Tek_FRODO: A new version of FRODO for Tektronix graphics stations

Dmitriy G. Vassylyev and Alexei A. Adzhubei

Engelhardt Institute of Molecular Biology, Moscow, Russia

A new version of the molecular graphics program FRODO was developed to allow the range of Tektronix graphics stations to be used for molecular modeling and crystallographic applications. The work was divided into two parts: first, the universal molecular modeling graphic package (Tek_MMGP) was written to enable basic modeling operations for Tektronix stations. Second, all routines of FRODO involving computer graphics were modified to fit the new hardware environment, and linked with Tek_MMGP. The resulting package, Tek_FRODO, has been used successfully for crystallographic refinement in several projects. The program, written in FORTRAN, is ready to be ported to any of Tektronix 3D graphics stations; it is available from the authors on request.

Keywords: molecular modeling, FRODO, protein structure, crystallographic refinement, Tektronix

INTRODUCTION

Several graphic programs have been developed in recent years to meet the demand for fast and reliable data processing and modeling of the three-dimensional (3D) structure of biological macromolecules. One of the first and most widely used was the interactive molecular modeling program FRODO.[1] It has been applied mainly to crystallographic model building and refinement of 3D structures of biological macromolecules but can be used successfully also in the modeling of protein–substrate, protein–DNA interactions, etc. The first version[1] was written by T.A. Jones in 1976 for Vector General VG 3400 display, and since then upgraded versions of the program have been developed and modified for a variety of graphics stations.[2–5]

The aim of this work was to develop a molecular modeling package incorporating FRODO for Tektronix 3D graphics stations. We intend to use this package in our crystallographic projects, protein engineering and protein structure comparison studies.

Color Plates for this article are on page 110.

Address reprint requests to Dr. Adzhubei at his present address: Biomolecular Modelling Laboratory, Imperial Cancer Research Fund, Lincoln's Inn Fields, P.O. Box 123, London, WC2A 3PX, U.K.

Received 3 September 1991; revised 16 October 1991; accepted 22 October 1991

TEKTRONIX MOLECULAR MODELING GRAPHIC PACKAGE (Tek_MMGP)

The package was designed especially to be compatible with the whole range of Tektronix 3D graphics stations, starting from the lower end Tektronix 4129 (700KB main memory, INTEL 80286 graphic processor, 8 bit planes of display memory, 1280 × 1024 pixel screen resolution, 50000 vectors/second, Tilling processor). We used a MicroVAXII as a host computer, connected over an RS 232 interface to a Tektronix terminal. One of the main problems was that we had to write a package equally suitable for low- and high-performance graphics stations. This task was most complicated for the Tektronix 4129, where the comparatively low speed of the graphic processor was combined with the absence of hardware support for double buffering, no analog-to-digital converters, and only 8 bit planes of display memory (4 bits or 16 color indices in double buffering). Tektronix graphics logic is also very different from that of the Evans & Sutherland PS300 series for which PSFRODO,[2] the program we used as a starting version, was originally written. Proceeding from this, we decided to develop a basic molecular modeling graphic package, that would allow us to port easily any interactive graphic program to our graphics station; to avoid difficulties when software modification was required; and to adopt programs for different Tektronix graphics stations without large modifications.

Tek_MMGP is written in ANSI standard FORTRAN 77. However some nonstandard VAX FORTRAN elements are used in the program: symbolic names include underscore and contain more than six characters; END DO is used to terminate the range of a DO statement; INCLUDE statement is used to read statements in the included file. Only standard FORTRAN intrinsic functions are referred to by Tek_MMGP subprograms.

The final Tek_MMGP includes the following program packages:

(1) The package BASE_4129 provides software interface between the host computer and a Tektronix graphics station. We did not use PLOT10 or any other standard

graphic packages in our work; thus BASE_4129 serves to link FORTRAN user programs with a Tektronix graphics commands interpreter. We tried to make this interface as simple as possible for application programming. For example, we wrote special programs to activate graphics input (GIN) devices (mouse, thumbwheels, and tablet) and to accept data from them by their mnemonic names instead of digital codes. Thus

```
CALL ENABLE_GIN_L1('MOUSE',
                   'PICK','PRESS',100)
```

is used to enable graphics input for 100 mouse *pick* operations in the *press* mode. One subprogram call gives all necessary information on possible reports from the graphics station to the host computer. It includes also a mnemonic name for subsequent reports (MACRO, GIN devices, INQUIRY reports, etc.). In this manner we simplified work with dialogue areas, and the generation of macro definitions as well.

(2) Tek_MENU is designed for the fast and easy generation of graphic menus on a Tektronix screen. The menu has tree-type structure and can integrate up to 10 logical levels. Each menu item that is not related to a submenu on a higher level can be activated. In this case the matching logical flag is set to .TRUE. and the command word is highlighted. Two types of menu can be generated on the screen:

- A vertical *y*-menu, which can consist of up to 512 menu items having 40 commands on one level, controlled by special graphic cursor. The font size and spacing between command words in the *y*-menu is automatically scaled by the program as a function of the number of control words in the current menu.
- A horizontal *x*-menu having a maximum of 512 commands and up to 16 items in one menu. (This menu's commands are operated by functional keys on a Tektronix keyboard.) On the screen, the *y*-menu occupies a column on the right side and the *x*-menu is positioned in two lines at the bottom.

Tek_MENU includes 20 routines (approximately 4000 FORTRAN operators) but is very simple in use for application programming. Two subroutine calls should be used: one to define the *y*- or *x*-menu and another to interpret menu information. The following example illustrates a *y*-menu defined in the user program.

```
CALL DEFINE_MENU_Y(1,'END',
          'WORD1$WORD2$WORD3$WORD4$')
CALL DEFINE_MENU_Y(2,'WORD1',
              'WORD11$WORD12$WORD13$')
CALL DEFINE_MENU_Y(2,'WORD3',
              'WORD31$WORD32$WORD33$')
CALL DEFINE_MENU_Y(3,'WORD32',
            'WORD321$WORD322$WORD323$')
```

where first parameter defines the level of the current menu; the second defines the reference control word at the upper level (the first-level reference command can be used to abort the program); and the third defines the set of control words (up to 40) that appear on the screen. Our experience shows it is very easy to create and modify menu trees from Tek_MENU for any graphics program, and to use menu information inside the application program.

(3) Tek_DSTR helps one to develop interrelated display structures that can be referred to as independent fragments of the image in the double buffering mode. The implementation of the display structure logic was necessary to avoid blinking of the screen while rotating or moving objects, etc. in real time. Display structures (DS) allow one to simplify the programming of complicated pictures. These structures form trees, so they could be addressed (rotated or moved) separately, but a DS will be moved (rotated or scaled) along with any DS of the corresponding higher logical level. By default, all display structures on the first logical level will be rotated, moved and scaled jointly, but they (and subsequently their substructures) have unique special attributes, which gives the possibility of excluding a specified DS from any operation such as rotation, translation, scaling or stereo viewing. Moreover, a DS on any level has its own features, such as color, visibility and number of allowed Tektronix graphic segments. Up to 100 different DS can be created by one program. Though Tek_DSTR includes approximately 3000 FORTRAN commands, its interface allows it to be used quite easily in application programming.

(4) Tek_DBUF (2500 operators) emulates the double-buffering mode on graphics stations that have no hardware support for double buffering. The program is written as an independent part of the Tek_DSTR complex and operates with DS in the double-buffering mode. Its main task is to relate the DS to a particular double-buffered operation. It also allows one to work with an object in stereo mode. Tek_DBUF is universal, simple to use and easy to implement.

(5) Tek_PSEUDO_AD (pseudo analog-to-digital converters; i.e., dial emulation) is the molecular-oriented program package that performs transformation of a 3D wire frame object or its fragments. The main operations are: rotation around *xyz*-axes, 3D translation, rotation around predefined vector (torsion rotations), scaling, moving of the selected atom with distance control (bonds are set to "rubberbanding" mode), generation of stereo and orthogonal projection images, "rocking" and color selection for an object, atom labeling and close contacts control. Tek_PSEUDO_AD has 3000 operators, including calls to Tek_DSTR and Tek_DBUF packages and VMS asynchronous input/output routines. This package, as well as the Tek_MMGP package as a whole, is openly structured for modification, so that it can be altered quickly for use with other graphics programs and hardware equipment.

Tek_FRODO STRUCTURE AND OPERATIONS

After the Tek_MMGP package had been completed, it was possible to program a Tektronix-compatible version of FRODO without major difficulties. We modified all graph-

ics-dependent PSFRODO subroutines and linked them to the Tek_MMGP package. Since we did not change PSFRODO structure and commands, we do not give details of its operation in this paper. The description of FRODO can be found elsewhere.[1-6] We point out only that FRODO can be divided into several functional parts (programs): CHAT is used to set and change practically all parameters and files, which allows the user to generate the required part of electron density map and of the molecule, whose coordinates can be interactively modified. SAM lists, changes, inserts and deletes coordinates of compounds present in the FRODO internal molecular file; it also produces formatted coordinate files for communication with external programs. INTER allows the user to modify interactively coordinates of molecular fragments generated on the screen, and to control distances, angles and coordinates of atoms during this work. REFINE is a model regularization routine used to correct stereochemistry of the fragment, usually after its interactive modification. MOLCUL generates 3D models of the molecule (coordinates cannot be changed interactively); it is used mainly for structure analysis and for illustrations. Each of the programs has its own set of commands; in Tek_FRODO these commands are menu-driven (from the screen of graphics station) for INTER routine, and have to be entered from the VAX terminal for other routines. The user manual[6] of PSFRODO remains generally the same for Tek_FRODO. The main differences are:

(1) All CHAT, REFINE, SAM and MOLCUL commands in Tek_FRODO are menu-driven and can be activated from the screen, but parameters in this version of the program are entered from the MicroVAX terminal because of the hardware limitations of the Tektronix 4129 station. This input is taken from the graphics display for more powerful graphics stations.

(2) Dial emulators are menu-driven and operated by *x*-menu commands. The speed of operation is controlled from a Tektronix keyboard.

(3) The molecular display routine (MOLCUL) was modified to generate space-filling, shaded images of molecules. For low-performance graphics stations, the generation time for complicated shaded objects, such as molecular models, becomes critical. Thus, for two main types of molecular models in our program—balls and sticks and Van der Waals (VDW) spheres—we designed special graphics primitives and applied an efficient algorithm to maintain balance of speed and quality of image generation. We also applied a special depth-cueing algorithm, based on the different reflectivities of 3D shaded primitives along the *z*-coordinate of the image, to increase the stereo effect of the picture (Color Plate 1). Picture generation and shading takes about 3 min. on a Tektronix 4129 (13 min. in the depth-cueing mode) for a ball-and-stick model of 1000 atoms and 1.5 min. (3 min. in the depth-cueing mode) for a WDW model containing 500 atoms.

APPLICATION RESULTS AND DISCUSSION

While Tek_FRODO, installed on one of the high-end Tektronix stations, would not differ practically from FRODO on an Evans & Sutherland or a Silicon Graphics terminal, its application on a Tektronix 4129 obviously has some limitations. Since the speed of operation with the full screen picture is not very high, we preferred to use the stereo mode and specially designed stereo glasses to work with complicated pictures of electron densities in real time, to fit atoms to an electron density map. The graphics system was completed and ready for use in January 1991, and we present below crystallographic projects in which Tek_FRODO was successfully employed. The first project was the refinement of catalase from *Bacterium Micrococcus lysodeikticus*. The authors used a homologous protein[7] to build the main chain model and fit some of the side chains to the density using the rotation function and automatic refinement procedures.[8] The first, most critical, improvement of the structure was made using Tek_FRODO and required about 120 hours on a Tektronix 4129. During this work 5 insertions and 4 deletions in amino acid sequence were made; 35 residues, which were far from electron density after refinement, were more accurately positioned (Color Plate 2). The crystallographic *R*-factor (33% at 3 Å resolution before graphic modeling) was reduced to 26.4% at 2.5 Å after refinement of the new coordinates.

Tek_FRODO also was used to set the guanosine vanadate model (24 atoms) correctly into a difference electron density map of its complex with ribonuclease C2 from *Aspergillus clavatus*.[9] The work took approximately one hour of Tektronix 4129 time, taking into account that it was necessary to control the ideal geometry of guanosine vanadate during manipulations with its fragments, since no standard geometry for this compound is available for regularization with the REFINE option of FRODO. In the protein itself (107 amino acids) positions of 4 amino acid side chains, that appeared to be far from the electron density were corrected. Tek_FRODO will be used in further refinement steps of this complex.

The initial interpretation of an electron density map for a protein of unknown structure, i.e., C_α-tracing, requires a large part of the electron density map to be present on the screen. To allow interactive operations with a large number of vectors on the relatively slow graphics station we introduced special "view" functions for rotation and "volume" functions for translation and scaling of the picture. The view functions can be used to rotate a picture around the *x*- (*y*-,*z*-) axes with a fixed angle increment (15°, 30° or 90°), and volume functions generate a box on the screen that can be moved and scaled interactively, defining the size and position of the picture. These options make it possible to work interactively with a large portion of the electron density map (about 30,000 vectors in the stereo mode), and are very useful for crystallographic model building. The initial model building of two protein structures, using a Tektronix 4129 station, is now in progress.

The original graphic/modeling package Tek_MMGP was written for a Tektronix graphics station. The structure of Tek_MMGP and of the modified version of FRODO, Tek_FRODO, allow the installation and use of these program complexes on low-end, as well as high-performance Tektronix machines. Tek_FRODO program can be used efficiently even on a relatively slow Tektronix 4129 station.

Tek_MMGP and Tek_FRODO programs are available from the authors on request (FAX No. 7095 9382187). Enquires can also be addressed to A.A.A., FAX No. (UK) 044 071 2693479.

ACKNOWLEDGMENTS

We thank Prof. T. Alwin Jones (University of Uppsala), the first author of FRODO, for the source codes of PSFRODO.

We would like to thank S.V. Korolev (Institute of Molecular Biology Academy of Sciences USSR) for his help with Tek_FRODO.

We thank W.R. Melik-Adomyan and G. Murshudov (Institute of Crystallography, Moscow) for their permission to use the intermediate results of the refinement of catalase; and K. Polyakov (Institute of Crystallography, Moscow) for permission to use the results of the refinement of ribonuclease complex with guanosine vanadate.

Tek_MMGP is registered in the State Fund of Algorithms and Programs, Russia.

REFERENCES

1 Jones, T.A. *J. Appl. Crystallogr.* 1978, **11**, 268
2 Jones, T.A. *Computational Crystallography* (D. Sayre, Ed.) Clarendon Press, Oxford, 1982, p. 303
3 Bush, B.L. in: *Computers and Chemistry*. Pergamon, Oxford, 1984, vol. 8, p. 1
4 Pflugrath, J.W., Saper, M.A. and Quicho, F.A. in: *Methods and Applications in Crystallographic Computing*. (S. Hall and T. Ashida, Eds.) Oxford Univ. Press, London, 1984, p. 404
5 Cambillau, C., Horjales, E. and Jones, T.A. *J. Mol. Graphics* 1984, **2**, 53
6 Sack, J.S., Bush, B.L., Jones, T.A., Pflugrath, J.W. and Saper, M.A. Documentation for PS300 FRODO: molecular graphics program for the PS300. 1988
7 Fita, I., Silva, A., Murthy, M.R. and Rossmann, M.G. *Acta Cryst.* 1986, **B26**, 497
8 Yusiphov, E.F., Grebenko, A.I., Barynin, V.V., Murshudov, G.N., Vagin, A.A., Melik-Adomyan, V.R. and Vainshtein, B.K. *Crystallography (USSR)* 1989, **34**, 1451–1456
9 Polyakov, K.M., Strokopytov, B.V., Vagin, A.A., Tischenko, G.N., Bezborodova, S.I. and Vainshtein, B.K. *Crystallography (USSR)* 1987, **32**, 918–926