Tracking single channel in protein dynamics

Petr Beneš
Faculty of Informatics
Masaryk University
Botanická 68a
602 00 Brno
Czech Republic
xbenes2@fi.muni.cz

Petr Medek
Faculty of Informatics
Masaryk University
Botanická 68a
602 00 Brno
Czech Republic
medek@fi.muni.cz

Jiří Sochor
Faculty of Informatics
Masaryk University
Botanická 68a
602 00 Brno
Czech Republic
sochor@fi.muni.cz

ABSTRACT

In this paper we present a new approach to the processing of molecule dynamics. The method performs the tracking of a channel in a sequence of molecule snapshots which represent atom positions in the molecule in certain time intervals. The centerline of the tracked channel is refined using the Delaunay triangulation from the actual snapshot resulting in a new optimized centerline. This method allows us easily to animate the behaviour of the channel in the sequence. The method can also be used to detect the channel geometry in snapshots, where recent methods are not able to find this channel. In addition, the method yields information about channel parameters which vary over time. We can evaluate opening and closing of the input channel.

Keywords

channel, protein dynamics, tracking, Voronoi diagram, Delaunay triangulation

1. INTRODUCTION

Biochemists usually want to observe the behaviour of a protein in a particular part of the molecule, e.g. to observe the exit route of a substrate. Channels as defined in [Med07] can be used to visualize this information. A channel which leads through an empty space in the molecule can for example be wide for a significant period of time or the substrate might initiate the opening of a narrow channel when passing by. This information helps chemists to predict the behaviour of a molecule before performing real experiments.

Most of the methods of channel computation are designed to process a single static protein molecule. There are only a few methods for analysing the dynamics of protein molecules. Since the dynamics of a protein molecule is a continuous movement, it is sampled into a sequence of snapshots representing atom positions in given time intervals. The snapshots are usually aligned so that the global position and rotation of the molecule is fixed in all snapshots and the snapshots only represent local movement of atoms.

Recent methods typically process each of these snapshots separately as static molecules and cluster obtained results at the end of the computation. Therefore, none of these methods is specialized for tracking a certain channel throughout the whole sequence. The visualization of this information can improve the process of protein analysis significantly.

The method proposed in this paper is able to detect a particular channel in each snapshot of the dynamics and the resulting channels are spatially close to each other. This allows us to animate the progress of a channel over time easily.

We can use also this method to compute a channel in the snapshots, where the classic approaches are not able to detect this channel since they compute only limited number of channels in each snapshot. However, there are situations where we need to know the channel geometry in each snapshot. Using the proposed method, the missing channel can be computed from the surrounding snapshots where the channel geometry is known.

The main advantage of the proposed method is not only to improve the visualization of channel progress over time. As demonstrated in the results section, the

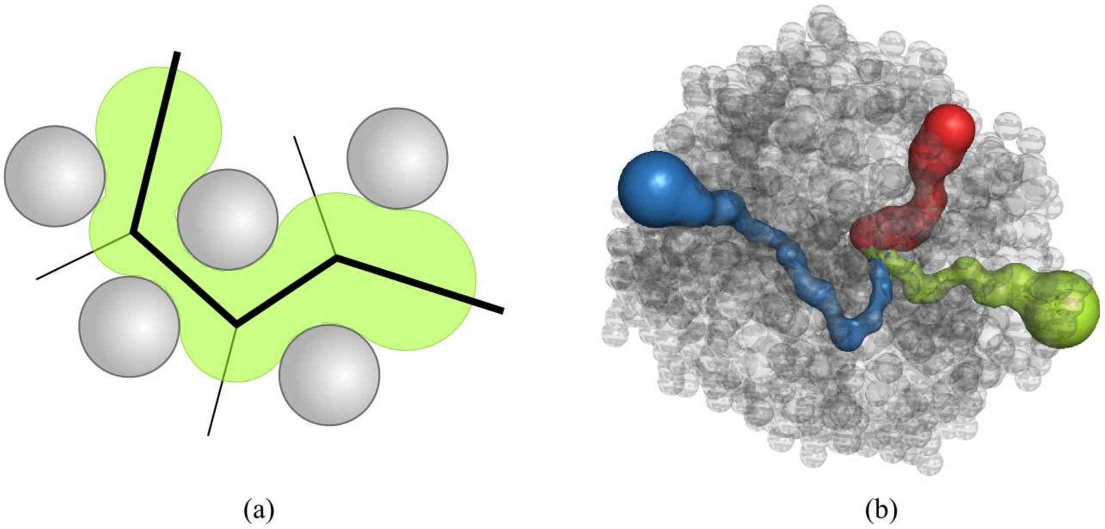

Figure 1. (a) Demonstration of a channel. This channel is ideal, i.e. its centerline leads along Voronoi edges. (b) Channels computed in a real static molecule and visualized using pyMol software.

overall evaluation of channels computed by this method can also bring new information about molecule and channel behaviour.

2. RELATED WORK

For the computation of channels in static molecules there have been many different methods proposed. All these methods process the molecule as a geometric model where each atom corresponds to a sphere with given position and radius in the three-dimensional space. Other biochemical properties are not considered. A channel in the molecule (also referred to as tunnel) is defined as a centerline and the volume ([Med07], Fig. 1). A centerline is a continuous curve and the volume is formed by the union of spheres inserted at each point of the centerline. The radius of all these inserted spheres is maximal so that it does not intersect any other atom in the molecule.

An approach introduced in [Pet06] is based on space rasterisation. This approach suffers from several disadvantages resulting from discrete sampling.

Other methods [Med07, Pet07, Yaf08] are based on the Delaunay triangulation (DT) and the Voronoi Diagram (VD) computed for the molecule. These methods are faster, more precise and more efficient than rastering solutions. However, they are designed to process a single static molecule and so they are not able to return information about channel properties varying across snapshots (such as the progress of the width of a channel over time). Nevertheless, the channels and their trajectories in the static snapshots can be used as an input for the tracking method proposed in this paper.

A method which is able to determine the progress of channels in protein molecules over time is called molecular dynamics (MD) [Ald59]. A small molecule (substrate) is positioned inside the protein molecule and a physical simulation starts. Random forces are applied to the substrate during the simulation and collisions and interactions of the substrate with the protein are evaluated. It is probable that the substrate molecule reaches the protein surface. This method is able to find certain exit route of a substrate leading from a given position inside the protein, so called active site, to the protein surface. Note that this method does not require the whole dynamics to be computed before starting the simulation (actually, it computes the dynamics itself in a run-time simulation). The movements of atoms are computed continuously during the simulation according to the result of force interactions. The method is immensely time consuming (one simulation takes hours to days to evaluate). Due to the application of random forces, it is not guaranteed that a channel is found. If a channel is not detected, it does not mean that this channel does not exist.

The complex approach proposed in [Ben09] requires a sequence of snapshots to be known in advance, either from some real screening or existing simulation. It computes channels in snapshots separately using any method for the computation of channels in a static molecule and clusters them afterwards for the whole dynamics. Each cluster represents the progress of a particular channel throughout the dynamics. Since the methods for the computation of channels in a single snapshot produce only a limited number of channels, it is probable that

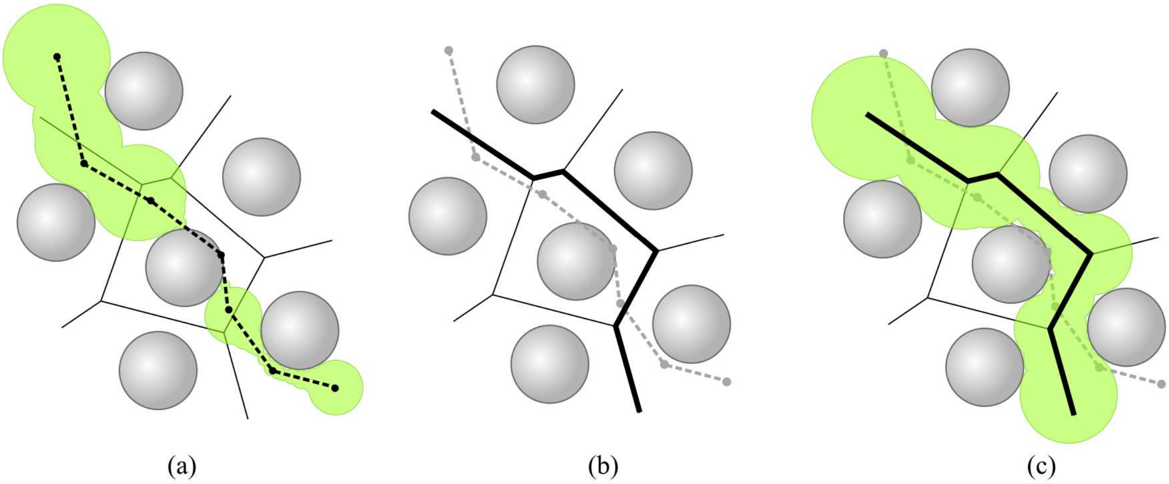

Figure 2. (a) Initial trajectory (dashed polyline) and the corresponding channel, (b) Voronoi diagram with optimized trajectory emphasized, (c) Channel defined by optimized trajectory

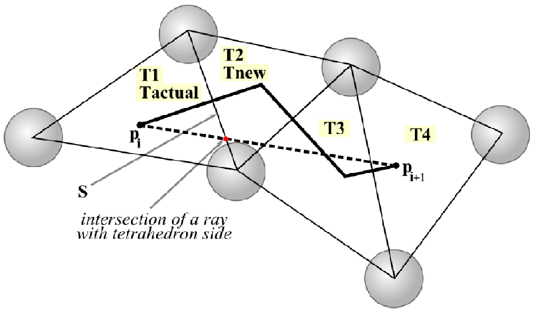

Figure 3. The demonstration of the algorithm for the segment <p_i, p_{i+1}>

some clusters will not cover the whole sequence. We can consider a dynamic channel to be closed in the snapshots, where the cluster provides no information about the geometry of this dynamic channel. However, this complicates the visualization of the channel dynamics.

```
Input:  initial trajectory t = $p_1..p_n$,
        tetras of DT for the actual
        snapshot
Output: optimized trajectory

for each <$p_i$, $p_{i+1}$>, i ∊ 1..n-1
{
  $T_{actual}$ = tetra containing $p_i$;
  while ($T_{actual}$ not contains $p_{i+1}$)
  {
     s = side of $T_{actual}$ intersected
         by <$p_i$, $p_{i+1}$>;
     $T_{new}$ = tetra sharing s with $T_{actual}$;
     // function c(T) returns center
     // of gravity for tetrahedron T
     output(<c($T_{actual}$),c($T_{new}$)>);
     $T_{actual}$ = $T_{new}$;
  }
}
```

Algorithm 1. The optimization of an initial trajectory in a single snapshot of a molecule.

3. PROPOSED METHOD

The method proposed in this paper is able to track the progress of a specific part of the protein molecule over time. The specific part is described by a channel. Its centerline is referred to as *initial trajectory* for further tracking.

The initial trajectory can be the centerline of an already known important channel or it can also be an exit route of a substrate computed by molecular dynamics.

Algorithm

In each snapshot, we optimize the initial trajectory so that the channel formed by this trajectory has the maximal possible volume. If we do not optimize the trajectory, the channel can be very narrow or it can even have zero or negative width (see Fig. 2a). The optimized trajectory follows edges of a Voronoi diagram of the protein molecule (Fig. 2b) and thus the resulting channel can be much wider (see Fig. 2c).

The algorithm utilizes the duality between VD and DT. The initial trajectory is mapped onto a sequence of tetrahedra in the DT. This sequence can be converted to Voronoi edges easily.

The initial trajectory is represented as a polyline with vertices $p_1,...,p_n$. These input points define $n-1$ line segments. For each of the segments <p_i, p_{i+1}> ($i=1,...,n-1$) the tetrahedron T_{actual} containing p_i is located and marked as actual. Then we determine the tetrahedron side *s* which intersects the ray between p_i and p_{i+1}. As the next step we move into the tetrahedron T_{new} which shares the side *s* with actual tetrahedron T_{actual}. Finally, T_{new} is marked as actual. The process is depicted in Fig. 3. The line segment

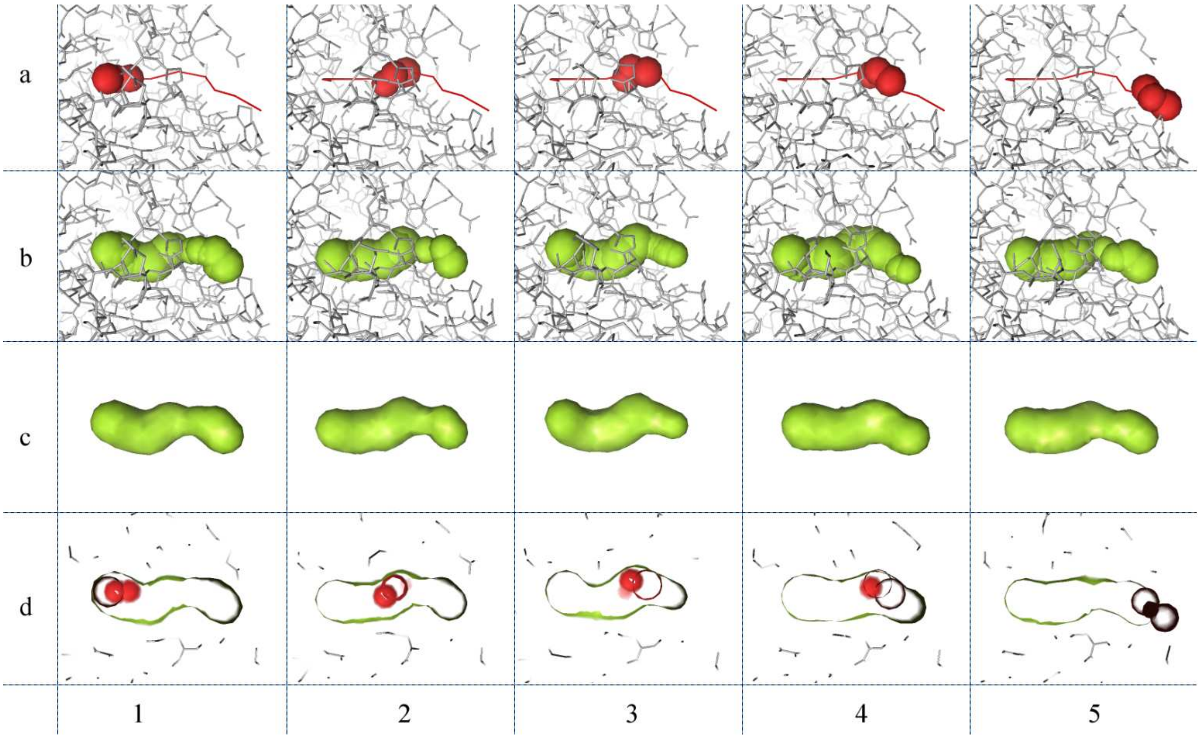

Figure 4. (a) The exit path of the substrate molecule in time (1-5) in 1mj5. This path was used as the initial trajectory. (b-d) Different types of visualization of a channel dynamics in certain snapshots (1-5).

$<p_i, p_{i+1}>$ is substituted by the polyline which is formed by Voronoi edges adjacent to the tetrahedra in the DT which were traversed during the above procedure. The procedure is summarized in Alg. 1.

Note that the implementation of Delaunay triangulation enables to determine all neighbours of a given tetrahedra in constant time. In addition, the geometry test required for each processed tetrahedron is the calculation of the intersection of a line segment and a triangle (tetrahedron side) in three dimensions, which is fast and simple.

Since each segment $<p_i, p_{i+1}>$ is replaced by the Voronoi edges dual to the tetrahedra intersected by $<p_i, p_{i+1}>$, the spatial distance between initial and optimized trajectory is the minimal possible.

Notice that also the approach presented in [Med07] with minor changes could be used to get the widest channel between each two segment endpoints. Nevertheless, the computation would be more time consuming as the Dijkstra's algorithm would be used instead of fast following the ray-tetrahedra intersections in the DT. In addition, the channel might be much longer and its centerline might lead far from the initial trajectory. This fact would certainly complicate the smooth and continuous animation of a channel over time.

Time complexity

The time required to compute the Delaunay triangulation is quadratic with respect to the number of atoms in the molecule [Pre85]. The subsequent tracking of an input trajectory is linear with respect to the number of atoms. The overall complexity is $O(k*n^2)$ where n is the number of atoms and k is the number of molecule snapshots.

The computation time can be reduced by computing only a subset of Delaunay triangulation which would be located near the input trajectory. If a trajectory covers only a small part of the molecule the computation time can be reduced significantly.

4. RESULTS

The first dynamics analysed by the proposed method is the protein molecule *1mj5* consisting of 50 snapshots. This sequence was achieved by MD simulation of the molecule and substrate. Therefore, the exit route of the substrate is known in this case. This exit route is used as the initial trajectory in the computation. When we visualize the results of the analysis, we can observe the substrate initiates opening of the channel, i.e. the channel gets wider in places where the substrate passes.

Different types of visualization of these results are depicted in Fig. 4. Five snapshots (1-5) are chosen from the dynamics to illustrate the progress over time. The results are visualized in different ways (a-d). The first of them (a) shows the exit route of a substrate and the trajectory of this route. The others show the resulting channel with a centerline located on the optimized trajectories in different snapshots displayed

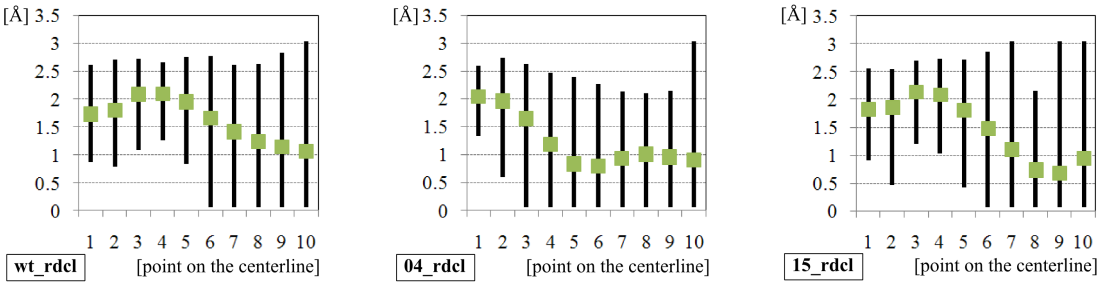

Figure 5. Charts depicting channel statistics for selected protein dynamic sequences. The x-axis denotes uniformly distributed points on a centerline and the y-axis denotes the variation of channel width in these points throughout the dynamics: maximum width, minimum width and average width in the whole sequence.

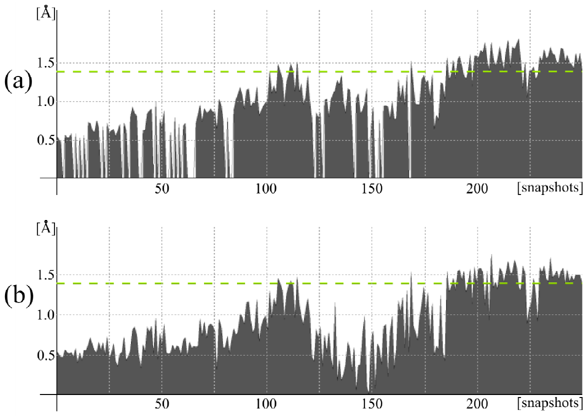

Figure 6. The analysis of the width of a channel in the sequence of 250 snapshots using (a) the clustering method, (b) the proposed method. The dashed green line denotes the biochemically important value 1.4Å

as a set of spheres (b) or a surface (c). In (d), the whole scene is clipped using the front clipping plane approximately in the middle of the channel. In this case we can observe the substrate molecule passing through the channel.

Notice that the previous example demonstrates a possible use of this method on dynamics where an exit route is known before. If such route is not known, chemists have to define the initial trajectory they want to observe. The definition can be done by hand or the widest channel from a single snapshot of the dynamics can be used.

The behaviour of the channel in the analysis indicates whether a certain substrate would be able to pass through this channel.

The second data set consisted of a set of nine molecules of type *rdcl*, which were structurally similar. They were mutants of the same protein molecule (only a few residues were different in each of the mutants). The dynamics of each mutant consisted of 400 snapshots. We tracked the same initial trajectory in all dynamic sequences.

The behaviour of resulting channels was analyzed in ten uniformly distributed points along their centerlines. The first point refers to the channel endpoint in the active site and the tenth point is the channel endpoint located near the molecule surface.

For each dynamics, statistics about the pulsing of a channel in each of these segments were computed. The statistic for selected molecules is shown in Fig. 5. The average width, minimal width and maximal width (y-axis) are shown for each of the ten points on the x-axis. One of the possible interpretations of the data in these charts is the following. All channels tend to be more stable near the active site whereas certain opening and closing of a channel happens near the molecule surface. It can be seen that in case of *wt_rdcl* (Fig. 5, *wt_rdcl*) the first half of a channel remained open during the whole sequence with radius varying from 0.8Å[1] to 2.6Å whereas the radii in the second half varied more significantly.

This information helps chemists to estimate which of the mutants is the most suitable for a certain substrate molecule to penetrate into the protein.

The visualizations in Fig. 4 were created using pyMol software [DeL02].

As a third test case, we have analysed the width of a channel in the sequence of 250 snapshots of *21_rdcl.cl* using the clustering method [Ben09] and the proposed method. In Fig. 6, it can be seen that both methods provide similar results. In the case of clustering method (Fig. 6a) the width of a channel is usually slightly larger. However, there are snapshots in which the channel is not detected. On the contrary, the proposed method (Fig. 6b) detects the channel in all snapshots. Therefore we can use results of this method to add the missing channel data.

We have also evaluated the distances between channels in all consecutive snapshots according to the distance function defined in [Ben09]. In comparison with the graph cutting clustering method, the distance

[1] 1 Å = 10^{-10} m

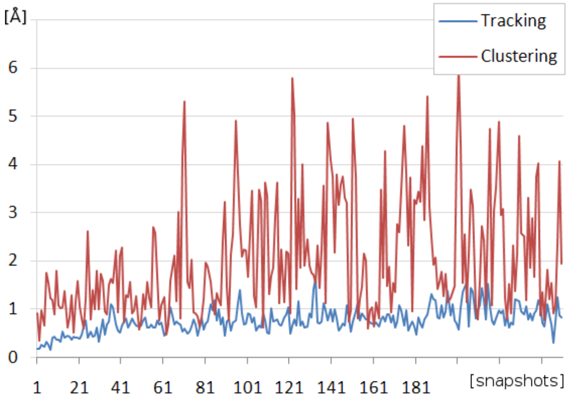

Figure 7. Distances between channels in consecutive snapshots for tracking and clustering method. First 250 snapshots containing channels were considered.

between channels computed using the proposed method is much smaller. The example for the sequence *21_rdcl.cl* can be seen in Fig. 7. The average distance for the proposed tracking method was 0.77Å whereas the average distance for the clustering method was 2.03Å. Due to the fact that the average distance is small, the progress of a channel over time can be easily animated.

5. CONCLUSION

The proposed method allows tracking an initial trajectory in the dynamics. The method is based on computational geometry and is fast and robust. Except for computation of the Delaunay triangulation, it uses only basic geometry tests.

We have also presented several applications of this method. The optimization is of key importance when performing smooth animation of channel progress across snapshots over time.

The properties of resulting optimized trajectories can provide useful information about the protein molecule. The possible interpretation of such results has been suggested. The presented method can also be used in snapshots where recent methods are not capable of detecting the channel geometry.

As for the future work, we plan to integrate this method within the complex software application Caver Viewer (http://loschmidt.chemi.muni.cz/caver/) which is designed for the visualization and analysis of protein molecules.

6. ACKNOWLEDGMENTS

This work was supported by The Ministry of Education of The Czech Republic, Contract No. LC06008 and by The Grant Agency of The Czech Republic, Contract No. 201/07/0927.

7. REFERENCES

[Ald59] Alder, B. J., and Wainwright, T. E., Studies in molecular dynamics. i. general method. The Journal of Chemical Physics, vol. 31, no. 2, pp. 459–466, 1959.

[Ben09] Beneš, P., Medek, P., and Sochor, J. Computation of channels in protein dynamics. IADIS International Conference Applied Computing 2009, Rome, pp. 251-258, 2009

[DeL02] DeLano, W.L. The PyMOL Molecular Graphics System (2002) on World Wide Web http://www.pymol.org

[Med07] Medek P., Beneš P., Sochor J.: Computation of tunnels in protein molecules using Delaunay triangulation. Journal of WSCG 15, 1–3, pp. 107–114, 2007.

[Pet06] Petřek, M., Otyepka, M., Banáš, P., Košinová P., Koča, J., and Damborský J. CAVER: a new tool to explore routes from protein clefts, pockets and cavities. BMC Bioinformatics, 2006.

[Pet07] Petřek, M., Košinová, P., Koča, J., and Otyepka M.: Mole: A Voronoi diagram-based explorer of molecular channels, pores, and tunnels. Structure 15, 11, pp. 1357–1363, 2007.

[Pre85] Preparata, F.P., and Shamos, M.I. Computational Geometry: An introduction. Springer-Verlag, 1985.

[Yaf08] Yaffe, E., Fishelovitch, D., Wolfson, H. J., Halperin, D., and Nussinov, R.: Molaxis: Efficient and accurate identification of channels in macromolecules. Proteins, 73(1):72-86, 2008.