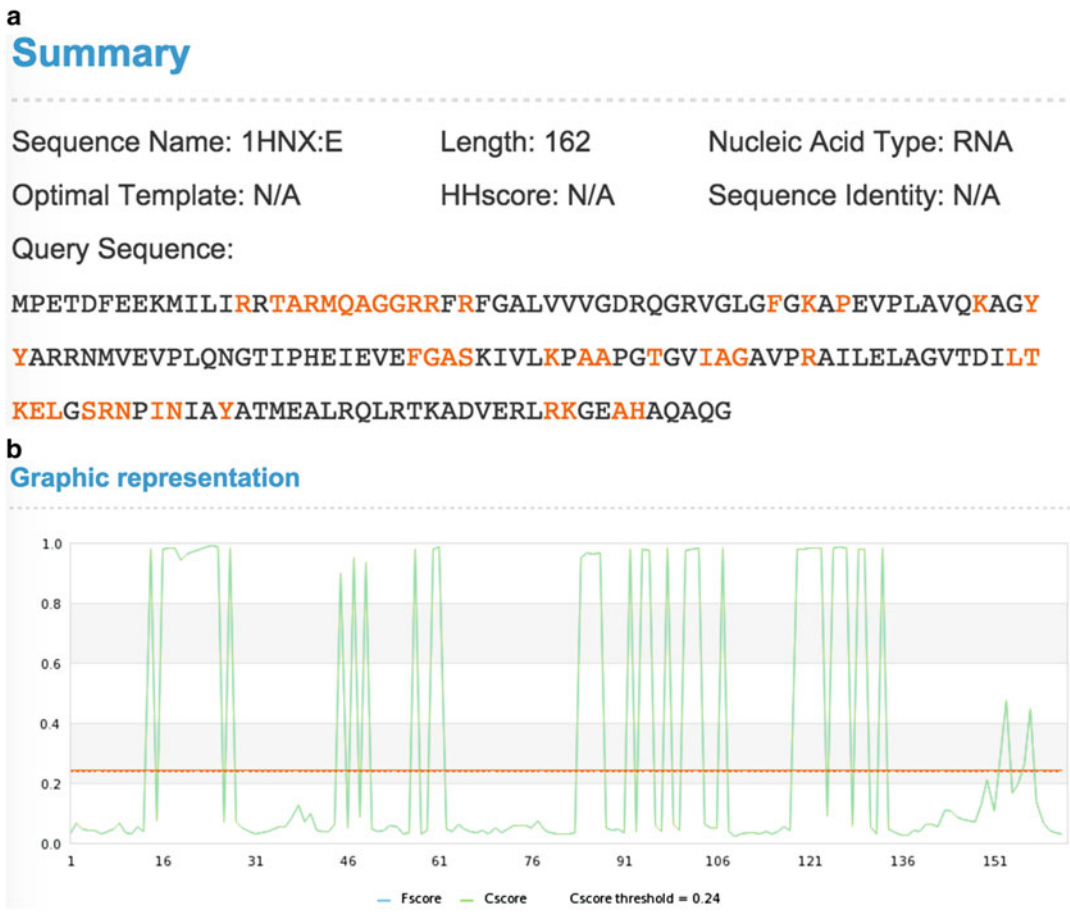


Fig. 3 (**a**) SNBRFinder prediction results summary for the *T. thermophilus* S5 protein. Predicted RNA-binding residues are shown in red. (**b**) Graphical representation of SNBRFinder predictions for the *T. thermophilus* S5 protein. Fscore is the prediction score returned by the feature-based component, SNBRFinder^{F}, and Cscore is the prediction score returned by the combination of the feature-based component and homology/template-based component, SNBRFinder^{T}, of SNBRFinder.

c

Details about prediction results

Position	AA	Fscore	Tscore	Cscore	Tag
1	M	0.031	-	0.031	-
2	P	0.069	-	0.069	-
3	E	0.047	-	0.047	-
4	T	0.044	-	0.044	-
5	D	0.044	-	0.044	-
6	F	0.032	-	0.032	-
7	E	0.041	-	0.041	-
8	E	0.050	-	0.050	-
9	K	0.069	-	0.069	-
10	M	0.038	-	0.038	-

d

```
Sequence Name: 1HNX:E   Length: 162     Nucleic Acid
Type: RNA       Optimal Template: N/A   HHscore: N/A
Sequence Identity: N/A
Pos       AA        Fscore    Tscore    Cscore    Tag
1         M         0.031     -         0.031     -
2         P         0.069     -         0.069     -
3         E         0.047     -         0.047     -
4         T         0.044     -         0.044     -
5         D         0.044     -         0.044     -
6         F         0.032     -         0.032     -
7         E         0.041     -         0.041     -
8         E         0.050     -         0.050     -
9         K         0.069     -         0.069     -
10        M         0.038     -         0.038     -
```

Fig. 3 (continued) (**c**) Table showing SNBRFinder a sample of the detailed results for the *T. thermophilus* S5 protein. See text for additional details. (**d**) Downloadable results from SNBRFinder. Only a portion of the returned results is shown

displayed with predicted interfacial residues highlighted in red text; the query sequence name, length, nucleic acid type, as well as the PDB ID of the optimal template used for making the prediction, the HHscore, if any (*see* **Note 19**), and the % sequence identity (between the query and the optimal template) are also provided. For this example, SNBRFinder was not able to find an optimal template, so HHscore and sequence identity have a value

of N/A. Figure 3b shows a graphical representation of the results, which displays a plot of the Fscore and Cscore for each residue, and the Cscore threshold above which a residue is considered an interfacial residue (*see* **Note 20**). Because no optimal template was found for 1HNX chain E, the Fscore is equivalent to the Cscore. Figure 3c shows a detailed results table, which lists each amino acid residue, along with its associated Fscore, Tscore (if any), and Cscore, as well as the "tag" for each amino acid ("+" for interfacial residue, "-" for non-interfacial residue). Figure 3d shows a portion of the results in plain text format, which can be obtained by clicking the "Download the result" link in the top right corner of the "Result" page.

3.4 Using PS-PRIP to Predict Both RNA-Binding and Protein-Binding Residues in RNPs

PS-PRIP (Partner-Specific protein–RNA Interface Prediction) is a sequence motif-based method that can simultaneously predict interfacial residues for both the RNA and protein components of protein–RNA complexes [52] (*see* Subheading 2.5). PS-PRIP is a partner-specific method (*see* **Note 4**), which means that, given the sequences of a protein and several potential interacting RNAs, it can identify which amino acid residues contact each RNA binding partner. In other words, if the protein binds to different RNAs using distinct (or overlapping) interfaces, PS-PRIP can distinguish between these RNA-binding sites. PS-PRIP requires *both* the protein sequence and its partner RNA sequence as input. If the user does not have any potential RNA sequence(s) for testing, methods such as RPI-Seq or catRAPID can be used to infer potential partner RNAs for a specific protein (reviewed in refs. [62–65]). In addition to the sequences of the protein and its RNA-binding partners, PS-PRIP utilizes a dataset of interfacial motifs extracted from solved protein–RNA complexes in the PDB [68]. For predicting RNA-binding residues in proteins, the use of such interfacial motifs by PS-PRIP appears to provide improved precision over RNABindRPlus and other sequence-based interface prediction servers [52]. At present, the RNA-binding residues predicted by PS-PRIP are much more reliable than the protein-binding residues predicted in the bound RNA component.

1. Access the PS-PRIP server at http://pridb.gdcb.iastate.edu/PSPRIP/index.html.
2. Enter a protein sequence and the sequence for an RNA known or expected to be its binding partner in plain text format (protein sequence only and RNA sequence only, without any header information) into the text boxes provided on the homepage (*see* **Note 21**). Then click the "Submit" button.
3. Figure 4 shows results returned by PS-PRIP for the S5 protein from the 30S ribosomal subunit of *T. thermophilus*, which corresponds to protein chain E, in PDB structure 1HNX. In this case, the 16S rRNA corresponding to RNA chain A in the 1HNX structure was provided as input to PS-PRIP, in order to obtain a

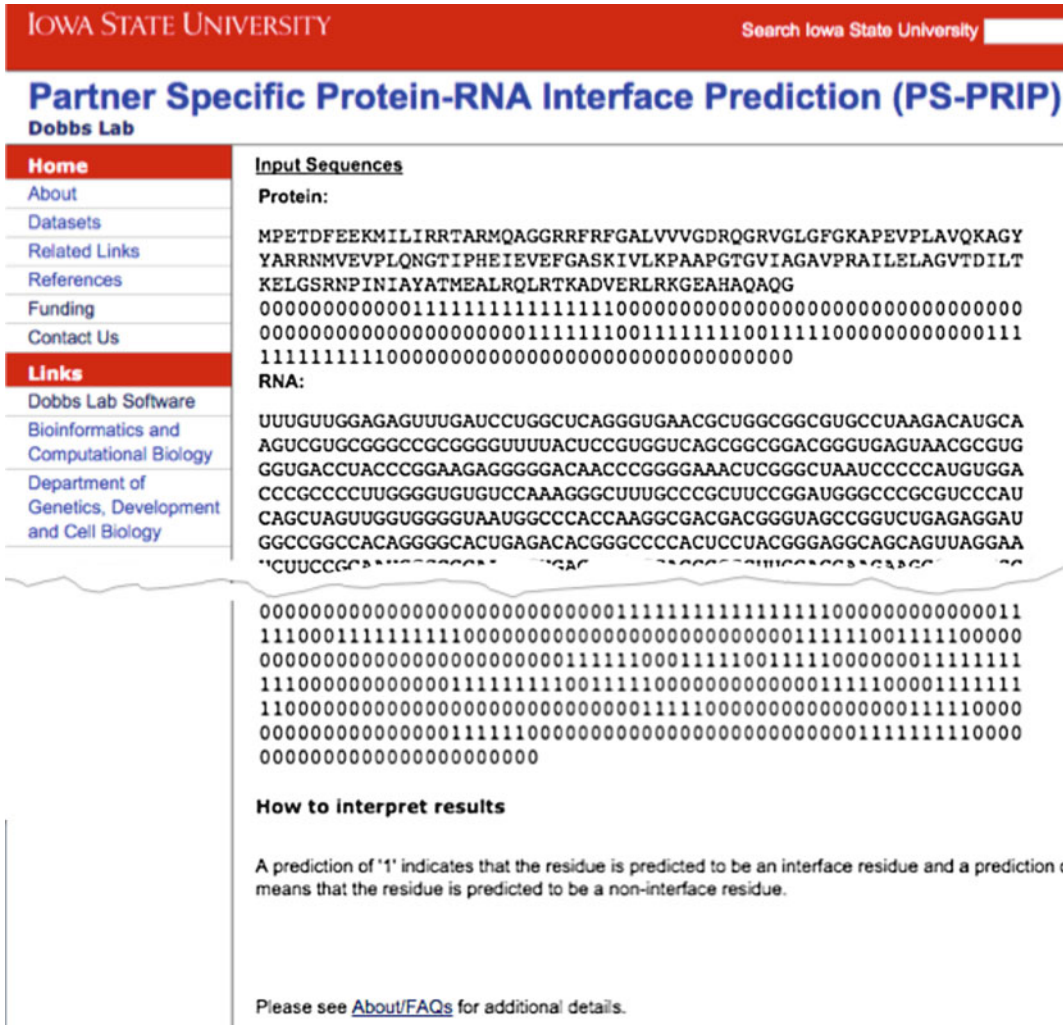


Fig. 4 PS-PRIP prediction results for the *T. thermophilus* S5 protein bound to 16S rRNA. Sequences shown correspond to protein chain E and RNA chain A in the PDB structure 1HNX. Under each sequence, the predicted interfacial residues are represented by a string of 1's and 0's, where "1" and "0" correspond to predicted binding and non-binding residues, respectively

"partner-specific" prediction. On the results page, the S5 protein sequence and 16S rRNA sequences are displayed. In the lines below each sequence, the interfacial residues are indicated by a string of 1's and 0's, where "1" and "0" correspond to predicted interfacial and non-interfacial residues, respectively.

3.5 Actual RNA-Binding Residues Compared with Predictions Using Three Different Methods

Figure 5 shows a comparison of the predicted RNA-binding residues in the *T. thermophilus* S5 ribosomal protein, for which a 3D structure is available in the PDB (1HNX; protein chain E, RNA chain A). The top line shows the amino acid sequence of the S5 protein, with red letters denoting the actual RNA-binding residues (58 out of 162 total residues), defined on the basis of a 5 Å

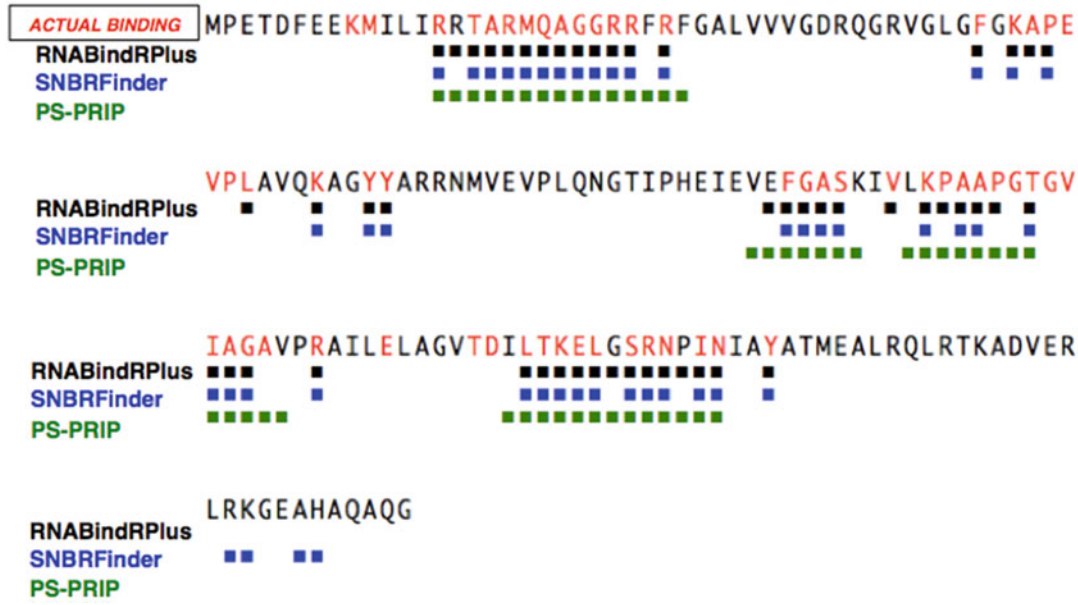


Fig. 5 Actual vs. predicted RNA-binding residues in the *T. thermophilus* S5 ribosomal protein sequence. *Top line:* Actual RNA-binding residues are shown in *red*, non-binding residues are *black. Lower lines:* Predictions obtained using RNABindRPlus, SNBRFinder and PS-PRIP. Colored boxes indicate predicted RNA-binding residues. Sequence corresponds to: PDB 1HNX; protein chain E

distance cutoff (*see* **Note 1**). RNA-binding residues predicted by RNABindRPlus, SNBRFinder and PS-PRIP are shown below. In this example, all three methods were able to identify the majority of the 58 RNAbinding residues: RNABindRPlus (46/58) SNBRFinder (41/58), PS-PRIP (33/58). A small number of false positive predictions were returned by RNABindRPlus (4), SNBRFinder (4), and a larger number by PS-PRIP (12).

In this particular example, "better than average" results were obtained because the S5 protein is a highly conserved component of the 30S ribosomal subunit. For the S5 protein, the RNA-binding residues predicted by PS-PRIP are less reliable than those predicted by RNABindRPlus and SNBRFinder. But, because the sequence of the bound RNA is also available, PS-PRIP also returns predictions for *protein*-binding residues in the 16S rRNA, which the other two servers cannot do. This example illustrates that although the overall performance of PS-PRIP was superior in terms of *precision* when tested on a benchmark dataset [52], both RNABindRPlus and SNBRFinder may perform better on certain proteins. Given the purpose of this chapter, the important point is that all three servers predict similar patches of RNA-binding residues, providing the user with a remarkably accurate prediction of the RNA-binding residues in the S5 protein, without using any structural information in order to make these predictions.

In closing, we again encourage users to submit query protein(s) of interest to at least two or three different servers from the list in Table 2, and to evaluate predictions in the context of the 3D structure, if available. All prediction results should be interpreted with caution: the computational tools are intended to help users identify the most probable RNA-binding residues in proteins, i.e., to generate hypotheses that can limit the number of experiments needed to determine RNA-binding residues using biochemical or biophysical approaches.

4 Notes

1. RNA-binding residues in proteins or other **"interfacial residues"** in the interface formed when a protein binds RNA (or DNA or another protein) are typically defined in one of two ways: (a) using a contact distance threshold, e.g., an interfacial residue is any amino acid with a heavy atom within *n* Å of a heavy atom in the bound RNA (where *n* typically ranges from 3.5 to 8 Å); (b) residues whose accessible surface area is reduced by >1 Å^2 upon complex formation [101]. It is very important to take into account how interfacial residues are defined when comparing the performance of various computational methods for predicting RNA-binding residues in proteins [47].
2. Two databases that once provided comprehensive information about interfaces in protein–RNA complexes in the PDB are no longer up-to-date: **PRIDB** [76] and **BIPA** [72]. Efforts to update PRIDB are underway. Two resources that are currently maintained and provide detailed information about interfaces in RBPs include: **NPIDB** [74] and **DBBP** [75].
3. A **position-specific scoring matrix (PSSM)** is a type of weighted scoring matrix derived from a set of aligned sequences that are considered to be homologous or functionally related [102]. PSSMs can be very sensitive because they capture important evolutionary information by exploiting the large number of protein sequences currently available.
4. A **partner-specific prediction method** takes into account the potential interacting partner(s) in predicting interfacial residues. For example, if a protein binds two distinct RNAs, RNA-1 and RNA-2, a partner-specific method will return one set of amino acids that specifically interact with only RNA-1, and a second set of amino acids that specifically interact with only RNA-2. Note that the two sets of RNA-binding residues may overlap.
5. At present, none of the available servers for predicting RNA-binding residues in proteins provide the user with existing information regarding experimentally determined RNA-binding residues (i.e., the servers always return *predicted* RNA-

binding residues, which may not be the same as the actual interfacial residues determined by experiment). Thus, as a first step, the user should always search published literature (via search engines such as **NCBI/PubMed** (http://www.ncbi.nlm.nih.gov/) or **Google Scholar** (http://scholar.google.com) and relevant databases (*see* Subheading 3.1) for existing experimental data regarding the specific RNA-binding protein(s) of interest. In addition to the resources described in Subheading 3.1 and Table 1, many new databases and servers that provide extensive information regarding protein–RNA complexes, RNA-binding proteins and their recognition sites, and in vivo protein–RNA interaction networks are becoming available. OMICtools (http://omictools.com) provides an extensive and up-to-date directory of these resources [103].

6. Users unfamiliar with **BLAST** should first read BLAST documentation and/or tutorials. A beginner's guide is available here: ftp://ftp.ncbi.nlm.nih.gov/pub/factsheets/HowTo_BLASTGuide.pdf.
7. **SmartBLAST** is a new version of BLAST that is faster than BLASTp and offers a user-friendly graphical view. For additional information, see: http://ncbiinsights.ncbi.nlm.nih.gov/2015/07/29/smartblast/.
8. **Tip:** Because proteins from humans are usually much better annotated than those from other organisms, valuable clues regarding potential RNA-binding domains or motifs in a protein can be obtained by visiting the NCBI GenBank Protein entry for the human homolog of a query sequence, if available.
9. Under the **"Related Information"** header on the GenBank Protein entry page, the user can access several different types of information, e.g., clicking on the **"Related Structures (Summary)"** link returns structurally related proteins found in NCBI's Molecular Modeling Database (MMDB), as well as an alignment of the query protein sequence with its potential homolog(s), and links for visualizing the 3D structures. Alternatively, the user can perform BLAST or Conserved Domain searches by clicking links under the **"Analyze this sequence"** header (located at the top of right-side panel), but it is usually more efficient to take advantage of precomputed information available under "Related Resources," e.g., "Blink" (for BLAST results, instead of "Run Blast"); or "CDD Search Results" (instead of "Identify Conserved Domains").
10. The **PDB Advanced Search** (http://www.rcsb.org/pdb/search/advSearch.do?search=new) is a powerful tool that allows the user to BLAST a sequence of interest against all structures in the database, to identify GO annotations, citations in publications, etc. In addition, the PDB offers several

built-in visualization tools (http://www.rcsb.org/pdb/secondary.do?p=v2/secondary/visualize.jsp—RCSBviewer) as well as links to additional resources and software for analyzing macromolecular structures (http://www.rcsb.org/pdb/static.do?p=general_information/web_links/index.html)

11. The **NDB** [69] focuses on structures that contain either RNA or DNA and provides links to many valuable RNA sequence and structure analysis tools (http://ndbserver.rutgers.edu/ndbmodule/services/index.html) as well as software for identifying RNA motifs and for predicting secondary and tertiary structures of RNA molecules (http://ndbserver.rutgers.edu/ndbmodule/services/softwares.html).
12. Currently, there is a wait of approximately 10 min per protein sequence submitted to RNABindRPlus. The rate-limiting step is generating the PSSMs using PSI-BLAST [98]. To obtain results more quickly, the user is encouraged to split large jobs into several smaller submissions (e.g., if the user would like to submit 100 proteins, she/he should submit 5 smaller jobs of 20 proteins each).
13. A faster version of this server, **FastRNABindR**, is under development. When it becomes available, a link to FastRNABindR will be provided on the RNABindRPlus website (http://ailab1.ist.psu.edu/RNABindRPlus/).
14. The user should submit the protein sequence in upper case letters to the RNABindRPlus web server. Note that this server predicts RNA-binding residues in proteins, so RNA nucleotides are not valid input.
15. The homology-based component of RNABindRPlus, **HomPRIP**, searches for homologs of the query protein. Excluding similar sequences (>30% sequence identity) ensures that the homolog and the query protein are not the same. This is useful for stringently evaluating performance of RNABindRPlus in comparison with other methods, but is not the best strategy for a user interested in identifying potential RNA-binding residues. To obtain the best possible prediction of RNA-binding residues, the user should take full advantage of all available homologous sequences (i.e., should *not* eliminate any potential homologs).
16. The **IC_score** (interface conservation score) measures the correlation between the interface and non-interface residues of a query protein Q and its putative sequence homolog H when the two are aligned. It is a measure of how well the RNA-binding residues of Q are conserved (and subsequently, can be predicted from known interface residues of homologous proteins) in protein H. However, computing the IC_score requires knowledge of interface residues in both the query protein and

its homolog. Fortunately, for a query protein with unknown RNA-binding residues, the IC_score can be estimated using BLAST alignment statistics between Q and H [57].

17. SNBRFinder allows submission of at most five sequences each time, for any of the submission options. When entering multiple UniProt IDs, IDs should be separated by commas.
18. Like RNABindRPlus, SNBRFinder allows the user to specify which sequences to exclude when searching for homologous templates, using a sequence identity cutoff. Protein templates that are more similar to the query protein are likely to return better results than templates that are less similar. The sequence identity cutoff utilized depends on the user's objective (*see* **Note 15**). To obtain the best possible prediction of RNA-binding residues, the user should take full advantage of all available homologous sequences. In contrast, for a rigorous performance comparison with other methods, a lower sequence identity cutoff should be used (i.e., to evaluate the sensitivity and specificity of the methods).
19. **HHscore** is a score that indicates the similarity score between the query protein and its best homolog/template.
20. SNBRFinder calculates the probability score of each residue being an RNA-binding residue using the following formula:

$$\text{Cscore} = \begin{cases} \alpha\,\text{Fscore} + (1-\alpha)\,\text{Tscore} & \text{if HHscore} \geq \text{cutoff} \\ \text{Fscore} & \text{otherwise} \end{cases}$$

where Fscore is the output of SNBRFinder^{F} (support vector machine component) and Tscore is the output of SNBRFinder^{T} (template-based component), $\alpha = 0.6$ and cutoff = 85 %.

21. A current limitation of PS-PRIP is that it has a minimum length requirement for both the protein and RNA sequences: proteins must be ≥25 amino acids in length and RNAs must be ≥100 nucleotides in length.

Acknowledgments

This work was supported in part by NSF DBI0923827 to DD, by NIH GM066387 to VGH and DD, by a Presidential Initiative for Interdisciplinary Research (PIIR) award to DD from Iowa State University, and by the Edward Frymoyer Chair in Information Sciences and Technology held by VGH at Pennsylvania State University. RRW is currently supported by an appointment to the ARS-USDA Research Participation Program administered by the Oak Ridge Institute for Science and Education (ORISE) through an interagency agreement between the US Department of Energy

(DOE) and USDA. ORISE is managed by ORAU under DOE contract number DE-AC05-06OR23100. We thank Carla Mann and Usha Muppirala for valuable discussions.

References

1. Re A, Joshi T, Kulberkyte E et al (2014) RNA-protein interactions: an overview. Methods Mol Biol 1097:491–521
2. Lee Y, Rio DC (2015) Mechanisms and regulation of alternative pre-mRNA splicing. Annu Rev Biochem 84:291–323
3. Fu X-D, Ares M Jr (2014) Context-dependent control of alternative splicing by RNA-binding proteins. Nat Rev Genet 15(10):689–701
4. Singh G, Pratt G, Yeo GW et al (2015) The clothes make the mRNA: past and present trends in mRNP fashion. Annu Rev Biochem 84:325–354
5. Bryant CD, Yazdani N (2016) RNA binding proteins, neural development and the addictions. Genes Brain Behav 15(1):169–186.
6. Hogg JR, Collins K (2008) Structured noncoding RNAs and the RNP renaissance. Curr Opin Chem Biol 12(6):684–689
7. Cech TR, Steitz JA (2014) The noncoding RNA revolution-trashing old rules to forge new ones. Cell 157(1):77–94
8. Castello A, Hentze MW, Preiss T (2015) Metabolic enzymes enjoying new partnerships as RNA-binding proteins. Trends Endocrinol Metab 26(12):746–757
9. Beckmann BM, Horos R, Fischer B et al (2015) The RNA-binding proteomes from yeast to man harbour conserved enigmRBPs. Nat Commun 6:10127
10. Lin Y, Protter DS, Rosen MK et al (2015) Formation and maturation of phase-separated liquid droplets by RNA-binding proteins. Mol Cell 60(2):208–219
11. Kafasla P, Skliris A, Kontoyiannis DL (2014) Post-transcriptional coordination of immunological responses by RNA-binding proteins. Nat Immunol 15(6):492–502
12. Darnell RB (2010) RNA regulation in neurologic disease and cancer. Cancer Res Treat 42(3):125–129
13. Wurth L, Gebauer F (2015) RNA-binding proteins, multifaceted translational regulators in cancer. Biochim Biophys Acta 1849(7):881–886
14. Pilaz LJ, Silver DL (2015) Post-transcriptional regulation in corticogenesis: how RNA-binding proteins help build the brain. Wiley Interdiscip Rev RNA 6(5):501–515
15. Gerstberger S, Hafner M, Tuschl T (2014) A census of human RNA-binding proteins. Nat Rev Genet 15(12):829–845
16. Neelamraju Y, Hashemikhabir S, Janga SC (2015) The human RBPome: from genes and proteins to human disease. J Proteomics 127(Pt A):61–70
17. Vaquerizas JM, Kummerfeld SK, Teichmann SA et al (2009) A census of human transcription factors: function, expression and evolution. Nat Rev Genet 10(4):252–263
18. Tsvetanova NG, Klass DM, Salzman J et al (2010) Proteome-wide search reveals unexpected RNA-binding proteins in *Saccharomyces cerevisiae*. PLoS One 5(9)
19. Castello A, Fischer B, Eichelbaum K et al (2012) Insights into RNA biology from an atlas of mammalian mRNA-binding proteins. Cell 149(6):1393–1406
20. Hashemikhabir S, Neelamraju Y, Janga SC (2015) Database of RNA binding protein expression and disease dynamics (READ DB). Database (Oxford) 2015:bav072
21. Tamburino AM, Ryder SP, Walhout AJ (2013) A compendium of *Caenorhabditis elegans* RNA binding proteins predicts extensive regulation at multiple levels. G3 (Bethesda) 3(2):297–304
22. Ray D, Kazan H, Cook KB et al (2013) A compendium of RNA-binding motifs for decoding gene regulation. Nature 499(7457):172–177
23. Jiang J, Chan H, Cash DD et al (2015) Structure of *Tetrahymena* telomerase reveals previously unknown subunits, functions, and interactions. Science 350(6260):aab4070. doi: 10.1126/science.aab4070
24. Zhang X, Ding K, Yu X et al (2015) In situ structures of the segmented genome and RNA polymerase complex inside a dsRNA virus. Nature 527(7579):531–534
25. Chen Y, Varani G (2013) Engineering RNA-binding proteins for biology. FEBS J 280(16):3734–3754
26. Wei H, Wang Z (2015) Engineering RNA-binding proteins with diverse activities. Wiley Interdiscip Rev RNA 6(6):597–613
27. Lunde BM, Moore C, Varani G (2007) RNA-binding proteins: modular design for efficient function. Nat Rev Mol Cell Biol 8(6):479–490

28. Varadi M, Zsolyomi F, Guharoy M et al (2015) Functional advantages of conserved intrinsic disorder in RNA-binding proteins. PLoS One 10(10):e0139731
29. Calabretta S, Richard S (2015) Emerging roles of disordered sequences in RNA-binding proteins. Trends Biochem Sci 40(11):662–672
30. Terribilini M, Lee JH, Yan C et al (2006) Prediction of RNA binding sites in proteins from amino acid sequence. RNA 12(8):1450–1462
31. Puton T, Kozlowski L, Tuszynska I et al (2012) Computational methods for prediction of protein-RNA interactions. J Struct Biol 179(3):261–268
32. Ke A, Doudna JA (2004) Crystallization of RNA and RNA-protein complexes. Methods 34(3):408–414
33. Wu H, Finger LD, Feigon J (2005) Structure determination of protein/RNA complexes by NMR. Methods Enzymol 394:525–545
34. Carlomagno T (2014) Present and future of NMR for RNA-protein complexes: a perspective of integrated structural biology. J Magn Reson 241:126–136
35. Binshtein E, Ohi MD (2015) Cryo-electron microscopy and the amazing race to atomic resolution. Biochemistry 54(20):3133–3141
36. Hennig J, Sattler M (2015) Deciphering the protein-RNA recognition code: combining large-scale quantitative methods with structural biology. Bioessays 37(8):899–908
37. Faoro C, Ataide SF (2014) Ribonomic approaches to study the RNA-binding proteome. FEBS Lett 588(20):3649–3664
38. McHugh CA, Russell P, Guttman M (2014) Methods for comprehensive experimental identification of RNA-protein interactions. Genome Biol 15(1):203
39. Campbell ZT, Wickens M (2015) Probing RNA-protein networks: biochemistry meets genomics. Trends Biochem Sci 40(3):157–164
40. Cook KB, Hughes TR, Morris QD (2015) High-throughput characterization of protein-RNA interactions. Brief Funct Genomics 14(1):74–89
41. Cook KB, Kazan H, Zuberi K et al (2011) RBPDB: a database of RNA-binding specificities. Nucleic Acids Res 39(Database issue):D301–D308
42. Li X, Kazan H, Lipshitz HD et al (2014) Finding the target sites of RNA-binding proteins. Wiley Interdiscip Rev RNA 5(1):111–130
43. Kazan H, Morris Q (2013) RBPmotif: a web server for the discovery of sequence and structure preferences of RNA-binding proteins. Nucleic Acids Res 41(Web Server issue):W180–W186
44. Banerjee H, Singh R (2008) A simple cross-linking method, CLAMP, to map the sites of RNA-contacting domains within a protein. Methods Mol Biol 488:181–190
45. Kramer K, Sachsenberg T, Beckmann BM et al (2014) Photo-cross-linking and high-resolution mass spectrometry for assignment of RNA-binding sites in RNA-binding proteins. Nat Methods 11(10):1064–1070
46. Qamar S, Kramer K, Urlaub H (2015) Studying RNA-protein interactions of pre-mRNA complexes by mass spectrometry. Methods Enzymol 558:417–463
47. Walia RR, Caragea C, Lewis BA et al (2012) Protein-RNA interface residue prediction using machine learning: an assessment of the state of the art. BMC Bioinformatics 13(1):89
48. Zhao H, Yang Y, Zhou Y (2013) Prediction of RNA binding proteins comes of age from low resolution to high resolution. Mol Biosyst 9(10):2417–2425
49. Nagarajan R, Gromiha MM (2014) Prediction of RNA binding residues: an extensive analysis based on structure and function to select the best predictor. PLoS One 9(3):e91140
50. Si J, Cui J, Cheng J et al (2015) Computational prediction of RNA-binding proteins and binding sites. Int J Mol Sci 16(11):26303–26317
51. Mitchell A, Chang HY, Daugherty L et al (2015) The InterPro protein families database: the classification resource after 15 years. Nucleic Acids Res 43(Database issue):D213–D221
52. Muppirala UK, Lewis BA, Mann CM et al (2016) A motif-based method for predicting interfacial residues in both the RNA and protein components of protein-RNA complexes. Pac Symp Biocomput 2016:445–455. doi:10.1142/9789814749411_0041
53. Williamson JR (2000) Induced fit in RNA-protein recognition. Nat Struct Biol 7(10):834–837
54. Ellis JJ, Jones S (2008) Evaluating conformational changes in protein structures binding RNA. Proteins 70(4):1518–1526
55. Sankar K, Walia R, Mann C et al (2014) An analysis of conformational changes upon RNA-protein binding. In: ACM BCB 2014 5th ACM conference on bioinformatics, computational biology, and health informatics, Washington, DC, 2013. ACM New York, NY, USA ©2014 pp 592–593 doi: 10.1145/2649387.2660790
56. Spriggs RV, Jones S (2009) RNA-binding residues in sequence space: conservation and interaction patterns. Comput Biol Chem 33(5):397–403
57. Walia RR, Xue LC, Wilkins K et al (2014) RNABindRPlus: a predictor that combines machine learning and sequence homology-based methods to improve the reliability of

predicted RNA-binding residues in proteins. PLoS One 9(5):e97725
58. Yang X, Wang J, Sun J et al (2015) SNBRFinder: a sequence-based hybrid algorithm for enhanced prediction of nucleic acid-binding residues. PLoS One 10(7):e0133260
59. Tuszynska I, Matelska D, Magnus M et al (2014) Computational modeling of protein-RNA complex structures. Methods 65(3):310–319
60. Gupta A, Gribskov M (2011) The role of RNA sequence and structure in RNA—protein interactions. J Mol Biol 409(4):574–587
61. Panwar B, Raghava GP (2015) Identification of protein-interacting nucleotides in a RNA sequence using composition profile of trinucleotides. Genomics 105(4):197–203
62. Mann C, Muppirala UK, Dobbs DL (2016) Computational prediction of RNA-protein interactions. Methods Mol Biol. In press
63. Muppirala UK, Lewis BA, Dobbs D (2013) Computational tools for investigating RNA-protein interaction partners. J Comput Sci Syst Biol 6:182–187
64. Cirillo D, Livi CM, Agostini F et al (2014) Discovery of protein-RNA networks. Mol Biosyst 10(7):1632–1642
65. Marchese D, Livi CM, Tartaglia GG (2016) A computational approach for the discovery of protein-RNA networks. Methods Mol Biol 1358:29–39
66. Zhao H, Yang Y, Janga SC et al (2014) Prediction and validation of the unexplored RNA-binding protein atlas of the human proteome. Proteins 82(4):640–647
67. Kumar M, Gromiha MM, Raghava GP (2011) SVM based prediction of RNA-binding proteins using binding residues and evolutionary information. J Mol Recognit 24(2):303–313
68. Berman HM, Westbrook J, Feng Z et al (2000) The protein data bank. Nucleic Acids Res 28(1):235–242
69. Coimbatore Narayanan B, Westbrook J, Ghosh S et al (2014) The nucleic acid database: new features and capabilities. Nucleic Acids Res 42(Database issue):D114–D122
70. de Beer TA, Berka K, Thornton JM et al (2014) PDBsum additions. Nucleic Acids Res 42(Database issue):D292–D296
71. Laskowski RA, Hutchinson EG, Michie AD et al (1997) PDBsum: a Web-based database of summaries and analyses of all PDB structures. Trends Biochem Sci 22(12):488–490
72. Lee S, Blundell TL (2009) BIPA: a database for protein-nucleic acid interaction in 3D structures. Bioinformatics 25(12):1559–1560
73. Jones P, Binns D, Chang HY et al (2014) InterProScan 5: genome-scale protein function classification. Bioinformatics 30(9):1236–1240
74. Kirsanov DD, Zanegina ON, Aksianov EA et al (2013) NPIDB: nucleic acid—protein interaction database. Nucleic Acids Res 41(D1):D517–D523
75. Park B, Kim H, Han K (2014) DBBP: database of binding pairs in protein-nucleic acid interactions. BMC Bioinformatics 15(Suppl 15):S5
76. Lewis BA, Walia RR, Terribilini M et al (2011) PRIDB: a protein-RNA interface database. Nucleic Acids Res 39(Database issue):D277–D282
77. Shulman-Peleg A, Nussinov R, Wolfson HJ (2009) RsiteDB: a database of protein binding pockets that interact with RNA nucleotide bases. Nucleic Acids Res 37(Suppl 1):D369–D373
78. Kumar MDS, Bava KA, Gromiha MM et al (2006) ProTherm and ProNIT: thermodynamic databases for proteins and protein-nucleic acid interactions. Nucleic Acids Res 34(Database issue):D204–D206
79. Vanegas PL, Hudson GA, Davis AR et al (2012) RNA CoSSMos: characterization of secondary structure motifs—a searchable database of secondary structure motifs in RNA three-dimensional structures. Nucleic Acids Res 40(Database issue):D439–D444
80. Petrov AI, Zirbel CL, Leontis NB (2013) Automated classification of RNA 3D motifs and the RNA 3D Motif Atlas. RNA 19(10):1327–1340
81. Chojnowski G, Walen T, Bujnicki JM (2014) RNA Bricks—a database of RNA 3D motifs and their interactions. Nucleic Acids Res 42(Database issue):D123–D131
82. Livi CM, Klus P, Delli Ponti R et al (2015) catRAPID signature: identification of ribonucleoproteins and RNA-binding regions. Bioinformatics. Oct 31. pii: btv629. [Epub ahead of print]
83. Wang L, Brown SJ (2006) BindN: a web-based tool for efficient prediction of DNA and RNA binding sites in amino acid sequences. Nucleic Acids Res 34(suppl 2):W243–W248
84. Wang L, Huang C, Yang MQ et al (2010) BindN+ for accurate prediction of DNA and RNA-binding residues from protein sequence features. BMC Syst Biol 4(Suppl 1):S3
85. Zhao H, Yang Y, Zhou Y (2011) Structure-based prediction of RNA-binding domains and RNA-binding sites and application to

structural genomics targets. Nucleic Acids Res 39(8):3017–3025

86. Kim OTP, Yura K, Go N (2006) Amino acid residue doublet propensity in the protein–RNA interface and its application to RNA interface prediction. Nucleic Acids Res 34(22):6450–6460
87. Carson MB, Langlois R, Lu H (2010) NAPS: a residue-level nucleic acid-binding prediction server. Nucleic Acids Res 38(Web Server Issue):W431–W435
88. Pérez-Cano L, Fernández-Recio J (2010) Optimal protein-RNA area, OPRA: a propensity-based method to identify RNA-binding sites on proteins. Proteins 78(1):25–35
89. Kumar M, Gromiha MM, Raghava GPS (2008) Prediction of RNA binding sites in a protein using SVM and PSSM profile. Proteins 71(1):189–194
90. Ma X, Guo J, Wu J et al (2011) Prediction of RNA-binding residues in proteins from primary sequence using an enriched random forest model with a novel hybrid feature. Proteins 79(4):1230–1239
91. Maetschke SR, Yuan Z (2009) Exploiting structural and topological information to improve prediction of RNA-protein binding sites. BMC Bioinformatics 10(1):341
92. Miao Z, Westhof E (2015) Prediction of nucleic acid binding probability in proteins: a neighboring residue network based score. Nucleic Acids Res 43(11):5340–5351
93. Tong J, Jiang P, Lu Z-H (2008) RISP: a web-based server for prediction of RNA-binding sites in proteins. Comput Methods Programs Biomed 90(2):148–153
94. Terribilini M, Sander JD, Lee JH et al (2007) RNABindR: a server for analyzing and predicting RNA-binding sites in proteins. Nucleic Acids Res 35(Web Server issue):W578–W584
95. Yang Y, Zhao H, Wang J et al (2014) SPOT-Seq-RNA: predicting protein-RNA complex structure and RNA-binding function by fold recognition and binding affinity prediction. Methods Mol Biol 1137:119–130
96. Remmert M, Biegert A, Hauser A et al (2012) HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. Nat Methods 9(2):173–175
97. Altschul SF, Gish W, Miller W et al (1990) Basic local alignment search tool. J Mol Biol 215(3):403–410
98. Altschul SF, Madden TL, Schaffer AA et al (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25(17):3389–3402
99. Lambert N, Robertson A, Jangi M et al (2014) RNA Bind-n-Seq: quantitative assessment of the sequence and structural binding specificity of RNA binding proteins. Mol Cell 54(5):887–900
100. Paz I, Kosti I, Ares M Jr et al (2014) RBPmap: a web server for mapping binding sites of RNA-binding proteins. Nucleic Acids Res 42(Web Server issue):W361–W367
101. Jones S, Daley DT, Luscombe NM et al (2001) Protein-RNA interactions: a structural analysis. Nucleic Acids Res 29(4):943–954
102. Stormo GD, Schneider TD, Gold L et al (1982) Use of the "Perceptron" algorithm to distinguish translational initiation sites in *E. coli*. Nucleic Acids Res 10(9):2997–3011
103. Henry VJ, Bandrowski AE, Pepin AS et al (2014) OMICtools: an informative directory for multi-omic data analysis. Database (Oxford). doi:10.1093/database/bau069

Chapter 16

Computational Approaches for Predicting Binding Partners, Interface Residues, and Binding Affinity of Protein–Protein Complexes

K. Yugandhar and M. Michael Gromiha

Abstract

Studying protein–protein interactions leads to a better understanding of the underlying principles of several biological pathways. Cost and labor-intensive experimental techniques suggest the need for computational methods to complement them. Several such state-of-the-art methods have been reported for analyzing diverse aspects such as predicting binding partners, interface residues, and binding affinity for protein–protein complexes with reliable performance. However, there are specific drawbacks for different methods that indicate the need for their improvement. This review highlights various available computational algorithms for analyzing diverse aspects of protein–protein interactions and endorses the necessity for developing new robust methods for gaining deep insights about protein–protein interactions.

Key words Protein–protein interaction, Binding partner, Interface residue, Binding affinity

1 Introduction

Proteins are involved in several biological reactions by means of interactions with other proteins or with other molecules such as nucleic acids, carbohydrates, and ligands. Among these interaction types, protein–protein interactions (PPIs) are considered to be one of the key factors as they are involved in most of the cellular processes. Protein–protein complexes can be classified into various types such as dimeric-multimeric, homodimeric-heterodimeric, transient-permanent, and obligate-nonobligate based on different aspects such as the number of subunits, type of the interacting proteins, biological significance of the complexes, and the interaction time [1].

Understanding the underlying principles of PPIs provides important clues for designing efficient drugs for treating various diseases [2]. Several investigations have been carried out on different perspectives: (1) prediction of interaction pairs from a given set of protein sequences or structures, (2) identification and prediction of binding sites from protein–protein complexes, (3) prediction of

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_16,

binding site residues from sequence or structure information, (4) protein–protein binding affinity prediction, and (5) important interactions for the formation of protein–protein complexes and their recognition mechanism. Recently, several studies have been carried out on various aspects of PPIs, which include understanding the recognition mechanism [3], role of specific interactions [3–5], identification of the binding sites from protein structures [6–12], and predicting the interaction sites from amino acid sequence [13–15]. Further, prediction methods have been reported for the identification of interaction partners by utilizing protein structure [16, 17] and sequence information [18–20]. The features employed for devising these methods are mainly based on evolutionary information [17], physicochemical properties [20], structural similarity [12, 17] etc.

In this review, we address the studies based on different aspects of PPIs using various bioinformatic/computational biology approaches. We systematically highlight the methods developed exclusively for predicting the interacting partners, interface residues, and binding affinity of protein–protein complexes along with potential applications and necessity for the development of more robust state-of-the-art methods for efficient prediction. Further, we provide insights into various features that are reported to be influencing the specificity in PPIs.

2 Prediction of Protein–Protein Interaction Pairs

The information about the pair of proteins, which are interacting with each other, is obtained with several experimental methods such as yeast two-hybrid, Förster/fluorescence resonance energy transfer (FRET), surface plasmon resonance, and isothermal titration calorimetry. These PPI data are deposited in well-maintained databases and are listed in Table 1. In addition, tools such as PIPE2 [21] provide a platform for integration and annotation of such data. Text mining-based methods search for statistically significant co-occurrences between gene names in online resources and public repositories. Several methods have been developed, which try to automate extraction of the interacting proteins through their coexistence in paragraphs, abstracts, and sentences [22–29]. A method known as eFIP (extracting functional impact of Phosphorylation) [26] employs various natural language processing (NLP) techniques for locating the text, which mention protein phosphorylation in addition to PPIs. BioRAT [22] is a standalone tool that locates and downloads literature reports based on the user query. On the other hand, PPIExtractor [27] utilizes feature coupling generalization to recognize names of the proteins and subsequently visualizes the PPI network. HPIminer [28] is a web-based tool, which performs text mining for human PPI and visualizes the interaction networks. A comprehensive review about the methods for

Table 1
Available databases analyzing protein–protein interactions

Name	Description	Link	Reference
Interaction databases			
DIP	Database of interacting proteins	http://dip.doe-mbi.ucla.edu/dip/Main.cgi	[120]
BioGrid	Biological general repository for interaction datasets	http://thebiogrid.org/	[121]
HPRD	Human protein reference database	http://www.hprd.org/	[122]
STRING	Search tool for the retrieval of interacting genes/proteins	http://string-db.org/	[123]
IntAct	Provide curated data from literature and user-submitted interactions	http://www.ebi.ac.uk/intact/	[124]
Databases for interface			
PDBsum	Pictorial database of 3D structures from PDB	http://www.ebi.ac.uk/pdbsum	[125]
Thermodynamic databases			
PINT	Provide thermodynamic parameters along with experimental conditions	http://www.bioinfodatabase.com/pint/	[126]
PDBbind	Resource for experimental binding affinity data for complexes with PDB ids	http://www.pdbbind-cn.org/	[127]
KDBI	Provide experimental kinetic data	http://xin.cz3.nus.edu.sg/group/kdbi/kdbi.asp	[128]
SKEMPI	Repository for binding free energy changes upon mutation	http://life.bsc.es/pid/mutation_database/database.html	[129]
ASEDB	Alanine scanning energetics database	http://nic.ucsf.edu/asedb/	[130]
Protein–protein interaction affinity database	Provide experimentally determined affinity data along with structures of both complex and free proteins	http://bmm.crick.ac.uk/~bmmadmin/Affinity/	[131]

Note: accessed as on 30th October, 2015

predicting PPIs has been reported recently [29]. Utilizing data available in the PPI databases, several methods based on statistical analysis and machine learning techniques have been reported for identifying the interacting pairs. In this section, we review various methods, which are broadly categorized into five types: (1) genomic context, (2) protein sequence, (3) protein domains, (4) tertiary structure, and (5) biological context.

2.1 Genomic Analysis-Based Methods

The primitive methods for predicting PPIs based on genomic analysis utilize the principle of co-localization or gene neighborhood [30]. They exploit the notion that genes, which undergo physical interaction or functional association, will be in close physical proximity in the genome [31–34]. The most plausible case of this phenomenon involves archaeal and bacterial operons, where genes that work together are usually transcribed on the same poly-cistronic mRNA. In such cases, proteins, which are involved in the same pathway or process, are frequently encoded on the same poly-cistronic messenger. Moreover, operons encoding for co-regulated genes are generally conserved [30]. However, the choice of reference genomes might affect the performance of such methods [35].

2.2 Methods Based on Protein Sequences

Several methods have been reported for predicting PPIs based on sequence information alone [17–19, 36–43]. These methods employ various physicochemical properties such as hydrophobicity, charge, neighboring information etc. [17, 36, 38] or the frequencies of small motifs with residue combinations [37]. Most of the methods use machine learning, especially support vector machines for predicting the tendency of the query PPI pair [17, 36–38, 40].

2.3 Methods Using Domain Information

A domain can be defined as an elementary unit of protein structure and evolution that can fold and function independently. Presence of specific protein domains are reported to be influencing the PPIs [44, 45]. Exploring this aspect, several reported methods predict interactions based on the presence of particular domains in query proteins [46–52]. These methods determine associations by analyzing the domains on pairs of proteins that have been reported by previous experimental PPI detection methods. Initially, they count occurrences of the domain pairs (x,y) on pairs of interacting proteins, such that domain x is present on one protein and domain y is present on the other. Further, the domain pairs are associated with an interaction if they are found to be enriched among interacting protein pairs. In addition, PPIs depend on other factors such as subcellular localization and post-translational modifications. It may be noted that about 80% of interactions do not occur through domain-domain binding [53]. This indicates the possible limitation of the domain-based approach in generalizing it for various types of PPI pairs.

2.4 Methods Based on Tertiary Structure

The available docking methods utilize the tertiary structure of two proteins to predict the structure of the resulting complex [54]. However, these methods mostly try to find the best structure of the complex by considering electrostatic complementarity and shape of the protein surfaces. Apart from predicting the interaction partners, these methods also search for the optimal fit between the two proteins [55–58]. Hence, several methods employ a combination of sequence and structure-based features to predict the interactions [59]. The methodology of such methods include steps such as (1) determining whether members of a given protein pair

have sequence similarity with other proteins in a solved complex and (2) assessing whether the candidate proteins could form a similar complex. Generally, it is assumed that if proteins have more than 20% sequence identity, they interact in the same way [60]. Major limitations of these structure-based methods include the need for solved structures of the candidate proteins and they are known to be less accurate for interactions involving conformational changes at the interface [57].

2.5 Approaches Based on Biological Context

Gene expression analysis facilitates the determination and identification of not only genes that are active in a given state but also the sets of genes that are co-regulated in various different states [30]. It has been reported that according to microarray analysis, several interacting proteins are co-expressed [61–63]. Although the current gene expression methods cannot directly determine whether a pair of proteins interact or not, several computational approaches have been developed that could use the expression data towards the prediction of PPI and gene regulatory networks [30, 61–64].

3 Prediction of Binding Site Residues

Binding site residues in proteins are one of the key factors that enable us to understand and unravel the mechanisms that underlie biomolecular recognition process. Proteins tend to interact with other proteins through much larger and structurally more intricate surfaces, in contrast to their interaction pattern in case of other smaller substrates [65]. The straightforward way of identifying interaction sites in protein–protein complexes is to analyze the three-dimensional structures. Different definitions for binding site residues based on (1) distance, (2) change in accessible surface area, and (3) interaction energy have been widely used in literature. In distance-based criterion, a residue is said to be interacting, if it has at least one contact with any of the heavy atoms in the partner within a cut-off distance of 4–6 Å [66]. Accessible surface area (ASA) is calculated by theoretically rolling a probe (typically of water with radius 1.4 Å) around the surface of a molecule. Usually change in ASA of >0.1 $Å^2$ upon complex formation is considered for identifying binding site residues in protein–protein complexes [67]. Energy-based approach utilizes the interaction energy between all atoms in the pair of proteins and the contribution from the atoms in a residue is summed up to obtain the interaction energy of a residue. Residues, which have the interaction energy of less than 1 kcal/mol, are treated as binding site residues [5]. The databases, which contain the information about interacting residues, are listed in Table 1. In the absence of complex structures, several methods have been proposed to predict the binding sites from the structure of a free protein or just from the amino acid sequence. The currently available methods are included in Table 2.

Table 2
Tools for studying protein–protein interactions

Name	Features	Link	Reference
Methods for predicting interface residues			
Sequence-based			
PSIVER	PSSM and predicted solvent accessibility	http://tardis.nibio.go.jp/PSIVER/	[86]
SPPIDER	Solvent accessibility; different methods available for sequence and structure	http://sppider.cchmc.org/	[66]
ISIS	PSSM, predicted secondary structure and solvent accessibility	https://www.predictprotein.org/	[14]
PPiPP	Partner-specific method; contact propensity	http://mizuguchilab.org/netasa/ppipp/	[15]
PS-HomPPI	Partner-specific method; conservation score	http://ailab1.ist.psu.edu/PSHOMPPIv1.2/	[88]
NPS-HomPPI	Conservation score	http://ailab1.ist.psu.edu/NPSHOMPPI/	[88]
LORIS	PSSM and predicted solvent accessibility	https://sites.google.com/site/sukantamondal/software	[87]
Structure-based			
Promate	Residue conservation, propensity, and geometric properties	http://bioinfo41.weizmann.ac.il/promate/promate.html	[7]
ConsPPISP	PSSM and solvent accessibility	http://pipe.sc.fsu.edu/ppisp/	[73]
PINUP	Residue interface propensity and conservation score	http://sysbio.unl.edu/services/PINUP/	[75]
Meta-PPISP	Based on promate, ConsPPISP and PINUP	http://pipe.sc.fsu.edu/meta-ppisp/	[78]
WHISCY	Residue propensity and solvent accessibility	http://www.nmr.chem.uu.nl/Software/whiscy/startpage.htm	[74]
PredUs	Solvent accessibility	https://honiglab.c2b2.columbia.edu/PredUs/	[81]
VORFFIP	Residue propensity, solvent accessibility, residue environment by Voronoi diagrams and conservation score	http://www.bioinsilico.org/cgi-bin/VORFFIP/htmlVORFFI/home	[80]
PIER	Statistical properties of atomic groups	http://abagyan.ucsd.edu/PIER/index.cgi	[76]
eFindSite PPI	Relative surface accessibility, interface propensity, and sequence entropy	http://brylinski.cct.lsu.edu/efindsiteppi	[82]

(continued)

Table 2
(continued)

Name	Features	Link	Reference
Methods for binding affinity prediction			
PPA-Pred	Residue propensity and predicted binding sites; sequence-based method	http://www.iitm.ac.in/bioinfo/PPA_Pred/	[105]

Note: accessed as on 30th October, 2015

3.1 Structure-Based Methods

The availability of three-dimensional structures of protein–protein complexes prompted researchers to develop various structural parameters and use them to identify the binding sites [68]. Jones and Thornton [6] analyzed the interactions as patches of residues on the surface, which have been used to predict the binding sites using support vector machines and Bayes networks [69, 70]. However, the most recent methods carried out analysis of individual residues instead of considering surface patches. [7, 66, 71–82]. The most commonly utilized features for developing such methods include solvent accessibility [72, 73, 80, 82], residue propensity [7, 74, 75, 80], local structural similarity [12], Voronoi diagrams [80], geometric [7] and thermodynamic properties [82]. A comparative evaluation of various prediction methods on a common dataset has been reported very recently by Maheshwari and Brylinski [83].

3.2 Sequence-Based Methods

Kini and Evans [84] reported a method based on the occurrence of proline at the flanking segments of interaction sites. Gallet et al. [85] utilized the hydrophobic moment of sequence stretches to identify the interaction sites. Later, several methods have been reported to predict the binding sites using position-specific scoring matrices (PSSM) [14, 86, 87], predicted solvent accessibility [7, 14, 86], and predicted secondary structure [14]. Further, few methods have been developed by considering the sequence of both partners in a complex utilizing conservation score and contact propensity [15, 88]. The ability to predict binding site residues using sequence information alone is the major advantage of these methods. However, the methods based on PSSMs are often time-consuming and the prediction might vary depending on the database used for the alignment search.

4 Computational Methods for Protein–Protein Affinity Prediction

Predicting the affinity of protein–protein complexes has been a topic of active research for more than two decades. The availability of experimental data on binding affinity prompted researchers to

explore the principles and develop methods for prediction. Chothia and Janin [89] reported that the buried surface area (BSA) is directly related to the binding affinity. Horton and Lewis [90] developed the first method by considering both polar and apolar fractions of the surface, which are buried upon complex formation. Further, Ma et al. [91] developed an empirical scoring function for calculating the binding free energies and reported a correlation of 0.94 on a set on 20 complexes. Jiang et al. [92] devised a knowledge-based energy function, which could predict the binding affinities of 28 complexes with R^2 of 0.56. Another method reported by Audie and Scarlata [93] showed R^2 of 0.97 on a training set of 24 complexes. Su et al. [94] utilized atom pair potential in developing a method that showed a correlation of 0.76 on a set of 86 complexes.

Kastritis and Bonvin [95] published a binding affinity benchmark, which contains manually curated experimental binding affinity data (Kd) from literature for a set of 81 protein–protein complexes. They also verified the predictive power of the available affinity prediction methods including scoring functions and observed that their performance was poor on the validation set. Further, they grouped the complexes in the dataset based on experimental techniques used to measure the binding affinity in the literature such as surface plasmon resonance (SPR), isothermal titration calorimetry (ITC), fluorescence spectroscopy, spectrophotometric assays, radio ligand binding, and stopped flow fluorimetry. They observed that few methods showed better performance on complexes associated with the experimental techniques such as SPR, ITC, and various spectrophotometric assays. Based on these results, they postulated that noise in experimental data could be an additional factor that influenced the performance of the various algorithms in predicting the protein–protein binding affinity and also suggested the need to device more efficient and robust experimental techniques to obtain high quality affinity data.

Later, Kastritis et al. [96] published protein–protein binding affinity data for 144 complexes along with three-dimensional structures of complexes and free proteins. Using these data, various methods have been proposed to accomplish the task of predicting protein–protein binding affinity from structural information using knowledge-based approach [97, 98] and quantitative structure-activity relationship [99, 100]. Moal et al. [97] have devised a consensus method, by combining four different machine learning algorithms and observed that the correlation (r) with experimental affinity was similar to the one that obtained with buried surface area in rigid complexes [96]. Vreven et al. [98] reported a correlation of 0.63 using multiple regression technique. However, the performance on antigen antibody complexes was insignificant ($r = 0.24$). Tian et al. [99] have considered the importance of conformational changes in biomolecular recognition for developing a model. They derived a total of 840 features covering all possible

combinations of amino acid residues at the interface for different binding conformations of 144 protein–protein complexes. The feature selection step using genetic algorithm resulted in a set of 378 features and most of them represent steric and hydrophobic interactions. However, the number of selected features (378) was much higher than the number of experimental data (144), indicating a possible overfitting [101]. Recently, Vangone and Bonvin [102] proposed a method by utilizing the features derived from residue contacts and noninteracting surface, and showed that its performance was better than other available methods.

Huge difference between the available sequences and three-dimensional structures of protein–protein complexes suggests the necessity for developing methods for predicting protein–protein binding affinity based on the sequence information alone. This information would potentially provide experimental biologists with preliminary knowledge about the interaction strength for the complexes of interest. To address this issue, we developed two sequence-based methods: (1) classifying protein–protein complexes based on their binding affinity and (2) predicting the absolute value for affinity using functional information. The binary classification model showed the accuracy in the range of 76.1–85.7% on different validation data sets [103]. Further, this has been used as an efficient tool for analyzing large-scale PPI data form various organisms by constructing interaction networks [104]. The regression model devised for predicting real value of binding affinity (PPA-Pred) (Table 2) has been based on a hypothesis that the binding affinity of a protein–protein complex depends on the function being carried out by that complex in a biological system [105]. This method showed reliable performance across various functional classes and further we demonstrated its efficiency compared to the baseline predictor [106]. However, considering the need for high amount of training data, especially for sequence-based methods, we realize the need for incorporation of more recent affinity data and refining the prediction methods to make them reliable and robust.

4.1 Important Features Influencing Binding Affinity

Specific features are reported to be determining factors of protein–protein binding affinity in various analyses [101]. A brief description of them is as follows:

4.1.1 Buried Surface Area (BSA)

BSA is the area of the interacting proteins that loses accessibility to solvent upon complex formation. Chothia and Janin [89] showed that BSA is the primary descriptor to be related to the binding affinity, and specifically to the intrinsic bond energy. Further, BSA compensates the area, which is not buried intramolecularly within the potentially unstable subunits. It is a macroscopic parameter for hydrophobic interactions of proteins and its magnitude has been estimated to be 0.025 kcal/mol/Å^2 of the hydrophobic surface

removed from contact with water [101]. Apart from being a favorable attraction of hydrophobic surfaces, this hydrophobic interaction also expresses the gain in entropy of the water molecules released upon complex formation.

Later, Kastritis et al. [96] demonstrated that for a set of 70 rigid complexes (interface rmsd < 1 Å; where interface rmsd is the root mean square displacement of the C-α atoms of interface residues in the two partners), BSA alone showed a correlation of 0.54. However, the correlation vanished for complexes with interface rmsd > 1 Å. This suggests the dominant role played by interface surface in determining the binding affinity and specificity, for proteins undergoing less or negligible conformational changes. One of the earlier reports suggests that the rigid interface residues contribute significantly to the stabilization of the interface structure in the unbound state [107]. Chakravarty et al. [67] reassessed the BSA in relation with local conformational changes in protein–protein complexes. Recently, Janin [108] showed that interface information along with conformational changes could show a notable performance on a diverse dataset. Our previous analyses demonstrated the importance of interface residues especially, aromatic and positively charged residues in binding affinity prediction [103, 105]. A recent analysis suggests that the noninteracting surface also has a role in modulating the protein–protein binding affinity [109]. The authors observed that two features viz. polar residues on the surface and charged residues are useful in estimating the affinity. When combined with the classical interface properties, these two descriptors could reasonably explain the binding affinity of all the complexes in the dataset of 143 data that include both rigid and flexible cases with a correlation of 0.50.

4.1.2 Anchor Residues and Hot Spots

It has been reported that a relatively small number of interface residues, known as warm spots and hot spots, account for the majority of the binding energy [110, 111]. Hot and warm spots are defined as the residues whose mutation to alanine results in a destabilization of the bound state by greater than or equal to 4 and 1–2 kcal/mol, respectively. In contrast, null spots do not result in such a free energy difference. The contribution of a residue to the binding free energy can be experimentally measured using alanine scanning mutagenesis [112, 113]. Mutation of a residue to alanine essentially removes the side chain, leaving only the β-carbon. Subsequently, the kinetic analyses may provide information regarding the role played by individual amino acids in protein binding. It is noteworthy that mutating the reference residue to glycine might theoretically be a better option because the whole side chain is removed. However, mutation to glycine is not preferred as it might introduce global or local changes to the conformation of the molecule.

Various typical features of hot spot residues include (1) they are more conserved than the non-hotspots [114], (2) they are occluded from solvent [111, 115], (3) their amino acid composition is different from that of the non-hotspot residues [14], and (4) they are mostly found in the central region of the interface region [111]. Usually hot spots that are buried at the interface are surrounded by polar regions of higher packing density [116]. They clearly demonstrate that hydrophobic interactions are not the absolute determinant for binding. However, the bulkier residues tend to be found more frequently in hot spots and they have the largest surface area [117].

4.1.3 Allosteric Regulators

Initially, allostery has been defined as the regulation of a protein by a small molecule that is not its substrate [118]. However, it is modified to account for regulation of a given protein by a change in its tertiary or quaternary structure induced by small molecule. Usually, allosteric effects are now being considered as changes in the structure or dynamics of a protein by a modulator that can be of any type ranging from a small molecule to another protein [119]. Such changes might be responsible for shifting the population of inactive protein to its active form and subsequently alter the binding affinity and one of such examples includes the binding of oxygen to hemoglobin. Other examples besides oxygen include electron donor organic molecules (such as ATP) and post-translational modification events (e.g., phosphorylation) [101]. These modifications results in alteration of the binding affinity of interaction partners through the changes in the structure and/or dynamics of the proteins that interact.

4.2 Limitations and Obstacles

In spite of significant progress in the field of protein–protein binding affinity prediction, there is much more scope for improvement of the overall predictive ability of the currently available methods [101]. The possible reasons for the limitations of the currently available methods could be (1) effects of temperature, pH, solvent, and concentration are usually ignored due to insufficient number of data, (2) the quality of the experimental structure or affinity data might be ambiguous, (3) the structure-based models basically relate a structure that has been solved in its crystalline state to the affinity measured in solution state, which might lead to ambiguous results as a result of the different natures of the two states, (4) considering the simplest binding mechanisms, i.e., the lock and key model and ignoring the conformational changes that take place upon binding or due to allosteric modifications and (5) not properly accounting the underlying energetic aspects of the free proteins, and (6) using large number of features, which cause overfitting of data.

5 Conclusion

Development and efficient utilization of computational methods for understanding various important aspects of protein–protein interactions such as interaction partners, interface, and binding affinity aids in strengthening our knowledge about the mechanism of recognition and specificity in protein–protein complexes. In this review, we highlighted the developments and their contribution for providing deeper insights in respective areas of study. Further, we discussed the importance of various features in defining the binding affinity and in turn, the specificity in protein–protein complexes with reference to previous reports from literature. Finally we suggested the scope and need for developing more efficient state-of-the-art methods for deriving highly reliable inferences.

Acknowledgements

KY thanks the University Grants Commission (UGC), Government of India, for providing research fellowship. We thank the Bioinformatics facility and Indian Institute of Technology Madras for computational facilities. The work was partially supported by the Department of Science and Technology, India, to MMG (SR/SO/BB-0036/2011).

References

1. Nooren IM, Thornton JM (2003) Diversity of protein-protein interactions. EMBO J 22:3486–3492
2. Sudha G, Nussinov R, Srinivasan N (2014) An overview of recent advances in structural bioinformatics of protein-protein interactions and a guide to their principles. Prog Biophys Mol Biol 116:141–150
3. Gromiha MM (2010) Protein bioinformatics: from sequence to function. Elsevier, New Delhi
4. Bahadur RP, Chakrabarti P, Rodier F, Janin J (2004) A dissection of specific and non-specific protein–protein interfaces. J Mol Biol 336:943–955
5. Gromiha MM, Yokota K, Fukui K (2009) Energy based approach for understanding the recognition mechanism in protein–protein complexes. Mol Biosyst 5:1779–1786
6. Jones S, Thornton JM (1997) Prediction of protein–protein interaction sites using patch analysis. J Mol Biol 272:133–143
7. Neuvirth H, Raz R, Schreiber G (2004) ProMate: a structure based prediction program to identify the location of protein–protein binding sites. J Mol Biol 338:181–199
8. Fernandez-Recio J, Totrov M, Abagyan R (2004) Identification of protein- protein interaction sites from docking energy landscapes. J Mol Biol 335:843–865
9. Fernandez-Recio J, Totrov M, Skorodumov C, Abagyan R (2005) Optimal docking area: a new method for predicting protein–protein interaction sites. Proteins 58:134–143
10. La D, Kihara D (2012) A novel method for protein–protein interaction site prediction using phylogenetic substitution models. Proteins 80:126–141
11. La D, Kong M, Hoffman W, Choi YI, Kihara D (2013) Predicting permanent and transient protein-protein interfaces. Proteins 81:805–818
12. Jordan RA, Yasser EM, Dobbs D, Honavar V (2012) Predicting protein-protein interface residues using local surface structural similarity. BMC Bioinformatics 13:41

13. Ofran Y, Rost B (2003) Predict protein-protein interaction sites from local sequence information. FEBS Lett 544:236–239
14. Ofran Y, Rost B (2007) ISIS: interaction sites identified from sequence. Bioinformatics 23:e13–e16
15. Ahmad S, Mizuguchi K (2011) Partner-aware prediction of interacting residues in protein–protein complexes from sequence data. PLoS One 6:e29104
16. Shoemaker BA, Panchenko AR (2007) Deciphering protein–protein interactions. Part II. Computational methods to predict protein and domain interaction partners. Plos Comput Biol 3:595–601
17. Tuncbag N, Gursoy A, Keskin O (2011) Prediction of protein–protein interactions: unifying evolution and structure at protein interfaces. Phys Biol 8:035006
18. Martin S, Roe D, Faulon JL (2005) Predicting protein–protein interactions using signature products. Bioinformatics 21:218–226
19. Pan XY, Zhang YN, Shen HB (2010) Large-scale prediction of human protein– protein interactions from amino acid sequence based on latent topic features. J Proteome Res 9:4992–5001
20. Zhang YN, Pan XY, Huang Y, Shen HB (2011) Adaptive compressive learning for prediction of protein–protein interactions from primary sequence. J Theor Biol 283:44–52
21. Ramos H, Shannon P, Brusniak MY, Kusebauch U, Moritz RL, Aebersold R (2001) The protein information and property explorer 2: gaggle-like exploration of biological proteomic data within one webpage. Proteomics 11:154–158
22. Corney DP, Buxton BF, Langdon WB, Jones DT (2004) BioRAT: extracting biological information from full-length papers. Bioinformatics 20:3206–3213
23. Rzhetsky A, Iossifov I, Koike T, Krauthammer M, Kra P, Morris M, Yu H, Duboue PA, Weng W, Wilbur WJ, Hatzivassiloglou V, Friedman C (2004) GeneWays: a system for extracting, analyzing, visualizing, and integrating molecular pathway data. J Biomed Inform 37:43–53
24. Tsuruoka Y, Miwa M, Hamamoto K, Tsujii JI, Ananiadou S (2011) Discovering and visualizing indirect associations between biomedical concepts. Bioinformatics 27:i111–i119
25. Elefsinioti A, Saraç ÖS, Hegele A, Plake C, Hubner NC, Poser I, Sarov M, Hyman A, Mann M, Schroeder M, Stelzl U, Beyer A (2011) Large-scale de novo prediction of physical protein-protein association. Mol Cell Proteomics 10:M111–M010629
26. Tudor CO, Arighi CN, Wang Q, Wu CH, Vijay-Shanker K (2012) The eFIP system for text mining of protein interaction networks of phosphorylated proteins. Database bas044
27. Yang Z, Zhao Z, Li Y, Hu Y, Lin H (2013) PPIExtractor: a protein interaction extraction and visualization system for biomedical literature. IEEE Trans Nanobiosci 12:173–181
28. Subramani S, Kalpana R, Monickaraj PM, Natarajan J (2015) HPIminer: a text mining system for building and visualizing human protein interaction networks and pathways. J Biomed Inform 54:121–131
29. Papanikolaou N, Pavlopoulos GA, Theodosiou T, Iliopoulos I (2015) Protein-protein interaction predictions using text mining methods. Methods 74:47–53
30. Skrabanek L, Saini H, Bader G, Enright A (2008) Computational prediction of protein–protein interactions. Mol Biotechnol 38:1–17
31. Tamames J, Casari G, Ouzounis C, Valencia A (1997) Conserved clusters of functionally related genes in two bacterial genomes. J Mol Evol 44:66–73
32. Dandekar T, Snel B, Huynen M, Bork P (1998) Conservation of gene order: a fingerprint of proteins that physically interact. Trends Biochem Sci 23:324–328
33. Overbeek R, Fonstein M, D'souza M, Pusch GD, Maltsev N (1999) The use of gene clusters to infer functional coupling. Proc Natl Acad Sci U S A 96:2896–2901
34. Bowers PM, Pellegrini M, Thompson MJ, Fierro J, Yeates TO, Eisenberg D (2004) Prolinks: a database of protein functional linkages derived from coevolution. Genome Biol 5:R35
35. Muley VY, Ranjan A (2012) Effect of reference genome selection on the performance of computational methods for genome-wide protein-protein interaction prediction. PLoS One 7:e42057
36. Bock JR, Gough DA (2001) Predicting protein–protein interactions from primary structure. Bioinformatics 17:455–460
37. Shen J, Zhang J, Luo X, Zhu W, Yu K, Chen K, Li Y, Jiang H (2007) Predicting protein–protein interactions based only on sequences information. Proc Natl Acad Sci U S A 104:4337–4341
38. Guo Y, Yu L, Wen Z, Li M (2008) Using support vector machine combined with auto covariance to predict protein–protein interactions from protein sequences. Nucleic Acids Res 36:3025–3030
39. Pitre S, North C, Alamgir M, Jessulat M, Chan A, Luo X, Green JR, Dumontier M,

Dehne F, Golshani A (2008) Global investigation of protein–protein interactions in yeast Saccharomyces cerevisiae using re-occurring short polypeptide sequences. Nucleic Acids Res 36:4286–4294

40. Yu C-Y, Chou L-C, Chang DTH (2010) Predicting protein-protein interactions in unbalanced data using the primary structure of proteins. BMC Bioinformatics 11:167
41. Zhao CY, Jiang M (2014) Predicting protein-protein interactions from protein sequences using probabilistic neural network and feature combination. J Inform Comput Sci 11:2397–2406
42. You ZH, Lei YK, Zhu L, Xia J, Wang B (2013) Prediction of protein-protein interactions from amino acid sequences with ensemble extreme learning machines and principal component analysis. BMC Bioinformatics 14(Suppl 8):S10
43. You ZH, Li J, Gao X, He Z, Zhu L, Lei YK, Ji Z (2015) Detecting protein-protein interactions with a novel matrix-based protein sequence representation and support vector machines. BioMed Res Int. doi:10.1155/2015/867516
44. Pawson T, Gish GD, Nash P (2001) SH2 domains, interaction modules and cellular wiring. Trends Cell Biol 11:504–511
45. Pawson T, Nash P (2003) Assembly of cell regulatory systems through protein interaction domains. Science 300:445–452
46. Sprinzak E, Margalit H (2001) Correlated sequence-signatures as markers of protein-protein interaction. J Mol Biol 311:681–692
47. Deng M, Mehta S, Sun F, Chen T (2002) Inferring domain-domain interactions from protein-protein interactions. Genome Res 12:1540–1548
48. Wojcik J, Boneca IG, Legrain P (2002) Prediction, assessment and validation of protein interaction maps in bacteria. J Mol Biol 323:763–770
49. Ye Y, Godzik A (2004) Comparative analysis of protein domain organization. Genome Res 14:343–353
50. Liu S, Zhang C, Zhou Y (2005) Domain graph of Arabidopsis proteome by comparative analysis. J Proteome Res 4:435–444
51. Kim I, Liu Y, Zhao H (2007) Bayesian methods for predicting interacting protein pairs using domain information. Biometrics 63:824–833
52. Hayashida M, Akutsu T (2014) Domain-based approaches to prediction and analysis of protein-protein interactions. Int J Knowl Discov Bioinformatics 4:24–41
53. Schelhorn S-E, Lengauer T, Albrecht M (2008) An integrative approach for predicting interactions of protein regions. Bioinformatics 24:i35–i41
54. Smith GR, Sternberg MJE (2002) Prediction of protein-protein interactions by docking methods. Curr Opin Struct Biol 12:28–35
55. Lu L, Arakaki AK, Lu H, Skolnick J (2003) Multimeric threading-based prediction of protein-protein interactions on a genomic scale: application to the Saccharomyces cerevisiae proteome. Genome Res 13:1146–1154
56. Liu S, Zhang C, Zhou H, Zhou Y (2004) A physical reference state unifies the structure-derived potential of mean force for protein folding and binding. Proteins 56:93–101
57. Aloy P, Russell RB (2006) Structural systems biology: modelling protein interactions. Nat Rev Mol Cell Biol 7:188–197
58. Ohue M, Matsuzaki Y, Uchikoga N, Ishida T, Akiyama Y (2014) MEGADOCK: an all-to-all protein-protein interaction prediction system using tertiary structure data. Protein Pept Lett 21:766
59. Aloy P, Russell RB (2002) Interrogating protein interaction networks through structural biology. Proc Natl Acad Sci U S A 99:5896–5901
60. Aloy P, Ceulemans H, Stark A, Russell RB (2003) The relationship between sequence and interaction divergence in proteins. J Mol Biol 332:989–998
61. Ge H, Liu Z, Church GM, Vidal M (2001) Correlation between transcriptome and interactome mapping data from Saccharomyces cerevisiae. Nat Genet 29:482–486
62. Grigoriev A (2001) A relationship between gene expression and protein interactions on the proteome scale: analysis of the bacteriophage T7 and the yeast Saccharomyces cerevisiae. Nucleic Acids Res 29:3513–3519
63. Jansen R, Greenbaum D, Gerstein M (2002) Relating whole-genome expression data with protein-protein interactions. Genome Res 12:37–46
64. Krishnadev O, Srinivasan N (2008) A data integration approach to predict host-pathogen protein-protein interactions: application to recognize protein interactions between human and a malarial parasite. In Silico Biol 8:235–250
65. Ofran Y (2009) Prediction of protein interaction sites. In: Computational protein-protein interactions. CRC Press, Boca Raton, FL, pp 167–184
66. Porollo A, Meller J (2007) Prediction-based fingerprints of protein–protein interactions. Proteins 66:630–645

67. Chakravarty D, Guharoy M, Robert CH, Chakrabarti P, Janin J (2013) Reassessing buried surface areas in protein–protein complexes. Protein Sci 22:1453–1457
68. Tuncbag N, Kar G, Keskin O, Gursoy A, Nussinov R (2009) A survey of available tools and web servers for analysis of protein–protein interactions and interfaces. Brief Bioinform 10(3):217–232
69. Bradford JR, Westhead DR (2005) Improved prediction of protein-protein binding sites using a support vector machines approach. Bioinformatics 21:1487–1494
70. Bradford JR, Needham CJ, Bulpitt AJ, Westhead DR (2006) Insights into protein-protein interfaces using a Bayesian network prediction method. J Mol Biol 362:365–386
71. Fariselli P, Pazos F, Valencia A, Casadio R (2002) Prediction of protein-protein interaction sites in heterocomplexes with neural networks. Eur J Biochem 269:1356–1361
72. Bordner AJ, Abagyan R (2005) Statistical analysis and prediction of protein-protein interfaces. Proteins 60:353–366
73. Chen H, Zhou H-X (2005) Prediction of interface residues in protein-protein complexes by a consensus neural network method: test against NMR data. Proteins 61:21–35
74. DeVries SJ, VanDijk ADJ, Bonvin AMJJ (2006) WHISCY: what information does surface conservation yield? Application to data-driven docking. Proteins 63:479–489
75. Liang S, Zhang C, Liu S, Zhou Y (2006) Protein binding site prediction using an empirical scoring function. Nucleic Acids Res 34:3698–3707
76. Kufareva I, Budagyan L, Raush E, Totrov M, Abagyan R (2007) PIER: protein interface recognition for structural proteomics. Proteins 67:400–417
77. Negi SS, Schein CH, Oezguen N, Power TD, Braun W (2007) InterProSurf: a web server for predicting interacting sites on protein surfaces. Bioinformatics 23:3397–3399
78. Qin S, Zhou HX (2007) meta-PPISP: a meta web server for protein-protein interaction site prediction. Bioinformatics 23:3386–3387
79. Murga LF, Ondrechen MJ, Ringe D (2008) Prediction of interaction sites from apo 3D structures when the holo conformation is different. Proteins 72:980–992
80. Segura J, Jones PF, Fernandez-Fuentes N (2011) Improving the prediction of protein binding sites by combining heterogeneous data and Voronoi diagrams. BMC Bioinformatics 12:352
81. Zhang QC, Deng L, Fisher M, Guan J, Honig B, Petrey D (2011) PredUs: a web server for predicting protein interfaces using structural neighbors. Nucleic Acids Res 39:W283–W287
82. Maheshwari S, Brylinski M (2015) Template-based identification of protein–protein interfaces using eFindSite PPI. Methods. doi:10.1016/j.ymeth.2015.07.017
83. Maheshwari S, Brylinski M (2015) Predicting protein interface residues using easily accessible on-line resources. Brief Bioinformatics. doi:10.1093/bib/bbv009
84. Kini RM, Evans HJ (1996) Prediction of potential protein-protein interaction sites from amino acid sequence. Identification of a fibrin polymerization site. FEBS Lett 385:81–86
85. Gallet X, Charloteaux B, Thomas A, Brasseur R (2000) A fast method to predict protein interaction sites from sequences. J Mol Biol 302:917–926
86. Murakami Y, Mizuguchi K (2010) Applying the Naïve Bayes classifier with kernel density estimation to the prediction of protein-protein interaction sites. Bioinformatics 26:1841–1848
87. Dhole K, Singh G, Pai PP, Mondal S (2014) Sequence-based prediction of protein–protein interaction sites with L1-logreg classifier. J Theor Biol 348:47–54
88. Xue LC, Dobbs D, Honavar V (2011) HomPPI: a class of sequence homology based protein-protein interface prediction methods. BMC Bioinformatics 12:244
89. Chothia C, Janin J (1975) Principles of protein-protein recognition. Nature 256:705–708
90. Horton N, Lewis M (1992) Calculation of the free energy of association for protein complexes. Protein Sci 1:169–181
91. Ma XH, Wang CX, Li CH, Chen WZ (2002) A fast empirical approach to binding free energy calculations based on protein interface information. Protein Eng 15:677–681
92. Jiang L, Gao Y, Mao F, Liu Z, Lai L (2002) Potential of mean force for protein-protein interaction studies. Proteins 46:190–196
93. Audie J, Scarlata S (2007) A novel empirical free energy function that explains and predicts protein-protein binding affinities. Biophys Chem 129:198–211
94. Su Y, Zhou A, Xia X, Li W, Sun Z (2009) Quantitative prediction of protein-protein binding affinity with a potential of mean force considering volume correction. Protein Sci 18:2550–2558

95. Kastritis PL, Bonvin AMJJ (2010) Are scoring functions in protein-protein docking ready to predict interactomes? Clues from a novel binding affinity benchmark. J Proteome Res 9:2216–2225
96. Kastritis PL, Moal IH, Hwang H, Weng Z, Bates PA, Bonvin AMJJ, Janin J (2011) A structure-based benchmark for protein-protein binding affinity. Protein Sci 20:482–491
97. Moal IH, Agius R, Bates PA (2011) Protein-protein binding affinity prediction on a diverse set of structures. Bioinformatics 27:3002–3009
98. Vreven T, Hwang H, Pierce BG, Weng Z (2012) Prediction of protein-protein binding free energies. Protein Sci 21:396–404
99. Tian F, Lv Y, Yang L (2012) Structure-based prediction of protein-protein binding affinity with consideration of allosteric effect. Amino Acids 43:531–543
100. Zhou P, Wang C, Tian F, Ren Y, Yang C, Huang J (2013) Biomacromolecular quantitative structure-activity relationship (BioQSAR): a proof-of-concept study on the modeling, prediction and interpretation of protein-protein binding affinity. J Comput Aided Mol Des 27:67–78
101. Kastritis PL, Bonvin AMJJ (2013) On the binding affinity of macromolecular interactions: daring to ask why proteins interact. J R Soc Interface 10:20120835
102. Vangone A, Bonvin AMJJ (2015) Contacts-based prediction of binding affinity in protein–protein complexes. eLife 4:e07454
103. Yugandhar K, Gromiha MM (2014) Feature selection and classification of protein-protein complexes based on their binding affinities using machine learning approaches. Proteins 82:2088–2096
104. Yugandhar K, Gromiha MM (2015) Analysis of protein-protein interaction networks based on binding affinity. Curr Protein Pept Sci 17:72–81
105. Yugandhar K, Gromiha MM (2014) Protein-protein binding affinity prediction from amino acid sequence. Bioinformatics 30:3583–3589
106. Yugandhar K, Gromiha MM (2015) Response to the comment on "protein-protein binding affinity prediction from amino acid sequence". Bioinformatics 31:978
107. Swapna LS, Bhaskara RM, Sharma J, Srinivasan N (2012) Roles of residues in the interface of transient protein-protein complexes before complexation. Sci Rep 2:1–9
108. Janin J (2014) A minimal model of protein-protein binding affinities. Protein Sci 23:1813–1817
109. Kastritis PL, Rodrigues JPGLM, Folkers GE, Boelens R, Bonvin AMJJ (2014) Proteins feel more than they see: fine-tuning of binding affinity by properties of the non-interacting surface. J Mol Biol 426:2632–2652
110. Cunningham BC, Wells JA (1989) High-resolution epitope mapping of hGH-receptor interactions by alanine-scanning mutagenesis. Science 244:1081–1085
111. Bogan AA, Thorn KS (1998) Anatomy of hot spots in protein interfaces. J Mol Biol 280:1–9
112. Cunningham BC, Wells JA (1993) Comparison of a structural and a functional epitope. J Mol Biol 234:554–563
113. Clackson T, Wells JA (1995) A hot spot of binding energy in a hormone-receptor interface. Science 267:383–386
114. Hu Z, Ma B, Wolfson H, Nussinov R (2000) Conservation of polar residues as hot spots at protein interfaces. Proteins 39:331–342
115. DeLano WL (2002) Unraveling hot spots in binding interfaces: Progress and challenges. Curr Opin Struct Biol 12:14–20
116. Halperin I, Wolfson H, Nussinov R (2004) Protein-protein interactions: coupling of structurally conserved residues and of hot spots across interfaces. Implications for docking. Structure 12:1027–1038
117. Janin J (2009) Basic principles of protein-protein interaction. In: Computational protein-protein interactions. CRC Press, Boca Raton, pp 1–19
118. Monod J, Changeux J-P, Jacob F (1963) Allosteric proteins and cellular control systems. J Mol Biol 6:306–329
119. Kuriyan J, Eisenberg D (2007) The origin of protein interactions and allostery in colocalization. Nature 450:983–990
120. Salwinski L, Miller CS, Smith AJ, Pettit FK, Bowie JU, Eisenberg D (2004) The database of interacting proteins: 2004 update. Nucleic Acids Res 32(Suppl 1):D449–D451
121. Stark C, Breitkreutz BJ, Reguly T, Boucher L, Breitkreutz A, Tyers M (2006) BioGRID: a general repository for interaction datasets. Nucleic Acids Res 34(Suppl 1):D535–D539
122. Prasad TSK, Goel R, Kandasamy K, Keerthikumar S, Kumar S, Mathivanan S, Telikicherla D, Raju R, Shafreen B, Venugopal A, Balakrishnan L, Marimuthu A, Banerjee S, Somanathan DS, Sebastian A, Rani S, Ray S, Kishore CJH, Kanth S, Ahmed M, Kashyap MK, Mohmood R, Ramachandra YL, Krishna V, Rahiman BA, Mohan S, Ranganathan P, Ramabadran S, Chaerkady R, Pandey A (2009) Human protein reference

database-2009 update. Nucleic Acids Res 37:D767–D772

123. Szklarczyk D, Franceschini A, Kuhn M, Simonovic M, Roth A, Minguez P, von Mering C (2011) The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored. Nucleic Acids Res 39(Suppl 1):D561–D568
124. Orchard S, Ammari M, Aranda B, Breuza L, Briganti L, Broackes-Carter F, Campbell NH, Chavali G, Chen C, del-Toro N, Duesbury M, Dumousseau M, Galeota E, Hinz U, Iannuccelli M, Jagannathan S, Jimenez R, Khadake J, Lagreid A, Licata L, Lovering R C, Meldal B, Melidoni AN, Milagros M, Peluso D, Perfetto L, Porras P, Raghunath A, Ricard-Blum S, Roechert B, Stutz A, Tognolli M, van Roey K, Cesareni G, Hermjakob H (2013) The MIntAct project—IntAct as a common curation platform for 11 molecular interaction databases. Nucleic Acids Res gkt1115
125. deBeer TAP, Berka K, Thornton JM, Laskowski RA (2014) PDBsum additions. Nucleic Acids Res 42:D292–D296
126. Kumar MS, Gromiha MM (2006) PINT: protein-protein interactions thermodynamic database. Nucleic Acids Res 34(suppl 1):D195–D198
127. Wang R, Fang X, Lu Y, Yang CY, Wang S (2005) The PDBbind database: methodologies and updates. J Med Chem 48:4111–4119
128. Kumar P, Han B-C, Shi Z, Jia J, Wang YP, Zhang YT, Liang L, Liu QF, Ji ZL, Chen YZ (2009) Update of KDBI: kinetic data of biomolecular interaction database. Nucleic Acids Res 37:D636–D641
129. Moal IH, Fernández-Recio J (2012) SKEMPI: a structural kinetic and energetic database of mutant protein interactions and its use in empirical models. Bioinformatics 28:2600–2607
130. Thorn KS, Bogan AA (2001) ASEdb: a database of alanine mutations and their effects on the free energy of binding in protein interactions. Bioinformatics 17:284–285
131. Vreven T, Moal IM, Vangone A, Pierce BG, Kastritis PL, Torchala M, Chaleil R, Jimenez-Garcia B, Bates PA, Fernandez-Recio J, Bonvin AMJJ, Weng Z (2015) Updates to the integrated protein–protein interaction benchmarks: docking benchmark version 5 and affinity benchmark version 2. J Mol Biol 427:3031–3041

Chapter 17

In Silico Prediction of Linear B-Cell Epitopes on Proteins

Yasser EL-Manzalawy, Drena Dobbs, and Vasant G. Honavar

Abstract

Antibody-protein interactions play a critical role in the humoral immune response. B-cells secrete antibodies, which bind antigens (e.g., cell surface proteins of pathogens). The specific parts of antigens that are recognized by antibodies are called B-cell epitopes. These epitopes can be *linear*, corresponding to a contiguous amino acid sequence fragment of an antigen, or *conformational*, in which residues critical for recognition may not be contiguous in the primary sequence, but are in close proximity within the folded protein 3D structure.

Identification of B-cell epitopes in target antigens is one of the key steps in epitope-driven subunit vaccine design, immunodiagnostic tests, and antibody production. In silico bioinformatics techniques offer a promising and cost-effective approach for identifying potential B-cell epitopes in a target vaccine candidate. In this chapter, we show how to utilize online B-cell epitope prediction tools to identify linear B-cell epitopes from the primary amino acid sequence of proteins.

Key words Antibody-protein interaction, B-cell epitope prediction, Linear B-cell epitope prediction, Epitope mapping, Epitope prediction

1 Introduction

Antibodies, which are glycoproteins produced in membrane-bound or secreted form by B lymphocytes, mediate specific humoral immunity by engaging various effector mechanisms that serve to eliminate the bound antigens [1]. The characterization of antibody-protein interactions has been the focus of extensive research. This work has advanced our understanding of the adaptive immune system and contributed to important practical applications, such as identifying subunit vaccine targets [2, 3]. When an antibody binds to a protein, the resulting binding sites in the antibody and the protein are called the paratope and epitope, respectively. Among the several experimental methods for mapping B-cell epitopes and paratopes [2, 3], X-ray crystallography is perhaps the most preferred method because of its accuracy. Due to the high cost and technical challenges presented by experimental methods for mapping epitopes and paratopes, there is an urgent need for reliable in silico methods for identifying binding sites in antibody-protein complexes [4].

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_17,

B-cell epitopes are classified as either linear or conformational. Linear epitopes are fragments of continuous amino acids in the protein sequence. Conformational epitopes consist of amino acid residues that may be separated in the protein primary sequence, but are brought into physical proximity via protein folding. Although more than 90% of epitopes are estimated to be conformational [5], most experimental studies and computational methods focus on mapping linear B-cell epitopes.

In this chapter, we discuss different computational methods for predicting linear and conformational B-cell epitopes and outline procedures for in silico identification of linear B-cell epitopes from amino acid sequence. Because the predictive performance of individual linear B-cell prediction methods is far from satisfactory, we propose a procedure that combines predictions from multiple predictors to obtain more reliable consensus predictions. Our approach also uses known or predicted 3D structures of target proteins to filter out false predictions. Due to the very limited availability of *sequence-based* conformational B-cell epitope prediction tools, consensus predictions are not currently feasible at present. However, with anticipated increase in the amount of experimental data, further advances in predicting conformational epitopes can be expected.

2 Materials

2.1 Data

In this protocol, the query is the primary sequence of a target protein (e.g., vaccine candidate). This vaccine candidate may be determined based on a literature survey (e.g., [6]) or using reverse vaccinology tools [7–9]. In some cases, the user may focus on protein fragments reported in literature or found to be conserved based on a multiple sequence alignment of the target protein sequences from multiple strains of the pathogen.

2.2 Linear B-Cell Epitope Prediction Tools

Early computational methods for mapping linear B-cell epitopes in an amino acid sequence assumed some correlation between a certain single physicochemical property of an amino acid (e.g., hydrophilicity, flexibility, or solvent accessibility propensity) and the likelihood that the amino acid would be part of a linear B-cell epitope [10–12]. BcePred [13] predicts linear B-cell epitopes using a combination of physicochemical properties as opposed to propensity measures based on a single amino acid property. BepiPred [14] combines the hydrophilicity scale proposed by Parker et al. [12] with a Hidden Markov Model (HMM) predictor. All these methods provide *residue-based* predictions, in that they assign a score to each residue in the query protein sequence; the higher the score assigned to a residue is, the more likely it belongs to a linear B-cell epitope (*see* Fig. 1 for an example).

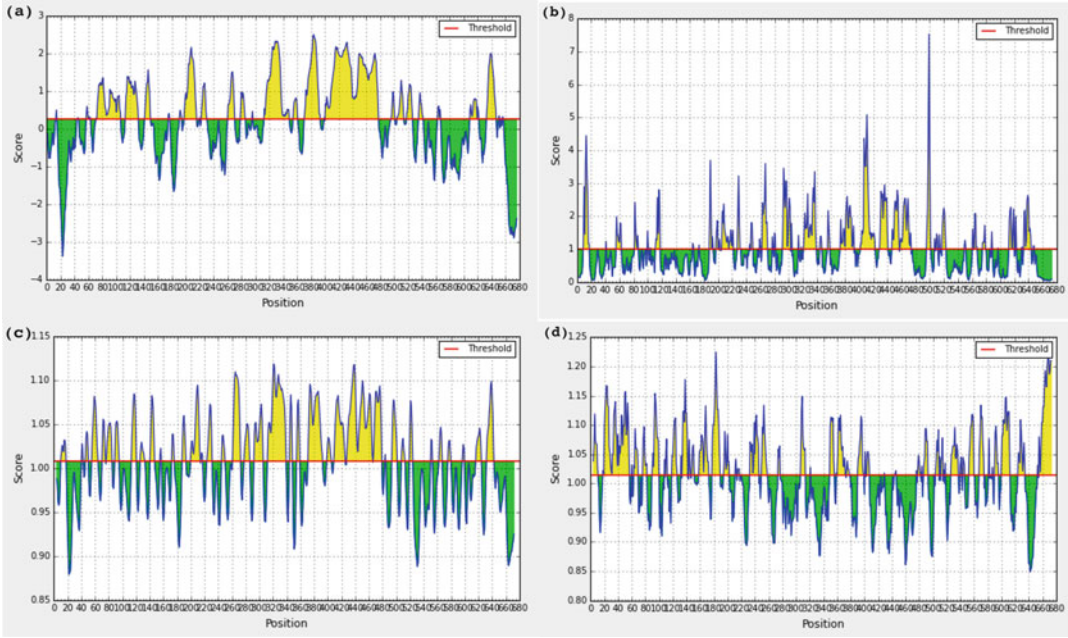


Fig. 1 Propensity scale profiles for the Ebola virus GP protein (UniProt ID Q05320) generated using (**a**) BepiPred, (**b**) surface accessibility, (**c**) flexibility, (**d**) antigenicity. Regions with scores above the *red line* are more likely to contain linear B-cell epitopes

Alternatively, several machine learning methods classify amino acid peptide chains of specific lengths as either epitopes or non-epitopes. BCPred [15] predicts linear B-cell epitopes of length 12, 14, 16, 18, 20, 22 amino acids using a Support Vector Machine (SVM) classifier and a string kernel. FBCPred [16] is a variant of BCPred for predicting linear B-cell epitopes of virtually any length. COBEpro [17] uses a two-stage procedure for predicting linear B-cell epitopes. In the first stage, an SVM classifier is used to assign scores to fragments of the query antigen. In the second stage, a prediction score is assigned to each residue in the query antigen based on the SVM scores for the peptide fragments. LBtope [18] provides improved predictions of linear B-cell epitopes by training classifiers using experimentally validated *non*-epitopes, whereas all previous methods used randomly sampled fragments from UniProt as the non-epitope training data. Recently, we showed that further improvements in the reliability of linear B-cell epitope predictions can be obtained by using ensemble classifiers that combine multiple linear B-cell epitope predictors [19].

2.3 Conformational B-Cell Epitope Prediction Tools

The problem of conformational B-cell epitope prediction can be defined as follows: Given the primary or the tertiary structure of a query protein, what are the interfacial residues involved in the complex formed between the query protein and an antibody.

This is essentially a subproblem of the more general problem of protein-protein interface prediction [20, 21], where the goal is to identify interfacial residues in a query protein that form a complex with any other protein (including antibodies). Unfortunately, protein-protein interface predictors trained on large data sets of protein-protein interfaces are not sufficiently reliable for predicting antibody-protein interfaces [22].

Partly due to the small number of solved antibody-protein structures, relatively few methods for predicting conformational B-cell epitopes have been proposed in the literature. The performance of the available methods remains far from satisfactory [4, 22]. Table 1 summarizes current B-cell epitope prediction methods that are available in the form of freely accessible web servers or downloadable software packages. In this table, we have categorized B-cell epitope prediction methods as *sequence-based* or *structure-based*, according to whether the method accepts the primary sequence vs. the 3D structural coordinates of the query protein as input. We have also categorized the methods as *residue-based* or *patch-based*. *Residue-based* methods return a prediction score for each residue in the query protein. *Patch-based* methods decompose the surface of the query protein into patches and return a single prediction score for each patch. Each patch could be interpreted as an epitope of an antibody-protein complex.

The vast majority of available tools for predicting conformational B-cell epitopes are *structure-based* in that they require the solved/predicted unbound structure of the target protein as input to the predictor. Hence, their applicability is limited by the availability of an experimentally determined 3D structure (from the PDB

Table 1

Summary of antibody-protein binding site (conformational B-cell epitope) online prediction tools

Tool	URL of web server	Comments
CBTOPE	http://www.imtech.res.in/raghava/cbtope/	Sequence-based, residue-based
DiscoTope	http://www.cbs.dtu.dk/services/DiscoTope/	Structure-based, residue-based
ElliPro	http://tools.immuneepitope.org/ellipro/	Structure-based, residue-based
EPCES	http://sysbio.unl.edu/EPCES/	Structure-based, patch-based
Epitopia	http://epitopia.tau.ac.il/	Structure-based, residue-based
EPSVR	http://sysbio.unl.edu/EPSVR/	Structure-based, patch-based
PEPITO	http://pepito.proteomics.ics.uci.edu/	Structure-based, residue-based
SEPPA	http://lifecenter.sgst.cn/seppa2/	Structure-based, residue-based

[23]) or a homology model for the query protein (*see* **Note 1**). To address this limitation, BEST [24] and CBTOPE [25] have been proposed for predicting conformational B-cell epitopes using amino acid derived information.

All of the methods described in Table 1 are antibody-independent B-cell epitope prediction methods [26], in the sense that they do not take advantage of information about the binding antibody in predicting the antibody binding site on the antigen. Recently, some antibody-specific B-cell epitope prediction methods have been proposed (*see* **Note 2**). Antibody-specific B-cell epitope prediction methods are motivated in part by: (1) the success of partner-specific protein-protein interface predictors [27, 28] and allele-specific major histocompatibility complex (MHC) binding site predictors [29, 30]; and (2) the observation that virtually any surface accessible region of an antigen can become the target of *some* antibody and elicit an immune response [26, 31] and hence it is much more useful to focus on the binding site for a specific antibody.

3 Methods

In this section, we focus on *sequence-based* tools for identifying linear B-cell epitopes.

3.1 Predicting Linear B-Cell Epitopes

Given the amino acid sequence of a protein of interest, apply the following procedure to obtain a list of predicted linear B-cell epitopes within the query sequence:

1. Go to submission page of BCPREDS server (*see* Fig. 2) accessible at http://ailab.ist.psu.edu/bcpred/predict.html.
2. Paste the amino acid sequence of the target protein.
3. Select the prediction method. The server currently supports three methods: BCPred [15], AAP [32], and FBCPred [16]. The user is encouraged to try all three methods (*see* **step 9**).
4. Select the length of the epitope. BCPred and APP methods can handle queries for a set of prespecified lengths (12, 14, 16, 18, 20, 22). FBCPred predicts linear B-cell epitopes of any length specified by the user. Some tips and guidelines for deciding on epitope length are provided in **Note 3**.
5. Select the specificity of the classifier (*see* **Note 4**).
6. Uncheck "report only non-overlapping epitopes" if you want the server to report all predicted epitopes with probability greater than the cut-off corresponding to the select classifier specificity in **step 6**. Otherwise, highly ranked non-overlapping epitopes will be also reported (*see* **Note 5**).
7. Click "Submit query" to obtain predicted epitopes in the query sequence.

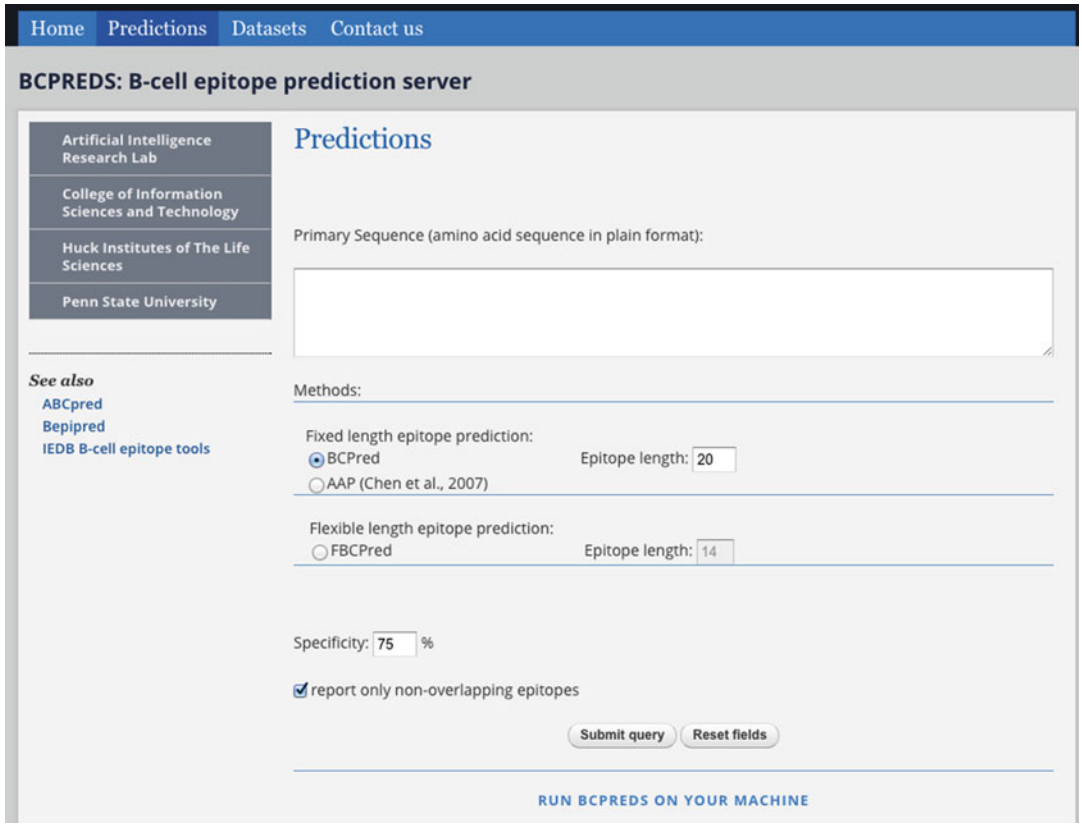


Fig. 2 Submission page of BCPREDS web server available at: http://ailab.ist.psu.edu/bcpred/predict.html

8. Repeat **steps 1–8** for other supported prediction methods. Discard epitopes predicted by only a single method. The intuition behind this is that consensus predictions are usually more reliable than predictions obtained from a single prediction method.
9. Figure 3 shows the output of BCPREDS, in which non-overlapping epitopes predicted by the three prediction methods are combined and consensus predictions are identified (**bold** residues in the sequence).
10. Users are also encouraged to consider predictions by other servers (e.g., COBEPro [17]) by following essentially the same procedure described here to submit queries.
11. Evaluating the results: If possible, the user should filter out likely "false positives," i.e., predicted epitopes that do not lie on the surface of the protein by mapping the predicted epitopes onto a solved or predicted 3D structure of the query protein (*see* **Note 6**). In addition, the user might use the Immune Epitopes Database Analysis Resource (IEDB-AR) [33] to generate propensity scale profiles for the query protein

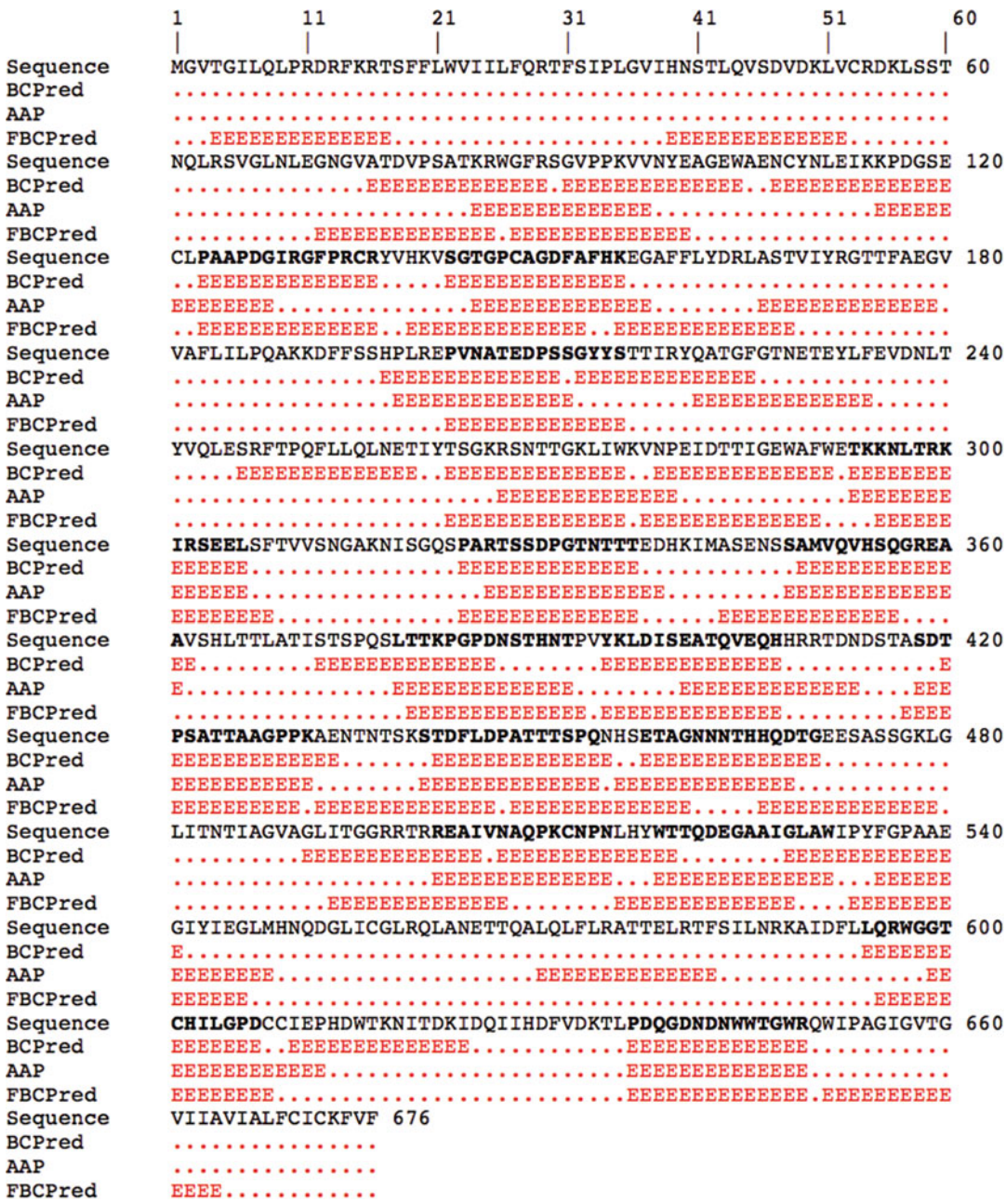


Fig. 3 Linear B-cell epitopes predicted using three different linear B-cell epitope predictors currently supported by BCPREDS: BCPred, AAP, and FBCPred. Bold residues indicate epitope residues predicted by at least two methods

(*see* **Note 7**). Although these profiles cannot provide reliable predictions of linear B-cell epitopes (*see* **Note 8**), they could be useful in highlighting potential antigenic regions of interest to confirm predictions by BCPREDS.

4 Notes

1. In the absence of solved 3D structure for a query protein, computational tools like I-TASSER [40] could be used to predict the 3D structure of that protein. I-TASSER is a template-based method for protein structure and function prediction. The pipeline consists of four major steps: template identification, structure reassembly, atomic model construction, and final model selection.
2. Antibody-specific B-cell epitope prediction methods take into account the binding *antibody* sequence or structure in order to predict conformational B-cell epitopes in a query antigen sequence of known structure. EpiPred [34] is a fully *structure-based* method that requires the structures of an antigen and its putative binding antibody. Bepar [35] and ABepar [36] are fully *sequence-based* methods that take the sequences of the interacting antigen and antibody as input. PEASE server [37] predicts conformational B-cell epitopes in an antigen of known structure, given the sequence of the binding antibody.
3. Deciding on optimal epitope length is not trivial. In fact existing tools cannot reliably predict optimal linear B-cell epitopes because most of the experimentally validated linear B-cell epitopes used to train these predictors are not optimal in length. However, it makes sense to use lengths between 12 and 16 amino acids because the lengths of known epitopes are within that range [15].
4. There is always a trade-off between specificity and sensitivity. Higher specificity means lower false positive rate at the expense of missing some true positives (i.e., epitopes). We recommend using low specificity cut-offs and combining predictions from several tools to eliminate false positive predictions.
5. A query protein sequence of L amino acids has L-k+1 potential linear B-cell epitopes of length equal k. BCPREDS predictors assign a score to every candidate epitope and report epitopes with scores higher than the cut-off corresponding to user-specified specificity. To eliminate highly overlapping predicted epitopes and identify antigenic regions, the user might configure the tool to show non-overlapping epitopes.
6. Interactive molecular viewers like JMol [38] and PyMol [39] take PDB coordinate files as input and allow user to visualize protein 3D structures and highlight particular amino acid residues and support scripts and plugins for other tasks (e.g., determine interface residues or finding and highlighting surface residues).
7. The Immune Epitopes Database Analysis Resource (IEDB-AR) B-cell tool available at http://tools.iedb.org/bcell/ generates

propensity scale profiles for submitted amino acid sequences using BepiPred [14] and five other propensity scales. Figure 1 shows example profiles generated for Ebola Virus GP protein (UniProt ID Q05320) using surface BepiPred and three propensity scales (accessibility [10], flexibility [11], and antigenicity [31]).

8. Blythe and Flower [37] have conducted a comprehensive assessment of about 500 amino acid physicochemical propensity scales in predicting linear B-cell epitopes (using a data set of 50 proteins) and showed that the performance of the best method is only slightly better than random guessing. This result was the main motivation of the machine learning-based methods for predicting linear B-cell epitopes.

Acknowledgments

This work was supported by NIH grant GM066387 to VGH and DD, by Edward Frymoyer Chair of Information Sciences and Technology at Pennsylvania State University to VGH, and by a Presidential Initiative for Interdisciplinary Research (PIIR) award from Iowa State University to DD.

References

1. Abbas AK, Lichtman AH, Pillai S (2014) Cellular and molecular immunology: with student consult online access. Elsevier Health Sciences, Philadelphia, PA
2. Abbott WM, Damschroder MM, Lowe DC (2014) Current approaches to fine mapping of antigen–antibody interactions. Immunology 142(4):526–535
3. Reineke U, Schutkowski M (2009) Epitope mapping protocols, vol 524, Methods in molecular biology. Humana Press, New York
4. EL-Manzalawy Y, Honavar V (2010) Recent advances in B-cell epitope prediction methods. Immunome Res Suppl 2:S2
5. Walter G (1986) Production and use of antibodies against synthetic peptides. J Immunol Methods 88(2):149–161
6. Wu X, Li X, Zhang Q, Wulin S, Bai X, Zhang T, Wang Y, Liu M, Zhang Y (2015) Identification of a conserved B-cell epitope on duck hepatitis A type 1 virus VP1 protein. PLoS One 10(2):e0118041
7. Palumbo E, Fiaschi L, Brunelli B, Marchi S, Savino S, Pizza M (2012) Antigen identification starting from the genome: a "Reverse Vaccinology" approach applied to MenB. In: Neisseria meningitidis: advanced methods and protocols. Methods in molecular biology, vol 799. Springer, pp 361–403
8. Donati C, Rappuoli R (2013) Reverse vaccinology in the 21st century: improvements over the original design. Ann N Y Acad Sci 1285(1):115–132
9. Xiang Z, He Y (2013) Genome-wide prediction of vaccine targets for human herpes simplex viruses using Vaxign reverse vaccinology. BMC Bioinformatics 14(Suppl 4):S2
10. Emini EA, Hughes JV, Perlow D, Boger J (1985) Induction of hepatitis A virus-neutralizing antibody by a virus-specific synthetic peptide. J Virol 55(3):836–839
11. Karplus P, Schulz G (1985) Prediction of chain flexibility in proteins. Naturwissenschaften 72(4):212–213
12. Parker J, Guo D, Hodges R (1986) New hydrophilicity scale derived from high-performance liquid chromatography peptide retention data: correlation of predicted surface residues with antigenicity and X-ray-derived accessible sites. Biochemistry 25(19):5425–5432
13. Saha S, Raghava G (2004) BcePred: prediction of continuous B-cell epitopes in antigenic sequences using physico-chemical properties. In: Artificial immune systems. Lecture notes in computer science, vol 3239. Springer, pp 197–204
14. Larsen J, Lund O, Nielsen M (2006) Improved method for predicting linear B-cell epitopes. Immunome Res 2(2):1–7

15. EL-Manzalawy Y, Dobbs D, Honavar V (2008) Predicting linear B-cell epitopes using string kernels. J Mol Recognit 21(4):243–255
16. EL-Manzalawy Y, Dobbs D, Honavar V (2008) Predicting flexible length linear B-cell epitopes. In: Computational systems bioinformatics. NIH Public Access, pp 121–132
17. Sweredoski MJ, Baldi P (2009) COBEpro: a novel system for predicting continuous B-cell epitopes. Protein Eng Design Select 22(3):113–120
18. Singh H, Ansari HR, Raghava GP (2013) Improved method for linear B-cell epitope prediction using Antigen's primary sequence. PLoS One 8(5):e62216
19. EL-Manzalawy Y, Honavar V (2014) Building classifier ensembles for B-cell epitope prediction. In: Immunoinformatics. Methods in molecular biology, vol 1184. Springer, pp 285–294
20. Esmaielbeiki R, Krawczyk K, Knapp B, Nebel J-C, Deane CM (2015) Progress and challenges in predicting protein interfaces. Brief Bioinformatics bbv027
21. Xue LC, Dobbs D, Bonvin A, Honavar V (2015) Protein-protein interface predictions by data-driven methods: a review. FEBS Lett 589(23):3516–3526
22. Yao B, Zheng D, Liang S, Zhang C (2013) Conformational B-cell epitope prediction on antigen protein structures: a review of current algorithms and comparison with common binding site prediction methods. PLoS One 8(4):e62249
23. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat T, Weissig H, Shindyalov IN, Bourne PE (2000) The protein data bank. Nucleic Acids Res 28(1):235–242
24. Gao J, Faraggi E, Zhou Y, Ruan J, Kurgan L (2012) BEST: improved prediction of B-cell epitopes from antigen sequences. PLoS One 7(6):e40104
25. Ansari HR, Raghava G (2010) Identification of conformational B-cell epitopes in an antigen from its primary sequence. Immunome Res 6(6):1–9
26. Sela-Culang I, Ofran Y, Peters B (2015) Antibody specific epitope prediction—emergence of a new paradigm. Curr Opin Virol 11:98–102
27. Xue LC, Dobbs D, Honavar V (2011) HomPPI: a class of sequence homology based protein-protein interface prediction methods. BMC Bioinformatics 12(1):244
28. Minhas A, ul Amir F, Geiss BJ, Ben-Hur A (2014) PAIRpred: partner-specific prediction of interacting residues from sequence and structure. Proteins 82(7):1142–1155
29. El-Manzalawy Y, Dobbs D, Honavar V (2011) Predicting MHC-II binding affinity using multiple instance regression. Comput Biol Bioinformatics IEEE/ACM Trans 8(4):1067–1079
30. Trolle T, Metushi IG, Greenbaum JA, Kim Y, Sidney J, Lund O, Sette A, Peters B, Nielsen M (2015) Automated benchmarking of peptide-MHC class I binding predictions. Bioinformatics btv123
31. Rubinstein ND, Mayrose I, Halperin D, Yekutieli D, Gershoni JM, Pupko T (2008) Computational characterization of B-cell epitopes. Mol Immunol 45(12):3477–3489
32. Chen J, Liu H, Yang J, Chou K-C (2007) Prediction of linear B-cell epitopes using amino acid pair antigenicity scale. Amino Acids 33(3):423–428
33. Zhang Q, Wang P, Kim Y, Haste-Andersen P, Beaver J, Bourne PE, Bui H-H, Buus S, Frankild S, Greenbaum J (2008) Immune epitope database analysis resource (IEDB-AR). Nucleic Acids Res 36(suppl 2):W513–W518
34. Krawczyk K, Liu X, Baker T, Shi J, Deane CM (2014) Improving B-cell epitope prediction and its application to global antibody-antigen docking. Bioinformatics 30(16):2288–2294
35. Zhao L, Li J (2010) Mining for the antibody-antigen interacting associations that predict the B cell epitopes. BMC Struct Biol 10(Suppl 1):S6
36. Zhao L, Wong L, Li J (2011) Antibody-specified B-cell epitope prediction in line with the principle of context-awareness. Comput Biol Bioinformatics IEEE/ACM Trans 8(6):1483–1494
37. Sela-Culang I, Benhnia MR-E-I, Matho MH, Kaever T, Maybeno M, Schlossman A, Nimrod G, Li S, Xiang Y, Zajonc D (2014) Using a combined computational-experimental approach to predict antibody-specific B cell epitopes. Structure 22(4):646–657
38. Herraez A (2006) Biomolecules in the computer: Jmol to the rescue. Biochem Mol Biol Educ 34(4):255–261
39. DeLano WL (2002) Pymol: an open-source molecular graphics tool. CCP4 Newslett Protein Crystallogr 40:82–92

Chapter 18

Prediction of Protein Phosphorylation Sites by Integrating Secondary Structure Information and Other One-Dimensional Structural Properties

Yongchao Dou, Bo Yao, and Chi Zhang

Abstract

Studies on phosphorylation are important but challenging for both wet-bench experiments and computational studies, and accurate non-kinase-specific prediction tools are highly desirable for whole-genome annotation in a wide variety of species. Here, we describe a phosphorylation site prediction webserver, PhosphoSVM, that employs Support Vector Machine to combine protein secondary structure information and seven other one-dimensional structural properties, including Shannon entropy, relative entropy, predicted protein disorder information, predicted solvent accessible area, amino acid overlapping properties, averaged cumulative hydrophobicity, and subsequence k-nearest neighbor profiles. This method achieved AUC values of 0.8405/0.8183/0.7383 for serine (S), threonine (T), and tyrosine (Y) phosphorylation sites, respectively, in animals with a tenfold cross-validation. The model trained by the animal phosphorylation sites was also applied to a plant phosphorylation site dataset as an independent test. The AUC values for the independent test data set were 0.7761/0.6652/0.5958 for S/T/Y phosphorylation sites, respectively. This algorithm with the optimally trained model was implemented as a webserver. The webserver, trained model, and all datasets used in the current study are available at http://sysbio.unl.edu/PhosphoSVM.

Key words Phosphorylation site prediction, Non-kinase-specific tool, Support vector machine

1 Introduction

Phosphorylation, one of the most essential posttranslational modifications in eukaryotes, plays a crucial role in a wide range of cellular processes. Studies on kinases and their substrates are important for understanding signaling networks in cells, and helpful for developing new treatments for signaling-defect diseases, such as cancer. The number of kinases was estimated to be around 500–1000 in animals and plants [1–3], and they usually induce phosphorylation on serine (S), threonine (T), tyrosine (Y), as well as histidine (H) amino acid residues in eukaryotic proteins. All experiments on phosphorylation site discovery are time-consuming and expensive to perform, and

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_18,

hence, computational prediction of protein phosphorylation sites becomes increasingly popular as an important complementary approach in protein phosphorylation site studies.

Nearly 40 methods for the prediction of phosphorylation sites, including kinase-specific and non-kinase-specific tools, with different algorithms and strategies, were described in the literature since 1999 [4]. A non-kinase-specific prediction tool requires only the protein sequence as input, and reports the likelihood of each S/T/Y amino acid residue being phosphorylated by any possible kinase, and hence, non-kinase-specific tools may be able to detect phosphorylation sites for which the associated kinase is unknown or the number of known substrate sequences of the associated kinase is few. With the development of the next-generation sequencing technology, many genomes of nonmodel organisms have been sequenced, and more kinases in those species have been discovered, some of which have no sufficient substrate information to train the kinase-specific prediction algorithms. Therefore, non-kinase-specific tools are required for a wider variety of species that have high-specificity for whole-genome annotation [4].

Here, we describe a non-kinase-specific protein phosphorylation site prediction webserver, PhosphoSVM, which uses the Support Vector Machine (SVM) method to integrate protein secondary structure information with seven other one-dimensional structural properties. In addition to protein secondary structure (SS), the other sequence-based properties are Shannon entropy (SE), relative entropy (RE), predicted protein disorder (PD), predicted solvent accessible surface area (ASA), amino acid overlapping properties (OP), averaged cumulative hydrophobicity (ACH), and subsequence k-nearest neighbor profiles (KNN). PhosphoSVM was cross-validated and independently tested [5]. This method achieved the values of the area under the receiver operating characteristic curve (AUC) of 0.8405/0.8183/0.7383 for S/T/Y phosphorylation sites, respectively, in animals with cross-validation, and 0.7761/0.6652/0.5958 for S/T/Y phosphorylation sites, respectively, in plants as an independent test.

2 Materials

2.1 Datasets

The training data sets came from P.ELM version 9.0 [6] and PPA version 3.0 [7–9]. The known phosphorylation sites in P.ELM are mainly for animals, whereas the PPA set is only for *Arabidopsis thaliana*. Therefore, the P.ELM and PPA data sets are relatively independent of each other. Any sites identified as phosphorylation sites by the computational methods in these databases were not considered as positives or negatives. Redundant protein sequences in these two datasets were removed by BLASTClust [10] with the cutoff of 30% sequence identity. The similarity between any two subsequences of phosphorylation sites, amino acids around phosphorylated amino

acid residues, was also checked to ensure the sequence identity was smaller than 30%. In both datasets, experimentally identified phosphorylation sites were considered as positive sites and a subset of the other S/T/Y sites were used as negative ones. In the P.ELM dataset, the ratios of positive to negative sites for S/T/Y are 4.65, 3.77, and 7.66%, respectively, compared with 3.14, 4.19, and 6.55% for the PPA dataset. A subset of negative sites was randomly selected for model training so that there were the same numbers of positive and negative phosphorylation sites (*see* **Note 1**).

2.2 Webserver

The webserver, PhosphoSVM, was developed for non-kinase-specific protein phosphorylation site prediction, which is available at http://sysbio.unl.edu/PhosphoSVM/. Fig. 1 displays the input page for the webserver that allows users to cut and paste the protein sequence. The only input required for PhosphoSVM is the query protein sequence in plain text or FASTA format (*see* **Note 2**). The standard 20 characters for amino acids are accepted, and any characters not included in those 20 will be removed by the webserver (*see* **Note 3**). Only one sequence per run is allowed for inputs. If multiple protein sequences are entered as the input, only the first one will be processed. Once a user submits a job by clicking the submit button, a new page will appear, which acknowledges the

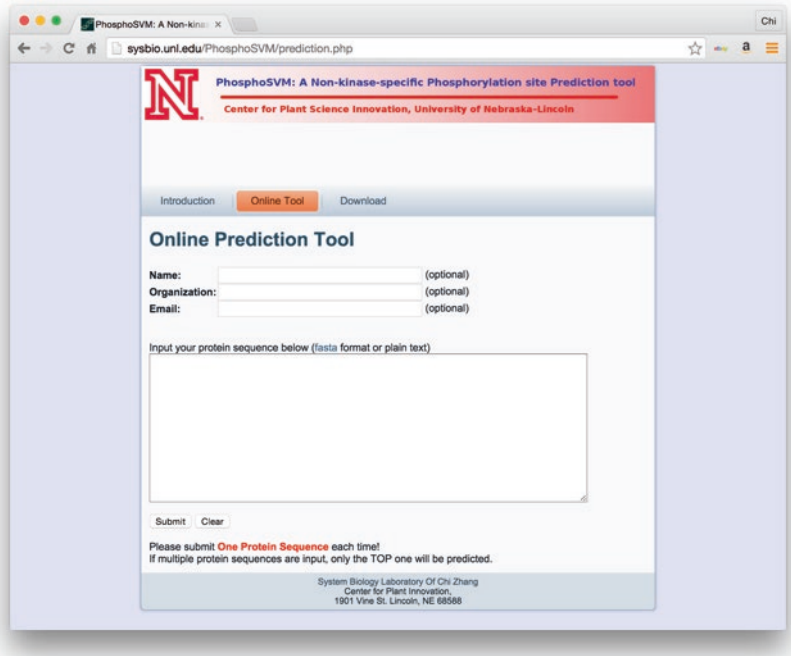


Fig. 1 The input window of the PhosphoSVM webserver. The only required input is the protein sequence, which can be copied and pasted into the main text box on this page. Name, Organization, and Email are optional

successful submission and displays an URL in red that will be used to check the prediction results (*see* **Note 4**). The input sequence is first screened against the database of all received input-sequences to see if it has been predicted before. If the same sequence has been predicted before, the existing results will be returned directly. Otherwise, the protein sequence is subsequently passed on to the predictor running in the background, which will search all S/T/Y sites in the given protein sequence, generate feature vectors for each candidate, and finally use a SVM classifier to score all candidate sites (*see* **Note 5**). The scores of all candidate sites, classified into three groups, S/T/Y, will be returned and displayed on the output page, which is shown in Fig. 2. Usually, a candidate site is considered as positive if its score is larger than 0.5, otherwise as negative. With a tenfold cross-validation, the AUC values achieved are 0.8405, 0.8183, and 0.7383 for S, T, and Y, respectively for the P.ELM data set. At the maximal F-measure point, PhosphoSVM achieved 94.04 and 95.90 % Sp for S-type phosphorylation sites in P.ELM and PPA, respectively. The performance for residue S is significantly better than the other two. The results are permanently saved in the database, and users can access the results with the URL obtained after they submit their input sequence.

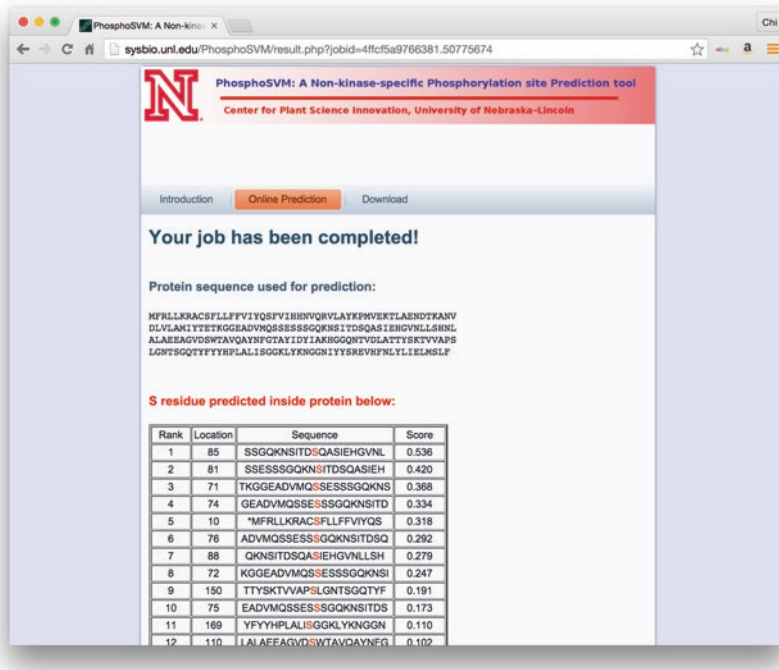


Fig. 2 The result window of the PhosphoSVM webserver. All candidate sites for a given protein sequence are classified into S, T, and Y groups. All candidate sites in one group are ranked based on their predicted scores. The rank, location, subsequence, and score for a given site are displayed

2.3 Downloading

Both the compiled training/test data sets and the well-trained SVM model used by the webserver are available for downloading at http://sysbio.unl.edu/PhosphoSVM/download.php.

3 Methods

3.1 SVM Package and Model Parameters

This webserver uses the SVM package LIBSVM [11]. The parameters, the window size and parameters of C, the cost, and γ for RBF kernel in SVM were optimized on the P.ELM dataset. An example window size of seven means that the given residue had three neighbors on each side in the subsequence (*see* **Note 6**). The parameter sets of window size and SVM parameters, γ and C, used by this webserver are (21, 0.003, 4), (19, 0.003, 4), and (15, 0.007, 2) for S/T/Y phosphorylation sites, respectively.

3.2 Feature Vectors

For a given amino acid residue, a subsequence with all residues adjacent to it in a certain window size is used to create the feature vector for the SVM. This subsequence will be encoded with a multidimensional vector based on protein secondary structure and seven other one-dimensional structural properties: Shannon entropy (SE), relative entropy (RE), protein disorder (PD), solvent accessible surface area (ASA), overlapping properties (OP), averaged cumulative hydrophobicity (ACH), and k-nearest neighbor profiles (KNN). In the following, each attribute is described in details.

3.2.1 Secondary Structure

Protein functions are dependent on their structures, and phosphorylation sites are enriched in some specific secondary structures [12]. The secondary structure (SS) attribute describes the structural environment of a phosphorylation site and its surrounding amino acid residues. The most accurate way to obtain the information of secondary structure would be from the 3D structures of proteins, but for a given protein sequence without known 3D structures, currently, the secondary structures can come from prediction. For this webserver, PSIPRED [13] is used to predict SS for a given protein sequence. The SS attribute of each residue in the feature vector has three bits to show the possibility scores of three types of secondary structures (H, E, and C).

3.2.2 Shannon Entropy and Relative Entropy

Shannon entropy (SE) and Relative entropy (RE) scores quantify the conservation of phosphorylation sites (*see* **Note 6**). SE and RE were calculated by weighted observed percentages (WOP), which was extracted by PSI-BLAST [10]. For a given full-length protein sequence that could potentially have phosphorylation sites, PSI-BLAST was applied to it against the NCBI BLAST Nonredundant protein database. The WOP vector for a position represents the position-specific distribution of 20 amino acids. The SE and RE scores for the given position are defined as:

$$\mathrm{SE} = -\sum_{i=1}^{20} p_i \log\left(p_i\right) \tag{1}$$

$$\mathrm{RE} = \sum_{i=1}^{20} p_i \log\left(\frac{p_i}{p_0}\right) \tag{2}$$

where $pi = ai/\Sigma aj$, aj is the j-th value in the WOP vector for this given position and p_0 is the protein BLOSUM62 background distribution. If a position has complete conservation, the SE score has the smallest value, 0.

3.2.3 Protein Disorder

For each residue, the protein disorder (PD) status is predicted using DISOPRED [14], and the prediction result has two scores, each between 0 and 1, corresponding to either a structured or disordered status (*see* **Note 8**).

3.2.4 Accessible Surface Area

All phosphorylation sites are on the surface of substrate proteins. Large solvent accessibility is hence also an important feature for the catalytic residues. Therefore, the solvent accessible surface area (ASA) information of each residue was included into the algorithm as well. For this webserver, RVP-net [15] is used to predict the relative solvent ASA for each residue in a given protein sequence (*see* **Note 9**). Each amino acid residue has a real value in the range of (0, 1) for the ASA attribute.

3.2.5 Overlapping Properties

Taylor's overlapping properties (OP) are: Polar {NQSDEC TKRHYW}, Positive {KHR}, Negative {DE}, Charged {KHRDE}, Hydrophobic {AGCTIVLKHFYWM}, Aliphatic {IVL}, Aromatic {FYWH}, Small {PNDTCAGSV}, Tiny {ASGC}, and Proline {P} [16] (*see* **Note 10**). Amino acid residues are encoded using 10-bit vectors where the dimensions of the corresponding properties are set to 1 and remaining positions are 0, i.e. A (0000100010), ..., V (0000110100).

3.2.6 Average Cumulative Hydrophobicity

Average cumulative hydrophobicity (ACH) is quantified by computing the average of the cumulative hydrophobicity indices around the central amino acid residue of a candidate subsequence over the sliding windows with sizes of 3, 5, 7, …, 21, respectively (*see* **Note 11**). Therefore, there are 10 bits in the feature vector for ACH scores for one given candidate subsequence. Hydrophobicity index proposed by Sweet and Eisenberg [17] is used by this webserver, where 20 standard amino acids (A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y) have the values of (0.62, 0.29, −0.90, −0.74, 1.19, 0.48, −0.40, 1.38, −1.50, 1.06, 0.64, −0.78, 0.12, −0.85, −2.53, -0.18, −0.05, 1.08, 0.81, 0.26), respectively.

3.2.7 K-Nearest Neighbor Profiles (KNN)

Similar patterns often appear in the local sequences of phosphorylation sites, and this information is helpful for phosphorylation prediction, especially for kinase-specific phosphorylation site prediction. To quantify this kind of information, the KNN score is introduced. A KNN score for one given sequence is the portion of positive phosphorylation sites in its k nearest neighbors in the training set, where the distance between two sequences is proportional to their sequence similarity; a pair of similar sequences have a short distance (*see* **Note 12**). The BLOSUM62 substitution matrix is used to calculate similarities between amino acids, and the sequence similarity is defined as the sum of all amino acid substitution scores. Several different k parameters are used to calculate KNN profiles for a given sequence. For S- and T-type sites, the parameters of k of KNN are (0.25 %, 0.5 %, …, 5 %) of the training data set, and thus the KNN profile attribute had 20 bits. For Y-type sites, the parameters of k are (1 %, 2 %, …, 5 %) for five bits because the size of the training set for Y-type sites is small.

4 Notes

1. For phosphorylation site prediction model training, it has been shown that the optimal ratio of positive to negative sites is one [18].
2. The query sequence must be a protein amino acid sequence in FASTA format. The gene in the DNA/RNA sequence has to be converted to amino acid sequence first by users. Unknown amino acids (e.g., X) must be removed.
3. Standard amino acid characters are "ACDEFGHIKLM NPQRSTVWY". Any characters not included in these 20 ones will be removed by the webserver, such as "X" and ".".
4. The URL looks like http://sysbio.unl.edu/PhosphoSVM/result.php?jobid=4ffcf5a9766381.50775674. The string after "jobid=" is the ID specifically assigned for the submitted job by the webserver, which is various for different cases. Please save this URL for retrieving the outputs in the future.
5. Usually, it will take a while for the prediction step. The waiting time depends on the length of the input protein sequence, and the length of the job queue of the webserver. The average waiting time is about 15 min.
6. The parameters were trained on the P.ELM dataset with the tenfold cross-validation method. All positive and negative sites on proteins in the 10th group were scored by the trained model. After ten rounds, all positive and negative sites in the whole dataset obtained prediction scores for analysis. The optimal set of parameters resulting in the highest AUC values were

obtained by a grid search within the interval of $(1, 25)$ in steps of 2 for the window size, $(0.001, 0.01)$ in steps of 0.001 for γ, and $(2^{-5}, 2^{4})$ in steps of $\times 2$ for C. Since this method was found to be sensitive to parameter C, an additional fine linear search in $(1, 6)$ in steps of 1 for parameter C was conducted, while the other parameters kept the same grid sizes as before.

7. SE is commonly used for the prediction of functionally important amino acid residues in protein sequences [19, 20]. RE measures the conservation of amino acids compared with the background distribution, and the deviation from a background distribution is also important for functionally important amino acid residues [21].
8. Previous works suggested that protein disorder information is helpful for the discrimination between phosphorylation and nonphosphorylation sites [12].
9. The prediction of ASA does not have high resolution, and the phosphorylation sites that become accessible upon protein conformational changes cannot be evaluated by currently existing methods.
10. OP reflects the amino acid groups with common physicochemical properties, and were used for the identification of protein motifs [22], prediction of T-cell epitopes [23], and prediction of functionally important amino acid residues in a given protein sequence [24–26] etc.
11. Average cumulative hydrophobicity (ACH) of amino acids has been demonstrated to be an important attribute for functional important amino acid residues in proteins [26–28], because it quantifies the propensity of an amino acid residue and its surrounding residues to be exposed to solvents.
12. A KNN attribute vector actually quantifies the neighborhood of a phosphorylation site in the similarity network of all known sites. This feature has been used and described in detail by Musite for phosphorylation site prediction [29].

Acknowledgement

This project was supported by funding under CZ's startup funds from University of Nebraska, Lincoln, NE. This work was completed utilizing the Holland Computing Center of the University of Nebraska.

References

1. Caenepeel S, Charydczak G, Sudarsanam S, Hunter T, Manning G (2004) The mouse kinome: discovery and comparative genomics of all mouse protein kinases. Proc Natl Acad Sci U S A 101(32):11707–11712. doi:10.1073/pnas.0306880101
2. Manning G, Whyte DB, Martinez R, Hunter T, Sudarsanam S (2002) The protein kinase complement of the human genome. Science 298(5600):1912–1934. doi:10.1126/science.1075762
3. Vlad F, Turk BE, Peynot P, Leung J, Merlot S (2008) A versatile strategy to define the phosphorylation preferences of plant protein kinases and screen for putative substrates. Plant J 55(1):104–117. doi:10.1111/j.1365-313X.2008.03488.x
4. Trost B, Kusalik A (2011) Computational prediction of eukaryotic phosphorylation sites. Bioinformatics 27(21):2927–2935. doi:10.1093/bioinformatics/btr525
5. Dou Y, Yao B, Zhang C (2014) PhosphoSVM: prediction of phosphorylation sites by integrating various protein sequence attributes with a support vector machine. Amino Acids 46(6):1459–1469. doi:10.1007/s00726-014-1711-5
6. Diella F, Gould CM, Chica C, Via A, Gibson TJ (2008) Phospho.ELM, a database of phosphorylation sites—update. Nucleic Acids Res 36(Database issue):D240–D244. doi:10.1093/nar/gkm772
7. Heazlewood JL, Durek P, Hummel J, Selbig J, Weckwerth W, Walther D, Schulze WX (2008) PhosPhAt: a database of phosphorylation sites in Arabidopsis thaliana and a plant-specific phosphorylation site predictor. Nucleic Acids Res 36(Database issue):D1015–D1021. doi:10.1093/nar/gkm812
8. Durek P, Schmidt R, Heazlewood JL, Jones A, MacLean D, Nagel A, Kersten B, Schulze WX (2010) PhosPhAt: the Arabidopsis thaliana phosphorylation site database. An update. Nucleic Acids Res 38(Database issue):D828–D834. doi:10.1093/nar/gkp810
9. Zulawski M, Braginets R, Schulze WX (2013) PhosPhAt goes kinases—searchable protein kinase target information in the plant phosphorylation site database PhosPhAt. Nucleic Acids Res 41(Database issue):D1176–D1184. doi:10.1093/nar/gks1081
10. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25(17):3389–3402
11. Fan RE, Chen PH, Lin CJ (2005) Working set selection using second order information for training support vector machines. J Mach Learn Res 6:1889–1918
12. Iakoucheva LM, Radivojac P, Brown CJ, O'Connor TR, Sikes JG, Obradovic Z, Dunker AK (2004) The importance of intrinsic disorder for protein phosphorylation. Nucleic Acids Res 32(3):1037–1049. doi:10.1093/nar/gkh253
13. McGuffin LJ, Bryson K, Jones DT (2000) The PSIPRED protein structure prediction server. Bioinformatics 16(4):404–405
14. Ward JJ, Sodhi JS, McGuffin LJ, Buxton BF, Jones DT (2004) Prediction and functional analysis of native disorder in proteins from the three kingdoms of life. J Mol Biol 337(3):635–645. doi:10.1016/j.jmb.2004.02.002
15. Ahmad S, Gromiha MM, Sarai A (2003) RVP-net: online prediction of real valued accessible surface area of proteins from single sequences. Bioinformatics 19(14):1849–1851
16. Taylor WR (1986) The classification of amino acid conservation. J Theor Biol 119(2):205–218
17. Sweet RM, Eisenberg D (1983) Correlation of sequence hydrophobicities measures similarity in three-dimensional protein structure. J Mol Biol 171(4):479–488
18. Biswas AK, Noman N, Sikder AR (2010) Machine learning approach to predict protein phosphorylation sites by incorporating evolutionary information. BMC Bioinformatics 11:273. doi:10.1186/1471-2105-11-273
19. Capra JA, Singh M (2007) Predicting functionally important residues from sequence conservation. Bioinformatics 23(15):1875–1882. doi:10.1093/bioinformatics/btm270
20. Mihalek I, Res I, Lichtarge O (2004) A family of evolution-entropy hybrid methods for ranking protein residues by importance. J Mol Biol 336(5):1265–1282. doi:10.1016/j.jmb.2003.12.078
21. Johansson F, Toh H (2010) A comparative study of conservation and variation scores. BMC Bioinformatics 11:388. doi:10.1186/1471-2105-11-388
22. Wu TD, Brutlag DL (1995) Identification of protein motifs using conserved amino acid properties and partitioning techniques. Proc Int Conf Intell Syst Mol Biol 3:402–410
23. Gok M, Ozcerit AT (2012) Prediction of MHC class I binding peptides with a new feature encoding technique. Cell Immunol 275(1–2):1–4. doi:10.1016/j.cellimm.2012.04.005

24. Wu CY, Hwa YH, Chen YC, Lim C (2012) Hidden relationship between conserved residues and locally conserved phosphate-binding structures in NAD(P)-binding proteins. J Phys Chem B. doi:10.1021/jp3014332
25. Dou Y, Zheng X, Yang J, Wang J (2010) Prediction of catalytic residues based on an overlapping amino acid classification. Amino Acids 39(5):1353–1361. doi:10.1007/s00726-010-0587-2
26. Dou Y, Wang J, Yang J, Zhang C (2012) L1pred: a sequence-based prediction tool for catalytic residues in enzymes with the L1-logreg classifier. PLoS One 7(4):e35666. doi:10.1371/journal.pone.0035666
27. Zhang T, Zhang H, Chen K, Shen S, Ruan J, Kurgan L (2008) Accurate sequence-based prediction of catalytic residues. Bioinformatics 24(20):2329–2338. doi:10.1093/bioinformatics/btn433
28. Wang L, Brown SJ (2006) BindN: a web-based tool for efficient prediction of DNA and RNA binding sites in amino acid sequences. Nucleic Acids Res 34(Web Server issue):W243–W248. doi:10.1093/nar/gkl298
29. Gao J, Thelen JJ, Dunker AK, Xu D (2010) Musite, a tool for global prediction of general and kinase-specific phosphorylation sites. Mol Cell Proteomics 9(12):2586–2600. doi:10.1074/mcp.M110.001388

Chapter 19

Predicting Post-Translational Modifications from Local Sequence Fragments Using Machine Learning Algorithms: Overview and Best Practices

Marcin Tatjewski, Marcin Kierczak, and Dariusz Plewczynski

Abstract

Here, we present two perspectives on the task of predicting post translational modifications (PTMs) from local sequence fragments using machine learning algorithms. The first is the description of the fundamental steps required to construct a PTM predictor from the very beginning. These steps include data gathering, feature extraction, or machine-learning classifier selection. The second part of our work contains the detailed discussion of more advanced problems which are encountered in PTM prediction task. Probably the most challenging issues which we have covered here are: (1) how to address the training data class imbalance problem (we also present statistics describing the problem); (2) how to properly set up cross-validation folds with an approach which takes into account the homology of protein data records, to address this problem we present our *folds-over-clusters* algorithm; and (3) how to efficiently reach for new sources of learning features. Presented techniques and notes resulted from intense studies in the field, performed by our and other groups, and can be useful both for researchers beginning in the field of PTM prediction and for those who want to extend the repertoire of their research techniques.

Key words Phosphorylation, Feature extraction, Feature selection, Class imbalance, Cross-validation

1 Introduction

A living cell is a dynamic entity capable of maintaining complex functions and adjusting its metabolism in response to the constantly changing environment. This functional complexity can be observed at different levels of cellular organization: from information stored in the genome, to the regulation of its expression, throughout proteins characterization, their interactions, finally intracellular signaling and beyond.

Taking human as an example, it is currently well established that the genome comprises 20,000–25,000 genes that give origin to more than 1,000,000 proteins (including variants of the same protein) present in the proteome [1, 2]. There is a large discrepancy between the number of genes and the number of protein

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_19,

variants encoded by them. This apparent paradox can, however, be explained by taking into account several mechanisms such as alternative splicing, or post-translational modifications that enable this relatively small number of genes to encode the number of proteins that is two orders of magnitude higher.

Here, we will focus on one particular mechanism, namely the post-translational modifications (PTM). These are defined as enzymatic covalent modifications of proteins occurring after the translation. It has been estimated that about 5 % of the human proteome are enzymes involved in catalyzing over 200 types of known PTMs [3]. The enzymes performing these different PTMs belong to many different classes, such as:

- *kinases* that add phosphate groups to amino acid chains,
- *phosphatases* that remove phosphate groups to amino acid chains,
- *transferases* that transfer functional groups, sugars, lipids, or peptides at amino acid sites,
- *ligases* that add functional groups, sugars, lipids, or peptides,
- *proteases* that cleave peptide bonds at predefined sites.

PTMs can occur at different stages of protein life-cycle and are usually reversible. Many proteins also contain *autocatalytic domains* enabling them to modify themselves. These factors explain the key role that PTMs play in the dynamic regulation of protein activity, at a pace adequate to the constantly changing environment.

PTMs play a paramount role not only in physiological processes taking place in a healthy cell, but have also been shown to be key players in etiology of many diseases. The pathologies linked to abnormal or altered PTMs include cancer, diabetes, and neurodegenerative diseases.

Several studies have been focusing on better understanding how PTMs work and what are the determinants of an amino acid site that can be modified. The actual molecular changes following several PTMs are known and quite often they have great impact on the function of the modified protein. Let us have a closer look at one particular PTM: phosphorylation of tyrosine 530 (Tyr 530) in the Src protein (Fig. 1). Src protein is a *tyrosine kinase*, i.e. an enzyme that catalyzes phosphorylation of tyrosines in substrate proteins. It controls the flow of information regulating cell growth and thus playing important role in cell physiology. Abnormal Src function has been implicated in etiology of several types of cancer [4]. The Src itself is controlled by phosphorylation/dephosphorylation of a tyrosine 530 in its tail part. The protein consists of the SH3, the SH2, and the kinase domain. The linker connects the SH3 and the SH2 domain. The tail part contains the key tyrosine Tyr 530. Upon the phosphorylation (panel a, PDB 2src), the phosphate group binds to the SH2 domain and the whole Src is "closed." Upon dephosphorylation (panel b, PDB 1y57), the protein "opens" exposing a protein

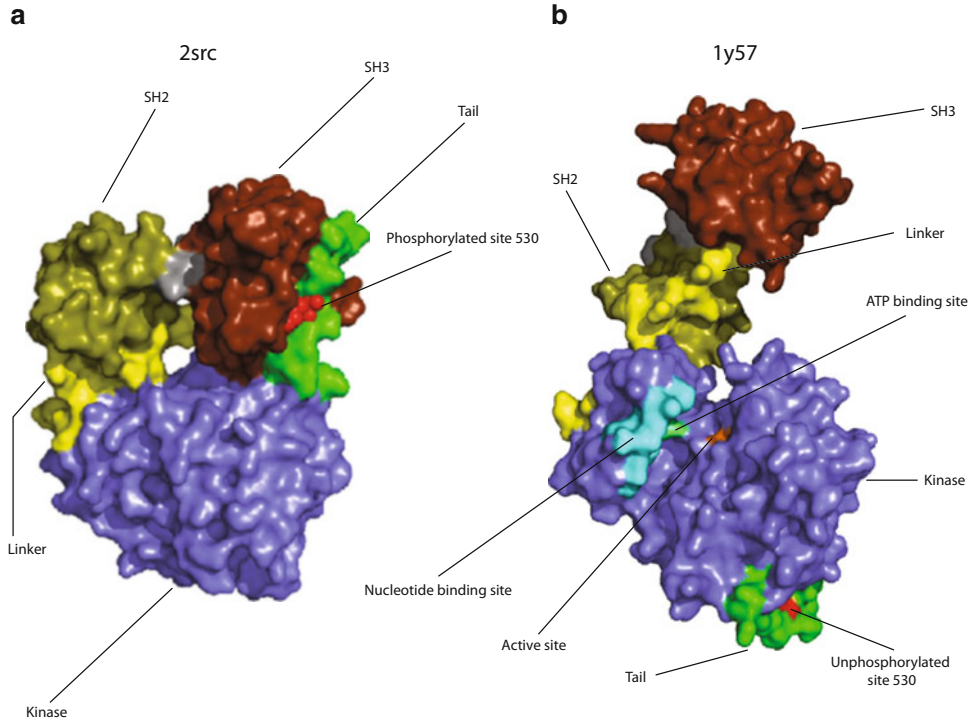


Fig. 1 An example of biological importance of PTMs, here a phosphorylation of tyrosine. The Src protein controls the flow of information regulating cell growth. It belongs to a class of tyrosine kinases, i.e. the enzymes catalyzing phosphorylation of substrate proteins. The Src itself is controlled by phosphorylation/dephosphorylation of a tyrosine in its tail part. The protein consists of the SH3, the SH2, and the kinase domain. The linker connects the SH3 and the SH2 domain. The tail part of the protein contains the key tyrosine Tyr530. When phosphorylated (panel (**a**), PDB 2src), the phosphate group binds to the SH2 domain and the whole Src is "closed." Upon dephosphorylation (panel (**b**), PDB 1y57), the protein "opens" exposing a protein binding groove in the SH3 domain (not shown) and the active site located in the kinase domain and previously buried inside the protein [5, 6]

binding groove in the SH3 domain and the active site located in the kinase domain and previously buried inside the protein thus enabling Src to perform its catalytic functions [5, 6].

Although challenging, full characterization of PTMs is necessary for more complete understanding of both normal biological processes and in elucidating etiology of diseases.

As experimental verification of PTM occurrences requires significant effort, researchers intensively explore in silico approaches to finding sites of potential modifications. These methods focus on predicting new PTM sites with the use of knowledge gained from observing the amino acid neighborhoods of the already verified modifications. The approaches used range from classic machine-learning algorithms like Random Forest, Support Vector Machine or Artificial Neural Networks [7–9] to methods more specific for the domain like Position-Specific Scoring Matrices or comparative evolutionary approaches [10–13].

Despite progress made in computational prediction of PTM sites, several open challenges still exist that require further investigation. Below, we mention some of these open challenges:

- Whether to include or not the available protein structural information in the prediction process and how to do it effectively (Subheading 19.4.5)?
- How to determine the size of the amino acid neighborhood of the site to maximize the predictive power? These issues can be addressed jointly with the problem of using structural information (Subheading 19.3.3.1).
- What features to consider for constructing predictors (Subheadings 19.3.3.2 and 19.4.6).

In this work, we attempt to familiarize the reader with the fundamentals of constructing machine-learning predictors of PTMs based on local sequence fragments. We also highlight some more advanced topics in the field. Subheading 19.2 discusses the resources available in the field, and the following Subheading 19.3 presents in a succinct manner all steps necessary to build a predictor. The final Subheading 19.4 focuses on advanced topics and issues that can be encountered in PTM prediction. Figure 2 guides the reader through particular sections of the article using a diagram of different phases and entities that are present in the process of PTM prediction.

2 Materials

2.1 Databases for the Domain

In order to be able to make predictions of potential PTM sites, we need to get access to a substantial (a large number of examples) data set of protein sequence sites that were proven to be subject of a modification. Typically, these proofs come from experimental studies of two types:

Low-throughput (LTP) Small scale experiments, in which scientists confirm their findings using robust techniques.

High-throughput (HTP) High scale experiments usually involving mass spectrometry techniques. These are usually probabilistic by nature.

There are several databases available, which aggregate such PTM data from scientific literature. The most commonly used ones are:

UniProtKB is a general protein database which contains many other biological information apart from the PTM annotation for proteins. It is one of the most comprehensive sources of modification data as it includes many different types of PTMs. Another of its advantages lies in the easy programmatic access to all of the

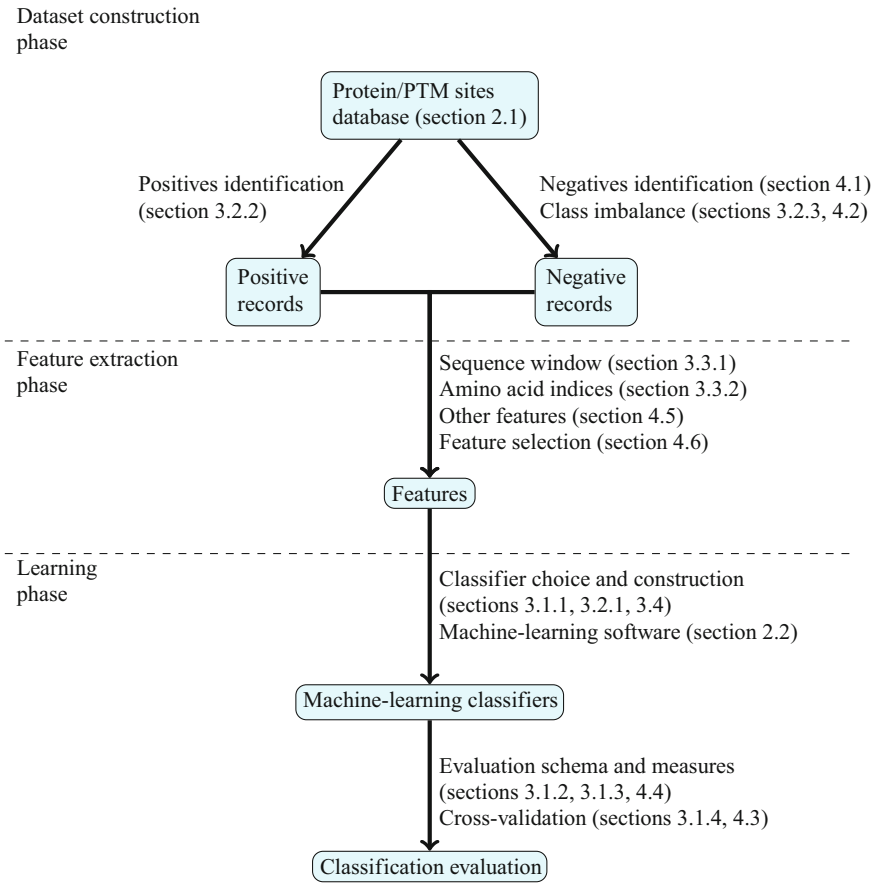


Fig. 2 Diagram presenting different phases and entities in the process of PTM prediction. References in *brackets* point to sections of the article which explain particular topics

database resources aggregated on protein level. On the other hand, it is important to be aware that many PTMs stored in this database are inferred by homology, which means they were not even confirmed by HTP, or LTP experiments [14].

PhosphoSitePlus contrary its name, it does contain not only phosphorylations. It also includes acetylations, ubiquitylations, methylations, and others. However, it focuses on phosphorylations and provides most of the data for this type of modification. Most of the sites gathered there are confirmed by HTP studies [15].

Phospho.ELM This is one of the databases solely focused on phosphorylation sites—it does not contain data regarding any other modification types [16]. The instances are typically confirmed by either LTP or HTP experiments.

Contrary to UniProt, PhosphoSitePlus and Phospho.ELM provide information about the site source being an LTP or HTP study. Table 1 provides statistics showing the number of sites for a

Table 1
Numbers of sites for chosen types of PTMs

PTM type	UniProt All[a,b]	UniProt Man.[a,c]	PhosphoSitePlus[a]	Phospho.ELM[d]
Phosphoserine	113,319	101,016	153,974	31,754
Phosphothreonine	33,811	22,132	63,576	7449
Phosphotyrosine	13,383	9126	42,001	3370
Acetylation	58,581	34,374	36,278	–
Methylation	28,935	9795	8096	–
Ubiquitylation	–	–	56,705	–
Amidation	4203	3436	–	–
Hydroxylation	2702	2625	–	–

Statistics include annotations on all levels of confirmation status[a]Latest informationavailable online[b]All typesof assertions[c]Manualassertions only[d]Asof 2011 [16]

few selected PTM types in the discussed databases. In order to learn about other PTM databases, please refer to an extensive review written by Kamath et al. in 2011 [17].

2.2 Machine Learning Software

Several software libraries are available for performing prediction using machine learning algorithms. Often the choice of particular package is driven by the programming language which the user is most familiar with. Among the vast set of available tools, the following three especially deserve a short description as being the most widely used and recognized:

MATLAB A multi-paradigm numerical programming environment (and a programming language) with a large number of so-called toolboxes extending its capabilities (http://mathworks.com/products/matlab).

scikit-learn Popular library for the Python language, allows only programmatic use [18].

Spark's *MLlib* Apache's machine learning library for Spark and Hadoop. It is possible to use it from Java, Python, Scala, and R.

WEKA Java library that apart from programmatic access has also a standalone application with a Graphical User Interface. Its previous popularity seems to be decreasing. One of the reasons for that is it is implemented in a memory-intensive way which is especially troublesome when working with large data sets. In the previous decade it was often the first-choice machine learning tool [19].

Neither scikit-learn nor MLlib provides implementation of Artificial Neural Networks. However, they include most of the other important machine-learning algorithms.

3 Methods

3.1 Machine Learning Basics

The first informal definition of machine learning has been given by Arthur Samuel in 1959 [20]. He defined *machine learning* as a ... *field of study that gives computers the ability to learn without being explicitly programmed*. In order for a computer program to learn, it has to be *trained* using a number of examples provided by the experimenter in a *training set*. Each example, an *instance* consists of a number of variables, *features* (also called *attributes*) and a number of *outcomes* (or *decision attributes*). If one or more decision attributes are provided, we use the term *decision system*, and if no decision attributes are present, we are dealing with an *information system*.

There exist two main types of machine learning: *unsupervised learning* and *supervised learning*. The unsupervised learning aims at discovering patterns present in the data and consists of all the methods and algorithms that take an information system as input. Different types of clustering are a good example of unsupervised learning. In contrast, supervised learning aims at constructing *predictive models* that are capable of *classifying* (predicting the outcome) for previously unseen instances. To achieve this, the supervised learning methods take a decision system as input and attempt at linking patterns present in the data to particular outcomes.

Often, prior to constructing classifier, it is desirable to constrain the set of features that will be taken into account by the model. This helps alleviating the so-called *curse of dimensionality*, i.e. exponential growth of the required number of observations (instances) caused by linear increase in the number of features. The reduction of the features space can be achieved performing *feature extraction* and *feature selection* prior to constructing model. Feature extraction is a process in which the domain knowledge is applied to reduce the set of considered features, e.g. expert knowledge is employed to eliminate features that are least likely to explain the outcome given the current state of knowledge on the modeled phenomenon. Feature selection is a term describing a number of methods that, in a systematic way, search different feature subsets and evaluate predictive power of every such subset (often also of every single feature). It is not uncommon that both feature extraction and feature selection are applied in a sequential manner. In addition to addressing the curse of dimensionality, by reducing feature space, the experimenter is often able to reduce noise (improve signal to noise ratio) and to better control over-fitting of the model.

After constructing a classifier, one wants to evaluate quality of predictions, i.e. to establish how good the predictions it produces are. One common scheme employed for this purpose is to use the so-called *test set*, a set of instances that were not a part of training set and for which the outcome is known. Once the classifier has been trained, it is used to predict the outcome for the test set

instances. Since the true outcome is known for these instances, one can simply measure concordance between the predicted and the actual outcome for all test set instances. Once the quality of the classifications has been established, the constructed predictive model (classifier) can be used to predict outcomes for previously unseen instances. We will, in a more detailed way, revisit all the analyses steps in the next paragraphs.

To set the reference frame, let us focus on predicting whether a particular amino acid sequence fragment will be subject to a particular post-translational modification. One convenient and straightforward way of representing an amino acid sequence is to use an alphabet of 20 symbols representing 20 amino acids each. Thus, a sequence of three amino acids: alanine, proline, and isoleucine will become Ala-Pro-Ile or, using more compact notation, API. Now, let us assume that we have gathered some examples of sequences where amino acid proline is subject to hydroxylation and becomes hydroxyproline (positive examples). We also have a set of sequences where proline is not modified (negative examples). Our task is to construct a model capable of detecting prolines potentially subject to hydroxylation. The related decision system can be represented as in Table 2.

In the example decision system from Table 2, we have n instances, where there are $p - 1$ positive examples (modification occurs) and $n - p + 1$ negative examples (no modification). In the system, we have $m + 2$ attributes, where one is the unique instance id and the other is a binary decision attribute. All *site* attributes are discrete as they take 20 distinct values—amino acid codes. Now, to illustrate feature extraction, let us state that probably not all amino

Table 2
An example of a decision system for predicting one particular PTM type from the amino acid sequence

id	*site_m*	…	$site_{-1}$	$site_0$	$site_{+1}$	…	*sitem*	Is hydroxylated?
1	G	…	A	P	R	…	L	Yes
2	G	…	V	P	R	…	M	Yes
3	G	…	A	P	R	…	C	Yes
…	…	…	…	…	…	…	…	…
p	G	…	A	P	R	…	I	No
$p + 1$	G	…	R	P	K	…	L	No
n	G	…	R	P	R	…	V	No

Site at index 0 is the position of modification, while other2m sites represent theneighboring residues. First $p - 1$rows contain positive examples, while the rest are the negatives

acids in the sequence determine whether a given site is modified or not. Domain knowledge (molecular biology) comes at help and lets us limit the number of considered positions to, say, 50 immediate neighbors of the potentially modified site. Thus, we can reduce our space to 101 features. But is it all 100 amino acids that determine modification status? If domain knowledge does not provide clear answers to this questions, one can use a feature selection algorithm to further narrow down the space to the most relevant features, i.e. the ones with high predictive value. Such processed decision systems can be used for constructing a classifier.

3.1.1 Classification/ Prediction Task

To-date, a number of algorithms have been developed to construct predictive models given a decision system. To mention just some of the most popular approaches, one can choose between decision trees (DT), neural networks (NN), support-vector machines (SVM), random forests (RF), rough sets (RS), fuzzy sets (FS), or Bayesian methods. None of the existing methods is superior to all others for all types of data, but each of these methods has its pros and cons determining the scope of its applicability. In general, when choosing the classification algorithm, one can consider some of its specific inherent features including:

- type of data it can handle (e.g., continuous vs. discrete),
- classification transparency, i.e. how easy it is to get insight into the actual classification process (a set of easy-to-read rules vs. a matrix of weights),
- how does the amount of required resources scale with the size of the data.

Subheading 19.3.4 discusses which of the above algorithms are often used for the PTM prediction problem.

3.1.2 Evaluation Scheme

As already mentioned, performance evaluation is a vital step in the modeling process. In previous paragraph (Subheading 19.3.1), we briefly mentioned that a test-training set pair can be used to evaluate classifier. We also talked about feature selection and about different modeling algorithms. At each of these steps, quality of classification measures can be used for slightly different purposes. When comparing classifiers built using different machine learning algorithms, one would like to measure the performance in order to select the optimal (given some a priori set criteria) classifier. This is usually done by keeping aside a subset of the original data, the *validation set* and treating the remaining data as the *training set*. Since the outcome is known for the instances in the validation set, one can measure classification performance by comparing this known *true outcome* with the outcome predicted by the classifier. Once the classifier has been selected, one would like to measure its predictive quality. While this has seemingly been already measured when

choosing the optimal classifier, this estimate is biased by the fact that we used a particular random subset of data to construct the validation set which was used to select one of the model parameters (here the model type itself). This, in turn, may result in *over-fitting*, i.e. overly optimistic estimates of classification performance. To remedy this problem, another subset of the original data, the so-called *test set* should be used to measure the performance of the selected classifier. Instances constituting the test set should be present in neither the validation set nor the training set.

3.1.3 Evaluation Measures

Several measures exist that reflect different aspects of classification performance. Here, we will briefly discuss the measures commonly used in PTM prediction-related literature. Unless mentioned, we will be focusing on binary outcome, i.e. the outcome where only two values are possible: 0 or 1, true or false, positive or negative, modified or non-modified, etc.

Given a pair training-test set, for every instance in the test set, one can compare the *predicted outcome* and the observed *actual outcome*. Four different scenarios are possible that can be summarized in the form of a *confusion matrix* (Table 3).

Several measures are in use that describe different properties of a classifier based on the confusion matrix. Perhaps the most often discussed property characterizing a classifier is the proportion of correctly classified instances called the *accuracy* and defined as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (1)$$

The proportion of true positives yielded by the classifier is referred to as *precision*:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2)$$

Indeed, intuitively, the less false positives the classifier yields, the more precise it is. In addition to the above outlined measures, a classifier is often characterized by its *sensitivity* (also referred to as the *true positives rate* or the *recall rate*) and its *specificity*. Both sensitivity and specificity are very important measures when talking

Table 3
An example of a confusion matrix

		Actual outcome	
		Modified	Not modified
Predicted outcome	Modified	TP	FP
	Not modified	FN	TN

about medical applications. For instance, a very sensitive medical test is unlikely to fail at detecting the disease but if not specific, it will also produce false alarms. Observe that 1 – Specificity estimates the likelihood that a healthy patient will be classified as sick. Formally, sensitivity is defined as:

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

and specificity as:

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{4}$$

While an ideal classifier would be sensitive and specific at the same time, in practice it is difficult to achieve. Different applications call for different classifier optimization criteria, e.g. is it worse to miss the disease in a patient or to administer a treatment with strong side-effects to a healthy patient who happened to be classified as sick? Thus, when selecting a classifier or tuning its parameters, it is convenient to visualize *Sensitivity* vs. 1 – Specificity in a so-called *Receiver Operating Characteristic* (ROC) curve.

Despite being extensively used, the accuracy as defined in the one of the above equations (Eq. 1) can easily be biased by the so-called *class imbalance*. The class imbalance term describes a situation where different decision classes are unevenly represented in the data set. Let us consider an extreme situation where the test set consists of 95 positive and 5 negative instances and observe that a naïve classifier that always labels an instance as positive will achieve very high accuracy while in reality it classifies all negative examples wrong! However, it has been shown [21] that area under the ROC curve (AUC) is free from such bias. Thus, AUC is commonly used to estimate the quality of a classifier. For a hypothetical ideal classifier $AUC = 1$ while $AUC = 0.5$ is equivalent to just tossing a coin.

To measure the quality of classification with only two possible outcomes (binary classification), the *Matthews correlation coefficient* (MCC) [22] is also used that considers true and false positives and negatives:

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP}) \times (\text{TP} + \text{FN}) \times (\text{TN} + \text{FP}) \times (\text{TN} + \text{FN})}} \tag{5}$$

The MCC value of +1 represents a perfect prediction, 0 a random prediction, and – 1 a total disagreement between prediction and observation [22, 23].

Yet another value used in measuring prediction quality is the *F-measure* which, in case of a binary classification problem, is a weighted average (traditionally a harmonic mean) of the precision and recall and takes values from 0 to 1 for the perfect classifier [23]:

$$F_1 = \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{6}$$

3.1.4 Cross-Validation

In reality, it is often difficult to sacrifice a subset of original data to construct validation and test set. Quite often, one is limited by the number of available observations (instances) and if only a subset of the original data is used to train classifiers, there is an increased risk that the examples present in the training set do not represent the whole *universe*, i.e. that there is not enough information to properly describe the *concept*—the modeled phenomenon. For instance, if the decision is binary and the original data contains 90 % positive examples and only 10 % negative examples, further sub-setting may lead to a situation that, e.g. the training set contains instances covering only some particular patterns of condition attribute values leading to negative outcome.

One way to address this problem and to decrease the bias caused by random sub-sampling of instances to training and test set is to use several different pairs of training-test sets and average performance measures obtained from training and evaluating classifier on each such pair. This approach is called *cross-validation*. In a k-fold cross-validation, the original training set is split into k equal parts. Next, k-times $k - 1$ parts are used to train the classifier and the remaining part is used for evaluation purposes (as a test set). At every iteration, a different i-th of the k parts is selected to be the test set ($i = 1, \ldots, k$). In a *stratified cross-validation*, in each of the k parts, the original distribution of decision classes (the ratio of instances belonging to particular decision classes). When k is equal to the number of instances in the original training data set, we talk about a *leave-one-out cross-validation* (or, a *jackknife cross-validation*[1]).

The specific nature of protein training records in the PTM prediction problem makes the default randomized way of forming cross-validation sets faulty. In Subheading 19.4.3 we discuss how to manage this problem with our *folds-over-clusters* cross-validation algorithm.

3.2 Data Set Construction

3.2.1 Separate Predictors for Each Type of PTM

Different types of modifications have different biological nature, thus for each type we need to construct separate predictors. In effect, in order to detect potential modification sites of all possible PTM types, we need to run all of our predictors on the input data.

Moreover, in order to construct all these different predictors we need to obtain separate training sets for each PTM type. Obviously, predictor of type T modification can only by trained on data records related to occurrences of type T modification.

[1] Less correct name since *jacknife* is a resampling method rather than a cross-validation type.

3.2.2 Positive and Negative Records Identification

The approach to gathering data records of PTM occurrences is rather straightforward. When building a data set for predictor of type T modifications we just extract instances of T modification sites from the relevant databases described in Subheading 19.2.1.

On the other hand, the problem of finding data records for non-occurrence of PTMs is not that simple. There exists no method capable of confirming that a particular protein sequence site is not a subject of modification. Therefore, there is also no database storing that kind of information. In effect, in order to extract negative PTM examples we need to make assumptions, e.g. that if a modification was not confirmed to happen at a particular protein sequence site, then it never occurs at this place. More detailed analysis of the aspect of negative data records is presented in Subheading 19.4.1.

3.2.3 Class Balance in Training Set

The above presented approach to finding sites of PTM non-occurrence gives us access to an enormous set of negative data records; this is in strong contrast to a constrained set of positive examples. There are a few factors limiting the available information about modification occurrence sites:

1. From the currently available data it seems that minority of amino acids in a typical protein sequence are subject to PTMs.
2. To-date, experiments verifying the presence of modifications have been performed only on a limited number of proteins.
3. Testing for presence of type $T1$ PTM often does not tell us much about potential occurrence of a different type $T2$ PTM.
4. Not all experimental results on presence of modifications are reflected in the information stored in the domain databases (see statistics of differences in Table 1).

In result, when constructing training and testing data sets, we face a difficult question of defining the balance between positive and negative records.

There exist several approaches to address such a problem, yet not all of them are suitable for PTM prediction problem. Our proposal of a solution is under-sampling negative records in order to obtain around ten times more negative records than positive examples. For more detailed analysis of this problem and other potential approaches, please refer to Subheading 19.4.2.

3.2.4 Kinase-Specific Phosphorylation Predictors

One special type of PTMs are phosphorylations. These modifications modify serine, tyrosine, threonine, or histidine by attaching phosphate groups to these amino acids. Importantly, phosphorylations are performed by a special class of enzymes, protein kinases. In other words, presence of a kinase is essential for a phosphorylation to take place. Since protein kinases are a broad family of molecules which have different characteristics, it is necessary to take this diversity into account when creating phosphorylation predictors.

It is commonly reiterated that "there is no average phosphorylation site" [11, 24], which is a heavy constraint on the potential performance of non-kinase-specific predictors. Therefore, currently a more widespread and more efficient approach is to build kinase-specific predictors [11]. This method is enabled by the fact that most of the domain databases (including all three mentioned in Subheading 1) store data about the kinase involved in phosphorylation whenever such information is available.

Building enzyme-specific predictors for other types of PTMs would also be worth consideration. However, in the data available for modifications other than phosphorylations the information about involved enzymes is limited.

3.3 Feature Extraction

In Subheading 19.3.1, we have already mentioned that in order to train a machine learning predictor, one has to gather a set of features describing each data record. In PTM prediction, the typical approach is to extract the amino acids from the immediate neighborhood of the actual modification site and treat them as features (or use their properties as features).

3.3.1 Sequence Window Size

Defining the size of the sequence neighborhood that should be extracted is another challenge. Different authors used from 7 up to 101 residues, including the modification site itself [11, 25]. For instance, in the case of phosphorylation it is practically very difficult to determine what should be the optimal window size, as it depends on the part of the protein sequence that actually interacts with the protein kinase.

Interacting amino acids can be identified only if we knew the actual 3D structure of a protein, as residues close in 3D might be very distant in sequence. Due to still relatively scarce knowledge of protein 3D structures, the use of methods accurately defining size of a relevant neighborhood is limited. Therefore, it is necessary to make some assumptions guiding the choice of the optimal sequence fragment size used for predictions. Careful analysis of the results reported in the domain literature points to 21 amino acids as a good compromise between introducing unnecessary noise and capturing enough relevant biological information [11]. However, it is worth noticing that the above number has been optimized for predicting phosphorylations and may not be optimal for predicting other types of PTMs.

3.3.2 Advanced Features: Amino Acid Indices

It is possible to build PTM predictors using only raw protein sequence as a source of features. However, an effective approach to improve such basic model is to represent each amino acid by its physicochemical characteristics. An accessible and easy to use source of such features is the Amino Acid Index database [26, 27]. Its subset—AAIndex1—contains more than 500 linear characteristics of amino acids.

In order to understand the advantage coming from using AAIndex1 it is important to comprehend the principles behind

creation of machine learning predictors for PTM detection. Essentially, such methodology enables us to discover protein sequence motifs which are triggers for modifications. However, using raw amino acid sequence may lead us to learning relatively rigid motifs. On the other hand, changing amino acids into sets of their physicochemical attributes gives the algorithm more flexibility in motif definition. Often it may happen that for a PTM occurrence it is important to have an amino acid with particular characteristic at some neighboring residue index rather than to have a particular amino acid at this place [9]. Similar effect is achieved with the use of position-specific scoring matrices. Figure 3 visualizes how the sequence window is transformed into AAIndex attributes.

Since the AAIndex1 data set is very large, it is important to select a small subset of physicochemical indices that reflect the most important characteristics of amino acids. Table 4 presents eight features that were identified as a representative set, covering all the major attributes types present in AAIndex1 [28].

The discussed subsets of features from the AAIndex database are not the only possibility. For instance, it is also common to use explicit information on the protein secondary structure or features like protein disorder [29], surface area accessible to solvent [30], or hydrophobicity [31]. Some prediction methods were using position-specific scoring matrices [32] and k-nearest neighbor profiles [33] to improve predictive power of PTM classifiers. For an overview of features used in PTM prediction refer to, e.g. dbPTM database documentation (http://dbptm.mbc.nctu.edu.tw).

3.4 Classifier Choice

The following three algorithms are most commonly used in research aiming at building PTM predictors (these are also algorithms of choice in other prediction/classification tasks in bioinformatics).

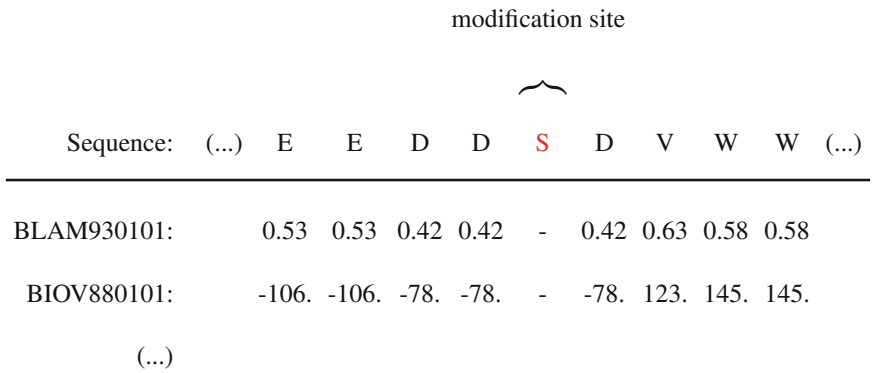


Fig. 3 Translating sequence neighborhood of a PTM site into values of physicochemical features from AAIndex1 database

Table 4
Eight physicochemical features selected from AAIndex1 database as a representative set covering most of the present attribute types [28]

Amino acid index	Description	Feature cluster[a]
BLAM930101	Alpha helix propensity of position 44 in T4 lysozyme (Blaber et al. 1993)	Electric properties
BIOV880101	Information value for accessibility; average fraction 35 % (Biou et al. 1988)	Hydrophobicity
MAXF760101	Normalized frequency of alpha-helix (Maxfield-Scheraga 1976)	Alpha and turn propensities
TSAJ990101	Volumes including the crystallographic waters using the ProtOr (Tsai et al. 1999)	Physicochemical properties
NAKH920108	AA composition of MEM of multi-spanning proteins (Nakashima-Nishikawa 1992)	Residue propensity
CEDJ970104	Composition of amino acids in intracellular proteins (percent) (Cedano et al. 1997)	Composition
LIFS790101	Conformational preference for all beta-strands (Lifson-Sander 1979)	Beta propensity
MIYS990104	Optimized relative partition energies—method C (Miyazawa-Jernigan 1999)	Intrinsic propensities

[a]As reported by Saha et al. [28]

Random Forest consists of building an ensemble of decision tree predictors. The purpose is to reduce the variance associated with using a single decision tree. In order to achieve that, trees need to be uncorrelated. Low correlation of trees is ensured by building training sets for each tree with bagging [34] and using just a random subset of features at each split node [35]. An important advantage of this method is little parameter adjustment required for its successful application [36]. Moreover, the computational cost of training a random forest is relatively low.

Support Vector Machine determines a hyperplane that splits the decision classes with the largest possible margin in the feature space. Important strength of this method is the "kernel trick" which enables cost-effective hyperplane determination in space of higher dimensionality than the original feature space. Unfortunately, successful usage of this approach requires skill with the use of different kernels and significant effort at the stage of searching for appropriate parameters. In general the method has higher computational cost in learning phase than Random Forest [37].

Artificial Neural Networks were invented as a set of methods that were meant to simulate the neuronal activity of the human brain. They can be designed in complex schemas which also has an effect on their high computational complexity.

To a person with limited experience in machine-learning, we strongly recommend the use of Random Forest as off-the-shelf algorithm.

4 Discussion

4.1 Identification of Negative Data Records

As described in Subheading 19.3.2.2, identifying examples of PTM occurrences (positive examples) with help of domain databases (Subheading 19.2.1) is relatively easy. However, the case with records of PTM non-occurrence (negative examples) is more complicated. As such, popular databases do not provide information about positions in protein sequence that are not subject to PTM under any circumstances. A solution is to just use sites that were not marked as being modified. That is, when building a data set for predictor of type *T* modification, as negative records we can gather a predefined number of sequence sites that were not marked as being subject of type *T* modification (they still can be subject of a different type of PTM).

However, one must be aware of the fact that there exist potentially quite large number of modification sites that have never been experimentally checked for being subject to a PTM in question (*T*). Thus, they are not included in PTM databases. One idea to address this issue is to gather negative records only from the proteins that do have some PTM annotations (not necessarily for the modification type *T*) as this guarantees that some experimental examination was performed on them. Nevertheless, it still does not guarantee that the particular sequence site we extract was subject to relevant experiments [24].

More strict approach is to gather non-occurrence sites of type *T* modification only from proteins that do contain annotation of type *T* PTMs. This, however, may lead to a significant bias as some proteins may be naturally less prone to be subject of particular types of modifications. This will introduce bias to data set construction and negatively impact the ability of our predictor to learn how to recognize proteins not having any modification sites.

4.2 Positive and Negative Class Imbalance

Subheading 19.3.2.3 explains the reasons for the presence of significant class imbalance between positive and negative records of PTM occurrences.

In order to shed light to the amounts of available data and possible class imbalance, in Tables 5 and 6 we present statistics of PTM annotations available in UniProt. What's more, Fig. 4 shows how known PTM sites are distributed across proteins that have at least a single modification annotation.

From the presented data we observe that the potential class imbalance between positive and negative PTM occurrence sites may range from below 1:100 up to around 1:20,000. This intrinsic property of PTM data has the following implications for the prediction problem:

Table 5
Statistics of PTM annotations in UniProt database

Figure	Value
All proteins	53,333,247
Proteins with any PTM annotation	361,336
Proportion of proteins containing any PTM annotation	0.68%

Table 6
Descriptive statistics illustrating disproportion between known PTM occurrence sites and potential sites of modifications

PTM type	PTM sites marked	PTM-relevant amino acids	Known vs. potential
Phosphoserine	113,319	8,896,770	1:77.5
N6-acetyllysine	43,335	8,067,140	1:185.0
N5-methylglutamine	14,926	5,735,174	1:383.0
4-hydroxyproline	1,805	6,896,532	1:3,820.0
Phosphoserine by CK1	1,245	8,896,770	1:7,145.0
Leucine amide	737	14,191,541	1:19,255.0
Phosphoserine by PKA	692	8,896,770	1:12,856.0

The presented "known vs. potential" ratio is the best available estimate ofpotential class imbalance between positive and negative PTM site records.However, many more modification sites may be present, but not yet discovered.Analysis was performed for UniProt on proteins containing at least one PTMannotation

1. When designing a PTM predictor one should put more stress on specificity rather than on sensitivity. The strong domination of negative examples implies that a low-specificity predictor would flood the user with false positives. Deeper analysis of this aspect is presented in Subheading 19.4.4.
2. Some technique of balancing positives and negatives has to be used as the majority of machine learning algorithms will fail to perform an efficient discrimination between classes.

In the case of PTM prediction, we are especially concerned with the problem of negative records domination. The reason for that is the fact that we are convinced that a large number of sites which today we have to treat as negatives are in fact positives which have not been experimentally validated yet. Therefore, should we include too many negative examples in our training set,

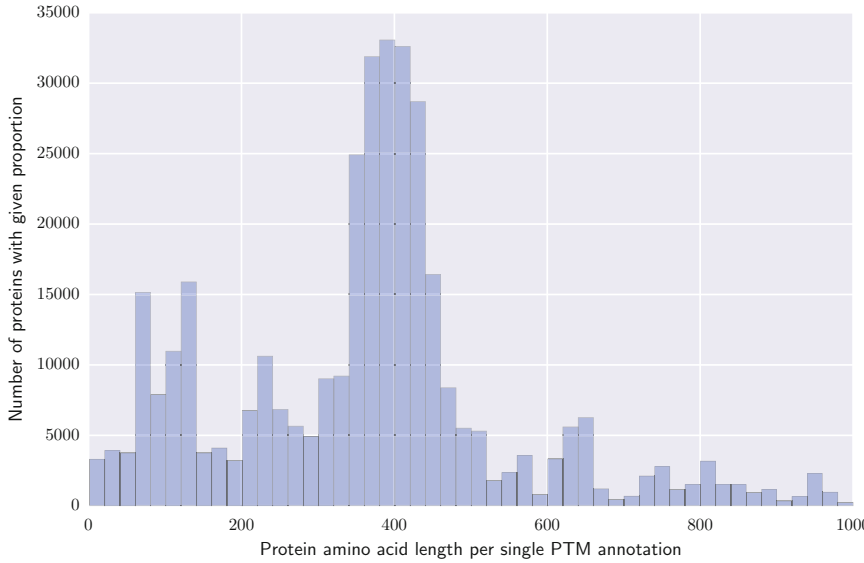


Fig. 4 Distribution of proteins from UniProt by the ratio of their total amino acid length to number of PTM annotations. Around 3000 proteins for which the ratio was higher than 1000 are omitted in the figure

considerable number of them would likely be mislabeled positives. Thus, a significant amount of noise will be introduced that, in some cases, would heavily impair discriminatory ability of our classifier.

The following techniques are commonly used for class imbalance problems in machine learning [38]:

Over-sampling Repeated use of records from underrepresented decision class or their artificial creation from estimated distribution. In case of over-sampling positive records in PTM prediction we do not recommend the latter approach, because even a minimal deviation from a modification-triggering amino acid motif may lead to obtaining a sequence segment that stops to be subject of the modification. That is, modification-triggering motifs might be very subtle.

Under-sampling Choosing only the subset of available records from the overrepresented class. This approach seems especially suitable for the negative records in PTM problem. Many of the sites which we have to treat as negatives are in fact positive examples which have not been tested for modifications yet. Therefore, taking all of them into the training set would lead to masking/shadowing the known positives, thus making the class discrimination task impossible to perform. On the other hand, randomly choosing a subset of negatives minimizes this risk. We recommend the use of this technique in training PTM predictors. For predicting

phosphorylations, the 1:1 ratio has been proposed as a good compromise between true-positive rate and false-positive rate [8].

Cost-modification Modifying the relative cost associated with misclassification of positive and negative class.

4.3 Construction of Cross-Validation Folds Considering Sequence Homology

The choice of folds in a standard cross-validation procedure, described in Subheading 19.3.1.4, is usually realized through random splitting of the full training-testing set into subsets of equal size and equal proportion of positives and negatives. Such functionality is also commonly offered in the majority of popular machine learning libraries. While correct in the case of many other types of studies, application of such approach to protein data may lead to somewhat misleading results. In many domains construction of cross-validation folds does not take into account the relationships present in the underlying biological data which, in turn, leads to overoptimistic performance estimates [39, 40].

In the case of PTM prediction, we have to be aware of having included many homologous data records in our training set. This is caused by the fact that similar proteins from the same family often have modification sites at the same sequence positions. The amino acid neighborhoods of these sites tend to be almost the same, if not exactly identical. Putting such homologous records simultaneously into both the training and the test sets of the same cross-validation fold inevitably leads to overfitting. In order to avoid this problem we recommend to base the data split on homology clustering of the full training set. One practical implementation of this solution is by using the leave-cluster-out method [39]. However, this algorithm might not be optimal when we have several singletons in our clustered training set (which is often the case for protein sequence data). Leave-cluster-out also does not preserve the ratio of negative to positive data records and does not allow for controlling the number of cross-validation folds.

The problem of arranging data record clusters into a predefined number of folds that have possibly the most similar size and at the same time even approximately preserve the ratio of negatives to positives is an NP-hard problem (the *subset sum problem* can be reduced to this). However, using slightly suboptimal split does not seem to be very harmful regarding prediction performance or overfitting problems. Therefore, we propose the *folds-over-clusters* heuristic algorithm that tries to find a "good-enough" split. The details of the procedure are presented in Algorithm 1, while the main steps can be summarized in the following way:

1. Using amino acid motif similarity cluster the full training set into subsets of homologous records.
2. Sort the obtained clusters by per cluster negatives to positives ratio.
3. Randomly assign one cluster to every cross-validation fold.

4. Distribute the remaining clusters among the folds by iterative application of the following steps:
   (a) If a fold reached required size (number of records divided by number of folds), continue.
   (b) If the negatives to positives ratio in the fold is too high, add the cluster with the lowest ratio of negatives to positives from the remaining clusters (the ones unassigned to any fold).
   (c) Otherwise, add cluster with the highest negatives to positives ratio from the clusters still unassigned to folds.

Since the folds-over-clusters algorithm begins fold enrichment with clusters having extreme negatives to positives ratios, it has time to balance these extremes over the run of it is second loop. During last iterations of the second loop only the clusters with ratios close to the global ratio of negatives to positives remain, as they were kept in the middle of sorted clusters collection.

One could question whether trying to approximately preserve the global ratio of negatives to positives inside each fold is important at all. In order to understand that let us imagine the worst case—when in a twofold cross-validation all the positives are assigned to a single fold. In such case machine learning algorithms are hardly able to infer the proper discrimination of negative and positive classes in the data. Thus classification performance is heavily affected.

4.4 Importance of Optimizing for Specificity

In Subheading 19.4.2, we have already discussed the imbalance of positive and negative records in the training data for PTM predictor construction. The major factor underlying this imbalance is the fact that protein modifications are relatively rare events: there exist many more potential modification sites than the actual occurring modifications. This has important implications on the application of PTM predictors in experimental biology. Experimentalists' main aim is to identify unknown PTM sites. Following an in silico discovery, sites should be subject to experimental validation before being deemed an actual PTM site. Therefore, in silico studies should primarily help narrowing down the set of potential candidates. In order to serve this purpose, the tools should focus on maximizing specificity at the expense of lower sensitivity (the relationship between these measures is described in Subheading 19.3.1.3). More detailed consideration of this topic, along with examples and study of overfitting has been recently performed by Daniel Schwartz [41].

4.5 Searching Prediction Features Outside of Local Sequence

The majority of existing PTM prediction methods focuses on the use of sequence as the source of learning features. However, there are also two clear trends in reaching for additional data. These other types of features are:

Protein structure As we have already mentioned in Subheading 19.3.2.4, an important factor in a modification process is the enzyme which catalyzes the actual chemical reaction (e.g., kinases discussed in Subheading 19.3.2.4 are the enzymes catalyzing phosphorylations). The interaction between a protein and an enzyme makes the structure of the protein an important factor for the modification process. Protein structure defines how accessible a potential PTM site is to the enzyme and which residues constitute the actual neighborhood of the PTM site during the contact with the enzyme. Therefore, it seems plausible that incorporating the structural information might improve the performance of PTM prediction. However, studies testing this approach reported somewhat ambiguous results [7, 42]. Even if this direction proves to be successful, it may be of limited use as the available structural information are still scarce. Nevertheless, it seems to be a promising and interesting topic for further research [11].

Evolutionary information The most recent innovation in the field are attempts to benefit from the use of evolutionary information. Since PTM sites are often conserved across different organisms, it is possible to identify putative sites by across-organism comparative studies, even without the use of machine learning methods [10–12].

4.6 Feature Selection

When working on machine learning tasks it is often the case that we are able to extract a large number of features from our data. Building predictive models with the use of all features poses challenges of high computational cost, problems with interpretation of the predictors and may introduce unwanted noise. Therefore, it is a common practice to select only the features which provide significant gains in the performance of predictive model.

In the case of PTM prediction, it is rather easy to include many more features than we and our algorithms can handle. For instance, using only all available features from AAIndex (*see* Subheading 19.3.3.2) along with a long window of amino acids leaves us with several thousands of attributes. This can be especially problematic with PTM types for which the training data is scarce as it leads to the so-called *ill-defined* problem where the number of available examples is much lower than the number of features. Enriching the model with features related to protein structure or evolutionary information alleviates the problem even more.

The specific problem of abundant set of features in the AAIndex database has been addressed by attempts to establish a set of representative attributes by using, e.g. clustering [28].

On the other hand, the challenge of selecting from set of heterogeneous features or features from different sequence positions needs to be tackled with more generic approaches. The classic step-wise methods can be computationally too expensive in this context.

However, we can use methods based on random forest which seem to overcome the mentioned difficulties at least to some extent [43, 44].

Algorithm 1 *Folds-over-clusters* algorithm for splitting training set into k cross-validation folds of PTM site records. Split is based on homology-clustering of amino acid motifs of the immediate neighborhood of the site. Algorithm produces similar size folds which approximately preserve the initial ratio of negatives to positives. Clustering can be performed with CD-HIT [45], BLASTClust [46] or similar programs

Data: Training set containing n modification site records with negatives to positives rario r.

Result: Split of the training set into k folds containing approximately n/k records with r ratio of negatives to positives. If a fold contains records from a homology cluster, it has to include all records from this cluster.

```
cluster the training set based on homology;
sort clusters based on their negatives to positives ratio;
foreach fold do
    add random cluster to the fold and remove this cluster from unused clusters set;
end
while there are unused clusters left do
    foreach fold do
        if no unused clusters left then
            break;
        end
        if fold achieved required size then
            continue;
        end
        if fold has too high negatives to positives ratio then
            add cluster with the lowest negatives to positives ratio
        else
            add cluster with the highest negatives to positives ratio
        end
    end
end
```

Acknowledgements

Marcin Tatjewski was supported by the European Union from resources of the European Social Fund. Project PO KL "Information technologies: Research and their interdisciplinary applications", Agreement UDA-POKL.04.01.01-00-051/10-00. Marcin Tatjewski and Dariusz Plewczynski were supported by Polish National Science Centre (grant numbers: 2015/16/T/ST6/00493, 2014/15/B/ST6/05082 and 2013/09/B/NZ2/00121) and EU COST BM1405 and BM1408 actions. Marcin Kierczak was supported by the Swedish Foundation for Strategic Research and the Swedish Research Council.

References

1. Uhlen M, Ponten F (2005) Antibody-based proteomics for human tissue profiling. Mol Cell Proteomics 4:384–393
2. Jensen ON (2004) Modification-specific proteomics: characterization of post-translational modifications by mass spectrometry. Curr Opin Chem Biol 1:33–41
3. Walsh C (2006) Posttranslational modification of proteins: expanding nature's inventory. Roberts and Company Publishers, Englewood, CO
4. Irby RB, Yeatman TJ (2000) Role of Src expression and activation in human cancer. Oncogene 19(49):5636–5642
5. Brown M, Cooper JA (1996) Regulation, substrates and functions of Src. Biochim Biophys Acta 1287:121–149
6. Abram CL, Courtneidge SA (2000) Src family tyrosine kinases and growth factor signaling. Exp Cell Res 254:1–13
7. Blom N, Gammeltoft S, Brunak S (1999) Sequence and structure-based prediction of eukaryotic protein phosphorylation sites. J Mol Biol 294(5):1351–1362. doi:10.1006/jmbi.1999.3310
8. Biswas AK, Noman N, Sikder AR (2010) Machine learning approach to predict protein phosphorylation sites by incorporating evolutionary information. BMC Bioinf 11(1):273. doi:10.1186/1471-2105-11-273
9. Plewczynski D, Basu S, Saha I (2012) AMS 4.0: consensus prediction of post-translational modifications in protein sequences. Amino Acids 43(2):573–582. doi:10.1007/s00726-012-1290-2
10. Jalal S, Arsenault R, Potter AA, Babiuk LA, Griebel PJ, Napper S (2009) Genome to kinome: species-specific peptide arrays for kinome analysis. Sci Signal 2(54):pl1. doi:10.1126/scisignal.254pl1
11. Trost B, Kusalik A (2011) Computational prediction of eukaryotic phosphorylation sites. Bioinformatics (Oxford, England) 27(21):2927–2935. doi:10.1093/bioinformatics/btr525
12. Trost B, Arsenault R, Griebel P, Napper S, Kusalik A (2013) DAPPLE: a pipeline for the homology-based prediction of phosphorylation sites. Bioinformatics (Oxford, England) 29(13):1693–1695. doi:10.1093/bioinformatics/btt265
13. Robertson AJ, Trost B, Scruten E, Robertson T, Mostajeran M, Connor W, Kusalik A, Griebel P, Napper S (2014) Identification of developmentally-specific kinotypes and mechanisms of Varroa mite resistance through whole-organism, kinome analysis of honeybee. Front Genet 5:139. doi:10.3389/fgene.2014.00139
14. The UniProt Consortium (2014) UniProt: a hub for protein information. Nucleic Acids Res 43(D1):D204–D212. doi:10.1093/nar/gku989
15. Hornbeck PV, Zhang B, Murray B, Kornhauser JM, Latham V, Skrzypek E (2015) PhosphoSitePlus, 2014: mutations, PTMs and recalibrations. Nucleic Acids Res 43(Database issue):D512–D520. doi:10.1093/nar/gku1267
16. Dinkel H, Chica C, Via A, Gould CM, Jensen LJ, Gibson TJ, Diella F (2011) Phospho.ELM: a database of phosphorylation sites–update 2011. Nucleic Acids Res 39(Database issue):D261–D267. doi:10.1093/nar/gkq1104
17. Kamath KS, Vasavada MS, Srivastava S (2011) Proteomic databases and tools to decipher post-translational modifications. J Proteomics 75(1):127–144. doi:10.1016/j.jprot.2011.09.014

18. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2012) Scikit-learn: machine learning in python. J Mach Learn Res 12:2825–2830. 1201.0490
19. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH (2009) The WEKA data mining software. In: ACM SIGKDD explorations newsletter, vol 11, issue 1, p 10. doi:10.1145/1656274.1656278
20. Samuel A (2000) Some studies in machine learning using the game of checkers. IBM J Res Dev 44(1.2):206–226. doi:10.1147/rd.441.0206
21. Provost F, Fawcett T, Kohavi R (1998) The case against accuracy estimation for comparing induction algorithms. In: Proceedings of the fifteenth international conference on machine learning. Morgan Kaufmann, San Francisco, pp 445–453
22. Matthews B (1975) Comparison of the predicted and observed secondary structure of {T4} phage lysozyme. Biochim Biophys Acta Protein Struct 405(2):442–451. http://dx.doi.org/10.1016/0005-2795(75)90109-9
23. Powers DM (2011) Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. J Mach Learn Technol 2(1):37–63
24. Neuberger G, Schneider G, Eisenhaber F (2007) pkaPS: prediction of protein kinase A phosphorylation sites with the simplified kinase-substrate binding model. Biol Direct 2:1. doi:10.1186/1745-6150-2-1
25. Jung I, Matsuyama A, Yoshida M, Kim D (2010) PostMod: sequence based prediction of kinase-specific phosphorylation sites with indirect relationship. BMC Bioinf 11(Suppl 1):S10. doi:10.1186/1471-2105-11-S1-S10
26. Kawashima S (2000) AAindex: amino acid index database. Nucleic Acids Res 28(1):374. doi:10.1093/nar/28.1.374
27. Kawashima S, Pokarowski P, Pokarowska M, Kolinski A, Katayama T, Kanehisa M (2008) AAindex: amino acid index database, progress report 2008. Nucleic Acids Res 36(Database issue):D202–D205. doi:10.1093/nar/gkm998
28. Saha I, Maulik U, Bandyopadhyay S, Plewczynski D (2012) Fuzzy clustering of physicochemical and biochemical properties of amino acids. Amino Acids 43(2):583–594. doi:10.1007/s00726-011-1106-9
29. Iakoucheva LM, Radivojac P, Brown CJ, O'Connor TR, Sikes JG, Obradovic Z, Dunker AK (2004) The importance of intrinsic disorder for protein phosphorylation. Nucleic Acids Res 32(3):1037–1049. doi:10.1093/nar/gkh253
30. Lee TY, Hsu JBK, Lin FM, Chang WC, Hsu PC, Huang HD (2010) N-Ace: using solvent accessibility and physicochemical properties to identify protein N-acetylation sites. J Comput Chem 31(15):2759–2771. doi:10.1002/jcc.21569
31. Chen YZ, Chen Z, Gong YA, Ying G (2012) SUMOhydro: a novel method for the prediction of sumoylation sites based on hydrophobic properties. PloS One 7(6):e39195. doi:10.1371/journal.pone.0039195
32. Pejaver V, Hsu WL, Xin F, Dunker AK, Uversky VN, Radivojac P (2014) The structural and functional signatures of proteins that undergo multiple events of post-translational modification. Protein Sci 23(8):1077–1093. doi:10.1002/pro.2494
33. Li A, Wang L, Shi Y, Wang M, Jiang Z, Feng H (2005) Phosphorylation site prediction with a modified k-nearest neighbor algorithm and blosum62 matrix. In: 27th Annual International conference of the engineering in medicine and biology society, 2005 (IEEE-EMBS 2005), pp 6075–6078. doi:10.1109/IEMBS.2005.1615878
34. Breiman L (1996) Bagging predictors. Mach Learn 24(2):123–140. doi:10.1007/BF00058655
35. Ho TK (1998) The random subspace method for constructing decision forests. IEEE Trans Pattern Anal Mach Intell 20(8):832–844. doi:10.1109/34.709601
36. Breiman L (2001) Random forests. Mach Learn 45(1):5–32. doi:10.1023/A:1010933404324
37. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297. doi:10.1007/BF00994018,arXiv:1011.1669v3
38. Japkowicz N, Stephen S (2002) The class imbalance problem: a systematic study. Intelligent Data Anal 6(5):429–449
39. Kramer C, Gedeck P (2010) Leave-cluster-out cross-validation is appropriate for scoring functions derived from diverse protein data sets. J Chem Inf Model 50(11):1961–1969. doi:10.1021/ci100264e
40. Zubek J, Tatjewski M, Boniecki A, Mnich M, Basu S, Plewczynski D (2015) Multi-level machine learning prediction of protein-protein interactions in Saccharomyces cerevisiae. PeerJ 3:e1041. doi:10.7717/peerj.1041
41. Schwartz D (2012) Prediction of lysine post-translational modifications using bioinformatic tools. Essays Biochem 52:165–177. doi:10.1042/bse0520165
42. Durek P, Schudoma C, Weckwerth W, Selbig J, Walther D (2009) Detection and characterization of 3D-signature phosphorylation

site motifs and their contribution towards improved phosphorylation site prediction in proteins. BMC Bioinf 10(1):117. doi:10.1186/1471-2105-10-117

43. Rudnicki WR, Kierczak M, Koronacki J, Komorowski J (2006) A statistical method for determining importance of variables in an information system. In: Rough sets and current …, pp 557–566. doi:10.1007/11908029_58
44. Draminski M, Rada-Iglesias A, Enroth S, Wadelius C, Koronacki J, Komorowski J (2008) Monte Carlo feature selection for supervised classification. Bioinformatics (Oxford, England) 24(1):110–117. doi:10.1093/bioinformatics/btm486
45. Li W, Godzik A (2006) Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics (Oxford, England) 22(13):1658–1659. doi:10.1093/bioinformatics/btl158
46. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ, Pennsylvania T, Park U (1990) Basic local alignment search tool. J Mol Biol 215(3):403–410. doi:10.1016/S0022-2836(05)80360-2

Chapter 20

CX, DPX, and PCW: Web Servers for the Visualization of Interior and Protruding Regions of Protein Structures in 3D and 1D

Balázs Ligeti, Roberto Vera, János Juhász, and Sándor Pongor

Abstract

The CX and DPX web-based servers at http://pongor.itk.ppke.hu/bioinfoservices are dedicated to the analysis of protein 3D structures submitted by the users as Protein Data Bank (PDB) files. CX computes an atomic protrusion index, *cx* that makes it possible to highlight the protruding atoms within a protein 3D structure. DPX calculates a depth index, *dpx* for buried atoms, and allows one to visualize the distribution of buried residues. CX and DPX visualize 3D structures colored according to the calculated indices and return PDB files that can be visualized using standard programs. A combined server site, the Protein Core Workbench allows visualization of *dpx*, *cx*, solvent-accessible area as well as the number of atomic contacts as 3D plots and 1D sequence plots. Online visualization of the 3D structures and 1D sequence plots are available in all three servers. Mirror sites are available at http://hydra.icgeb.trieste.it/protein/.

Key words Atomic depth index, Protrusion index, Solvent-accessible surface area, Atomic contact numbers, Protein core

1 Introduction

As large amounts of protein structure data are generated, there is a growing need for simple methods that experimentalists and students can use to visualize protein structures. Many simple properties can be calculated and visualized with programs such as Rasmol [1], MolMol [2], Deep-View [3], or PyMOL [4]; however, the visualization of core and protruding regions that often play interesting functional roles in proteins are not routinely included in standard programs.

Over the past years our group has been developing and testing algorithms for protein [5–21] as well as nucleic acid structure analysis [22–29]. These methods have been incorporated into web-based server programs hosted at our web site and have been gradually extended to the calculation and visualization of a variety of parameters [21, 27, 29]. Visualization of molecular properties consists in mapping numerical data to low dimensional structural

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2_20,

templates which in turn can be either theoretical (such as an ideal α-helix) or experimentally determined (such as a protein 3D structure). The simplest case of visualization is the sequence plot in which numeric values are assigned to a 1D theoretical template i.e. positions along the protein sequence.

The protein server programs reported here take protein structure inputs and provide (1) 1D plots of structural properties, (2) 3D visualization of the properties mapped to protein structure, as well as (3) numerical outputs in the form of PDB files or a tabulated sequence plots. These outputs allow users to view and explore their data interactively so as to produce figures directly on the server page. In addition the servers return the output data also in tabular form for later use by dedicated visualization molecular graphics programs.

In this chapter we describe the substantially updated versions of two servers, DPX and CX that are based on geometric properties of individual atoms within a macromolecular structure. DPX visualizes the protein interior using a depth index, *dpx* which is high for atoms within the protein core and small for those near the surface [11, 16, 17, 21]. CX calculates a measure of atomic exposure, *cx*, which is high for atoms in protruding regions [14, 21]. Both *dpx* and *cx* are defined for atoms and not for amino acid residues. Nevertheless the programs calculate various residue averages (for instance the average *dpx* score for main chain, side-chain, or all atoms of a residue, respectively) which can then be used to produce 1D plots. Below we describe both servers, along with the underlying principles and a few application case studies.

2 Software

The programs underlying the web-based servers described here were written in ANSI C (C89), the web pages were written in HTML5 with the JavaScript framework, AngularJS (version 1.2.0), JavaScript and PHP. The JSMOL [30] framework is used for displaying the 3D structures of the molecules. External programs called within the C programs include NACCESS [31], GNUPLOT [32]. The web-based servers were tested with Mozilla Firefox (version 41.0.2) Google Chrome (version 46.0), Internet Explorer (version 11.0.2) web browsers.

3 Methods

3.1 DPX: Visualizing the Protein Core

3.1.1 Theory

The thinking of biologists has been profoundly influenced by simplified views that divide protein structures into loosely defined regions such as surface and core, the latter denoting the interior of the protein which is not in immediate contact with the solvent around the protein. Although intuitively clear, the mathematical definition of

the core region is not straightforward since, among others, one needs to define what solvent contact means in numerical terms. Since the seminal paper of Richards [33], an atom is considered to be in contact with a solvent if it can be touched by a spherical probe of radius *r* rolled around the protein structure. The radius of the probe has to be defined, which immediately leads to alternative definitions based on different probe diameters, for instance a radius of 1.4 Å is considered to be "water like" and suitable to find water accessible atoms. Additional ambiguities arise from the fact that not all atoms of an amino acid residue will be in contact with the solvent. In order to steer clear of these ambiguities we defined a depth index, *dpx*, as the distance of an atom from the nearest water-accessible atom, the latter characterized by a greater than zero solvent accessibility calculated with a water-like probe. By definition, *dpx* is greater or equal to zero, and the maximum value is half of the maximal diameter of the protein molecule. *dpx* is large for buried atoms and zero for atoms that are in contact with the solvent.

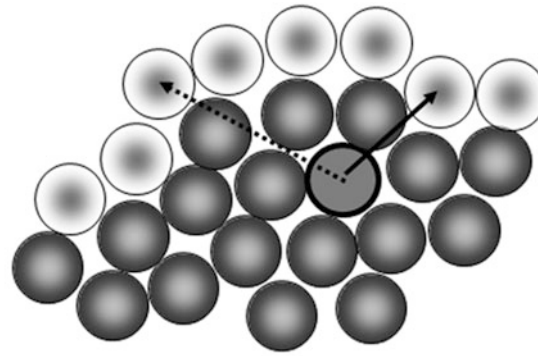
3.1.2 Program Description

The algorithm first calculates solvent accessibility of each atom based on the Richards probe principle [33], then it determines the distance from the nearest water accessible atom according to the conditions outlined in Fig. 1. This value is reported as the *dpx* score of the atom.

The atomic *dpx* score is reported in the PDB file of the input structure, and can be used to color the atoms for viewing in 3D, using the JMOL program [30]. The all-atom residue average of the *dpx* score is also reported as a 1D sequence plot which can be viewed on the screen (Fig. 2), in addition a tabulated output is also available in which all residue averages are shown. The program is written in C, the home-page is available at http://pongor.itk.ppke.hu/cx

DPX reads standard PDB coordinate files as the input. In the present form, DPX is aimed at the analysis of the interior of single chain monomeric proteins, so users are encouraged to inspect and, if necessary, edit the PDB file in order to have only one chain in the file. The program reads only ATOM lines. Thus, HETATM lines describing nonstandard residues, cofactors, metal ions, and water molecules are not taken into account.

The results include (a) a PDB file in which the atomic dpx values are in the last column of the atom lines (i.e. they replace the B-factor values stored in characters 61–66) and solvent accessible surface area replaces the occupancy values stored in characters 55–60; (b) the same PDB file can be viewed online using JMOL (Fig. 3); (c) Residue-aggregated values of atomic *dpx* values (average *dpx* for all atoms, main chain atoms, side-chain atoms) given in tabulated form; (d) Sequence plot of residue-aggregated values, visualized online using GNUPLOT.



$$If\ asa_i > 0 \Rightarrow dpx_i = 0$$
$$If\ asa_i = 0\ AND\ asa_j > 0 \Rightarrow dpx = min(d_{ij})$$

Fig. 1 Schematic representation of the DPX principle. Solvent exposed atoms are shown in white, buried atoms in gray; the atom (i) for which *dpx* is calculated is in black. Arrows represent distances between (i) and surface atoms. *asa* denotes atomic solvent accessibility calculated by DPX according to the Richards probe principle [33], the default probe radius being 10 Å

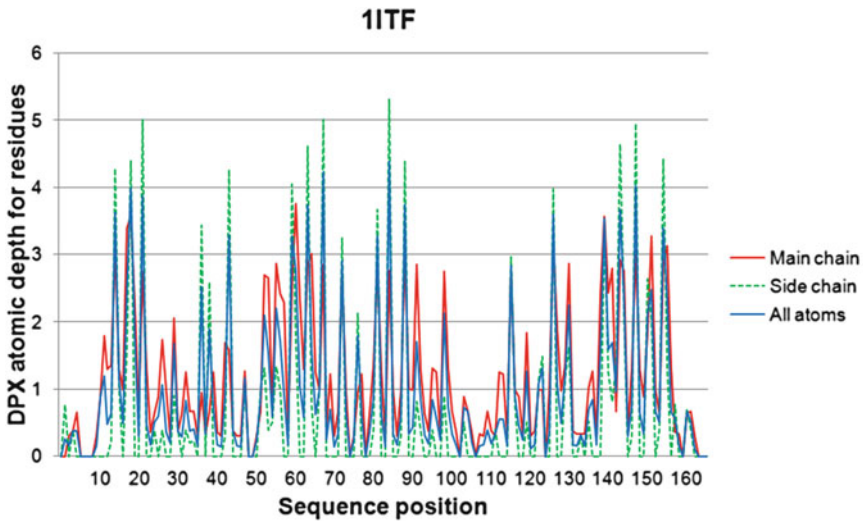


Fig. 2 Example of a dpx (atomic depth index) sequence plot. Dpx is an atomic property that needs to be aggregated according to various methods so as to give a residue property. For simplicity, only one of these should be used in a figure

3.2 CX: Visualization of Protruding Regions in Protein Structures

3.2.1 Theory

The identification of protruding, or highly convex regions in proteins is important for studying functionally important sites including antigenic determinants, proteolysis cleavage sites, or protein–protein interfaces. Traditional methods use complex, residue-based algorithms for the identification of these potential functional sites [34–36]. These methods are computationally

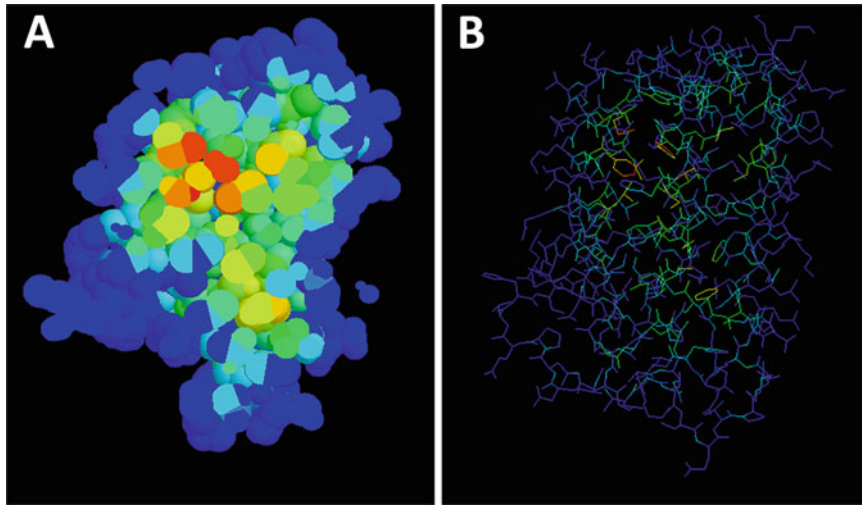


Fig. 3 Structure of the human interferon alpha 2A colored according to *dpx* (atom depth) (PDB: 1itf [37]) A) The CPK model of the protein is shown in slab mode and atoms colored according to their *dpx* values (high *dpx* = red, low *dpx* = blue). B) Wire representation of the same molecule

intensive and parameter-dependent. In addition, residue-based indices are coarse-grained descriptors of the real geometry of the protein surface. Here we developed the *cx*, a simple and atomic-level protrusion index.

3.2.2 Program Description

For each non-hydrogen atom of a polypeptide chain, the program first counts the number atoms within a certain radius R. Then it calculates the ratio of the free volume within the sphere V_{ext} to the volume occupied by atoms V_{int} (principle shown in Fig. 4). The program is written in C, the visualization interface is available at http://pongor.itk.ppke.hu/cx. 1D plots are produced by GNUPLOT [32], the 3D structures are visualized online with JMOL [30].

CX reads standard PDB coordinate files as the input. The program reads only ATOM lines. Thus, HETATM lines describing nonstandard residues, cofactors, metal ions, and water molecules are not taken into account. By default, the program treats each chain in the PDB file as an independent molecule (i.e. the atoms of chain B are not taken account when calculating the protrusion index for the atoms of chain A) but the results are written into a single file.

The program produces the following outputs: (a) a coordinate file in PDB format in which the atomic displacement parameter (B-factor, or temperature factor stored in characters 61–66) is replaced by the cx value, and the number of nontrivial atomic contacts (no atoms within the sphere in Fig. 4, that belong to a different residue) replaces the occupancy values stored in characters 55–60. This file can be thus displayed on the user's own workstation using standard molecular graphics programs, atoms can be colored by their *cx* values. (b) The same file can be visualized on

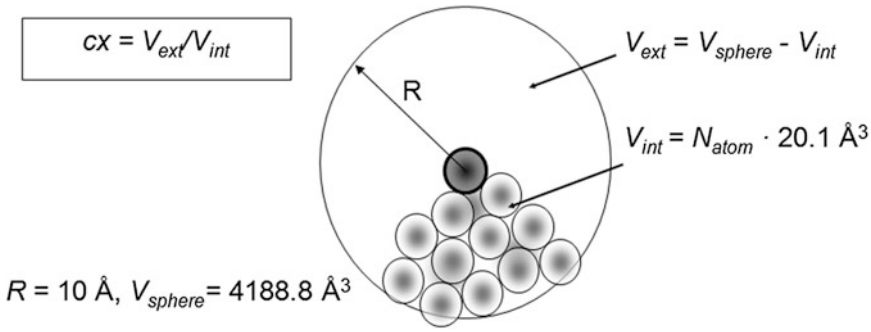


Fig. 4 Schematic representation of the CX algorithm. The *cx* score of an atom is defined for non-hydrogen protein atoms as V_{ext}/V_{int} i.e. the ratio of the external volume, not occupied by atoms, divided by the volume occupied by protein atoms. V_{int} is calculated by multiplying N_{atom}, the number of non-hydrogen atoms within a distance *R*, with 20.1 Å^3, the average volume of a non-hydrogen atom in a protein [40] The default radius of the spherical probe is 10 Å so the external volume can be calculated by subtracting V_{int} from 4188.8 Å^3, the volume of the sphere. For protein atoms, the ratio $cx = V_{ext}/V_{int}$ is between 0 and ~15, with protruding atoms having higher *cx* values

line using JMOL (Fig. 5); (c) Residue-aggregated values of atomic *cx* values (average *cx* for all atoms, main chain atoms, side-chain atoms) given in tabulated form; (d) Sequence plot of residue-aggregated values, visualized online using GNUPLOT.

3.3 PCW: The Protein Core Workbench

This server is a new development which is meant for users who are already familiar with the principles CX and DPX, and who want to create publication style plots of their molecules. PCW combines the features of CX and DPX, with a few other options that allow the users to create combined sequence plots. The plottable parameters are *cx*, *dpx*, accessible surface area, and number of atomic contacts. The accessible surface area (ASA) is calculated according to the Richards principle (default probe radius is 1.4 Å), the number of atomic contacts (NAC) is the number of those atoms within a sphere of given radius (default = 10 Å) that do not belong to the same residue as the atom in question. Both ASA and NAC are atomic parameters and their averages for all atoms, main chain atoms and side chain atoms are calculated for residue-based plots. An example is shown in Fig. 6.

4 Notes

1. The web-based servers at PPKE have been created for the analysis of user-submitted protein 3D structures in terms of buried and solvent-exposed atoms. *DPX* calculates an atomic depth index, *dpx* [11, 16, 17, 21] suitable for the visualization of the protein core, *CX* calculates a protrusion index, *cx* [14, 21], suitable for the visualization of protruding segments of the submitted structure. The Protein Core Workbench PCW allows users

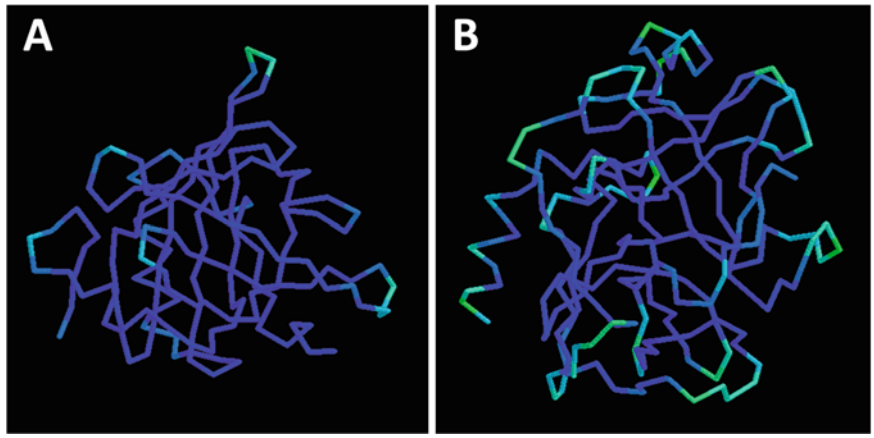


Fig. 5 Protein structures colored according to *cx* (protrusion index). **A**) Structure of a pheromone-binding murine alpha-2-globulin protein PDB: 1mup [38]. The main-chain wire model of the protein is colored according to the atomic *cx* values (high *cx* = turquoise, low *cx* = *blue*). **B**) Structure of the HIV-1 integrase protein PDB: 1tgn [39]

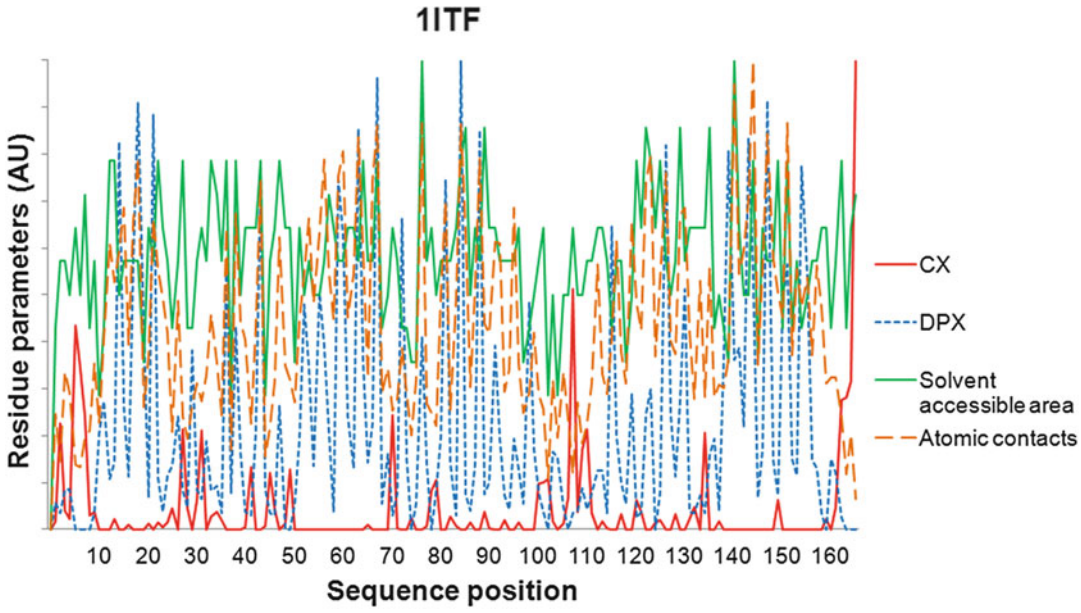


Fig. 6 Example of a combined sequence plot produced by the PCW server. The input PDB file was that of human interferon alpha 2A (PDB: 1itf [37]) The output parameters *cx*, *dpx*, accessible solvent area, and number of atomic contacts are plotted in arbitrary units (scaling the maximum to 1.0). For a better transparency the number of parameters included in one plot can be reduced

already familiar with the basic principles to prepare combined 1D plots in which selected versions *dpx*, *cx*, accessible surface area, and atomic contacts are plotted against the sequence position.

2. The servers return modified PDB files that allow users to visualize the protein structures colored according to the calculated indices. All servers provide online 1D plots created by

GNUPLOT [32], which allow localization of peculiar segments within the query and allow visualization of the 3D structures using JMOL [30]. Users wishing to use more sophisticated visualization programs can download the results in the form of pdb files for 3D visualization and tabulated ASCII files for sequence plots.

3. Online visualization by GNUPLOT [32] and JMOL [30] produce draft quality figures that can be used for documentation purposes and for selecting a parameters, colors, view angles for publication-level figures that will be drawn by high-level visualization programs.
4. As a rule, the servers cannot handle multiple chains or NMR models, i.e., the input is supposed to contain one single chain. An exception is CX which can handle multiple chains, so that residues shielded by the other chain will not be shown as protruding.
5. All the servers are provided with help files that describe the detailed instructions, the theory, the literature citations as well as the instructions for installing the accessory programs wherever necessary.
6. The servers allow users to experiment with the plotting options and to select settings for their final, publication style figures. It is noted that the online version of the JMOL and GNUPLOT program is able to visualize figures in draft quality which is useful primarily for documentation purposes. However, users may need to use dedicated molecular graphics programs to produce high quality, publication-ready figures from the output files downloaded from the servers.

References

1. Sayle RA, Milner-White EJ (1995) RASMOL: biomolecular graphics for all. Trends Biochem Sci 20(9):374
2. Koradi R, Billeter M, Wuthrich K (1996) MOLMOL: a program for display and analysis of macromolecular structures. J Mol Graph 14(1):51–55, 29–32
3. Kaplan W, Littlejohn TG (2001) Swiss-PDB viewer (deep view). Brief Bioinform 2(2): 195–197
4. Schrodinger LLC (2010) The PyMOL molecular graphics system, Version 1.3r1
5. Carugo O, Cemazar M, Zahariev S, Hudaky I, Gaspari Z, Perczel A, Pongor S (2003) Vicinal disulfide turns. Protein Eng 16(9):637–639
6. Carugo O, Pongor S (2001) A normalized root-mean-square distance for comparing protein three-dimensional structures. Protein Sci 10(7):1470–1473
7. Carugo O, Pongor S (2002) The evolution of structural databases. Trends Biotechnol 20(12):498–501
8. Carugo O, Pongor S (2002) Recent progress in protein 3D structure comparison. Curr Protein Pept Sci 3(4):441–449
9. Carugo O, Pongor S (2002) Protein fold similarity estimated by a probabilistic approach based on C(alpha)-C(alpha) distance comparison. J Mol Biol 315(4):887–898
10. Dhir S, Pacurar M, Franklin D, Gaspari Z, Kertesz-Farkas A, Kocsor A, Eisenhaber F, Pongor S (2010) Detecting atypical examples of known domain types by sequence similarity searching: the SBASE domain library approach. Curr Protein Pept Sci 11(7):538–549
11. Gaspari Z, Angyan AF, Dhir S, Franklin D, Perczel A, Pintar A, Pongor S (2010) Probing dynamic protein ensembles with atomic

proximity measures. Curr Protein Pept Sci 11(7):515–522

12. Gaspari Z, Vlahovicek K, Pongor S (2005) Efficient recognition of folds in protein 3D structures by the improved PRIDE algorithm. Bioinformatics 21(15):3322–3323
13. Kertesz-Farkas A, Dhir S, Sonego P, Pacurar M, Netoteia S, Nijveen H, Kuzniar A, Leunissen JA, Kocsor A, Pongor S (2008) Benchmarking protein classification algorithms via supervised cross-validation. J Biochem Biophys Methods 70(6):1215–1223
14. Pintar A, Carugo O, Pongor S (2002) CX, an algorithm that identifies protruding atoms in proteins. Bioinformatics 18(7):980–984
15. Pintar A, Carugo O, Pongor S (2003) Atom depth in protein structure and function. Trends Biochem Sci 28(11):593–597
16. Pintar A, Carugo O, Pongor S (2003) Atom depth as a descriptor of the protein interior. Biophys J 84(4):2553–2561
17. Pintar A, Carugo O, Pongor S (2003) DPX: for the analysis of the protein core. Bioinformatics 19(2):313–314
18. Pongor S, Skerl V, Cserzo M, Hatsagi Z, Simon G, Bevilacqua V (1993) The SBASE domain library: a collection of annotated protein segments. Protein Eng 6(4):391–395
19. Sonego P, Pacurar M, Dhir S, Kertesz-Farkas A, Kocsor A, Gaspari Z, Leunissen JA, Pongor S (2007) A protein classification benchmark collection for machine learning. Nucleic Acids Res 35(Database issue):D232–D236
20. Vlahovicek K, Carugo O, Pongor S (2002) The PRIDE server for protein three-dimensional similarity. J Appl Cryst 35:648–649
21. Vlahovicek K, Pintar A, Parthasarathi L, Carugo O, Pongor S (2005) CX, DPX and PRIDE: WWW servers for the analysis and comparison of protein 3D structures. Nucleic Acids Res 33(Web Server issue):W252–W254
22. Brukner I, Dlakic M, Savic A, Susic S, Pongor S, Suck D (1993) Evidence for opposite groove-directed curvature of GGGCCC and AAAAA sequence elements. Nucleic Acids Res 21(4):1025–1029
23. Brukner I, Sanchez R, Suck D, Pongor S (1995) Trinucleotide models for DNA bending propensity: comparison of models based on DNaseI digestion and nucleosome packaging data. J Biomol Struct Dyn 13(2):309–317
24. Brukner I, Sanchez R, Suck D, Pongor S (1995) Sequence-dependent bending propensity of DNA as revealed by DNase I: parameters for trinucleotides. EMBO J 14(8):1812–1818
25. Gabrielian A, Vlahovicek K, Pongor S (1997) Distribution of sequence-dependent curvature in genomic DNA sequences. FEBS Lett 406(1–2):69–74
26. Munteanu MG, Vlahovicek K, Parthasarathy S, Simon I, Pongor S (1998) Rod models of DNA: sequence-dependent anisotropic elastic modelling of local bending phenomena. Trends Biochem Sci 23(9):341–347
27. Vlahovicek K, Kajan L, Pongor S (2003) DNA analysis servers: plot.it, bend.it, model.it and IS. Nucleic Acids Res 31(13):3686–3687
28. Vlahovicek K, Munteanu MG, Pongor S (1999) Sequence-dependent modelling of local DNA bending phenomena: curvature prediction and vibrational analysis. Genetica 106(1–2):63–73
29. Vlahovicek K, Pongor S (2000) Model.it: building three dimensional DNA models from sequence data. Bioinformatics 16(11):1044–1045
30. Herraez A (2006) Biomolecules in the computer: Jmol to the rescue. Biochem Mol Biol Educ 34(4):255–261
31. Hubbard SJ, Thornton JM (1993) NACCESS. 2.1.1 edn., Department of Biochemistry and Molecular Biology University College, London
32. Williams T, Kelley C (2011) Gnuplot 4.5: an interactive plotting program. http://gnuplot.info
33. Lee B, Richards FM (1971) The interpretation of protein structures: estimation of static accessibility. J Mol Biol 55(3):379–400
34. Connolly ML (1986) Measurement of protein surface shape by solid angles. J Mol Graph 4:3–6
35. Nishikawa K, Ooi T (1986) Radial locations of amino acid residues in a globular protein: correlation with the sequence. J Biochem 100(4):1043–1047
36. Taylor WR, Thornton JM, Turnell WG (1983) An ellipsoidal approximation of protein shape. J Mol Graph 1(2):30–38
37. Klaus W, Gsell B, Labhardt AM, Wipf B, Senn H (1997) The three-dimensional high resolution structure of human interferon alpha-2a determined by heteronuclear NMR spectroscopy in solution. J Mol Biol 274(4):661–675
38. Bocskei Z, Groom CR, Flower DR, Wright CE, Phillips SE, Cavaggioni A, Findlay JB, North AC (1992) Pheromone binding to two rodent urinary proteins revealed by X-ray crystallography. Nature 360(6400):186–188
39. Dyda F, Hickman AB, Jenkins TM, Engelman A, Craigie R, Davies DR (1994) Crystal structure of the catalytic domain of HIV-1 integrase: similarity to other polynucleotidyl transferases. Science 266(5193):1981–1986
40. Richards FM (1974) The interpretation of protein structures: total volume, group volume distributions and packing density. J Mol Biol. 82(1):1–14

INDEX

Yaoqi Zhou et al. (eds.), *Prediction of Protein Secondary Structure*, Methods in Molecular Biology, vol. 1484,
DOI 10.1007/978-1-4939-6406-2, © Springer Science+Business Media New York 2017

G

H

I

K

L

M

N

P

R

S

T

W

X