Kontopoulos et al. BMC Structural Biology (2016) 16:4
DOI 10.1186/s12900-016-0055-7

BMC Structural Biology

SOFTWARE Open Access

Structuprint: a scalable and extensible tool for two-dimensional representation of protein surfaces

Dimitrios Georgios Kontopoulos[1*], Dimitrios Vlachakis[2*], Georgia Tsiliki[3] and Sofia Kossida[4]

Abstract

Background: The term 'molecular cartography' encompasses a family of computational methods for two-dimensional transformation of protein structures and analysis of their physicochemical properties. The underlying algorithms comprise multiple manual steps, whereas the few existing implementations typically restrict the user to a very limited set of molecular descriptors.

Results: We present Structuprint, a free standalone software that fully automates the rendering of protein surface maps, given - at the very least - a directory with a PDB file and an amino acid property. The tool comes with a default database of 328 descriptors, which can be extended or substituted by user-provided ones. The core algorithm comprises the generation of a mould of the protein surface, which is subsequently converted to a sphere and mapped to two dimensions, using the Miller cylindrical projection. Structuprint is partly optimized for multicore computers, making the rendering of animations of entire molecular dynamics simulations feasible.

Conclusions: Structuprint is an efficient application, implementing a molecular cartography algorithm for protein surfaces. According to the results of a benchmark, its memory requirements and execution time are reasonable, allowing it to run even on low-end personal computers. We believe that it will be of use - primarily but not exclusively - to structural biologists and computational biochemists.

Keywords: Molecular cartography, Protein surfaces, Visualization, Surface comparison, Structural biology

Background

Over the last two decades, the growth rate of the Protein Data Bank has been exponential. As structural data for biomolecules are increasingly made available, the study of homologous proteins can be performed not only at the level of sequence, but also at the level of three-dimensional structure. This has led to the development of numerous sophisticated methods, concerning, among others, the analysis of structural evolution [1] and the structure-based design of new drugs [2].

For the comparison of protein surfaces in particular, a family of methods is based on the reduction of the dimensionality of the system. The concept of projecting a three-dimensional protein structure to two dimensions was first introduced by Fanning et al. under the term 'molecular cartography' [3]. They presented this notion as a novel method for studying the entire surface of a protein, emphasizing on the topography of antigenic sites. It involved conversion of the protein structure into a triaxial ellipsoid, followed by its transformation into a graticule (a latitude/longitude grid). Pawłowski and Godzik later expanded on this approach by annotating protein surface maps according to the physicochemical properties of the exposed residues (e.g., charge or hydrophobicity), as a means to compare evolutionarily related proteins [4].

Even though a number of modifications to the aforementioned methodologies for two-dimensional protein representation have been proposed [5–7], molecular cartography has not found much use in the literature. This may be partly due to the significant amount of effort that is required to manually convert the atomic coordinates of a PDB file first into a spherical structure and then into a

* Correspondence: d.kontopoulos13@imperial.ac.uk; dvlachakis@bioacademy.gr
[1]Department of Life Sciences, Imperial College London, Silwood Park Campus, Ascot, UK
[2]Bioinformatics & Medical Informatics Team, Biomedical Research Foundation, Academy of Athens, Athens, Greece
Full list of author information is available at the end of the article

© 2016 Kontopoulos et al. **Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The Creative Commons Public Domain Dedication waiver (http://creativecommons.org/publicdomain/zero/1.0/) applies to the data made available in this article, unless otherwise stated.

map. Visualizing the distribution of a particular physicochemical property on the surface further increases the complexity and the overall approach becomes increasingly tedious. A few applications that implement molecular cartography algorithms are available (SURF'S UP! [8], PST [9], Udock [10]), but the range of supported physicochemical descriptors for visualization is typically limited to charge and hydrophobicity. Integrating other predictors is either unfeasible or not straightforward for the end user, creating an obstacle for specialized analyses. Moreover, an application that harnesses the power of multiprocessor systems to simultaneously render multiple protein surface maps is not to this day available. This would be very useful, for example, when visualizing entire molecular dynamics simulations or comparing the members of a large protein family.

To fill these gaps, we introduce Structuprint, a new tool for visualization of protein surfaces in two dimensions. Its name is a combination of the terms 'structure' and 'fingerprint', alluding to the fingerprint-like figures that it generates (see Fig. 1 for an example). Structuprint can produce single 2D maps starting from a PDB file, or GIF animations from multiple files. It is designed with a focus on scalability and extensibility. The tool can utilize multiple CPU cores on GNU/Linux and OS X machines and can easily incorporate any physicochemical predictors provided by the user, other than those in its own default set. The following sections describe the design choices behind its algorithm, present the results from a benchmark and show three characteristic examples of use.

Implementation

Amino acid properties database

Values for 328 properties/descriptors were calculated for the 20 common amino acids with MOE 2010.10 [11] and were stored within an SQLite database. In particular, the database contains 11 categories of descriptors: i) 33 adjacency and distance matrix descriptors [12–16] (e.g., Balaban's connectivity topological index [14]); ii) 41 atom/bond count descriptors [17, 18] (e.g., the number of double bonds); iii) 18 conformation dependent charge descriptors [19] (e.g., the water accessible surface area of polar atoms); iv) the 16 Kier and Hall connectivity and kappa shape indices [20, 21] (e.g., the Zagreb index); v) 21 MOPAC descriptors [22] (e.g., the ionization potential); vi) 48 partial charge descriptors (e.g., the total positive partial charge); vii) 12 pharmacophore feature descriptors (e.g., the number of hydrophobic atoms);

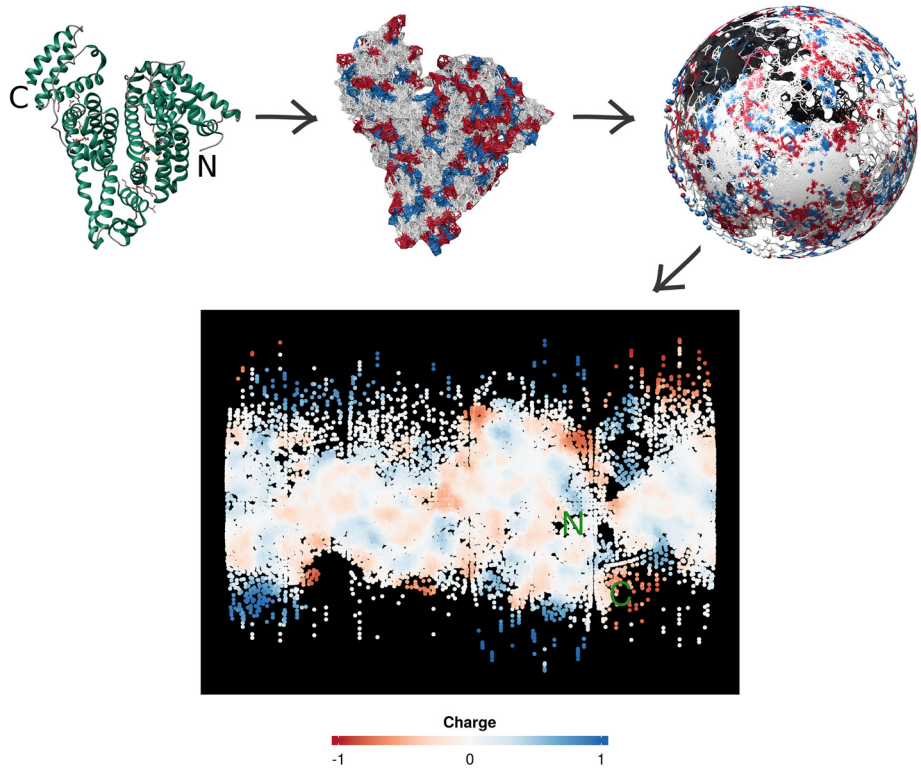

Fig. 1 The main steps of the algorithm executed by Structuprint. Here, a mould of the surface of the 3D structure of the leporine serum albumin ([PDB: 4F5V]) is first generated. The property values (e.g., charge) of the amino acids below the mould are retained. Then, the dummy atoms consisting the mould are mapped onto a sphere. Finally, the sphere is projected onto a map using the Miller cylindrical transformation and a smoothing of the property values is performed. The elements of the upper half of the figure were rendered with UCSF Chimera

viii) 11 potential energy descriptors (e.g., the solvation energy); ix) 16 physical properties [18, 23–27] (e.g., the molecular weight); x) 18 subdivided surface areas; xi) 94 surface area, volume, and shape descriptors (e.g., globularity). A detailed explanation of each descriptor is provided in the properties codebook which accompanies the tool. By drawing values from this database, Structuprint can visualize the distribution of a property across protein surfaces. Users can extend it by adding measurements for more chemical components or provide their own custom SQLite database in order to incorporate novel descriptors.

Algorithm

Generation of a mould of the surface of a protein

The main steps of the algorithm implemented by Structuprint are shown in Fig. 1. The tool first produces a mould of the protein structure's surface in two steps. The structure is initially placed within a 3D grid with cell dimensions of 1 × 1 × 1 Å. Then, one dummy atom is inserted in each empty grid cell that neighbours a single protein atom. This process was previously described by Vlachakis et al. [28] and is extended here, with dummy atoms being assigned the identity of the amino acid to which their neighbouring protein atom belongs. This results to a quite accurate approximation of the underlying protein surface at the level of residue atoms.

Transformation of the mould into a sphere

The next step involves the conversion of the dummy atoms mould to a sphere. To this end, the algorithm calculates the coordinates of the centre of mass of the mould **c** - i.e., the average position of all atoms -, and the maximum distance of any atom $\mathbf{v}_i$ from the centre of mass (*radius*):

$$\mathbf{c} = (x_c,\ y_c,\ z_c) = \left(\frac{\sum_{i=1}^{n} x_i}{n},\ \frac{\sum_{i=1}^{n} y_i}{n},\ \frac{\sum_{i=1}^{n} z_i}{n}\right) \quad (1)$$

$$radius = \max_{1 \le i \le n} \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2 + (z_i - z_c)^2} \quad (2)$$

The coordinates of each atom are normalized with respect to the centre of mass:

$$\mathbf{v}_i' = \left(x_i',\ y_i',\ z_i'\right) = (x_i - x_c,\ y_i - y_c,\ z_i - z_c) \quad (3)$$

Then, to transfer the dummy atoms onto the surface of a sphere, each vector $\mathbf{v}_i'$ is scaled to a length equal to the *radius*:

$$\mathbf{w}_i = \left(x_i'',\ y_i'',\ z_i''\right) = \frac{radius}{\sqrt{x_i'2 + y_i'2 + z_i'2 \cdot \mathbf{v}_i'}} \quad (4)$$

Projection of the sphere onto a map

The Cartesian coordinates of each $\mathbf{w}_i$ are converted to latitude/longitude values (in units of radians) using the following set of equations:

$$latitude_i = \tan^{-1} \frac{z_i''}{\sqrt{x_i''2 + y_i''2}} longitude_i = \tan^{-1} \frac{y_i''}{x_i''} \quad (5)$$

For the two-dimensional projection, several techniques were initially tested (e.g., the sinusoidal projection [29] and the Hammer projection [29, 30]), before deciding on the Miller cylindrical projection [29, 31]:

$$\mathbf{m}_i = \left(longitude_i,\ \frac{5}{4} \cdot \ln\left[\tan\left(\frac{\pi}{4} + \frac{2}{5} \cdot latitude_i\right)\right]\right) \quad (6)$$

This projection was selected on the basis of its simplicity and ease of understanding. It is one of the most popular projections in cartography, as it can depict the entirety of the sphere, including the poles. Latitude and longitude lines are parallel and straight. Projection-induced distortion is zero at the equator, increases gradually towards higher latitudes, and becomes maximal at the poles. This leads to significant overestimation of the distance among atoms at the upper and lower parts of the figure (Fig. 1), similarly to the areal exaggeration of Greenland and Antarctica. Nevertheless, the Miller cylindrical projection introduces less polar distortion than the Mercator projection, on which it is based.

Map smoothing

The previous step resulted in a map of the protein surface with data points coloured by a property of choice. However, this 'primary' map is not suitable for detecting areas with an overall concentration of atoms with high or low property values, which is one of the main benefits of this cartographic approach. For instance, a small area with both negatively and positively charged residue atoms would not appear as almost neutrally charged, but as a tiny dipole. To prevent the appearance of small 'hot spots' and redistribute the property values among neighbouring data points, the algorithm includes a smoothing step. The map is iteratively divided in grid squares of varying dimensions, from 0.001° × 0.001° to 0.5° × 0.5°, with a step increase of 0.001°. In each iteration of this

process, grid cells are assigned the average value of all data points within them. Finally, the value of every data point is defined as the average value of its corresponding grid cell across all iterations. This smoothing method ensures that areas with pronounced accumulation of high or low values are easily discernible from those with a mixed population.

User interfaces

The default interface of Structuprint is a cross-platform, command-line interface (CLI). It consists of two executables: structuprint_frame and structuprint. The structuprint_frame executable produces a TIFF figure from a single input PDB file, using the R package ggplot2 [32] for plotting. The structuprint executable is responsible for processing multiple superimposed PDB files - either serially or in a parallel manner -, generating a TIFF figure per input file and a final GIF animation, rendered with the Imager Perl module [33]. Most parameters of the underlying algorithms can be modified by the user, such as the delay between animation frames, the background colour, and the appearance of ID numbers on final figures. A full descriptive list of the available parameters for both executables can be found in Structuprint's manual, distributed along with the application and also available from its website.

Other than the CLI, Structuprint also comes with a Graphical User Interface (GUI), available by default only on GNU/Linux systems. The GUI is built with the Gtk2 toolkit and offers a user-friendly interface to all the command line arguments and options. As an example of its capabilities, in Fig. 2 Structuprint's GUI is producing an animation on a multiprocessor machine using 30 cores.

Parallelism

On Unix-like systems (e.g., GNU/Linux, OS X), Structuprint supports task parallelism when generating animations. Using the Parallel::ForkManager Perl module [34], Structuprint can take advantage of multiple CPU cores by assigning each input PDB file to a different processor. The simultaneous rendering of multiple individual frames considerably reduces the total execution time, allowing for visualization of entire molecular dynamics simulations within a reasonable time frame.

Results and discussion

Benchmark

To understand how execution time and memory consumption scale with the number of atoms in an input PDB file, we ran Structuprint against 700 randomly selected structures from the Protein Data Bank (Additional file 1). For simplification purposes, multi-model PDB entries were excluded, as a large proportion of the atoms would overlap in 3D space, being essentially indistinguishable. The benchmark was performed on a GNU/Linux system with an Intel Xeon E5-1650 v2 CPU at 3.50 GHz and 31.4 GB of memory. Structuprint was launched 10 times per PDB file and the execution time was measured as the median time for completion. Memory usage was measured similarly. We then performed linear regressions using execution time and memory consumption as dependent variables and number of atoms as the independent variable. In both regressions, we applied a Box-Cox transformation [35] to the dependent variable to ensure that the residuals were normally distributed. The final fitted models are shown in Fig. 3. Execution time increases linearly with the number of atoms, whereas

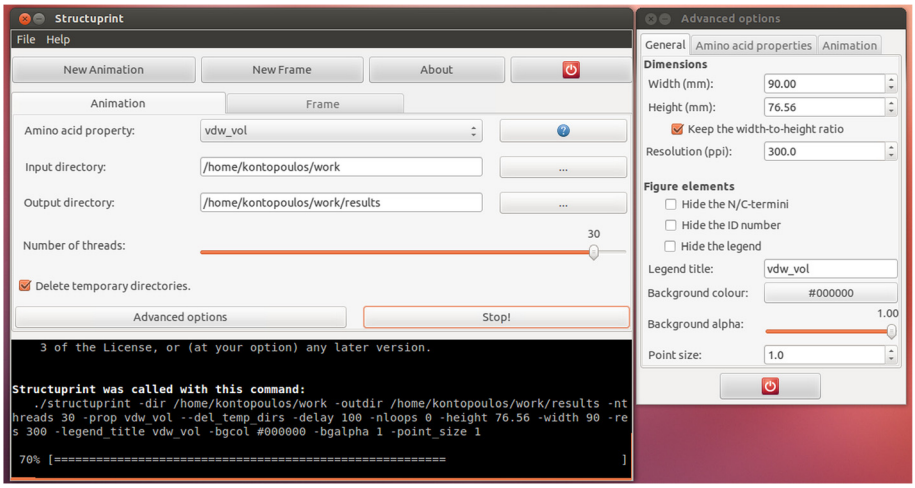

Fig. 2 Structuprint's Graphical User Interface. The main window is split between two tabs for preparation of 1) animations and 2) single static maps. The default parameters of the algorithm can be modified using the 'Advanced options' popup window. When Structuprint is rendering a figure, its progress is shown in a temporary terminal

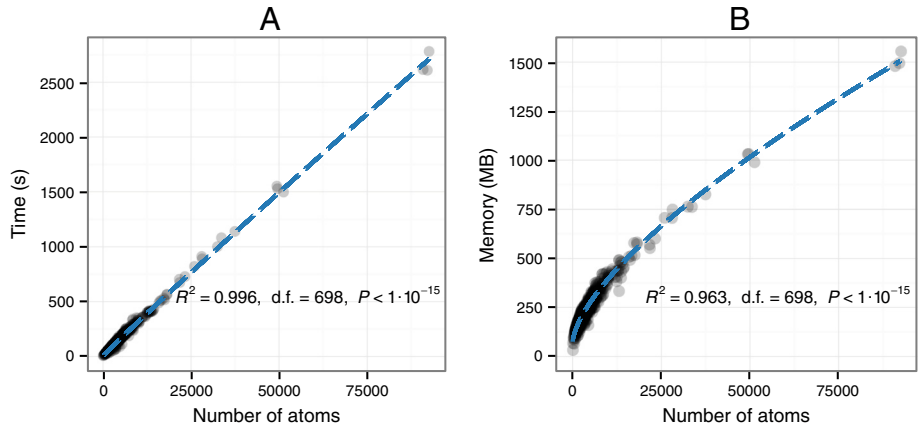

Fig. 3 Execution time (**a**) and memory consumption (**b**) of Structuprint, as a function of the atom count (n). The runtime complexity is O(n), whereas the memory complexity is $O(\sqrt{n})$. The uneven distribution of atom counts reflects the composition of the Protein Data Bank. As of March 2015, ~99 % of entries in the PDB had an atom count of 61,000 or less, with the overall mean being 9,006 atoms

memory consumption only increases with the square root of the atom count. For example, on the aforementioned system it took 88 seconds and 211 MB of RAM to generate a Structuprint figure for a relatively small protein with 2,461 atoms ([PDB:1YLP]).

Examples of usage

To illustrate the utility of this tool, we present three different examples of usage in this section. Two-dimensional visualization with Structuprint enhances the representation of protein surfaces and facilitates the interpretation of the results in all three cases.

Visualization of molecular dynamics simulations

A seldom explored application of molecular cartography involves the generation of 2D animations from a series of PDB files. Here, we visualized a portion of a folding

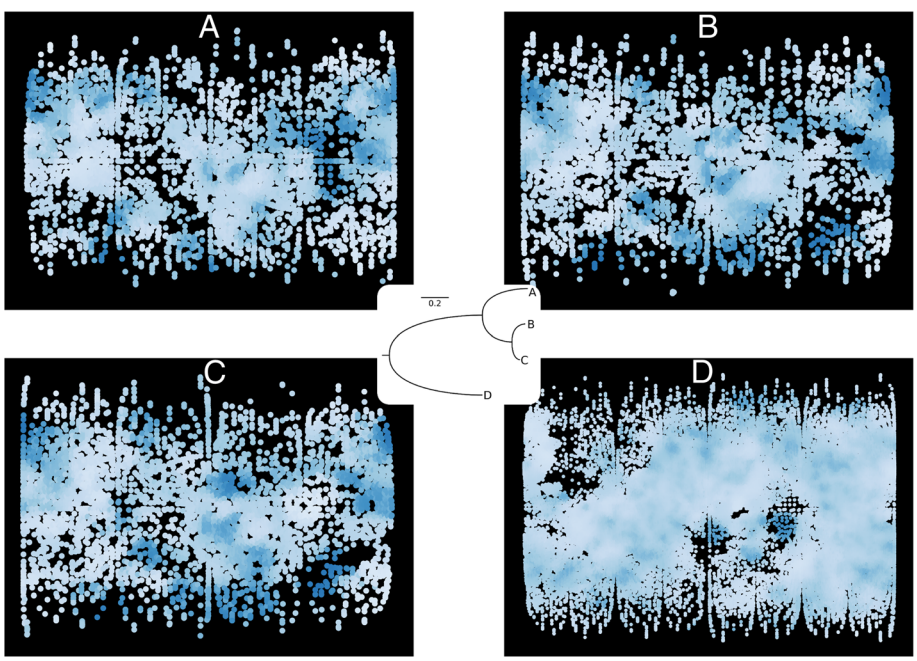

Fig. 4 Evolution of protein surfaces, as represented via Structuprint figures. **a**–**c**: Plastocyanin orthologs from *Spinacia oleracea*, *Ulva pertusa*, and *Ulva prolifera*, respectively. **d** Chloroplastic fructose 1,6-biphosphatase from *Spinacia oleracea*. The colour depth denotes the FASA_H value across each map, with darker areas having higher values of the descriptor. Despite the obvious conservation of surface shape and hydrophobicity, 2D maps can distinguish even slight differences among evolutionarily related proteins. Inset: The maximum likelihood phylogenetic tree of the proteins in panels **a**–**d**

simulation of a variant of the chicken villin headpiece subdomain (HP-35 NleNle) from the Folding@Home project [36]. The part of the input simulation was 50 ps long, with one frame being extracted every 0.25 ps. Each frame was structurally superimposed to the previous one with UCSF Chimera's MatchMaker tool [37]. Then, two separate animations were produced: one of the simulation frames in ribbon representation and one of the corresponding 2D maps, with the topological polar surface area - a measure of polarity - as the property of choice. For comparison purposes, these two animations are jointly shown in Additional file 2. This approach simplifies the detection of conformational changes during the course of the simulation, along with fluctuations in the distribution of physicochemical variables.

Depiction of surface conservation

The evolution of protein surfaces and the conservation - or lack - thereof is another domain in which Structuprint can be applied. As an example, we performed a brief phylogenetic analysis of three orthologs of plastocyanin - a protein involved in electron transfer in photosynthesis [38] - for which crystallographic structures were available. The amino acid sequences of spinach plastocyanin (*Spinacia oleracea* [Swiss-Prot:P00289]) and those of two green algal species (*Ulva pertusa* [Swiss-Prot:P56274], *Ulva prolifera* [Swiss-Prot:P07465]) were retrieved from the UniProt database, along with the sequence of the spinach chloroplastic fructose 1,6-biphosphatase ([Swiss-Prot:P22418]) that would be later used as an outgroup. The sequences were aligned with ProbCons 1.12 [39] and the best model of amino acid substitution was determined with RAxML 8.1.16 [40]. Ten maximum likelihood trees were then inferred with RAxML using the biphosphatase as the outgroup sequence, and the best scoring tree was selected. Next, 2D protein surface maps of the corresponding 3D structures ([PDB:1AG6, 1IUZ, 7PCY, 1SPI]) were produced with Structuprint, after performing a structural superposition. For this example we used a more complex descriptor, FASA_H:

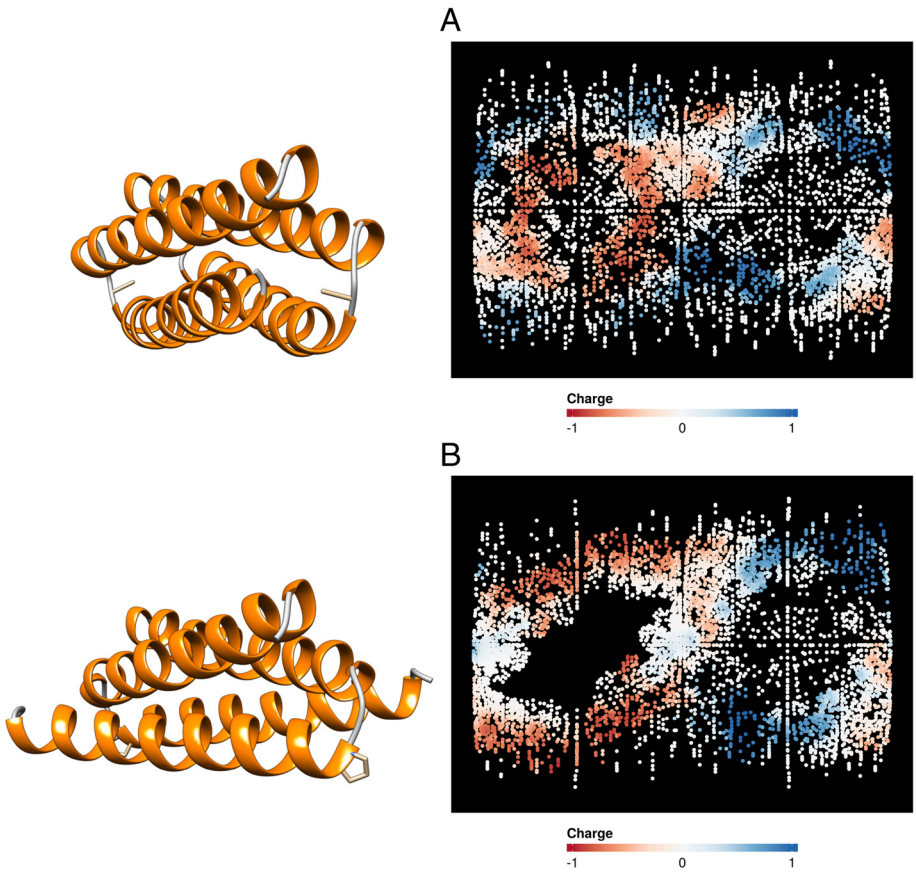

Fig. 5 Three- and two-dimensional depiction of the native Rop structure (**a**) and the A31P mutant (**b**). In the 3D representation, the amino acid side chain of the 31st residue - in the turn region - is shown in stick style. Positively charged residues are shown with blue colour, negatively charged ones with red, and non-charged residues with white. With the 2D representation generated by Structuprint, large differences can be observed not only in the shape of the surface, but also in the location of exposed negatively charged residues

$$FASA_H = \frac{water\ accessible\ surface\ area\ of\ hydrophobic\ atoms}{water\ accessible\ surface\ area\ of\ all\ atoms} \quad (7)$$

The results are shown in Fig. 4. There is significant conservation of both surface structure and hydrophobicity patterns among all three species, with the algal orthologs (Fig. 4b, c) exhibiting greater similarity, as expected. Finally, the representation of the chloroplastic fructose 1,6-biphosphatase (Fig. 4d) is vastly different from the others, highlighting the long sequence distance among these proteins.

Comparison of conformational changes, e.g., due to mutations

A third proposed application of Structuprint involves visually contrasting protein surfaces before and after events such as mutations, ligand binding, pH or temperature alterations. We exemplify this case using a mutant of Rop, a small regulatory protein from *Escherichia coli* with a native tertiary structure of a homodimeric four-helix bundle. The native structure has been shown to be disrupted by a single amino acid substitution (Ala31 → Pro) in the turn region [41]. To show the consequences of this mutation, we generated Structuprint maps of the wild type protein ([PDB:1ROP]) and the A31P mutant ([PDB:1B6Q]) after superposition. Figure 5 illustrates the mutation-induced conformation change, comprising different surface shape and grouping of negatively charged residues.

Conclusions

We have developed a user-friendly application for two-dimensional visualization of protein surfaces, optionally supporting multicore processing and user-provided physicochemical descriptors. Structuprint provides an alternative view of molecular surfaces, which - as shown in the previous section - could be of great use to a variety of researchers, including biochemists, structural biologists, and biophysicists.

Availability and requirements

Project name: Structuprint

Project home page: http://dgkontopoulos.github.io/Structuprint/

Operating systems: Prebuilt packages and installers are available for GNU/Linux distributions (Ubuntu 14.04, Debian 8, Fedora 22, CentOS 7, openSUSE 13.2), Windows, and OS X. For all other operating systems, installation from the source code is required. The GUI is available by default only for GNU/Linux systems.

Programming languages: Perl 5, R

License: GNU GPLv3+

Any restrictions to use by non-academics: None

Availability of data and materials

The datasets supporting the conclusions of this article are included within the article and its additional files.

Additional files

Additional file 1: Table of PDB entries used in the benchmark. Accession codes and atom counts of 700 random, non-multimodel PDB entries that were included in the benchmark. (CSV 12 kb)

Additional file 2: Conventional and molecular cartographic visualizations of a molecular dynamics simulation of the chicken villin headpiece subdomain (HP-35 NleNle). Comparison between animations produced with conventional rendering methods (UCSF Chimera), and with 2D maps generated by Structuprint. The right half shows the movement of exposed amino acids with high topological polar surface area values (blue) during the course of the simulation. (GIF 4858 kb)

Abbreviations

2D: two-dimensional; 3D: three-dimensional; A31P: ala31 → pro mutant; CLI: command-line interface; CPU: central processing unit; FASA_H: fractional water accessible surface area of hydrophobic atoms over all atoms; GB: gigabyte; GUI: graphical user interface; HP-35 NleNle: villin headpiece subdomain double norleucine mutant (Lys24Nle/Lys29Nle); MB: megabyte; MOE: molecular operating environment; MOPAC: molecular orbital package; PDB: protein data bank; RAM: random-access memory; RAxML: randomized axelerated maximum likelihood.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

DGK contributed to the conception of the software, designed, developed and packaged the software, ran benchmarks, prepared the figures, and drafted the manuscript. DV, GT, and SK conceived of and designed the software, reviewed and revised the draft. All authors read and approved the final manuscript.

Acknowledgements

The authors express their gratitude to two anonymous reviewers for helpful comments, and to all researchers who made their data publicly available on the Protein Data Bank, the UniProt database, or on Simtk.org. No funding was received for this project.

Author details

[1]Department of Life Sciences, Imperial College London, Silwood Park Campus, Ascot, UK. [2]Bioinformatics & Medical Informatics Team, Biomedical Research Foundation, Academy of Athens, Athens, Greece. [3]School of Chemical Engineering, National Technical University of Athens, Athens, Greece. [4]IMGT®, The International ImMunoGeneTics Information System®, Université de Montpellier, Laboratoire d'ImmunoGénétique Moléculaire LIGM, UPR CNRS 1142, Institut de Génétique Humaine, Montpellier, France.

Received: 10 November 2015 Accepted: 2 February 2016
Published online: 24 February 2016

References

1. Orengo CA, Thornton JM. Protein families and their evolution - a structural perspective. Annu Rev Biochem. 2005;74:867–900. doi:10.1146/annurev.biochem.74.082803.133029.
2. Cheng T, Li Q, Zhou Z, Wang Y, Bryant SH. Structure-based virtual screening for drug discovery: a problem-centric review. AAPS J. 2012;14(1):133–41. doi:10.1208/s12248-012-9322-0.
3. Fanning DW, Smith JA, Rose GD. Molecular cartography of globular proteins with application to antigenic sites. Biopolymers. 1986;25(5):863–83. doi:10.1002/bip.360250509.
4. Pawłowski K, Godzik A. Surface map comparison: studying function diversity of homologous proteins. J Mol Biol. 2001;309(3):793–806. doi:10.1006/jmbi.2001.4630.

5. Chirgadze Y, Kurochkina N, Nikonov S. Molecular cartography of proteins: surface relief analysis of the calf eye lens protein gamma-crystalin. Protein Eng. 1989;3(2):105–10. doi:10.1093/protein/3.2.105.
6. Badel-Chagnon A, Nessi J, Buffat L, Hazout S. "Iso-depth contour map" of a molecular surface. J Mol Graph. 1994;12(3):162–8. doi:10.1016/0263-7855(94)80082-0.
7. Yang H, Qureshi R, Sacan A. Protein surface representation and analysis by dimension reduction. Proteome Sci. 2012;10(Suppl 1):S1. doi:10.1186/1477-5956-10-S1-S1.
8. Sasin JM, Godzik A, Bujnicki JM. SURF'S UP! - protein classification by surface comparisons. J Biosci. 2007;32(1):97–100. doi:10.1007/s12038-007-0009-0.
9. Koromyslova AD, Chugunov AO, Efremov RG. Deciphering fine molecular details of proteins' structure and function with a Protein Surface Topography (PST) method. J Chem Inf Model. 2014;54(4):1189–99. doi:10.1021/ci500158y.
10. Levieux G, Montes M. Towards real-time interactive visualization modes of molecular surfaces: examples with Udock. IEEE VR 2015 Workshop on Virtual and Augmented Reality dedicated to Molecular Science (VARMS). 2015.
11. Molecular Operating Environment (MOE). 2010.10. 1010 Sherbooke St. West, Suite #910, Montreal, QC, Canada, H3A 2R7: Chemical Computing Group Inc; 2010. https://www.chemcomp.com/MOE-Molecular_Operating_Environment.htm. Accessed 19 Feb 2016.
12. Wiener H. Structural determination of paraffin boiling points. J Am Chem Soc. 1947;69(1):17–20.
13. Balaban AT. Five new topological indices for the branching of tree-like graphs. Theor Chim Acta. 1979;53:355–75.
14. Balaban AT. Highly discriminating distance-based topological index. Chem Phys Lett. 1982;89(5):399–404. doi:10.1016/0009-2614(82)80009-2.
15. Petitjean M. Applications of the radius-diameter diagram to the classification of topological and geometrical shapes of chemical compounds. J Chem Inf Comput Sci. 1992;32(4):331–7. doi:10.1021/ci00008a012.
16. Pearlman RS, Smith KM. Novel software tools for chemical diversity. In: Kubinyi H, Folkers G, Martin YC, editors. 3D QSAR in drug design: three-dimensional quantitative structure activity relationships. Volume 2. Netherlands: Springer; 1998. p. 339–53. doi:10.1007/0-306-46857-3_18.
17. Lipinski CA, Lombardo F, Dominy BW, Feeney PJ. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. Adv Drug Deliv Rev. 1997;23(1–3):3–25. doi:10.1016/S0169-409X(96)00423-1.
18. Oprea TI. Property distribution of drug-related chemical databases. J Comput Aided Mol Des. 2000;14(3):251–64. doi:10.1023/A:1008130001697.
19. Stanton DT, Jurs PC. Development and use of charged partial surface area structural descriptors in computer-assisted quantitative structure-property relationship studies. Anal Chem. 1990;62(21):2323–9. doi:10.1021/ac00220a013.
20. Kier LB, Hall LH. The nature of structure-activity relationships and their relation to molecular connectivity. Eur J Med Chem. 1977;12:307–12.
21. Hall LH, Kier LB: The molecular connectivity chi indexes and kappa shape indexes in structure-property modeling. In: Lipkowitz KB, Boyd DB, editors. Reviews in Computational Chemistry. Volume 2. Hoboken, New Jersey: John Wiley & Sons, Inc.; 1991. p. 367–422. doi:10.1002/9780470125793.ch
22. Stewart JJP. MOPAC manual. 7th ed. 1993.
23. Lide DR, editor. CRC handbook of chemistry and physics. Boca Raton: CRC Press; 1994.
24. Wildman SA, Crippen GM. Prediction of physicochemical parameters by atomic contributions. J Chem Inf Comput Sci. 1999;39(5):868–73. doi:10.1021/ci990307l.
25. Ertl P, Rohde B, Selzer P. Fast calculation of molecular polar surface area as a sum of fragment-based contributions and its application to the prediction of drug transport properties. J Med Chem. 2000;43(20):3714–7. doi:10.1021/jm000942e.
26. Hou TJ, Xia K, Zhang W, Xu XJ. ADME evaluation in drug discovery. 4. Prediction of aqueous solubility based on atom contribution approach. J Chem Inf Comput Sci. 2004;44(1):266–75. doi:10.1021/ci034184n.
27. Kazius J, McGuire R, Bursi R. Derivation and validation of toxicophores for mutagenicity prediction. J Med Chem. 2005;48(1):312–20. doi:10.1021/jm040835a.
28. Vlachakis D, Kontopoulos DG, Kossida S. Space constrained homology modelling: the paradigm of the RNA-dependent RNA polymerase of dengue (type II) virus. Comput Math Methods Med. 2013;2013:108910. doi:10.1155/2013/108910.
29. Snyder JP. Map projections - a working manual, U.S. Geological survey professional paper 1395. Washington, DC: United States Government Printing Office; 1987.
30. Hammer E. Über die Planisphäre von Aitow und verwandte Entwürfe, insbesondere neue flächentreue iihnlicher Art. Petermanns Geogr Mitt. 1892;38(4):85–7.
31. Miller OM. Notes on cylindrical world map projections. Geogr Rev. 1942;32(3):424–30.
32. Wickham H. ggplot2: elegant graphics for data analysis. New York: Springer; 2009.
33. Cook T. Imager - Perl extension for generating 24 bit images. https://metacpan.org/pod/Imager. Accessed 27 Sep. 2015.
34. Champoux Y. Parallel::ForkManager - A simple parallel processing fork manager. https://metacpan.org/pod/Parallel::ForkManager. Accessed 27 Sep. 2015.
35. Box GEP, Cox DR. An analysis of transformations. J R Stat Soc Series B Stat Methodol. 1964;26(2):211–52.
36. Ensign DL, Kasson PM, Pande VS. Heterogeneity even at the speed limit of folding: large-scale molecular dynamics study of a fast-folding variant of the villin headpiece. J Mol Biol. 2007;374(3):806–16. doi:10.1016/j.jmb.2007.09.069.
37. Pettersen EF, Goddard TD, Huang CC, Couch GS, Greenblatt DM, Meng EC, Ferrin TE. UCSF Chimera - a visualization system for exploratory research and analysis. J Comput Chem. 2004;25(13):1605–12. doi:10.1002/jcc.20084.
38. Shibata N, Inoue T, Nagano C, Nishio N, Kohzuma T, Onodera K, Yoshizaki F, Sugimura Y, Kai Y. Novel insight into the copper-ligand geometry in the crystal structure of *Ulva pertusa* plastocyanin at 1.6-Å resolution: structural basis for regulation of the copper site by residue 88. J Biol Chem. 1999;274(7):4225–30. doi:10.1074/jbc.274.7.4225.
39. Do CB, Mahabhashyam MSP, Brudno M, Batzoglou S. ProbCons: Probabilistic consistency-based multiple sequence alignment. Genome Res. 2005;15(2):330–40. doi:10.1101/gr.2821705.
40. Stamatakis A. RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. Bioinformatics. 2014;30(9):1312–3. doi:10.1093/bioinformatics/btu033.
41. Glykos NM, Cesareni G, Kokkinidis M. Protein plasticity to the extreme: changing the topology of a 4-α-helical bundle with a single amino acid substitution. Structure. 1999;7(6):597–603. doi:10.1016/S0969-2126(99)80081-1.

Submit your next manuscript to BioMed Central and we will help you at every step:

- We accept pre-submission inquiries
- Our selector tool helps you to find the most relevant journal
- We provide round the clock customer support
- Convenient online submission
- Thorough peer review
- Inclusion in PubMed and all major indexing services
- Maximum visibility for your research

Submit your manuscript at www.biomedcentral.com/submit

BioMed Central