A molecular editor program

Richard M. Hilmer

Central Research and Development Department, E.I. DuPont de Nemours & Co., Wilmington, DE, USA

A program called MOLEDITOR has been developed for manipulating images of molecules and various geometric objects in real time and in three dimensions using Evans and Sutherland PS300 graphics displays and Digital VAX computers. In addition to the normal viewing operations of rotation, translation, clipping and scaling, the program allows molecules and objects to be placed in local frames for motion relative to other molecules or objects and allows several modes of motion about individual bonds. MOLEDITOR also allows the display and manipulation of molecules and objects within crystal lattices and of the lattices themselves via manipulation of the unit cell parameters. The program also supports display and manipulation of animated frames from dynamics simulations. It is also possible to dynamically monitor coordinates of and distances between items. An easy-to-use scripting capability allows the user to generate sequences of motion that can be recorded directly from the display.

Keywords: docking, bond motion, crystal lattice, symmetry, dynamics, scripting

INTRODUCTION

The MOLEDITOR program has been developed within DuPont over a number of years, with input from people in a wide range of disciplines. Originally, the main users were people interested in drug design and in small molecule modeling, but for the last few years, the main users have been polymer and protein crystallographers. The aim of the program has never been to replace all the current software that is available, but to provide functionality that none of the currently available software provides, principally display of geometric objects, crystal lattice packing and scripting. Some of the features, such as the ability to display dynamics trajectories, have recently become available in some programs, but were not available at the time they were needed. The program continues to evolve and grow as the needs of the users dictate.

USER INTERFACE

MOLEDITOR supports all the various input devices available for the PS300 systems, including the keyboard, data tablet, dials, joysticks and function keys. The keyboard must be used for certain input, such as the name of a file, but can also be used to input an entire command. The data tablet is used for picking, either options from the menu or items from the screen. Whenever the user uses the data tablet to point at an item such as a menu option or an atom, bond, or geometric object on the screen, that item lights up to indicate what the user is about to select. When the correct item is highlighted, the user presses the button or presses the pen to select that item. This avoids the problem of cancelling items that were incorrectly picked. The tablet input and keyboard input are interlocked so the user can either type or pick as he or she desires. As the user types commands, the menu appearing on the screen changes to reflect the current state of the input command. As the user picks from the screen, the command line is generated as though the user were typing it in. For some of the simpler commands, the function keys are used to duplicate the function of picking menu options, and they also generate the command line as though the user had typed it in.

The dials and joysticks are used to control the motions of molecules and objects on the screen, and their functions can be dynamically reassigned by command so that it is possible to connect any dial or joystick to any controllable function or set of functions at any given time. The LED display on the dial or joystick always shows what function is currently connected. All motions are controlled in the most natural way possible, so that the user always knows how the display will be changed when he turns a dial.

INPUT AND OUTPUT

The only common file format for molecules that MOLEDITOR currently supports is the Brookhaven Protein Data Bank (PDB) file. As they are read in, all molecules are placed into the global frame of motion. On output, the coordinates of each atom in a molecule are normalized to the global frame before they are put into the file. Thus, if a molecule is in a local frame of motion, is an element within a crystal lattice, has flexible bonds, or all of the above, the coordinates in the file are those relative to the global frame.

Files depicting dynamic motion are expected to be in a specially compacted binary format. Output such as that generated by the AMBER program for its trajectory files must be reprocessed to compact it before reading it in with MOLEDITOR.

If the user wishes to save the current status of the system, he can generate a structure file that is a snapshot of all the relevant data. He can then explore the system as he sees fit

Address reprint requests to Dr. Hilmer at the Central Research and Development Department, E.I. DuPont de Nemours & Co., Wilmington, DE 19880-228, USA.
Received 24 April 1989; accepted 6 June 1989

and can always return to the previously saved state of the system by reading in the saved structure file. Structure files also save a lot of time by allowing the user to save the current status of his system from one graphics session to another.

MOLEDITOR has built into it six geometric objects; the sphere, cylinder, box, plane, point and shoe (an asymmetric object useful in crystallographic studies). In addition, the user can define new types of objects from a simple file format. The user-defined objects then become menu-selectable, the same as the built-in objects. An object can be scaled in one or more dimensions, depending on its type. Spheres, for instance, scale in all three dimensions simultaneously (radius); cylinders scale by radius and length, and so on. It is easy to represent large portions of molecules by geometric objects, for example, and alpha-helix in a protein by a cylinder, simply by scaling the object and placing it over the molecule.

NAMED SETS

A user can refer to objects, molecules or portions of molecules as sets, to be treated in some way. Portions of molecules that can be named are atoms, residues, segments (a group of atoms hanging from a flexible bond) and symmetry elements (an asymmetric unit in a crystal lattice). One command allows the user to assign a name to a set and to refer to that named set in later commands. The named set will appear on the appropriate menu during set selection. Sets can be subjected to the Boolean operations of union, difference and intersection. A valid command might read "CREATE SET HPHILACID ALL RESIDUES ~ASP ~GLU ~TYR," which could be followed later by "COLOR RED SET HPHILACID EXCEPT RESIDUES 55–70 MOLECULE 2."

APPEARANCE ATTRIBUTES

Objects have only two appearance attributes: visibility and color. Visibility determines whether an item appears on the screen, and color consists of four numbers denoting hue, saturation, intensity and transparency. Color attributes are functions that can be dynamically controlled from the dials. On the PS300, transparency does not have any effect, but external rendering programs controlled by MOLEDITOR do use the value. In addition, these programs use other parameters describing up to four light sources, their location, intensity and color, all of which are dynamically controllable within MOLEDITOR. MOLEDITOR does not yet support raster rendering on PS300's with raster capability.

Molecules have three features which can be displayed—bonds, atom labels and covalent or van der Waals surfaces, which can be generated within MOLEDITOR for any set of atoms. All three of these features have visibility and color attributes associated with them. In addition, atom labels (which appear at the coordinates of the atoms) have a label type attribute, and can be either chemical symbols, chemical symbol plus residue identifier, residue name or residue number.

TRANSFORMATION EQUATION

The overall transformation equation for a given atom or object is as follows:

$$[V_u][B][C][S][L][G][P] = [V_t]$$

where **[Vu]** is a four-dimensional vector **[Xu, Yu, Zu, 1]** comprising the untransformed coordinates of the atom and a homogeneous scaling coordinate of 1, and **[Vt]** is the equivalent vector, untransformed to screen coordinates. **[B]** is a set of matrices that are present only if the atom being transformed is dependent from a flexible bond. **[C]** is a set of matrices that are present only if the molecule is part of a crystal lattice. Matrix **[S]** is always present and is used to accumulate the effects of local frame motions. **[L]** is a matrix that is present only if the molecule or crystal lattice is within a local frame of motion. **[G]** is the matrix controlling global motions and is always present. **[P]** is a set of two matrices that control scaling and viewing transformations, and it too is always present.

As previously stated, on input, the coordinates read in from the file are stored in **[Vu]**. The matrices **[B]**, **[C]** and **[L]** are not present, and matrix **[S]** is initialized to identity. On output, the coordinates are normalized by applying the transformations **[B]**, **[C]**, **[S]** and **[L]** before writing them into the output file.

VIEWING

Viewing consists of scaling, clipping, perspective and stereo. Perspective projection can be turned off, if desired, to show the orthogonal projection. Several stereo modes are supported. The front and back clipping planes are controlled together as a slab that has variable position and thickness (controllable by dials). One important feature of the slab is that it is tied to the scaling function so that the slab scales at the same rate as the rest of the display. Hence, the slab thickness is measured in angstroms, as is its position. The viewing matrices denoted by **[P]** are defined as follows:

$$[P] = \begin{bmatrix} S & 0 & 0 & 0 \\ 0 & S & 0 & 0 \\ 0 & 0 & -S & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \frac{FD}{X} & 0 & 0 & 0 \\ 0 & \frac{FD}{Y} & 0 & 0 \\ \frac{FE}{X} & 0 & \frac{D-FS(P+W/2)}{-SW} & F \\ 0 & 0 & \frac{(P-W/2)[D-FS(P+W/2)]}{-W} & D \end{bmatrix}$$

F = conversion factor from angstroms to inches
D = distance from eye to screen in inches
E = offset of eye from centerline in inches
X = half-width of viewport in inches
S = scale factor
P = slab position in angstroms
W = slab width in angstroms
Y = half-height of viewpoint in inches

for perspective projection, and

$$[P] = \begin{bmatrix} S & 0 & 0 & 0 \\ 0 & S & 0 & 0 \\ 0 & 0 & -S & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \frac{F}{X} & 0 & 0 & 0 \\ 0 & \frac{F}{Y} & 0 & 0 \\ \frac{FE}{XD} & 0 & \frac{1}{-SW} & 0 \\ 0 & 0 & \frac{(P-W/2)}{-W} & 1 \end{bmatrix}$$

for orthographic projection. In each case, the first matrix is a scaling matrix, and the second is a viewing matrix. In MOLEDITOR, both matrices are computed in a single user-written function.

GLOBAL MOTIONS

Global motions consist of rotation and translation to control the location and angle of view of all items being displayed. The user can display a set of global axes for a better idea of what transformations have been done previously. As previously stated, all input and output coordinates are relative to these axes; that is, global motions do not affect output coordinates. All global rotations and translations are screen-oriented. This means that each time a dial is turned that affects global motion, a matrix is generated that is post-multiplied onto matrix **[G]**, the global transformation matrix.

LOCAL MOTIONS

The user can define up to four local frames of motion, used for manipulating molecules, objects or crystal lattices in global space. Whenever a local frame is defined, all items previously in that local frame have their coordinates updated by the local motion matrix, and those items are placed back into the global frame. This is accomplished by postmultiplying the matrix **[S]** for these items by the last value of the matrix **[L]**, and eliminating the matrix **[L]** for those items. The newly specified set of items are then placed into the local frame of motion, and the local motion matrix **[L]** is initialized to identity. By this mechanism, the user can move items around in space (thereby changing their coordinates in the output files).

Local motions are also screen-oriented, despite the fact that a local frame of motion lies within the global frame of motion. If the user does a rotation around Y in the local frame, the object will rotate about the vertical axis of the screen, regardless of how the global frame has previously been rotated. The mechanism used is as follows. When a dial is turned that affects a local frame motion, an update matrix is generated. This update matrix is premultiplied by the current global matrix **[G]**, and postmultiplied by the inverse of **[G]**, and the result is then postmultiplied onto **[L]**.

CRYSTAL LATTICE MOTIONS

MOLEDITOR will support two independent crystal lattice operations simultaneously, thus allowing the user to study phenomena involving interpenetrating or overlapping crystal lattices. When the user defines a crystal lattice, a set of matrices denoted as **[C]** is inserted into the transformation equation for the items he has chosen. **[C]** is defined as follows:

$$[C] = [Cs_1][Cr][Ct][U][Cs_2][U^{-1}]$$

Matrix $[Cs_1]$ is a static matrix that serves to position an asymmetric unit so that the rotations and translations to be applied to it will happen in the correct direction. When the crystal lattice is defined, the matrix **[S]** is copied into $[Cs_1]$ and then **[S]** is initialized to identity. Rotations and translations of asymmetric units are defined by dynamic matrices **[Cr]** for rotation and **[Ct]** for translation. These are the same sorts of translation and rotation matrices as used for global and local motions, but they are applied separately in this case, because the **[Cr]** matrix that must be solved for Eulerian angles describing the rotations each time **[Cr]** is modified and because updates to the **[Ct]** matrix are dependent on the current value of **[U]**; whereas updates to **[Cr]** may or may not be, depending on the mode of motion desired. Matrix **[U]**, which is dependent only on the unit cell parameters, converts orthogonal coordinates to fractional coordinates; its inverse, $[U^{-1}]$, converts fractional coordinates back to orthogonal. These two matrices change dynamically as the unit cell parameters are changed from the dials. $[Cs_2]$ is a set of static matrices which position the asymmetric units within the crystal lattice according to the symmetry constraints of the space group. These are expressed in fractional coordinates, and are typically read from a data file containing the matrices for all the space groups. It is important to note that all the matrices comprising **[C]** are single matrices except $[Cs_2]$, and that the set of $[Cs_2]$ matrices are applied to a single display structure containing all the other matrices and vectors to the left of them in the equation. The effect of this is that multiple copies of the same object appear on the screen, but PS300 memory is saved by having only one copy of the asymmetric unit being displayed. The matrix **[U]** is defined as follows:

$$[U] = \begin{bmatrix} a & 0 & 0 & 0 \\ b\cos\gamma & b\sin\gamma & 0 & 0 \\ c\cos\beta & \frac{c(\cos\alpha - \cos\beta\cos\gamma)}{\sin\gamma} & \frac{c\sqrt{q}}{\sin\gamma} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$q = 1 + 2\cos\alpha\cos\beta\cos\gamma - \cos^2\alpha - \cos^2\beta - \cos^2\gamma$
a, b, c = lengths of unit cell sides
α, β, γ = unit cell angles

This matrix and its inverse and several others are computed in a user-written function in MOLEDITOR, as is the matrix **[Cr]** and the Eulerian angles representing it.

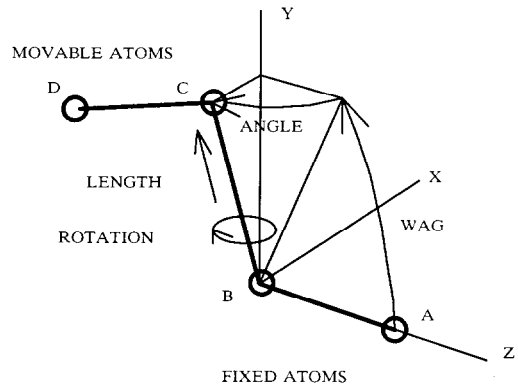

Figure 1. Bond motions

BOND MOTIONS

Up to eight bonds can be defined as flexible, and they can be nested. Matrix **[B]** contains a pair of matrices, one dynamic and one static, for each flexible bond that affects an atom. The matrices might be represented as follows:

[B] = [BDi][Bsi][BDj][Bsj] · · · · [BDn][Bsn]

MOLEDITOR allows significantly more bond motion than just rotation. Specifically, the user can control bond length, rotation, bond angle, and an out-of-plane motion called wag. These motions are defined as shown in Figure 1. Atoms A and B are fixed atoms, and C and D are movable. The bond between B and C is flexible. When a bond is defined as flexible, MOLEDITOR computes a matrix that will align the bond between A and B along the positive Z-axis with B at the origin, and C lying in the X–Z plane. This matrix is the equivalent of **[BDi]** and its inverse is placed into **[Bsi]**. The matrix **[BDi]** is factored into four parameters and is actually generated as the product of four transformation matrices in the following order. First, a Z-translation is applied, corresponding to the bond length. Second, a Z-rotation is applied to position the D atom to the correct dihedral angle. Third, an X-rotation is applied to position the bond outside the X–Z plane. This value is initially set to zero, meaning the bond between B and C will still lie initially in the X–Z plane. Fourth, a Y-rotation is applied to set the bond at the correct angle. The user is now free to modify the four parameters in any way he chooses, which gives him the ability to position the bond anywhere in space and to rotate about the bond and control its length. Note that the bond angle referred to by the Y-rotation is not the true bond angle between atoms A, B and C, but is instead the projection of that angle onto the X–Z plane.

DYNAMICS SIMULATION ANIMATION

In the first phase of displaying dynamic trajectories, the user must tell MOLEDITOR which file he wants read and which frames in the file. MOLEDITOR then reads the compressed file, expanding the data as it goes, and saves the desired frames in a random access file named MOVIE.TMP. If MOVIE.TMP already exists, the new data is appended to it. In this way, the user can build up animated sequences containing exactly the frames he wants. When all the data has been read in, the user can begin the second phase, displaying the sequence, via the command "DISPLAY MOVIE." MOLEDITOR will read the MOVIE.TMP file and send each frame to the PS300. Using the dials, the user can control the number of the first frame displayed, the number of the last frame, and how fast the frames are scanned. If the frame scan clock is stopped completely, the user can dial in individual frames.

DYNAMIC DISTANCE AND COORDINATE MEASUREMENT

MOLEDITOR can dynamically monitor the coordinates of atoms or points, as well as distances between atoms, points, and planes. Distance to a plane is always along a vector perpendicular to the plane. Distance between two planes is not allowed, because two planes are seldom exactly parallel, but any other combination of distances is allowed. It is possible, for instance, to fit a plane through a set of atoms and then monitor the distance between any atom and that plane, regardless of the motions applied to either the atom or the plane. Distances are displayed as a white line between the two items being monitored, with a number at the midpoint of the line that specifies the length of the line in angstroms. Coordinates are displayed as three numbers slightly offset from the item being monitored, the three numbers being the x, y and z coordinates of that item relative to the global frame of motion.

SCRIPTING

When the user types a command to MOLEDITOR, he has the option of saving the command in a script file when it is executed. Matrices, vectors and scalar data read back from the PS300 are also saved in the script file. The advantage of this is that, at some later time, the user can read the script file as though he were typing in the commands, and MOLEDITOR will execute them just as it did the first time. In addition, a SCRIPT command exists that allows the user to simulate turning a dial by reading commands from a script file. SCRIPT commands are executed in one of two modes: immediate or timed. If, for instance, the user types "SCRIPT GLOBAL YROT 90 IMMEDIATE," MOLEDITOR will rotate the global frame 90 degrees to the left in one step. If, however, he types "SCRIPT GLOBAL YROT 90 TIMED," MOLEDITOR remembers the command only until the user types another command to start the script clock, such as "SCRIPT CLOCK 10." At that point, MOLEDITOR will rotate the global frame 90 degrees to the left over a 10-second period. It is possible to script more than one function simultaneously, but the motions slow down somewhat when more than two are running. Any function that can be controlled by a dial can also be scripted.

SUMMARY

The program MOLEDITOR has been used successfully to study inorganic systems and organic systems that are either biomolecular or polymeric in nature. The Color Plates and descriptions in this article will give the reader an idea of some of the types of applications for which the program has been used.

APPLICATION EXAMPLES

MOLEDITOR has been used in a variety of applications, some of which are described below.

Color Plate 1 shows a perspective view of polymer chain packing in Kapton®. The molecular array was generated from a single chain by the application of crystal symmetry operations spanning several cells. The array was then optimally packed using the dynamically adjustable crystal cell option of MOLEDITOR, prior to input of the structure into molecular dynamics simulations. For analysis purposes, the simulated dynamics movie can be displayed with periodic replication (Space Group P1), corresponding to the periodic boundary conditions of the dynamics simulation.

Color Plate 2 shows a stereoscopic image of the molecular packing of lysozyme molecules projected on the crystal habit. The image was made by replicating several lysozyme cells using the crystal symmetry option, defining the crystal habit as an object by connection of cell apices, and then selectively deleting molecules to reveal those lying along developed crystal face directions.

Color Plate 3 shows a superposition of two variants of the protein calmodulin, oriented to illustrate comparative features of their electrostatic fields. In this case, each molecular coordinate set has an object associated with it (the electrostatic field computed with an external program). Association of each molecule with an independent local frame allows them to be superimposed or docked as desired.

Color Plate 4 shows a time exposure of a few picoseconds of a molecular dynamics movie simulating the interaction of the enzyme phospholipase A2 with a phospholipid micelle. The movie was generated from two independent molecular dynamics simulations, which were then associated with two independent local reference frames. It then becomes possible to script a movie of the approach and interaction of the dynamic molecule and micelle and then to slab through the array (as shown) to illustrate details of interior micelle packing.

Examples of additional applications can be seen in References 1–6.

NOTE

MOLEDITOR is available on a limited basis. Apply to the author for a license agreement.

REFERENCES

1 Wendoloski, J.J., Matthew, J.B., Weber, P.C. and Salemme, F.R. Molecular dynamics of a cytochrome c-cytochrome b5 electron transfer complex. *Science* 1987, **238**, 794–796

2 Salemme, F.R., Genieser, Lars, Finzel, B.C., Hilmer, R.M. and Wendoloski, J.J. Molecular factors stabilizing protein crystals. *Journal of Crystal Growth* 1988, **90**, 273–282

3 Wendoloski, J.J., Wasserman, Z. and Salemme, F.R. Computer simulations of biological interactions and reactivity. *J. Computer-Aided Molecular Design* 1987, **1**, 313–322

4 Wendoloski, J.J., Kimatian, S., Schutt, C.E. and Salemme, F.R. Molecular dynamics simulation of a phospholipid micelle. *Science* 1989, **243**, 636–638

5 DeGrado, W.F., Wasserman, Z.R. and Lear, J.D. Protein design, a minimalist approach. *Science* 1989, **243**, 622–628

6 Lear, J.D., Wasserman, Z.R. and DeGrado, W.F. Synthetic amphiphilic peptide models for protein ion channels. *Science* 1988, **240**, 1177–1181