PyKnot: a PyMOL tool for the discovery and analysis of knots in proteins

Rhonald C. Lua*

Department of Molecular and Human Genetics, Baylor College of Medicine, Houston, TX 77030, USA

Associate Editor: Burkhard Rost

ABSTRACT

Summary: Understanding the differences between knotted and unknotted protein structures may offer insights into how proteins fold. To characterize the type of knot in a protein, we have developed PyKnot, a plugin that works seamlessly within the PyMOL molecular viewer and gives quick results including the knot's invariants, crossing numbers and simplified knot projections and backbones. PyKnot may be useful to researchers interested in classifying knots in macromolecules and provides tools for students of biology and chemistry with which to learn topology and macromolecular visualization.

Availability: PyMOL is available at http://www.pymol.org. The PyKnot module and tutorial videos are available at http://youtu.be/p95aif6xqcM.

Contact: rhonald.lua@gmail.com

1 INTRODUCTION

Knots are rare in proteins despite their length (Lua and Grosberg, 2006). When knotted proteins do occur, experiments and computational studies suggest that knots have a significant effect on protein stability or folding (Sayre *et al.*, 2011; Sulkowska *et al.*, 2008, 2009; Wagner *et al.*, 2005). Although it is still unclear whether knots are selected in evolution for their utility (Taylor, 2007), in at least one case, that of a ubiquitin hydrolase, the existence of a five-crossing knot (labeled 5_2 in Alexander–Briggs notation) (Lua and Grosberg, 2006; Virnau *et al.*, 2006) is speculated to serve as protection against degradation by the proteasome as ubiquitin hydrolase tries to rescue other proteins from degradation (Virnau *et al.*, 2006).

Computer programs to visualize and analyze knots in proteins are not only of mathematical and recreational interest, but they may also be put into practical use in protein structure prediction. As knots are rare in real proteins, knot-finding programs may be used in protein structure prediction methods to filter predicted models, where knotted structures occur more frequently (Khatib *et al.*, 2006, 2009).

There are a host of computer programs and web services available to analyze knots. KnotPlot (R.G. Scharein, http://knotplot.com/) is a program that visualizes and manipulates mathematical knots.

KNOTS (Kolesov *et al.*, 2007) and pKNOT (Lai *et al.*, 2007) are web servers that survey the Protein Data Bank (PDB) for knotted proteins and also allow users to upload or specify a PDB structure to be analyzed and viewed.

PyKnot takes a different and unique approach by directly integrating knot analysis tools into the desktop application PyMOL, a very popular molecular graphics viewer (The PyMOL Molecular Graphics System, Version 1.3, Schrödinger, LLC.). To our knowledge, PyKnot is the only such plugin for any biomolecular viewer, offering direct access to knot-mathematical techniques such as computation of topological invariants, gauss codes and Reidemeister moves on a knot projection. This combination of techniques was used previously to study knots in lattices, loops and proteins (Lua *et al.*, 2004; Lua and Grosberg, 2006; Moore *et al.*, 2004). PyKnot can also reduce the backbone of the protein to a few segments (not necessarily of equal length), a simplification that enables users to see the knot more clearly and which has its roots in the so-called Koniaris–Mutukhumar–Taylor algorithm (Koniaris and Muthukumar, 1991; Taylor, 2000). The reduced backbone forms the basis of a heuristic approach to estimating the location of the knot's core along the protein sequence.

2 USAGE

2.1 Structure input and viewing

Structure data files can be downloaded from the Protein Data Bank web site, then opened and viewed in PyMOL. Alternatively, PDB structures can be downloaded directly in PyMOL by using the PDB Loader Service plugin that comes with a typical PyMOL installation.

2.2 Knot analysis with PyKnot

PyKnot is written in Python (file named 'PyKnot.py'). PyKnot is easily installed like any other PyMOL plugin by going to the 'Plugin' menu on the PyMOL console window, choosing 'Install' (under 'Manage Plugins' in version 1.3), then locating and selecting the 'PyKnot.py' file.

The analysis of a typical protein in PyMOL using PyKnot is illustrated in a series of youtube videos (http://www.youtube.com/playlist?list=PLA1FF79D72FD5DD0F). To classify the knot in a PDB structure, open the PyKnot window by selecting 'PyKnot' under the 'Plugin' menu, enter the structure name and chain indicator into the first two boxes and click 'Analyze knot'. The knot type is reported as a label next to the structure on the graphics window. The results of the analysis are also printed to the console window.

---

*To whom correspondence should be addressed.

There are several important steps in analyzing the knot in a biomolecule such as a protein:

1. Knot closure: to the C-alpha backbone of a protein, segments connecting the N-terminal to the C-terminal are added such that the backbone forms a closed circuit. The user can choose among three methods: 'outward connection of terminals', which places the new segments away from the center of the protein (this was described as the 'CENTER' method in a previous work (Lua and Grosberg, 2006)); 'direct connection of terminals', which directly connects the N-terminal to the C-terminal; and 'custom extension from terminals', which lets users provide the directions of the N-terminal and C-terminal extensions. Occasionally, a structure may contain structural gaps due to missing residues. PyKnot completes the backbone by creating artificial bonds that bridge across the gap. This choice is equivalent to the default option in KNOTS to 'connect gaps in the structure by straight line' (Kolesov *et al.*, 2007). The presence of gaps and virtual bonds is also reported to the user.

2. Reidemeister moves and gauss codes: in order to compute the knot invariants—the mathematical signatures of the knot type—a representation of the knot in the form of a gauss code must be computed from a projection of the backbone onto any plane (the x–y plane by default). The gauss code encodes the sequence of crossings in the projection (starting from the N-terminal), labeling each crossing with a number and attaching a sign $(+/-)$ indicating the relative directions of the over-crossing and under-crossing segments, plus another label ('a' or 'b') indicating whether the current crossing goes over ('a', above) or under ('b', below) another segment of the knot. An enormous simplification of the gauss code is then obtained by iterative application of a blend of Reidemeister moves (Lua *et al.*, 2004), which typically reduces the number of crossings (i.e. the length of the gauss code) by at least one order of magnitude. In unknotted proteins, the number of crossings often gets reduced to 0 after this step.

3. Knot invariants: PyKnot computes three invariants—the numerical invariant based on the Alexander polynomial $|\Delta(-1)|$ (Vologodskii *et al.*, 1974) and the Vassiliev invariants of Degrees 2 and 3 (v_2 and v_3) (Polyak and Viro, 1994). PyKnot compares the computed knot invariants to the known invariants of the unknot and knots up to seven crossings. If the invariants and the number of crossings match that of a particular knot in this list, then the structure is assigned that knot type. If only the knot invariants match, and the number of crossings do not, then a knot type is still assigned, but a '?' is attached at the end of the label to indicate uncertainty. If the knot invariants do not match the known invariants of the unknot and knots up to seven crossings, then the knot is simply labeled as 'KNOTTED'.

4. Backbone reduction: another important simplification that aids in visualization and analysis of the knot can be achieved by progressively eliminating unnecessary backbone atoms while still preserving the knot type, similar to that described in Khatib *et al.* (2006). The backbone atoms to be eliminated can be selected according to the shape of the triangle formed by three consecutive atoms ('Tight corners first') or they can be selected randomly ('Random'). In the PDB structure of a haloacid dehalogenase (PDB code 3bjx) (Schmidberger *et al.*, 2008) that was previously discovered to contain the most complicated protein knot thus far (a Stevedore's knot with six crossings or (6_1) (Bolinger *et al.*, 2010)), a PyKnot analysis without the 'Reduce backbone' option reports a gauss code with 12 crossings and with knot invariants that match the Stevedore's knot. Upon selecting 'Reduce backbone' and 'Tight corners first', the number of crossings becomes 7. With the 'Reduce backbone' and 'Random' options selected, the number of crossings can be decreased to 6, enabling PyKnot to definitively conclude that 3bjx contains the Stevedore's knot.

5. Knotted core: the location along the sequence of the knot's core is determined heuristically and statistically in PyKnot by generating an ensemble of random reduced backbones. From each reduced backbone, the residues corresponding to the second atom and the second-to-last-atom constitute an estimate of the sequence boundaries of the knot's core. PyKnot then obtains a more refined estimate of the starting (ending) residue of the knot's core from the ensemble by selecting the starting (ending) residue that is nearest the center of mass of the ensemble of starting (ending) residues. This approach, although different from the typical method of successively deleting residues at both ends of the polymer and finding the smallest region that remains knotted (Kolesov *et al.*, 2007; Lai *et al.*, 2007), gives results that are often close to those from KNOTS and pKNOT (the two web servers often differ with each other in their estimates of the knotted core as well). The estimated location of the knotted core should be treated as a guideline, as the heuristic approach may overestimate the size of the knotted core if the backbone wraps around the core.

3 CONCLUSION

PyKnot is an easy-to-use add-on to the intuitive graphical user interface of PyMOL that allows users to quickly assess the knot in a protein. Although it can also perform analysis of knots in other macromolecular backbones (e.g. DNA/RNA backbone defined by phosphate atoms), in the future, we expect PyKnot to offer a more extensive analysis of links as well as knots in and among DNA and other biopolymers.

ACKNOWLEDGEMENTS

REFERENCES

Bolinger,D. *et al.* (2010) A Stevedore's protein knot. *PLoS Comput. Biol.*, **6**, e1000731.
Khatib,F. *et al.* (2009) Pokefind: a novel topological filter for use with protein structure prediction. *Bioinformatics*, **25**, i281–i288.
Khatib,F. *et al.* (2006) Rapid knot detection and application to protein structure prediction. *Bioinformatics*, **22**, e252–e259.

Kolesov,G. *et al*. (2007) Protein knot server: detection of knots in protein structures. *Nucleic Acids Res*., **35** (Web Server issue), W425–W428.

Koniaris,K. and Muthukumar,M. (1991) Knottedness in ring polymers. *Phys. Rev. Lett.*, **66**, 2211–2214.

Lai,Y.L. *et al*. (2007) pKNOT: the protein KNOT web server. *Nucleic Acids Res*., **35** (Web Server issue), W420–W424.

Lua,R.C. *et al*. (2004) Fractal and statistical properties of large compact polymers: a computational study. *Polymer*, **45**, 717–731.

Lua,R.C. and Grosberg,A.Y. (2006) Statistics of knots, geometry of conformations, and evolution of proteins. *PLoS Comput. Biol.*, **2**, e45.

Moore,N.T. *et al*. (2004) Topologically driven swelling of a polymer loop. *Proc. Natl Acad. Sci. USA*, **101**, 13431–13435.

Polyak,M. and Viro,O. (1994) Gauss diagram formulas for Vassiliev invariants. *Int. Math. Res. Notes*, **11**, 445–453.

Sayre,T.C. *et al*. (2011) Protein stabilization in a highly knotted protein polymer. *Protein Eng. Des. Sel.*, **24**, 627–630.

Schmidberger,J.W. *et al*. (2008) The crystal structure of DehI reveals a new alpha-haloacid dehalogenase fold and active-site mechanism. *J. Mol. Biol*., **378**, 284–294.

Sulkowska,J.I. *et al*. (2009) Dodging the crisis of folding proteins with knots. *Proc. Natl Acad. Sci. USA*, **106**, 3119–3124.

Sulkowska,J.I. *et al*. (2008) Stabilizing effect of knots on proteins. *Proc. Natl Acad. Sci. USA*, **105**, 19714–19719.

Taylor,W.R. (2000) A deeply knotted protein structure and how it might fold. *Nature*, **406**, 916–919.

Taylor,W.R. (2007) Protein knots and fold complexity: some new twists. *Comput. Biol. Chem.*, **31**, 151–162.

Virnau,P. *et al*. (2006) Intricate knots in proteins: function and evolution. *PLoS Comput. Biol.*, **2**, e122.

Vologodskii,A. *et al*. (1974) Problem of knots in statistical mechanics of polymer chains. *Sov. Phys. JETP*, **39**, 1059–1063.

Wagner,J.R. *et al*. (2005) A light-sensing knot revealed by the structure of the chromophore-binding domain of phytochrome. *Nature*, **438**, 325–331.