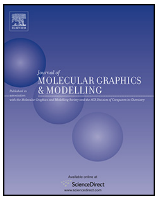
Journal of Molecular Graphics and Modelling 53 (2014) 59–71

Contents lists available at ScienceDirect

Journal of Molecular Graphics and Modelling

journal homepage: www.elsevier.com/locate/JMGM



A collaborative visual analytics suite for protein folding research

William Harvey[a,*], In-Hee Park[b], Oliver Rübel[c], Valerio Pascucci[d], Peer-Timo Bremer[e], Chenglong Li[f], Yusu Wang[g,**]

[a] Department of Computer Science and Engineering, The Ohio State University, Columbus, OH, United States
[b] Chemical Physics Program, The Ohio State University, Columbus, OH, United States
[c] Visualization Group, Lawrence Berkeley National Laboratory, Berkeley, CA, United States
[d] Scientific Computing and Imaging Institute, University of Utah, Salt Lake City, UT, United States
[e] Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA, United States
[f] Chemical Physics Program and College of Pharmacy, The Ohio State University, Columbus, OH, United States
[g] Department of Computer Science and Engineering, The Ohio State University, Columbus, OH, United States

ARTICLE INFO

Article history:
Received 14 May 2014
Accepted 17 June 2014
Available online 14 July 2014

Keywords:
Molecular simulation data
Visualization tool

ABSTRACT

Molecular dynamics (MD) simulation is a crucial tool for understanding principles behind important biochemical processes such as protein folding and molecular interaction. With the rapidly increasing power of modern computers, large-scale MD simulation experiments can be performed regularly, generating huge amounts of MD data. An important question is how to analyze and interpret such massive and complex data.

One of the (many) challenges involved in analyzing MD simulation data computationally is the high-dimensionality of such data. Given a massive collection of molecular conformations, researchers typically need to rely on their expertise and prior domain knowledge in order to retrieve certain conformations of interest. It is not easy to make and test hypotheses as the data set as a whole is somewhat “invisible” due to its high dimensionality. In other words, it is hard to directly access and examine individual conformations from a sea of molecular structures, and to further explore the entire data set. There is also no easy and convenient way to obtain a global view of the data or its various modalities of biochemical information.

To this end, we present an interactive, collaborative visual analytics tool for exploring massive, high-dimensional molecular dynamics simulation data sets. The most important utility of our tool is to provide a platform where researchers can easily and effectively navigate through the otherwise “invisible” simulation data sets, exploring and examining molecular conformations both as a whole and at individual levels. The visualization is based on the concept of a *topological landscape*, which is a 2D terrain metaphor preserving certain topological and geometric properties of the high dimensional protein energy landscape. In addition to facilitating easy exploration of conformations, this 2D terrain metaphor also provides a platform where researchers can visualize and analyze various properties (such as contact density) overlayed on the top of the 2D terrain. Finally, the software provides a collaborative environment where multiple researchers can assemble observations and biochemical events into storyboards and share them in real time over the Internet via a client-server architecture.

The software is written in Scala and runs on the cross-platform Java Virtual Machine. Binaries and source code are available at http://www.aylasoftware.org and have been released under the GNU General Public License.

1. Introduction

Proteins are the building blocks of cells and are responsible for nearly all cellular functions [1]. One of the fundamental goals of biological research is to understand the mechanics of an organism in terms of its constituent proteins. Recent advancements in gene sequencing have provided massive amounts of genomic data encoding the amino acid sequences of the constituent proteins of an organism. However, this sequence alone is insufficient for

[*] Corresponding author.
[**] Corresponding author. Tel.: +1 6142921309; fax: +1 6142922911.
E-mail addresses: harveywi@cse.ohio-state.edu (W. Harvey), yusu@cse.ohio-state.edu (Y. Wang).

http://dx.doi.org/10.1016/j.jmgm.2014.06.003
1093-3263/© 2014 Elsevier Inc. All rights reserved.

determining the native conformational structure (i.e. the stable three-dimensional shape), and consequently the function, of a protein. It is through the process of folding that the conformational structure of a protein transitions from a random coil to the functional native conformation uniquely determined by its amino acid sequence [3].

Understanding this folding process is of paramount importance. Hence, significant effort has been devoted to investigating the dynamics and kinetics of protein folding. Molecular dynamics (MD) simulations are key tools in this effort. For example, an important theory of protein folding (the so-called "energy landscape" theory [39,8,25]) assumes that folding occurs through an organizing ensemble of structures via many possible pathways and intermediates. Through analysis of MD simulation data, key local and global interactions can be detected along the folding pathway obtained by, for example, projecting simulation data onto certain parameters (so-called *reaction coordinates*), e.g. native contacts, radius of gyration, and principal components [32,27,52,48].

With the computational power of massively parallel modern computers, large-scale MD simulations are now routinely performed, yielding huge amounts of simulation data. As such, there is a pressing need for better tools to help users explore and interpret these massive, high-dimensional simulation data sets. In this paper, we describe an intuitive and effective visualization platform to facilitate the exploration of simulation data in a collaborative environment.

1.1. Related work in molecular simulation data visualization

Given a set of MD simulation data, one can interpret it as a sampling of the conformational space of a given molecule. One important concept associated with a molecular conformation is its energy, which is often quantified in terms of free or potential energy. This energy function (defined on the molecular conformational space) has been fundamental in understanding protein conformational spaces and folding mechanisms.

The molecular conformational space is intrinsically very high dimensional. In order to obtain a global view of a molecular simulation data set, a natural approach is to project the data to $\mathbb{R}^2$ (or $\mathbb{R}^3$) in some way. For example, one can project the molecular conformations into $\mathbb{R}^2$ by choosing a pair of *reaction coordinates*,[1] then represent each conformation in terms of these two coordinates [41,34,53]. One can then visualize the energy function on this low-dimensional projection by plotting the contours (isocurves) of the energy function. However, it is often difficult to judge the goodness of a set of reaction coordinates, and there is little consensus on this in the literature [39,32,52]. Furthermore, projecting data onto specific reaction coordinates may cause loss of information about other important properties.

Rather than committing to a set of reaction coordinates, one could instead use a general-purpose dimensionality reduction algorithm to project the data into low dimensions; e.g., [4,13,19,42]. For example, Das et al. [13] and Plaku et al. [42] use a nonlinear dimensionality reduction approach based on the Isomap algorithm [47] to produce a 2D map from a set of MD conformations. By assigning an energy-correlated color to each point in the 2D map, their algorithm ScIMAP produces effective visualizations for several data sets generated from coarse-grained simulation. Hamprecht et al. [19] use multidimensional scaling techniques to obtain a low-dimensional representation of conformational space. They also employ an interesting basin spanning tree idea for visualizing the structure within energy basins. This in turn provides a glimpse into the topography of the high-dimensional energy landscape.

While many elegant dimensionality reduction algorithms have been developed in the past few years (see e.g. [47,7,43,21,12,11] in the field of machine learning), these methods usually aim to preserve either global or local (distance) metrics. If the intrinsic dimension of the protein conformational space is far above 2 or 3, then distance distortion is unavoidable and can be arbitrarily large. If we now visualize a scalar field (such as the potential energy function) over the projected domain, then topological features (also referred to as topographical features), such as peaks and valleys in the low-dimensional projection, are often merely artifacts of distance distortion; that is, they do not correspond to true "peaks" and "valleys" in the high dimensional energy landscape. One example of this phenomenon is given by Harvey and Wang [20].

Given the importance of the energy function defined on the molecular conformation space, there has been a different line of work using the so-called disconnectivity graph to capture energy basins (valleys) and the connection between them through saddles. The disconnectivity graph was first proposed in [6] for potential energy, and has since been extended to free energy and has been widely used to understand the high-dimensional energy landscape for complex systems; see e.g., [15,29,26,49,50]. The disconnectivity graph is usually a rooted tree with leaves corresponding to local energy minima and internal nodes corresponding to connecting saddles of the energy function. While the disconnectivity graph provides a good way to show the key topographical features of the high dimensional energy landscape, it is not easy to explore and access the input molecular simulation data through this tree representation. As we will see later, the method proposed in this work will provide a terrain platform for such trees to enable visual analysis of the high dimensional simulation data.

Finally, we note that Stone et al. [46] presented an immersive visualization environment which uses specialized data structures and interactive techniques for trajectory visualization. However, their approach mainly focuses on efficient trajectory animation and does not deal with the problem of visual summarization or navigation of the data in its entirety. We also note that the term "terrain metaphor" was used before in [54], which presents a nice tool for visualizing and analyzing *patterns* in folding trajectories. In particular, the terrain in [54] is built based on patterns (identified by clustering algorithms) in folding trajectories and the frequency of their occurrences. The principle and visualization methodology are both different from our current work.

1.2. New approach and contribution

When metric preservation in dimensionality reduction becomes hopeless, we search for other sources of relevant information which can be conveyed in low dimensions. Given the importance of (free and potential) energy of a protein conformation in understanding molecular conformational spaces and folding mechanisms, we aim to preserve characteristic features of the energy landscape (which is the graph of the high dimensional energy function) using a low-dimensional metaphor. Specifically, the information which we preserve is similar to that which is encoded in the disconnectivity graph. However, rather than conveying this information in the form of a tree, we communicate this data in the form of a 2D landscape (see the right panel in Fig. 1) which provides some additional advantages and opportunities for interaction and visualization.

In particular, recall that given a set of molecular simulation data, we consider it to be a sampling of the conformational space C of this molecule. Now consider the energy landscape $E : C \to \mathbb{R}$, which is simply a high dimensional scalar function with the function value at each point (conformation) being the energy of this conformation. We then build a two-dimensional landscape $L : [0, 1] \times [0, 1] \to \mathbb{R}$

[1] Reaction coordinates can be thought of as parameters potentially important for describing molecular conformations or folding dynamics.

(i.e., a scalar field defined on the square $[0, 1] \times [0, 1]$) as a metaphor for E with the property that E and L share the same contour tree (we will make this precise shortly in Section 2). L is called a *topological landscape metaphor* for the scalar field E, a concept originally proposed by Weber et al. [51] and further developed in [20]. See the right panel in Fig. 1, where we show the landscape (terrain) L (the graph of the function L), with the vertical direction indicating the energy function value. Intuitively, L preserves the mountain peaks and valleys (basins) of E. Merging or splitting of peaks/valleys in L indicates corresponding events in the high dimensional energy landscape E. Furthermore, areas of mountain peaks and valleys are proportional to the volumes of their high-dimensional counterparts as well.

While the current formulation of the software uses the contour tree of the high-dimensional energy function to build the 2D terrain, it would be straightforward to substitute an alternative structure in place of the contour tree to achieve a variety of different landscapes. Any tree with a scalar function defined over its nodes could be used as a substitute for the contour tree. Hence we can also build a landscape metaphor for the disconnectivity graph [6,49] for data exploration and navigation.

Finally, the Mapper algorithm, developed and used in [45] for generic data analysis and successfully utilized in several biomedical applications [37,31], uses a graph structure to summarize a high-dimensional scalar field. This graph is embedded in three dimensions and serves as a platform for high-dimensional data exploration. Our software explores an orthogonal direction to the Mapper algorithm, where we focus on building an intuitive and interactive information visualization framework centered on a *2D terrain*: the terrain metaphor facilitates easy selection and inspection of molecular conformations, and allows the overlay of other (e.g. biochemical) information on the terrain (see Section 3.3).

1.2.1. Our contribution

We integrate this landscape metaphor to build an interactive, collaborative visual analytics tool for exploring massive, high-dimensional molecular dynamics simulation data sets. See Fig. 1 for one part of the interface of our software, where we link molecular structures and secondary structural information to the 2D landscape metaphor.

- Our tool is *interactive and intuitive*: Researchers can now “see” and navigate the entire set of conformations as a whole, as well as interactively select and examine individual conformations. Users can, for example, by traversing our 2D landscape and annotating key conformations, perform conformational analysis from the local-fluctuation induced sub-ensembles (contained within certain energy minima basins) all the way to the global unfolded conformations (e.g., energy maxima). The data exploration utility of our software can facilitate researchers in forming and testing folding mechanism hypotheses.
- Our tool provides *a platform for information integration*: Researchers can easily visualize other information of interest as overlays on the 2D terrain. Two specific examples include the fraction of certain secondary structure elements formed and the fraction of native contacts present in each conformation. Such integration would be much harder to visualize on a tree or a graph.
- Our tool also provides a *collaborative* environment where multiple researchers can assemble interesting observations and biochemical events into storyboards, share them, and have discussions in real time over the Internet via a client-server architecture.

We present various encouraging preliminary examples to illustrate these points and the utility of our software in Section 3.

To the best of our knowledge, no previous system exists that can provide concise summarization of the global structure of MD simulation data while allowing local interactive exploration of the sampled conformations. In particular, the full structural information of every simulated conformation is preserved in our software, and can be easily accessed using our 2D landscape. We believe that an intuitive and effective tool for exploring the large-scale high-dimensional simulation data is essential, yet currently lacking. Our tool makes an important step forward in closing this gap.

2. Methods

2.1. Background on topological landscapes

This section serves as an accessible introduction to the mathematical and topological foundations of our proposed visualization

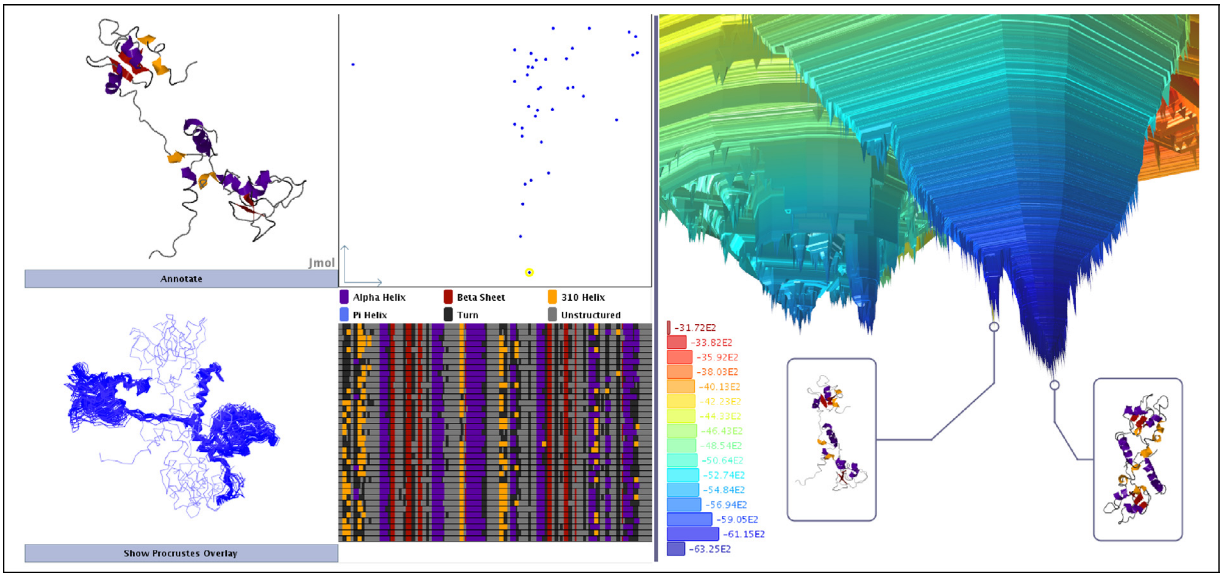


Fig. 1. Screenshot demonstrating the integrated views of the data set explorer. Clockwise from right: energy landscape visualization with annotated conformations shown as thumbnails; secondary structure information of the conformations in a selected topological component; 3D rotating procrustes superimposition of the backbones of a selected ensemble of conformations; Jmol view of one of the conformations; reaction coordinates (x: PCA coordinate, y: function value) of the conformations in the selected region.

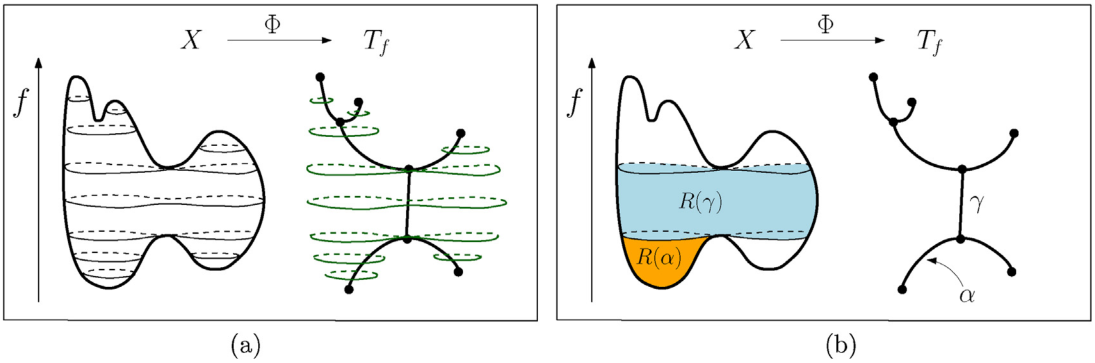


Fig. 2. (a) A two-manifold X with a function f, and its contour tree T_f. The value of f at each point is its height in the vertical direction. (b) Examples of topological components: lightly shaded region $R(\gamma)$ for the contour tree arc γ and darker region $R(\alpha)$ corresponding to arc α.

framework for non-experts in computational topology. For a more rigorous treatment of theoretical concepts we refer readers to the work of Weber et al. [51], and Harvey and Wang [20].

2.1.1 . The contour tree of a scalar function

Let M be a simply connected domain, and $f : M \to \mathbb{R}$ a scalar function defined on M. A *level set* of f with function value α (denoted $f^{-1}(\alpha)$) is defined as the set of points with function value α; that is, $f^{-1}(\alpha) = \{x \in X \mid f(x) = \alpha\}$. A level set may contain multiple connected components, each of which is called a *contour*. In what follows, it can be helpful to imagine the level sets and contours of a mountain range (i.e. a height function defined on a plane or a sphere) while reminding oneself that these concepts are fully applicable to higher-dimensional domains as well.

Now, imagine smoothly varying α while observing the changes that happen to the contours (i.e., the connected components in the level set $f^{-1}(\alpha)$). As α increases, the connected components in the level set $f^{-1}(\alpha)$ may appear, disappear, split, or merge. The *contour tree* of f encodes these changes. For example, Fig. 2 shows a simple height function on a two-manifold and its corresponding contour tree. Intuitively, the contour tree T_f is obtained by continuously collapsing each contour of f into a single point. Mathematically, there exists a continuous surjective map $\Phi : X \to T_f$ such that two points $x, y \in X$ have the same image $\Phi(x) = \Phi(y)$ in T_f *if and only if* x and y belong to the same contour of f for some value α.

More specifically, as we sweep through a minimum of the scalar field, a new connected component is created in the level set, and thus a degree-1 node emerges in the contour tree. A down-fork degree-3 node in the contour tree corresponds to the event where two contours merge into a single one, while an up-fork degree-3 node corresponds to the split of one contour into two connected components. As we sweep through a maximum of the scalar field, a connected component will disappear, yielding another degree-1 node in the contour tree. Each arc of contour tree corresponds to the evolution of a connected component (contour) in the level set, from the time it is created, till it either splits, merges with another contour, or disappears. Given a contour tree arc γ, we call the union of points from X mapped to a point in γ the *topological component* $R(\gamma)$ associated with γ; that is, $R(\gamma) = \{x \in X \mid \Phi(x) \in \gamma\}$. See Fig. 2(b). We note that at a down-fork node in the contour tree, two valleys (corresponding to the union of topological components of all arcs in the left and right sub-trees, respectively) also merge. Similarly, an up-fork node in the contour tree also indicates a mountain-splitting event in the original scalar field $f : X \to \mathbb{R}$. Hence intuitively, the contour tree T_f encodes those valley-merging and mountain-splitting events of the input scalar field that we will aim to capture for the high-dimensional protein energy landscape.

2.1.2. Topological landscapes

With the contour tree of f in hand, it is possible to produce a family of functions in the plane with identically matching contour trees. Each such function is of the form $g : [0, 1] \times [0, 1] \to \mathbb{R}$, and the graph of it is a surface in $\mathbb{R}^3$ (with the vertical direction showing the function value), which we refer to as a *topological landscape metaphor* for the high-dimensional scalar field f. This concept was first proposed and computed by Weber et al. [51]. Intuitively, each landscape metaphor encodes the same mountain valley and peak information as its high-dimensional counterpart: when two valleys merge (resp. a mountain splits into two peaks) in the metaphor, the corresponding high dimensional valleys also merge (resp. mountains split). Harvey and Wang [20] show that that not only can one create a complete set of 2D landscapes matching the contour tree of a high-dimensional scalar field, but one can also guarantee that the area of each topological component (intuitively, volume of each peak/valley region, recall Fig. 2) in the 2D landscape is the same as the volume of its high-dimensional counterpart. We call each such landscape $g : [0, 1] \times [0, 1] \to \mathbb{R}$ a *volume-preserving landscape metaphor* for f. In our software, we use the algorithm of [20] to compute a volume-preserving landscape metaphor for the input high-dimensional energy function.

2.2. The visual analytics framework

We now describe our visual analytics platform for MD simulation data in detail. There are two main components which we will describe in Sections 2.2.1 and 2.2.2 respectively. The first is the pre-processing of data to prepare input molecular simulation data into a form that can be fed into the visualization platform. The second one is the collaborative visualization platform with which users can interact.

2.2.1. Pre-processing stage

The input to our software package is a set of conformations in a MD simulation data set, where each conformation is given as a PDB file. We call such an input an *MD data set*. Note that we treat simulation data simply as a collection of conformations; any trajectory information that may exist during the sampling process (such as in an NPT or REMD simulation) is not considered in building our terrain metaphor. However, such information can be shown by overlaying it on top of the terrain metaphor later.

The preprocessing steps are outlined in Fig. 3. First, the MD data set (Fig. 3(a)) is converted to a cloud of points in high dimensions $\mathbb{R}^D$ (Fig. 3(b)). Next, we need to connect these discrete points in $\mathbb{R}^D$ to approximate the hidden protein conformational space, as well as generate a scalar function $\tilde{E}$ that approximates the energy function defined on the protein conformational space (see Fig. 3(c)). Finally, we compute the contour tree T from the scalar function $\tilde{E}$, together

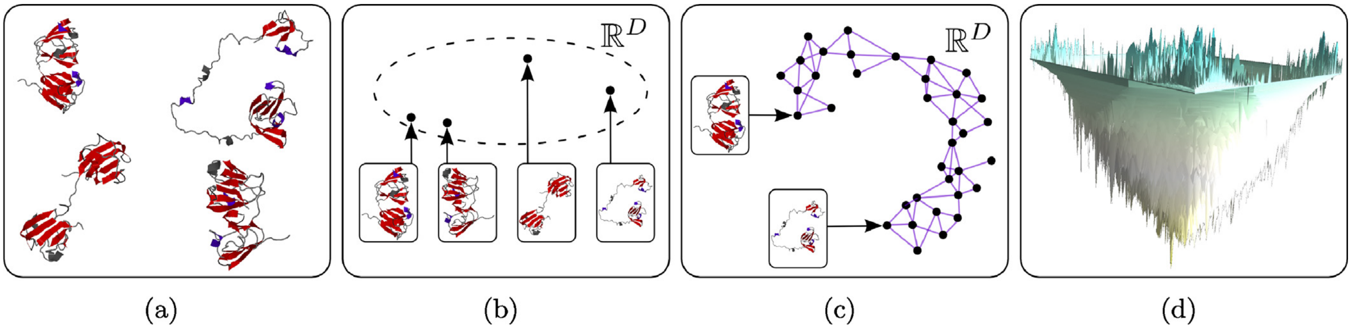


Fig. 3. Summary of data set preprocessing. (a) A set of conformations with potential energies is provided as input to the software. (b) Each conformation is mapped to a point in $\mathbb{R}^D$. (c) Each point is connected with its neighbors to form a neighborhood graph. (d) The neighborhood graph is combined with the potential energy function and transformed into a topological landscape.

with some other information. We will feed such a tree T to our visualization platform to generate a 2D terrain metaphor (Fig. 3(d)). The details of each preprocessing step are described below.

The software is written in Scala [38], runs on the cross-platform Java Virtual Machine, and is API-compatible with Java programs. Binaries and source code are available at http://www.aylasoftware.org and released under the GNU General Public License.

2.2.1.1. Step 1: Metrizing conformations. The first step in the pipeline is to map each conformation to a point in high-dimensional Euclidean space such that the proximity of two points indicates similarity of their corresponding conformations. If a suitable set of reaction coordinates are known *a priori*, and if the Euclidean distance under these reaction coordinates is effective in capturing conformational changes of interest, then these reaction coordinates could serve as an appropriate map.

An appropriate set of reaction coordinates may not be known in advance. In this case, the software provides a default mapping by representing each conformation as a vector of pairwise distances between its α-carbon atoms. A collection of conformations of a protein with n α-carbon atoms would map to a cloud of points in $\mathbb{R}^{\binom{n}{2}}$. Note that users can easily overwrite this default mapping with other choices. For example, instead of using the $\binom{n}{2}$ number of pairwise distances between α-carbon atoms, one can also simply concatenate the x, y, z-coordinates of each α-carbon atom, and map a protein conformation to a point in dimension $3n$. There are different trade-offs: using pairwise distances makes the resulting coordinates invariant to rigid transformation; however, it cannot distinguish a structure from its mirror image. Using concatenation of x, y, z-coordinates can distinguish mirror images, but will be sensitive to rigid transformations.

Finally, we apply principal component analysis (PCA) to this set of point cloud data, and dimensions corresponding to small eigenvalues can be optionally discarded to remove noise. The optional reduction in the number of dimensions improves the efficiency of neighborhood graph construction (and potentially the quality, as dimensionality reduction can have a denoising effect). Note that the resulting point cloud data is still high-dimensional, and that the software will perform geometric calculations in this high-dimensional space to avoid metric distortion.

2.2.1.2. Step 2: Domain and scalar field approximation. Now we have converted a set of input protein conformations to a set P of high dimensional points in $\mathbb{R}^D$, which can be thought of as samples taken from the hidden protein conformational space C embedded in $\mathbb{R}^D$. From these points, we need to approximate both the hidden domain C and the energy function defined on it.

We can use the following simplicial complex K to approximate the domain C: first, every point in P is connected to its k-nearest neighbors; that is, we obtain a k-NN graph G_P for the set of conformations P. Next, we add a triangle p, q, u into K whenever all three edges of this triangle are already in G_P. One can continue this process for higher dimensional simplices: a d-simplex $\{p_0, \ldots, p_d\}$ is added if all pairwise edges from these points are present in G_P. (A d-simplex $\{p_0, \ldots, p_d\}$ is the convex hull spanned by $d+1$ points. A 0-simplex is a point, a 1-simplex is an edge and a 2-simplex is a triangle.) Using more formal terms, the simplicial complex K is the so-called clique complex induced by the k-NN graph G_P. This complex is also similar to the so-called Rips complex, widely used in the computational geometry and machine learning communities to approximate a hidden domain (see e.g. [2,9,44] and references within).

One important thing to note is that this simplicial complex K described above is completely and uniquely decided by the k-NN graph G_P. Hence we only need to compute G_P and K is then implicitly given. Furthermore, it turns out that to compute the contour tree, the graph G_P is sufficient. In other words, although we describe the complex K that approximates the hidden protein conformational space C, we do not need to construct it for our purpose: the k-NN graph G_P and the function value at vertices will suffice.

Specifically, suppose we are given a piecewise-linear (PL) function $\tilde{E} : K \to \mathbb{R}$ defined on this simplicial complex K, meaning that the function values are defined at vertices of K (which is P), and linearly interpolated within any higher-dimensional simplices. It turns out that only the 1-skeleton of K, namely, its edge set G_P, is needed to compute the contour tree T corresponding to the PL-function $\tilde{E}$ [10].

2.2.1.3. The choice of energy. Finally, to compute the approximated (piecewise-linear) energy function $\tilde{E} : K \to \mathbb{R}$, we only need to compute $\tilde{E}(p)$ for every conformation p. In the current implementation, we set $\tilde{E}(p)$ as the potential energy plus the solvation energy (based on the generalized Born GB^{0BC} implicit model), calculated via an AMBER FF99SB force field. We note that often it is desirable to compute the free energy of each conformation and build a 2D metaphor for the free energy landscape. The users can override the default potential energy computation by supplying a different (say free) energy value for each conformation. In what follows, we use the term high-dimensional *energy landscape* loosely to refer to either the free energy landscape or the potential energy landscape, whichever is available.

To summarize, in Step 2, our software first uses the fast exact nearest neighbor search algorithm of Hwang et al. [18] to efficiently calculate a k-nearest neighbor graph G_P for the set of conformations

P. We then compute the energy value for each protein conformation p in P, and set it as $\tilde{E}(p)$. This induces a PL-function $\tilde{E} : K \to \mathbb{R}$, and we use the contour tree algorithm by Carr et al. [10] to compute the corresponding contour tree T using only G_P.

2.2.1.4. Step 3: Contour tree and terrain metaphor generation. The contour tree algorithm [10] also computes, for each vertex (i.e., a conformation in P), its image in the output tree T. Hence for each contour tree arc γ of T, we can now collect the set of conformations mapped to this arc, denoted by P_γ, which is the set of conformations of P contained inside the topological component $R(\gamma)$ associated with γ. (Recall Fig. 2(b): all the conformations of P which fall inside the lightly shaded region $R(\gamma)$ are collected as P_γ.) We use P_γ to estimate the volume $vol(\gamma)$ of the topological component $R(\gamma)$ associated with γ. Estimating the volume of a high-dimensional region is a non-trivial problem. To obtain $vol(\gamma)$, one can estimate the density $w(p)$ at each sample point $p \in P_\gamma$ and take the weighted sum $\sum_{p \in P_\gamma}(1/w(p))$ as the volume measure. This, however, is time consuming and not necessarily accurate for high-dimensional data. Instead, we use the following simple strategy to obtain a value that reflects the volume: we take the centroid c of the set of points in P_γ, and return the average distance from all points in P_γ to c as $vol(\gamma)$. Strictly speaking, this is not a measure of volume, but still a measure of size. This is the default method to set up $vol(\gamma)$ in our software. Our tool also provides the option of simply computing $vol(\gamma) = |P_\gamma|$, namely, the number of sampled points in the topological component $R(\gamma)$. The latter method has a bias towards densely sampled regions, such as those near the global minimum. However, this will emphasize the valleys around global minimum, which could be a desirable effect for researchers in some scenarios.

The contour tree T, with each arc γ associated with a volume measure $vol(\gamma)$, constitutes the input to the algorithm of Harvey and Wang [20] to produce a 2D terrain metaphor. This algorithm produces a collection of all possible terrains. In our software, by default, we take the specific 2D metaphor $L : [0, 1] \times [0, 1] \to \mathbb{R}$ generated by mapping the contour corresponding to global maximum of the contour tree to the boundary of the 2D domain (i.e., the boundary of square $[0, 1] \times [0, 1]$). The intuition behind this choice is that researchers are typically interested in low-energy conformations and this choice emphasizes valleys of the energy landscape.

Finally, before the contour tree and associated information is uploaded to the server, there is an optional step which involves calculating the secondary structure of all conformations using the DSSP [28] algorithm. As we will see in the next section, our software provides some dedicated user interface components for exploring the secondary structures of conformations which tie in to the interactive elements of the topological landscape.

To summarize, at the end of pre-processing, the collection of conformations (each accompanied by its potential energy value and SSE decomposition if computed) and the contour tree T (where each arc is associated both with the indices of conformations P_γ mapped to it and the volume estimate of the corresponding topological component) are generated and ready to transfer to the server. The landscape metaphor is produced at the server after the preprocessed data is uploaded. Later, when a client computer connects to the collaboration server, it may access any available preprocessed data sets, and researchers can then interact with their corresponding topological landscapes using the platform that we will describe next.

2.2.2. Interactive collaborative visual analytics

When pre-processing is complete, a user uploads the MD data set to a collaboration server. One or more researchers can interact with the data set by using the visual analytics client software to connect to the server. Once connected, the 2D topological landscape is generated, and users can begin exploring the data.

To facilitate users in exploring, selecting and annotating protein conformations, the visual analytics client software features a collection of coordinated views of the data under analysis. Detailed examples of using the software will be provided in Section 3.

2.2.2.1. Data exploration interface. The main user interface of the visual analytics client is shown in Fig. 1. The 2D topological landscape metaphor is shown in the right panel. Recall that the 2D terrain is the graph of a function defined on a 2D domain, where the vertical direction (height) is the energy value. The users can interact with this terrain, rotating it, moving it around, and zooming in/out. The default color on the terrain shows the energy value. The color bars in the left corner indicate the range and frequency of function values, with energy values decreasing from the top to the bottom. An important utility of our software is to allow users to visualize other sources of information such as the contact density or formation of secondary structure elements on the 2D landscape. Such information will be input in the form of a scalar function, with one function value supplied for each conformation. Examples will be given in Section 3.

The users can also simplify the landscape at different levels of detail so as to single out important valleys and remove noise. Simplification is guided by standard topological persistence [14] which pairs degree-1 nodes (minima and maxima) in the contour tree with degree-3 nodes (down-forks and up-fork saddles). Roughly speaking, importance (persistence) is assigned to tree branches, and we can remove those branches with small persistence during simplification. Removing a branch in the contour tree means to merge the corresponding topological component to the one corresponding to the sibling branch when generating the 2D landscape metaphor. See Fig. 4 for an example.

Secondly, the software allows users to interactively select a region of interest (i.e. a topological component) from the 2D landscape. Once a region is selected, the interface provides information in several other panels for users to explore the details of conformations contained in this region.

Specifically, when the user clicks a point on the landscape, the topological component R containing this point is selected and becomes highlighted. Suppose this region R is the topological component corresponding to the contour tree arc γ. Then the collection $P_\gamma \subseteq P$ of conformations mapped to this arc γ will be retrieved (recall our discussion in Step 3 in pre-processing). The top middle panel of Fig. 1 provides a PCA embedding of P_γ with each point representing a protein conformation in P_γ. The user can then choose any point from this panel to select a specific conformation. The bottom middle panel shows the secondary-structure formation information for all conformations in P_γ, with each row representing one conformation. A 3D procrustes [17] superimposition of the backbones of all conformations in P_γ is shown in the left-bottom panel, so that the users can get an idea of the diversity of structures contained in the selected region. Finally, if the user selects a specific conformation (through the top middle panel), then a Jmol [24] view of this structure is shown in the top left panel. For both top-left and bottom-left panels, the users can interactively manipulate the structures shown.

2.2.2.2. Annotations and collaboration. The client component allows multiple users to collaborate using an integrated visual analytics environment. A single user can select landscape components for further analysis, navigate high-level patterns and low-level details based on the structure of the landscape, and record interesting folding events and important conformations in the form of *annotations*. Specifically, an annotation is a protein conformation, whose location is marked on the landscape metaphor, and supplied

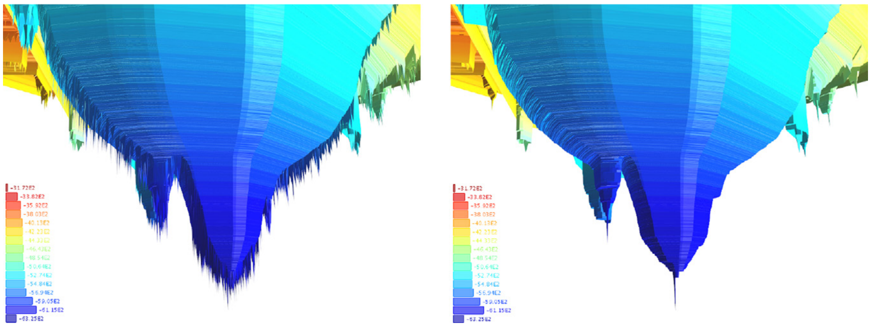


Fig. 4. (Left): a terrain and (right): its simplification.

with a short description from the user. Multiple annotations can be organized into a *storyboard* using a simple drag-and-drop interface. Annotations and storyboards created by users are automatically sent to the server, where they are pushed out to other connected users, ensuring that all collaborating researchers have a current version of the latest data. Annotations and storyboards provide a natural mechanism for collaboration as well as making the analysis process reproducible and traceable. Users can communicate with each other using an integrated chat interface. Fig. 5 demonstrates the basic workflow in creating a storyboard. Fig. 6 provides a snapshot of the interface where multiple users can collaborate and discuss.

Finally, we remark that the client-server architecture of the software provides several benefits. The server houses potentially large MD data sets and coordinates communication and collaboration between clients. When a client wishes to interact with a data set, only small subsets of the entire MD data set need to be downloaded from the server, which helps to make the interaction real-time.

3. Results and discussion

The goal of our software is to provide exploration utilities to enable domain experts and researchers to examine the simulation data, generate new insights, and help to build and test hypotheses. In this section we provide some examples to illustrate the basic utility and the potential of our software by applying it to two data sets described below.

3.1. Data sets

3.1.1. Survivin data

The survivin protein is overexpressed in tumor cells as a homodimer and is a target for anti-cancer drugs. The survivin data set [41] consists of three MD dynamics simulations: small-scale NPT MD [30] (437 conformations), medium-scale REMD (20,000 conformations) with low temperature ranging from 300 K to 340 K, and large-scale REMD [36] (232,559 conformations) with a wide range

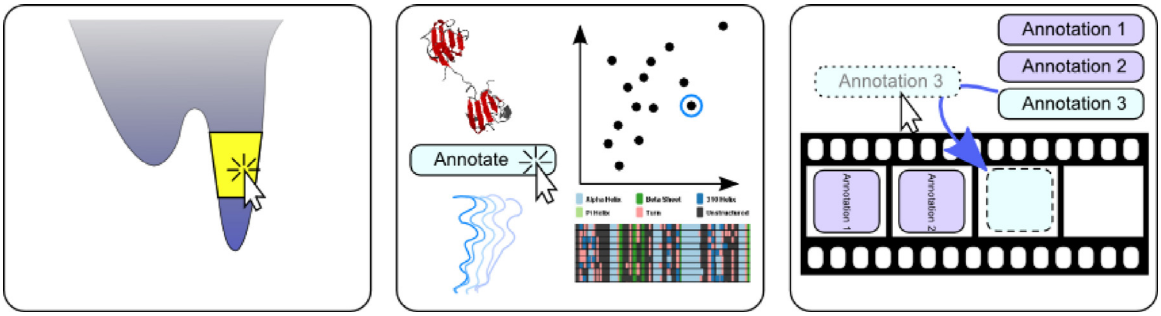
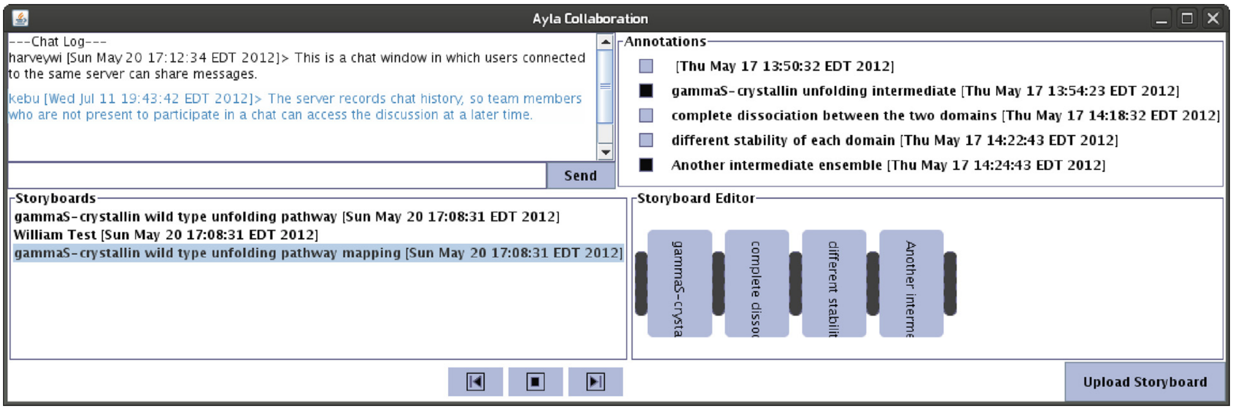


Fig. 5. Creating a storyboard. Left: user selects one or more topological components of the topological landscape. Center: user finds a conformation of interest in the localized reaction coordinates, and creates an annotation describing the conformation. Right: user drags and drops annotations into a filmstrip to create storyboards.



Ayla Collaboration

---Chat Log---
harveywi [Sun May 20 17:12:34 EDT 2012]> This is a chat window in which users connected to the same server can share messages.
kebu [Wed Jul 11 19:43:42 EDT 2012]> The server records chat history, so team members who are not present to participate in a chat can access the discussion at a later time.

Send

Annotations

□ [Thu May 17 13:50:32 EDT 2012]
■ gammaS-crystallin unfolding intermediate [Thu May 17 13:54:23 EDT 2012]
□ complete dissociation between the two domains [Thu May 17 14:18:32 EDT 2012]
□ different stability of each domain [Thu May 17 14:22:43 EDT 2012]
■ Another intermediate ensemble [Thu May 17 14:24:43 EDT 2012]

Storyboards

gammaS-crystallin wild type unfolding pathway [Sun May 20 17:08:31 EDT 2012]
William Test [Sun May 20 17:08:31 EDT 2012]
gammaS-crystallin wild type unfolding pathway mapping [Sun May 20 17:08:31 EDT 2012]

Storyboard Editor

gammaS-crysta | complete dissoc | different stabilit | Another interme

Upload Storyboard

Fig. 6. Screenshot of the user interface for the collaboration tools. Clockwise from upper-left: chat panel, annotation list, storyboard editor, and storyboard list. A storyboard can be played back using the three control buttons at the bottom.

of temperature from 280 K to 600 K. The PDB code for survivin is 1E31 and the homodimer has 228 residues (Δ5-117 and a Zinc atom per each domain) in the simulation system. Recall that the input to our software is simply the set of conformations produced during the simulation where each conformation is stored in a PDB file. Putting all conformations together, we have a set of 252,996 conformations. The total size of this data set is 57 GB of PDB files.

3.1.2. gammaS-crystallin data

Crystallins are water-soluble proteins found in the vertebrate eye which increase the refractive index of the lens without obstructing light [22]. The gammaS-crystallin protein has been linked to congenital cataract development [5]. See Refs. [33,16] for additional information. The gammaS-crystallin MD data set explored in this paper consists of 121,514 conformations, totaling 33.4 GB of PDB files. These are generated by a large-scale REMD sampling using a temperature range of 280 K to 551 K. The PDB code for crystallin is 2A5M that contains 177 residues.

We remark that we have chosen these two data sets in part due to the high complexity of the proteins involved, demonstrating the applicability of our tool to study and analyze highly complex systems. Unlike some fast folding proteins and/or coarse-grained simulations, these proteins and the REMD simulations induce complex energy landscapes which can be captured and shown by our landscape metaphor.

3.1.3. Timing

Here we provide the pre-processing time of the above two data sets. The timings have been collected using a laptop equipped with a 3.07 GHz Intel Core i7 CPU and 12 GB RAM. The parameter k (for building the k-NN graph) is chosen as $k = 19$ in these experiments. For the survivin data set with 252,996 total conformations, Steps 1 and 2 take a combined total of 2787.5 s (Step 1 takes 2288.9 s while Step 2 takes 568.6 s). For the gammaS-crystallin data set which contains 121,514 conformations, Steps 1 and 2 take a combined total of 1415.0 s (Step 1 takes 841.7 s while Step 2 takes 573.3 s). The size of these molecules is reported above.

3.2. Exploring large ensembles of molecular dynamics data sets

First, one of the most fundamental uses of our tool is viewing and navigating the data in its entirety. This is largely made possible by the landscape representation of the ensemble of conformations.

As illustrated earlier in Fig. 1, the landscape of the survivin data set shows a narrow funnel containing the native confirmation at its bottom tip. Using the interface in Fig. 1, the users can traverse the landscape metaphor, identify potential structures of interest (such as the conformation at the tip of the neighboring valleys of the main valley containing global minimum), and examine one or more of them to acquire knowledge and understanding about the data. When one or more key conformations are found, users can record them by creating an annotation with a short description in the landscape metaphor.

To further facilitate easy annotation of simulation data, the software provides a simple utility where, when users choose two conformations, say p and q, it returns a path $\pi(p, q)$ between them. The path $\pi(p, q)$ is the shortest path between p and q in the proximity graph G_P that we used to build the contour tree, and consists of conformations from the input simulation data set P. This path $\pi(p, q)$ may or may not have biological meaning. (It may be useful to add more choices of generating such a potential path, by say, a shortest weighted path where the weight of an edge reflects the likelihood of transition between its two nodes based on energy difference.) The goal is to provide a means for users to study a sequence of conformations connecting p and q. We note that such a path usually passes through the lowest saddle point connecting the two valleys containing p and q. It could be interesting to investigate whether such a saddle point corresponds to the energy barrier between p and q and whether $\pi(p, q)$ approximates a meaningful physical transition path from p to q. An example is shown in Fig. 7, where the minima of the two primary basins within the folding funnel are chosen as p and q, and the yellow path is the shortest path $\pi(p, q)$ returned by our software. By viewing the structures (in the left panels) of the protein conformation as we traverse along this path, it is clear that the deeper basin consists of more compact conformations and more complete development of secondary structure than the shallower basin. The software provides an easy-to-use mechanism for marking annotations along this path, which will also facilitate the process of creating storyboards. We note that a trajectory produced by a MD simulation can be viewed in a similar manner as a path on the terrain.

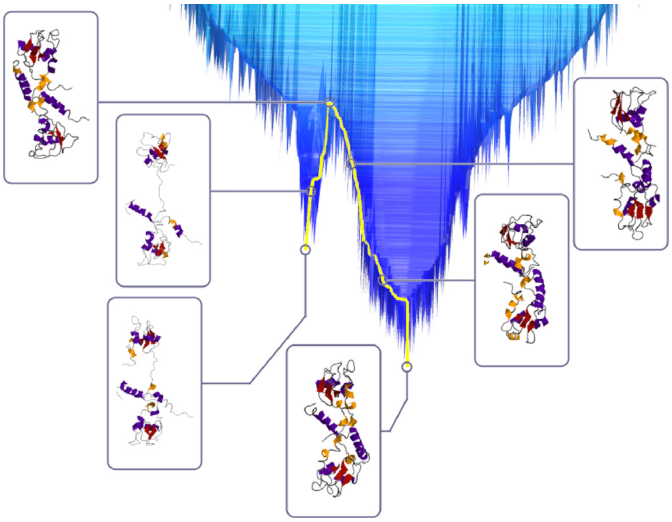


Fig. 7. Folding funnel of the survivin protein with some key conformations identified: the terrain in this figure is a portion of the large terrain shown in Fig. 2. The shortest path connecting the two minima is shown in yellow. In this example, the highest point of this path is the saddle point where two basins first merge intuitively, from this splitting point, two paths diverge to reach the minimum of each valley.

3.2.1. Annotation and storyboard building

Next, we turn our attention to the storyboard builder as a tool for cataloging interesting events (such as folding events). To obtain a coarse-grained, global picture of the conformational changes of the survivin protein, we map out the distinctive events along its folding pathway in the form of a storyboard. While the storyboard of Fig. 7 was created by selecting two conformations and inspecting the intermediary conformations along the shortest path connecting them, this storyboard, as shown in Fig. 8, was created manually, incorporating prior knowledge about the hypothetical unfolding pathway of survivin.

Fig. 8 illustrates the set of annotated conformations in this storyboard numbered according to their sequence in the storyboard. The corresponding annotation descriptions were supplied during construction of the storyboard by a domain expert familiar with the folding dynamics of survivin:

> (1) The low-energy conformation ensemble. (2) While preserving the dimer interface and packing of each monomer, the N-terminal loops open. This results in a loose packing of each monomer. (3) The dimer interface and BIR domains are intact, but as the C-terminal, long alpha-helix becomes a random coil, the BIR domains begin to separate from the interface. (4) While the dimer interface is still intact, the BIR domain gets away from the apoptosis/mitosis switch loop (i.e. the dimer interface loop).

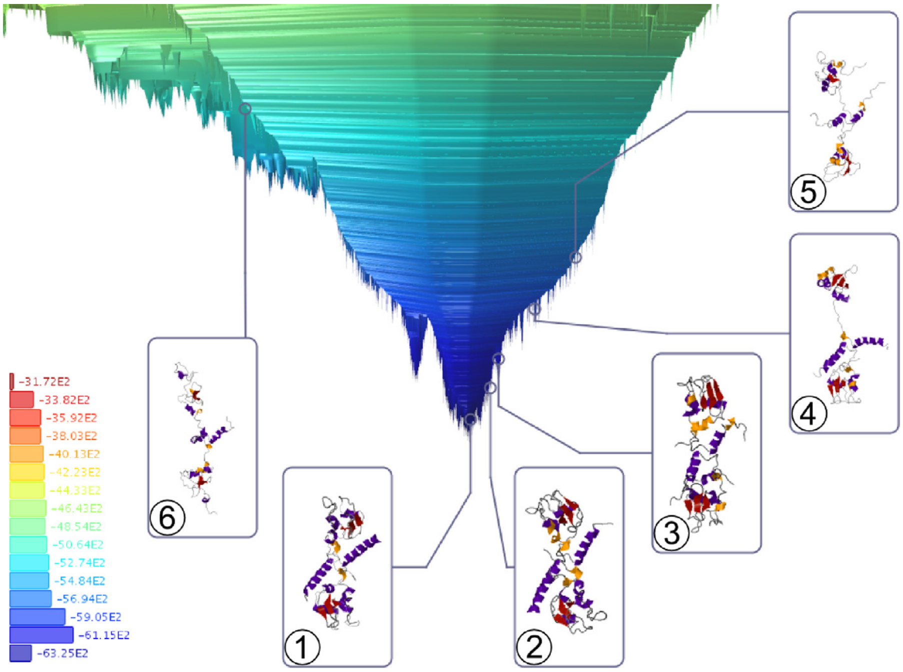


Fig. 8. Storyboard illustrating the unfolding process of survivin. The sequence of conformations shown were identified and assembled into a storyboard by a domain expert familiar with the folding dynamics of survivin.

(5) and (6) The BIR domain loses its stable zinc coordinate while the dimeric interface loop is still intact.

This hypothetical unfolding mapping may be consistent with recent research on survivin regarding the very resistant dimer interface. Pavlyukov et al. [40] propose that, due to the higher affinity of the dimerization region to various ligands than the BIR domain, antitumor drugs targeting the survivin monomer could be more efficient than peptide mimetics of Smac/DIABLO.

Likewise, a storyboard was created to record some interesting unfolding events of gammaS-crystallin (see Fig. 9). This is obtained by exploring conformations from the low-energy basins around the global energy minimum, which can serve as a subset of “native-like” ensemble that includes partially folded structures induced by local-fluctuation in energy.

(1) The domain-dissociated conformation with each domain compact, resembling beta B2-crystallin (in its natively domain-dissociated form) [35]. The functionally domain-dissociated form may be an alternative way to maintain stability by forming a dimer or multimer through its subdomain association. (2) A native-like structure. (3) A native-like structure, albeit without the alpha/3_{10} helix that directly interface with the surface. This is perhaps the first denaturation-prone site. (4) Intermediate opening conformation upon loss of inter-domain interaction between Ile60 . . . Ile137. (5) The domain-dissociated conformation with each domain less compact. Once the solvent-exposure shielding function of the alpha helix and 3_{10} helix was lost, the hydrophobic core was exposed to the solvent; the Tyr/Trp-corner maintained the beta-structure to the end. (6) and (7) One domain remains roughly compact, while the other one starts to dissolve.

Flaugh et al. [16] proposed three hypothetical kinetic models for the folding mechanism of gammaD-crystallin (see Fig. 9 in Flaugh et al. [16]). By inspecting conformations in the low-energy basins, as well as conformations slightly above the rectangular region in Fig. 9, it seems that the folding kinetics of gammaS-crystallin fits a

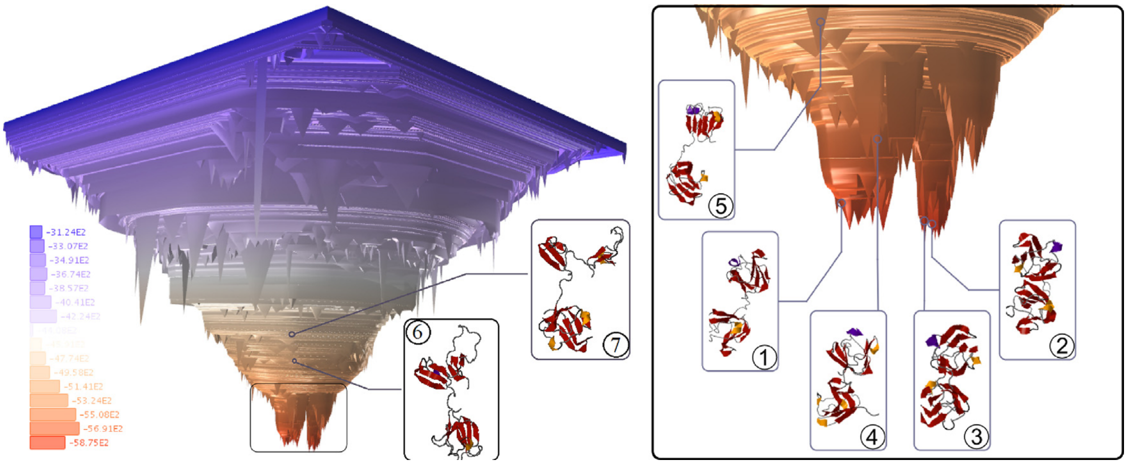


Fig. 9. Storyboard illustrating the unfolding process of gammaS-crystallin. The right picture is a zoomed-in view of the region in the rectangle of the left picture.

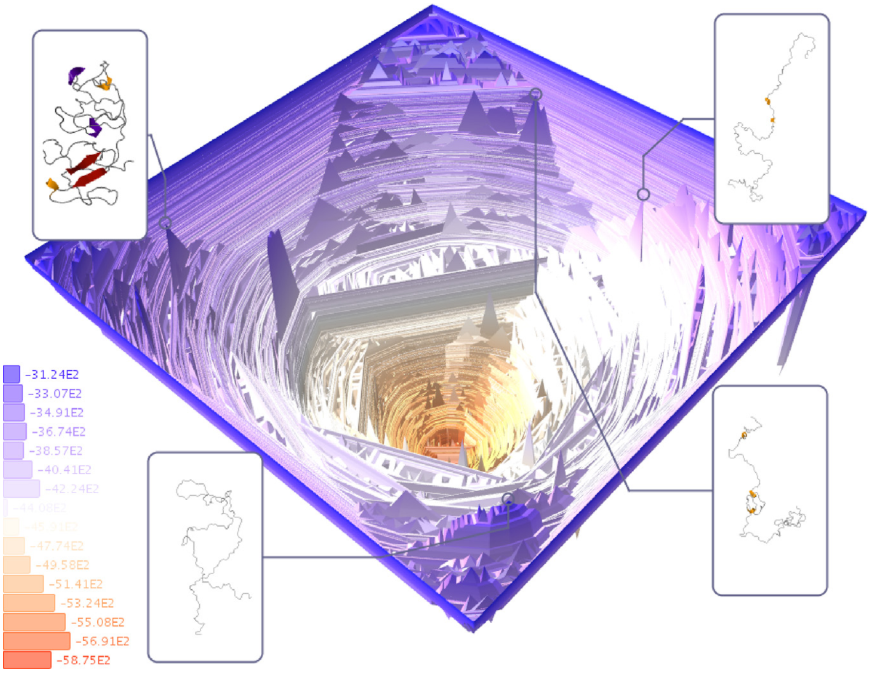


Fig. 10. Oblique view of the gammaS-crystallin landscape with a few local maxima identified.

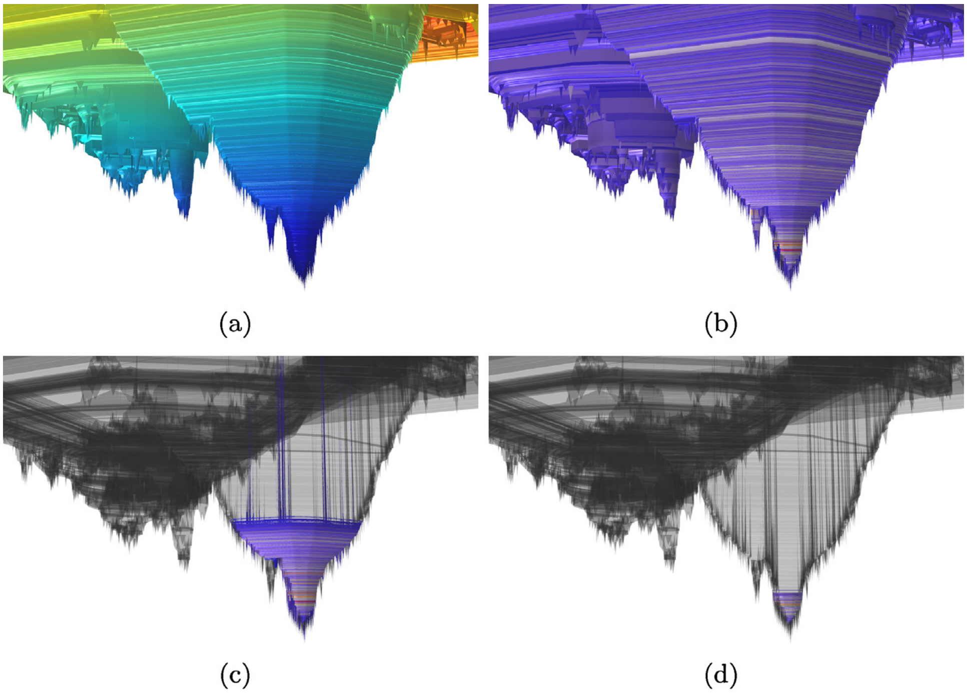


Fig. 11. Comparing the three different survivin simulation data sets. Sideview of landscape showing (a) the potential energy function, (b) large-scale REMD simulation, (c) medium-scale REMD simulation, and (d) NPT MD simulation.

three-state model (from top to bottom of the energy landscape). First, one domain is folded (annotations #(6)–(7)), then the other one folds (annotations #(4)–(5)). Last the two domains are brought together and the compact dimer is formed (annotations #(2)–(3)). Hence one can hypothesize that, out of the three proposed models, the Model 2 from Fig. 9 of Flaugh et al. [16] could be a probable mechanism for gammaS-crystallin folding pathway. This could stimulate more detailed study into the relative stability of each domain by analyzing the conformations involved in low-energy basins in order to test this hypothesis.

We remark that, due to the intrinsic complexity of these exemplar proteins, understanding folding pathways is very challenging. However, as the above example shows, our software provides a way for researchers to both obtain a global view of the data, and to access individual conformations. Researchers can focus on their conformational analysis either among local-flucation induced conformations (those contained in energy minima basins), or all the way up to global unfolding events (high energy maxima). The above example illustrates the potential of our software in assisting researchers to obtain understanding of the data as well as to form hypotheses.

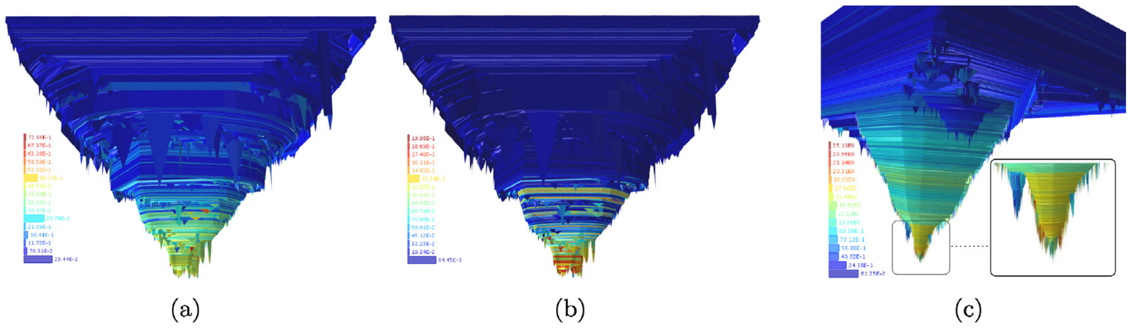


Fig. 12. Displaying secondary structure formation on the landscape. (a) Indicates 3_{10} helix formation and (b) indicates beta sheet formation for gammaS-crystallin. (c) Alpha helix formation for survivin with the bifurcation enlarged for clarity.

In addition to exploring the basins of the potential energy function, the topological landscape reveals information encapsulating the topological arrangement of local maxima and their corresponding neighborhoods. Fig. 10 shows an oblique view of the gammaS-crystallin landscape which reveals some maxima. We show a few conformations whose energies are locally maximal.

3.3. *The landscape as a platform for information visualization*

In addition to data navigation and annotation, another important utility of our software involves harnessing the landscape as a platform for visualizing and analyzing auxiliary information. For example, one can imagine visualizing any popular reaction coordinate from the literature on top of the terrain metaphor. Currently, by modeling each additional feature of interest as another function $g : P \rightarrow \mathbb{R}$ (i.e., a function value is supplied to each conformation), the software provides a visualization of g on top of the landscape metaphor (with colors indicating function values). The users can obtain a global view of how this information g varies across the entire sampled domain, and how it correlates with the energy function used to build the landscape metaphor. They can also localize specific regions where points of interests are identified through the visualization (which is the height in the 2D landscape). In what follows, we provide some examples of this utility: while our examples are simple, they illustrate general directions in which researchers can potentially use our software.

3.3.1. *Comparing conformational samplings*

We can use our visual analytic platform to compare how different simulation methods sample the conformational space. To illustrate the principle of this usage, recall that for the survivin data, we have three simulation data sets. The landscape metaphor is generated by the union of these data sets. Hence we can now visualize and compare the different regions sampled in each data set. This is shown in Fig. 11 by highlighting the parts of the landscape visited by the different data sets.

As we can see, the different data sets clearly differ in scope, resolution, and focus. The resulting regions sampled by these three data sets roughly form a nested structure with certain subsets overlapping. It is also interesting to note that the medium-scale REMD simulation is missing the valley next to the main valley *even though it sampled around this valley significantly*: see Fig. 11(c). It could be worth investigating and understanding why this happened, and potentially provide feedback for sampling methods. We note that while these observations are not surprising for this simple test case, one can apply the same principle to comparing data sets produced by different simulation methods (say, two different enhanced sampling methods such as REMD and simulated tempering); and identify where they may differ in terms of sampling completeness and efficiency.

3.3.2. *Secondary structure formation*

Fig. 12 demonstrates how secondary structure formation can be displayed as a colormap on the landscape. To calculate, for example, a scalar value for a conformation C which captures how much its alpha helix structure resembles that of the native conformation $\hat{C}$, we simply locate all alpha helix residues of $\hat{C}$ and count how many of them are labeled as alpha helices in C. This provides a scalar value for each conformation which is displayed as a colormap, revealing how well alpha helices are forming across the landscape. The same technique can be applied to the other secondary structure elements as well.

Using this technique, we can see that 3_{10} helices, shown in Fig. 12(a), and beta sheets, shown in Fig. 12(b), correlate well with the potential energy landscape for gammaS-crystallin. Applying the same technique to the survivin data set reveals an interesting insight into the nature of the bifurcation of the main, deep folding funnel. From Fig. 12(c) it is clear that this bifurcation is related to formation of alpha helices. More specifically, the deeper of the two basins exhibits strong alpha helix formation, whereas the shallower basin does not. Upon closer inspection by looking at the structures from the shallower basin region, it turns out that the alpha helix in the C-terminus fails to form in the shallower basin. Again, observations like these as generated by our software could lead to further investigation and potentially insights about the folding process.

3.3.3. *Contact density*

The contact density of a conformation measures its compactness. It is defined as the percentage of pairwise alpha-carbon atoms whose distances are within a given threshold (typically 6 Å). It is believed to correlate with the folding of a protein, which is confirmed in Fig. 13(a). Since gammaS-crystallin and survivin are composed of two domains (i.e. gammaS-crystallin consists of N-terminal and C-terminal domains; survivin consists of two homogeneous monomers), one can also look at the contact density for each monomer to see whether they fold in a similar manner. From Fig. 13(b) and (c), it turns out that the two monomers behave differently within the small basin next to the major one. Specifically, monomer A becomes compact faster than monomer B. Now recall Fig. 7: by inspecting the conformations along this path, we note that one of the monomers is almost intact while the other one becomes loose in the small basin (valley). So the two monomers behave differently in the folding process based on current simulation data. It would be interesting to investigate to see whether this indicates that monomer A acts as a "seed" to facilitate the formation of monomer B, or that this is simply an artifact from the sampling method (such as insufficient sampling). However, we note that this difference between monomers is consistent with the biological function of survivin in chromosome passenger complex formation as one monomer acts as a spatial anchor and the other monomer dynamically interacts with partner proteins of borealin and INCENP

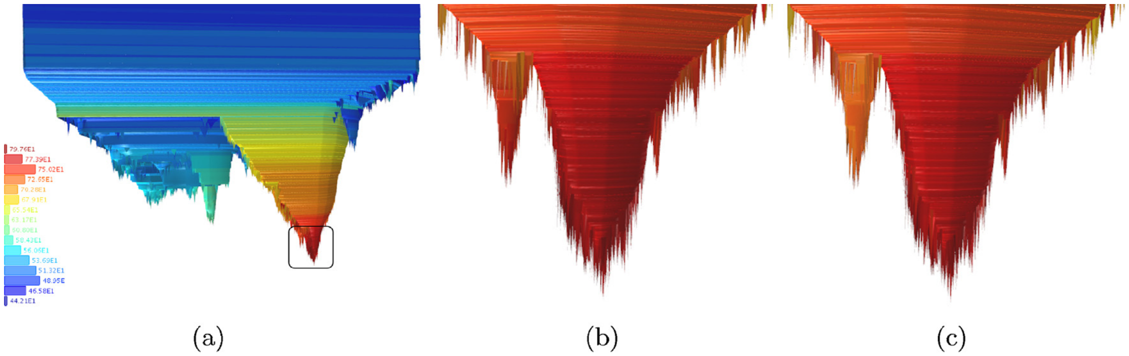


Fig. 13. Contact densities of survivin. (a) The contact density of the entire protein structure correlates well with the folding funnel structure. (b) The contact density of monomer A. (c) The contact density of monomer B. Note that the shallower basin has a noticeably lower contact density for monomer B than for monomer A.

[23]. Nevertheless, this observation could offer another potential point for further investigation, thus facilitating the researchers in forming hypotheses about folding processes.

4. Conclusions

We propose a visual analytics platform for analysis of massive, high-dimensional MD simulation data sets in a collaborative environment. The platform is built upon the idea of the topological landscape as a metaphor for the high-dimensional (potential) energy landscape. This landscape metaphor preserves important topological and geometric properties of the input data. More importantly, it provides a platform for users to explore and navigate the high-dimensional simulation data, allows them to access each individual conformation in the input simulation data set, and enables analysis of data at multiple scales.

The proposed platform is the first software which allows users to interactively explore and navigate through massive molecular simulation data sets. Our software is intuitive and easy to use, allowing users to access and “see” both individual conformations and the entire data set as a whole. The users can record their findings as they explore the data set via annotations and storyboards, and share them with other users in a collaborative environment. In addition to the utility of data exploration and navigation, our software also provides a platform for integrating and visualizing other information overlayed on top of the 2D terrain. We have provided encouraging experimental results to illustrate the utility of our software. We believe that our software can greatly facilitate researchers in analyzing simulation data as well as forming and testing hypotheses.

This work describes a first implementation of the software. The goal of our current experimental results is to demonstrate the utility of our software, and to encourage and stimulate broader usage of our tool in analyzing simulation data. There are several future directions that we will explore. From the software point of view, we will further polish the tool, improve the user interface, and more importantly, add additional utility as we receive feedback from users. We will also investigate how to add kinetic information, crucial for analyzing simulation data, to our terrain metaphor. From the point of view of advancement of biological/biochemical science, we plan to start by focusing on several specific case studies such as TRP-cage and villin headpiece, using the features of the software to aid in searching for novel phenomena and for acquiring new insights. We plan to investigate more complicated folding behavior such as that exhibited by ankyrins (which have experimental folding data) and other dimeric domain proteins.

Acknowledgements

We would like to thank anonymous reviewers for helpful comments. And we thank the Ohio Supercomputer Center for generous computing resources. This work is partially supported by National Science Foundation under projects DBI-0750891 and CCF-1319406.

References

[1] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, P. Walter, Molecular Biology of the Cell, 4th ed., Garland Science, New York, 2002.
[2] D. Attali, A. Lieutier, D. Salinas, Vietoris–Rips complexes also provide topologically correct reconstructions of sampled shapes, Comput. Geom. 46 (4) (2013) 448–465.
[3] C.B. Anfinsen, Principles that govern the folding of protein chains, Science 181 (96) (1973) 223–230.
[4] B. Bienfait, J. Gasteiger, Checking the projection display of multivariate data with colored graphs, J. Mol. Graph. Model. 15 (1997) 203–215.
[5] S.P. Bhat, Crystallins, genes and cataract, Prog. Drug Res. 60 (2003) 205–262.
[6] O.M. Becker, M. Karplus, The topology of multidimensional potential energy surfaces: theory and application to peptide structure and kinetics, J. Chem. Phys. 106 (1997) 1495.
[7] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Comp. 15 (6) (2003) 1373–1396.
[8] J.D. Bryngelson, J.N. Onuchic, N.D. Socci, P.G. Wolynes, Funnels, pathways, and the energy landscape of protein folding: a synthesis, Proteins 21 (3) (1995) 167–195.
[9] F. Chazal, S. Oudot, Towards persistence-based reconstruction in Euclidean spaces, in: ACM Symp. Comput. Geom., 2008, pp. 2–241.
[10] H. Carr, J. Snoeyink, U. Axen, Computing contour trees in all dimensions, Comput. Geom. Theory Appl. 24 (3) (2003) 75–94.
[11] S Dasgupta, Y. Freund, Random projection trees and low dimensional manifolds, in: Proc. 40th Annu. ACM Symp. Theory of Computing, STOC'08, ACM, New York, NY, USA, 2008, pp. 537–546.
[12] D.L. Donoho, C. Grimes, Hessian eigenmaps: locally linear embedding techniques for high-dimensional data, Proc. Natl. Acad. Sci. U. S. A. 100 (10) (2003) 5591–5596.
[13] P. Das, M. Moll, H. Stamati, L.E. Kavraki, C. Clementi, Low-dimensional free-energy landscapes of protein-folding reactions by nonlinear dimensionality reduction, Proc. Natl. Acad. Sci. U. S. A. 103 (26) (2006) 9885–9890.
[14] H. Edelsbrunner, D. Letscher, A. Zomorodian, Topological persistence and simplification, Discrete Comput. Geom. 28 (2002) 511–533.
[15] D.A. Evans, D.J. Wales, Free energy landscapes of model peptides and proteins, J. Chem. Phys. 118 (2003) 3891.
[16] S.L. Flaugh, M.S. Kosinski-Collins, J. King, Contributions of hydrophobic domain interface interactions to the folding and stability of human gammaD-crystallin, Protein Sci. 14 (3) (2005) 569–581.
[17] J.C. Gower, Generalized procrustes analysis, Psychometrika 40 (1) (1975) 33–51.
[18] Y. Hwang, B. Han, A. Hee-Kap, A fast nearest neighbor search algorithm by nonlinear embedding, in: 25th IEEE Conf. Computer Vision Pattern Recog. (CVPR), 2012, pp. 3053–3060.
[19] F.A. Hamprecht, C. Peter, X. Daura, W. Thiel, W.F. van Gunsteren, A strategy for analysis of (molecular) equilibrium simulations: configuration space density estimation, clustering, and visualization, J. Chem. Phys. 114 (2001) 2079.
[20] W. Harvey, Y. Wang, Generating and exploring a collection of topological landscapes for visualization of scalar-valued functions, Comput. Graph. Forum 29 (3) (2010) 993–1002.
[21] Piotr Indyk, Jiri, Matousek, Low-distortion embeddings of finite metric spaces, in: J.E. Goodman, J. O'Rourke (Eds.), Handbook of Discrete and Computational Geometry, CRC Press, 2004, pp. 177–196 (Chapter 8).
[22] J. Jester, Corneal crystallins and the development of cellular transparency, Semin. Cell Dev. Biol. 19 (2) (2008) 82–93.
[23] A. Jeyaprakash, U. Klein, D. Lindner, J. Ebert, E. Nigg, E. Conti, Structure of a Survivin–Borealin–INCENP core complex reveals how chromosomal passengers travel together, Cell 131 (2007) 271–285.

[24] Jmol: An Open-Source JAVA Viewer for Chemical Structures in 3D, 2013 http://www.jmol.org/ (accessed 10.02.13).

[25] M. Karplus, Behind the folding funnel diagram, Nat. Chem. Biol. 7 (7) (2011) 401–404.

[26] S.V. Krivov, M. Karplus, Free energy disconnectivity graphs: application to peptide models, J. Chem. Phys. 117 (2002) 10894, http://dx.doi.org/10.1063/1.1517606.

[27] I.V. Kalgin, M. Karplus, S.F. Chekmarev, Folding of a SH3 domain: standard and hydrodynamic analyses, J. Phys. Chem. B 113 (38) (2009 September) 12759–12772.

[28] W. Kabsch, C Sander, Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features, Biopolymers 22 (12) (1983) 2577–2637.

[29] Y. Levy, O.M. Becker, Effect of conformational constraints on the topography of complex potential energy surfaces, Phys. Rev. Lett. 81 (1998) 1126.

[30] M. Litniewski, Molecular dynamics method for simulating the constant temperature volume and temperature–pressure system, J. Phys. Chem. 97 (15) (1993) 3842–3848.

[31] P.Y. Lum, G. Singh, A. Lehman, T. Ishkanov, M. Vejdemo-Johansson, M. Alagappan, J. Carlsson, G. Carlsson, Extracting insights from the shape of complex data using topology, Sci. Reports 3 (1236) (2013), http://dx.doi.org/10.1038/srep01236.

[32] H. Lei, Z.-X. Wang, C. Wu, Y. Duan, Dual folding pathways of an alpha/beta protein from all-atom ab initio folding simulations, J. Phys. Chem. 131 (16) (2009) 165105.

[33] B. Mahler, K. Doddapaneni, I. Kleckner, C. Yuan, G. Wistow, Z. Wu, Characterization of a transient unfolding intermediate in a core mutant of γs-crystallin, J. Mol. Biol. 405 (3) (2011) 840–850.

[34] B. Ma, R. Nussinov, Energy landscape and dynamics of the hairpin G peptide and its isomers: topology and sequences, Protein Sci. 12 (9) (2003) 1882–1893.

[35] J.T. Macdonald, A.G. Purkiss, M.A. Smith, P. Evans, J.M. Goodfellow, C. Slingsby, Unfolding crystallins: the destabilizing role of a β-hairpin cysteine in βb2-crystallin by simulation and experiment, Protein Sci. 14 (5) (2005) 1282–1292.

[36] A. Mitsutake, Y. Sugita, Y. Okamoto, Generalized-ensemble algorithms for molecular simulations of biopolymers, Biopolymers 60 (2) (2001) 96–123.

[37] M. Nicolau, A.J. Levine, G. Carlsson, Topology based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival, Proc. Natl. Acad. Sci. 108 (2011) 7265–7270.

[38] O. Martin, A. Philippe, C. Vincent, D. Iulian, D. Gilles, E. Burak, M. Sean, M. Stphane, M. Nikolay, S. Michel, S. Lex, S. Erik, Z. Matthias, An Overview of the Scala Programming Language. Technical Report, 2nd ed., EPFL, Lausanne, Switzerland, 2006.

[39] J.N. Onuchic, P.G. Wolynes, Theory of protein folding, Curr. Opin. Struct. Biol. 14 (February (1)) (2004) 70–75.

[40] M.S. Pavlyukov, N.V. Antipova, M.V. Balashova, T.V. Vinogradova, E.P. Kopantzev, M.I. Shakhparonov, Survivin monomer plays an essential role in apoptosis regulation, J. Biol. Chem. 286 (26) (2011) 23296–23307.

[41] I.-H. Park, C. Li, Dynamic ligand-induced-fit simulation via enhanced conformational samplings and ensemble dockings: a survivin example, J. Phys. Chem. B 114 (15) (2010) 5144–5153.

[42] E. Plaku, H. Stamati, C. Clementi, L.E. Kavraki, Fast and reliable analysis of molecular motion using proximity relations and dimensionality reduction, Proteins: Struct. Funct. Bioinf. 67 (4) (2007) 897–907.

[43] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.

[44] D. Sheehy, Linear-size approximations to the Vietoris–Rips filtration, in: ACM Sympos. Comput. Geom., 2012, pp. 9–248.

[45] G. Singh, F. Memoli, and G. Carlsson.

[46] J.E. Stone, K.L. Vandivort, K. Schulten, Immersive out-of-core visualization of large-size and long-timescale molecular dynamics trajectories, in: Proc. 7th Intl. Conf. Advances in Visual Comput. – Part II, ISVC'11, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 1–12.

[47] J.B. Tenenbaum, V. Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.

[48] M. Vendruscolo, E. Paci, C.M. Dobson, M. Karplus, Three key residues form a critical contact network in a protein folding transition state, Nature 409 (February (6820)) (2001) 641–645.

[49] D.J. Wales, Energy Landscapes: Applications to Clusters, Biomolecules and Glasses, Cambridge Univ. Press, Cambridge, UK, 2004.

[50] D.J. Wales, The energy landscape as a unifying theme in molecular science, Philos. Trans. R. Soc. A 363 (2005) 357–377.

[51] G. Weber, P.-T. Bremer, V. Pascucci, Topological landscapes: a terrain metaphor for scientific data, IEEE Trans. Vis. Comput. Graph. 13 (6) (2007) 1416–1423.

[52] W. Zheng, E. Gallicchio, N. Deng, M. Andrec, R.M. Levy, Kinetic network study of the diversity and temperature dependence of Trp-Cage folding pathways: combining transition path theory with stochastic simulations, J. Phys. Chem. B 115 (6) (2011) 1512–1523.

[53] R. Zhou, Trp-Cage: folding free energy landscape in explicit water, Proc. Natl. Acad. Sci. 100 (2003) 13280–13285.

[54] R. Zhou, L. Parida, K. Kapila, S. Mudur, PROTERAN: animated terrain evolution for visual analysis of patterns in protein folding trajectory, Bioinformatics 23 (1) (2007) 99–106.