Rknots
Topological analysis of knotted biopolymers with R

Journal Article

Author(s):
Comoglio, Federico; Rinaldi, Maurizio

Publication date:
2012-05

Permanent link:
https://doi.org/10.3929/ethz-b-000048958

Rights / license:
In Copyright - Non-Commercial Use Permitted

Originally published in:
Bioinformatics 28(10), https://doi.org/10.1093/bioinformatics/bts160
A chain is knotted if it does not disentangle after being pulled from Web servers for protein knots analysis (Kolesov et al., 2007). The topological study of knotted biological polymers is an active yet simply to use, generalized computational methods are required. To understand structural properties of knotted polymers, rigorous both material properties and polymer chain dynamics (Koniaris and Muthukumar, 1991). We also provide an extension of the definition of knot is relaxed and transferred to open curves. Biological or synthetic polymers are open chains. In this context, formal introduction to the subject). However, the vast majority of deals with closed structures (see Supplementary Material for an acceptance on March 31, 2012.

Received on December 24, 2011; revised on February 27, 2012; accepted on March 31, 2012.

1 INTRODUCTION

The topological study of knotted biopolymers is an active interdisciplinary field of research. Bolinger et al. (2010) and Marenduzzo et al. (2008) have recently been reported. Savery et al. (2011) and in certain viral capsids there are evidences of mechanisms preventing knotted DNA formation (Burnier et al., 2008). In order to understand structural properties of knotted polymers, rigorous yet simply to use, generalized computational methods are required. Web servers for protein knots analysis (Kolesov et al., 2007) nicely accomplished this for proteins. However, the underlying framework cannot be generalized to cope with more complex structures and it cannot be customized.

In this note we present Rknots, an R package combining a generalized framework for the topological analysis of knotted biopolymers with the benefits of R programming. Different structures can be analyzed with a simple syntax and methods have been implemented accounting for modularity. Rknots requires a standard R installation and depends on bio3d (Frant et al., 2006) available from http://mccammon.ucsd.edu/~bgrant/bio3D/; rgl [Adler,D. and Murdoch,D. (2011) rgl: 3D visualization device system (OpenGL)] and rsympy [Grotendorst,G. et al. (2010) rSymPy: R interface to SymPy computer algebra system. R package version 0.2-1.1]. In the following we will provide an overview of the available methods. A case study on a knotted protein will be used as an example. Additional examples can be found in the package manual and vignette at http://cran.r-project.org/web/packages/Rknots.

2 APPROACH

Proteins can be loaded in .pdb format from the file system or by fetching the Protein Data Bank (PDB; Berman et al., 2000) and they undergo a dedicated preprocessing (see Supplementary Material). Coordinates are then stored in the S4 Knot class (see the package vignette for details) and the following workflow applies afterwards. First, open chains are closed and a knot diagram is obtained through the here proposed principal component analysis projection (PCAP) algorithm (see Supplementary Material). Figure 1B-D illustrate the results of a simulation on 1000 proteins sampled randomly from the PDB in comparison to a set of generic projections or to standard projection (see Supplementary Material for details). Second, the structure is reduced to the minimal set of points topologically equivalent to the original structure, by applying a reduction algorithm. This step removes structural redundancies, speeding up downstream computations. Two structure reduction algorithms, Alexander-Briggs (1924) and MSR (Koniaris and Rinaldi, 2011), are implemented in Rknots. The former has been proposed for the first time in polymers knot theory by Koniaris and Muthukumar (Koniaris and Rinaldi, 2011) and was implemented accounting for modularity. Different structures can be analyzed with a simple syntax and methods have been implemented accounting for modularity. Rknots requires a standard R installation and depends on bio3d (Frant et al., 2006) available from http://mccammon.ucsd.edu/~bgrant/bio3D/; rgl [Adler,D. and Murdoch,D. (2011) rgl: 3D visualization device system (OpenGL)] and rsympy [Grotendorst,G. et al. (2010) rSymPy: R interface to SymPy computer algebra system. R package version 0.2-1.1]. In the following we will provide an overview of the available methods. A case study on a knotted protein will be used as an example. Additional examples can be found in the package manual and vignette at http://cran.r-project.org/web/packages/Rknots.

To whom correspondence should be addressed.

© The Author 2012. Published by Oxford University Press. All rights reserved. For Permissions, please email: journals.permissions@oup.com
The following example illustrates a typical Rknots session for the analysis of a protein of interest. A monomeric left-handed trefoil knotted protein (PDB ID 2k0a) has been selected for this case study. First, the protein is imported with the function `loadProtein`:

```r
library(Rknots)
pdb <- loadProtein(pdbID = "2K0A", cutoff = 7)
```

Second, a `Knot` object is created simply by providing the protein coordinates of the single polypeptide chain to `newKnot`:

```r
chain <- newKnot(pdb$A)
```

Third, the open chain is closed and projected using the `closeAndProject` function. The knot diagram can then be visualized with `plot`:

```r
chain <- closeAndProject(chain)
plot(chain, lwd = 1.5)
```

Then, the knot type is determined by computing a polynomial invariant with the `computeInvariant` function. By setting `invariant = "HOMFLY"`, the HOMFLY polynomial is returned:

```r
computeInvariant(chain, invariant = "HOMFLY")
```

Finally, further information on the knot type can be obtained with the function `getKnotType` (see the package vignette for details).

## 4 CONCLUSION

Rknots is the first package providing generalized tools for the study of knotted biopolymers. The modularity of the package allows integration in custom pipelines. We encourage contributions from other members of the research community.

**Funding:** This work was partially funded by the Italian Ministry of Education, University and Research (MIUR).

**Conflict of Interest:** none declared.

### REFERENCES