Visualization of Tunnels in Protein Molecules

Barbora Kozlíková
Faculty of Informatics
Masaryk University
Botanická 68a
60200, Brno, Czech Republic
xkozlik@fi.muni.cz

Filip Andres
Faculty of Informatics
Masaryk University
Botanická 68a
60200, Brno, Czech Republic
xandres@fi.muni.cz

Jiří Sochor
Faculty of Informatics
Masaryk University
Botanická 68a
60200, Brno, Czech Republic
sochor@fi.muni.cz

Abstract

This paper presents two novel techniques for visualization of tunnels in complex molecules of proteins. Long-term research in the field of protein analysis proved that the reactivity of the protein molecule depends on the presence of tunnels. These structures are very important mainly in the process of finding new pharmaceuticals. Visualization of a tunnel is the next very important step after the analysis because it enables the biochemists to determine the crucial regions of the tunnel, which can have a substantial effect in the process of designing new medication.

Previous methods for the visualization of tunnels define a tunnel as a set of intersecting spheres. Our approach exploits tetrahedra obtained from the process of tunnel analysis based on the Voronoi diagrams and Delaunay tetrahedrization. We proposed two novel algorithms that visualize a tunnel as a surface derived from the tetrahedra, which form the boundary constraint of the tunnel in the space of the molecule.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism;

Keywords: protein, tunnel, visualization, tetrahedrization, Loop subdivision

1 Introduction

Proteins are very complex molecules playing a crucial role in all live organisms. They consist of a large number of atoms, grouped into amino acids. Every atom corresponds to some chemical element and the model of protein associates specific values with it, such as the type of the element, its van der Waals radius, visualization colour etc. Among other visualization approaches, we may show the molecule as a set of intersecting spheres located in the three-dimensional space. These spheres have its predefined radius corresponding to the van der Waals radius of the atom (Figure 1).

Comprehensible visualization of such molecular structures is a well-known problem and researchers focus on it for many years. Currently chemists use many methods that reveal the respective chemical and spatial dependencies within the molecule. To visualize new features, we usually have to introduce new methods and combine them properly with commonly used ones.

Among these new features belong tunnels. The tunnels are spatial structures, which represent paths used for transporting a small
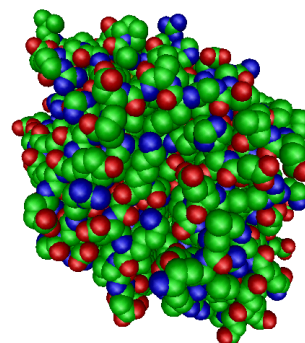

Figure 1: *Molecule of Haloalkane dehalogenase visualized as a set of spheres with the van der Waals radii.*

molecule of substrate to the active site of the protein. The presence of tunnels in proteins influences their reactivity.

Consider a molecule of substrate with a specific size. We want to test whether this substrate can react with a protein. It means that we look for a path from the outside of the protein to its inner part. This path must have a proper width, which is greater or equal to size of the substrate. The absence of such path inhibits the chemical reaction between the substrate and the protein. These paths are called **tunnels**. If we are able to detect these tunnels, we can exclude in advance the combinations of substrates and proteins that cannot lead to chemical reaction. In consequence, the analysis of proteins leading to the tunnel detection enables the biochemists to accelerate the process of designing new drugs. As the analytical process consists of many trials and human decisions, visualization of computed tunnels plays crucial role. It is very important mainly in situations when we obtain more than one tunnel from the analysis and biochemists have to decide which tunnel is the most promising one for entrance of the substrate molecule.

The interdisciplinary nature of the problem strongly influenced our research. It combines issues related to chemistry and computer graphics. The issues originate from three domains – chemical background of the problem, computational geometry used for the tunnel detection, and visualization of such specific structures as tunnels are. As for the chemical relevance of tunnels, they determine the path from the inside of the molecule to its outer boundary and vice-versa. A small molecule of substrate uses this path to access a site, which is located in the inner part of the protein. This site is called the *active site* of the molecule and it is the place where a chemical reaction between the substrate and protein can undergo (Figure 2). The active site is set as a starting point for the computation of a set of tunnels. The main issue is that the molecule itself is not a static system. The neighborhood of the molecule as well as other biochemical factors causes the movement of its atoms. This also influences the behavior of the tunnel, which can change its shape and width or even disappear. Moreover, it is convenient for bio-

chemists to find more than one in the protein molecule because the best tunnel found by geometrical algorithms does not have to be the best one from the chemical point of view; there are many chemical aspects, which influence the quality and relevance of tunnels.
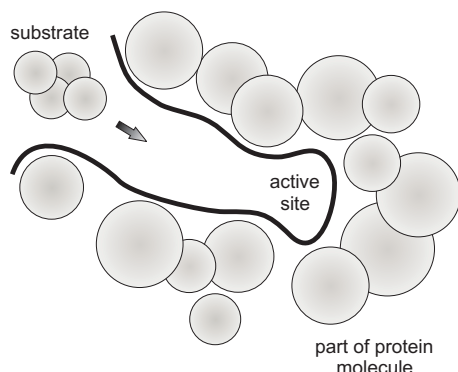

Figure 2: *A small molecule of substrate enters the tunnel and heads to the active site of the protein molecule.*

In our approach, we based tunnel computation on Voronoi diagrams and Delaunay tetrahedrization. Our algorithm computes an empty space ranging from the active site to the molecular surface. The atoms of the molecule bound this hollow and define the points in the centers of Voronoi cells. In this context, biochemists assess quality of tunnels by the minimum width of the tunnel, and the computation of the best tunnels proceeds with respect to this criterion.

However, the main reason why we use Voronoi and Delaunay structures is that it allows unambiguously distinguish tunnels. We need to detect the same tunnel in subsequent configurations, as we need to track it during the movement of the molecule in time. This paper is not concerned with the problem of analysis and visualization of dynamic chemical structures. However, the Delaunay tetrahedrization provides a suitable structure for tracing the evolution and behavior of the different tunnels in subsequent time slots, and we will use it in our future research related to dynamic movement of proteins and tunnels.
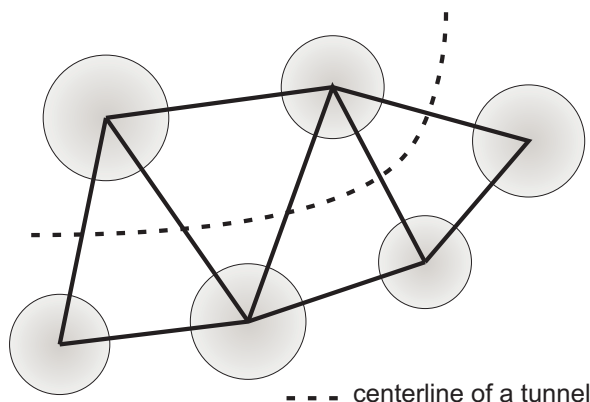

Figure 3: *Tetrahedrization of atoms forming a tunnel (projected to 2D).*

Having the information obtained from the tunnel analysis, we need to visualize it. For biochemists, the visualization of the results of tunnel analysis is very important; they want to see the resulting tunnels and visually examine the space of the tunnel and its surroundings. Therefore, the tunnel visualization has to combine both

the chemical and spatial information about the structure of a tunnel. Previous methods for the visualization of tunnels represent the tunnel as a set of intersecting spheres. This method does not involve the exact boundary of the tunnel and its shape. We proposed novel approaches based of Delaunay tetrahedrization, which exploit the tetrahedra forming the exact boundary of the computed tunnel. Its trajectory and a bounding volume with a specific shape, which is influenced by the surrounding atoms, represent the tunnel. As the output of the analysis, we obtain the following information, which we use as an input for tunnel visualization:

1. The trajectory (centerline) of the tunnel and values that represent the width of the tunnel at the sampled points on the centerline. The density of sampling can be set prior to the tunnel computation. To visualize the resulting tunnel volume, we may show it as a set of intersecting spheres.

2. A set of successive triangles from which we can reconstruct the tetrahedra forming a tunnel.

The first structure forms the basis of the existing methods for the tunnel visualization. The tetrahedra reconstructed from the second possible output of the analysis bound the space of the computed tunnel. The vertices of tetrahedra are located at the atoms surrounding the tunnel (Figures 3, 4). As will be shown in section 4, this structure is more suitable for the comprehensible visualization. The tetrahedra define the tight boundary of a tunnel and they serve as a better starting point for visualization than a set of intersecting spheres. However, it is obvious from Figure 4 that direct visualization of the tunnel as a set of tetrahedra is insufficient and we have to propose techniques, which enhance its appearance.
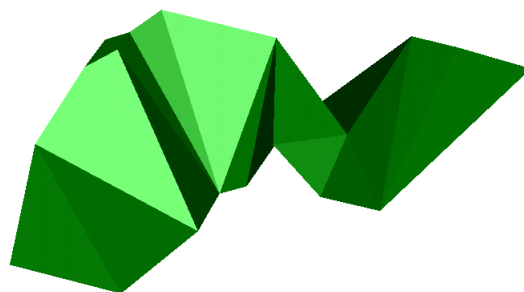

Figure 4: *A set of tetrahedra obtained from the Delaunay tetrahedrization; the tetrahedra approximate a tunnel.*

The first attempt on tunnel detection was implemented in CAVER program ([Petřek et al. 2006]) and received great response from biochemists. The improved technique [Medek et al. 2007] providing results that are more accurate, employs tetrahedrization. Biochemist community tested the algorithm and their reaction was very positive. In consequence, there was a huge increase of users of the CAVER program [Medek et al. 2007] and it raised the demands for the better visualization of tunnels. The large amount of expected users of a new technique is a good motivation to continue research in this area.

2 Related Work

Protein analysis leading to the tunnel detection appeared recently in the field of computer graphics and the analysis and visualization of tunnels represent the application of computer graphics to this specific problem. Most of the related work was published in chemical

journals, for example [Damborský et al. 2007; Petřek et al. 2006]. These papers mostly present chemical background of this problem.

Analysis of tunnels uses the traditional algorithms known from the field of computational geometry [Barber et al. 1996]; these algorithms are adapted for the construction of tunnels. The original approach to the computation of tunnels was based on the space division and the implementation of this technique is used in the first version of program called CAVER [Petřek et al. 2006]. The new algorithm based on Voronoi diagrams and Delaunay tetrahedrization [Medek et al. 2007] provides more correct and accurate tunnels and increases the speed of their computation. It is included in the latest version of CAVER.

Existing methods for visualization of the protein structures mainly focused on different techniques for representation of the whole molecule (its atoms, bonds, etc.). However, most of these techniques are not suitable for the tunnel visualization. For the representation of these specific structures there is currently available only one approach which was implemented in the program PyMOL [DeLano 2002] and in the program VMD [Humphrey et al. 1996]; these programs visualize the tunnel as the set of intersecting spheres (Figure 5). The important issue in this approach is the elimination of polygons, which lie in the inner part of the tunnel, as we want to visualize only a boundary of the tunnel.

Another possible approach is to visualize a tunnel using some of the existing techniques for the visualization of the molecular surface. Two main techniques are Reduced Surfaces [Olson and Spehner 1996] and Alpha Shapes [Edelsbrunner and Mücke 1994; Akkiraju 1996]. In [Akkiraju 1996] author describes the algorithm, which is also suitable for the visualization of cavities in proteins.

Both of these techniques provide similar results. They follow the idea that the molecule is a set of intersecting spheres. For example, the basic principle of the algorithm described in [Olson and Spehner 1996] is to set some probe which sweeps over the border atoms of the molecule and defines a topological boundary. This boundary then forms the surface of the molecule. The idea can be extended to tunnels because we obtain a tunnel in a similar form, i.e. as a set of intersecting spheres. Compared to our approach, this technique produces smooth surfaces but it does not provide the representation of free space between atoms defining the tunnel. This technique was also implemented in PyMOL [DeLano 2002]. In Figure 5, there is the tunnel visualized as a set of intersecting spheres and the same surface smoothed.

3   Problem definition

The aim of this research is to take the structure obtained from the tunnel analysis and the related information and show the shape of a tunnel in such way that it can reveal other interesting aspects such as the direction of the tunnel, its width, neighboring atoms and their contribution to the boundary of the tunnel. In simple terms, we consider a set of spheres in the three-dimensional space. These spheres are of different radii and bound some empty space – a tunnel. Our goal is to visualize the boundary of this tunnel as accurately as possible because the width of the tunnel limits the size of the molecule of substrate, which can potentially react with the protein.

However, the above-mentioned statement is oversimplified. For biochemists, many more conditions influence the behavior of the tunnel and we should take all the aspects into consideration, not only the geometry. Considering different biochemical dependencies, we simply cannot involve many of these into tunnel computation because of their non-algorithmic nature. This leads to the requirement that the tunnel analysis should compute more than one tunnel leading from the active site and their suitable visualization

is necessary. Using visualization the biochemists may evaluate the relevance of these tunnels and concentrate only on the important ones. The tunnels can overlap or even share some parts (see Figure 6). In this case, the visualization of tunnels as a set of spheres is almost unusable, as the complex situations where the tunnels split, we cannot visualize transparently. This can be seen in the figure 15 where we compare the visualization using sphere boundary approach with our method.
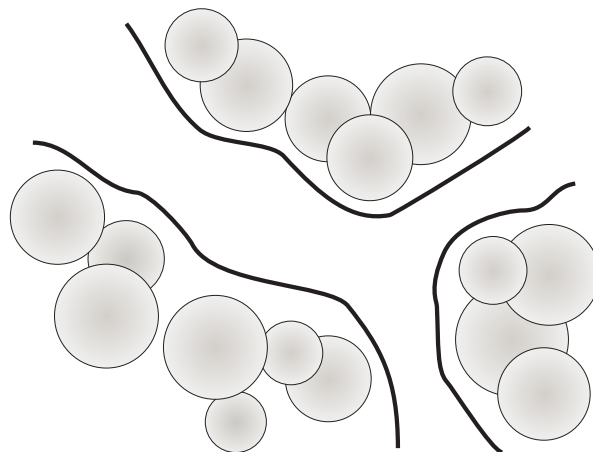

Figure 6: *Tunnel divided into two branches.*

In this paper, we focus first on the problem of visualization of one tunnel in a molecule of protein. We propose two possible solutions to the problem of tunnel visualization. Then we show that our solution provides very good results in the visualization of more than one tunnel and is usable for the better orientation in the space of the molecule. For the better comprehensibility we show in the following figures only the resulting tunnel (or tunnels). You can see the whole molecule of protein with computed tunnels in the figure 17.

4   Visualization based on the Delaunay tetrahedrization

We propose two novel approaches to the tunnel visualization. Both of them are based on the Delaunay tetrahedrization which we obtain as an output from the tunnel analysis. The output consists of a set of successive tetrahedra approximating a tunnel, where each two successive tetrahedra share one triangle. We classify the triangles of a tetrahedron according to their position in the tunnel.

We define a *gate* as the triangle of a tetrahedron with the following characteristics:

1. The triangle is situated in the inner part of the tunnel, it does not form the boundary of the tunnel. This triangle is shared by two tetrahedra.

2. The centerline of the tunnel crosses this triangle.

Gates are arranged in a sucessive order. The important characteristic of the neighbour gates is that they have two shared vertices. From these gates we can reconstruct all the information we need in the process of visualization.

We remark that for visualization purpose the tunnel is represented as a closed surface, i.e. the beginning and end of the tunnel are closed.

We propose two novel approaches to the tunnel visualization. Both of them we based on Delaunay tetrahedrization, which we obtain
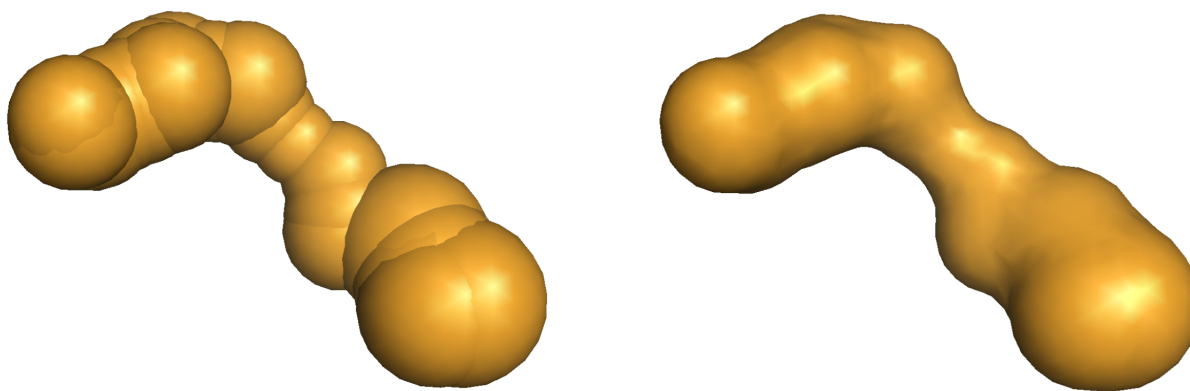
Figure 5: *Techniques for the tunnel visualization implemented in PyMOL: the tunnel visualized as a set of intersecting spheres (left); the same tunnel with a smoothed surface (right).*

as an output from the tunnel analysis. The output consists of a set of successive tetrahedra approximating a tunnel, where each two successive tetrahedra share one triangle. We classify the triangles of a tetrahedron according to their position in the tunnel.

We define a *gate* as the triangle of a tetrahedron with the following characteristics:

1. The triangle is situated in the inner part of the tunnel, it does not form the boundary of the tunnel. This triangle is shared by two tetrahedra.

2. The centerline of the tunnel crosses this triangle.

Gates are arranged in a sucessive order. The important characteristic of the neighbour gates is that they have two shared vertices. From these gates, we can reconstruct all the information we need in the process of visualization.

We remark that for visualization purpose we represent the tunnel as closed surface, i.e. the beginning and end of the tunnel are closed.

4.1 First approach – visualization using subdivision scheme

Present techniques for the tunnel visualization do not comprehensively show the shape and direction of the tunnel. As mentioned before, these features are substantial for the detection of the most relevant tunnels. The first algorithm we based on the idea of thinning the surface of a tunnel starting from an approximating boundary given as a set of triangles obtained from the tetrahedrization. We refine the surface given as triangle mesh using the Loop subdivision algorithm[Loop 1987]. We chose this scheme because, in comparison with other subdivision schemes (Doo - Sabin, Catmull - Clark), it provides smoother surfaces.

The major advantage of thinning approach is obvious whenever we visualize more than one tunnel at once (see Figure 15).

4.1.1 First step – reconstruction of boundary triangles

The input of the algorithm obtained from the tunnel analysis is a set of successive gates of a tunnel. From these gates we reconstruct the boundary triangles of the tunnel – we call them *faces*. The main observation is that every face has one vertex shared by two neighbor gates and two vertices, which are not shared. This reconstruction phase is illustrated in the Figure 7. The upper shows sequence of the gates of tetrahedra (two successive gates share edge)

and the lower picture illustrates the boundary triangles (faces) obtained from these gates.
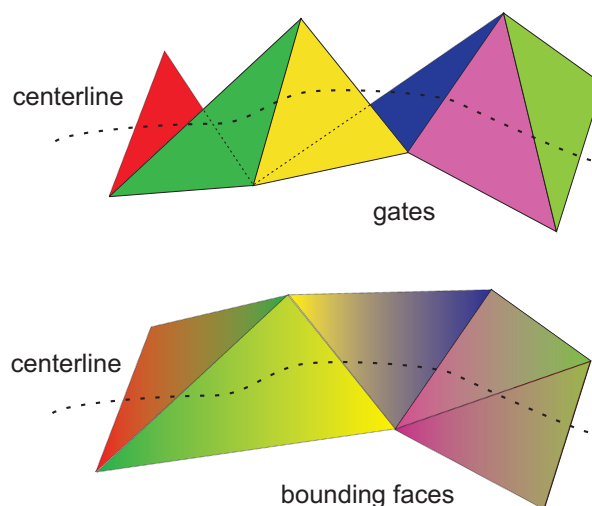

Figure 7: *Top: the set of successive gates obtained from the tetrahedrization. Bottom: the boundary faces computed from these gates.*

The reconstructed faces, the first, and the last gate form the resulting surface – these two gates are a part of the surface because as aforementioned we define a tunnel as a closed surface. The other gates are omitted.

4.1.2 Second step – refinement of the surface

After the reconstruction of a boundary, we refine the surface using subdivision.

As mentioned above the vertices of the tetrahedra are centers of atoms, which bound the tunnel. If we visualized the tunnel as a set of border triangles (faces), the inner part of the tunnel would hide some parts of atoms. However, to show the situation clearly we can thin the tunnel so that the atoms would not interfere with the boundary of the tunnel. Instead of adjusting the tetrahedra structure, we decided to use the Loop subdivision algorithm to obtain thinner boundary. Using it, the visual representation of a tunnel preserves its characteristics but it is appropriately transparent. We

obtain the surface, which does not interfere with the surrounding atoms, and we can omit the phase of thinning the tetrahedra input. Loop subdivision produces suitable results for our issue. Moreover, the results were much better than in case when we applied subdivision after the thinning the original tetrahedra model. In the latter case, the resulting tunnel was too narrow. Because we approximate
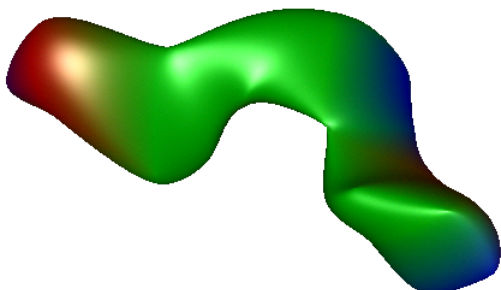

Figure 8: *The resulting surface created with the first algorithm based on the subdivision scheme. The surface is colored according to the surrounding atoms.*

the tunnel as a closed surface, we can apply the subdivision scheme uniformly.

Because of the thinning of the tunnel in the subdivision phase this algorithm does not show the tunnel as tight and wide as possible but it improves the clarity of the overall view. It also shows the potentially problematic parts of the tunnel – the narrowest parts which limit the size of a molecule of substrate entering to the active site of the molecule (Figure 8).

Another feature, which can enhance the detection of the best tunnels, is the coloring of the tunnel. It contributes to the better orientation in the space with the molecule. A color of an atom corresponds to its chemical element (each element has its unique color). We color triangles of the surface according to the atoms closest to this triangle. The distance of the atom from the triangle determines the color contribution of the atom to this triangle and the final color is blended from the colors of the atoms which influence this triangle (see Figure 8). Considering chemical importance of this property, we can derive it from various attributes of the amino acids. The atoms forming the tunnel belong to some amino acid and these structures have various characteristics, which influence their behavior, such as the hydrophobicity/hydrophilicity of an amino acid. Hence, the definition of the surrounding atoms is very important.

The greatest benefit is that this algorithm provides very good results in cases when we visualize more than one tunnel (Figure 9). With this approach, we can clearly visualize the correlation between tunnels and their mutual circumfluence.

When the biochemists determine the most important tunnels in protein, it is advantageous to highlight the chemically relevant tunnels and deemphasize the tunnels, which are not so crucial. However, biochemists need the less important tunnels also visualized. Our experiments showed that the suitable arrangement is to visualize surfaces of the important tunnels as a set of triangles and the less important ones as a set of points (see Figure 10).
This representation can possibly form the basis for the visualization of the tunnel transformations in time space when we have to emphasize the important movements of the tunnel.
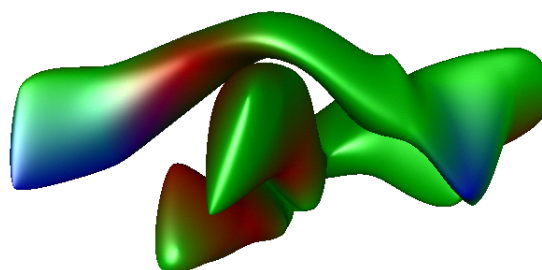

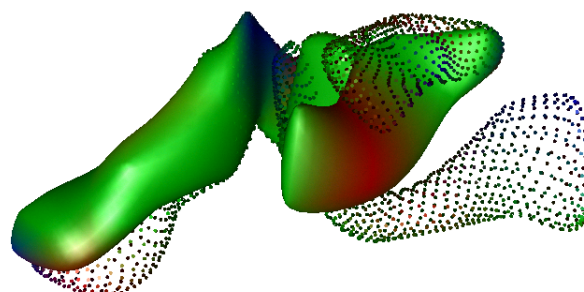**Figure 9:** *Three tunnels visualized with the first algorithm.*

Figure 10: *Two important tunnels visualized as triangle surfaces, other tunnels shown as sets of surface points.*

4.2 Second approach – visualization of a tight tunnel boundary

This approach we designed for the purpose of the visualization of the exact tunnel. When the biochemists determine the most relevant tunnels, the next step is the analysis of the behavior of the tunnel with regard to the presence of the substrate molecule. In this phase, the accurate visualization of the tunnel boundary is essential. With the currently used computation algorithm, which constructs the tunnel as a set of spheres it, is possible that we omit some relevant tunnel because it is too narrow. However, the real width of this tunnel could be greater. This is illustrated in the Figure 14. Apparently, if we sample the centerline of the trajectory incorrectly (and this can often occur because we do not have a method for the detection of the best sampling function) we may significantly reduce the resulting width of the tunnel compared to the real one.

The idea of this technique is to visualize the exact tunnel with respect to the tetrahedrization. Figure 11 shows the boundary of the tunnel formed by the border triangles (faces) from the tetrahedrization and the atoms.

This algorithm produces more accurate surface of a tunnel – it focuses on the visualization of the exact boundary. It has the following steps:

1. reconstruction of boundary triangles

2. traversing the space that contains the tunnel and the reconstruction of the tunnel surface with respect to atoms
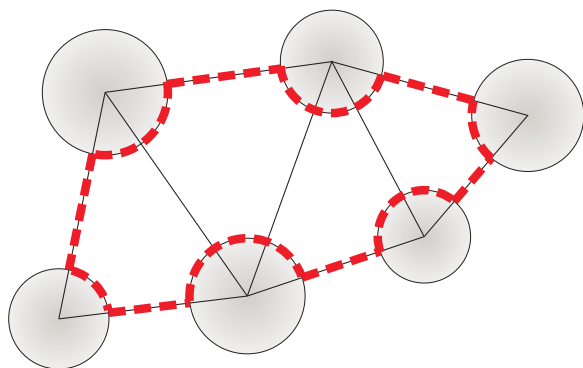
Figure 11: *The border of the tunnel is formed by faces and atoms.*

4.2.1 First step – getting faces from tetrahedrization

The first phase is identical to the first algorithm – the reconstruction of boundary triangles from gates.

4.2.2 Second step – sampling of the space and surface reconstruction

In the second step we compute the bounding box of a tunnel as the minimal and maximal values in all three axes. The bounding box serves as a limitation of a space, which we will traverse to obtain the boundary of the tunnel. Then we define a cube with a specific size (volume element) which defines the accuracy of the sampling. We call this cube a *probe*.

First we lexicographically sort all the atoms which form the boundary of the tunnel and for each atom we create a list of faces which are incidental with this atom. We set a starting point of the sampling as the bottom front left corner of the bounding box and we proceed as follows. We consider the bounding box of the tunnel divided into slices in one axis. Every slice we define as a two-dimensional array of probes. We traverse this array and for each probe, we determine the type of the probe. It can be one of the following types:

1. the whole probe is inside the tunnel

2. the whole probe is outside the tunnel

3. the probe is on the boundary between the inner and outer part of the tunnel

In the first two cases the probe is not important for our purpose (it does not form the surface of the tunnel) so we can omit them. The third case defines the boundary of the tunnel (highlighted in the figure 12) and this probe will form the surface of the tunnel.
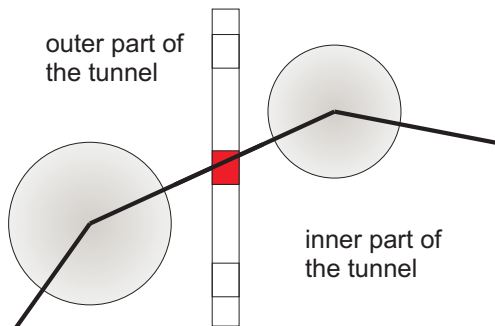

Figure 12: *Analysis of the type of probe at a single slice.*

After the analysis of the currently processed slice, we have to update the following two data structures where we store the lists of current atoms and faces.

1. **The list of current atoms** – the atom is current (active) in slices which include the value in the range

$$[C_a - radius, C_a + radius]$$

where C_a is the center of the atom.

2. **The list of current faces** – the face is current when any of its atoms are current or not processed yet. For the detection of the current faces, we use a counter attached to every face, which determines if the face was completely processed. When we detect some atom as active, we increase the counter for all corresponding faces. The face is inactive when its atoms are inactive and the counter has the value of three (all three vertices of the face were processed).

The next step is the reconstruction of the surface of the tunnel. For this purpose we use a modified Marching Cubes algorithm [Chernyaev 1995] which provides better results that the algorithm described in [Lorensen and Cline 1987] (produces the surface without holes). After the processing of all slices, all atoms and faces inactive, we obtain a set of polygons as a result. These polygons represent the exact surface of the tunnel.
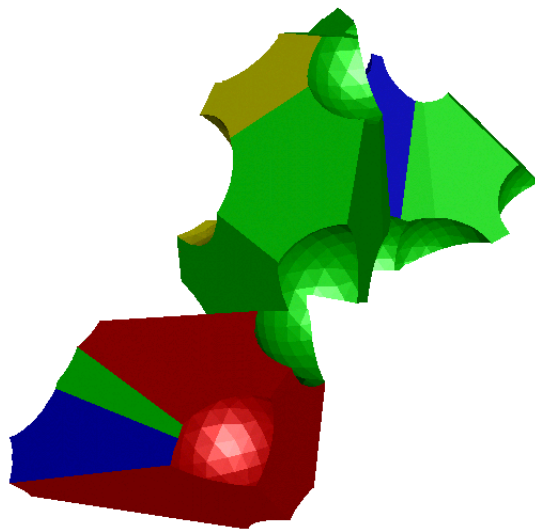

Figure 13: *The exact boundary of a tunnel visualized by the second algorithm.*

5 Results

The first method allows the better understanding of the shape of the tunnel and the relationships between neighboring tunnels. It serves as the starting point for the biochemists in the process of tunnel suitability determination. The second technique provides the most precise boundary of the tunnel. We use after we specify the most suitable tunnel and the following step is to bring the substrate molecule to the active site of the protein. In this phase, the visualization of the exact boundary is essential.

Both algorithms we implemented in the Java programming language. For the visualization part, we used OpenGL. Our algorithms
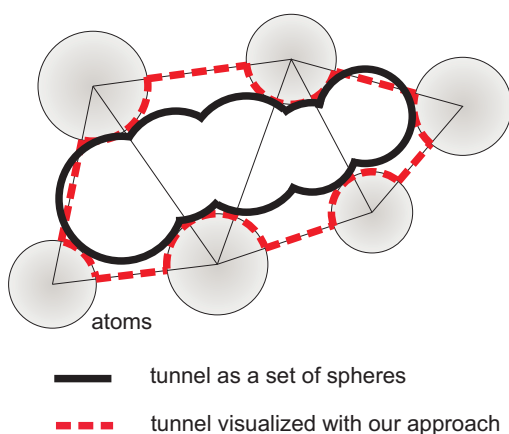
Figure 14: *The comparison between the visualization using a set of spheres and our second approach. The black full line shows the surface visualized as a set of spheres and the red dashed line represents surface obtained from our algorithm. The differences between these two surfaces form the free space added by our approach.*

achieve a significant improvement in comparison with the existing methods mainly in the visualization of many tunnels (see Figure 15). While the previous methods do not express the behavior of tunnels in space, i.e. their mutual circulation, our approach captures all these aspects. In the Figure 17 you can see the results of our algorithms for two different molecules of proteins (tunnels visualized with our first algorithm). In the first figure, there are five tunnels and in the second figure, you can see seven tunnels.

The main contribution of our approach to the work of biochemists is the possibility to clearly visualize and choose the most suitable tunnel. For this purpose the appropriate technique is our first approach (see Figures 15, 16). Scientists can clearly distinguish between relevant and non-essential tunnels and they can focus on the most important tunnel and visualize it. With our second approach, they may see the more precise representation of the boundary of the tunnel. When designing new techniques for tunnel visualization, we closely cooperate with biochemists. However, they usually cannot define their needs for visualization. Therefore, our task is to estimate and design the appearance of the tunnel characteristics and let biochemists asses the results.

6 Conclusion and Future Work

We proposed two novel approaches to the visualization of tunnels in molecules of proteins. These structures are very important mainly in biochemistry and pharmacy. Their visualization helps the scientists in better exploration and understanding of such a complex structures as proteins are. With our approach based on the Voronoi diagrams and Delaunay tetrahedrization, users perform analysis of proteins leading to the computation of tunnels. We visualize the tunnel data with novel techniques. First, we reconstruct the triangles, which bound the tunnel (faces), and then we apply one of our algorithms. The first algorithm subdivides the set of faces and the resulting surface provides a very good representation of the shape and the direction of the tunnel. From the second method, we can obtain results, which are more precise, and the surface represents more voluminous boundary of the computed tunnel. In comparison with the existing methods, our algorithms represent more precisely the continuance of the tunnel, its width in particular places, the distinction between different tunnels in one molecule of a protein and the more precise surface representation. In this paper, we focused

on the visualization of the static tunnel. In the future, we want to include the fourth dimension to our problems and find the techniques for the visualization of a tunnel in the time space. We plan to develop methods for the visualization of events that can occur during the movement of the tunnel. The tunnel can connect with other tunnels, become wider, shrink in some parts or it even can close for some period or forever. All these events biochemists need to visualize and analyze. Future techniques should also solve clear visualization of more than one tunnel in time. The analysis of protein dynamics and its visualization is the most important goal for the future.

Acknowledgements

References

AKKIRAJU, N. 1996. *Molecule Surface Triangulation from Alpha shapes*. PhD thesis.

BARBER, C. B., DOBKIN, D. P., AND HUHDANPAA, H. 1996. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software 22*, 4, 469–483.

CHERNYAEV, E., 1995. Marching cubes 33: Construction of topologically correct isosurfaces.

DAMBORSKÝ, J., PETŘEK, M., BANÁŠ, P., AND OTYEPKA, M. 2007. Identification of tunnels in proteins, nucleid acids, inorganic materials and molecular ensembles. *Biotechnology Journal 2*, 1 (January), 62–67.

DELANO, W. L., 2002. Pymol: An open-source molecular graphics tool, March.

EDELSBRUNNER, H., AND MÜCKE, E. P. 1994. Three-dimensional alpha shapes. *ACM Transactions on Graphics 13*, 1, 43–72.

HUMPHREY, W., DALKE, A., AND SCHULTEN, K. 1996. Vmd - visual molecular dynamics. *Journal of Molecular Graphics 14*, 33–38.

LOOP, C. T., 1987. Smooth subdivision surfaces based on triangles, August.

LORENSEN, W. E., AND CLINE, H. E. 1987. Marching cubes: A high resolution 3d surface construction algorithm. *Computer Graphics 21*, 3 (July), 163–169.

MEDEK, P., BENEŠ, P., AND SOCHOR, J. 2007. Computation of tunnels in protein molecules using delaunay triangulation. *Journal of WSCG 15*, 1-3 (February), 107–114.

OLSON, A. J., AND SPEHNER, J.-C. 1996. Reduced surface: An efficient way to compute molecular surfaces. *Biopolymers 38*, 3 (March), 305–320.

PETŘEK, M., OTYEPKA, M., BANÁŠ, P., KOSINOVÁ, P., KOČA, J., AND DAMBORSKÝ, J. 2006. Caver: a new tool to explore routes from protein clefts, pockets and cavities. *BMC Bioinformatics 7*, 316 (June).
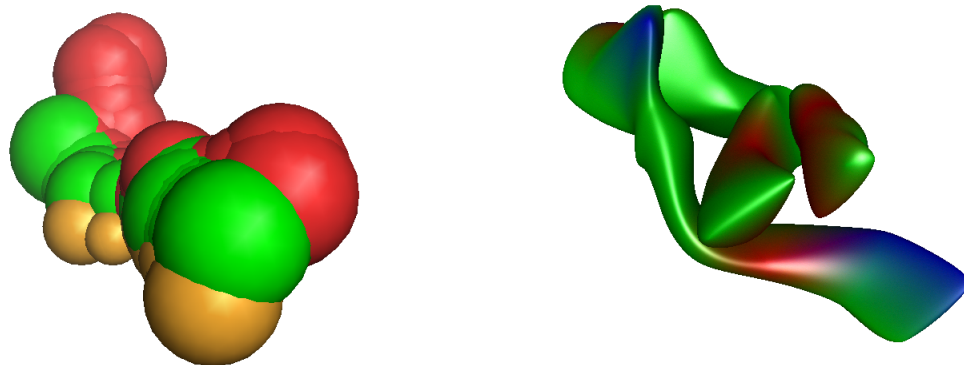
Figure 15: *Three tunnels visualized as a set of spheres in PyMOL (left); the same tunnels from the same position with our first approach (right).*
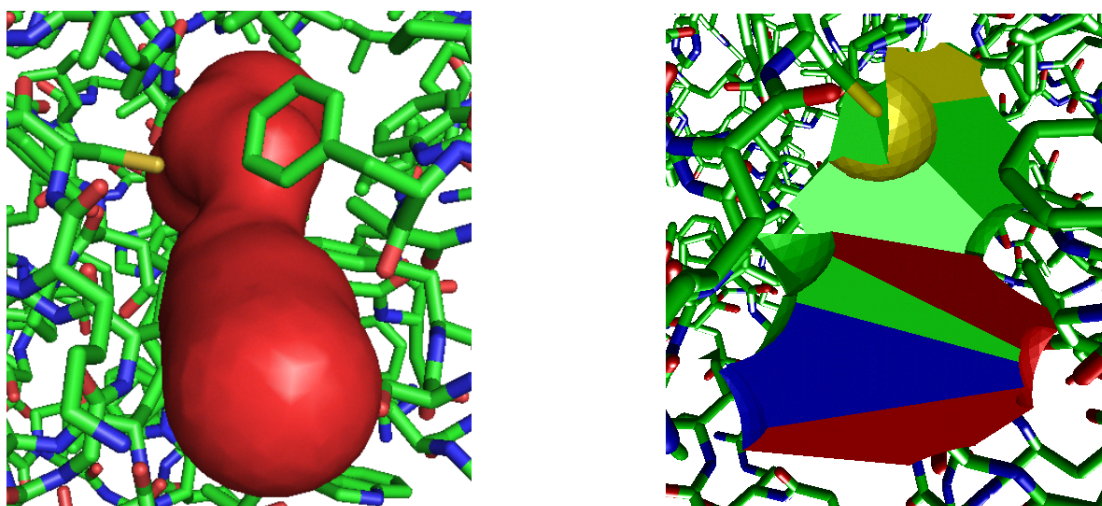

Figure 16: *A tunnel visualized as a surface obtained from smoothing the set of spheres in PyMOL (left); the same tunnel with our second algorithm (right).*
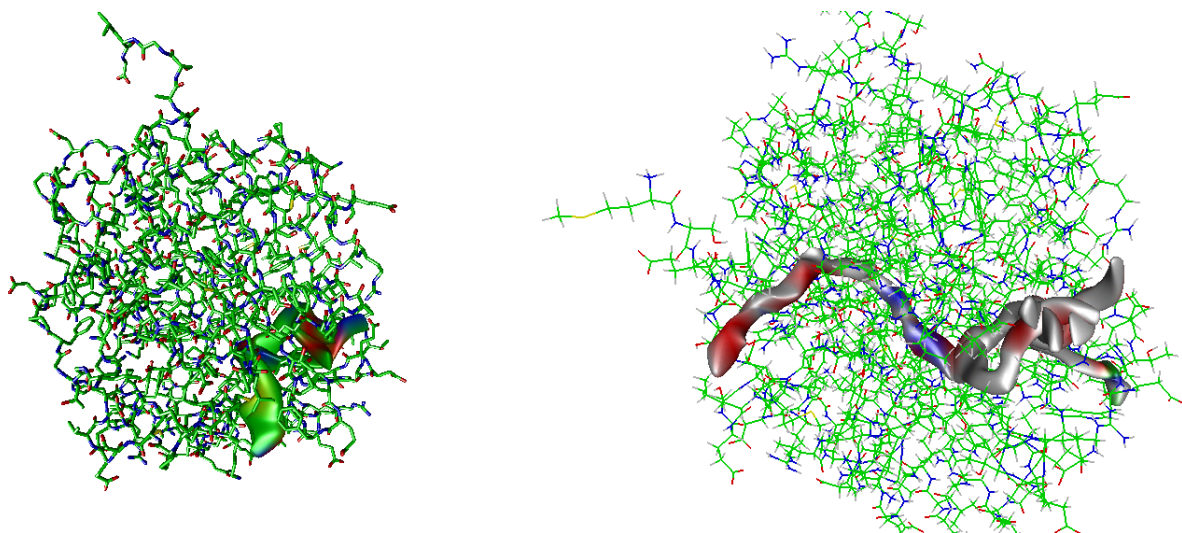

Figure 17: *Molecules with computed tunnels.*