27
6-3-80

ornl

OAK
RIDGE
NATIONAL
LABORATORY

UNION
CARBIDE

MASTER

ORNL/CSD/TM-109

# HLYWD: A Program for Post-Processing Data Files to Generate Selected Plots or Time-Lapse Graphics

J. K. Munro, Jr.

OPERATED BY
UNION CARBIDE CORPORATION
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

ORNL/CSD/TM-109

# HLYWD: A PROGRAM FOR POST-PROCESSING DATA FILES TO GENERATE SELECTED PLOTS OR TIME-LAPSE GRAPHICS

J. K. Munro, Jr.

Sponsor: J. T. Hogan

Date Published - May 1980

COMPUTER SCIENCES DIVISION
at
Oak Ridge National Laboratory
Post Office Box X
Oak Ridge, Tennessee 37830

**NOTICE** This document contains information of a preliminary nature. It is subject to revision or correction and therefore does not represent a final report.

Union Carbide Corporation, Nuclear Division
Operating the
Oak Ridge Gaseous Diffusion Plant . Paducah Gaseous Diffusion Plant
Oak Ridge Y-12 Plant . Oak Ridge National Laboratory
under Contract No. W-7405-eng-26
for the
DEPARTMENT OF ENERGY

DISCLAIMER

TABLF OF CONTENTS

LIST OF FIGURES . . . . . v

LIST OF TABLES . . . . . vii

ACKNOWLEDGMENTS . . . . . ix

ABSTRACT . . . . . 1

1. INTRODUCTION . . . . . 1

2. PROGRAM USE . . . . . 7

2.1. INPUT FILE PREPARATION . . . . . 7
2.2. CONTROLLEE GENERATION FROM SOURCE FILES . . . . . 9
2.3. INITIAL DIAGNOSTIC CHECKS . . . . . 10
2.4. MOVIE GENERATION . . . . . 10

3. PROGRAM DESCRIPTION . . . . . 11

3.1. LOGICAL STRUCTURE OF THE MAIN ROUTINE . . . . . 11
3.2. LOGICAL STRUCTURE OF SUBROUTINE SELECT . . . . . 11
3.3. INPUT FILES REQUIRED . . . . . 13
3.4. DESCRIPTION OF INPUT DATA . . . . . 15

4. PROGRAM APPLICATIONS . . . . . 25

4.1. REVIEW OF APPLICATIONS MADE DURING CODE DEVELOPMENT . . . . 25
4.2. REVIEW OF PROCEDURES FOUND TO BE USEFUL . . . . . 28
4.3. COMMENTS ON LIMITATIONS AND PROBLEMS WITH THE PROGRAM . . 32

REFERENCES . . . . . 35

APPENDIX A. Listings of Input Files . . . . . 37

APPENDIX B. Program Source Listing . . . . . 39

## LIST OF FIGURES

Figure 3.1. Diagram showing sequence of possible tasks desired to make a movie in production mode . . . . . 12

Figure 3.2. Structure of control routine used to post-process files and generate 3D plots . . . . . 14

Figure 4.1. Plot of a function using the parameter value NS=3 . . . 30

Figure 4.2. Plot of the same function shown in previous figure (but at a different time) using the parameter value NS=1 . . . . . 31

## LIST OF TABLES

Table 2.1. Format of a set of data to be processed into a 3D plot . . 8

Table 3.1. Input data formats for generating segments of text on film . . . . . 20

Table 3.2. Format of data in files to be post-processed . . . . . 23

## ACKNOWLEDGMENTS

The author wishes to thank J. T. Hogan for his interest in and support of the development of the program HLYWD.

This work was funded by the Fusion Energy Division at the Oak Ridge National Laboratory.

# HLYWD: A PROGRAM FOR POST-PROCESSING DATA FILES TO GENERATE SELECTED PLOTS OR TIME-LAPSE GRAPHICS

J. K. Munro, Jr.

## ABSTRACT

The program HLYWD is a post-processor of output files generated by large plasma simulation computations or of data files containing a time sequence of plasma diagnostics. It is intended to be used in a production mode for either type of application; i.e., it allows one to generate along with the graphics sequence, segments containing title, credits to those who performed the work, text to describe the graphics, and acknowledgement of funding agency. The current version is designed to generate 3D plots and allows one to select type of display (linear or semi-log scales), choice of normalization of function values for display purposes, viewing perspective, and an option to allow continuous rotations of surfaces. This program was developed with the intention of being relatively easy to use, reasonably flexible, and requiring a minimum investment of the user's time. It uses the TV80 library of graphics software and ORDERLIB system software on the CDC-7600 at the National Magnetic Fusion Energy Computing Center at Lawrence Livermore Laboratory in California.

KEY WORDS: Computer movies, Post-processor, 3D plots, Production mode, Modular software, Text frames, Applications

---

## 1. INTRODUCTION

The program HLYWD is a post-processor of output files generated by large plasma simulation computations or of data files containing a time sequence of plasma diagnostics. It is intended to be used in a production mode for either type of application; i.e., it allows one to generate along with the graphics sequence, segments containing title, credits to those who performed the work, text to describe the graphics, and acknowledgement of funding agency. The current version is designed

to generate 3D plots and has some features which have been found very useful. These features include options to select type of display (linear or semi-log), choice of scaling of function values, provisions to select viewing angle and to allow for continuous rotations of viewing perspective through a desired angle. This program was developed with the intention of being relatively easy to use, to have reasonable flexibility, and to require a minimum investment of time. It uses the TV80 [NMFECC on-line documentation] library of graphics software written at Lawrence Livermore Laboratory (LLL) and the system routines available in ORDERLIB [NMFECC on-line documentation] on the CDC-7600 at the National Magnetic Fusion Energy Computing Center (NMFECC) at LLL.

This program was originally written as a post-processor to make movies of the time history of various plasma properties as simulated by the Oak Ridge Tokamak Transport Code [Hogan (1976); Munro et al., (1977); Hogan (1978)]. The post-processing mode was quickly found to be the most efficient and reliable means for making movies because of the many problems [Munro (1979)] which one can encounter. Subsequent experience continues to reinforce the wisdom of separating time consuming computations of a large simulation code from the movie-making task. The post-processing mode allows the movie maker to try many ideas to improve the visual appearance and smoothness of flow of the action of plasma property histories contained in the families of files which are to be processed. One can decide whether a linear or semi-log display of function values works best, or whether the viewing angle allows the viewer to see the structure or features of a surface which are physically significant or of particular interest, e.g., if one is

looking for evidence of some process or effect. The post-processing mode greatly simplifies the job of diagnosing all kinds of problems which can arise.

The program is intended to allow one to use it on a production basis, i.e., to be easy to use, reliable, and require a minimum amount of time. For many types of applications, little or no further development should be necessary. Once adapted to a particular user's needs, the program should allow him to generate a basic, simple, but complete movie. One has the option of including film segments with a title, credits to program developer and users, explanations of the graphics to be shown, and acknowledgement of support from funding agencies. All these text segments can be omitted as desired during development and debugging of graphics software. One can choose how many times a text frame should be repeated or let the program determine the number of frame repetitions.

A modular design was followed as much as possible during the evolution of the program. The main subroutine reads the input for program control and display options, then calls a sequence of different routines to do specific tasks. The routine SELECT generates all the graphics from the files which must be post-processed. If one wants to use different graphics software, this is where one would have to begin making changes. Subroutine SELECT does a number of very useful basic functions, so one should be careful about throwing it out completely if one needs a different set of graphics programs, especially if one is post-processing a family of files.

The options available for determining how graphics are to be displayed should be sufficient for most applications. They represent a reasonable beginning and should allow one to make improvements or additions with a minimum of difficulty. One can choose options in several categories in such a way to allow for a number of combinations of ways to display surface plots. With minor changes in program logic, one can use a sequence of different options in a given movie to do a number of different types of display as part of a single post-processing job. The post-processing mode also permits one to use a set of different options in a sequence of separate program executions to post-process all or part of the family of input files.

Sometimes one discovers that the sets of numbers in the files to be post-processed do not permit one to get a smoothly flowing visual sequence in a movie. The changes from one frame to the next happen on such a fast time scale as to produce a jittery appearance on film. The program permits one to interpolate linearly in time to smooth out such jitter if this is desired. Interpolation can be used to slow the action if desired. One needs to be careful about using this feature lest the viewer get an impression of "physical effects" which, in fact, do not exist. This feature has a limitation that sometimes can be a problem: it assumes function values are associated with a fixed spatial grid.

In addition to its usefulness in making movies, this program can be used to generate families of graphics files which one can examine using the utility routine FR8ØPLOT [NMFECC On-line Documentation]. If one is not interested necessarily in generating an entire movie but desires to examine the time evolution of a process in detail to check

for effects which might otherwise be missed, one can in this fashion view selected frames or sequences of frames from the family of graphics files at a graphics terminal. One can then use the USC option of FR8ØPLOT to generate a small set of individual frames of graphics for output on a Versatec printer/plotter.

Section 2 describes in a step-by-step procedure how to use the program to generate a movie. Section 3 contains a brief description of the program, input files, and the input data required. Applications and discussion of procedures for using the program are given in Section 4. Sample input data sets and a program listing are given in Appendices A and B, respectively. These same data sets and source listings are available on the NMFECC system and may be obtained from FILEM by executing on the A76ØØ:

FILEM RDS 2553Ø .PUBLIC ALWITH. HLYWD

In the files which one gets, the source listings will be contained in consecutively numbered files HLYWDØ, HLYWD1, etc.; HLYWDIN1 ≡ INPUT, HLYWDIN2 ≡ INFIL. The sequence of files HLYWDIN3, HLYWDIN4, etc., are the leading members of a family of files to be post-processed and can be used to get some initial experience with the program. The program PLOT3D may be obtained from FILEM by executing on the A76ØØ

FILEM RDS 2553Ø .PUBLIC PLOT3D BPLOT3D

File BPLOT3D is the corresponding binary file.

# 2. PROGRAM USE

## 2.1. INPUT FILE PREPARATION

The following description assumes one plans to use the program the way it is currently written. To generate a movie, one needs to produce a sequence of graphs on film. These graphs are produced from a sequence of sets of numbers. For this program, each set of numbers must contain a title, the numbers of nodes in the x- and y-directions, the total number of values in the array to be plotted (currently required to be equal to the total number of nodes), length scales for the x- and y-axes, and the array of values to be plotted. The format of a set is described in Table 2.1. Use of a title at the beginning of a set allows one to intermingle different kinds of sets of numbers for different movies made from a single computation. For example, one could write sets of output for current density, temperature, pressure, particle density, etc., at each time step of a transport calculation. At each time step, the current density data set would have the title "current density," the ion temperature data set would have the title "ion temperature," etc. The titles can then be used as possible matches for a search pattern supplied to select which type of data set is to be read at each time step for the sequence of graphs for the movie.

The amount of output from a big calculation could be quite large, requiring a sequence of files. Each file in such a sequence is the member of a family. Each member is given a name based on the name of the first member by successively incrementing the right-most bit of the file name according to its representation in computer memory. Thus, if one anticipates a very large family of output files, one should choose

TABLE 2.1. Format of a set of data to be processed into a 3D plot

| Line Number (Relative) | Format | Variable List or Array | Description of Output |
|---|---|---|---|
| 1 | 2A10, A1 | TITLE(I), I=1,3 | A hollerith string used to identify the beginning of a set of data for a 3D plot. |
| 2 | 3I4 | ND | Total number of function values to be plotted (must equal total number of nodes = NN*NC) |
| | | NN | The number of nodes in the x-direction |
| | | NC | The number of nodes in the y-direction |
| 3 | 2F10.0 | SX | Scale (length) for x-direction |
| | | SY | Scale (length) for y-direction |
| 4 | 3X, 7E11.4 | PSAVE(I), I=1,ND | Array of function values to be plotted. |
| ⋮ | | | |
| $INT(\frac{ND}{7}) + 1$ | | | |

a name which ends with the integer zero. One would then also have to use that same name for HLYWDIN3, shown on the first line of the listing of file INFIL in Appendix A, to generate a complete movie for that family.

Two more files are required: the file INFIL just mentioned and a file INPUT (an example of this file is also listed in Appendix A). For the file INFIL, one only needs to replace the name HLYWDIN3 by the name of the member file where one wishes the movie to begin. For the file INPUT, one should refer to Section 3.4. for a detailed description of the NAMELIST option and control variables. The search pattern used to select the type of data set to be read is specified in the input parameter TTLCFØ. One's choice of values for the other input parameters depends on the type of movie one is making and will depend somewhat on what one learns from the experience of making a few movies. If one wants to make a fast diagnostic run to check whether graphics files are being generated correctly or whether the graphics are being displayed satisfactorily, one should set the parameters in file INPUT as follows: NS=3, IGFS=2Ø,ØØØ.

## 2.2. CONTROLLEE GENERATION FROM SOURCE FILES

To generate a controllee, one must load (binary) files in the following order:

HLYWD-*program-files* BPLOT3D, TV8ØLIB, ORDERLIB,

One must be careful to check the various HLYWD source files for PARAMETER statements in order to determine which ones need processing by the pre-compiler. This check is not necessary when using the CHATR compiler.

## 2.3. INITIAL DIAGNOSTIC CHECKS

It is usually well worth the time and effort to run the program using only the first member of the family of input data files and the parameter values NS=3, IGFS=20,000 in file INPUT to generate a family of small graphics files for use in making an initial diagnostic check. This check will allow one to establish whether the movie making program is working properly without possible waste of precious bank account time and will permit a quick visual inspection of the graphics using FR80PLOT at a graphics CRT or by printing one of the graphics files using NETPLOT with the F. option. A quick visual inspection is useful to be sure orientation, scale, etc. are reasonable.

## 2.4. MOVIE GENERATION

After one is reasonably sure input files and movie program are all set to go, one can then execute the program to generate the movie. If a diagnostic check was made, do not forget to reset the values of NS(=1) and IGFS (=300,000) in file INPUT. After the program has finished running, the FR80 files can be given to the system using

GIVE ENDWITH. *file-family-name* 2 K. END <CR>

One should consider saving graphics files in case film is lost and a new movie must be made.

# 3. PROGRAM DESCRIPTION

## 3.1. LOGICAL STRUCTURE OF THE MAIN ROUTINE

The main routine is a controller for a series of tasks. The modular approach is intended to allow a user to adapt the basic program structure more easily to his specific needs. Figure 3.1 shows the sequence of tasks requested by the main routine. These tasks are common to any production movie making. The logical control currently used only allows one to generate the graphics with or without any text frames and to do a preliminary search of files being post-processed to locate a global maximum value for the whole series of sets of numbers to be read and plotted. The global maximum value found can then be used to adjust scaling of the graphics. If desired, one could add to or modify the global maximum search task to obtain additional global values in order to make other adjustments in a series of graphics.

Text generating routines for text frames that will not vary from one production run to another are written as stand-alone units with all text contained within the routine. The only input required for these routines is the number of frames of the text which one desires. If this number has a value less than zero, the routine determines the number of frames that are generated on the basis of two seconds of viewing time per line of text. A user should only have to modify these routines one time to suit each application.

## 3.2. LOGICAL STRUCTURE OF SUBROUTINE SELECT

Subroutine SELECT contains the control logic for generating the graphics. The actual graphics calls are made in routine AFRAME. The files to be post-processed are read by routine GTFRAM. Routine ADVFIL

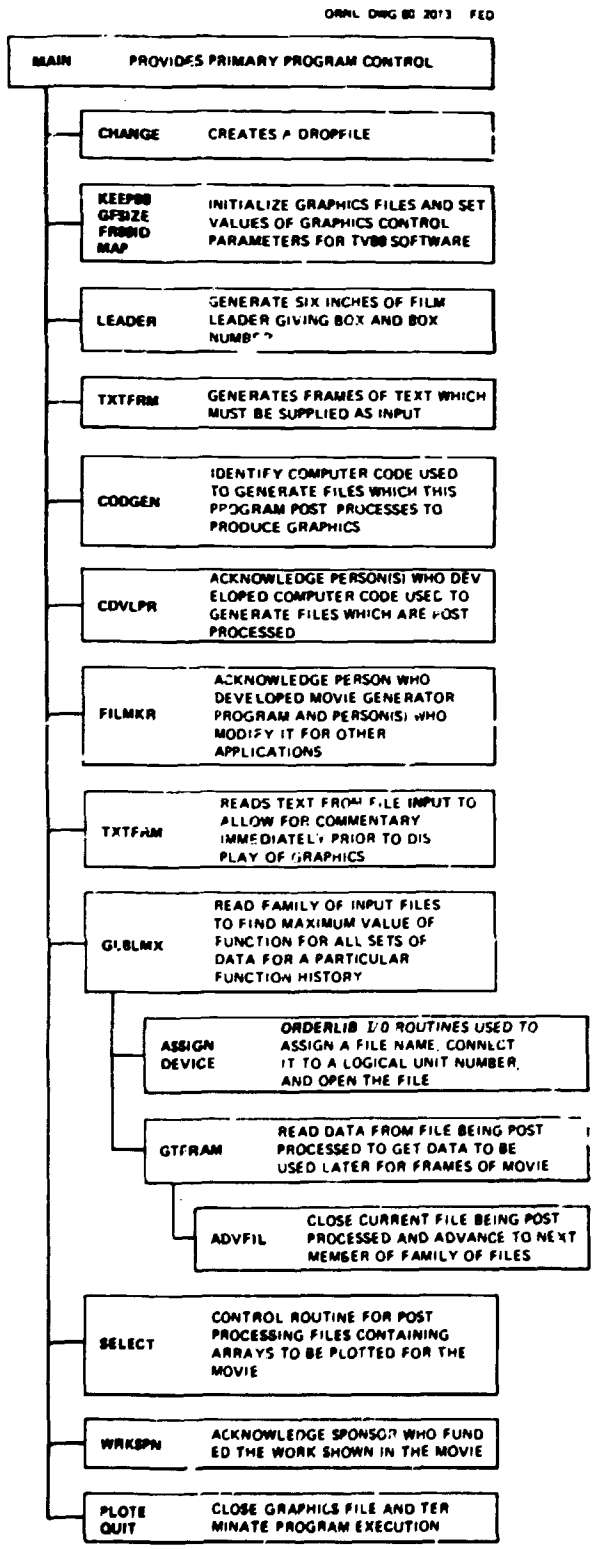


Figure 3.1. Diagram showing sequence of possible tasks desired to make a movie in production mode.

[Munro (1979)] advances from one member of the family of files being post-processed to the next. Scaling of arrays to be plotted is done by routine NORMLZ; transformation of viewing direction and change of the number of nodes used for plotting are handled by routine FARRAY [Munro (1979)]. Routine SELECT was designed with the objective to permit a user to adapt the movie generator program to be used for other applications than to generate 3D plots. The original intent was to use a structure that would require minimum modifications except for routine AFRAME which a user would have to write for himself if he wanted to use the program for a different application. The structure of routine SELECT and the routines which it calls is summarized in Fig. 3.2.

## 3.3. INPUT FILES REQUIRED

Three input files are required: INPUT, INFIL, and *post-process-filename*. File INPUT contains the NAMELIST input data used for program control and the text input and control values for the text frames. File INFIL contains NAMELIST input data needed to read the files to be post-processed. The name of the first member file to be post-processed is in this NAMELIST list. One can thus begin post-processing with any member of a family of files that one desires as long as one does not need some information written only once at the beginning of the first member of the family. The input file *post-process-filename* is the name of the family-member file where post-processing begins.

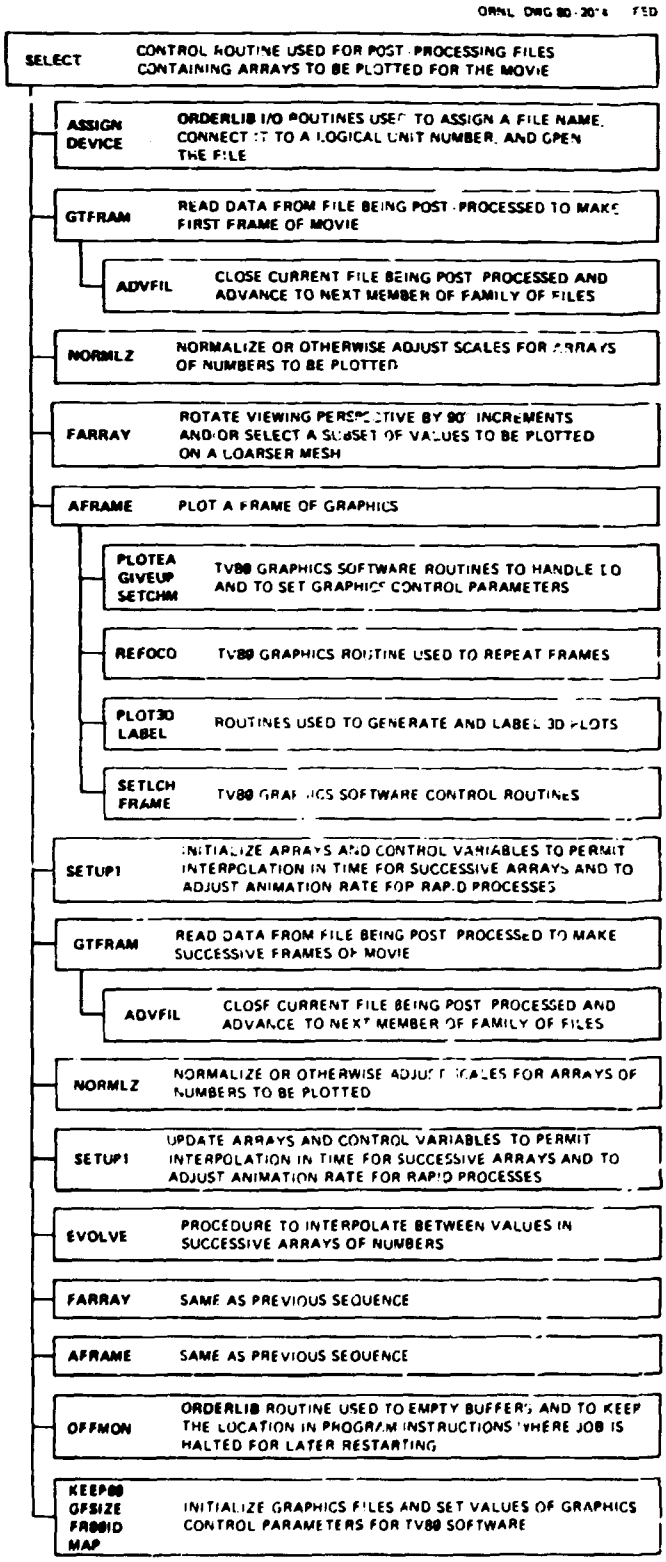


Figure 3.2. Structure of control routine used to post-process files and generate 3D plots.

## 3.4. DESCRIPTION OF INPUT DATA

Description of file INPUT

The NAMELIST FLMPRM contains option and control variables. They are enumerated and described below.

IOPTØ1: Controls type and amount of scaling of array values for plotting

=Ø values are not altered

=1 maximum value of entire set of arrays to be plotted is found and used to normalize all array values. The normalized global maximum can be adjusted with input parameter SP. Arrays are plotted on a linear scale.

=2 function values in each array are normalized by the maximum value of the array which itself is normalized to unity. For semilog display, a lower limit cut-off or floor is assigned by the input parameter SP. This option is useful for displaying changes in spatial distribution and shape of distribution functions but does not show absolute growth and decay since no global maximum value is located.

(default=1)

IOPTØ2: Controls the type of display

=Ø generates linear plots

=1 generates semi-log plots in the vertical direction

(default=Ø)

IOPTØ3: Controls the mode of display for 3D plots with respect to surface rotation

=Ø no rotation

=1 causes surface to rotate between $0^{\circ}$ and $180^{\circ}$. In actual practice, the limits of rotation work better for $10^{\circ} < \theta < 170^{\circ}$.

(default=Ø)

IOPTØ4: Controls suppression or generation of text

=Ø suppress all text frames

=1 generate all text frames

(default=Ø)

IOPTØ5: Controls orientation of surfaces for 3D display by rotating them through different multiples of $90^{\circ}$.

=Ø no rotation

=1 rotate figure $90^{\circ}$

=2 rotate figure $180^{\circ}$

=3 rotate figure $270^{\circ}$

(default=Ø)

MOVIE: Variable used to supply a unique name for the graphics output files. The name must be limited to 5 characters and must be right adjusted in the field (must use hollerith format 5RnameØ)

(default=5RFILMS)

NTFD: The number of times a frame is to be displayed. When NTFD < Ø, the routines which generate text determine the number of times a frame is to be displayed. For the plot frames, though, |NTFD| determines the number of times a frame is to be displayed.

(default=1)

NVA: The number of angles at which a given surface is to be viewed if it is rotated through an angle DPHI. Each successive rotation in the animation sequence is made through an angle DPHI/NVA.

(default=1)

INT: Controls order of interpolation of array values for successive times. This value is stored in array location NA(5).

=Ø no interpolation

=1 linear interpolation, usually the best choice.

>1 successively higher orders of interpolation using a Lagrange algorithm. This is not recommended, but is available.

(default=1)

NS: Determines coarseness of mesh for 3D surface plots. This variable is very useful for generating plots quickly.

=1 use every point in 2D array

=2 use every other point

=3 use every third point

.
.
.

=n use every $n^{th}$ point

(default=1)

IGFS: Specifies graphics output file size in words (decimal)

(default=3ØØØØØ)

SP: Parameter used to scale plots. When IOPTØ1=1 a global maximum value for all frames is found. For this option,

SP may be used to scale plots further relative to this global maximum. The global maximum is reduced by whatever factor is assigned to SP. When IOPTØ1=2, the value of SP determines the lower limit cut-off or floor for plotting distribution functions. If a semi-log display option is used (IOPTØ2=1), the value of SP determines the number of orders of magnitude that are allowed in the range $0 \leq f(x,y) \leq 1$; the maximum value of the distribution function is assumed to be unity.

(default=3Ø)

DPHI: Angle interval (in degrees) through which surface is rotated if IOPTØ3=1.

(default=Ø)

PHI: Viewing angle (in degrees) for displaying a surface.

(default=6Ø.)

TIMSCL: Parameter used to adjust time scale for value of time printed on 3D surface plots.

(default=13.26)

TDEL: Amount by which time variable is incremented for successive frames in a time evolution sequence.

(default=Ø.1)

TTLCFØ: Used for hollerith string comparisons to select arrays of numbers for post-processing into graphics frames.

<default=1ØH F-P F(VX>

Input for the NAMELIST FLMPRM is terminated when the first item not specified in the list in encountered. Following the NAMELIST block of data, one can specify text input to be read and processed by routine

TXTFRM called by the main program. The input format for this text is described in Table 3.1.

Examples of input

Listings of sample input files are given in Appendix A for the files INPUT and INFIL. An excerpt from one of the files that was post-processed by the program is also given to illustrate the use of the format described in Table 3.2. These files are also available, as mentioned in the introduction, from the FILEM file storage system of the NMFECC network.

TABLE 3.1. Input data formats for generating segments of text on film

| Line Number | Columns | Description of variables and arrays | |
|---|---|---|---|
| 1 | | FORMAT (3I2) | |
| | 1-2 | LT | Number of lines of text. If LT=Ø, program execution terminates. |
| | 3-4 | LB | Number of blank lines. Total lines on a frame LINES=LT+LB. Number of blank lines must include number of lines of new text to be added later in delayed sequence to complete the frame. |
| | 5-6 | NT | Number of lines of new text added in delayed sequence. If no new text is to be added to original block, let NT=-1. |
| 2 | | FORMAT (I3) | |
| | 1-3 | NF | Number of frames generated for a frame which must be repeated. If NF > Ø, the value of NF read in determines the number of frames; if NF < Ø, the number of frames is determined by the program. |
| 3 . . . (2+LINES) | | FORMAT (5A10) | |
| | 1-42 | TEXT(I),I=1,5 | Single line of text. After each line is processed, LINES is decremented by one and the next line is processed. After all lines are processed, the program proceeds to the input for the next frame of text. |

Description of file INFIL

The NAMELIST FILPRM contains file attributes and buffering information used for reading files to be post-processed by the movie generator program. They are listed and described below.

JNAME: The (right-adjusted) name of the first member of the family of files to be processed.

IFL: A control parameter used to specify disposition of a disk file to the system I/O routine ASSIGN.

=Ø open the disk file with name given by JNAME (the default value).

=-1 close the disk file with name JNAME.

LBUF: The length of the buffer (default size = 2ØØ4 octal).

LUN: The logical unit number assigned to file named JNAME (default=5).

This input file was made distinct from the file INPUT in order to provide greater flexibility for making several different movies for which the input values in INPUT do not change, but the values in INFIL change for each different simulation case for which a movie is to be made.

Description of files to be post-processed

A detailed description will not be given here since these files will change with each different application. Sufficient description of a typical case will be given to illustrate the most common application of the program so far and to enable a new user to make some runs to familiarize himself with the entire movie-making process. The input format is shown in Table 3.2. The hollerith string assigned to the input parameter TTLCFØ in NAMELIST FLMPRM of file INPUT is used as the

base template in a pattern search to find successive sets of a specified kind of array to be plotted. All arrays in the files to be post-processed must be labelled. These labels are read as the array TITLE and identify the first line of a potential sequence of lines to be read. The string in array element TITLE(1) is matched against that in TTLCFØ to determine whether the data and array which follow are to be read. This process repeats until the files to be post-processed have all been read.

TABLE 3.2. Format of data in files to be post-processed

| Line Number (Relative) | Format | Variable List or Array | Description of Input |
|---|---|---|---|
| 1 | 2A10,A1 | TITLE(I), I=1,3 | A hollerith string used to identify where a particular type of data set begins. |
| 2 | 4I4, 2E20.10 | NA(I), I=1,4 | ND (I=1), total number of values in array to be plotted |
| | | | NN (I=2), the number of nodes in the x-direction |
| | | | NC (I=3), the number of nodes in the y-direction |
| | | | NUMB (I=4), not used. |
| | | PSIMAX | Maximum value of poloidal magnetic flux (not used) |
| 3 | 2F10.0 | SX | Scale (length) for x-direction |
| | | SY | Scale (length) for y-direction |
| 4 | 3X, 7E11.4 | PSAVE(I) I=IL,IU | Array of values to be plotted, read in one line at a time, seven values per line |
| . | | | |
| . | | | |
| . | | | |
| $INT(\frac{ND}{7})+1$ | | | |

# 4. PROGRAM APPLICATIONS

The introduction section mentions several applications briefly but gives no idea of possible limitations or problems which one might encounter. There is almost no indication of procedures one needs to consider to make efficient use of this program and of one's own time. This section provides this information based on experience obtained during the development of the program.

## 4.1. REVIEW OF APPLICATIONS MADE DURING CODE DEVELOPMENT

The original specifications for the program stated that it had to be designed for use in a production environment, that it be easy to use, reliable, and capable of generating a complete movie ready for use in a technical presentation. All this seemed to be reasonable enough for one who had never tried to generate a movie with a computer. Implementing these specifications was not straightforward, however.

The first request for a movie was to generate a sequence of frames showing the evolution of a solution to a Fokker-Planck equation describing neutral beam slowing in a plasma during a computer-simulated discharge. Enough 3D plots of the evolving solution were to be generated to make a film segment with a viewing time of 30-60 seconds. This application very quickly pointed out the need for options to allow for a semi-log scale for displaying distributions which can vary many orders of magnitude over their spatial range and for transforming the array of numbers used to make a 3D plot so that the surface which is shown to the viewer can be rotated to allow surface features of special interest to be seen more easily. Introduction of a semi-log scale option for displaying distribution functions required setting a lower

limit cut-off. All values below the cut-off value are set to this lower limit.

Experience has demonstrated repeatedly that one should allow at least 2-3 weeks for film turn-around time. The fastest turn-around time observed yet has been 4-5 days. One unfortunate aspect of the length of turn-around time is the occasional discovery that 20 or 30 frames of graphics have been superimposed onto a single frame as result of a hardware error. If the graphics files, which may have been perfectly okay, had not been saved, they had to be generated again from the beginning. This kind of situation occurred frequently enough that one clearly needed a way to check graphics files for possible problems before giving them to the system for processing into film. Procedures for checking graphics files are described by Munro [1979] elsewhere.

The program PLOT3D [McCall and Gumm (date unknown)] used to generate the surface plots also was found to have limitations. One can use it to show continuous rotations through an arbitrary angular interval. The useful range is probably $10^{\circ}$ - $80^{\circ}$ if one is to avoid overwriting identifying information in the upper left hand corner of each frame. Otherwise one could use the range $10^{\circ}$ - $170^{\circ}$. Outside this range one sees strange effects. The option which allows one to rotate the viewing perspective by successive $90^{\circ}$ increments together with the continuous rotations available with the PLOT3D program should give one adequate flexibility to see any surface details of interest.

Later requests specified movies for current density and pressure evolution. Since the same family of files contained the sequences of arrays for these two plasma quantities, the capability to allow one to search for a specified label had to be added and provision made to

specify as part of the input the label to be located. Sequence number information was also added to aid in the job of splicing film segments to make the movies. The 3D plots for each frame for the first few sets of current density and pressure movies were all normalized to unity. This allowed one to see the distributions of these quantities evolve spatially, but the maximum value of each distribution on each frame was always one. It was not possible to get an impression of how the quantity changed on an absolute basis during the plasma discharge. To allow for this possibility, the program had to be modified to search all the arrays selected for processing to find the maximum for the entire set, then begin again to generate the graphics using the global maximum to scale each frame.

The global maximum found, however, could not be used with the PLOT3D program without being scaled down by a factor of five or six. Otherwise the entire sequence of graphics showed an evolution which did not remain in a desired proportion, i.e., as the maximum values of the surface for each frame increased, a point would be reached where the base of the plot would suddenly start to recede in order to allow further increases in the maximum values to remain within the plotting window. This may not necessarily be a problem for some applications. Even if it is, it may not always be a serious problem, so one might be able to tolerate it.

An option to permit use of a semi-log scale was not provided with the IOPTØ1=1 choice because most viewers are probably accustomed to watching geometrical surfaces change in time on a linear scale rather than logarithmic. Most viewers would probably not be accustomed to interpreting what they see with their usual intuition if a process were

displayed on a logarithmic scale. One can certainly learn to deal visually with processes displayed on such a scale. If one wants to do this, one needs to rewrite or modify the subroutine NORMLZ.

Another feature introduced later in the program development is the ability to generate text frames of one's own choosing. This allows for flexibility after modifying format statements in subroutines that generate text frames that one wants to remain unchanged from movie to movie. If one wants different text interspersed through a movie, one can generate it all at the place during program execution where that is now permitted, then intersperse it later when editing and splicing the film. This is probably the most efficient procedure to follow.

## 4.2. REVIEW OF PROCEDURES FOUND TO BE USEFUL

Several options and modifications in the program were introduced to make it easier to check graphics files for possible problems before giving them to the system (user-2) to be processed into film and to enable one to diagnose problems easier when they occur. The input parameter NS permits one to use a smaller number of nodes for generating the 3D surface plots. A value NS=3 gives almost an order of magnitude increase in speed for both generating the graphics files and for getting CRT plots from these files. This is a useful feature to use in checking to see whether one made an appropriate choice of viewing perspective and angle, proper scale, or type of plot. One can more easily check through a family of graphics files on a graphics terminal to be sure that blank frames appear only where they are wanted and skip through the graphics files (using utility routine FR8ØPLOT) to be sure all the frames were generated which one expects to find. Such

a check is always made following modifications of the program or to determine whether the files for post-processing appear to have the kinds of results which are worth making into a movie. Figures 4.1 and 4.2 show the results one gets when NS=3 and 1, respectively.

The parameter IGFS is used to specify the size of the members of a family of graphics files. The ability to specify its value is especially useful if one wants to obtain a sequence of plots from a Versatec printer/plotter, but desires to limit the number of frames contained in a member file so that processing of the file is not delayed by an output queuing algorithm which assigns the output priority on the basis of file size. For example, a value of IGFS=20000 ensures that a graphics file is less than 200 blocks in length of PDP-10 disk space. If one uses a small graphics file size such as this for post-processing an entire family of input files, one will get a large number of files, but one then can get Versatec output for very selected segments of the entire output sequence. This may be highly desired on occasions when one wants to examine details of some process which happens on a fast time scale in comparison to the rest of what happens in the output sequence.

The parameter values NS=3 and IGFS=20000 are almost always used now when beginning a new movie making project. The graphics output is then examined at a graphics terminal using the utility routine FR80PLOT and individual frames selected for plotting on the Versatec printer/plotter (using the USC option) or else particular members of the family of graphics files are selected to be plotted. If everything is determined to be in order, then the values of NS and IGFS are returned to their defaults and the program is run to generate a family

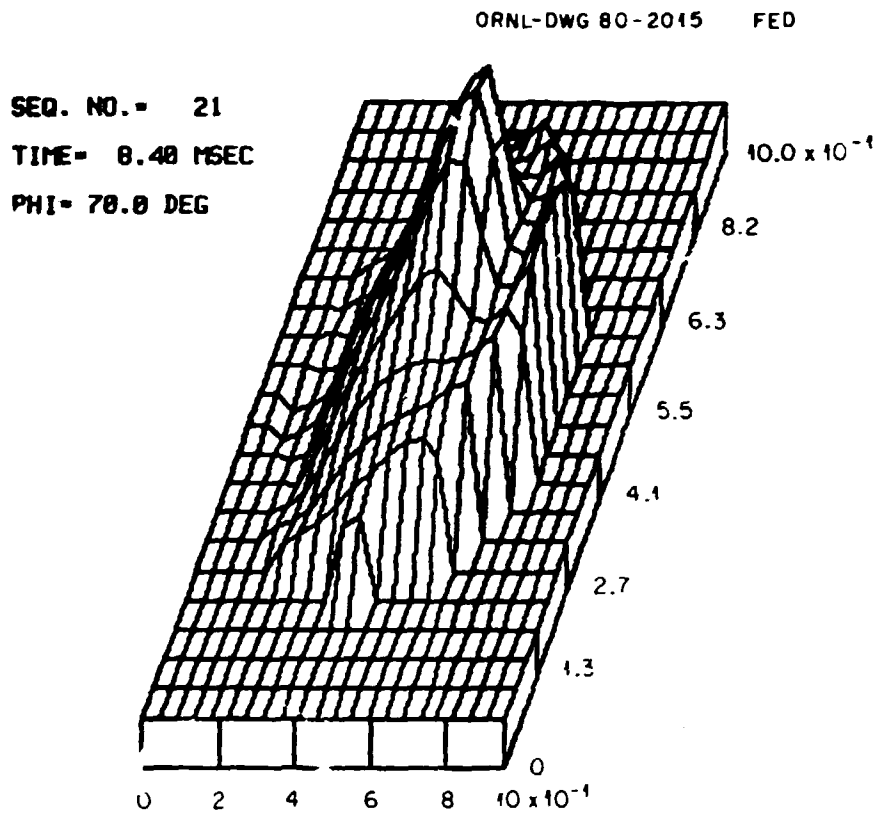


Figure 4.1. Plot of a function using the parameter value NS=3.

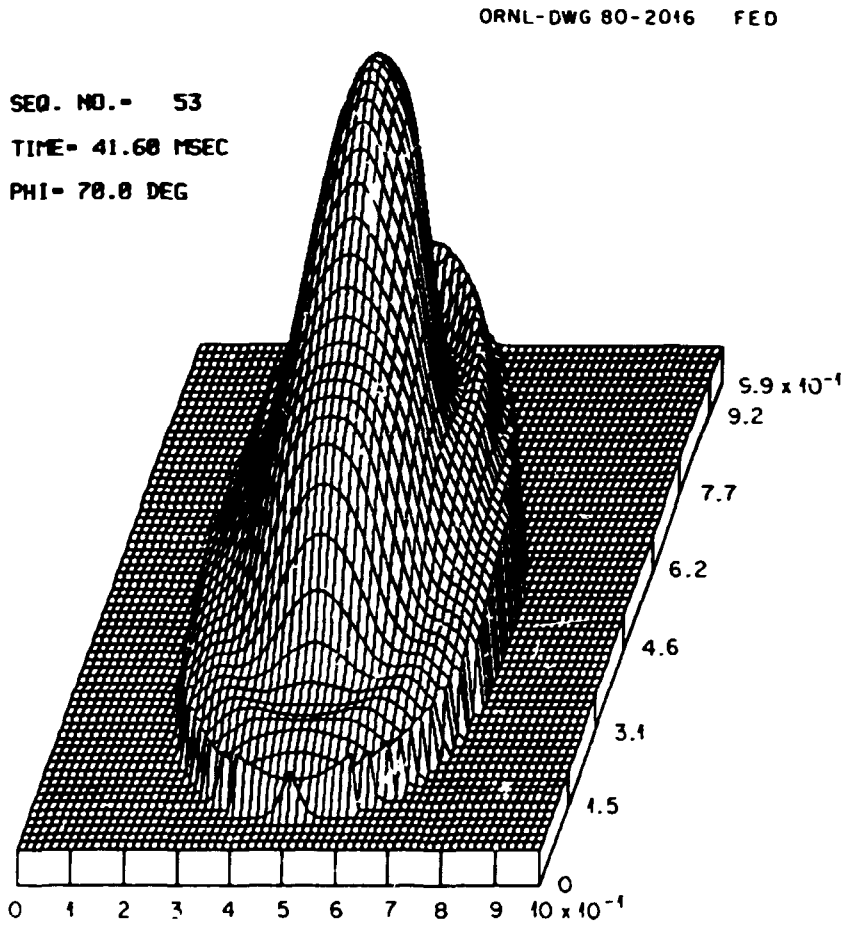


Figure 4.2. Plot of the same function shown in previous figure (but at a different time) using the parameter value NS=1.

of files for processing into film. This family, too, is inspected at a graphics terminal by looking at frames at the beginning and end of each member file and perhaps one frame in the middle to get an impression of the results one should get from each member file. It is worth keeping a record of such a check for later reference in case problems occur in the processing of the files into film or in case something should happen to the film in the delivery or editing process. Problems do occur.

Finally, after the graphics files have been given to the system for processing, one should notify someone at the NMFECC (currently Max Allison) in the graphics group that files have been given to the system, the names of the files, when they were given, and the amount of film (number of feet, based on 40 frames per foot one expects to receive. The service one gets in return for going to the trouble to supply this information is usually well worth it.

## 4.3. COMMENTS ON LIMITATIONS AND PROBLEMS WITH THE PROGRAM

There is a potential problem with the use of the REFOCO routine in subroutine AFRAME to generate the 3D plots. If a tape reaches its capacity in the middle of a frame of graphics so that the remainder of the frame must be continued on a second tape, one loses all the information on the second and succeeding tapes. This is a fault of the REFOCO routine and other TV80 software. If this is a problem, one can either choose to count the number of frames and suspend generation of graphics before the family of graphics files exceeds the tape capacity, then continue with a new family of files; or one can replace the statements as follows

| Current Statement | Replacement Statement |
|---|---|
| CALL REFOCO(NTFD) | DO 4Ø INF=1, NTFD |
| . | . |
| . | . |
| . | . |
| CALL REFOCO(Ø) | 4Ø CONTINUE |

The first alternative will require more work to add the necessary logic and control, but will generate the same amount of output as the current program does. The program will also execute faster if frames are repeated. If frames are not repeated, the second alternative will probably work better. It is certainly easier to implement. However, if one then decides to repeat frames, execution time and number of graphics files generated will increase in proportion to the number of repetitions specified.

The interpolation procedure is limited to cases where the function to be plotted is specified at the same spatial location at each time value and where the number of function values does not change for successive times. This limitation was a problem for one application where field line contours at successive times were given. These contours changed shape and moved around in space at successive times. As they changed, the number of points used to specify them changed. A completely different program had to be written to generate a movie for this application.

## REFERENCES

Carpenter, E. and Solbeck, S., Livermore Time Sharing System, Part IV: Library Files, Chapter 304: ORDERLIB Subroutine Library, Lawrence Livermore Laboratory Document M-026. Edition 2 (1976).

Hogan, J. T., "Multifluid Tokamak Transport Models," in Methods of Computational Physics, V. 16: Applications to Controlled Fusion Research, edited by J. Killeen, Academic Press, New York, 1976.

Hogan, J. T., "The Accessibility of High-Beta Tokamak States," Nuclear Fusion 19, 753 (1979).

McCall, J. R. and Gumm, Barbara, "PLOT3D: A FORTRAN Subroutine for Three-Dimensional Plotting," report obtained through Graphics Coordinator, National Magnetic Fusion Energy Computing Center, Lawrence Livermore Laboratory, Livermore, CA 94550

Munro, J. K., Hogan, J. T., Howe, H. C., and Arnurius, D. E., A User's Manual for the Oak Ridge Tokamak Transport Code, ORNL/TM-5262, Oak Ridge, Tennessee (February 1977).

Munro, J. K., A Guide to Making Time-Lapse Graphics Using the Facilities of the National Magnetic Fusion Energy Computing Center, ORNL/CSD-54, Oak Ridge, Tennessee (to be published).

NMFECC On-line Documentation [Documents continuously undergo additions and revisions. Current documentation is available to network users through the DOCUMENT program.] Documentation Group, National Magnetic Fusion Energy Computing Center, Lawrence Livermore Laboratory, Livermore, CA 94550

## APPENDIX A. Listings of Input Files

The input files for the program HLYWD are listed below with some commentary to assist the beginning user of the program.

File INPUT [=HLYWDIN1 from FILEM directory 25530 .PUBLIC]

```
 1    IOPT01=1 IOPT02=0 IOPT03=0 IOPT04=0 IOPT05=3
 2    MOVIE=10H  BENCHECK
 3    NTFD=-4
 4    NVA=1
 5    INT=0
 6    NS=1
 7    IGFS=300000
 8     SP=5.0
 9    DPHI=0. PHI=70.
10    TIMSCL=.4 TDEL=1.0
11    TTLCF0=10H CURRENT C
12    ENDFLMPRM
13    4 0 0
14    0 0 0
15     0 FRAMES
16    0 0 0
17     0 FRAMES
18    -2 FRAMES
19     CURRENT DENSITY EVOLUTION FOR SEQUENCE
20           OHMIC HEATING,
21           NEUTRAL BEAM HEATING, THEN
22           PELLET INJECTION
23    0 0 0
24     0 FRAMES
25    2 0 0
26    -2 FRAMES
27    EFFECT OF PELLET INJECTION IN SIMULATION
28          SHOWS UP QUITE DRAMATICALLY
29    0 0 0
30     0 FRAMES
```

The NAMELIST data set input is terminated by the string ENDFLMPRM on line 12. The text information which follows is ignored in this case because the number of lines specified is null.

File INFIL [=HLYWDIN2 from FILEM directory 25530 .PUBLIC]

```
 1    JNAME=10H  HLYWDIN3
 2     IFL=0
 3    ENDFILPRM
```

<u>File family beginning with HLYWDIN3, HLYWDIN4, etc.</u> [from directory 25530 .PUBLIC]

Only a partial file listing is given here sufficient to show how the format statements are used in routine GTFRAM to read the arrays to be plotted (line numbers given here are relative; a new set of data begins on line 9 with the title for that set). Line 1 shown here is actually line 600 in the file HLYWDIN3.

```
 1       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
 2       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
 3       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
 4       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
 5       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
 6       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
 7       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
 8       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
 9    CURRENT CONTOURS1
10   4225  65  55  22    1.0000000000E+00
11           7.        4.
12       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
13       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
14       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
15       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
16       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
17       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
18       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
19       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
20       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
21       0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01 0.0000E-01
```

## APPENDIX B. Program Source Listing

An effort has been made to provide a reasonable amount of internal documentation in the source listings so that between these listings and the user's guide, one should be able to use or adapt the program for one's needs with a minimum of difficulty.

### Source file HLYWD0

This file contains the main routine, routines used to read arrays to be plotted, and routines used to interpolate between values in successive arrays. It also contains the main subroutines [SELECT] which contains the logic and control for reading and plotting the arrays from the files which are post-processed.

```
      PROGRAM HLYWD (INPUT,TAPE2=INPUT,INFIL,TAPE1=INFIL)
      PARAMETER (MXRA=4225, MXNTRP=3, NADIM=10)
C  ------------------------------------------------------------------
C  AUTHOR:    J K MUNRO JR
C             COMPUTER SCIENCES DIVISION
C             UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C             OAK RIDGE, TN  37830
C  PURPOSE...
C        THIS IS THE MAIN ROUTINE OF A PROGRAM FOR GENERATING COMPUTER
C        MOVIES OF THE FUNCTIONS GENERATED BY LARGE SCALE (AND CODE)
C        COMPUTER SIMULATIONS OF INITIAL VALUE PROBLEMS.
C  ------------------------------------------------------------------
C  VERSION...
C        THIS VERSION USES FR80 CALLS
C        DATE (YR MO DA HR MN) = 7901291430
C  ==================================================================
C  OPTIONS AND SWITCH SETTINGS...
C  ------------------------------------------------------------------
C        IOPT01: NORMALIZATION AND SCALING OF PLOTS
C                =0 UNNORMALIZED
C                =1 NORMALIZED TO 1.0 FOR THE MAXIMUM VALUE OF THE FUNCTION
C                =2 SCALED
C        IOPT02: TYPE OF PLOT
C                =0 LINEAR
C                =1 SEMILOG
C        IOPT03: SWITCH TO CAUSE 3D PLOT TO ROTATE, 0 .LE. THETA .LE. 180
C                =0 NO ROTATION
C                =1 ROTATION
C        IOPT04: SWITCH TO SUPPRESS OR GENERATE TEXT HEADE'
```

```
C                =0 SUPPRESS TEXT
C                =1 GENERATE TEXT
C        IOPT05: SWITCH TO SELECT BASIC FIGURE ORIENTATION (FOR 3D PLOTS)
C                =0 NO ROTATION
C                =1 ROTATE FIGURE 90 DEGREES
C                =2 ROTATE FIGURE 180 DEGREES
C                =3 ROTATE FIGURE 270 DEGREES
C  ==================================================================
C  OTHER CONTROL VARIABLES...
C  ---------------------------------------------------------------------
C       NTFD: NUMBER OF TIMES A FRAME IS TO BE DISPLAYED.
C             WHEN THIS NUMBER IS < 0, PROGRAM DETERMINES ITS VALUE
C             FOR EACH DIFFERENT SET OF TEXT FOR TEXT FRAMES.
C        NVA: NUMBER OF ANGLES AT WHICH A GIVEN FIGURE IS TO BE VIEWED
C        INT: ORDER OF INTERPOLATION, VALUE STORED IN NA(5)
C         NS: DETERMINES NUMBER OF DATA POINTS USED FOR 3-D PLOTS
C             =1 EVERY POINT IS USED
C             =2 EVERY OTHER POINT IS USED
C             =3 EVERY THIRD POINT IS USED, ETC.
C         SP: SCALE FACTOR FOR NORMALIZING PLOTS WHEN IOPT01=2
C       DPHI: ANGLE INTERVAL (DEG) THROUGH WHICH SURFACE IS ROTATED (IOPT03=1)
C        PHI: VIEWING ANGLE (DEG) USED TO DISPLAY A SURFACE
C     TIMSCL: TIME SCALE ADJUSTMENT PARAMETER TO SCALE TIME VALUE IN PLOT
C       TDEL: INCREMENT FOR TIME VARIABLE FOR SUCCESSIVE PLOTS IN A TIME
C             EVOLUTION SEQUENCE; DETERMINES NUMBER OF INTERPOLATED FRAMES
C  ---------------------------------------------------------------
      LCM COMBUF
      DIMENSION PSAVE(MXRA),TIMRA(5)
      DIMENSION NA(10)
      COMMON /COMBUF/PDBUF(2004B)
      COMMON /CONTROL/IGFS,IFL,LBUF,LUN,JNAME
      COMMON /GOBCOM/IG(38)
      COMMON /SEARCH/TTLCFO
      COMMON /SIZE/SP,SX,SY
      NAMELIST /FLMPRM/ IOPT01,IOPT02,IOPT03,IOPT04,IOPT05,MOVIE,
     . NTFD,NVA,INT,NS,IGFS,SP,DPHI,PHI,TIMSCL,TDEL,TTLCFO
      NAMELIST /FILPRM/JNAME,IFL,LBUF,LUN
      DATA TIMRA/5*0./
      DATA IOPT01/1/,IOPT02/1/,IOPT03/0/,IOPT04/0/,IOPT05/0/
      DATA MOVIE/5RFILMS/,TTLCFO/10H  F-P F(VX/,JNAME/10H      OPLT/
      DATA NTFD/1/,NVA/1/,INT/1/,NS/1/,IGFS/300000/,
     . SP/30./,DPHI/0./,PHI/60./,TIMSCL/13.26/,TDEL/0.1/
      DATA IFL/0/,LBUF/2004B/,LUN/5/
      CALL CHANGE("+GOHCP")
      CALL KEEP80(1,3)
      READ(2,FLMPRM)
      WRITE(59,1000)IOPT01,IOPT02,IOPT03,IOPT04,IOPT05,
     . MOVIE,TTLCFO,
     . NTFD,NVA,INT,NS,IGFS,
     . SP,DPHI,PHI,TIMSCL,TDEL
 1000 FORMAT(1H ,"IOPT01=",I3,"  IOPT02=",I3,"  IOPT03=",I3,
     . "  IOPT04=",I3,"  IOPT05=",I3/
     . 1X,"MOVIE=",A10,"  TTLCFO=",A10/
     . 1X,"NTFD=",I5,"  NVA=",I5,"  INT=",I5,"  NS=",I5,"  IGFS=",I8/
     . 1X,"SP=",E13.6,"  DPHI=",E13.6,"  PHI=",E13.6,"  TIMSCL=",
```

```
     . E13.6,"  TDEL=",E13.6)
      IG(36)=10HBOX A55  J
      IG(37)=10HOHN K. MUN
      IG(38)=10HRO, JR.
      CALL GFSIZE(3,IGFS)
      CALL FR80ID(10H ORNL FED ,1,3)
      CALL KEEP80(MOVIE,3)
C  ------------------------------------------------------------------
C  SUPPRESS HEADER AND INTRODUCTORY TEXT FRAMES (=0) OR PRINT THEM (=1)
C  ------------------------------------------------------------------
      IF(IOPT04 .EQ. 0)  GO TO 5
      CALL MAP(1.,1024.,1.,1024.)
C  ------------------------------------------
C  GENERATE 32CM OF FILM LEADER WITH BOX AND ID
C  ------------------------------------------
      LSEC=1
      CALL LEADER(LSEC)
C  --------------
C  GENERATE TITLE
C  --------------
      CALL TXTFRM
C  ------------------------------------------------------
C  IDENTIFY COMPUTER CODE USED TO GENERATE DATA FOR MOVIE
C  ------------------------------------------------------
      CALL CODGEN(NTFD)
C  ---------------------------------------
C  GIVE CREDIT TO SIMULATION CODE DEVELOPER
C  ---------------------------------------
      CALL CDVLPR(NTFD)
C  ----------------------------
C  GIVE CREDIT TO MOVIE PRODUCER
C  ----------------------------
      CALL FILMKR(NTFD)
C  --------------------------
C  INSERT DESCRIPTION OF MOVIE
C  --------------------------
      CALL TXTFRM
    5 CONTINUE
      NA(5)=INT
      NA(6)=NS
      TIMRA(3)=TDEL
      CALL MAP(1.,1024.,1.,1024.,0.15,0.86,0.285,1.)
      NFR=IABS(NTFD)-1
C  ---------------------------------
C  GENERATE GRAPHICS SEGMENT OF MOVIE
C  ---------------------------------
      INPUT DATA FILPRM, 1, 59
      IF(IOPT01 .EQ. 1)  CALL GLBLMX(PSAVE,NA)
      CALL SELECT(TIMRA,NA,NFR,NVA,IOPT01,IOPT02,IOPT03,IOPT05,PHI,
     . DPHI,TDEL,TIMSCL)
      IF(IOPT04 .EQ. 0)  GO TO 10
C  ------------------------------------------------------
C  GENERATE TEXT GIVING CREDIT TO AGENCY SPONSORING WORK
C  ------------------------------------------------------
      CALL WRKSPN(NTFD)
```

```
      CALL LEADER(LSEC)
   10 CALL PLOTE
      CALL QUIT
      END
C-----------------------------------------------------------------------
      SUBROUTINE GLBLMX(PSAVE,NA)
      LCM COMBUF
      DIMENSION PSAVE(1),NA(1),IERRA(10)
      COMMON /COMBUF/ PDBUF(2004B)
      COMMON /CONTROL/IGFS,IFL,LBUF,LUN,JNAME
      COMMON /SIZE/SP,SX,SY
      DATA SPG/0./,LEN/0/,IERRA/10*0/
      KNAME=JNAME
      CALL ASSIGN(LUN,0,KNAME,IFL,PDBUF,LBUF)
      CALL DEVICE(4HOPEN,KNAME,LEN,IERR,IOC,LEV)
    1 CALL GTFRAM(KNAME,PSAVE,TD,NA,IERRA)
      IF(IERRA(1) .EQ. 1)  GO TO 40
      ND=NA(1)
      SPX=PSAVE(1)
      DO 10  IS=1,ND
   10 IF(PSAVE(IS) .GT. SPX)  SPX=PSAVE(IS)
      IF(SPX .GT. SPG)  SPG=SPX
      IF(IERRA(2) .NE. 1)  GO TO 1
      SP=SPG/SP
   40 WRITE(59,"(""END SEARCH FOR GLOBAL MAXIMUM: SP =="",E12.5)") SP
      RETURN
      END
C-----------------------------------------------------------------------
      SUBROUTINE GTFRAM(NAME,PSAVE,TD,NA,IERRA)
      PARAMETER (MXRA=4225, MXNTRP=3, NADIM=10)
C  ---------------------------------------------------------------
C  AUTHOR:   J K MUNRO JR
C            COMPUTER SCIENCES DIVISION
C            UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C            OAK RIDGE, TN  37830
C  PURPOSE...
C         ROUTINE READS DATA FOR A FRAME OF GRAPHICS ONE LINE AT
C         A TIME, ALWAYS CHECKING FOR END OF FILE SINCE INPUT
C         DATA MAY BE SPLIT ACROSS MANY MEMBERS OF A FAMILY OF FILES
C  ---------------------------------------------------------------
      DIMENSION TITLE(8),PSAVE(MXRA)
      DIMENSION NA(NADIM),IERRA(1)
      COMMON /SEARCH/TTLCFO
      COMMON /SIZE/SP,SX,SY
      DATA LUN/5/
      DATA TTLEOF/"ENDOFILE"/,TITLE/8*10HINITIAL
     .IZE/
C  ----------------------------
C  SEARCH FOR DATA SET IDENTIFIER
C  ----------------------------
   10 READ(5,200) (TITLE(I), I=1,3)
  200 FORMAT(2A10,A1)
      IF(EOF,5) 12,14
   12 CALL ADVFIL(LUN,NAME,LEN,IOC,IERR)
      IF(IERR .EQ. 1)  GO TO 60
```

```
      GO TO 10
   14 IF(TITLE(1) .EQ. TTLEOF) GO TO 499
      IF(TITLE(1) .EQ. TTLCFO)  GO TO 20
      GO TO 10
   20 CONTINUE
C  --------  ---------------------------------------------
C  READ ARRAY DIMENSIONS AND INDICES (NA: ND,NN,NC,NUMB,...)
C  ---------------------------------------------------------
   30 READ(5,210) (NA(I), I=1,4),PSIMAX
  210 FORMAT(4I4,2E20.10)
      IF(EOF,5) 32,34
   32 CALL ADVFIL(LUN,NAME,LEN,IOC,IERR)
      IF(IERR .EQ. 1)  GO TO 60
      GO TO 30
   34 ND=NA(1)
      IF(ND .LE. 0) GO TO 30
   35 READ(5,220) SX,SY
  220 FORMAT(2F10.0)
      IF(EOF,5) 36,38
   36 CALL ADVFIL(LUN,NAME,LEN,IOC,IERR)
      IF(IERR .EQ. 1)  GO TO 60
      GO TO 35
C  ---------------------------------------------
C  READ ARRAY TO BE PLOTTED, ONE LINE AT A TIME
C  ---------------------------------------------
   38 IL=1
      IU=7
   40 READ(5,230) (PSAVE(I), I=IL,IU)
  230 FORMAT(3X,7E11.4)
      IF(EOF,5) 42,44
   42 CALL ADVFIL(LUN,NAME,LEN,IOC,IERR)
      IF(IERR .EQ. 1)  GO TO 60
      GO TO 40
   44 IL=IU+1
      IU=IU+7
      IF(IL .GT. ND)  GO TO 50
      IF(IU .GT. ND)  IU=ND
      GO TO 40
   50 TD=TD+1.
      RETURN
  499 IERRA(1)=1
      RETURN
   60 IERRA(2)=IERR
      IERR=0
      RETURN
      END
C-----------------------------------------------------------------------
      SUBROUTINE NORMLZ(PSAVE,NA,IOPT01,IOPT02)
C  ---------------------------------------------------
C  AUTHOR:    J K MUNRO JR
C             COMPUTER SCIENCES DIVISION
C             UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C             OAK RIDGE, TN  37830
C  PURPOSE...
C          ROUTINE NORMALIZES AND SCALES DATA FOR PLOTTING
```

```
C  ------------------------------------------------------
      DIMENSION PSAVE(1),NA(1)
      COMMON /SIZE/SP,SX,SY
      ND=NA(1)
      IF(IOPT01 .GE. 2)  GO TO 1
      FMAX=SP
      GO TO 15
    1 FLOOR=SP-1.
      FLIMIT=10.0**(-IFIX(FLOOR))
      FMAX=PSAVE(1)
      DO 10 IFNC=2,ND
      IF(PSAVE(IFNC).LT.FLIMIT) PSAVE(IFNC)=FLIMIT
      IF(IOPT02.EQ.0) GO TO 10
      PSAVE(IFNC)=ALOG(PSAVE(IFNC))+FLOOR
   10 IF(PSAVE(IFNC) .GT. FMAX)  FMAX=PSAVE(IFNC)
   15 FMAXR=1./FMAX
      DO 20 IFNC=1,ND
   20 PSAVE(IFNC)=FMAXR*PSAVE(IFNC)
   30 CONTINUE
      RETURN
      END
C-----------------------------------------------------------------
      SUBROUTINE LEADER(NSEC)
      CALL PLOTEA
      CALL GIVEUP(0)
      CALL SETCHM(1)
      DO 10  NLV=1,NSEC
      CALL REFOCO(20)
      CALL SETLCH(300.,450.,0,0,3)
      CALL CRTBCD("BOX A55")
      CALL FRAME
      CALL REFOCO(0)
   10 CONTINUE
      RETURN
      END
C-----------------------------------------------------------------
      SUBROUTINE SELECT(TIMRA,NA,NFR,NVA,IOPT01,IOPT02,IOPT03,
     . IOPT05,PHI,DPHI,TLSL,TIMSCL)
      PARAMETER (MXRA=4225, MXNTRP=3, NADIM=10)
C  ------------------------------------------------------------
C  AUTHOR:   J K MUNRO JR
C            COMPUTER SCIENCES DIVISION
C            UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C            OAK RIDGE, TN  37830
C  PURPOSE...
C         ROUTINE SELECTS THE GRAPHICS SOFTWARE PACKAGE AND LOGIC TO
C         BE USED FOR GENERATING A PARTICULAR MOVIE
C  ------------------------------------------------------------
      LCM COMBUF
      LOGICAL FTO
      DIMENSION PSAVE(MXRA),PS(MXNTRP,MXRA),TIMRA(1),NA(NADIM),IERRA(10)
      DIMENSION TS(MXNTRP),FS(MXNTRP),PSM(MXRA),F(MXRA)
      COMMON /COMBUF/ PDBUF(2004B)
      COMMON /CONTROL/IGFS,IFL,LBUF,LUN,JNAME
      COMMON /GOBCOM/IG(38)
```

```
      COMMON /SIZE/SP,SX,SY
      DATA TS/MXNTRP*0./,FS/MXNTRP*0./,FTO/.TRUE./,IERRA/10*0/
      DATA LEN/0/
      IF(.NOT.FTO)  GO TO 2
      IERRA(2)=0
      DO 1  IN=1,MXRA
      PSAVE(IN)=0.
      PSM(IN)=0.
      F(IN)=0.
      DO 1  IM=1,MXNTRP
      PS(IM,IN)=0.
    1 CONTINUE
    2 INT=NA(5)
      KNAME=JNAME
      CALL ASSIGN(LUN,0,KNAME)
      CALL DEVICE(4HOPEN,KNAME,LEN,IERR,IOC,LEV)
      IF(IERR .NE. 0  .OR.  IERR .NE. 2) WRITE(59,201)  IERR,KNAME
  201 FORMAT(1H ,"I/O ERROR",I3," UPON ATTEMPT TO OPEN",A10,"  IN
     . SUBROUTINE SELECT")
C     IF(MOVIE .EQ. 0)  GO TO 500
      SIGN=1.
C  ------------------------------
C  GET DATA FOR A FRAME OF GRAPHICS
C  ------------------------------
   10 CALL GTFRAM(KNAME,PSAVE,TD,NA,IERRA)
      IF(IERRA(1) .EQ. 1  .OR.  IERRA(2) .EQ. 1)  GO TO 40
C  ----------------------
C  NORMALIZE OR SCALE DATA
C  ----------------------
      IF(IOPT01 .GT. 0)  CALL NORMLZ(PSAVE,NA,IOPT01,IOPT02)
      TC=TD
C  ------------------------------------------------------------
C  ROTATE FUNCTION SURFACE AND/OR CHANGE NUMBER OF GRID POINTS
C  ------------------------------------------------------------
      CALL FARRAY(IOPT05,PSAVE,NA,F)
      TIME=TC*TIMSCL
C  ----------------------------------------------------------------
C  PLOT A FRAME OF GRAPHICS, A SURFACE (FUNCTION OF TWO VARIABLES)
C  ----------------------------------------------------------------
   35 CALL AFRAME(F,NA,NFR,NVA,IOPT03,PHI,DPHI,SIGN,TIME)
      SIGN=-SIGN
      IF(INT .EQ. 0)  GO TO 10
      IS=0
C  ---------------------------------------------------
C  INITIAL ARRAYS TO PERMIT INTERPOLATION IN TIME VARIABLE
C  ---------------------------------------------------
      CALL SETUP1(PSAVE,TD,PS,TS,FS,NA,IS)
      IS=1
C  ----------------------------------------
C  GET DATA FOR SUCCESSIVE FRAMES OF GRAPHICS
C  ----------------------------------------
  200 CALL GTFRAM(KNAME,PSAVE,TD,NA,IERRA)
      IF(IERRA(1) .EQ. 1  .OR.  IERRA(2) .EQ. 1)  GO TO 40
C  ----------------------
C  NORMALIZE OR SCALE DATA
```

```
C  ----------------------
      IF(IOPT01 .GT. 0)  CALL NORMLZ(PSAVE,NA,IOPT01,IOPT02)
C  ---------------------------------------
C  UPDATE ARRAYS TO ALLOW FOR INTERPOLATION
C  ---------------------------------------
      CALL SETUP1(PSAVE,TD,PS,TS,FS,NA,IS)
  210 IF(TC .GT. TD)  GO TO 200
      ND=NA(1)
C  ---------------------------------------------------------
C  INTERPOLATE BETWEEN SUCCESSIVE ARRAYS OF FUNCTION SURFACES
C  ---------------------------------------------------------
      CALL EVOLVE(PS,TS,FS,PSM,TC,ND,INT)
C  ---------------------------------------------------------
C  ROTATE FUNCTION SURFACE AND/OR CHANGE NUMBER OF GRID POINTS
C  ---------------------------------------------------------
      CALL FARRAY(IOPT05,PSM,NA,F)
      TDEL=TIMRA(3)
      TC=TC+TDEL
      TIME=TC*TIMSCL
C  ---------------------------------------------------------------
C  PLOT A FRAME OF GRAPHICS, A SURFACE (FUNCTION OF TWO VARIABLES)
C  ---------------------------------------------------------------
      CALL AFRAME(F,NA,NFR,NVA,IOPT03,PHI,DPHI,SIGN,TIME)
      SIGN=-SIGN
      TC=TC+TDEL
      IF(SENSESWITCH,1) 30, 20
   30 CALL OFFMON
      MOVIE=NA(10)
      CALL KEEP80(1,3)
      IG(36)=10HBOX A55  J
      IG(37)=10HOHN K. MUN
      IG(38)=10HRO, JR.
      CALL GFSIZE(3,IGFS)
      CALL FR80ID(10H ORNL FED ,1,3)
      CALL KEEP80(MOVIE,3)
      CALL MAP(1.,1024.,1.,1024.,0.15,0.86,0.285,1.)
   20 GO TO 210
   40 WRITE(59,"(""END OF INPUT DATA"")")
      RETURN
      END
C-----------------------------------------------------------------
      SUBROUTINE SETUP1(PSAVE,TD,PS,TS,FS,NA,IS)
      PARAMETER (MXRA=4225, MXNTRP=3, NADIM=10)
C  ---------------------------------------------------------
C  AUTHOR:   J K MUNRO JR
C            COMPUTER SCIENCES DIVISION
C            UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C            OAK RIDGE, TN  37830
C  PURPOSE...
C         SUBROUTINE SETS UP AND UPDATES ARRAYS FOR USE WITH
C         INTERPOLATION PROCEDURE IN ROUTINE EVOLVE
C  ---------------------------------------------------------
      DIMENSION PSAVE(1),NA(NADIM),PS(MXNTRP,MXRA),TS(MXNTRP),FS(MXNTRP)
      ND=NA(1)
      NN=NA(5)
```

```
      IF(IS .EQ. 0  .OR.  NN .LT. 1)  GO TO 10
      DO 30  IORD=1,NN
   30 TS(IORD)=TS(IORD+1)
      TS(NN+1)=TD
      DO 40  IRA=1,ND
          DO 50  IORD=1,NN
          FS(IORD)=1./(TS(IORD+1) - TS(IORD))
   50     PS(IORD,IRA)=PS(IORD+1,IRA)
   40 PS(NN+1,IRA)=PSAVE(IRA)
      RETURN
   10 TS(2)=TD
      DO 20  IRA=1,ND
   20 PS(2,IRA)=PSAVE(IRA)
      RETURN
      END
C-----------------------------------------------------------------
      SUBROUTINE EVOLVE(PS,TS,FS,PSM,TC,ND,NN)
      PARAMETER (MXRA=4225, MXNTRP=3, NADIM=10)
C  ---------------------------------------------------------------
C  AUTHOR:   J K MUNRO JR
C            COMPUTER SCIENCES DIVISION
C            UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C            OAK RIDGE, TN  37830
C  PURPOSE...
C         SUBROUTINE INTERPOLATES BETWEEN DATA FOR DIFFERENT TIMES TO
C         PROVIDE FOR ADJUSTING TIME SCALES OF EVOLUTION OF A PROCESS
C  ---------------------------------------------------------------
      DIMENSION PS(MXNTRP,MXRA),TS(MXNTRP),FS(MXNTRP),PSP(MXNTRP),
     . PSM(MXRA)
      DATA IEX/0/
      IF(NN .EQ. 0)  RETURN
      IF(NN .GT. 1)  GO TO 200
      DO 100  IRA=1,ND
      PSM(IRA)=FS(1)*((TS(2)-TC)*PS(1,IRA) + (TC-TS(1))*PS(2,IRA))
  100 CONTINUE
      RETURN
  200 IND1=0
      N1=NN+1
      DO 210  IRA=1,ND
          DO 220  IO=1,N1
  220     PSP(IO)=PS(IO,IRA)
      PSM(IRA)=YLAG(TC,TS,PSP,IND1,N1,N1,IEX)
      IF(IEX) 230,210,250
  230 WRITE(59,"(""EXTRAPOLATION OCCURRED BELOW TABLE FOR TC="",F6.2,
     . "" MSEC"")") TC
      RETURN
  250 WRITE(59,"(""EXTRAPOLATION OCCURRED ABOVE TABLE FOR TC ="",F6.2,
     . "" MSEC"")") TC
  210 CONTINUE
      RETURN
      END
```

Source file HLYWD1

This file contains the routines AFRAME, which generates a frame of graphics with a 3D plot, and FARRAY, which changes the viewing perspective of a 3D plot or changes the coarseness of the mesh used for generating the plots.

```
C-----------------------------------------------------------------------
      SUBROUTINE AFRAME(F,NDEXF,NFR,NVA,IOPT03,PHI,DPHI,SIGN,TIME)
C  ------------------------------------------------
C  AUTHOR:    J K MUNRO JR
C             COMPUTER SCIENCES DIVISION
C             UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C             OAK RIDGE, TN  37830
C  PURPOSE...
C          GENERATE A FRAME OF FR80 GRAPHICS COMMANDS
C  ------------------------------------------------
      DIMENSION F(1),NDEXF(1),LABAXX(2),LABAXY(2)
      DIMENSION ZX(65),ZY(65),LABHEX(10),LABHEY(10)
      COMMON /SIZE/SP,SX,SY
      DATA MFSEQ/0/,LABAXX/"VEL IN"," X-DIR"/,
     . LABAXY/"VEL IN"," Y-DIR"/
      DATA LABHEX/"    NE","UTRAL ","BEAM S","LOWING"," PLOTT",
     . "ED IN ","VELOCI","TY COO","RDINAT","ES    "/
      DATA LABHEY/" SOLUT","ION (D","ISTRIB","UTION ","FUNCTI",
     . "ON) TO"," FOKKE","R-PLAN","CK EQU","ATION "/
C *** NVA = NUMBER OF ANGLES FOR VIEWING A GIVEN FIGURE
      NAC=1
      DELPHI=0.
      MFSEQ=MFSEQ+1
      IF(IOPT03.EQ.0) GO TO 5
      NAC=NVA
      DELPHI=DPHI/FLOAT(NVA)
    5 CONTINUE
C  ZX : AN ARRAY OF VALUES USED TO PARAMETERIZE THE FAMILY OF CURVES
C  ZY : THE INDEPENDENT VARIABLE FOR EACH CURVE
C  SR : RATIO OF THE SIDES OF THE BASE = SY/SX
      SR=SY/SX
      ZXL=1.*SR
      ZYL=1.
      FMIN=F(1)
      FMAX=F(1)
      ND=NDEXF(1)
      DO 7  J=1,ND
      IF(F(J) - FMIN) ,8,8
      FMIN=F(J)
      GO TO 9
    8 IF(F(J) - FMAX)9,9,
      FMAX=F(J)
    9 CONTINUE
```

```
    7 CONTINUE
      ZZL=(FMAX-FMIN)/SP
      NX=NDEXF(7)
      NY=NDEXF(8)
      ZDELX=1./FLOAT(NY)
      ZDELY=1./FLOAT(NX)
      DO 10  INX=1,NY
   10 ZX(INX)=FLOAT(INX-1)*ZDELX
      DO 20  INY=1,NX
   20 ZY(INY)=FLOAT(INY-1)*ZDELY
      CALL PLOTEA
      CALL GIVEUP(0)
      NTFD=NFR+1
      DO 50 INA=1,NAC
      CALL SETCHM(1)
      CALL REFOCO(NTFD)
      CALL PLOT3D(PHI,ZX,ZY,F,ZXL,ZYL,ZZL,NX,NY,NX)
C     CALL LABEL(LABAXX,LABAXY,LABHEX,LABHEY)
      CALL SETLCH(2.,1000.,0,0,2,0)
      WRITE(100,202) MFSEQ
  202 FORMAT(1H ,"SEQ. NO.=",I5)
      CALL SETLCH(2.,950.)
      WRITE(100,200) TIME
  200 FORMAT(1H ,"TIME=",F6.2," MSEC")
      CALL SETLCH(2.,900.)
      WRITE(100,201) PHI
  201 FORMAT(1H ,"PHI=",F5.1," DEG")
      CALL FRAME
      CALL PLOTEA
      CALL GIVEUP(0)
      CALL REFOCO(0)
      PHI=PHI+DELPHI*SIGN
   50 CONTINUE
      RETURN
      END
C-----------------------------------------------------------------------
      SUBROUTINE FARRAY(IOPT05,PSAVE,NDEXF,F)
C  ---------------------------------------------------------------
C  AUTHOR:    J K MUNRO JR
C             COMPUTER SCIENCES DIVISION
C             UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C             OAK RIDGE, TN  37830
C  PURPOSE...
C          ROTATE 3-D PLOTS AND ADJUST COARSENESS OF MESH FOR QUICK
C          PLOTTING AT A GRAPHICS TERMINAL
C  ---------------------------------------------------------------
      DIMENSION PSAVE(1),F(1),NDEXF(1)
      COMMON /SIZE/SP,SX,SY
      NDX=NDEXF(2)
      NDY=NDEXF(3)
      NP=IOPT05+1
      NS=NDEXF(6)
      NSM=NS-1
      NX=(NDX+NSM)/NS
      NY=(NDY+NSM)/NS
```

```
      NDEXF(7)=NX
      NDEXF(8)=NY
      GO TO (100,200,300,400), NP
  100 DO 10  IX=1,NDX,NS
      I=(IX+NSM)/NS
      DO 10  IY=1,NDY,NS
      J=(IY+NSM)/NS
      IDJ=(I-1)*NY+J
      IDJTH=(IX-1)*NDY+IY
      F(IDJ)=PSAVE(IDJTH)
   10 CONTINUE
      RETURN
  200 DO 20  IX=1,NDX,NS
      I=NX-(IX+NSM)/NS+1
      DO 20  IY=1,NDY,NS
      J=(IY+NSM)/NS
      IDJ=(J-1)*NX+I
      IDJTH=(IX-1)*NDY+IY
      F(IDJ)=PSAVE(IDJTH)
   20 CONTINUE
      ZSX=SX
      SX=SY
      SY=ZSX
      RETURN
  300 DO 30  IX=1,NDX,NS
      I=NX-(IX+NSM)/NS+1
      DO 30  IY=1,NDY,NS
      J=NY-(IY+NSM)/NS+1
      IDJTH=(IX-1)*NDY+IY
      IDJ=(I-1)*NY+J
      F(IDJ)=PSAVE(IDJTH)
   30 CONTINUE
      RETURN
  400 DO 40  IX=1,NDX,NS
      I=(IX+NSM)/NS
      DO 40  IY=1,NDY,NS
      J=NY-(IY+NSM)/NS+1
      IDJ=(J-1)*NX+I
      IDJTH=(IX-1)*NDY+IY
      F(IDJ)=PSAVE(IDJTH)
   40 CONTINUE
      ZSX=SX
      SX=SY
      SY=ZSX
      RETURN
      END
```

Source file HLYWD2

This file contains the routines for generating text frames that remain relatively fixed, such as credits to computer program developers, codes used for computation of solutions shown in the sequence of plots, acknowledgement of funding agency, etc.

```
      SUBROUTINE CODGEN(NF)
C  ------------------------------------------------------------
C  AUTHOR:  J K MUNRO JR
C           COMPUTER SCIENCES DIVISION
C           UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C           OAK RIDGE, TN  37830
C  PURPOSE: GENERATE A TEXT FRAME IDENTIFYING COMPUTER CODE(S) USED
C           TO GENERATE THE TABLES OF NUMBERS TO BE PLOTTED FOR THE
C           MOVIE
C  ------------------------------------------------------------
      DATA H/48./,HH/24./,LINES/3/
      CALL PLOTEA
      CALL GIVEUP(0)
      CALL SETCHM(1)
      CALL FRAME(5)
      NFV=NF
      IF(NF .LT. 0)  NFV=40*LINES
      TH=LINES*H + (LINES-1)*HH
      TM=0.5*(1024.-TH)
      ZY=1024.-TM
      NFN=20
      NGRPV=0
      NGRP=NFV/20
      NREM=NFV-NGRP*20
    1 CALL REFOCO(NFN)
C  ------------------------------------------------------------
C  LINE OF TEXT MUST BE CENTERED IN A FIELD 42 CHARACTERS WIDE
C  ------------------------------------------------------------
      CALL SETLCH(0.,ZY,0,0,3)
      CALL CRTBCD("            COMPUTER CODE USED")
      ZY=ZY-2.*H
      CALL SETLCH(0.,ZY)
      CALL CRTBCD("     OAK RIDGE TOKAMAK TRANSPORT CODE")
      CALL FRAME
      CALL REFOCO(0)
      NGRPV=NGRPV+1
      IF(NGRPV .LT. NGRP) GO TO 1
      NFN=NREM
      IF(NREM .GT. 0) GO TO 1
      CALL FRAME(5)
      RETURN
      END
C  ------------------------------------------------------------
```

```
      SUBROUTINE CDVLPR(NF)
C  ------------------------------------------------------------
C  AUTHOR:  J K MUNRO JR
C           COMPUTER SCIENCES DIVISION
C           UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C           OAK RIDGE, TN  37830
C  PURPOSE: GENERATE A TEXT FRAME ACKNOWLEDGING DEVELOPER(S) OF THE
C           COMPUTER CODE(S) USED AS THE BASIS FOR MAKING THE MOVIE
C  ------------------------------------------------------------
      DATA H/48./,HH/24./,LINES/6/,ISW/0/,LAT/3/,NT/2/
      ZSP15=1.5*H
      CALL PLOTEA
      CALL GIVEUP(0)
      CALL SETCHM(1)
      CALL FRAME(5)
C  ---------------------------------------------
C  SET ISW=1 IF ROUTINE MUST COMPUTE VIEWING TIME
C  ---------------------------------------------
      IF(NF .LT. 0)  ISW=1
      TH=LINES*H + (LINES-1)*HH
      TM=0.5*(1024.-TH)
    5 ZY=1024.-TM
      NFV=NF
      IF(ISW .EQ. 1)  NFV=40*NT
      NFN=20
      NGRPV=0
      NGRP=NFV/20
      NREM=NFV-NGRP*20
    1 CALL REFOCO(NFN)
C  ------------------------------------------------------------
C  LINE OF TEXT MUST BE CENTERED IN A FIELD 42 CHARACTERS WIDE
C  ------------------------------------------------------------
      CALL SETLCH(0.,ZY,0,0,3)
      CALL CRTBCD("            CODE DEVELOPED BY")
      ZY=ZY-ZSP15
      CALL SETLCH(0.,ZY)
      CALL CRTBCD("              JOHN T. HOGAN")
      IF(LAT .EQ. 0)  GO TO 10
      CALL FRAME
      CALL REFOCO(0)
      NGRPV=NGRPV+1
      IF(NGRPV .LT. NGRP) GO TO 1
      NFN=NREM
      IF(NREM .GT. 0) GO TO 1
C  ------------------------------
C  CONTINUE TEXT ON DOWN THE FRAME
C  ------------------------------
      NT=LAT
      LAT=0
      GO TO 5
   10 ZY=ZY-ZSP15
      CALL SETLCH(0.,ZY)
      CALL CRTBCD("          FUSION ENERGY DIVISION")
      ZY=ZY-H
      CALL SETLCH(0.,ZY)
```

```
      CALL CRTBCD("      OAK RIDGE NATIONAL LABORATORY")
      ZY=ZY-H
      CALL SETLCH(0.,ZY)
      CALL CRTBCD("           OAK RIDGE, TN  37830")
      CALL FRAME
      CALL REFOCO(0)
      NGRPV=NGRPV+1
      IF(NGRPV .LT. NGRP) GO TO 1
      NFN=NREM
      IF(NREM .GT. 0) GO TO 1
      CALL FRAME(5)
      RETURN
      END
C  ------------------------------------------------------------------
      SUBROUTINE FILMKR(NF)
C  ------------------------------------------------------------------
C  AUTHOR:  J K MUNRO JR
C           COMPUTER SCIENCES DIVISION
C           UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C           OAK RIDGE, TN  37830
C  PURPOSE: GENERATE A TEXT FRAME ACKNOWLEDGING PERSON(S) WHO MADE
C           THE MOVIE.
C  ------------------------------------------------------------------
      DATA H/48./,HH/24./,ISW/0/,LAT/4/,LINES/7/,NT/2/
      ZSP15=1.5*H
      CALL PLOTEA
      CALL GIVEUP(0)
      CALL SETCHM(1)
      CALL FRAME(5)
C  ---------------------------------------------
C  SET ISW=1 IF ROUTINE MUST COMPUTE VIEWING TIME
C  ---------------------------------------------
      IF(NF .LT. 0)  ISW=1
      TH=LINES*H + (LINES-1)*HH
      TM=0.5*(1024.-TH)
    5 ZY=1024.-TM
      NFV=NF
      IF(ISW .EQ. 1)  NFV=40*NT
      NFN=20
      NGRPV=0
      NGRP=NFV/20
      NREM=NFV-NGRP*20
    1 CALL REFOCO(NFN)
C  ------------------------------------------------------------
C  LINE OF TEXT MUST BE CENTERED IN A FIELD 42 CHARACTERS WIDE
C  ------------------------------------------------------------
      CALL SETLCH(0.,ZY,0,0,3)
      CALL CRTBCD("            MOVIE PRODUCED BY")
      ZY=ZY-ZSP15
      CALL SETLCH(0.,ZY)
      CALL CRTBCD("            JOHN K. MUNRO, JR")
      IF(LAT .EQ. 0)  GO TO 10
      CALL FRAME
      CALL REFOCO(0)
      NGRPV=NGRPV+1
```

```
      IF(NGRPV .LT. NGRP) GO TO 1
      NFN=NREM
      IF(NREM .GT. 0) GO TO 1
C  ------------------------------
C  CONTINUE TEXT ON DOWN THE FRAME
C  ------------------------------
      NT=LAT
      LAT=0
      GO TO 5
   10 ZY=ZY-ZSP15
      CALL SETLCH(0.,ZY)
      CALL CRTBCD("        COMPUTER SCIENCES DIVISION")
      ZY=ZY-H
      CALL SETLCH(0.,ZY)
      CALL CRTBCD("  UNION CARBIDE CORP., NUCLEAR DIVISION")
      ZY=ZY-H
      CALL SETLCH(0.,ZY)
      CALL CRTBCD(" LOCATED AT OAK RIDGE NATIONAL LABORATORY")
      ZY=ZY-H
      CALL SETLCH(0.,ZY)
      CALL CRTBCD("            OAK RIDGE, TN  37830")
      CALL FRAME
      CALL REFOCO(0)
      NGRPV=NGRPV+1
      IF(NGRPV .LT. NGRP) GO TO 1
      NFN=NREM
      IF(NREM .GT. 0) GO TO 1
      CALL FRAME(5)
      RETURN
      END
C  ------------------------------------------------------------
      SUBROUTINE WRKSPN(NF)
C  ------------------------------------------------------------
C  AUTHOR:  J K MUNRO JR
C           COMPUTER SCIENCES DIVISION
C           UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C           OAK RIDGE, TN  37830
C  PURPOSE: GENERATE A TEXT FRAME ACKNOWLEDGING AGENCY WHICH SUPPORTED
C           THE WORK AND THE MOVIE MAKING
C  ------------------------------------------------------------------
      DATA H/48./,HH/24./,LINES/5/
      CALL PLOTEA
      CALL GIVEUP(0)
      CALL SETCHM(1)
      CALL FRAME(5)
      NFV=NF
      IF(NF .LT. 0)  NFV=40*LINES
      TH=LINES*H + (LINES-1)*HH
      TM=0.5*(1024.-TH)
      ZY=1024.-TM
      NFN=20
      NGRPV=0
      NGRP=NFV/20
      NREM=NFV-NGRP*20
    1 CALL REFOCO(NFN)
```

```
C  ------------------------------------------------------------
C  LINE OF TEXT MUST BE CENTERED IN A FIELD 42 CHARACTERS WIDE
C  ------------------------------------------------------------
      CALL SETLCH(0.,ZY,0,0,3)
      CALL CRTBCD("       RESEARCH SPONSORED BY THE")
      ZY=ZY-H
      CALL SETLCH(0.,ZY)
      CALL CRTBCD("        OFFICE OF FUSION ENERGY")
      ZY=ZY-H
      CALL SETLCH(0.,ZY)
      CALL CRTBCD("       U. S. DEPARTMENT OF ENERGY")
      ZY=ZY-H
      CALL SETLCH(0.,ZY)
      CALL CRTBCD(" UNDER CONTRACT W-7405-ENG-26 WITH THE")
      ZY=ZY-H
      CALL SETLCH(0.,ZY)
      CALL CRTBCD("       UNION CARBIDE CORPORATION")
      CALL FRAME
      CALL REFOCO(0)
      NGRPV=NGRPV+1
      IF(NGRPV  LT. NGRP) GO TO 1
      NFN=NREM
      IF(NREM .GT. 0) GO TO 1
      CALL FRAME(5)
      RETURN
      END
```

Source file HLYWD3

This file contains the routine ADVFIL used to close and open successive members of a family of files. This file was originally intended to serve as the library of service routines which any movie generator program might need.

```
      SUBROUTINE ADVFIL(LUN,JNAME,LEN,IOC)
C *** VERSION DATE = YR-MO-DA-HR-MN = 7707201015
C  AUTHOR:   JOHN K MUNRO JR
C            COMPUTER SCIENCES DIVISION
C            UNION CARBIDE CORPORATION, NUCLEAR DIVISION
C            OAK RIDGE, TN  37830
C  PURPOSE...
C        ROUTINE CLOSES CURRENT FILE 'JNAME' BEING READ ON LOGICAL UNIT
C        NUMBER 'LUN' AND OPENS THE NEXT DISK FILE BELONGING TO THE FILE
C        FAMILY NAME.
      DATA IERR/0/,LID/0/
      CALL DEVICE(6HCLOSER,JNAME,IERR,IOC)
      IF(IERR .NE. 0)  WRITE(59,200) IERR,JNAME
  200 FORMAT(1H ,"I/O ERROR",I3," UPON ATTEMPT TO CLOSE ",A10)
      JNAME=JNAME+1
      CALL ASSIGN(LUN,LID,JNAME)
```

```
      CALL DEVICE(4HOPEN,JNAME,LEN,IERR,IOC)
      IF(IERR .EQ. 0 .OR. IERR .EQ. 2)  RETURN
      WRITE(59,201)  IERR,JNAME
  201 FORMAT(1H ,"I/O ERROR",I3," UPON ATTEMPT TO OPEN " A10)
      WRITE(59,"(""END OF RUN"")")
      CALL PLOTE
      CALL QUIT
      RETURN
      END
```