Bioinformatics, 31(9), 2015, 1490–1492
doi: 10.1093/bioinformatics/btu849
Advance Access Publication Date: 26 December 2014
Applications Note

OXFORD

Structural bioinformatics

An automatic tool to analyze and cluster macromolecular conformations based on self-organizing maps

Guillaume Bouvier*, Nathan Desdouits, Mathias Ferber, Arnaud Blondel and Michael Nilges*

Institut Pasteur, Unité de Bioinformatique Structurale; CNRS UMR 3528; Département de Biologie Structurale et Chimie; F-75015, Paris, France

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

Received on September 26, 2014; revised on November 25, 2014; accepted on December 21, 2014

Abstract

Motivation: Sampling the conformational space of biological macromolecules generates large sets of data with considerable complexity. Data-mining techniques, such as clustering, can extract meaningful information. Among them, the self-organizing maps (SOMs) algorithm has shown great promise; in particular since its computation time rises only linearly with the size of the data set. Whereas SOMs are generally used with few neurons, we investigate here their behavior with large numbers of neurons.

Results: We present here a python library implementing the full SOM analysis workflow. Large SOMs can readily be applied on heavy data sets. Coupled with visualization tools they have very interesting properties. Descriptors for each conformation of a trajectory are calculated and mapped onto a 3D landscape, the U-matrix, reporting the distance between neighboring neurons. To delineate clusters, we developed the flooding algorithm, which hierarchically identifies local basins of the U-matrix from the global minimum to the maximum.

Availability and implementation: The python implementation of the SOM library is freely available on github: https://github.com/bougui505/SOM.

Contact: michael.nilges@pasteur.fr or guillaume.bouvier@pasteur.fr

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Clustering is an ideal way to reduce huge high-dimensional data sets generated by conformational sampling, facilitating their analysis and visualization. In general, clustering requires prior knowledge, such as the relevant number of clusters. Furthermore, the encoding of the structures in 3D Cartesian atomic coordinates depends strongly on alignment, a problem when studying large conformational transitions such as folding/unfolding. Our novel algorithm to analyze conformational sampling addresses these points: (i) To abolish the dependence on structural alignment, we use the matrix of squared Euclidean distances as conformational descriptor. (ii) We exploit the property of large self-organizing maps (SOMs) to spontaneously and clearly cluster similar conformations in basins, and to separate dissimilar ones by barriers, by projecting the high-dimensional exploration onto 3D landscape maps.

2 Approach

The conformational descriptor is derived from the matrix D of squares of Euclidean distances $d_{i,k}$ between atoms i and k. To speed up the SOM training and reduce memory usage, we use a subset of atoms (for proteins, the n C_α atoms) and the resulting $n \times n$ matrix

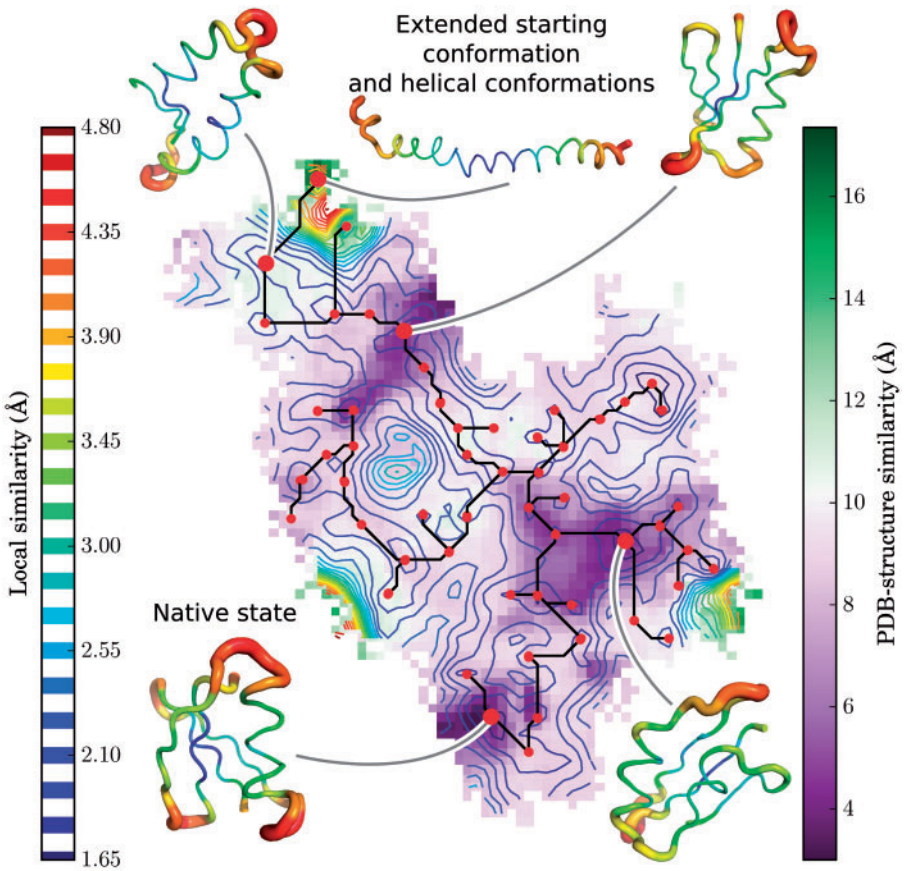

Fig. 1. The square root of the U-matrix, showing the local similarity between structures, is plotted as a contour plot. The RMSD from the native structure is shown in the prune-green heat-map. The U-matrix minima are shown with red dots and their minimum spanning linking tree with black lines. Some conformation sets associated with local minima of interest are depicted in putty cartoons. The root mean square fluctuation (RMSF) of the backbone is represented by the width of the main chain and the blue-green-red color code

D is condensed by computation of the eigenvectors of the covariance matrix, C:

$$C_{i,j} = \frac{1}{n}\sum_{k=1}^{n}\sum_{l=1}^{n}(d_{i,k}^2 - \overline{d_i}^2)(d_{l,j}^2 - \overline{d_j}^2) \quad (1)$$

whose eigenvalues, N_i, vanish for $i > 4$ (Kloczkowski *et al.*, 2009). $\overline{d_i}^2$ is the average of $d_{i,k}^2$ on all the atoms k. Then, we obtain our conformational descriptors by projection of D on N_i: $P_i = D \cdot N_i$, $i \in [1, \ldots, 4]$.

A toroidal map is used to make all neurons equivalent and avoid border effects. Each neuron is a $4n$ vector, n being the number of descriptor atoms, initialized with a uniform distribution covering the range of descriptor values. With our examples, a size of 50×50 neurons resulted in densities of 10 to 20 conformers per neuron, which proved appropriate for visualization of the U-matrix and analysis of the data. For other systems, the size of the map may need to be adjusted to obtain a similar density. Training proceeds in two phases, taking each input vector at least once per phase (by default once in the first phase and twice in second) to identify the closest neuron and update it and its neighborhood. The learning rate decreases exponentially from 0.50 to 0.25 in the first phase and from 0.25 to 0 for the second. The neighborhood function is a Gaussian, whose radius exponentially decreases from 6.25 (a quarter of the map toroid radius) to 3.00 in the first phase, and from 4.00 to 1.00, for the second. These parameters ensured convergence for various datasets of different sizes issued from molecular dynamics or other sampling algorithms (Bouvier *et al.*, 2014; Miri *et al.*, 2014; Nivaskumar *et al.*, 2014; Spill *et al.*, 2013).

SOMs are best visualized by the U-matrix (Unified distance matrix), filled with U-values, the mean distance of each neuron and its eight neighbors. Since our descriptor is in Å^2 units, we calculate the square root of the U-matrix to display results in Å units.

Since our SOMs are toroidal, clusters can straddle the matrix border, making visual interpretation difficult. To overcome this, we use the periodicity to position the U-matrix basins in an aggregated way with the largest barriers outside. This is done by 'flooding', inspired by the watershed algorithm (Meyer and Beucher, 1990). Flooding starts from the global minimum of the U-matrix. Periodic boundaries are applied to define neighbors. The water level rises continuously until the whole map is covered, except when a new basin is found. Then, the minimum of the new basin is identified by steepest descent and the flood continues from that point.

3 Implementation

The package is implemented in python and requires a PDB file and a trajectory in CHARMM DCD format as input. It includes the `makeVectorsFromdcd.py` utility to generate descriptors, the `SOM.py` library to compute the SOM and the U-Matrix, and the `SOMclust.py` library to perform flooding. A detailed

how-to is accessible: http://nbviewer.ipython.org/gist/bougui505/9955459.

4 Results

The SOM algorithm was applied to the analysis of 15 μs molecular dynamics at 330 K of a simplified sequence of a 56-residue α/β subdomain of the protein G (Guarnera *et al.*, 2009) (Fig. 1). This 750 000-frame trajectory presents multiple folding and unfolding events.

The map organizes the large diversity of explored conformations, from extended forms to the native state. The U-matrix depicts the covered conformational space and its topology. The minimum spanning tree between local minima reveals possible transition paths between conformational basins.

We found 53 conformational basins and corresponding minima. The transition $(i \rightarrow j)$ probability flux for each pair of basins (i, j) can be computed as explained in Guarnera *et al.* (2009). This Markov model provides the distribution of the first passage times in the folded basin. The distribution we obtain is similar to the one found by Guarnera *et al.* (2009) and can be fitted by a single exponential in a similar way, with a mean folding time of 137 ns, compared with 163 ns (see Fig. S1 in Supplementary Materials).

5 Conclusion

The package presented here exploits the ability of SOMs to spontaneously cluster macromolecular conformations. It provides a powerful tool for the analysis and visualization of the sampled conformational space even when they cover extremely different areas. It can also delineate more subtle conformational changes such as catalytic mechanisms of an enzyme (Bouvier *et al.*, 2014). Furthermore, the method can be applied to sparser sampling as in Nivaskumar *et al.* (2014) with only 3900 conformations. The SOM algorithm is a non-linear dimensionality reduction technique, in contrast to linear approaches such as principal component analysis. This is of major importance for the study of highly non-linear processes such as folding/unfolding. The SOM computation effort scales linearly with data size and this enables us to cluster even massive numbers of conformations. This is very important given the ever increasing length of simulations. It is largely automatic and has only very few parameters, which we have optimized for this type of application. The good agreement with the previous study by Guarnera *et al.* (2009) on the same trajectory is highly satisfying, in particular considering the fact that our method requires no interventions. The SOM library is very flexible and the conformational descriptors can be tailored to different cases of conformational clustering and analysis.

Funding

This work was funded by the European Union (FP7-IDEAS- ERC 294809 to M.N.). N.D. is supported by an AXA Research Fund doctoral fellowship.

Conflict of Interest: none declared.

References

Bouvier,G. *et al.* (2014) Functional motions modulating vana ligand binding unraveled by self-organizing maps. *J. Chem. Inf. Model.*, **54**, 289–301.

Guarnera,E. *et al.* (2009) How does a simplified-sequence protein fold? *Biophys. J.*, **97**, 1737–1746.

Kloczkowski,A. *et al.* (2009) Distance matrix-based approach to protein structure prediction. *J. Struct. Funct. Genomics,* **10**, 67–81.

Meyer,F. and Beucher,S. (1990) Morphological segmentation. *J. Visual Commun. Image Representation,* **1**, 21–46.

Miri,L. *et al.* (2014). Stabilization of the integrase-dna complex by mg2 + ions and prediction of key residues for binding hiv-1 integrase inhibitors. *Proteins,* **82**, 466–478.

Nivaskumar,M. *et al.* (2014). Distinct docking and stabilization steps of the pseudopilus conformational transition path suggest rotational assembly of type iv pilus-like fibers. *Structure,* **22**, 685–696.

Spill,Y.G. *et al.* (2013). A convective replica-exchange method for sampling new energy basins. *J. Comput. Chem.,* **34**, 132–140.