Bioinformatics, 31(7), 2015, 1138–1140
doi: 10.1093/bioinformatics/btu773
Advance Access Publication Date: 19 November 2014
Applications Note

OXFORD

Structural bioinformatics

The Victor C++ library for protein representation and advanced manipulation

Layla Hirsh[1,2], Damiano Piovesan[1], Manuel Giollo[1,3], Carlo Ferrari[3] and Silvio C. E. Tosatto[1]*

[1]Department of Biomedical Sciences, University of Padua, Viale G. Colombo 3, 35131 Padova, Italy, [2]Department of Engineering, Pontificia Universidad Católica del Perú, San Miguel, 32 Lima, Perú and [3]Department of Information Engineering, University of Padua, Via Gradenigo 6, 35121 Padova, Italy

*To whom correspondence should be addressed.
Associate Editor: Anna Tramontano

Received on August 19, 2014; revised on November 12, 2014; accepted on November 15, 2014

Abstract

Motivation: Protein sequence and structure representation and manipulation require dedicated software libraries to support methods of increasing complexity. Here, we describe the VIrtual Constrution TOol for pRoteins (Victor) C++ library, an open source platform dedicated to enabling inexperienced users to develop advanced tools and gathering contributions from the community. The provided application examples cover statistical energy potentials, profile–profile sequence alignments and *ab initio* loop modeling. Victor was used over the last 15 years in several publications and optimized for efficiency. It is provided as a GitHub repository with source files and unit tests, plus extensive online documentation, including a Wiki with help files and tutorials, examples and Doxygen documentation.

Availability and implementation: The C++ library and online documentation, distributed under a GPL license are available from URL: http://protein.bio.unipd.it/victor/.

Contact: silvio.tosatto@unipd.it

1 Introduction

Structural bioinformatics methods require valid software libraries to represent and manipulate proteins efficiently. A number of widely used tools have been developed over the years to visualize proteins, e.g. Chimera (Huang *et al.*, 2014), Swiss-PdbViewer (Guex *et al.*, 2009), MolIDE (Canutescu and Dunbrack, 2005) and VMD (Humphrey *et al.*, 1996) to name a few. Software libraries to manipulate proteins efficiently provide basic data representation and more advanced functionality with a different focus each. ESBTL (Loriot *et al.*, 2010) is mainly a Protein Data Bank (PDB) file parser. Biskit (Grünberg *et al.*, 2007) additionally provides functionality for analysis of molecular dynamics simulations, while PTools (Saladin *et al.*, 2009) focuses on molecular docking. OpenStructure (Biasini *et al.*, 2010) places more attention on structure visualization and energy calculation. The latter is also supported by MSL (Kulp *et al.*, 2012) and Tinker (Shi *et al.*, 2013), while BALL (Hildebrandt *et al.*, 2010) in addition provides many advanced optimization algorithms. Finally, StrBioLib (Chandonia, 2007) extracts sequence information from the protein structure and can be used as an interface to several available third-party tools.

The critical assessment of techniques for protein structure prediction (CASP) series of experiments (Moult *et al.*, 2014) demonstrates that structure prediction is increasingly becoming an engineering problem, where sophisticated methods have to be combined into extensive pipelines to provide state-of-the-art results (Khoury *et al.*, 2014). This has raised the barrier for entry into the field to a point where little new developments are possible, considering that most software libraries used in CASP are proprietary and not available as open source. Here, we propose the open-source VIrtual Constrution TOol for pRoteins (Victor) C++ library as a way to mitigate this problem. Victor is both an efficiently designed C++ library, able to manipulate protein structures with minimal computing time, and a collection of advanced components for protein sequence and structure manipulation. In particular, Victor

provides three sample applications: profile–profile sequence alignments (Wang and Dunbrack, 2004), statistical potentials (Tosatto, 2005) and loop modelling (Tosatto *et al.*, 2002). Each of these three applications has been extensively described in the literature and is beyond the scope of this article. To the best of our knowledge, neither is available as an open-source C++ library yet. Profile–profile sequence alignments, in particular, have been widely used to improve target-template alignment in CASP (Kryshtafovych *et al.*, 2014). Victor is composed of >60 000 lines of code and still expanding as it is used in the main author's teaching. It was developed in-house over the last 15 years with the contribution of tens of developers and has reached a high level of maturity. Victor is released to provide a platform for contributions from the interested community. It provides extensive online material in the form of a Wiki with help files, tutorials, Doxygen documentation and a list of applications built using Victor can be accessed from the URL: http://protein.bio.unipd.it/victor/. The actual GitHub repository with C++ source files, a precompiled Ubuntu 64-bit version and unit tests are available from URL: https://github.com/BioComputingUP/Victor.

2 Core library

The Victor C++ library currently contains two components for data representation and manipulation in separate directories: tools and Biopool. Tools provide basic manipulation methods, e.g. vector coordinates and file I/O. The core of the library is provided by the Biopool module, which defines all relevant data structures and algorithms to represent protein structures and manipulate them at a higher level of abstraction. The core data structures were carefully developed using design patterns (Gamma *et al.*, 1995), to provide an elegant and simple, yet powerful set of C++ classes. To allow the simple manipulation of protein structure through the more intuitive torsion angles, automating low-level geometric transformations, atom positions are coded both explicitly in 3D coordinates and as a position relative to the previous atom on a graph structure. This ensures consistency in the structure, while allowing the programmer to change the protein conformation rotating a torsion angle with a single line of code. Computational efficiency is guaranteed by updating the corresponding Cartesian coordinates only when necessary. All low-level geometrical transformations remain transparent to the user. Biopool is able to read properly all existing PDB files. Additional tools are also provided, such as protein secondary structure automatic assignment with an *ad hoc* implementation of the original DSSP algorithm (Kabsch and Sander, 1983). Extensive online documentation allows the interested programmer to learn how to manipulate the Biopool data structures.

3 Applications

The Victor library provides three main examples to demonstrate the range of possible applications, which are included as separate subdirectories: Energy, Align and Lobo. Extensive documentation, including detailed tutorials, is provided online to allow users to become familiar with the software and build on existing knowledge. Energy contains everything that is necessary to develop statistical potentials to evaluate protein structures. Two sample implementations of published methods included in the library, FRST (Tosatto, 2005) and TAP (Tosatto and Battistutta, 2007), can serve as a guide to develop additional methods. Both are contained in the Energy subdirectory and functioning code is provided both to generate the statistical potential itself as well as to use it on a PDB structure to calculate the potential energy. The interested user can thus easily develop additional statistical potentials.

The Align directory provides basic sequence alignment algorithms (Tosatto *et al.*, 2006) augmented with secondary structure element (Fontana *et al.*, 2005). Many different profile–profile scoring schemes (Wang and Dunbrack, 2004) are implemented, which have been extensively used in CASP to detect remotely homologous protein sequences. Code is also provided for variable gap penalties with additional terms for sequence to structure fit (Madhusudhan *et al.*, 2006) and advanced weighting schemes such as PSIC (Sunyaev *et al.*, 1999). Alignment parameters have been extensively benchmarked and the default parameters are optimized for performance.

Last but not least, the Lobo directory contains an application of *ab initio* loop modeling using a fast divide and conquer algorithm (Tosatto *et al.*, 2002). This makes extensive use of the functions to construct novel amino acids and manipulate the protein structure locally, providing sample code for more complex structural manipulations. It can easily be extended for *ab initio* structure prediction in combination with statistical potentials as target function.

4 Conclusions

The Victor library is an open source project devoted to the structural bioinformatics community. It provides a unique combination of methods for sequence and structure manipulation. Expansion is ongoing both through in-house development, as it is the basis for several more recent publications [e.g. RING (Martin *et al.*, 2011) and NeEMO (Giollo *et al.*, 2014)], and as part of the author's teaching activities, which include software development projects for students. We hope that the Victor library will contribute towards an easier development of advanced methods for structural bioinformatics.

Acknowledgements

To the Francesco Lovo, Enrico Negri and several students for contributing to the Victor project over the years as well as to members of the BioComputing UP lab for insightful discussions.

Funding

This project was funded by FIRB Futuro in Ricerca grant RBFR08ZSXY and University of Padua grant CPDR123473 to S.T.

Conflict of interest: none declared.

References

Biasini,M. *et al.* (2010) OpenStructure: a flexible software framework for computational structural biology. *Bioinformatics*, **26**, 2626–2628.

Canutescu,A.A. and Dunbrack,R.L. (2005) MollDE: a homology modeling framework you can click with. *Bioinformatics*, **21**, 2914–2916.

Chandonia,J.-M. (2007) StrBioLib: a Java library for development of custom computational structural biology applications. *Bioinformatics*, **23**, 2018–2020.

Fontana,P. *et al.* (2005) The SSEA server for protein secondary structure alignment. *Bioinformatics*, **21**, 393–395.

Gamma,E. *et al.* (1995) *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, USA.

Giollo,M. *et al.* (2014) NeEMO: a method using residue interaction networks to improve prediction of protein stability upon mutation. *BMC Genomics*, **15** (**Suppl 4**), S7.

Grünberg,R. *et al.* (2007) Biskit—a software platform for structural bioinformatics. *Bioinformatics*, **23**, 769–770.

Guex,N. *et al*. (2009) Automated comparative protein structure modeling with SWISS-MODEL and Swiss-PdbViewer: a historical perspective. *Electrophoresis*, **30** (**Suppl 1**), S162–173.

Hildebrandt,A. *et al*. (2010) BALL—biochemical algorithms library 1.3. *BMC Bioinformatics*, **11**, 531.

Huang,C.C. *et al*. (2014) Enhancing UCSF Chimera through web services. *Nucleic Acids Res.*, **42**, W478–W484.

Humphrey,W. *et al*. (1996) VMD: visual molecular dynamics. *J. Mol. Graph.*, **14**, 33–38, 27–28.

Kabsch,W. and Sander,C. (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, **22**, 2577–2637.

Khoury,G.A. *et al*. (2014) WeFold: a coopetition for protein structure prediction. *Proteins*, **82**, 1850–1868.

Kryshtafovych,A. *et al*. (2014) CASP10 results compared to those of previous CASP experiments. *Proteins*, **82** (**Suppl 2**), 164–174.

Kulp,D.W. *et al*. (2012) Structural informatics, modeling, and design with an open-source molecular software Library (MSL). *J. Comput. Chem.*, **33**, 1645–1661.

Loriot,S. *et al*. (2010) ESBTL: efficient PDB parser and data structure for the structural and geometric analysis of biological macromolecules. *Bioinformatics*, **26**, 1127–1128.

Madhusudhan,M.S. *et al*. (2006) Variable gap penalty for protein sequence-structure alignment. *Protein Eng. Des. Sel.*, **19**, 129–133.

Martin,A.J.M. *et al*. (2011) RING: networking interacting residues, evolutionary information and energetics in protein structures. *Bioinformatics*, **27**, 2003–2005.

Moult,J. *et al*. (2014) Critical assessment of methods of protein structure prediction (CASP)—round x. *Proteins*, **82** (**Suppl 2**), 1–6.

Saladin,A. *et al*. (2009) PTools: an opensource molecular docking library. *BMC Struct. Biol.*, **9**, 27.

Shi,Y. *et al*. (2013) The polarizable atomic multipole-based AMOEBA force field for proteins. *J. Chem. Theory Comput.*, **9**, 4046–4063.

Sunyaev,S.R. *et al*. (1999) PSIC: profile extraction from sequence alignments with position-specific counts of independent observations. *Protein Eng.*, **12**, 387–394.

Tosatto,S.C.E. *et al*. (2002) A divide and conquer approach to fast loop modeling. *Protein Eng.*, **15**, 279–286.

Tosatto,S.C.E. *et al*. (2006) Align: a C++ class library and web server for rapid sequence alignment prototyping. *Curr. Drug Discov. Technol.*, **3**, 167–173.

Tosatto,S.C.E. (2005) The victor/FRST function for model quality estimation. *J. Comput. Biol.*, **12**, 1316–1327.

Tosatto,S.C.E. and Battistutta,R. (2007) TAP score: torsion angle propensity normalization applied to local protein structure evaluation. *BMC Bioinformatics*, **8**, 155.

Wang,G. and Dunbrack,R.L. (2004) Scoring profile-to-profile sequence alignments. *Protein Sci.*, **13**, 1612–1626.