

Scientific Visualization of Chemical Systems

Richard E. Gillilan and Bruce R. Land

Visualization Group
Cornell Theory Center
Ithaca, NY 14853-380

Abstract

Within the past few years, computer graphics has played a decisive role in transforming the field of molecular modeling into an international industry. While computer scientists are conversant in the language of their own discipline, they are often unfamiliar with the terminology and unique graphical constructs of physical scientists. Similarly, physical scientists are often unfamiliar with the latest paradigms and technological advances in graphical computing. The successful marriage of Chemistry and Computer Graphics is an outstanding model of technology transfer and cross-disciplinary fertilization. Drawing on chemistry visualizations performed at the Cornell Theory Center, this workshop covers some of the terminology, tools and current problems that have evolved at the boundary between these two fields.

1 Introduction

Chemistry is a field with a long history of modeling and graphical representation. Nearly every student of Organic Chemistry learns the story of August Kekulé, who was puzzling over why carbon, of all atoms, could form so many compounds. According to his own account [1], one summer evening in 1854, he dozed off while riding on an omnibus. In his dream, he saw myriads of atoms "whirling in a giddy dance" before his eyes, forming ever longer chains. This was the beginning of a series of visions that would eventually lead him to the discovery of the structure of Benzene and other landmark principles of Organic Chemistry. Molecules by their nature are complex moving geometrical objects unseen to the naked eye. Because this microscopic world eludes our direct experience, models offer a visual language of sorts, which allows us to formulate questions, make statements and, in general, communicate.

Modern computer graphics technology allows researchers to share the experience and visions that were once trapped in the minds of the privileged few. Ideally, it extends our natural capabilities and frees our creativity from technical limitations of expression. Too often, however, computer graphics enters research at a late stage, mainly for purposes of presentation. Although chemists have been particularly good about developing practical research tools, there still exists a gap between tools of presentation and tools of discovery. Chemistry is nonetheless a model field in which computer graphics methodology has reached a high state of development. The unique constructs that have evolved have shed light on complex problems and yielded dramatic insights that might otherwise have gone long unnoticed. We provide a brief overview of rendering principles relevant to molecular graphics with emphasis on the new data-flow paradigm for graphical computing. Basic chemical terminology is then introduced, followed by a discussion of several of the most widely used graphical constructs in chemistry and their role in the discovery process. In the final section, we outline the steps that a researcher might take in developing their own scientific visualization using the IBM Visualization Data Explorer software as a tool.

2 Graphics principles and techniques

A great deal of what is now established interactive graphics technique for Chemistry can be traced back to the days of vector graphics machines when points and lines were the only renderable objects. It is remarkable just how much can be achieved with these simple forms and a rudimentary lighting model. The reason why chemists adopted this technology at such an early stage may be that molecules are represented quite well as simple stick figures. Even today, this is the default representation in most software packages.

High-speed workstations with specialized graphics engines have greatly expanded the realism of models, the primary tool available today being the polygon surface. When three vertex points, for example, define a triangular surface, one can model the effect of lighting by assigning the color of points interior to the triangle on the basis of incident light angles. The surface can also have an intrinsic color obtained by interpolation of colors defined at vertices. Advanced graphics systems are even capable of mapping raster images to polygon faces. This process is called *texture mapping*. One can just as well define solid volumes by specifying how points are connected together into a three-dimensional network. A connection list of this type is necessary for *volumetric rendering*, a method which is capable of rendering gas-cloud-like objects where each point of a three-dimensional volume has a data value. *Ray-tracing* and *radiosity rendering* have the greatest level of realism and the greatest demand for computing power and are capable of creating shadows and reflections. Few scientific visualization packages have implemented these methods since they are far too slow for interactive work; most professional animation software, however, has some capability for ray-tracing.

The most important component that realism has to offer scientific visualization is improved sense of depth. Depth cues are critical to understanding complex geometrical objects like an enzymes. The major depth cues, perspective, atmospheric effects (often called depth cueing), stereopsis, and rotational motion were all available in early vector machines, but surprisingly some are commonly neglected in contemporary visualization. Atmospheric effects, for example, make objects close to the viewer brighter [2]. Figure 1 shows the same molecule rendered with and without this effect. Note how the deep tunnel to the active site at center and two other recessed regions are essentially invisible without this effect. Stereopsis, rendering a different viewpoint for each eye, requires special hardware and doubles the rendering time, but many chemists use it regularly [3]. Rotational motion also contributes to depth-cueing since points closer to the viewer appear to move faster. This is easy to achieve in animation but fast rotation for interactive work requires a special rendering engine.

3 Chemical systems

One of the things that makes Chemistry visualization interesting and nontrivial is that atoms are so small that Newton's familiar principles of motion are

no longer always valid. The atom lives in a gray area of our understanding, halfway between the classical world of Newtonian motion and the novel, but reasonably well understood, world of quantum mechanics. Contrary to the impression many receive in introductory courses, the connection between classical and quantum-mechanical behavior in this regime is still not well understood and is the subject much recent research and controversy [4]. Nevertheless, atoms are basically hard spheres that sometimes stick together, and much useful chemistry can be understood in these terms!

An important example is the polypeptide. Amino acids are some of the basic building blocks of life. Each of the 22 common acids has an amino group on one end and an acid group on the other, allowing them to form long chains. Amino acids differ only in the particular molecule groups that hang off to the side of the chain (sidechains). Figure 2 highlights one amino acid in the middle of a long chain. Chemists are highly interested in the making and breaking of the *bonds* that hold atoms together so we usually represent bonds as lines or tubes.

There are times when the Newtonian description of molecules as balls and sticks is simply not appropriate. In computing the quantum-mechanical distribution of electrons, for example, it is more useful to view them as a continuous distribution rather than a set of points. This density of electrons is something like a ripple on a pond, only three dimensional. In graphics terminology this is a *scalar field*. The density value is sampled at various positions to create a set of points that can be connected for interpolation and rendering. The position and motion of atoms can also be represented quantum mechanically, though, being much heavier than electrons, atoms are accordingly more Newtonian in behavior. The lightest of atoms, Hydrogen and Helium, can participate in the strangest of quantum mechanical behavior: tunneling and diffraction. The simulation of atomic motion using Newton's equations is known as *molecular dynamics*. Quantum-mechanical simulation of this same motion is known as *quantum dynamics*, a much more difficult computational task.

In summary, there is no "real" representation for a molecule, only models designed to bring out important features.

3.1 Molecular graphics techniques

Perhaps one of the best and most dramatic illustrations of how simplified representations have enhanced our understanding of chemistry is the ribbon

model. We have discussed how amino acids join together in long chains to form polypeptides. Many larger polypeptides, however, appear as featureless blobs when displayed as ball-and-stick or space-filling models. The mass of tangled atoms and bonds is overwhelming to the eye and of little help in recognizing important structural features. Part of the confusion is simply due to the fact that amino acids have so many types of side chains which hang off in all directions. The underlying molecule itself is actually just a long repeating chain: α -Carbon, Carbon, Nitrogen, α -Carbon, Carbon, Nitrogen, \dots with short sidechains connected to the alpha Carbons. One additional feature of this backbone is that while the α -Carbon-Nitrogen bond can rotate freely, the Carbon-Nitrogen bond cannot. As a result, the α -Carbon, Oxygen, Carbon, Nitrogen, the next α -Carbon, and accompanying Hydrogen are often coplanar. The series of planes thus defined makes the amino acid chain behave more like a ribbon with the Carbon-Oxygen double bond parallel to the ribbon surface (see Figure 2). Early computer algorithms of this type were pioneered by Richardson [5] and further developed by Carson and Bugg [6]. An otherwise featureless protein blob so rendered becomes a recognizable collection of recurring structural motifs: the alpha helix, the beta sheet and so on. Incidentally, the problem of consistent ribbon orientation is an example of a common problem solved independently by chemistry and computer graphics researchers [7].

The molecular surface is another geometrical construction that has gained widespread popularity. Like the ribbon model, molecular surfaces are simplified representations that leave out information irrelevant or even confusing to the problem. Of all the atoms in a protein, usually only a few are actually exposed to the solvent. The rest remain buried deep within the molecule, completely inaccessible from the outside due to the close proximity of other atoms. Though select in number, the solvent-exposed atoms play a major role in the behavior of the protein in solvent as well as defining the overall shape.

A number of models have been proposed to study this aspect of protein structure, but one given by Richards [8] has been particularly successful. The actual implementation was worked out by Connolly [9] and is now in widespread use. Geometrically, the idea is straightforward. A sphere the size of a solvent molecule is used as a probe to determine which points can and cannot be reached subject to steric constraints. Mathematically, the surface is constructed in a piecewise fashion from hemispheres and portions of

tori. The different types of surface patches are created by the probe being in contact with different numbers of atoms simultaneously. The actual implementation is rather involved and will not be presented here but the advantages are clear: one obtains only the exposed "skin" of the molecule. Figure 3 is a cross section of the acetylcholinesterase molecule [10] showing the molecular surface, several important sidechains and a molecule of acetylcholine at the active site. This type of surface is used not only as a general guide to the shape of the molecule, but as an analytical tool for evaluating solvent effects and as a manifold for electrostatic potential calculations [11].

The technique of mapping a three-dimensional function, such as electric potential to a surface, is a recent invention which is gaining popularity. The fact that we can interpolate functions defined on a three-dimensional grid and find their value at any vertex of a polygon surface means that surface features such as folds and gorges can be correlated with property values through the use of color mapping. This aids in the identification of sites of interaction and molecular function. In electronic structure calculations, for example, the density of the lowest unoccupied molecular orbital (LUMO) can be mapped onto the surface of constant overall electron density to give an indication of the accessibility of sites for nucleophilic attack. In biophysics, electrostatic potential can be mapped to the molecular surface and color coded as has been done in Figure 3.

Surfaces of constant functional value, isosurfaces, have been in use for quite a long time. Originally constructed from stacks of contour plots, these surfaces are heavily used in X-ray structure determinations to guide the placement of atoms. Electronic structure computations have also long used this method. The discovery of the marching cubes algorithm [2] has only recently made it convenient to compute a renderable set of polygons on such a surface.

Visualization techniques invented for the study of fluid flow have proven valuable in other fields as well. The negative gradient of an electric potential is a vector field that can be treated much like the velocity field of a flowing fluid. Streamlines are simply the paths taken by particles flowing in the field. We have used this technique to study how charged substrate molecules might drift to enter the active site of the acetylcholinesterase enzyme [10]. The dramatic convergence of lines illustrates how the electric field may serve to funnel and concentrate charged species near the mouth of the active site (Figure 4). The effects of solvent on, and the detailed internal structure of the

charged species has been left out of the model as well as the motion of the enzyme. These effects are probably important but the principle physics of interest is captured.

As simulation technology improves, the number of new problems of representation will increase. Quantum dynamics simulations, for example, are now routinely carried out for systems of up to three variables, but four and more will soon be the norm. How can these functions of many variables evolving in time be visualized? As chemists simulate actual reactions more often, tools for examining the time-evolving electron densities will need to be developed [12]. On the biological and material science fronts, simulations are getting larger and larger in number of atoms included. Much of the mechanics of life takes place on a vast scale in comparison to the atom. Visualization techniques will be challenged with very large memory requirements. On all of these frontiers, the key to developing effective scientific visualization will ultimately be in understanding what should be left out and determining how best to represent what remains.

4 Building a scientific visualization

This section describes the four basic stages a researcher might use to develop a scientific visualization: *importation*, *exploration*, *customization* and *presentation*. Though IBM Data Visualization Data Explorer is used as an example throughout, we feel these stages are a good model for any visualization environment.

The process usually starts with the import of data from external sources. *Importation* is, in our experience, often the more frustrating and difficult part of the whole visualization process. To this end, any visualization environment should have one or more of the following components:

- specialized modules or options for convenient import of the most common data types encountered.
- general data-description language with graphical “point and click” interface to facilitate the import of generic tabular and tagged data formats.
- ability to send and receive data from external processes and to serve as a graphical front end to another program.
- ability to write customized import modules in a low-level language.

Since data arrives from multiple sources and is often expressed in different units or even different coordinate systems, some minor preparatory computations are usually required. In programs like IBM Visualization Data Explorer, these kinds of computations are easily implemented through the use of general compute modules.

Preliminary graphical display of the data is made with the default program settings. At this point, the researcher needs to identify how program settings and simulation variables, such as positions of cutting planes and values of isosurfaces, should be modified to provide a clearer view of the phenomenon. This is the *exploration* phase. Control and interaction need to be added to refine the simulation, extract useful information and explore newly discovered features. Good visualization environments will include advanced interactivity such as 3D cursors, picking and navigation controls. Easy manipulation of objects are critical in order for the researcher to translate the data set into a complete mental picture which can be used as a foundation on which chemical reaction scenarios are envisioned and tested.

Once familiar with the data set, users may write specialized macros or modules to make future visualizations with the same kind of data simpler and more convenient. This is the *customization* phase. Prior to the data-flow paradigm, a researcher had to choose between high-level applications, which were often rigid and problem-specific, and low-level programming, which is highly flexible but time-consuming and not always easy to modify and document. Data-flow programs such as IBM Visualization Data Explorer are modular in nature and have the power and generality of a language but are easier to use.

The final phase is *presentation*. The researcher has a story to tell, with all the traditional elements: the setting, the nature of the problem, the progression of success and failure leading to solution and finally, the solution. Few scientific graphics packages are designed, however, to control the flow of images in progression in the way that a professional animation package does. Scientific packages frequently do not have strong annotation and lettering capabilities, especially when it comes to video production. Graphics must also be presented in a medium suitable for larger audiences. Each medium has its own particular requirements and limitations but video is especially restrictive. Graphics that looks good on a computer screen invariably looks poor and illegible on a video monitor.

5 Summary

In many ways, Chemistry is a field primed for scientific visualization. The already established use of graphical representations and the intrinsic geometrical nature of the fundamental problems in the field paved the way for the adoption of new technology. The use of early primitive rendering types (point and line) were highly effective, so much so that they are still in use today. Both realism and speed are important factors that add to the researchers' ability to recognize shapes and spatial relationships. In our examination of two popular molecular graphics constructs, the ribbon and the molecular surface, we see that what is left out of a model is often as important as what is left in. The fact that molecules can exhibit both classical and quantum-mechanical behavior provides interesting challenges in visual representation. As more sophisticated theories of chemistry are realized in simulation, scientific visualization will play a key role in both interpreting results and giving form to ideas.

Acknowledgements

Special thanks to Wendy J. Bacon for editorial help and to Martin Berggren for technical assistance. We gratefully acknowledge the National Institutes of Health who support biomedical computing and visualization at the Cornell Theory Center. This work was conducted using the the Cornell Theory Center, which is funded in part by the National Science Foundation, New York State, IBM, and members of the Theory Center's Corporate Research Institute.

References

- [1] R.T.Morrison and R.N.Boyd, *Organic Chemistry*, 3rd Ed. Allyn and Bacon, Inc., Boston (1979)
- [2] J.D. Foley, A. van Dam, S.K. Feiner and J.F. Hughes, *Computer Graphics: Principles and Practice*, Addison-Wesley, New York (1990).
- [3] Larry F. Hodges, "Tutorial: Time-Multiplexed Stereoscopic Computer Graphics", *IEEE Computer Graphics and Applications*, March (1992)
- [4] M.-J.Giannoni, A. Voros and J.Zinn-Justin editors, *Chaos and Quantum Physics* Les Houches session LII , North Holland (1989)
- [5] J.S. Richardson, "The anatomy and taxonomy of protein structure", *Advances in Protein Chemistry* 34 167 (1981)
- [6] M. Carson and C.E. Bugg, "Algorithm for ribbon models of proteins", *Journal of Molecular Graphics* 5 (2) 103 (1987)
- [7] Jules Bloomenthal, "Calculation of Reference Frames Along a Space Curve", in *Graphics Gems*, edited by Adrew S. Glassner, Academic Press, New York (1990).
- [8] F.M. Richards, *Annual Reviews in Biophysics and Bioengineering* 6, 151 (1977)
- [9] M.L. Connolly, "Analytical Molecular Surface Calculation", *Journal of Applied Crystallography* 16 548 (1983)
- [10] J.L. Sussman, M.Harel, F. Frolow, C. Oefner, A. Goldman, L. Toker and I. Silman, "Atomic Structure of Acetylcholinesterase from Torpedo Californica: a prototypic acetylcholine-binding protein", *Science* 253, 872, (1991).
- [11] R.J. Zauhar and R.S. Morgan, "Computing the Electric Potential of Biomolecules: Application of a New Method of Molecular Surface Triangulation", *Journal of Computational Chemistry*, 11 (5) 603 (1990)
- [12] R.H. Wolfe, M. Needels and J.D. Joannopoulos, "The Electronic Structure of Oxygen in Silicon as Revealed by Volume Visualization of Ab Initio Calculations", *IEEE Visualization '91 Proceedings* p17 (1991)

[For color plate see page 927]

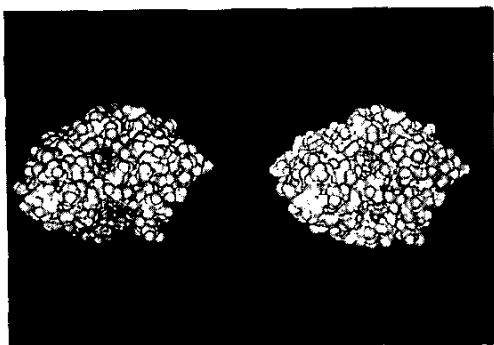


Figure 1: Molecule rendered with (left) and without (right) depth cueing.

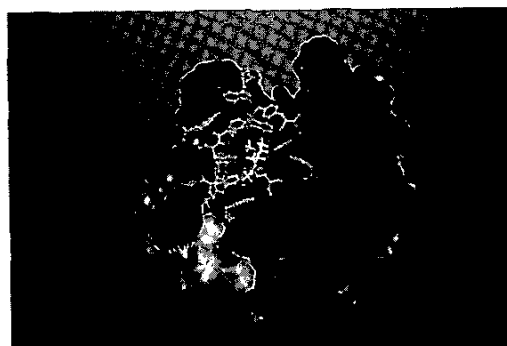


Figure 3: Cross section of acetylcholinesterase enzyme in molecular surface representation.

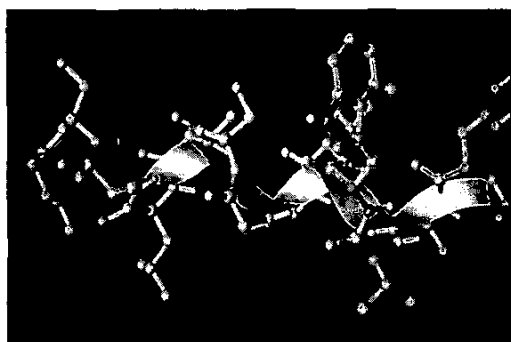


Figure 2: Typical polypeptide segment with one amino acid residue colored and a ribbon model superimposed.

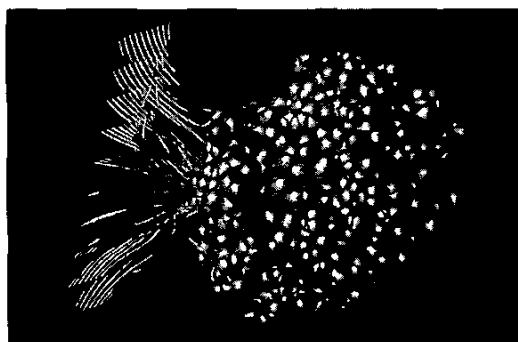


Figure 4: Streamlines of the electric field of acetylcholinesterase converging near the entrance to the active site.

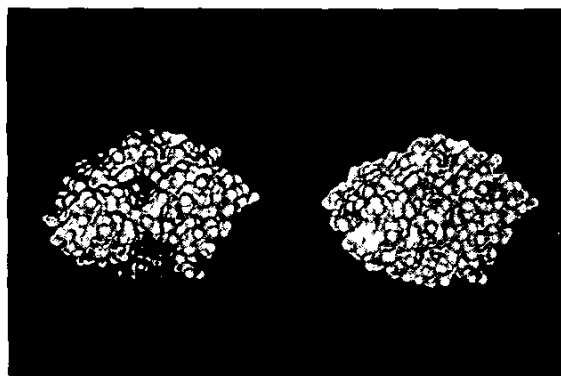


Figure 1: Molecule rendered with (left) and without (right) depth cueing.



Figure 3: Cross section of acetylcholinesterase enzyme in molecular surface representation.

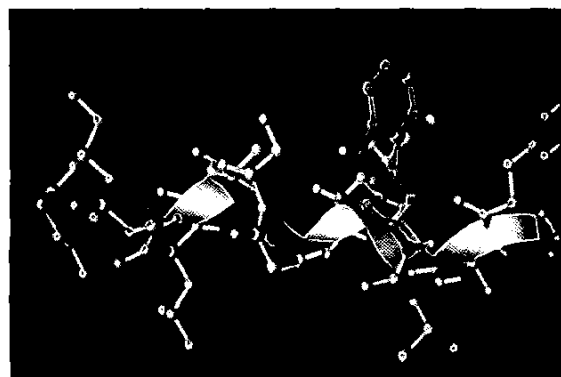


Figure 2: Typical polypeptide segment with one amino acid residue colored and a ribbon model superimposed.

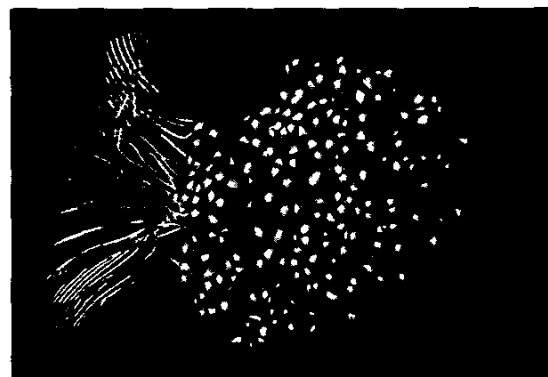


Figure 4: Streamlines of the electric field of acetylcholinesterase converging near the entrance to the active site.