An extensively modified version of MolScript that includes greatly enhanced coloring capabilities

Robert M. Esnouf[1]

The Laboratory of Molecular Biophysics, and The Oxford Centre for Molecular Sciences, Oxford, UK

Owing to its flexibility, MolScript has become one of the most widely used programs for generating publication-quality molecular graphics. Integration with the Raster3D package, to allow the production of photorealistic rendered images, has increased its popularity still further. However, this intensive use has shown the need for enhancement of some areas of the program, especially for controlling the coloring of atoms, bonds, and molecules. This work describes a heavily modified version of MolScript that has added syntax for describing complicated coloring schemes and also has new graphics commands. Enhancements include drawing split-bond ball-and-stick models, smoothly varying the color of molecules (color ramping), abrupt color changes within secondary structural units, and the creation of dashed bonds. Making use of these added features is simple because all MolScript syntax is still supported and one typically needs only to add a few control commands. The final section of this article suggests some uses for this modified MolScript and provides illustrative examples. © *1997 by Elsevier Science Inc.*

Keywords: molecular graphics, visualization

INTRODUCTION

The general-purpose molecular graphics program MolScript[1] is one of the most widely used programs for generating images for publication and lectures. Not only can it be used for detailed pictures of structures (CPK representations of atoms; ball-and-stick representations of molecules), it can also be used to draw simplified, schematic representations of macromolecules, especially proteins. These schematics include α-carbon (C_α) traces, smoothed coils through the protein backbone, arrows for β structure, and helices for α structure. The great flexibility inherent in the MolScript syntax allows for unlimited combinations of these simple graphical units in a single picture and significantly adds to the utility of the program.

The first versions of MolScript produced solely PostScript (Adobe Systems, Inc.) output, but version 1.4 of the program also includes an option allowing generation of photorealistic images when used in conjunction with the rendering program from the *Raster3D* package.[2,3] The rendering program allows different lighting effects, shadow casting, and (from version 2.2) objects to be rendered semitransparently.

Many other programs are also used for the generation of figures, such as *O*,[4] *GRASP*,[5] *RIBBONS*,[6] and *INSIGHT* (Biosym/MSI). Although these programs have many features in common with MolScript, each program has its own strengths and weaknesses. Two particularly useful features that MolScript does not support are split-bond coloring of ball-and-stick models (where each half of a bond is colored depending on atom types) and a "color-ramping" facility similar to that implemented in *O*[4] (where the color of each little graphical segment making up the molecule is given a color dependent on some property of the molecule). Two simple color-ramping examples are coloring according to the position in the primary sequence of a protein to produce a rainbow of colors running along the structure, and coloring according to atomic *B* factor to show mobile regions of a molecule.

When preparing novel and difficult figures most effects can be obtained by using a cocktail of existing programs, but this is often time consuming and the results not quite ideal. Better results could be obtained if a single program were upgraded to perform the most commonly desired tasks. The high quality of output from MolScript (particularly in combination with *Raster3D*), the ready availability of source code, and the large number of users make MolScript the obvious choice for such further development.

The version of MolScript described here has many signifi-

Color Plates for this article are on pages 112–113.

Address reprint requests to: Robert M. Esnouf, The Rega Institute for Medical Research, Katholieke Universiteit Leuven, Minderbroedersstraat 10, B-3000 Leuven, Belgium.

Received 23 January 1997; revised 27 March 1997; accepted 4 April 1997.

[1] Present address: The Rega Institute for Medical Research, Katholieke Universiteit Leuven, Minderbroedersstraat 10, B-3000 Leuven, Belgium.

Journal of Molecular Graphics and Modelling 15, 132–134, 1997
© 1997 by Elsevier Science Inc.
655 Avenue of the Americas, New York, NY 10010

1093-3263/97/$17.00
SSDI S0263-7855(97)00021-0

cant internal changes but still accepts all the syntax of MolScript version 1.4 and produces (virtually) identical output from it. New code for drawing sticks allows split-bond (and smoothly varying) coloring of bonds. The low-level graphics routines have been modified to allow each graphical segment to have its own color, thus making complex coloring schemes possible. Extra syntax has been added to take advantage of these (and other) new features. In particular, the syntax for defining color ramps allows structures to be drawn having almost any conceivable combination of constant colors, jumps in color, and smoothly varying color. Finally, to create a clear figure it is crucial to determine the right viewpoint, scale, and slab: new code allows views from *O*[4] (based on the *o2mol* program) and from *MidasPlus* (UCSF Computer Graphics Laboratory) to be incorporated easily into MolScript input files.

Institutions that have a license for MolScript (contact Dr. Per Kraulis, krpx@sgikrpc.sto.se.pnu.com, for license information) may obtain an executable version (preferably for SGI machines) and further documentation for this modified program from the author by e-mail to robert@biop.ox.ac.uk.

NEW GRAPHICS FEATURES

In MolScript version 1.4 a bond is represented by a single (tapered) cylindrical stick that is given a single (depth-cued) color. In the modified program a new parameter, *sticksegments* (default value 1), is introduced to control the number of segments each bond is divided into, and each segment can be individually colored and depth cued. By default, the resulting output looks identical to normal MolScript irrespective of the value of *sticksegments*. However, in conjunction with a color ramp statement (see below), setting *sticksegments* to 2 can produce split-bond coloring and with a large number of stick segments smoothly varying colors (also jumps in color, stripes, etc.) can be produced.

A second new graphics feature for PostScript output is provided by the *pscpk* parameter. In addition to the normal "windowed" CPK atoms, simple circles and more realistically lit and shaded CPK representations can be produced. The former are useful for drawing small spheres that retain their colors; the latter give a more convincing "rendered" appearance for large atom spheres. These representations may be freely mixed within a figure.

A way of dashing bonds, hydrogen bonds, and other objects has been implemented, although currently the syntax can be rather inconvenient (being a special case of color ramping). However, with care the results can look good, especially for output rendered with *Raster3D*.

Finally, the program allows solid planes to be defined for rings of atoms. The program recognizes the rings in standard amino acids and nucleotides automatically, but planes can also be defined that have arbitrary sets of atoms as vertices. Illustrating rings in this way simplifies complex pictures because the eye of the reader is drawn to the ring as a whole and not to individual atoms. It also provides a surface against which shadows can be cast, thus enhancing the 3D effect with *Raster3D*.

COLOR RAMPING

This version of MolScript extends the idea of color ramping to cover a wide variety of coloring schemes, not just smooth color changes over the sequence of a molecule. Color ramps are defined using the extra MolScript command *colour* followed by a definition of the color scheme. If a color ramp is defined for a graphical segment then the segment will be colored according to the ramp; otherwise the normal MolScript color definitions apply. This preserves compatibility with MolScript version 1.4 and makes using the ramps easier because one typically only needs to add a *colour* command (and possibly a few parameter changes) to an existing MolScript input file.

The process of coloring a graphical object (e.g., a small segment of helix) based on a color ramp is a two-stage process. The position within a ramp is described by a "ramp fraction" between 0.0 (at the start) and 1.0 (at the finish). The first stage in defining a ramp relates each ramp fraction to a color. The second stage occurs when graphical segments are drawn: for each segment the relevant ramp fraction is calculated in a manner dependent on the type of ramping (e.g., depending on position in the sequence or atomic *B* factor) and then the ramp fraction is used to obtain the color.

The syntax for defining the relationship between ramp fraction and color is straightforward: for example, the syntax defining a ramp that varies from blue to red is *from blue to red*. The color after the word *from* is used for the ramp fraction of 0.0 and the color after the word *to* is (usually) used for the ramp fraction of 1.0. Colors for intermediate fractions are simply interpolated from these control points based on the hue–saturation–brightness (HSB) color model, following the shortest path. Hence, this color ramp has magenta at its midpoint. To specify a ramp going via green one simply inserts *via green* into the definition. Multiple *via* statements can be used to insert extra control points and the ramp fraction at which each color should be produced can also be specified (e.g., *via green at residue A94*). If two control points for different colors are created at the same ramp fraction then a jump in color occurs at that point. Ramps may be defined in terms of other ramps and they can also be made to repeat, giving the syntax vast flexibility. One special case suppresses the output of graphical segments, thus allowing dashed structures to be produced.

MolScript calculates the position of each graphical segment on the basis of two atoms (one in the case of CPK spheres) and a fractional distance between these two atoms (e.g., in helices this relates to the *segments* parameter). For color ramping, the two atoms and the fraction are used to calculate the ramp fraction according to the type of ramp. Three types of ramp are implemented: ramping according to position in the primary sequence; ramping according to the atomic *B* factor, and ramping according to the colors defined for each atom (e.g., with the *set atomcolour* statement). The ramp fraction can be calculated from both the atoms and the fraction (*continuous* ramping; default for sequence and *B* factor ramps), from whichever atom is closest to the graphical segment (*discrete* ramping; default for atom color ramps), or solely from the fraction (*fractional* ramping; used for dashed structures). Each ramp type has its own sensible default definition and color scheme, but all features can be altered. For example, the residues defining a sequence ramp need not correspond to the whole molecule, nor need they form a single contiguous sequence. Also, the range of *B* factors over which the color varies can be altered (values outside this range are mapped to the colors at either end of the ramp).

EXAMPLE APPLICATIONS

Color Plate 1[7] shows new possibilities for representing the structure of a small molecule. The split-bond view (Color Plate 1a) uses lit and shaded spheres in PostScript, has the number of *sticksegments* set to 2, and the structure is drawn with the color ramp *colour ball-and-stick by atom.* Color Plate 1 b and c show the results of trying to fit the different enantiomers of this molecule to X-ray data for a complex between one of the enantiomers and a protein. The capped-stick models are colored according to the refined atomic B factors for each enantiomer. This representation highlights that the variation in B factor is much greater for the enantiomer shown in Color Plate 1c (the incorrect one), emphasizing that these X-ray data (to 2.6-Å resolution) are sufficient to distinguish between enantiomers.

Color Plate 2[8] shows how this version of MolScript can be used to draw dashed bonds (and other interactions) as well as the result of drawing planes for rings of atoms. Dashed bonds are, in fact, drawn as a series of cylinders with the curved and flat surfaces lit appropriately. If rendered with shadow casting, each segment would also cast its own shadow. Drawing planes for rings of atoms simplifies the picture and makes the ring orientation more obvious.

Color Plate 3[9–11] demonstrates several possibilities for coloring the same secondary structure representation of a protein (in this case triosephosphate isomerase, TIM; PDB code 6TIM[9]. Color Plate 3a shows the default sequence color ramp running from blue via green to red. Such coloring schemes show clearly the order of, and connections between, helices and strands. Color Plate 3b shows the variation in atomic B factor for the protein: high B factors (red) are typically located on the surface of the molecule. Replacing B factor values by some other property allows these properties to be similarly represented. Color Plate 3 c and d are both colored by the C_α atom colors of each residue. Color Plate 3c has the atom color set according to the degree of sequence conservation among a large number of TIM sequences. This emphasizes the active site and other conserved areas of the molecule. Color Plate 3d shows the distribution of charged side chains, highlighting that they are typically near to the protein surface.

ACKNOWLEDGMENTS

This version of the program is still MolScript, both internally and from the point of view of the user. I acknowledge the clear and well-commented code of Per Kraulis, without which such significant changes would not have been feasible. I also thank Dave Stuart and Jon Grimes for providing the initial stimulus and the army of (ex-)Oxford crystallographers for regularly requesting/demanding new features. Thanks also to Dave Stuart (Oxford), Derek Logan (Stockholm), and Geoff Barton (Oxford) for providing subject material for figures, to Richard Bryan for computing facilities, and to Steven Lee for help with figures. The Oxford Centre for Molecular Sciences is supported by the BBSRC, MRC, and EPSRC.

REFERENCES

1 Kraulis, P.J. MOLSCRIPT: A program to produce both detailed and schematic plots of protein structures. *J. Appl. Crystallogr.* 1991, **24,** 946–950

2 Bacon, D.J. and Anderson, W.F. A fast algorithm for rendering space-filling molecule pictures. (Abstract of paper presented at the Seventh Annual Meeting of the Molecular Graphics Society.) *J. Mol. Graphics* 1988, **6,** 219–220

3 Merritt, E.A. and Murphy, M.E.P. Raster3D version 2.0—a program for photorealistic molecular graphics. *Acta Crystallogr.* 1994, **D50,** 869–873

4 Jones, T.A., Zou, J.Y., Cowan, S.W., and Kjeldgaard, M. Improved methods for building protein models in electron density maps and the location of errors in these models. *Acta Crystallogr.* 1991, **A47,** 110–119

5 Nicholls, A., Sharp, K., and Honig, B. Protein folding and association: Insights from the interfacial and thermodynamic properties of hydrocarbons. *Proteins* 1991, **11,** 281–296

6 Carson, M. Ribbons 2.0. *J. Appl. Crystallogr.* 1991, **24,** 958–961

7 Pauwels, R., Andries, K., Desmyter, J., Schols, D., Kukla, M.J., Breslin, H.J., Raeymaeckers, A., Gelder, J.V., Woestenborghs, R., Heykants, J., Schellekens, K., Janssen, M.A.C., De Clercq, E., and Janssen, P.A.J. Potent and selective inhibition of HIV-1 replication *in vitro* by a novel series of TIBO derivatives. *Nature (London)* 1990, **343,** 470–474

8 Logan, D.T., Su, X.-D., Åberg, A., Regnström, K., Hajdu, J., Eklund, H., and Nordlund, P. Crystal structure of reduced protein R2 of ribonucleotide reductase: The structural basis for oxygen activation at a dinuclear iron site. *Structure* 1996, **4,** 1053–1064

9 Noble, M.E.M., Wierenga, R.K., Lambeir, A.-M., Opperdoes, F.R., Thunnissen, A.-M.W.H., Kalk, K.H., Groendijk, H., and Hol, W.G.J. The adaptability of the active site of trypanosomal triosephosphate isomerase as observed in the crystal structures of three different complexes. *Proteins* 1991, **10,** 50–69

10 Barton, G.J. Protein multiple sequence alignment and flexible pattern matching. *Methods Enzymol.* 1990, **183,** 403–428

11 Livingstone, C.D. and Barton, G.J. Protein sequence alignments: A strategy for the hierarchical analysis of residue conservation. *Comput. Appl. Biol. Sci.* 1993, **9,** 745–756