NEW PROGRAMS

MoG: Molecular graphics software for the Commodore Amiga

Andrew C. R. Martin

SciTech Software, Ashtead, Surrey, UK

As recently described by Garavelli, the Commodore Amiga 3000 computer is "nearly ideal" for desktop molecular modeling. The chief drawback to date, has been the lack of suitable software. This paper describes a new desktop molecular modeling package, MoG, which is suitable for both research and educational use. The speed of the Amiga 3000 means that MoG competes very favorably with software on IBM-PC machines, and its graphics capabilities allow excellent space-filling representations. The availability of cheap software-compatible home-computer versions of the Amiga places interactive molecular graphics within the reach of many senior high-school students, undergraduates and graduate students.

Keywords: desktop molecular modeling, computer graphics, personal computers

INTRODUCTION

The Commodore Amiga is a powerful personal computer, and the many reasons for choosing it as a machine for desktop molecular modeling have been described in detail by Garavelli.[1] Three basic software-compatible models are available, ranging from a 7.14-MHz Motorola 68000-based machine aimed at the home market, to the Amiga 3000. The latter is the most suitable machine for manipulation of larger molecular structures, and is based around the 25-MHz 68030 with a 68882 math coprocessor. It has a built-in display enhancer, which deinterlaces the high-resolution graphics modes for display on a multi-synch or VGA monitor. Thirty-two-bit memory is used throughout, and 68040 accelerator boards are now available.

In all models, the hardware includes custom sound and graphics coprocessor chips, and memory is easily expandable. The operating system, AmigaDOS, provides true preemptive time-sliced multitasking. It is built around a message-passing system (Exec), and the control of programs via the WIMP environment (Intuition) is event-driven in a manner similar to that used by X-Windows or OS/2 on IBM-PC hardware. The release of the Amiga 3000 has been accompanied by the availability of v2.0 of AmigaDOS, which has added many new features and has enhanced the Intuition environment.

The Amiga is capable of displaying up to 32 independent colors from a palette of 4096, 64 in *extra half-bright* mode (32 of the colors are half-intensity versions of the other 32) and 4096 in *hold and modify* (HAM) mode. HAM mode places restrictions on the placing of colors—only 16 register colors are available for placing at any position on the screen. Pixels are set to one of the other colors by copying two of the red, green, and blue components of the pixel to the left and modifying the third component. Twenty-four bit-plane graphics boards are now available allowing more than 16.7 million colors to be displayed at any position on the screen.

The Amiga supports a standard for file interchange known as interchange file format (IFF). IFF files use "forms" which may store (among other things) sound, text, and bit-mapped and structured graphics. Bit-mapped graphics are stored as IFF-ILBM (interleaved bit map) files, which are supported by all Amiga painting programs and multimedia software.

THE SOFTWARE

Work on an Amiga molecular graphics package was started in 1986, when high-quality graphics on other hardware was too expensive for personal use. Owing to other pressures, the development was halted until the spring of 1991, when the existing code was rewritten and extended substantially.

MoG uses a 16-color, high-resolution, interlaced display (a noninterlaced display mode is also available). On American NTSC machines, this gives a resolution of up to 640 × 400. European PAL machines have a higher vertical resolution of 512, while machines running under AmigaDOS v2.0 support overscan displays of up to 724 × 566. The animated display of stick representations of molecules uses double buffering for smoothness of rotation and, on an accelerated

Color Plates for this article are on page 46.

Address reprint requests to Dr. Martin, SciTech Software, 23 Stag Leys, Ashtead, Surrey, KT21 2TD, UK.

Received 28 January 1992; revised 19 February 1992; accepted 25 February 1992

Table 1. Times in milliseconds required to take a rotation step and draw a typical molecule containing the number of atoms (*N*) indicated. The 68000 timings were performed on an Amiga 1000—timings on an unexpanded Amiga 500 or Amiga 2000 with at least 1 Mbyte of memory will be identical. The 68030 and 68030/68882 timings were performed on a 25 MHz Amiga 3000; the 68030/68882 timings were performed with the version of MoG compiled to make use of the math coprocessor. The *blanking* time is the time required to clear a 540 × 400 display window. The *per atom* and *blanking* times were calculated by singular value decomposition solution.[8]

	Machine configuration		
N	68000	68030	68030/68882
10	135	104	100
50	306	132	118
100	535	170	132
500	2375	562	288
1000	4667	1063	493
1500	6833	1542	708
Per atom	4.519	0.974	0.406
Blanking	95.158	82.645	92.634

machine like the Amiga 3000, the speed of animation of molecules up to a few hundred atoms is limited mainly by the time taken to clear the screen between frames. This time is virtually constant across the computer configurations tested, as it makes use of the graphics coprocessor (blitter) chip. Timings for rotating molecules of sizes between 10 and 1500 atoms on different machines using a 640 × 400 screen are shown in Table 1. Rotation of 1000 atoms on an accelerated machine takes 0.4 seconds, which compares with 1 second for PCDRA.[2] Note, however, that the timing for PCDRA is based on split bonds, while that for MoG is not. The interactive display is clipped 100 pixels from the righthand side of the screen; the righthand portion of the screen contains 14 "gadgets", which when clicked with the mouse, cause rotation about *x*, *y*, or *z* axes, translation along the axes, and scaling of the image. Color Plate 1 shows[3] the Fv region of the antibody Gloop 2 as displayed by MoG.

Atoms may be labeled by clicking on them. Optionally, coordinate information will be displayed when an atom is clicked, and distances, angles, and torsion angles may all be measured. The current view matrix may also be obtained, and the view may be reset by entering a particular view matrix.

As is conventional with molecular graphics software, the three-dimensional structure is projected onto the two dimensions of the screen using an orthogonal (nonperspective) projection. A simple form of depth cueing is available whereby parts of the structure farther away are colored at two-thirds of the brightness of the foreground bonds. This may be switched off if the user requires additional independent colors.

MoG allows structures to be built on screen. One can add molecular fragments to the display and manipulate them independently. A library of fragments is supplied with the program. Bonds can be formed or broken, and rotations can be performed about torsion angles, with the current torsion angle being displayed during the rotation.

Screen images may be saved as IFF-ILBM files which may then be read into painting programs, multimedia or desktop publishing software. For higher quality printed output, PostScript (including EPSF) and HPGL are supported. A public domain HPGL interpreter allows diagrams to be printed on any Amiga-supported printer at the highest available resolution. Similar PostScript interpreters are also available.

Coloring

Extensive atom coloring options are available. Bonds may be colored on the basis of atom type, residue type, or residue number range. Currently, the software does not support "split" bonds, i.e., each bond will only appear in one color; a bond between a carbon and an oxygen will not be split at the midpoint into the colors specified for carbon and oxygen. Rather, in such a case, the color specified last will take precedence. To alleviate this problem to an extent, the color of bonds between particular atom pairs may be specified. The decision not to include split bond coloring was made to speed the animation of larger structures. If split bonds are to be considered, one must either calculate the midpoint of every bond, or store these points and rotate them with the rest of the structure. An interactive palette requester allows each of the 16 pens to be set to one of the 4096 available colors using RGB (red, green, blue) or HSL (hue, saturation, luminance) coloring models.

In addition to interactive coloring options, coloring files may be created with any ASCII editor. These use four commands (ATOM, TYPE, ZONE, and LINK) to specify coloring, and a fifth command (PEN) to set the palette. Typically, a coloring file might be created to color the backbone of a protein differently from the sidechains. For example, the command file shown in Figure 1 will color the backbone in blue and the C=O bonds in red. The default color for the rest of the structure (pen 1) is set to green.

```
! Set the foreground pens to green, blue and red.
PEN 1  0 15  0
PEN 2  0  0 15
PEN 3 15  0  0
! Set the background pens to darker versions.
PEN 7  0 10  0
PEN 8  0  0 10
PEN 9 10  0  0
! Now define the colours of the backbone
LINK N  CA 2
LINK CA C  2
LINK C  N  2
LINK C  O  3
```

Figure 1. A sample coloring file that will set the default pen color to green, with backbone atoms colored in blue and red. The red, green, and blue color components of each pen are set using values between 0 and 15.

Space-filling images

Space-filling representations of molecules may be generated using the separate program, CPK. This program may be run completely separately, or from within MoG. Three types of space-filling pictures may be generated: quick preview (a simple sphere stamping z-buffer algorithm), Phong shading, or full ray-tracing. The latter two methods are time-consuming, and particularly on an unaccelerated machine, may take some hours to render a picture. However, the quality of the images is high, with 64 colors (EHB mode) or 4096 colors (HAM mode). Once generated, the images may be saved as IFF-ILBM files. Color Plate 2 shows a preview image of the antibody[3] Gloop 2, while Color Plates 3 and 4 show ray-traced images of crambin[4] and B-form DNA.[5]

File formats

Rather than attempt to read and write the many coordinate formats that various pieces of software use, MoG uses its own format for storing coordinates. There are arguments for and against introducing yet another coordinate format, but it was felt that the advantages outweighed the disadvantages. The format adopted is quite simple. The first record contains the number of atoms and is used when reading the file to allocate sufficient memory to store the structure. This is followed by the minimum and maximum x, y and z coordinates that are used to perform the initial scaling of the image. The actual coordinate records contain the x, y and z coordinates, 4 links that indicate offsets to bonded atoms (these do not contain redundancy, and may be specified in the forward or reverse direction), and atom and residue label information.

Two utilities are supplied for creating MoG format files from PDB and CSSR files and vice versa. These utilities may be run as stand-alone programs, or from within MoG. The creation of MoG files from PDB files requires the calculation of connectivity. This is done on the basis of distances calculated within residues, and between C and N or O3′ and P of adjacent residues. Preparation of a MoG file from a 1000-atom PDB file takes 16 seconds on an Amiga 3000 making no use of the math coprocessor. Source code for a simple conversion utility is supplied, allowing the user to write conversion routines for other file formats.

Coding details

MoG is written in C, with the exception of one routine, ClipDraw( ), which is written in 68000 assembler for speed. ClipDraw( ) uses the Cohen-Sutherland clipping algorithm[6,7] to clip a line for display, and directly calls a system routine to draw a line on the screen. It is thus central to the graphical display process. The dynamic memory allocation available in C allows the size of molecules that may be displayed to be limited only by the amount of available memory.

Two versions of the software have been produced. The basic version will run on any Amiga having at least 1 Mbyte of memory (1.5 Mbytes are required to use MoG and CPK simultaneously). The second version requires a 68020, 68030 or 68040 processor and a 68881 or 68882 math coprocessor. This version is recommended when structures containing more than 100 atoms are being displayed and when Phong-shaded or ray-traced images are to be generated.

DISCUSSION

MoG provides the basis of a powerful graphics display and modeling program for the Commodore Amiga. In addition to its applications in research, the popularity of the Amiga as a home computer means that it is accessible to many school students and undergraduates. MoG thus provides chemistry, biology, and biochemistry students with the ability to investigate stereochemistry and the structure of organic molecules and proteins.

Clearly, there are many areas in which the software could be developed further. The two most obvious improvements are to add split bonds and a slab facility to clip the view of the structure along the z axis. Currently, atom selection is performed by the data conversion utility; there is no option within MoG to display a subset of the loaded atoms. There are a number of powerful ray-tracing and rendering programs available for the Amiga, some of which support 24-bit graphics boards. Rather than further duplicating such rendering software to produce CPK images by improving the current CPK program, it is intended to create an interface to generate input files for these programs.

The Amiga supports a version of the interprocess communications and scripting language, REXX. The Amiga version (ARexx) is now supplied as part of the system software, and ARexx support will be added to MoG. This will allow the integration of MoG with hypertext software, making the system ideal for educational purposes. In addition, it is hoped to add least-squares fitting, amino acid substitution, and simple energy calculation and minimization facilities.

REFERENCES

1 Garavelli, J.S. Molecular modeling on the Commodore Amiga. *J. Mol. Graph.* 1991, **9**, 24–26

2 Afshar, M., Nanard, M., Sallantin, J., and Haiech, J. PCDRA: PC interactive molecular representation and modeling system. *J. Mol. Graph.* 1990, **8**, 39–44

3 Jeffrey, P.D., Griest, R.E., Taylor, G.L., and Rees, A.R. Crystal structure of the Fab fragment of the anti-peptide antibody Gloop2 and 2.8 Å. manuscript in preparation

4 Hendrickson, W.A. and Teeter, M.M. *Nature (London)* 1981, **290**, 107

5 Drew, H.R., Samson, S., and Dickerson, R.E. *Proc. Natl. Acad. Sci. USA.* 1982, **79**, 4040

6 Kingslake, R. *An Introductory Course in Computer Graphics.* Chartwell-Bratt, Bromley, Kent, UK, 1986

7 Ammeraal, L. *Programming Principles in Computer Graphics.* John Wiley and Sons, New York, 1986

8 Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T. *Numerical Recipies, The Art of Scientific Computing.* Cambridge University Press, Cambridge, 1986