

# Dynamic Channels in Biomolecular Systems: Path Analysis and Visualization

Norbert Lindow\*  
 Dept. of Visualization  
 and Data Analysis,  
 Zuse Institute Berlin (ZIB)

Daniel Baum†  
 Dept. of Visualization  
 and Data Analysis,  
 Zuse Institute Berlin (ZIB)

Ana-Nicoleta Bondar‡  
 Theoretical Molecular Biophysics,  
 Dept. of Physics, FU Berlin§  
 Dept. of Physiology and Biophysics,  
 University of California, Irvine

Hans-Christian Hege¶  
 Dept. of Visualization  
 and Data Analysis,  
 Zuse Institute Berlin (ZIB)

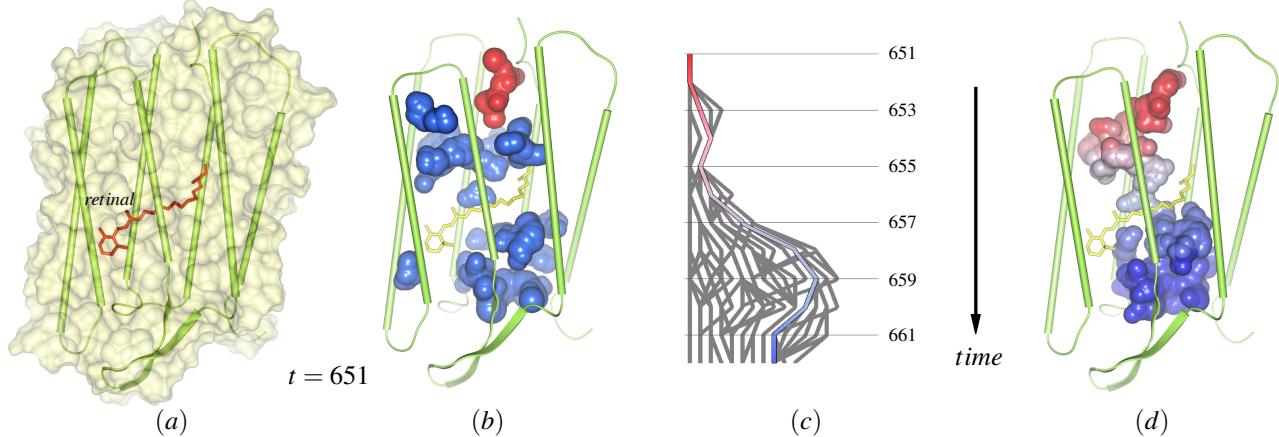


Figure 1: The cavity structure in bacteriorhodopsin. (a) A bacteriorhodopsin monomer displayed as surface and cartoons, with the retinal, shown as bonds, covalently bound to Lys216. (b) All cavities of the monomer as found at time step 651; the cavity shown in red is traced over time in panels (c-d). (c) Split and merge graph of path tracing. (d) A dynamic molecular path, connecting the cytoplasmic (top) and the extracellular side (bottom), based on the selection in the split and merge graph.

## ABSTRACT

Analysis of protein dynamics suggests that internal cavities and channels can be rather dynamic structures. Here, we present a Voronoi-based algorithm to extract the geometry and the dynamics of cavities and channels from a molecular dynamics trajectory. The algorithm requires a pre-processing step in which the Voronoi diagram of the van der Waals spheres is used to calculate the cavity structure for each coordinate set of the trajectory. In the next step, we interactively compute dynamic channels by analyzing the time evolution of components of the cavity structure. Tracing of the cavity dynamics is supported by timeline visualization tools that allow the user to select specific components of the cavity structures for detailed exploration. All visualization methods are interactive and enable the user to animate the time-dependent molecular structure together with its cavity structure. To facilitate a comprehensive overview of the dynamics of a channel, we have also developed a visualization technique that renders a dynamic channel in a single image and color-codes time on its extension surface. We illustrate the usefulness of our tools by inspecting the structure and dynamics of internal cavities in the bacteriorhodopsin proton pump.

**Index Terms:** J.3 [Computer Applications]: Life and Medical Sciences—Biology and Genetics; I.3.5 [Computer Graphics]: Comput. Geom. & Object Modeling—Boundary Representations.

\*e-mail: norbert.lindow@zib.de

†e-mail:baum@zib.de

‡e-mail:nbondar@zedat.fu-berlin.de

§present address

¶e-mail:hege@zib.de

## 1 INTRODUCTION

Molecular transporters couple the transfer of the substrate with protein conformational changes that may result in rearrangements of the internal cavity structure. All-atom molecular dynamics simulations, in which the classical time-dependent equations describing the motion of all atoms in the system are integrated numerically, allow us to investigate protein dynamics at atomic detail. A molecular dynamics simulation yields a *trajectory*: the coordinates of all atoms in the system as a function of time. The number of coordinate sets in the trajectory is determined by the integration time-step (for example, 1 femtosecond), and the length of the simulation (tens or hundreds of nanoseconds, or even microseconds and milliseconds, see Dror et al. [12]).

There are numerous tools to inspect and analyze molecular dynamics trajectories, including [43, 48, 56, 61]. These tools offer various representations of the molecular structure, for example, ball-and-stick models and protein secondary structures, and they also allow the user to analyze molecular structures in various ways. It is also possible to interactively view the dynamics of the molecular surface of macromolecules [31, 38, 33]. To depict metastabilities and conformational changes in biomolecules, specific techniques have been developed, including [53, 52, 7].

Here, we present a tool that enables interactive analysis of dynamic molecular paths in biomolecules and that is specifically designed for assessment of conformational changes in molecular transporters. We denote as *molecular path* (or *path*) a geometric curve in three-dimensional space, consisting of points whose distance to the van der Waals surface is maximal and does not fall below a given minimal value. This value approximates the radius of an ion or substrate that is transported within the molecule. The tool allows the user to analyze the evolution of paths over time associated to dynamic channels or cavities.

The starting point of our approach was presented in [37], where

Voronoi diagrams of spheres were used to compute geometry-based molecular paths of a static coordinate snapshot. We extend this method to molecular dynamics trajectories by computing time-dependent molecular paths. First, we compute molecular paths for all time steps; in the next step, the user can explore interactively specific paths and their spatiotemporal evolution. To support such an interactive exploration, we developed visualization tools that enable the user to easily identify segments of the trajectory where potentially relevant structural changes occur.

We demonstrate the applicability of our methods using a molecular dynamics trajectory of bacteriorhodopsin embedded in a hydrated lipid membrane. Bacteriorhodopsin is a light-driven proton pump in which proton translocation is coupled to protein structural rearrangements and relocation of internal water molecules (see, for example, [22, 1]). Because access of water and protons from the two sides of the lipid membrane is tightly regulated during vectorial ion transport, understanding the location, geometry (volume/spatial extent), and the dynamics of cavities large enough to host water is an important aspect of investigating conformation-coupled proton transport.

## 1.1 Bacteriorhodopsin Dynamics Trajectory

The coordinate snapshots for our analysis were taken from a molecular dynamics simulation of a bacteriorhodopsin trimer embedded in a hydrated 1-palmytoyl-2-oleoyl-sn-glycero-3-phosphatidylcholine (POPC) lipid membrane; the simulation system contained approximately 180,000 atoms. For the starting protein coordinates, we used the crystal structure from Luecke et al. [39]; the missing loop in the crystal structure was constructed using the crystal structure from Belrhali et al. [2]. The computations were performed using the NAMD software [28, 47] with the CHARMM [6] parameters for the protein [40] and lipid atoms [17], and the TIP3P model for the water molecules [26]. The retinal was treated as described in Gruia et al. [21], based on the parameters from the publications by Nina et al. [44] and Tajkhorshid et al. [57]. We used SHAKE [51] to constrain the length of bonds that involve hydrogen atoms, and a switching function between 8 and 12 Å to cut off the short range real space interactions; to compute the Coulomb interactions, we used the smooth particle mesh Ewald summation [11, 15]. The simulations were performed using Langevin dynamics with a Nose-Hoover piston to maintain the pressure at 1 bar and the temperature at 300K [18, 41]. During the molecular dynamics simulation, coordinate sets were saved every 1 ps; following the initial equilibration of the system, we used the reversible multiple time-step algorithm [19, 59] with time steps of 1fs, 2fs, and 4fs for the bonded forces, the short-range non-bonded forces, and the long-range electrostatic forces, respectively. The analysis here is based on 2000 equally spaced coordinate snapshots of one bacteriorhodopsin monomer from the last 20 ns of a 48ns trajectory.

## 2 RELATED WORK

Most approaches to identify cavities and channels in molecular structures place spheres of constant or varying size, or even water molecules, around the molecule. These sphere-placing algorithms can be categorized into methods that (a) use a grid data structure [36, 24, 23], (b) directly use the atoms of the surface [34, 27], or (c) use a different strategy, for example, a Monte-Carlo method [58], or simulated annealing [55]. While some of these techniques directly use the number and position of the spheres to describe the cavities and channels, other approaches [30, 34, 23] compute a surface representation from the positioned spheres.

Instead of placing spheres, some methods use voxel representations to describe molecular cavities and channels [16, 49, 33]. The latter two tools are both able to handle dynamic molecular data. While in [49], residence probabilities are used to include the dynamics of the molecule, in [33], voxels are traced interactively over

the trajectory starting from a selected enclosed cavity which is extended over time until the outside of the molecule is reached.

Another group of methods computes molecular paths - similar to ours - inside cavities and channels [46, 45, 42, 63, 37]. These paths are computed using a grid data structure on which Dijkstra's shortest path algorithm is run [46], or by applying the idea of the Voronoi diagram or its dual, the Delaunay triangulation. While using a grid data structure as well as the classical Voronoi diagram [45, 42, 63] results in an approximation, the use of the Voronoi diagram of spheres represents an exact path calculation [37].

Not all methods that compute molecular cavities and channels can be assigned to either of the groups mentioned above. One example is a technique using the alpha-shape theory [14] to identify cavities [3]. Others rely on the solvent excluded surface (SES) [50] and compute either the volume difference for different probe sphere radii [64, 62] or the distance map to the outside of the molecule starting from all points on the triangulated SES [9]. From this distance map, paths leading to the outside of the molecule can then be computed. To identify molecular channels through transmembrane proteins, a heuristic, iterative approach specifically developed for this kind of data was described in [8]. Another approach was followed by Cortes et al. [10], who applied rapidly-exploring random trees [35] to detect paths depending on the geometry and dynamics of a given substrate.

Of the methods described above, only the approaches [49, 33] can be used to analyze cavities and channels in time-dependent data. Here, we present a new method that extends the Voronoi-based approach [37] to compute molecular paths to dynamic data. In this extension, we compute all molecular paths for each time step of the molecular dynamics trajectory and compute dynamic paths, that is, paths evolving over time. Unlike the methods from [49, 33], our method does not depend on the resolution of a grid data structure: the paths computed in each time step are represented by an analytic description of the geometry as well as their extensions. Furthermore, a comprehensive overview on the spatiotemporal behavior of all cavities is provided.

## 3 DYNAMIC MOLECULAR PATHS

The computation of dynamic molecular paths consists of two stages. In the first stage, we compute the molecular paths for each time step separately in a pre-processing step. For this purpose, we apply the techniques described in [37] and summarized in Section 3.2. In the second stage, we interactively trace paths over time and identify splits and merges of path components. This is described in Section 3.3.

### 3.1 What is a Molecular Path?

Before we go into details about the computation of molecular paths, we want to clarify what exactly we mean by the term ‘molecular path’. Our definition is based on the geometry of the empty space around the atoms. Throughout this paper, a static molecular path is a continuous curve in three-dimensional space, whose distance to the van der Waals surface is as large as possible and does not fall below a given minimal value  $r_p$  – the radius of the sphere that we call *probe sphere*. In mathematical terms, a static molecular path is part of the topological skeleton of the distance function of the van der Waals spheres. It consists of maxima, index-2-saddles, and the connecting separatrices. The *extension surface* of each path is the surface that encloses the empty space around the path. An *extension volume* is a volume enclosed by an extension surface. A maximal set of extension volumes whose union is path-connected, forms a *cavity* or a *channel*. We call the network of paths corresponding to a single cavity, *path component* or *component*.

A dynamic molecular path is a collection of static molecular paths that get connected over time. To form a dynamic path, the

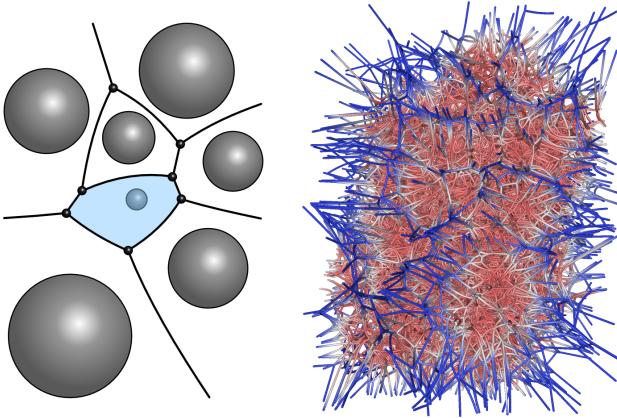


Figure 2: The left image shows an illustration of a two-dimensional Voronoi diagram; one Voronoi region is highlighted in blue. The image of the right side shows the Voronoi diagram computed for a coordinate snapshot of bacteriorhodopsin, cut with the bounding box of the protein. The color indicates the distance to the protein, ranging red to blue for distances close to and remote from protein atoms, respectively.

extension volumes of the static molecular paths of consecutive time steps must have enough geometrical overlap (see Section 3.3).

### 3.2 Computation of Static Paths

The complete topology of the distance function is described by the Voronoi diagram of the van der Waals spheres of the atoms. Consider a molecule with  $n$  atoms whose positions are  $p_i \in \mathbb{R}^3$  and whose van der Waals radii are  $r_i \in \mathbb{R}$ , with  $i = 1..n$ . The three-dimensional Voronoi diagram of spheres consists of regions, facets, edges, and vertices. The Voronoi region  $V_i$  of atom  $i$  is the set of all points  $p \in \mathbb{R}^3$  that are at least as close to the corresponding sphere of  $i$  as to any other sphere. It is defined by

$$V_i = \{p \mid \|p - p_i\| - r_i \leq \|p - p_j\| - r_j, i \neq j\}.$$

The facets, edges, and vertices are two-, one-, and zero-dimensional non-empty intersections of the Voronoi regions. Note that we assume that all atoms lie in general position, that is, each Voronoi vertex lies at the intersection of exactly four regions. If this is not the case, the algorithm for the computation of the diagram might fail. To achieve this condition, we randomly perturb the atom positions. The perturbations are chosen small enough so that the changes in the result are irrelevant for the path and cavity analysis. A two-dimensional illustration of a Voronoi diagram and the three-dimensional Voronoi diagram of spheres of one time step of bacteriorhodopsin are depicted in Figure 2.

The set of all maxima of the distance function is a subset of the Voronoi vertices. The index-2-saddles and the separatrices are part of the Voronoi edges. Thus, for our analysis, we will use only the Voronoi graph containing the edges and vertices. We compute the graph using the edge-tracing algorithm by Kim et al. [29], which starts from a single vertex and traces along its four possible edges. For each edge, the end vertex is detected and the tracing is continued along its three new edges. If the three intersecting Voronoi regions are not closed, the edge has no end vertex. In this case, we introduce a synthetic end vertex with no further edges and mark it as an ‘infinity vertex’. A detailed description of the algorithm and its optimizations can be found in reference [37].

The Voronoi graph contains many edges and vertices that are not part of molecular paths, because their distance to the atoms is below  $r_p$ , or because they lie outside the domain of the molecule. For

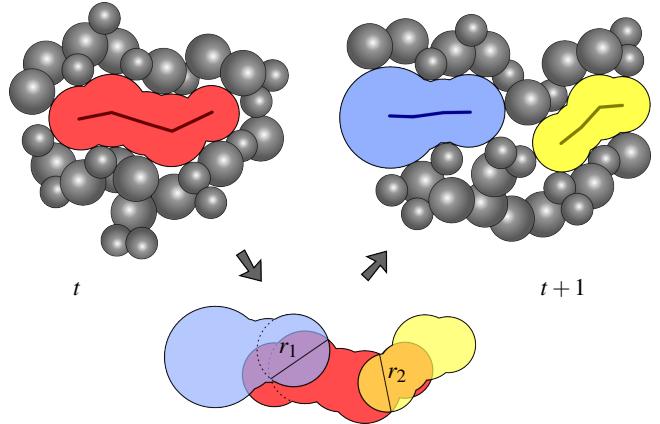


Figure 3: Illustration of the path tracing algorithm in 2D. The red component of time step  $t$  intersects the two components of time step  $t + 1$  with maximal radii  $r_1$  and  $r_2$  of the intersection circles. Here both radii are large enough (i.e., larger than the user-defined minimal radius of the intersection sphere) such that the tracing detects a split event.

this reason, we apply two main filtering steps as follows. In the first step, the graph is cut with the axis-aligned bounding box of the molecule and the vertices are filtered by their ambient occlusion values according to a user-defined threshold. Vertices that receive significant ambient light are far away from the molecule and, thus, are considered to be outside the domain of the molecule. These vertices are discarded from the subsequent path analysis. In contrast, vertices that are close to molecule or inside cavities receive less ambient light and are kept for the analysis. In the second step of the filtering scheme, we remove all edges whose minimal distances to the atoms are smaller than the user-defined value for  $r_p$ . Note that this value represents the radius of the probe sphere which is able to move along the paths without colliding with any atom. For a detailed description of all filters that can be applied, see ref. [37].

### 3.3 Tracing of Dynamic Paths

After all static paths in each time step have been computed, the user can trace the evolution of path components interactively over time. Each component consists of a set of paths inside a cavity or tunnel. We define the extension surface of a path as follows: Each path consists of a set of Voronoi vertices, where each vertex defines an empty sphere tangent to the four atom spheres whose Voronoi regions created the vertex. The outer surface of all empty spheres of a path defines its extension surface. Respectively, the surface of a component is the outer surface of all empty spheres of its paths.

To monitor the time evolution of components, the user can manually select an arbitrary time step and one or more connected components. If the user proceeds to the next time step, the selected components will be traced by detecting all components of the graph whose sets of empty spheres intersect the empty spheres of the previous time step by an intersection circle that has a larger radius than a user-defined minimal intersection radius  $r_{ic}$ . This is illustrated in Figure 3. A good choice for the minimal radius  $r_{ic}$  of the intersection circle is the radius  $r_p$  of the probe sphere that was used for the path filtering, or a slightly smaller value. The procedure for tracing the path components is detailed below.

For each empty sphere  $s$  of each connected component of time step  $t$ , we detect all empty spheres of time step  $t + 1$  that intersect the previous sphere with an intersection circle whose radius is larger than  $r_{ic}$ . If a sphere in time step  $t + 1$  fulfills this condition, the whole connected component of the corresponding vertex is selected. Note that we define an intersection circle of two spheres as the largest circle of the smaller sphere inside the intersection vol-

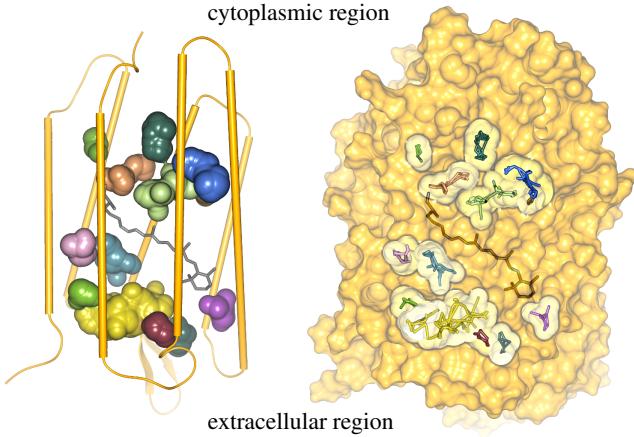


Figure 4: Visualization of all molecular path components of a single time step. The image on the left side depicts the path extensions rendered as skin surface, combined with the secondary structure of the molecule. Each color indicates a different path component. The panel on the right depicts the molecular surface, clipped by the path extensions.

ume of both spheres. That is, if the larger part of the smaller sphere is inside the larger sphere, we define the intersection circle to be the radius of the smaller sphere.

While tracing the connected components, an undirected ‘time graph’ is created. Each vertex in the graph represents a connected component at a certain time step. Two vertices of time steps  $t$  and  $t + 1$  are connected if the corresponding cavity of step  $t$  is traced to the cavity of step  $t + 1$ . Thus, the graph keeps track of splits and merges of components.

In order to support the visualizations that will be described in Section 4.3, each component is assigned an identification number that can be used, for example, to color the paths (see example in Figure 4). To enable the user to easily trace the components over time, we compute these identification numbers as follows. Consider a tracing step from time step  $t$  to  $t + 1$ . Note that the identification numbers of the components of  $t$  are already set and that we initialize the components of  $t + 1$  with an invalid identification number. First, we sort all components of  $t$  by their volumes, beginning with the largest one. To avoid an expensive volume computation, we approximate the volumes by summing up the volumes of all empty spheres of the component. For each sorted component  $c_t$ , we detect the largest component  $c_{t+1}$  of  $t + 1$  that is connected to  $c$  in the time graph and has an invalid identification number. If there is no such component, we take the next component of  $t$ , otherwise  $c_{t+1}$  gets the same identification number as  $c$ . In the final step, we set the unused identification numbers of all remaining components of  $t + 1$ , so that each component in a time step gets a unique number.

An additional important feature of our path tracing is that path components representing dead ends can be identified and removed. A dead end component has no edge in the graph to a component in the following time step. We trace these components back in time until we reach a splitting in the graph and remove the dead ends. This step noticeably simplifies the search for dynamic channels.

## 4 VISUALIZATION SYSTEM

### 4.1 Molecular Representations

The visualization system we developed offers several options for the display of the molecular structures. For example, the user can switch between ball-and-stick representation, different molecular surfaces, and secondary structure views. As surface representations, we provide the van der Waals surface, the solvent accessible

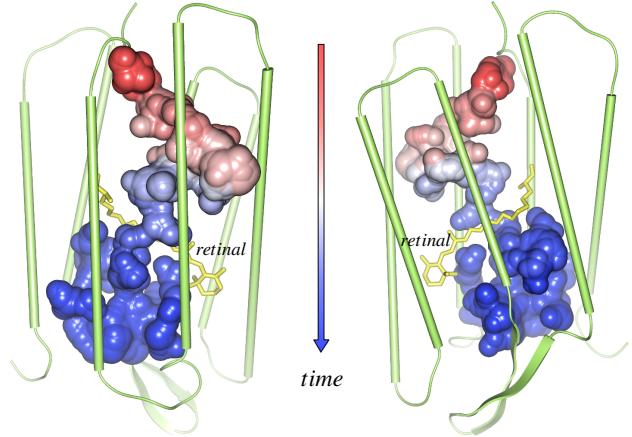


Figure 5: Visualization of a dynamic molecular channel. The surface shows the accumulated components of one path component traced over time. The time is shown by pseudo-coloring the extension surface. Here, red represents an early time and blue a later time. The time is linearly mapped to the color map.

surface (SAS), the solvent excluded surface (SES), and the molecular skin surface (MSS). The surface computations and rendering techniques are based on [38], which makes them suitable for dynamic molecular data. Furthermore, the user can add advanced lighting techniques such as screen space ambient occlusion or path lighting (for details see ref. [37]). For all molecular representations, attributes can be visualized using colors. These attributes can represent atom properties - for example atom types or electrostatic potentials -, or types of amino acids and side chains. To focus on specific regions of the molecule, the user can apply filters to hide parts of the molecule.

### 4.2 Molecular Paths

#### 4.2.1 Static Paths

For visualizing static molecular paths, we provide two graphical representations. The first is a three-dimensional depiction of the filtered Voronoi diagram. In this illustration, for each vertex we render a small sphere, and for each edge that connects two vertices, we render a cylinder. The rendering of this graph is based on the raycasting techniques by Sigg et al. [54]. The second representation depicts a smoothed version of the extension surfaces. For this, we use the skin surface of the empty spheres of the Voronoi vertices as described in [37]. The skin surface is a smooth surface defined by a set of spheres. It depends on a single parameter with values between 0 and 1, called the shrink factor. For shrink factors from 0 to 1, the surface changes from the convex hull of the spheres to the surface enclosing exactly the spheres. For more information about the skin surface and its fast rendering, we refer the reader to [13, 38]. As already mentioned, we use the identification number of the path components to color them. This allows the user to easily trace path components over time. An example of the skin surface visualization can be seen in the left image of Figure 4. An image of the graph visualization in combination with the SES of the molecule clipped by the skin surface is shown in the right image of Figure 4.

#### 4.2.2 Dynamic Paths

Dynamic paths consist of a possibly large number of components that are accumulated over time. Our skin surface approach offers an easy and fast way to render such a dynamic path. To do so, we simply use all spheres of all path components belonging to the dynamic path as input to the skin surface computation. The result is

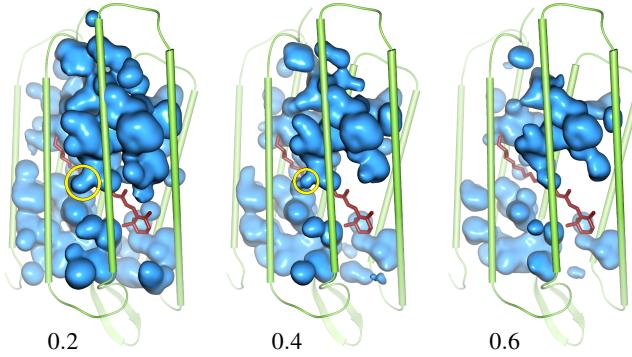


Figure 6: Secondary structure of bacteriorhodopsin and iso-surfaces of the spatial path probability density, i.e., the proportion of time in which a point is inside a cavity. The yellow circles highlight the only connection between the components in the top and bottom halves.

a smooth surface representing the extension surface of all components of the dynamic molecular path. This visualization represents a static view of a dynamic process. To add a dynamic component to this static view, the skin surface can be color-coded by the time of penetration along the path (see example in Figure 5).

#### 4.2.3 Probability of Paths

Dynamic paths are important to identify possible pathways that can be further studied. However, they represent a rather local view on the dynamics of the molecular paths w.r.t. the whole molecular dynamics trajectory. A more global representation of the path dynamics is given by the residence probabilities of all components. For a given time interval, the ‘residence probability’ of a point is the proportion of time in which the point is inside a cavity. These probabilities can be easily computed from our path components by accumulating the residence information of the empty spheres of all components over all time steps. The residence information can be sampled on a regular grid and visualized using volume rendering or iso-surfaces. Images for different iso-values for bacteriorhodopsin are shown in Figure 6. Note that the residence probabilities that we compute are similar to the ones described by Raunest and Kandt [49]. It is also related to the work of Jardón-Valadez et al. [25], in which water molecules are accumulated over time to generate a density which is visualized with an iso-surface.

### 4.3 Path Timeline

To derive an overview of the time evolution of specific paths, we augment the graphical representations of the protein and molecular paths by timeline visualizations. These comprise information from several time steps in a single image. The user can interactively change the range of the time steps displayed and thereby determine how global the view of the events is. In what follows, we present two different possibilities to render the path timeline (see Figure 7).

#### 4.3.1 Split and Merge Graph

The first visualization shows topological changes of the selected components, that is, splits and merges. Each component which is traced over time is depicted as a single polyline rendered from left to right. Time steps are depicted by vertical lines. An example of a split and merge graph is shown in Figure 7, top. When components merge, intersections of polylines can occur. Note that we do not optimize the graph layout to reduce the number of intersections, but simply render the path components and their connections from top to bottom. In the current implementation of the tool, we render all polylines with constant thickness; it could be easily extended to encode the size of the components as thickness of the

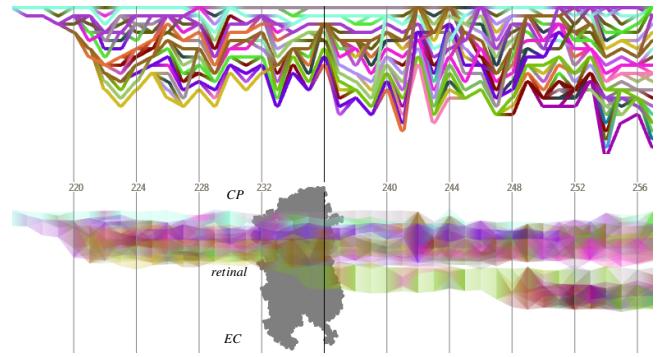


Figure 7: Path timeline visualizations. On top, the split and merge graph displays topological events of the path components. Below, the penetration graph displays the evolution of the position and the extension of the path components. The numbers give the time steps of the trajectory segment used here. The connection between the cytoplasmic (CP) and extracellular (EC) components of the path at time step 232 represents temporary merging of two geometrical paths, and not the formation of a water-filled channel. The gray sketch of the molecule facilitates an easy grasp of the penetration depth.

lines. If a component splits into several components, we identify the largest one which keeps the color of the component that is split. The same protocol is used for merges: if several components merge, the merged component is assigned the color of the largest previous component that merged. To enforce the identification of component correspondences between the timeline visualization and the three-dimensional visualization, the same component colors are used in both representations.

#### 4.3.2 Penetration Graph

The second graph displays the extensions of all selected components along a user-defined direction. A reasonable direction is the direction of the penetration of the cavities. Such penetration graphs are particularly interesting for membrane proteins, because the molecular paths of membrane proteins tend to be oriented along a single direction – the membrane normal. As in the split and merge graph, the time steps are displayed as vertical lines and the components are rendered as lines from left to right. But in the penetration graphs, the line for each component has a position and a thickness according to the position and extension of the path component in the major direction of penetration. Since in this view the lines of components overlap, we draw the lines in transparent mode and blend them. To help the user to easily understand the position and extension of a component along the major direction of penetration, a simplified 2D illustration of the protein is rendered in the background of the graph. In Figure 7, bottom, a penetration graph for bacteriorhodopsin is shown. The direction was chosen from the cytoplasm side (top) to the extracellular side (bottom). We traced a dynamic channel from a single selected path component. At the beginning, only the components in the upper part of the molecule get interconnected. Then, at time step 232, the components above the retinal get connected with components below the retinal which finally leads to the creation of a dynamic channel.

## 5 INTERACTIVE PATH EXPLORATION

Once the molecular paths for each time step have been computed, the user can interactively explore the paths of the molecular system. Our tool provides several ways to support interactive exploration. In this section, we describe these possibilities.

## 5.1 Animation

Since all our visualization methods are interactive, the first possibility is to animate the molecular dynamics trajectory with both molecule and the molecular paths representations as described in Section 4. This enables a visual analysis of the changes of the molecular structure as well as the changes of the size and shape of the path components. The coloration of the path components (Section 4.2.1) helps to visually trace the path components over time. Nevertheless, it remains a tedious task to identify all the components over time that together yield a dynamic path. As animations are limited in their ability to convey information [60], we rely more on interactive selection of time steps and cumulative depictions.

## 5.2 Tracing of Selected Components

To ease the identification of the components that contribute to a specific dynamic path, we include the option to select one or more components in a single time step. The selected components are then visualized using the skin surface; furthermore, they can be animated over time. Here, animation of the components means that we recursively trace all components into which the selected components split. For example, if the user selected a single component, the tool traces this component over time. At some point, the component might split into two components. Both of these components are then further traced. In the course of tracing, the components can further split or they can merge again.

## 5.3 Selection of Dynamic Paths

To enable a more compact representation of the path dynamics, our tool offers a further possibility to explore the computed molecular paths. The result will be a single visualization showing a dynamic path as described in Section 4.2.2. Such a dynamic path can be interactively selected by the user in the split and merge graph. The selection can be guided by the penetration graph, which gives the user a quick overview about the progression of penetration along the selected direction. Thus, interesting events along the timeline can be easily identified by the user. If an interesting time region has been identified, the user can select a component in the split and merge graph. After the selection of the start component, the user can proceed to subsequent time steps and select a component there. This is repeated until the final component is reached. The result is a single dynamic path in the split and merge graph. All the components along this path can now be used to compute the extension surface of this dynamic path. An example of a dynamic path is shown in Figure 5.

Note that we currently compute the path between two selected components with a modified depth first search on the split and merge graph. However, to force the selection of another path between two selected components, intermediate selections can be added, thus giving all the flexibility that is needed.

## 6 RESULTS

### 6.1 Parameter Settings

The molecular path computation and tracing as well as the visualization methods require a set of parameters. The parameters used for the results given in this section are described below.

The path computation depends on two parameters for the filtering: the ambient occlusion threshold, and the radius of the probe sphere. The ambient occlusion threshold is used to filter the vertices that are outside the domain of the molecule [37]. This parameter depends significantly on the protein and on the position of the paths within the protein. Since for bacteriorhodopsin we are mainly interested in paths deep inside the molecule, we set this parameter to 15%. Hence, all vertices that receive more than 15% of the overall possible ambient light are removed from the path computation. The radius of the probe sphere that can move along the paths depends on the aim of the path analysis. For example, in the case of a

biomolecule such as bacteriorhodopsin, the user may want to identify cavities that are large enough to host a water molecule. In the examples discussed here, we used a radius of 1.4 Å for tracing the path components.

The main parameter for the visualization is the shrink factor of the skin surface that is used for rendering the extension surface of the paths. This parameter must be large enough, otherwise the skin surface might intersect the van der Waals surface. We used a shrink factor of 0.7, which creates smooth path surfaces and is very close to the correct empty space around the paths.

## 6.2 Performance

For detailed computation and filtering timings of the molecular paths as well as rendering timings for the visualizations of these paths and the molecular structure, we refer the reader to [37]. In this section, we concentrate on the computation and visualization of the paths of a trajectory of one monomer of bacteriorhodopsin with 2000 time steps. The precomputation of the paths took 44 min on an Intel Xeon 5650 with 2.67 GHz. Thus, the average computation time for one time step was approximately 1.32 s.

During visualization, we obtained worst case frame rates when rendering the solvent excluded surface of the molecule clipped by the extension surface of the paths. For this setting, we reached frame rates between 40 and 50 fps including the path lightening of [37] and screen space ambient occlusion with depth cueing. For all other renderings, the frame rates were even higher. Note that for these performance measures we used a screen resolution of  $1024 \times 1024$  and visualized the whole protein with an average fill rate of 75%. The rendering was done for a single time step with an NVIDIA Geforce GTX 470 graphics card. If we proceed to another time step, the frame rate decreases due to the recomputation of the molecular surface and the recomputation of the extension surface of the paths. Nevertheless, the frame rate was still approximately 25 fps. The path tracing itself is so fast that it has nearly no influence on the performance.

## 6.3 Paths in the Bacteriorhodopsin Proton Pump

We illustrate the usefulness of the new tool presented in this paper by inspecting a molecular dynamics trajectory of bacteriorhodopsin. Molecular dynamics (Grudinin et al. [20]) and reaction path computations (Bondar et al. [5]) indicated that there is a structural barrier for the passage of water molecules from the cytoplasmic to the extracellular half of the protein: No water molecules diffuse across the retinal Schiff base region on the nanosecond-timescale molecular dynamics simulations [20], which may be explained by the high energy barriers (10–15 kcal/mol) associated with water translocation events [5]. The analysis of static paths, that is, the path components for a given time step, provides a clear illustration of the structural barrier in the center of the protein where retinal is located. The paths from the cytoplasmic half of the protein are not connected to those on the extracellular side (see Figure 4). But in the two halves of the protein there are several distinct paths, and each path is dynamical. Since the dynamics of the paths is governed by the motions of the protein groups that delineate the path, by following the time evolution of the individual path components we can explore the motions of the internal protein cavities (see Figure 5). The apparent connectivity between the cytoplasmic and extracellular regions of the protein is a result of the residence information of all path components over all time steps (see Figure 6).

## 7 CONCLUSION AND DISCUSSION

We presented a tool for the analysis and visualization of geometric molecular paths along molecular dynamics trajectories. This tool enables the user to interactively identify and trace the time evolution of channels and cavities in a biomolecule. Moreover, cavities

of consecutive time steps can be combined into a single view to investigate, for example, dynamic molecular channels. To the best of our knowledge, there exists only one other approach that allows interactive tracing of cavities [32]. Albeit it requires a pre-processing step, our method has the advantage to represent paths accurately and to provide advanced molecular visualizations.

It is important to stress that the molecular paths we compute are pure geometrical connectivities that depend on the choice of the probe radius, and that the occupancy of the cavities by water molecules is not considered in the analysis of the channels discussed here. That is, the time evolution of the path depicted in Figure 5 is a representation of the dynamics of the protein leading to changes in the geometrical path. The tool presented here comprises methods for rapid assessment of the location, shape, and dynamics of the internal cavities of a protein, and for identifying branching points of the geometrical paths. Efficient identification of multiple paths and their branching points may, for example, be important for large proteins of complex architecture such as the Photosystem II, where numerous interconnected (static) hydrogen-bonded networks were identified recently [4]. The method can also be easily extended to compute the volumes of internal protein cavities, and to monitor the changes in the volume of the cavities along the molecular dynamics trajectory.

To visualize cavities and channels, we render the extension surface of the paths using the skin surface of the empty spheres along the paths. We believe that the skin surface is well suited for this application for several reasons. First, it creates a smooth visualization of the paths which is very close to the correct geometry of the empty space given by the Voronoi diagram. Second, the surface itself is closed which allows computing the volume of cavities or channels. And third, the computation and visualization via raycasting is well understood [38]. Pursuant to these considerations, the skin surface can be rendered at interactive frame rates even for dynamic data.

The accuracy of the geometric path tracing largely depends on the density of empty spheres along the paths, and on the time resolution of the coordinate sets in the molecular dynamics trajectory. Regarding the density of the empty spheres along the paths, in the current representation, we use only the Voronoi vertices. If two vertices are too far away from each other and the best intersection sphere of the succeeding time step lies between them, the tracing could fail. In practice, this is not a problem, since the paths are generally sampled very densely. Moreover, we could easily add more vertices along the edges that are not sampled densely enough. Such an adaptive sampling could be achieved by comparing the intersection circle of two neighboring vertices and the minimal distance of the edge to the atom spheres. In what the resolution of the time of the trajectory is concerned, we note that the probability of a failure in path tracing increases significantly with a low time resolution (that is, at large time intervals between two consecutive coordinate sets). The resolution used here (10 ps) appears sufficient, although finer resolutions would give even smoother dynamic channels.

We used the approximated volumes of the path components to transfer the component identification numbers between consecutive time steps (Section 3.3). Tracing of the component identification numbers could be improved by using the correct intersection volume of the path components instead of approximated volumes.

The time line graph visualizations help to identify events like merges and splits of the path components and show the progression of the penetration. Although we allow the user to remove dead ends, these visualizations can become confusing if many components are shown. Therefore, we plan to investigate several possibilities to simplify the visualizations. One possibility is to optimize the graph layout of the split and merge graph by minimizing the number of intersections. Another possibility is to apply clustering algorithms to the graph. The main challenge here would be the definition of reasonable clustering criteria. Finally, we want to improve the man-

ual filtering and map on the cavities the time-dependent location of internal water molecules. For this, it is important to provide easy semi-automatic selection and interaction possibilities.

Apart from the minor limitations noted above, we think that the tool is valuable for analyzing molecular dynamics trajectories of biomolecules. To the best of our knowledge, it is the first tool that can identify dynamic molecular paths with a given minimum constriction size. We illustrated the usefulness of the tool using a segment of a molecular dynamics simulation of the bacteriorhodopsin proton pump. In the future, we plan to analyze more data sets and adjust the tool to specific needs. For example, it would be important to quantify the volume and the length of cavities inside proteins, and how these geometrical parameters change along a molecular dynamics trajectory.

## ACKNOWLEDGEMENTS

Ana-Nicoleta Bondar acknowledges financial support from the Marie Curie International Reintegration Award IRG-276920. A set of preliminary molecular dynamics simulations were performed using the University of California, Irvine High-Performance Beowulf Cluster (grant acknowledgment: National Institutes of General Medical Sciences GM-74637 and GM-086685).

## REFERENCES

- [1] M. Andersson, E. Malmerberg, S. Westenhoff, and G. Katona et al. Structural dynamics of light-driven proton pumps original research article structure. *Structure*, 17(9):1265–1275, 2009.
- [2] H. Belrhali, P. Nollert, and A. Royant et al. Protein, lipid and water organization in bacteriorhodopsin crystals: a molecular view of the purple membrane at 1.9 Å resolution. *Structure*, 7(8):909–917, 1999.
- [3] T. Binkowski, S. Naghibzadeh, and J. Liang. CASTp: Computed atlas of surface topography of proteins. *Nucleic Acids Res.*, 31(13):3352–5, 2003.
- [4] A.-N. Bondar and H. Dau. Extended protein/water h-bond networks in photosynthetic water oxidation. *BBA-Bioenergetics*, 2012.
- [5] A.-N. Bondar, S. Fischer, and J. Smith. Water pathways in the bacteriorhodopsin proton pump. *J. Membr. Biol.*, 239:73–84, 2011.
- [6] B. Brooks, R. Brucolieri, B. Olafson, D. States, S. Swaminathan, and M. Karplus. CHARMM: a program for macromolecular energy, minimization, and dynamics. *J. Comput. Chem.*, 4:187–217, 1983.
- [7] A. Bryden, G. N. Phillips, and M. Gleicher. Automated illustration of molecular flexibility. *IEEE Trans. Vis. Comput. Graphics*, 18(1):132–145, 2012.
- [8] M. P. Calace, T. Maiwald, and J. M. Thornton. PoreWalker: A novel tool for the identification and characterization of channels in transmembrane proteins from their three-dimensional structure. *PLoS Comput. Biol.*, 5(7), 2009.
- [9] R. G. Coleman and K. A. Sharp. Finding and characterizing tunnels in macromolecules with application to ion channels and pores. *Biophys. J.*, 96:632–645, 2008.
- [10] J. Cortés, T. Siméon, V. De Angulo, D. Guiyesse, M. Remaud-Siméon, and V. Tran. A path planning approach for computing large-amplitude motions of flexible molecules. *Bioinformatics*, 21(suppl 1):i116–i125, 2005.
- [11] T. Darden, D. York, and L. Pedersen. Particle mesh ewald: an  $n \log(n)$  method for ewald sums in large systems. *J. Chem. Phys.*, 98:10089–10092, 1993.
- [12] R. O. Dror, M. Jensen, D. W. Borhani, and D. E. Shaw. Exploring atomic resolution physiology on a femtosecond to millisecond timescale using molecular dynamics simulations. *J. Gen. Physiol.*, 135(6):555–562, 2010.
- [13] H. Edelsbrunner. Deformable smooth surface design. *Discrete Comput. Geom.*, 21(1):87–115, 1999.
- [14] H. Edelsbrunner and E. P. Mücke. Three-dimensional alpha shapes. *ACM Trans. Graph.*, 13:43–72, 1994.
- [15] U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. Pedersen. A smooth particle mesh Ewald method. *J. Chem. Phys.*, 103:8577–8593, 1995.

- [16] T. Exner, M. Keil, G. Möckel, and J. Brickmann. Identification of substrate channels and protein cavities. *J. Mol. Model.*, 4:340–343, 1998.
- [17] S. Feller and A. MacKerell Jr. An improved empirical potential energy function for molecular simulations of phospholipids. *J. Phys. Chem. B*, 104:7510–7515, 2000.
- [18] S. Feller, Y. Zhang, R. Pastor, and B. Brooks. Constant pressure molecular dynamics simulation: the Langevin piston method. *J. Chem. Phys.*, 103:4613–4621, 1995.
- [19] H. Grubmüller, H. Heller, A. Windemuth, and K. Schulten. Generalized Verlet algorithm for efficient molecular dynamics simulations with long-range interactions. *Mol. Simul.*, 6:121–142, 1991.
- [20] S. Grudinin, G. Büldt, V. Gordeliy, and A. Baumgaertner. Water molecules and hydrogen-bonded networks in bacteriorhodopsin – molecular dynamics simulations of the ground state and the m-intermediate. *Biophys. J.*, 88(5):3252–3261, 2005.
- [21] A. D. Gruia, A.-N. Bondar, J. C. Smith, and S. Fischer. Mechanism of a molecular valve in the halorhodopsin chloride pump. *Structure*, 13(4):617 – 627, 2005.
- [22] J. Herzfeld and J. Lansing. Magnetic resonance studies of the bacteriorhodopsin pump cycle. *Annu. Rev. Biophys. Biomol. Struct.*, 31, 2002.
- [23] B. K. Ho and F. Gruswitz. HOLLOW: generating accurate representations of channel and interior surfaces in molecular structures. *BMC Struct. Biol.*, 8(1):49+, 2008.
- [24] B. Huang and M. Schroeder. LIGSITEcs: Predicting ligand binding sites using the Connolly surface and degree of conservation. *BMC Struct. Biol.*, 6(1):19, 2006.
- [25] E. Jardón-Valadez, A.-N. Bondar, and D. J. Tobias. Coupling of retinal, protein, and water dynamics in squid rhodopsin. *Biophys. J.*, 99(7):2200–2207, 2010.
- [26] W. L. Jorgensen, J. Chandrasekhar, J. Madura, R. Impey, and M. Klein. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.*, 79:926–935, 1983.
- [27] G. P. B. Jr. and P. F. W. Stouten. Fast prediction and visualization of protein binding pockets with pass. *J. Comput-Aided Mol. Des.*, 14(4):383–401, 2000.
- [28] L. Kale, R. Skeel, M. Bhandarkar, and R. Brunner et al. NAMD2: greater scalability for parallel molecular dynamics. *J. Comput. Phys.*, 151:283–312, 1999.
- [29] D.-S. Kim, Y. Cho, and D. Kim. Euclidean Voronoi diagram of 3d balls and its computation via tracing edges. *Comput. Aided Des.*, 37(13):1412 – 1424, 2005.
- [30] G. J. Kleywegt and T. A. Jones. Detection, delineation, measurement and display of cavities in macromolecular structures. *Acta Crystallogr. D*, 50(2):178–185, 1994.
- [31] M. Krone, K. Bidmon, and T. Ertl. Interactive visualization of molecular surface dynamics. *IEEE Trans. Vis. Comput. Graphics*, 15(6):1391–1398, 2009.
- [32] M. Krone, M. Falk, S. Rehm, J. Pleiss, and T. Ertl. Interactive exploration of protein cavities. *Comput. Graph. Forum*, 30(3):673–682, 2011.
- [33] M. Krone, S. Grottel, and T. Ertl. Parallel contour-buildup algorithm for the molecular surface. In *Proc. IEEE Symposium on Biological Data Visualization (BioVis 2011)*, pages 17–22, 2011.
- [34] R. A. Laskowski. Surfnet: A program for visualizing molecular surfaces, cavities, and intermolecular interactions. *J. Mol. Graphics*, 13(5):323 – 330, 1995.
- [35] S. M. LaValle. Rapidly-exploring random trees: A new tool for path planning. Technical Report TR 98-11, Iowa State University, 1998. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.35.1853>.
- [36] D. G. Levitt and L. J. Banaszak. Pocket: a computer graphics method for identifying and displaying protein cavities and their surrounding amino acids. *J. Mol. Graph.*, 10(4):229–234, 1992.
- [37] N. Lindow, D. Baum, and H.-C. Hege. Voronoi-based extraction and visualization of molecular paths. *IEEE Trans. Vis. Comput. Graphics*, 17(12):2025 – 2034, 2011.
- [38] N. Lindow, D. Baum, S. Prohaska, and H.-C. Hege. Accelerated visualization of dynamic molecular surfaces. *Comput. Graph. Forum*, 29(3):943–952, 2010.
- [39] H. Luecke, B. Schobert, H.-T. Richter, J.-P. Cartailler, and J. K. Lanyi. Structure of bacteriorhodopsin at 1.55 resolution. *J. Mol. Biol.*, 291(4):899 – 911, 1999.
- [40] A. MacKerell, J. D. Bashford, M. Bellott, and R. Dunbrack et al. All-atom empirical potential for molecular modeling and dynamics studies of proteins. *J. Phys. Chem. B*, 102:3586–3616, 1998.
- [41] G. J. Martyna, D. J. Tobias, and M. L. Klein. Constant-pressure molecular-dynamics algorithms. *J. Chem. Phys.*, 101:4177–4189, 1994.
- [42] P. Medek, P. Beneš, and J. Sochor. Computation of tunnels in protein molecules using Delaunay triangulation. *J. of WSCG*, 15(1-3):107–114, 2007.
- [43] MegaMol. <http://vis-web.informatik.uni-stuttgart.de/trac/megamol>.
- [44] M. Nina, B. Roux, and J. Smith. Functional interactions in bacteriorhodopsin: a theoretical analysis of retinal hydrogen bonding with water. *Biophys. J.*, 68:25–39, 1995.
- [45] M. Petřek, P. Kosinová, J. Koca, and M. Otyepka. MOLE: A Voronoi diagram-based explorer of molecular channels, pores, and tunnels. *Structure*, 15(11):1357 – 1363, 2007.
- [46] M. Petřek, M. Otyepka, P. Banas, and P. Kosinová et al. CAVER: a new tool to explore routes from protein clefts, pockets and cavities. *BMC Bioinformatics*, 7(1):316+, 2006.
- [47] J. C. Phillips and K. Schulten et al. Scalable molecular dynamics with NAMD. *J. Comput. Chem.*, 26:1781–1802, 2005.
- [48] PyMOL. <http://pymol.org>.
- [49] M. Raunest and C. Kandt. dxTuber: Detecting protein cavities, tunnels and clefts based on protein and solvent dynamics. *J. Mol. Graph. Model.*, 2010.
- [50] F. M. Richards. Areas, volumes, packing, and protein structure. *Annu. Rev. Biophys. Bio.*, 6(1):151–176, 1977.
- [51] J.-P. Ryckaert, G. Ciccotti, and H. J. C. Berendsen. Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of nalkanes. *J. Comput. Phys.*, 23:327–341, 1977.
- [52] J. Schmidt-Ehrenberg. *Analysis and Visualization of Molecular Conformations*. PhD thesis, Freie Universität Berlin, Fachbereich Mathematik und Informatik, Dec 2008.
- [53] J. Schmidt-Ehrenberg, D. Baum, and H.-C. Hege. Visualizing dynamic molecular conformations. In *IEEE Visualization 2002*, pages 235–242. IEEE, 2002.
- [54] C. Sigg, T. Weyrich, M. Botsch, and M. Gross. GPU-based ray-casting of quadratic surfaces. In *Proc. Eurographics Symposium on Point-Based Graphics*, pages 59–65, 2006.
- [55] O. S. Smart, J. G. Neduvil, X. Wang, and B. A. Wallace et al. HOLE: a program for the analysis of the pore dimensions of ion channel structural models. *J. Mol. Graph.*, 14(6), 1996.
- [56] D. Stalling, M. Westerhoff, and H.-C. Hege. Amira: A highly interactive system for visual data analysis. In C. D. Hansen and C. R. Johnson, editors, *The Visualization Handbook*, pages 749 – 767. Elsevier, 2005.
- [57] E. Tajkhorshid, J. Baudry, K. Schulten, and S. Suhai. Molecular dynamics study of the nature and origin of retinal's twisted structure in bacteriorhodopsin. *Biophys. J.*, 78(2):683 – 693, 2000.
- [58] M. S. Till and M. Ullmann. McVol - a program for calculating protein volumes and identifying cavities by a Monte Carlo algorithm. *J. Mol. Model.*, 16(3):419–429, 2010.
- [59] M. Tuckerman and B. Berne. Reversible multiple time scale molecular dynamics. *J. Chem. Phys.*, 97:1990–2001, 1992.
- [60] B. Tversky, J. Morrison, and M. Betrancourt. Animation: can it facilitate? *Int. J. Hum.-Comput. St.*, 57(4):247–262, 2002.
- [61] VMD. <http://www.ks.uiuc.edu/Research/vmd>.
- [62] N. R. Voss and M. Gerstein. 3V: cavity, channel and cleft volume calculator and extractor. *Nucleic Acids Res.*, 38(Suppl 2):W555–W562, 2010.
- [63] E. Yaffe, D. Fishelovitch, H. J. Wolfson, D. Halperin, and R. Nussinov. MolAxis: Efficient and accurate identification of channels in macromolecules. *Proteins: Struct., Funct., Bioinf.*, 73(1):72–86, 2008.
- [64] J. Yu, Y. Zhou, I. Tanaka, and M. Yao. Roll: a new algorithm for the detection of protein pockets and cavities with a rolling probe sphere. *Bioinformatics*, 26(1):46–52, 2010.