



# Towards Real-Time Interactive Visualization Modes of Molecular Surfaces: Examples with Udock

Guillaume Levieux, Matthieu Montes

## ► To cite this version:

Guillaume Levieux, Matthieu Montes. Towards Real-Time Interactive Visualization Modes of Molecular Surfaces: Examples with Udock. IEEE VR 2015 workshop on Virtual and Augmented Reality dedicated to Molecular Science (VARMS), Mar 2015, Arles, France. 10.1109/VARMS.2015.7151723 . hal-01126603

**HAL Id: hal-01126603**

**<https://hal.science/hal-01126603>**

Submitted on 1 Feb 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Towards Real-Time Interactive Visualization Modes of Molecular Surfaces: Examples with Udock

Guillaume Levieux

guillaume.levieux@cnam.fr

ILJ, Laboratoire CEDRIC, EA4626,

Conservatoire National des Arts et Metiers

292 Rue Saint Martin, 75003 Paris, France

Matthieu Montes

matthieu.montes@cnam.fr

Laboratoire GBA, EA4627,

Conservatoire National des Arts et Metiers

292 Rue Saint Martin, 75003 Paris, France

## ABSTRACT

In the present work, we describe and discuss two interactive visualization techniques recently added to the Udock software. First, we propose to display a spherical projection of the protein's molecular surface properties. The resulting 2D map allows to get in one glance a global view of the protein. Second, we propose to let the user choose a specific level of detail when visualizing the protein's 3D surface. We describe a simple smoothing algorithm that achieves this goal in real time. These techniques are designed to enhance the usability of molecular visualization and interactive simulation software.

**Keywords:** Protein-Protein Docking, Interactive Simulation, Molecular Visualization, Molecular Surfaces

**Index Terms:** I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques J.3 [Life and Medical Sciences]: Biology and genetics—[K.8]: Personal Computing—Games

## 1 INTRODUCTION

Protein docking methods aim at predicting the geometry of macromolecular complexes that play a crucial role in biological processes. To perform such a task, molecular visualization is needed to extract all relevant information about the structural properties of the proteins to be docked. Since proteins are very complex objects, the total amount of information that needs to be visualized and integrated is very high. It is thus critical for molecular visualization software to provide the user with different visualization and interaction modes that are adapted to the kind of information the user is interested in.

The Udock software allows a quick and easy-to-handle exploration of the possible conformations of a protein complex [15]. When performing interactive human-driven docking, the user needs to generate a complex 3D mental model of the given proteins' structures and properties in order to propose relevant geometries of their possible complexes. The generation of this mental model makes interactive docking a very complex cognitive task. Udock is designed to help the user to build such a mental model, using adapted representations and manipulation techniques. Proteins are rendered as solvent excluded surfaces with a smoothed coloring of the electrostatic potential, so that the user understands the global shape and electrostatic properties of the protein. In Udock, proteins are docked using an original harpoon-like technique and the collisions between proteins are simulated in real time by a physics engine [4], allowing the user to try different docking poses in a quick and intuitive fashion. Finally a force-field based [2] interaction score is

computed and displayed in real time to guide the user during the whole interactive docking procedure.

The goal of Udock is to maximize its usability, as a molecular visualization and interactive simulation tool. The usability of a tool can be defined as follows: *effectiveness*: the ability of the users to complete a task using this tool, and the quality of the output, *efficiency*: the time or resources spent to complete the task and *satisfaction*: the subjective reaction of the user when handling the tool [8]. Our previous work with Udock was centered on effectiveness and satisfaction [15]. In the present work, we focus on the efficiency aspect of usability. We want the user to be able to manage the type and quantity of visualized information to invest his cognitive resources in the most efficient way.

In the last version of Udock, we have developed new interactive modes of representation of a protein's global shape and of its surfaces properties that will be presented hereafter: 1. interactive 2D maps of protein surfaces and 2. interactive smoothing of the SES. These interactive and real-time modes of representation will enhance the usability of molecular visualization and interactive simulation software.

## 2 RELATED WORK

Many interactive visualization software are available to visualize protein surfaces in real time such as Pymol [5], VMD [10] or Chimera [17]. They all provide standard representations of molecular surfaces such as Van Der Waals surface (or spacefill) and solvent excluded surface (SES or molecular surface [18]) with coloring options that highlight a particular property (atom types, [3], electrostatic, [21], or hydrophobic potential [14]).

In the Van der Waals surface representation, the atoms are depicted as spheres with a given radius (the Van der Waals radius). Its major drawback is that it does not represent the accessibility of the different atoms of the protein to potential other molecules with respect to the surrounding solvent. Hence, the work of Richards on solvent accessible surfaces (SAS) and solvent excluded surfaces (SES [18]) can address this issue. To construct the SAS, a spherical probe of a radius equivalent to a water molecule is rolled over the Van der Waals surface. The center of the probe defines the SAS, whether the surface of the probe defines the SES. The SES is the best choice of representation for molecular interactions since it shows the shape of the object and its accessible surface with respect to the surrounding solvent [13].

Due to the number of atoms constituting proteins, their surface is very complex to analyze and extracting relevant information about their shape is difficult. The most common effects used for enhancing the rendering are cast shadows and depth cueing but due to the complexity of the surface, they often fail to produce a high-level comprehensible picture of the shape to the user [19].

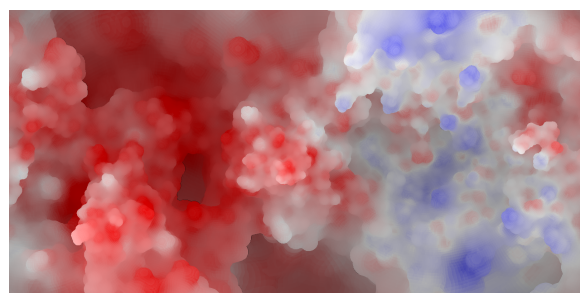
In order to abstract the complexity of the shape, Duncan and Olson developed a spherical harmonics approximation of protein surfaces [6]. To compute a similar abstraction and focus on the global shape of the object, we use in Udock a rendering technique

that allows the user to interactively reduce the surface’s level of detail in real-time.

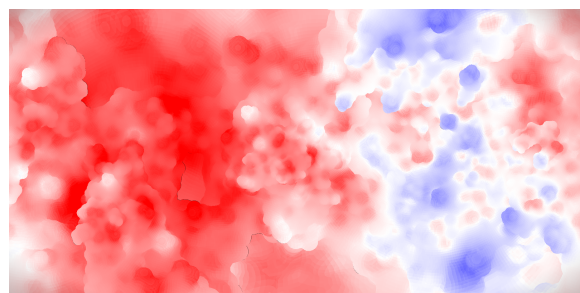
Different other approaches have been developed to reduce the complexity of protein surfaces for visualization purposes, in particular using 2D projections of molecular surfaces. These 2D maps were developed to characterize global surface properties (normal vectors, principal curvature [6]), compare the binding sites of proteins [7] [16] [22] or monitor the conformational changes of proteins occurring upon ligand binding [12] or during molecular dynamics simulations [11] [9] [20].

In Udock, we use 2D maps of the molecular surface of proteins for real-time interactive analysis of their properties and interactions. These 2D maps provide a global view of the whole protein surface properties and are complementary to classical molecular surface representations.

### 3 MAPS OF PROTEIN SURFACES



Conformal Mercator With Depth



Conformal Mercator Without Depth

Figure 1: Acetylcholinesterase (PDB ID: 1FFS) Conformal Mercator Maps

When performing interactive protein docking, the user has to identify the optimal geometry of a protein-protein complex. This complicated cognitive task relies on an accurate analysis of the structure and properties of the two proteins to be docked in order to identify the regions over the surface that would constitute the protein-protein interface. This analysis is performed using molecular visualization software that display the molecules in perspective projection.

However, when visualizing the whole structure of a protein, the perspective projection has the major drawback to never show the entirety of the structure: the backside of the protein is always hidden. To visualize this hidden part, the user needs to change his viewing position by manipulating the virtual camera. The backside of the molecule will then become visible, but at the expense of another part of the protein that will become the new hidden backside. As a result, the user has to constantly rely on memorization and manipulation when trying to analyze and integrate the information about the structure and properties of the whole protein.

To overcome this limitation, we propose to let the user switch to a spherical projection of the surface of the protein. This projection maps the entirety of the surface of the protein to a single 2D image, allowing the user to see the whole surface at one glance (fig. 1). In Udock, we use a triangulated mesh of the SES to visualize the protein. When the user switches to spherical projection, we use shaders to render a spherical projection of the SES as follows.

**Vertex Shader** In the vertex shader, we compute each vertex’s spherical coordinates  $(r, \theta, \phi)$ , with the origin of the sphere at the barycenter of the protein. Then, we transform these coordinates using different map projections. Currently, the user may choose between direct polar projection, Mercator conformal projection or Gall-Peters equal-area projection. As proteins are not totally spherical objects, some rays starting from the barycenter will hit its surface multiple times. When transforming the vertex, we compute  $z = 1 - r$  so that in that case, the depth culling keeps the farthest surface point. We also darken the vertex colour based on its depth to add a basic depth shading (fig. 1).

**Geometry Shader** Some SES triangles will have one vertex with  $\theta$  slightly over 0 and another vertex with  $\theta$  slightly below  $2\pi$ , resulting in large rasterized triangles covering a big part of the screen. In the geometry shader, we duplicate these triangles and offset each  $x$  component so that one triangle extends beyond the left of the screen and the other one extends beyond the right of the screen. These new triangles are then clipped by the pipeline.

Each type of projection allows to see the whole surface of the protein but at the cost of distortion of certain key structural features that gets stronger as we reach the edges of the map. To overcome the increased distortion problem at the edges of the map, the user can manipulate the rotation of the projected molecule. This allows to define the part of the surface that will be projected at the center of the 2D map and that will thus be less distorted.

### 4 SURFACE SMOOTHING

One way to provide users with a useful abstraction of a protein structure for molecular docking purposes is to hide the atoms that are deep inside the molecule, and thus less prone to interact with those of other proteins. The SAS and the SES provide such a representation.

In order to go further in this abstraction process, we developed an interactive surface smoothing method that will let the user define the level of detail of the surface in real-time. This allows the user to switch his focus from the whole shape of the protein to specific details of the surface (fig 2).

There exists documented ways to compute such a smoothed surface, like spherical harmonics approximation [6]. We currently rely on a basic three-passes approach.

**Pass 1 - Direct polar projection** First, we render the direct polar projection of the protein in a single pass, using the vertex shader and geometry shader described in the previous section, and store the depth buffer value in a texture. We use a direct polar projection because it corresponds to the most direct mapping between polar and texture coordinates.

**Pass 2 - Depth blurring** In the second pass, we render a single quad covering the whole screen, mapped with the depth texture obtained during the previous pass. We use a fragment shader to blur the depth texture by simply computing, for each fragment, the mean of the surrounding depth values, within a given radius. The size of the radius determines the strength of the smoothing. Since proteins are not totally spherical, if we use a constant radius, the fragments that are close to the center will be less smoothed than the fragments that are far from the center. The smoothing radius is thus scaled with regard to the depth of the given fragment.

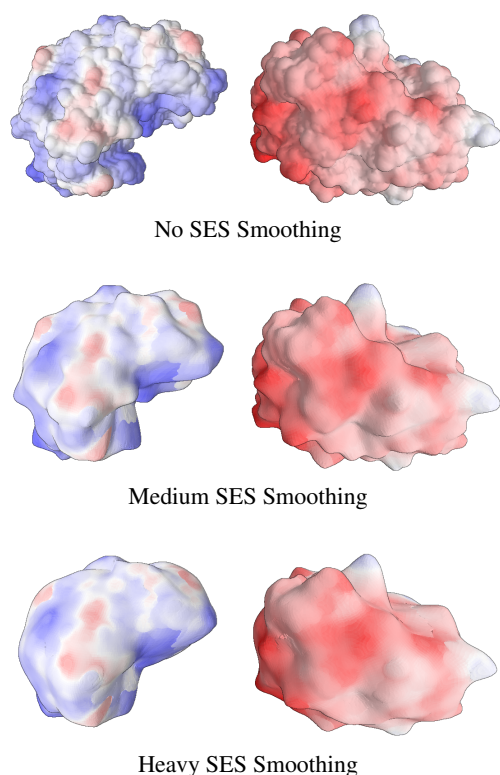


Figure 2: Barnase (left - PDB ID: 1BRS, chain A)/ Barstar (right - PDB ID: 1BRS, chain D) Solvent Excluded Surface at different smoothing levels

**Pass 3 - Smoothed SES rendering** In this last pass, we perform a standard perspective rendering of the SES, but we modify the vertex position to take into account the smoothed depth. In the vertex shader, we compute the polar coordinates of the input vertex. We use these coordinates to perform a lookup in the blurred depth texture, which gives us the distance at which a ray starting from the barycenter of the molecule and going to the current vertex would hit the smoothed SES. We use this distance to move the current vertex on the smoothed SES.

It is to note that the most time consuming pass is the smoothing pass, since for each pixel, we need to compute the sum of the surrounding pixels, and thus perform many texture lookups. However, we speed up this process by computing the sum of only a portion of these pixels, by varying the step size of the sum. The resulting smoothing is less accurate but can be computed in real time. Also, it is to note that we do not need to compute pass 1 and pass 2 for each rendered frame, but only when the user changes the smoothing radius. Once the blurred depth texture is computed and stored, we only need to render the smoothed SES with pass 3.

## 5 DISCUSSION AND FUTURE WORKS

**Map projections** There exists many map projections that could be used in Udock. We currently let the user choose between three of them, each one having its advantages and drawbacks (fig 3). Direct polar projection distorts local shape and surface area but has the most direct mapping between polar coordinates and texture coordinates. Mercator conformal projection maintains the local shape but distorts local surface area. Indeed, as illustrated in figure 3, the atoms appear more spherical in the Mercator map (shape preserving projection) than in the other maps, but as we reach the edge of the map, the atoms are getting bigger and bigger as their sur-

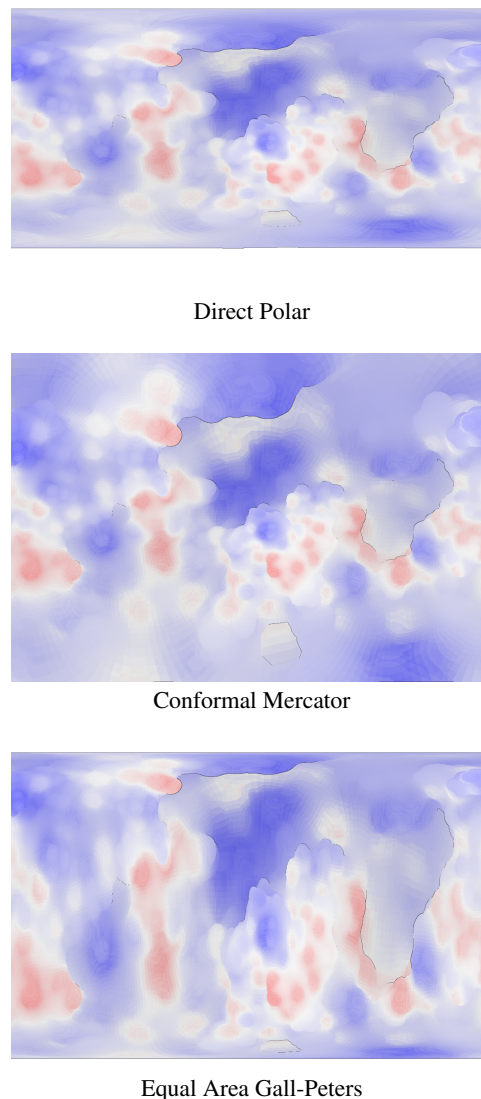


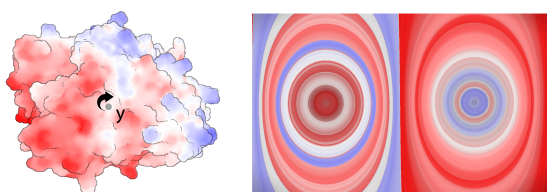
Figure 3: Barnase (PDB ID: 1BRS, chain A), Different Spherical Projections

face area gets more and more distorted. The Gall-Peters equal-area projection maintains local surface area but distorts local shape. As illustrated in figure 3, the atoms maintain a consistent surface area, but few seem to be spherical. Conformal Mercator map seems to be the best choice for visual inspection of protein structures with interactive protein-protein docking in sight since it is mainly about local shape matching, but this matter needs to be further investigated.

**Information on the map** To date, the smoothed electrostatic potential of the protein surface is displayed on the map. When the colour is shaded depending on the depth of the surface, the user gets more information about the shape of the surface. In the future, we plan to display additional information such as hydrophobic potential, atom types or secondary structures. We will let the user interactively select different layers of information to be added to the projected image.

**Manipulation techniques** The map projections we use in Udock suffer from distortion at their edges. To overcome this drawback, let the user chose which part of the surface will be projected





This figure presents the modifications of a 2D Conformal Mercator map when applying a rotation to the acetylcholinesterase protein (PDB ID 1FFS). We rotate the protein around the  $y$  axis, passing through polar coordinates  $(0, -\pi/2)$  and  $(0, +\pi/2)$ . We have then rendered the 2D Conformal Mercator map without clearing the frame buffer, to display the corresponding trajectories. We can clearly see the rotation axis and non intuitive rotation of the pixels.

Figure 4: Manipulation issue of Spherical Projections

at the image's center where the information is the less distorted. However, manipulating projections interactively is not straightforward. As illustrated in figure 4, applying a rotation to the protein before its projection often modifies the 2D map in a non intuitive fashion: the pixels of the map will often be translated in diverse directions. For instance, if the proteins rotation is applied along the axis passing through the north and south pole of the sphere, then it only offsets the longitude of each point, translating the 2D map on the  $x$  axis. This is a straightforward manipulation. But any other rotation will modify the latitude of the surface points, and the resulting map modification becomes less intuitive. Indeed, when any point of the surface lowers its latitude, the corresponding opposite surface point raises its latitude. In the 2D map, each surface point  $y$  coordinate is a function of its latitude, and thus some points will start to go up on the  $y$  axis while others will going down at the same time. Figure 4 depicts these trajectories, and we clearly can see the rotation axis, that differently modifies the two halves of the 2D map. We thus need to define an adapted interaction scheme for spherical projections.

**Interactive smoothing** In the current version of Udock, the shape of the SES can be smoothed interactively with the electrostatic potential of the original SES. As a result, since we move the vertices without recomputing the electrostatic potential on the smoothed surface, the displayed potential is less accurate. We plan to investigate how to smooth in real time the information displayed on the surface.

**Usability** One of the major next steps of this work is to evaluate the usability of both the 2D maps and the smoothed surface rendering. We postulate that it will be easier for the user to manipulate and mentally model the shape and properties of the studied proteins. To evaluate this, we propose to measure the performance of the users on interactive protein docking on reference proteins with known interaction geometry. By comparing the quality of the docking poses generated with the same amount of time, with and without the use of surface maps and surface smoothing, we may be able to determine whether these rendering techniques are really helpful to the users. We also plan to evaluate the usability of the system using subjective assessment like the System Usability Scale [1].

## 6 CONCLUSION

In the present work, we propose to display projections of a proteins structure and surface properties in order to alleviate the cognitive and manipulatory load required to create a mental model of its possible interaction geometries with a given partner. Of course, allowing such a global view of a proteins surface and its properties

comes with certain drawbacks, like shape or surface distortions and highlights a need for specific manipulation techniques.

We have also proposed to allow the user to dynamically smooth the SES of the protein, letting him choose his desired level of detail of the surface. We have described a very simple algorithm that allows to create and display such a smoothed surface in real time.

The focus of this research is to enhance the usability of molecular visualization systems. The presented manipulation and visualization modes are designed to make a step towards this goal. Besides pushing further the abstraction of the representation of protein surfaces and of their properties, the next step of this research is to experimentally validate the impact of such techniques in terms of usability.

## REFERENCES

- [1] J. Brooke. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.
- [2] D. Case, T. Darden, T. E. Cheatham III, C. Simmerling, J. Wang, R. Duke, R. Luo, R. Walker, W. Zhang, K. Merz, et al. Amber 12. *University of California, San Francisco*, 142, 2012.
- [3] R. Corey and L. Pauling. Molecular models of amino acids, peptides, and proteins. *Review of Scientific Instruments*, 24:621–627, 1953.
- [4] E. Coumans et al. Bullet physics library. *bulletphysics.org*, 2013.
- [5] W. DeLano. The pymol molecular graphics system (2002) delano scientific, palo alto, ca, usa, 2002.
- [6] B. S. Duncan and A. J. Olson. Approximation and characterization of molecular surfaces. *Biopolymers*, 33(2):219–229, 1993.
- [7] D. W. Fanning, J. A. Smith, and G. D. Rose. Molecular cartography of globular proteins with application to antigenic sites. *Biopolymers*, 25(5):863–883, 1986.
- [8] I. O. for Standardization. *ISO 9241-11: Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs): Part 11: Guidance on Usability*. 1998.
- [9] R. R. Gabdoulline, R. C. Wade, and D. Walther. Molsurfer: a macro-molecular interface navigator. *Nucleic acids research*, 31(13):3349–3351, 2003.
- [10] W. Humphrey, A. Dalke, and K. Schulten. Vmd: visual molecular dynamics. *Journal of molecular graphics*, 14(1):33–38, 1996.
- [11] P. Koehl and J. Hass. Automatic alignment of genus-zero surfaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(3):466–478, 2014.
- [12] A. D. Koromyslova, A. O. Chugunov, and R. G. Efremov. Deciphering fine molecular details of proteins structure and function with a protein surface topography (pst) method. *Journal of chemical information and modeling*, 54(4):1189–1199, 2014.
- [13] M. Krone, S. Grottel, and T. Ertl. Parallel contour-buildup algorithm for the molecular surface. In *Biological Data Visualization (BioVis), 2011 IEEE Symposium on*, pages 17–22. IEEE, 2011.
- [14] N. Kurochkina and B. Lee. Hydrophobic potential by pairwise surface area sum. *Protein engineering*, 8(5):437–442, 1995.
- [15] G. Levieux, G. Tiger, S. Mader, J.-F. Zagury, S. Natkin, and M. Montes. Udock, the interactive docking entertainment system. *Faraday Discuss.*, 169:425–441, 2014.
- [16] K. Pawlowski and A. Godzik. Surface map comparison: studying function diversity of homologous proteins. *Journal of molecular biology*, 309(3):793–806, 2001.
- [17] E. Pettersen, T. Goddard, C. Huang, G. Couch, and D. Greenblatt. Ec meng, and te ferrin. 2004. ucsf chimera—a visualization system for exploratory research and analysis. *J. Comput. Chem*, 25:1605–12, 2004.
- [18] F. M. Richards. Areas, volumes, packing and protein structure. *Annual review of biophysics and bioengineering*, 6:151–76, 1977.
- [19] M. Tarini, P. Cignoni, and C. Montani. Ambient occlusion and edge cueing for enhancing real time molecular visualization. *Visualization and Computer Graphics, IEEE Transactions on*, 12, 2006.
- [20] P. Todd, S. Todd, F. F. Leymarie, W. Latham, B. Jefferys, and L. Kelley. Foldsynth: A physics-based interactive visualisation platform for proteins and other molecular strands. *BioVis 2011 IEEE Symposium on Biological Data Visualization*, 2011.

- [21] P. K. Weiner, R. Langridge, J. M. Blaney, R. Schaefer, and P. A. Kollman. Electrostatic potential molecular surfaces. *Proceedings of the National Academy of Sciences*, 79(12):3754–3758, 1982.
- [22] H. Yang, R. Qureshi, and A. Sacan. Protein surface representation and analysis by dimension reduction. *Proteome Science*, 10, 2012.