Dynamic visualization of protein secondary structures

Matus Zamborsky*
Tibor Szabo†
Barbora Kozlikova‡

Faculty of Informatics, Masaryk University
Botanicka 68a
60200, Brno, Czech Republic

Abstract

Visualization of molecular structures and their characteristics represents a very popular and extensive area of computer graphics, in which the researchers are intensively interested for the last decades. During this time there have been developed many methods for visualization of molecules, which are trying to satisfy the needs of biochemists. These methods are mainly designed for the visualization of the particular molecule in a static position. For the more complex visualization methods special techniques have to be implemented in order to obtain a plausible method for visualization of secondary structures in time space.

This paper presents the possible solution of this problem by introducing the animation of the main backbone of the protein molecule onto which the particular objects representing the secondary structures are bound. These objects are replicated as many times as necessary and are closely connected to form a solid structure representing the whole molecule. In order to achieve high frame rates we are using the advanced GPU features, such as fragment and vertex shaders.

1 Introduction

Research in the field of computational biochemistry is inherently supported by computer graphics. The reason is quite straightforward - the product of very complex analyses performed by the biochemists is mostly represented as a set of numbers and letters. Without the proper visual appearance the biochemists would have to process all these data line by line and mostly have to have a good spatial imagination to interpret the data correctly. Integration of the computer graphics into this process meant the integration of the visual component, which enabled the biochemists to interactively explore the molecule in three-dimensional space. With proper visualization and manipulation techniques a user can pass through the molecule and see the real inner structure. Since the first attempts to visualize molecule in 3D space many new techniques had been developed, such as Van der Waals (VDW), Sticks, Balls and Sticks and others (see Figure 1). Each of them covers some specific needs of biochemists. But the common feature was that these methods visualize a molecule in a static position, so the dynamic movements of the molecule are not taken into account. However, these movements are very important, because they can significantly influence the behavior of the molecule.

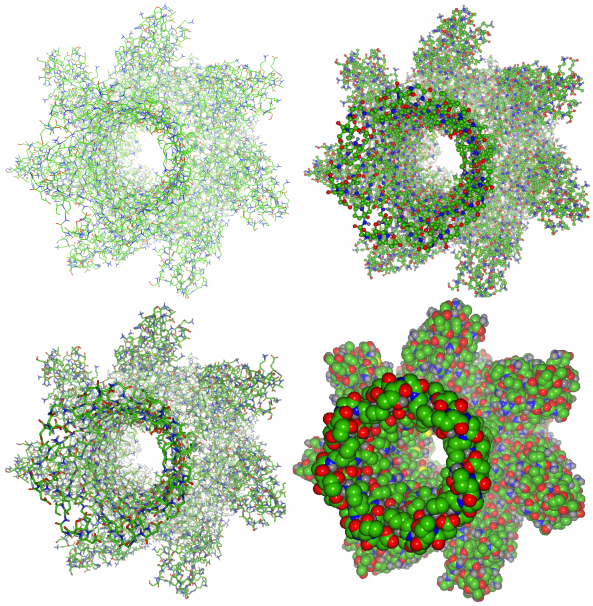

Figure 1: Examples of visualization methods of the 7ahl structure. Top row: left - Lines, right - Balls and Sticks; bottom row: left - Sticks, right - VDW

So now the following question suggests itself: why the methods for the dynamical visualization of molecules were not developed at the same time? The problem lied in the huge amount of data, with which an application for molecular visualization has to deal. The dynamic movement is represented by a set of thousands of snapshots which have to be processed and displayed in real-time. So the answer is, that in the past the computational power of computers was insufficient for such task. This situation

*xzambors@fi.muni.cz
†xszabo1@fi.muni.cz
‡xkozlik@fi.muni.cz

changed rapidly in the past years and now we are able to handle these data and visualize also the dynamic movements of the molecule.

In this phase, the extension of the current static visualization techniques to the dynamics was essential. Biochemists naturally want to preserve and use methods designed for a static molecule because they were designed many years ago to satisfy their demands. Our goal was to extend these methods and use them for displaying of the dynamic movements.

The simplest approach is to visualize the snapshots representing the movement continuously, one after another. The problem is that these snapshots were taken in some time steps which are not dense enough to show a smooth movement. In order to achieve the smooth animation of the movement we have to involve some additional techniques.

The extension of many of the existing methods is quite straightforward. The simple interpolation of the objects representing the molecule between the snapshots is sufficient to visualize movements smoothly. Among these techniques, the Balls and Sticks, Sticks or Lines methods come under this group. However, one of the mostly used techniques for protein visualization, called Cartoon, cannot be so easily extended to the dynamics. The Cartoon method displays so-called secondary structures detected in protein molecule, which add some level of abstraction to the visualization. It omits the displaying of all the atoms of the molecule and concentrates on the spatial configuration and the chemical dependencies between the parts of the protein chain (see Figure 2).

Figure 2: Cartoon method for secondary structures visualization

In this paper we would like to present our approach to the dynamic visualization of secondary structures. The simple interpolation between the snapshots of the Cartoon animation produces inadequate amount of triangles and the real-time visualization is therefore almost impossible to achieve. For that reason we have chosen another approach, which will be described properly in the following sections. The main idea lies in the animation of the backbone of the protein molecule (explained in the Protein Structure section (3)), onto which the model of the particular secondary structure is bound.

In the remaining sections of this article the reader can find the following information. In the very next section the current approaches to the secondary structures visualization are mentioned. Section 3 is dedicated to the short description of the protein structure, which is important for understanding the definition of protein backbone. It is used in our approach for the animation of protein movements. In the following section the process of secondary structures detection is described as well as the types of secondary structures. After this section the description of our algorithm follows. The last section contains conclusion, possible future extensions and also our results.

2 Related Work

Almost every existing application for molecular visualization provides users with the Cartoon method. From a huge amount of existing applications we will mention commonly used PyMOL [4], VMD [6], TexMol [1], GRASP [11], RasMol [13], MOLMOL [9] and many others.

According to this vast amount of applications there is no wonder that many different techniques for the visualization of secondary structures have been developed and implemented. In this section some of the existing approaches will be mentioned. The resulting appearance of secondary structures is very similar, the difference is in the technique used for generation of secondary structures. Although the detailed description of various objects representing secondary structures forms the content of the Secondary Structures (SS) section (4), for the better understanding of existing techniques some short explanation will be useful. According to the chemical dependencies between the atoms, we distinguish two main structures in proteins - alpha-helices and beta-sheets. Alpha-helix represents the helical structure of some specific part of the protein chain. Beta-sheet consists of several beta-strands which together represent the planar character of some parts of the chain. Not all the parts of the protein chain are components of some helix or sheet. These sequences are called turns (or coils) and join all helices and sheets to form a single protein chain. Figure 5 shows the examples of these structures.

TexMol application uses the impostor-based method for the visualization of secondary structures. Details of this method can be found in Bajaj et al.[1]. This method is very effective, although the results displayed are not very appealing in comparison with generating real 3D objects.

Authors of the article [10] present another approach. All the secondary structures - alpha-helices, beta-strands and turns - are modeled using non-uniform B-splines. The

control points of the spline coincide with the positions of C_α atoms in the chain, so this spline forms a shape of the particular secondary structures.

Many of the visualization methods simplify the task of alpha-helix visualization by introducing cylinders instead of helices. This method is used also in Hussein [7], where helices are visualized as cylinders between the first and last C_α atom of the helix. Sheets are drawn using Bezier curves, where again C_α atoms form the control points of the curve. The connections between helices and sheets are created using Hermite splines in order to form the continuous chain.

3 Protein Structure

All the protein molecules consist of the one or more chains of connected aminoacids. The structure of the chain is always the same: two carbon atoms, one nitrogen and one oxygen atom form one unit of the main chain (or the backbone) of the protein. Another part of this unit, the side chain, is formed by some specific aminoacid (also called residue), which influences behavior and spatial configuration of the protein. These units are connected via peptide bonds and together form a large chain. A single protein molecule can contain more such chains in its structre. Figure 3 shows two units of the chain (closed in the grey bubble). They are connected to each other via peptide bonds. The violet circle depicts a proper aminoacid. The most important movements in the protein take place at the backbone of the molecule. The movements of this structure serve as a basis for our new algorithm. Central atom of the whole unit is also the most significant and is called C_α - onto this atom the whole aminoacid group is bound.

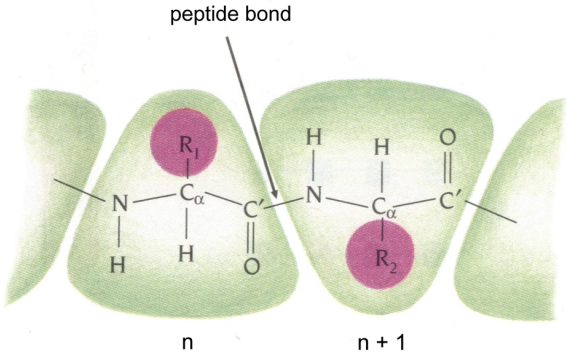

Figure 3: Segment of the protein structure (taken from [3])

4 Secondary Structures (SS)

Secondary structures of the protein perform some level of abstraction in the visualization process. The most detailed displaying method for protein visualization shows all the atoms and bonds of the whole molecule. However in many cases this representation can be too detailed and the user rather would like to observe the overall appearance of the molecule. This can provide the method called Alpha trace (4). This method displays only the backbone of the protein, which means, that the C_α atoms are connected together to form a long fibre representing the protein chain.

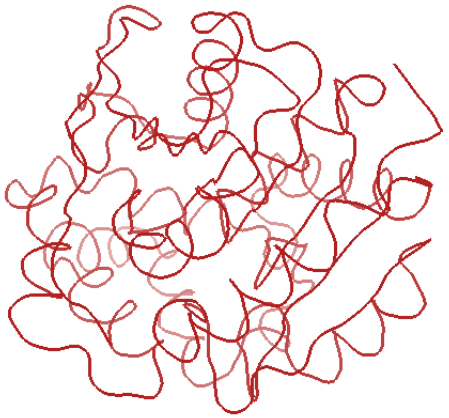

Figure 4: Alpha trace visualization method on 1cqw molecule

Secondary structures lie between these two extreme representations: they do not display the atoms of the molecule and provide the user with more information than in the case of Alpha trace. The main idea is to enhance the Alpha trace representation with some additional information about the chemical dependencies between atoms. This information is included among the secondary structures. To explain it more clearly, in alpha-helix there are some chemical bonds between the atoms lying in the neighbour turns of the helix. In beta-sheet, this situation arises between the atoms of the neighbour strands. All these dependencies are very important for the biochemists to understand the structure and behavior of the protein.

As it was already mentioned, there are two basic types of secondary structures, alpha-helices and beta-sheets. These two structures are connected together to the protein chain using the fibre called turn (or coil).

Alpha-helix is usually visualized in two possible ways. Basic simplified method displays the helix as a cylinder (see Figure 5, top right). The drawback of this method is that this visualization style does not take into account the shape of the helix. The more precise method displays also the curvature of the helix and the actual helix shape presents the real form much better. This curvature is given by the position of the backbone of the helix.

Beta-sheet is displayed as a set of beta-strands, which are situated on a curved plane. Each strand has its starting and ending part, which are clearly marked with an arrow. Figure 5 (bottom left) shows the typical visualization of such strand. Strands are also curved according to the posi-

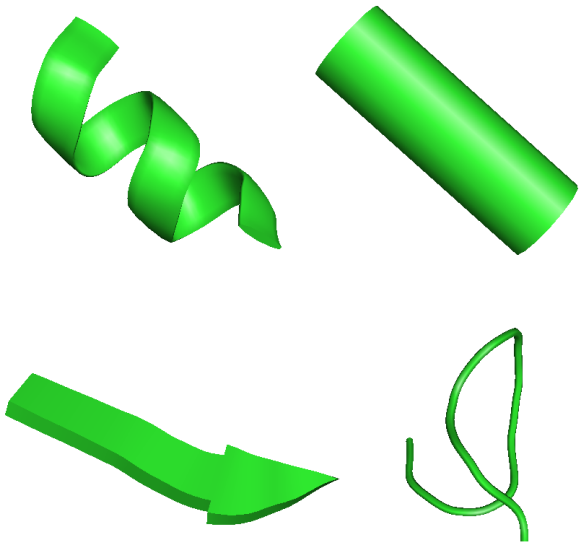

Figure 5: Visualization of secondary structures: Top row: left - alpha-helix, right - alpha-helix as cylinder; bottom row: left - beta-sheet, right - turn

tions of C_α atoms of these strands.

In order to present a protein molecule as a continuous chain, another additional structure has to be involved to connect the created helices and strands. The most suitable object is simple curved tube called turn or coil. This tube passes through the positions of C_α atoms, which are not a part of some helix or strand. The example of such turn can be seen in the bottom right picture of Figure 5.

These visualization styles for the secondary structures have been designed and used for many years and they suit to the needs of biochemists. Our goal is to use these objects for the visualization of the dynamic movements of the protein molecule.

5 Algorithm for SS Visualization

In this section our approach to the dynamic visualization of secondary structures will be explained.

5.1 Secondary Structures Computation

Before we start with the description of the visualization phase, we have to mention the actual secondary structures detection process. In our case we are working with the molecules in the PDB (Protein Data Bank) format ([2]), where the molecule is basically described as a set of atoms and their positions. Some additional information is also provided, such as the connection between atoms or the position of secondary structures in the chain, but this information is optional and we cannot rely on the presence of it in each PDB file. Therefore, first of all we have to apply an algorithm for secondary structures detection in order to obtain their positions in the chain.

Various algorithms for the secondary structures detection have been developed, such as DSSP ([8]), STRIDE ([5]) or DEFINE ([12]). These algorithms may perfom better on some specific structures, but generally they are giving similar results. Therefore we have chosen the DSSP (Define Secondary Structure of Proteins) algorithm and included it to our system. This geometrically based algorithm processes the coordinates of atoms in the PDB file. On the basis of this information together with the dihedral angles in the backbone and hydrogen bonds in the protein, it defines the position of secondary structures in the chain. As the output of this algorithm the user obtains the sequence of all the aminoacids of the protein marked with their secondary structure affiliation.

5.2 Visualization

After the secondary structures detection phase their visualization follows. Our goal is to display not only the protein secondary structures in the static position but also to visualize a movement of this structure. Simple visualization of the snapshots representing the state of the molecule in some time steps is not sufficient because it does not provide a smooth animation. Using the interpolation between two snapshots leads to enormous amount of triangles. As a consequence of this, real-time animation of the movement is very hardware-dependent. In our approach, we are trying to overcome all these problems and visualize the smooth animation in real-time and also enable the user to shift the animation slider in order to explore whatever part of the animation.

Before the actual animation process we have to prepare the objects from which the proper secondary structure will be created. More specifically, in some application for 3D modeling we create the patterns of the beginning, middle and end part of the each secondary structure, as is shown in the Figure 6. The pattern representing the middle part is then replicated as many times as necessary in order to create the secondary structure of the desired size.

The algorithm itself then processes the aminoacid chain. For the aminoacids which were detected by the DSSP algorithm as parts of some secondary structures we attach the particular pattern. In order to animate the movements of the secondary structures, the vertices of the pattern are stored in the relative position with the central C_α atom of the aminoacid.

For each aminoacid, the particular secondary structure segment is attached and these segments are blended together in order to create a solid model of the secondary structure. The animation itself then can be processed only by following the movements of the protein backbone (C_α atoms), which notably simplifies the whole process of animation.

The following pseudocode shows the computation of the matrix transforming the segment from the local coor-

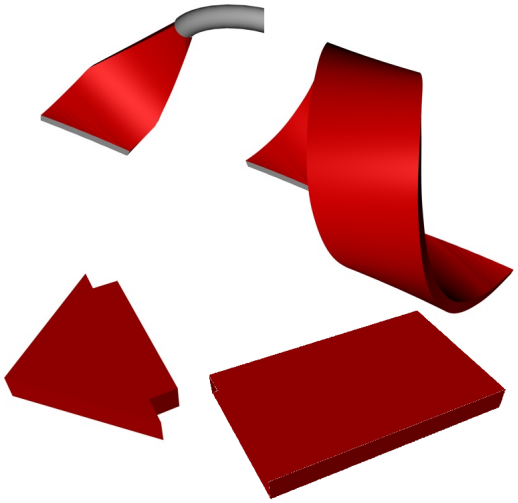

Figure 6: Patterns for the creation of secondary structures: from top left to bottom right: end part of helix, one turn of the helix middle part, beginning of the strand, middle part of the strand

dinate system to the space given by two carbon atoms and one oxygen atom. In this step no translation is performed yet.

```
CALCULATE_ROTATION_MATRIX(POSITION carbon1,
          POSITION carbon2, POSITION oxygen)
     BEGIN
          xdir = carbon2 - carbon1;
          ydir = oxygen - carbon1;
          NORMALIZE(xdir);
          NORMALIZE(ydir);
          zdir = xdir CROSS ydir;
          ydirnew = xdir CROSS zdir;

          RETURN MATRIX(xdir, ydirnew, zdir);
     END
```

After this step, all segments of the secondary structure are processed (1), where the segment is defined as the part of the protein backbone between two neighbour carbon atoms.

For each such segment, the first (2) and second (3) rotations are computed (first = rotations defined by the previous and current carbon, second = rotations defined by the current and next carbon).

The next step of computation is the calculation of the length of the previous (4) and the current (5) segment. From both of them we take just their half-length for next computation.

```
FOR segment FROM structure                    (1)
  BEGIN
    firstrot = CALCULATE_ROTATION_MATRIX      (2)
      (lastcarbon, currcarbon, lastoxygen);
    secondrot = CALCULATE_ROTATION_MATRIX     (3)
      (currcarbon, nextcarbon,
       curroxygen);

    firstscale = LENGTH                       (4)
      (currcarbon - lastcarbon)/2.0;
    secondscale = LENGTH                      (5)
      (nextcarbon - currcarbon)/2.0;

    firstmatrix = MATRIX_FROM_S_R_T           (6)
      (VECTOR(firstscale, 1.0, 1.0),
       firstrot, currcarbon);
    secondmatrix = MATRIX_FROM_S_R_T          (7)
      (VECTOR(secondscale, 1.0, 1.0),
       secondrot, currcarbon);

    FOR vertex FROM segmentvertices           (8)
      BEGIN
        vertexpos1 = TRANSFORM                (9)
          (firstmatrix, vertex);
        vertexpos2 = TRANSFORM               (10)
          (secondmatrix, vertex);

        lerpfactor = vertex.x * 0.5 + 0.5;   (11)
        vertex = LERP(vertexpos1,            (12)
          vertexpos2, lerpfactor);

      END
END
```

After that, we create a matrix (MATRIX_FROM_S_R_T) which transforms given vertices according to the position of the previous segment and translates them according to the position of the current carbon atom (6). The same operation is performed also for the current segment (7). These matrices are composed using scale, rotation and translation in this order.

Then, for all the vertices of the given segment (8), which is stored (paralelly with the X axis, reaching from -1.0 to 1.0) as the part of the desired secondary structure, the transformation using the previously computed matrices is performed (9), (10).

The last part of the algorithm performs linear interpolation between neighbouring segments. The coefficient for the interpolation is set to the value of the X coordinate (11), which is in the range of (-1.0, 1.0) and we transform it to the (0.0, 1.0) range. Then the linear interpolation itself between the computed positions is performed (12). Vertices, which had in the previous coordinate system the value X = -1.0, are transformed using the first matrix. Onto the vertices with previous value X = 1.0 the second matrix is applied. All the other vertices between these two limit positions are adequately transformed.

The resulting secondary structures in this form are prepared for the animation process,which is straightforward. It operates in the same manner as for the static visualization of secondary structures. Animation is performed again using GPU shaders, where segments in the local coordinate space are sent to GPU for processing together with positions of carbon and oxygen atoms. The same operations are performed in the vertex shader, where the proper positions of vertices are computed and blended.

6 Conclusions and future work

In this paper, we presented one possible solution of the problem of protein dynamic visualization using the so called Cartoon model. This model is one of the most used and popular among the biochemists because it provides the user with the adequate level of abstraction. In comparison with other existing solutions,it depicts important dependencies represented by the secondary structures.

Our approach animates the protein structure according to the movements of its backbone onto which the proper secondary structure objects are bounded. Using this technique the resulting animation is smooth and satisfies the initial demands.

According to the various features influencing the performance and quality of results (such as the choice of algorithm for secondary structures detection) the comparison between the existing methods and our approach is difficult. Moreover, our algorithm mainly solves the problem of the dynamic visualization of secondary structures, which is completely absent in the existing applications.

Actually, this algorithm represents just a small part of our work in this area. Together with the group of biochemists we are developing a new application for protein analysis and visualization which should bring new methods and approaches to the visualization of these structures. In the future, we would like to combine the Cartoon method of visualization with other existing or new techniques which can facilitate the work of biochemists and speed up the process of finding new medications.

7 Acknowledgments

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

References

[1] Chandrajit Bajaj, Peter Djeu, Vinay Siddavanahalli, and Anthony Thane. Texmol: Interactive visual exploration of large flexible multi-component molecular complexes. *IEEE Visualization '04*, pages 243–250, 2004.

[2] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne. The protein data bank. *Nucleic Acids Research*, 28:235–242, 2000.

[3] Carl Branden and John Tooze. *Introduction to Protein Structure*. Garland Publishing, December 1998.

[4] Warren L. DeLano. Pymol: An open-source molecular graphics tool, March 2002.

[5] Matthias Heinig and Dmitrij Frishman. Stride: a web server for secondary structure assignment from known atomic coordinates of proteins. *Nucleic Acids Res*, 32:500–502, 2004.

[6] William Humphrey, Andrew Dalke, and Klaus Schulten. Vmd - visual molecular dynamics. *Journal of Molecular Graphics*, 14:33–38, 1996.

[7] Ashraf S. Hussein. Analysis and visualization of gene expressions and protein structures. *Journal of Software*, 3(7), October 2008.

[8] Wolfgang Kabsch and Chris Sander. Dictionary of protein secondary structure: Pattern recognition of hydrogen bonded and geometrical features. *Biopolymers*, 22:2577–2637, 1983.

[9] R. Koradi, M. Billeter, and K. Wuthrich. Molmol: a program for display and analysis of macromolecular structures. *J Mol Graph - Journal of Molecular Graphics*, 14(1), Februar 1996.

[10] Oliver Kreylos, Nelson L. Max, Bernd Hamann, Silvia N. Crivelli, and E. Wes Bethel. Interactive protein manipulation. *IEEE Visualization 2003*, pages 581–588, 2003.

[11] Anthony Nicholls, Kim A. Sharp, and Barry Honig. Protein folding and association: Insights from the interfacial and thermodynamic properties of hydrocarbons. *Proteins: Structure, Function and Bioinformatics*, 11(4):281–296, 1991.

[12] F. M. Richards and C. E. Kundrot. Identification of structural motifs from protein coordinate data: secondary structure and first-level supersecondary structure. *Proteins*, 3(2):71–84, 1988.

[13] Roger Sayle and E. James Milner-White. Rasmol: Biomolecular graphics for all. *Trends in Biochemical Sciences (TIBS)*, 20(9):374, September 1995.