Visualization of solvation structures in liquid mixtures

Dan L. Bergman,* Leif Laaksonen,† and Aatto Laaksonen*

*Division of Physical Chemistry, Arrhenius Laboratory, Stockholm University, Stockholm, Sweden
†Center for Scientific Computing, Espoo, Finland

Spatial distribution functions of atomic densities, SDFs, have been proposed as a natural starting point for analysis of local molecular structure in liquids and solutions. The local structure in these systems is often complex and this is reflected in the fact that SDFs can be difficult to visualize. Among the different methods that can be used to visualize SDFs we discuss 3D isodensity surfaces, cross-sections, and 'comic book' animations. We also discuss the possibility of a simultaneous visualization of SDFs and other 3D fields, such as the electron density. These techniques are all intended to emphasize and bring out aspects of SDFs that promote a further understanding of the local molecular structure. OpenGL-based software has been used under X-Windows to implement these techniques, and we argue that high-quality molecular graphics need not be expensive. Data from a molecular dynamics simulation of an equimolar binary mixture of water and acetonitrile have been used to illustrate the discussion. © 1998 by Elsevier Science Inc.

Keywords: 3D solvation structure, MD simulation, spatial distribution function, water–acetonitrile mixture

INTRODUCTION

The study of liquids and solutions has a long history. Seen in this perspective, computer simulation has a brief history. Nevertheless, over the last decades, through an interplay between experiment[1] and simulation,[2] a fairly detailed picture of structure and dynamics on the molecular scale has evolved. Molecular dynamics (MD) simulations of molecular systems in condensed phases are often used to generate trajectories containing atomic coordinates and other dynamic data for individual atoms and molecules. In addition to the calculation of ensemble averaged properties, one can generate animations that illustrate

how molecules move. Without having seen any real molecules in a liquid or solution, we assume that molecules moving 'in a computer' behave much like real molecules. This assumption is supported by the fact that a number of experimentally accessible quantities have been reproduced well by computer simulations.

Computer animations show the chaotic nature of the thermal motion of molecules in the liquid state; it is difficult to discover any regularities in the liquid structure from snapshots of an animation. However, if one follows the motion of a particular molecule and its surroundings over a longer period of time, then the time-averaged structure shows regularities. The potential fields dictate how the atoms are distributed relative to each other in space. Attractive interactions, for example, cause some atomic positions to occur more frequently, and thus result in higher values for the distributions at those positions.

Radial distribution functions (RDFs),

$$g_{AB}(r) = \frac{\rho_B(r|r_A = 0)}{\rho_B} \qquad (1)$$

are frequently used to analyze liquid structure in several ways. [$g_{AB}(r)$ can be interpreted as the probability of finding an atom of type B at a distance r from an atom of type A divided by the probability of finding an atom of type B anywhere in the bulk liquid. $g_{AB}(r)$ thus approaches unity as r tends to infinity.] RDFs can, for example, be used to estimate the length of hydrogen bonds. Integration of an RDF yields the coordination number, that is, the average number of neighboring atoms of a certain type up to a fixed distance. Experimental RDFs are available as Fourier-transformed, partially resolved structure factors from neutron and X-ray diffraction studies.[1]

It should be remembered that RDFs exhaust the two-body structural information for atomic liquids only. In molecular liquids there are orientational correlations between the molecules, which are not included in the distance distributions. One way to extend the structural analysis is by systematical combination of different RDFs. The usefulness of this indirect method is limited by the fact that structural information is obscured and partly lost by averaging over the angular part of the atomic pair distributions. This can be seen in the liquid mixture of water and acetonitrile that we have chosen to illustrate the discussion in this article. Acetonitrile (see Figure
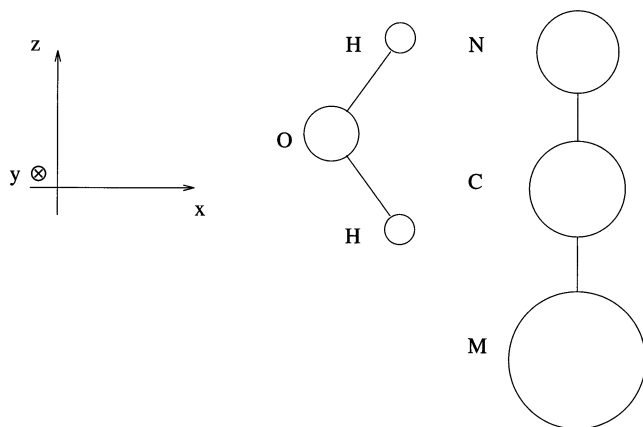
Figure 1. The geometries of the water and acetonitrile models are shown here. Left: *Water with* $d_{OH} = 1$ Å *and HOH* $= 109.47°$. Right: *Acetonitrile (N, nitrogen; C, carbon; M, methyl group) with* $d_{NC} = 1.157$ Å, $d_{CM} = 1.458$ Å, *and* $d_{NC} = 2.615$ Å. *The molecules have been oriented so that the axes of their local frames are parallel to the axes of the coordinate system in the figure.*

1) has a hydrophobic moiety around the methyl group and a hydrophilic moiety around the nitrile group. The halves of acetonitrile are thus differently hydrated in water solution, and the corresponding RDFs are the result of the superposition of different structures. Both structures are obscured when superimposed on each other.

Spatial distribution functions (SDFs) are defined by

$$g_{AB}(\mathbf{r}) = \frac{\rho_B(\mathbf{r}|\mathbf{r}_A = 0)}{\rho_B} \qquad (2)$$

Although formally similar to their radial counterparts, SDFs capture the three-dimensional (3D) structure around a chosen molecule. As for the RDFs above, A and B each denote an atom type and ρ_B denotes the bulk density of type B atoms. $\rho_B(\mathbf{r}|\mathbf{r}_A = 0)$ is the conditional density of type B atoms at the point $\mathbf{r} = (x, y, z)$ as measured in a coordinate system rigidly attached to the molecule containing atom A. The origin of this coordinate system is at the atom A. One could casually describe this as the average distribution of B atoms seen by an observer riding on an atom of type A.

Svishchev and Kusalik[3] were the first to apply SDFs to visualize local molecular structure in their study of pure liquid water. We have proposed that SDFs are a natural starting point for the analysis of local molecular structure in molecular liquids, and especially in liquid mixtures where strong orientational correlations between the molecules prevail.[4] We have applied the method to water–acetonitrile mixtures[4] and to water–methanol mixtures.[5] Since SDFs are easy to compute and because of the continuous development of computer simulations (in particular, better graphical presentation techniques and better potential models) we anticipate that SDFs will replace RDFs in many situations.

The local structure in molecular liquids is often complex and this is reflected in the SDFs. They are therefore rich in detail and information, and can be difficult to visualize. High-quality

computer graphics with rendering techniques is necessary to interpret the distributions. In this article we attempt

● To find powerful techniques to visualize SDFs, which can support the understanding of the solvation structure in liquid mixtures

● To demonstrate that inexpensive, high-quality display techniques are available and can be used to visualize the detail-rich SDFs

It is not our purpose to present a detailed analysis of the solution structure of the water–acetonitrile mixture: this has already been reported.[4]

COMPUTATIONAL DETAIL AND VISUALIZATION TOOLS

Computational chemistry and particularly molecular simulations regularly rely on molecular graphics at some stage in the validation or analysis process. It is difficult to obtain spatial understanding of a chemical process directly from a trajectory. Computer graphics can, of course, be used in several ways together with molecular simulations. Writing the atomic coordinates periodically on disk, after solving the equations of motions in MD simulations and letting a molecular graphics program read them from the file in a continuous manner, will 'open a window' into the microscopic world, mimicking the thermal motion of the molecules. This and similar techniques can be used to generate animations of scientific character. High-end applications of this category may include interactive virtual reality (VR) to offer the investigator the ability to move among the molecular images for a closer look at fast local molecular processes.

In this work we proceed quite differently. As a primary input to the graphics software we use the SDFs [cf. Eq. (2)], which have been calculated in advance. To obtain the SDFs as three-dimensional probability distributions, the space surrounding each atom is divided into a $100 \times 100 \times 100$ mesh of cubic boxes. The mesh is, at each time step, centered at the selected atom and the principal axes of the mesh are chosen to coincide with those of the molecule to which the atom belong. For each atom the number of other atoms in the boxes surrounding it were counted at each time step. A double average over time and all similar atom pairs[‡] is then used to construct the SDFs. We can describe this procedure formally by

$$g_{AB}(i, j, k) = \frac{\left\langle \frac{1}{N_A} \sum_{n_A=1}^{N_A} \sum_{n_B=1}^{N_B} I_{ijk}[R_{n_A}^A(\mathbf{r}_{n_B}^B - \mathbf{r}_{n_A}^A)] \right\rangle}{\rho_B l^3} \qquad (3)$$

where i, j, and k index the various boxes in the mesh, N_A denotes the number of atoms of type A, and n_A is an index running from 1 to N_A. $\mathbf{r}_{n_A}^A$ is the position vector of the atom numbered n_A of type A in the laboratory frame and $R_{n_A}^A$ is the rotational transformation from the laboratory frame to the local frame of atom n_A. I_{jkl} is the indicator function, taking the value 1 for $\mathbf{r} \in [l(i - \frac{1}{2}), l(i + \frac{1}{2})] \times [l(j - \frac{1}{2}), l(j + \frac{1}{2})] \times [l(k -$

[‡]Note that only ordered atom pairs are used and pairs involving two atoms in the same molecule are not counted.

½), $l(k + ½)$] and the value 0 elsewhere. l (0.2 Å) is the edge of the boxes. The angular brackets denote time average.

The processed trajectory has been taken from a 200 ps long MD simulation of an equimolar water–acetonitrile mixture,[4] carried out in the NVT ensemble at 300 K using rigid molecular models. A modified version of the *McMoldyn* package[6] was used to carry out the simulation. Ewald summation was used to treat the long-range Coulombic interactions[7] and a Nosé–Hoover thermostat was applied to maintain the temperature.[8,9] Additional details, concerning molecular and potential models can be found in Ref. 4. The molecular models used are shown in Figure 1.

In the past, computer graphics were created on powerful—but expensive—graphics workstations, owing to the lack of adequate 3D graphics libraries for a wide range of hardware platforms. After the introduction of the OpenGL application programming interface (API) by Silicon Graphics,[10] a rapid development took place in the generation of interactive 2D and 3D computer graphics. OpenGL is designed to be independent of the operating system and window system used. OpenGL is now available for PC hardware platforms and a variety of Unix workstations.

To utilize fully the speed of the OpenGL API, one still needs a hardware display accelerator. However, one can reach a fair graphics speed even with an inexpensive PC-based display card. Running OpenGL on a Unix platform limits the use to the local console display, or to the console of another OpenGL-enabled workstation. To overcome this problem, there are OpenGL-like libraries available, having all the OpenGL call functionality, but utilizing X-Windows as the transfer layer. A frequently used OpenGL-like library is the Mesa library.[11] It provides a good-quality 3D graphics library with an API similar to that of OpenGL. Because the Mesa library is a plain X-Windows product, it runs on almost all Unix workstations. There are also Windows/PC and Macintosh versions of Mesa available. Currently, the main disadvantage of the Unix Mesa is the limited drawing speed of X-Windows.

The graphics software employed in the present work is the powerful *gOpenMol* package.[12] Originally built around GL and the Silicon Graphics hardware environment, it has reemerged as an OpenGL application software with a practical, easy-to-use graphical interface, with many features to satisfy users from quantum chemistry to biocomputing. Some of the technical details are described in the appendix.

DISCUSSION

Since an SDF is a three-dimensional scalar field, matters would be simplified if one could use four dimensions to visualize it. Varying intensity could then be directly observed as the coordinate along the fourth axis. In this section we discuss the merits of different methods of visualizing SDFs. These methods are intended to reduce the dimensionality of the problem while still capturing the features of the distribution, relevant to the understanding of solvation structure. Two common ways to reduce dimensionality are the following:

● *Isosurfaces:* The equation $g_{AB}(\mathbf{r}) = c$ defines an isodensity surface in three dimensions, which consists of the points around the atom A where it is equally probable to find an atom of type B.

● *Cross-sections:* By fixing, for example, a Cartesian or a spherical coordinate, one may study how the function $g_{AB}(\mathbf{r})$ varies on a 2D surface. The third free dimension can be used to indicate the value of the function.

We have chosen the isosurface and the cross-section display styles as the main techniques with which to visualize the SDFs. Both methods obviously have their advantages and disadvantages, but by combining them and using modern animation techniques one is able to decipher the spatial distribution functions in an intuitive and powerful way.

Isosurfaces

In this work we have computed the value of each SDF on a $100 \times 100 \times 100$ mesh of cubic boxes (see Computational Detail, above). Essential features of the distributions, such as regions of high probability, can be visualized by drawing isosurfaces defined by fixing the value of $g_{AB}(\mathbf{r})$. This is an efficient method of finding the regions where the probability is close to the maximum or the minimum of the distribution. It is, however, difficult to draw several isosurfaces in the same figure, since they may cover each other. In a liquid, the structural correlations are the strongest at short range and regions of high probability are usually found close to the molecule that we have selected as our reference system. Low-probability isosurfaces can therefore be expected to cover high-probability isosurfaces, as long as the probability is above the bulk value. In theory, one could resolve this problem by making the surfaces corresponding to lower probability more transparent. An 'onion-style' image showing several isosurfaces at the same time could be obtained by this method. However, as the number of layers is increased, the image becomes difficult to interpret. Even after three layers this problem is readily apparent.

To understand the relative locations of regions of high probability corresponding to different atom types surrounding, for example, a reference water molecule, one can draw one isosurface for each atom type in the same image. This has been done in Color Plate 1, where we have placed the water molecule in the center of the image. The isosurfaces for oxygen around oxygen (red), nitrogen around oxygen (blue), and methyl around oxygen (green) are all defined by the relation $g_{OX}(\mathbf{r}) = 3.5$, where X denotes O, N, and M, respectively.

We can see that the regions of high probability partly overlap each other in the case of the OO and ON distributions. Obviously, two atoms cannot be at the same location at the same instant, but since the SDFs are time averages (200 ps in this work) the regions can overlap. By showing the isodensity surfaces corresponding to different atom types at the same time, one can illustrate a number of features. In Color Plate 1 we can, for example, see the tetrahedral nature of the hydrogen bond donor/acceptor capacity of water: the water oxygen–oxygen distribution has regions of high probability both above and below the reference water molecule. Nitrogen atoms, on the other hand, can form hydrogen bonds only when water donates a proton, hence there is no nitrogen surface below the reference water molecule. The green regions, where the probability of finding the hydrophobic methyl groups is high, are located on the sides of the reference water molecule. (These positions are energetically unfavorable for the oxygen and

nitrogen atoms, and it is likely that entropy effects push the methyl groups there.)

One should not think of the structures, shown by the SDFs, as frozen; rather, imagine the molecules in rapid motion. The structure we see is the result of perpetual motion. An example of this is the oxygen region on the acceptor side of the water molecule, where the lone pairs are thought to be. This large region is a result of the water molecule in the middle executing a rocking motion while its hydrogens are engaged in bonds with two other water molecules. (This becomes clear by watching computer animations of the trajectory.)

'Comic book' animations

The obvious drawback in the isosurface presentation is the lack of the fourth dimension, that is, the intensity scale. Both RDFs and SDFs are normalized to unity at bulk density. In the case of RDFs, bulk density is normally reached at an interatomic separation of 5–10 Å, at which the molecular close order with respect to the center molecule disappears owing to the thermal motion. The highest RDF peak is usually below 10. In liquid water,[§] for example, the highest peak is close to 3. In the case of SDFs the intensities can, locally, be considerably higher. For example, we could redraw Color Plate 1 by choosing intensities as high as 20 and we would still see solvation structure: the hydrogen bonds, shown in red and blue, would still be visible. In RDFs, local regions with high intensities are obscured by angular averaging with regions of low intensity. In the most unfortunate cases the high and low intensities cancel out, leaving no traces of structure in the RDFs.

To include the variation of intensities, a number of isosurfaces corresponding to different probabilities [i.e., different constants, c, in the equation $g_{AB}(\mathbf{r}) = c$] can be presented in a 'cartoon'. In Color Plate 2, a cartoon of the oxygen–oxygen surfaces is shown. In the upper left corner, the isosurface corresponding to $c = 4.0$ is shown. This surface consists of two parts corresponding to the hydrogen bond sites where the reference water molecule donates hydrogens. In the upper right corner the $c = 3.0$ surface is shown. At this lower intensity, the remaining hydrogen bonds on the acceptor side of the water molecule become visible. To the left in the second row the intensity is lowered to $c = 2.5$, which is enough for the second solvation shell to appear. In the same row to the right, the $c = 2.0$ surface is shown, and one can see how the regions corresponding to the second solvation shell have grown. When the intensity of the surface decreases toward the bulk value, the different regions grow into each other (see the bottom row of Color Plate 2, $c = 1.8$ and 1.5). The regions in space where the intensity drops below 1 form holes. This type of region can be found between the first and second solvation shells.

Cross-sections

The cartoon of isodensity surfaces in Color Plate 2 shows how different regions correspond to different probabilities. The sequence of surfaces is shown from the perspective of an observer at a distance. Since both the maximum and the minimum of the distribution usually are located close to the molecule, the low-probability surfaces will have interior parts that are invisible from the outside. To study the interior regions of low

probability we may, for example, use a cross-section of the isodensity surface. In Color Plate 3 a cross-section through two isodensity surfaces of the oxygen–oxygen distribution is shown. The two isodensity surfaces correspond to $c = 1.8$ (green) and $c = 1.5$ (red). The red interior surfaces surround regions of low probability.

An alternative approach is based on the full distribution function. By taking a cross-section through the full distribution, instead of through one or two isodensity surfaces, we obtain the distribution on a plane. We can represent each density by a different color and observe the details of the distribution; see Color Plate 4. The disadvantage of this approach is that we lose all information regarding the distribution behind and in front of the selected cross-section. It is therefore of key importance where the cross-section plane is placed. In Color Plate 4 it coincides with the plane of the water molecule. Another natural choice would be the plane perpendicular to the molecule. In Color Plate 5, two cross-sections are drawn at once: one plane is parallel and the other is perpendicular to the water molecule.

To obtain an overview of the full three-dimensional probability distribution, a sequence of parallel cross-sections can be used; see Color Plate 6. A cartoon of isosurfaces is informative regarding the shape of different regions, but it does not expose the probability mass distribution as clearly as a sequence of parallel cross-sections. Animations of this type are useful, in order to see the gradual changes. A natural extension of this approach is to increase the number of frames and to show them in sequence. This gives us a full animation that we can play back and forth at various speeds in the hope of discovering new information. To explore this possibility fully we have made a series of animations available.[13]

Introducing electron densities

Several quantum chemistry packages produce electron density maps or electrostatic potentials. These can be visualized in a manner analogous to the way atom densities are shown. The current *gOpenMol* software can be used to visualize quantum chemically calculated molecular and electronic properties. As a last part of this discussion, we briefly present one method to simultaneously visualize results obtained from quantum mechanical and molecular mechanical models. An example of this is shown in Color Plate 7, in which the reference water molecule is surrounded by an electron density map (including both core and valence electrons) and by the isosurface $g_{OO}(\mathbf{r}) = 3.5$. This combined quantum mechanical and molecular mechanical figure has been generated mainly to show that it can be done conveniently using current graphics software. In applications where hybrid types of MD methods are used, this type of graphic may be useful in the analysis of results. The *Gaussian94* package[14] has been used to calculate the electron density map for the reference water molecule, using basis sets of DZV quality.

CONCLUSION

We have presented several methods for visualizing spatial distribution functions. The different methods are complementary: isodensity surfaces are particularly useful in identifying regions of high probability and in obtaining an understanding of the relative arrangement of different atom types around the reference molecule. Cross-sections offer a detailed overview of

---

[§]At room temperature and atmospheric pressure.

the intensity distribution. Cartoon and animation techniques allow one to explore the full distribution in greater detail. These methods can effectively be combined to visualize various aspects of SDFs, relevant in the context of understanding liquid solvation structures from molecular dynamics simulations.

There are already a large number of advanced graphical programs and visualization tools available for console and X-Windows equipment, through the release of the *de facto* standard, 3D library OpenGL and the free look-alike library Mesa. One objective of this article has been to show that graphical analysis and animations no longer require Unix workstations, but can easily be done on the inexpensive low-end hardware we already have on our desks.

ACKNOWLEDGMENTS

APPENDIX: *gOpenMol*—AN OpenGL GRAPHICS SOFTWARE PACKAGE FOR COMPUTATIONAL CHEMISTRY

Modern software development should be based on an open concept in order to easily extend the functionality of the program with new plug-in modules. The old FORTRAN77 standard was not particularly helpful in that respect. The new FORTRAN90 language already provides a programmer with many helpful tools for making reusable and object-oriented code. The driving idea during the development of *gOpenMol*[12] has been to structure the code such that the extension of the core code, or *kernel*, with new functionality would be possible without a total redesign of the graphical interface. The command parser should be controlled by a powerful script language enabling multithreading.

The *gOpenMol* code is written using the ANSI C language. Dynamic memory allocation is used throughout the program, and *get* and *put* functions are used for allocating and returning pointers to arrays and structures in the program. The use of dynamic shared objects (DSOs) enables the loading of extensions to the core code without recompiling and relinking of the core modules. The user simply uses appropriate *get* and *put* functions to access the core data and a *load* command to include the new binary module to the program.

The powerful Tcl (tool command language),[15,16] which is an embeddable scripting language, and Tk, which is a graphical user interface toolkit based on Tcl, have been used as the basis for building a command parser and an extendible graphical user interface. The whole graphical user interface (GUI) is, at program startup, read in as a Tcl script. The GUI then talks with *gOpenMol* through the program command parser. This enables several interesting features: the user can, during the session, totally redesign and modify the GUI, to better reflect the user's own needs. It is also possible to read both the whole GUI and the input data through the World Wide Web (WWW). In this case only the core module is run locally. The availability of the Tcl/Tk toolkit and OpenGL on a variety of hardware platforms makes it easy to support *gOpenMol* on a broad range of hardware platforms. The same Tcl scripts, defining the graph-

ical interface and the underlying tools, run on all supported hardware platforms with no changes.

The help system is based on the HTML language. A Web browser, written in Tcl, is included with the GUI and can be used to access the help pages. The help pages can also be set up on a remote Web server and it can be accessed by any Web browser. Using a Tcl-based Web browser to access the help pages extends the help system in various ways. It is easily possible to add the interaction between the HTML pages and *gOpenMol* using Tcl. Apart from this, *gOpenMol* can also be used as a helper application for a Web browser. This can be done by first defining the file type *gom* (*gOpenMol* file type) as chemical/x-gom on your Web server, if one wants to publish files. The Web browser must then be configured to start *gOpenMol* whenever it receives a *gom* coordinate file supported by *gOpenMol*. *gOpenMol* can be configured similarly to accept other supported coordinate file types.

gOpenMol has been released as an alpha version and runs using the OpenGL or Mesa libraries on IBM AIX and SGI IRIX platforms and using the Mesa library in the PC-Linux environment. *gOpenMol* will be used as the graphical user interface in the OpenMol[17] set of quantum chemistry programs. *gOpenMol* is developed from the SCARECROW[18] molecular graphics package for the display and analysis of molecular dynamics trajectories, mainly using *GaussianXX*[14] software.

REFERENCES

1 Egelstaff, P.A. *An Introduction to the Liquid State*. Oxford Science Publications, Oxford, 1992

2 Allen, M.P., and Tildesley, D.J. *Computer Simulation of Liquids*. Oxford University Press, Oxford, 1989

3 Svishchev, I.M., and Kusalik, P.G. Structure in liquid water: A study of spatial distribution functions. *J. Chem. Phys.* 1993, **99**(4), 3049–3058

4 Bergman, D., and Laaksonen, A. Angularly resolved density distributions—a starting point for analysis of liquid structure. *Molecular Simulation* (in press)

5 Laaksonen, A., Kusalik, P.G., and Svishchev, I.M. Three-dimensional structure in water–methanol mixtures. *J. Phys. Chem. A* 1997, **101**(33), 5910–5918

6 Laaksonen, A. Computer simulation package for liquids and solids with polar interactions. I. McMOLDYN/H_2O: Aqueous systems. *Comput. Phys. Commun.* 1986, **42,** 271–300

7 Ewald, P.P. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Ann. Phys.* 1921, **64,** 253–287

8 Nosé, S. A molecular dynamics method for simulations in the canonical ensemble. *Mol. Phys.* 1984, **52**(2), 255–268

9 Hoover, W.G. *Computational Statistical Mechanics*. Elsevier, New York, 1991

10 Neider, J., Davis, T., and Woo, M. *OpenGL Programming guide. The Official Guide to Learning OpenGL*, release 1. Addison-Wesley, Reading, Massachusetts, 1993

11 Paul, B. The Mesa 3-D graphics library. http://www.ssec. wisc.edu/~brianp/Mesa.html. 1997

12 Laaksonen, L. *gOpenMol*, version 1.0A3. Center of Scientific Computing, Espoo, Finland. http://laaksonen. csc.fi/gopenmol/. 1997

13 http://laaksonen.csc.fi/gopenmol/gallery/. 1997

14 Frisch, M.J., Trucks, G.W., Schlegel, H.B., Gill, P.M.W., Johnson, B.G., Robb, M.A., Cheeseman, J.R.,

Keith, T.A., Petersson, G.A., Montgomery, J.A., Ragha-vachari, K., Al-Laham, M.A., Zakrzewski, V.G., Ortiz, J.V., Foresman, J.B., Cioslowski, J., Stefanov, B.B., Nanayakkara, A., Challacombe, M., Peng, C.Y., Ayala, P.Y., Chen, W., Wong, M.W., Andres, J.L., Replogle, E.S., Gomperts, R., Martin, R.L., Fox, D.J., Binkley, J.S., Defrees, D.J., Baker, J., Stewart, J.P., Head-Gordon, M., Gonzalez, C., and Pople, J.A. *Gaussian 94*, Revision A.1., Gaussian, Pittsburgh, Pennsylvania 1995

15 Ousterhout, J. *Tcl and the Tk Toolkit*. Addison-Wesley, Reading, Massachusetts, 1994

16 Welch, B. *Practical Programming in Tcl and Tk*. Prentice-Hall, Englewood Cliffs, New Jersey, 1997

17 The OpenMol community. http://www.mpa-garching. mpg.de/~opmolsrv/OpenMol.shtml. 1997

18 Laaksonen, L. A graphics program for the analysis and display of molecular dynamics trajectories. *J. Mol. Graphics* 1992, **10,** 33–34