DOCUMENT RESUME

ED 197 987 SE 034 162

AUTHOR Gales, Larry
TITLE Programmer's Guide for Subroutine PLOT3D. Physical Processes in Terrestrial and Aquatic Ecosystems, Computer Programs and Graphics Capabilities.
INSTITUTION Washington Univ., Seattle. Center for Quantitative Science in Forestry, Fisheries and Wildlife.
SPONS AGENCY National Science Foundation, Washington, D.C.
PUB DATE May 78
GRANT NSF-GZ-2980; NSF-SED74-17696
NOTE 33p.: For related documents, see SE 034 160-167 and SE 033 581-597. Contains light and broken type in computer print-outs.

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS Biological Sciences; *College Science; Computer Assisted Instruction; *Computer Graphics; *Computer Programs; Display Systems; Ecology; Higher Education; Interdisciplinary Approach; Mathematical Applications; Physics; *Programing; Science Education

ABSTRACT

This module is part of a series designed to be used by life science students for instruction in the application of physical theory to ecosystem operation. Most modules contain computer programs which are built around a particular application of a physical process. PLOT3D is a subroutine package which generates a variety of three-dimensional hidden line Cal Comp-type displays. The package features multiple plots per page, with or without annotation, or annotation without plots; multiple pages; automatic scaling; flexible tilting; two-dimensional interpolation over the image space; choice of size and location of plots on a page; choice of viewing angle and viewing distance; and various options which affect the appearance of the plot. PLOT3D communicates with the calling program through the following: (1) an argument list; (2) common blocks; (3) files; and (4) a set of file manipulation subroutines. An annotated listing illustrates the control, program, and input data cards for a sample run, along with the associated output. (Author/CS)

PHYSICAL PROCESSES IN TERRESTRIAL AND AQUATIC ECOSYSTEMS

COMPUTER PROGRAMS AND GRAPHICS CAPABILITIES

PROGRAMMER'S GUIDE FOR SUBROUTINE PLOT3D

by

Larry Gales

CENTER FOR QUANTITATIVE SCIENCE IN

FORESTRY, FISHERIES AND WILDLIFE

University of Washington

PROGRAMMER'S GUIDE FOR SUBROUTINE PLOT3D

by

Larry Gales

This instructional module is part of a series on Physical Processes in Terrestrial and Aquatic Ecosystems supported by National Science Foundation Training Grant No. GZ-2980

MAY 1978

PROGRAMMER'S GUIDE FOR SUBROUTINE PLOT3D

Identification

PLOT3D -A Subroutine Which Generates Three-Dimensional Hidden Line Displays

Author -Larry Gales

Date -May, 1978- Center for Quantitative Science in Forestry, Fisheries and Wildlife, University of Washington, Seattle, Washington 98195

Purpose

PLOT3D is a subroutine package which generates a variety of three-dimensional hidden line CalComp-type displays. The package features multiple plots per page, with or without annotation, or annotation without plots; multiple pages; automatic scaling; flexible titling; two-dimensional interpolation over the image space; choice of size and location of plots on a page; choice of viewing angle and viewing distance; and various options which affect the appearance of the plot. Each page of plots fits on a standard 8-1/2 x 11 inch page with margins of sufficient size to permit inclusion in three-ring binders. For a more detailed description of the purpose and usage of PLOT3D refer to its user's guide (Gales 1978).

Usage

PLOT3D communicates with the calling program through: (1) an argument list, (2) common blocks, (3) files, and (4) a set of file manipulation subroutines.

•Argument List:

PLOT3D is invoked by the following statement in the calling program:

```
CALL QQPL3D(TLF, ERF, DTF, DTF1, NX, NY,
•           LBOX, LHIDE, XMIN, XMAX, YMIN, YMAX,
•           ZMIN, ZMAX, XRICH, YRICH, DFAULT,
•           XVIEW, YVIEW, ZVIEW, XC, YC,
•           XSIZE, YSIZE, PAGE, TITLE, LOCTIT,
•           PXC, PYC, ERR)
```

where QQPL3D is the main entry point in PLOT3D. All of the arguments, except PXC, PYC, and ERR are input arguments only and are unaffected by the operation of the program. ERR is an output argument which is set greater than zero if, and only if, PLOT3D detects an error. PXC and PYC are input/output arguments and contain the location of the lower left corner of the previous plot upon exit from PLOT3D. PXC and PYC should never be assigned values in the calling program except prior to the very first call, in which case they should be set to zeroes. The types, restrictions, and descriptions of the arguments are as follows:

ARGUMENT LIST

NAME	TYPE	RESTRICTIONS	DESCRIPTION
TLF	Integer	See file descriptions	The unit number of a file written by the calling program which contains titles which annotate the plot.
ERF	Integer	See file descriptions	The unit number of a file written by PLOT3D which displays error messages.
DTF	Integer	See file descriptions.	The unit number of a binary file written by the calling program which contains the X, Y and Z coordinates of the image points to be plotted.
DTF1	Integer	See file descriptions	The unit number of a binary scratch file written by PLOT3D which contains the coordinates of points to be plotted.
NX NY	Integer	$2 \leq NX \leq 30$ $2 \leq NY \leq 30$	The number of grid cells in the image space along the x and y axes.
LBOX	Integer	0, 1, 2, or 3	The box control for subroutine PICTURE. This parameter controls the structure of the three-dimensional window, or box, which contains the plot. The options are: LBOX = 0: No box lines. LBOX = 1: Surface and box lines. LBOX = 2: Surface, box, and side bars. LBOX = 3: Surface, box, side bars and labels. Normally, LBOX = 3.
LHIDE	Integer	0, 1, 2, -1, -2	The hidden line control parameter for subroutine PICTURE. The options are: LHIDE = 0: No hidden lines removed. LHIDE = 1: Hidden lines of plot but not of box are removed. LHIDE = 2: All hidden lines are removed. LHIDE < 0: Visible portions of the underside are also plotte Normally, LHIDE = 2.

NAME	TYPE	RESTRICTIONS	DESCRIPTION
XMIN XMAX YMIN YMAX ZMIN ZMAX	Real	$\geq -10^{29}$ $\leq 10^{29}$	XMIN, XMAX, YMIN, YMAX, ZMIN, and ZMAX define a three-dimensional window which encloses all the data in the image space to be displayed. Data outside the window are not shown. If one or more of the following conditions hold: XMIN $\geq$ XMAX, YMIN $\geq$ YMAX, ZMIN $\geq$ ZMAX the computer considers the limits to be unspecified by the user and computes new limits for the axis or axes in question which are sufficient to include all the data along that axis or axes in the image space.
XRICH YRICH	Real	≥ 0	XRICH and YRICH are the Δx and Δy increments used in two-dimensional interpolation (enrichment). If either is zero, no enrichment takes place. The user should note that the values of XRICH and YRICH should be coordinated not only with the data, but also with the size of the window set by XMIN, XMAX, YMIN, and YMAX. If either is too small, the computer will consume too much computer time, whereas large values of XRICH or YRICH will affect the appearance of the plot. Note that interpolation should only be applied to data which are correctly organized on the binary data file.
DFAULT	Real	$\geq -10^{29}$ $\leq 10^{29}$	DFAULT is the default value assigned to all cells in the image space. DFAULT is usually set to zero.
XVIEW YVIEW ZVIEW	Real	|XVIEW| > XSIZE |YVIEW| > YSIZE |ZVIEW| > XSIZE |ZVIEW| > YSIZE	The point of view in space, in inches, from which the plot is seen by the user. Normally, XVIEW and YVIEW should be negative, indicating that the observer is behind the plot so that it appears in front of him, while ZVIEW should be positive, indicating that the observer is above the plot looking down. The user may have to experiment with different values for these viewpoints in order to obtain the desired view of the plot.

NAME	TYPE	RESTRICTIONS	DESCRIPTION
XC YC	Real	$0.5 \leq XC \leq 8$ $0.5 \leq YC \leq 10.5$	The location on a page of the lower left corner of a box which encloses the plot. The distances are in inches from the lower left corner of the page.
XSIZE YSIZE	Real	$XSIZE \geq .1$ $XC + XSIZE \leq 8$ $YSIZE \geq .1$ $YC + YSIZE \leq 10.5$	The size of a box, in inches, which is to contain the plot.
PAGE	Logical	T or F	If PAGE = T (true), then the current page is skipped and the current plot is written on the next page. If PAGE = F (false), the current plot is written on the current page.
TITLE	Logical	T or F	If TITLE = T (true), then the current plot consists of plot titles only, with no plot. If TITLE = F (false), then both plot and plot titles are written. Note that in the former case, the entire area defined by XSIZE, YSIZE is available for plot titles (or any other text), whereas in the latter case, the plot titles are either squeezed in near the top of the area defined by XSIZE, YSIZE, or just above it (see LOCTIT).
LOCTIT	Logical	T or F	If LOCTIT is T (true), then the plot titles will be contained wholly within the region defined by XC, YC, XSIZE, and YSIZE. If LOCTIT is F (false), the plot titles are placed above and outside this region.
PXC PYC	Integer	None (set by PLOT3D)	The location of the lower left corner of the previous plot (in inches). PXC, PYC are reset to zeroes whenever PAGE is true.
ERR	Integer	None (set by PLOT3D)	An output argument which is zero if PLOT3D detects no errors, or is greater than zero if it does detect errors. See the user's guide for a detailed explanation of the error codes.

- Common blocks:

PLOT3D uses blank common and four labeled common blocks named /QQPL1/, /QQPL2/, /QQPL3/, and /QQXYZ/. Blank common serves as temporary storage for the image space (F), a counter for the number of points mapped to each image cell (NP), and two temporary storage arrays (TEMPX, TEMPY), and is structured as follows:

```
COMMON//  F(30,30),     NP(30,30),     TEMPX(30,30),     TEMPY(30,30)
REAL      F,            NP,            TEMPX,            TEMPY
```

Therefore, the calling program must reserve 4 x 30 x 30 = 3600 (decimal) words of workspace at the start of blank common, e.g.,

```
COMMON// WSPACE(3600)
REAL     WSPACE
```

Since PLOT3D uses this area for temporary storage only and does not preserve values between calls, the calling program may access and modify this region in blank common any way it chooses, except that values stored in the first 3600 words will be destroyed when PLOT3D is called.

The three common blocks /QQPL1/, /QQPL2/, and /QQPL3/ are used only for internal operations in PLOT3D and can be ignored by the calling program. Common block /QQXYZ/, however, serves a vital role in facilitating binary input and output of data points for the image space both within the calling program and PLOT3D. /QQXYZ/ is structured as follows:

```
COMMON /QQXYZ/ X, Y, Z,   FINI, EOFT
REAL           X, Y, Z,   FINI
LOGICAL        EOFT
```

where X, Y, Z are the coordinates of one data point, FINI is the value assigned to the end-of-file indicator (-99999.0), and EOFT is a logical variable which is set true iff an end-of-file is read. Any binary data file passed to PLOT3D by a calling

program should be read, written, terminated, and rewound by the special file manipulation subroutines QQRXYZ, QQWXYZ, QQWEOF, and QQREW, respectively, contained in PLOT3D. The calling program must explicitly set FINI = -99999.0 before any of these routines is invoked; otherwise, FINI will be undefined and the results unpredictable.

• Files:

PLOT3D uses four files named TLF, ERF, DTF, and DTF1. TLF is the unit number of a formatted file which passes plot title information to PLOT3D. It must contain four card images, each of which is at least 66 characters long, which label the top of the plot. TLF is normally written by the calling program, although it may reside on an external file. It is automatically rewound by PLOT3D at the start and end of execution.

ERF is the unit number of a formatted file written by PLOT3D which displays any error messages.

DTF is the unit number of a binary file which passes the x, y, z coordinates of plot image data points to PLOT3D. DTF is normally written by the calling program using subroutine QQWXYZ and must be terminated by an end-of-file written by QQWEOF. The order of points on DTF depends on whether enrichment (interpolation) is selected. If no enrichment is called for, the points may be ordered randomly. If two-dimensional enrichment is called for, then the points must form a sequence of triangles. For a more complete description of enrichment and its effects on ordering, refer to the user's guide. DTF is automatically rewound by PLOT3D at the start and end of execution.

DTF1 is the unit number of a binary scratch file which is written and read by PLOT3D if enrichment is selected. DTF1 is automatically rewound by PLOT3D at the start and end of execution.

The characteristics of the files used by PLOT3D are summarized as follows:

FILE NAME	AUTOMATIC REWIND	READ BY PLOT3D	WRITTEN BY PLOT3D	UNIQUE UNIT NUMBER
TLF	Yes	Yes	No	Yes
ERF	No	No	Yes	Yes
DTF	Yes	Yes	No	Yes
DTF1	Yes	Yes	Yes	Yes

The column labeled "AUTOMATIC REWIND" means that subroutine PLOT3D rewinds the file at the start of its execution and then rewinds it again just before it returns to the calling program. The column labeled "UNIQUE UNIT NUMBER" specifies whether or not different file names may reference the same unit number. Here, all files must be unique.

Subroutine PLOT3D does not check the files for format errors nor does it check to see if file names reference valid unit numbers. These types of errors will generally trigger error messages and actions which are peculiar to a given computer installation.

- File Manipulation Subroutines:

PLOT3D contains four subroutines which manipulate the binary data files. These routines are invoked by PLOT3D and also by the calling program which prepares data for PLOT3D, and are always used in conjunction with the common block /QQXYZ/ which holds the x, y, z coordinates for a single point to be written or read. The

routines are:

QQRXYZ (FILE): Reads the x, y, z coordinates of one data point from FILE and stores the coordinates in /QQXYZ/.

QQWXYZ (FILE): Retrieves the x, y, z coordinates of a point stored in /QQXYZ/ and writes them on FILE.

QQWEOF (FILE): Writes the point (FINI, FINI, FINI) on FILE to indicate the end-of-file.

QQREW (FILE): Rewinds FILE.

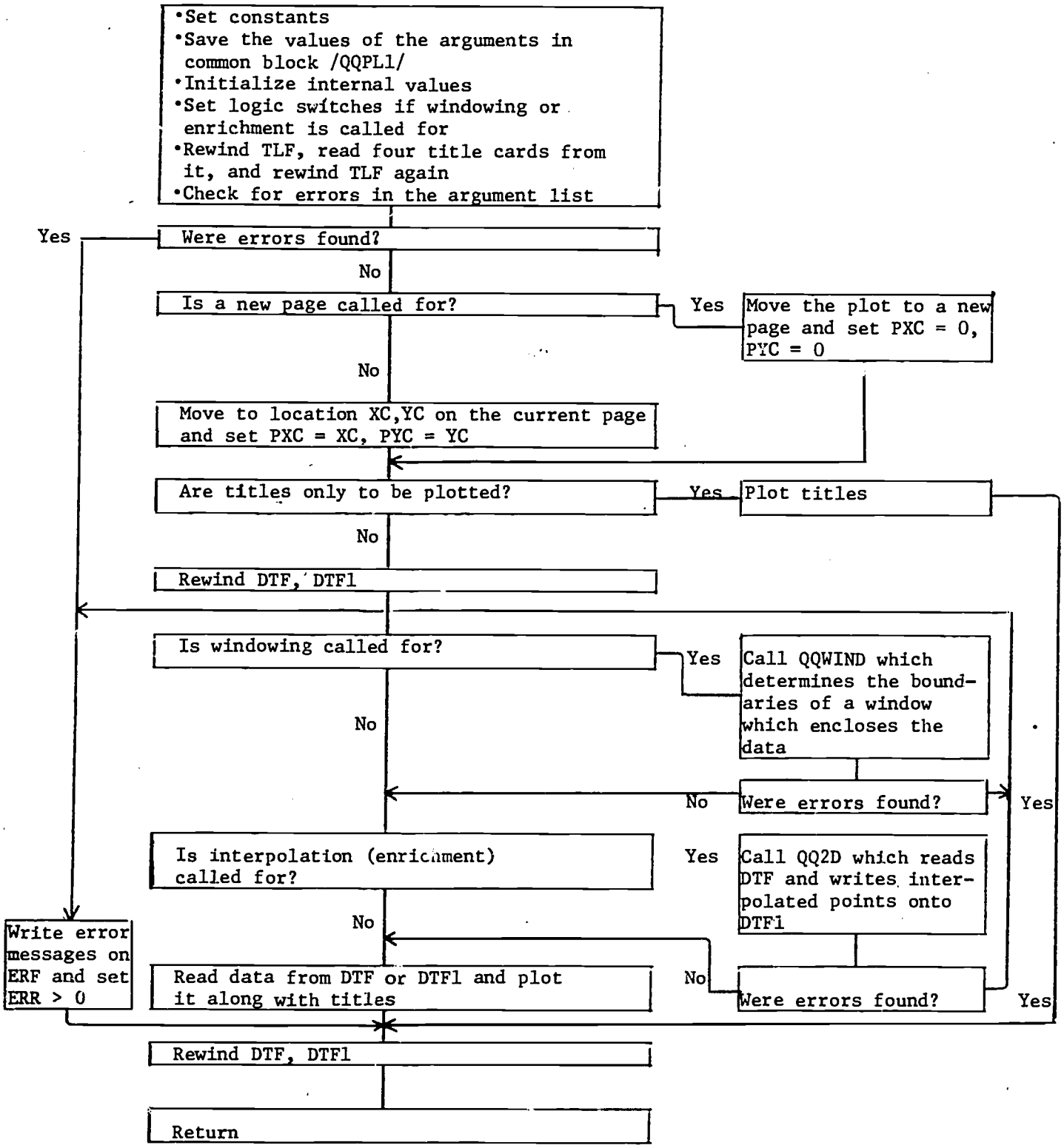
As an example of the use of these routines in the calling program, consider the following fragment of computer code which writes 1000 data points on unit 1, terminates unit 1 with an end-of-file, and then reads the points back in. Note FINI must be established before the routines are called and EOFT must be cleared before testing for an end-of-file. Also note that X= Y = Z = FINI = -99999.0 in common block /QQXYZ/ when the end-of-file is read. The code is as follows:

```
          . . .
          COMMON /QQXYZ/ X, Y, Z,  FINI, EOFT

C------ SET-END-OF FILE INDICATOR
       FINI = -99999.0

C
C----- REWIND TAPE 1
       CALL QQREW(1  )

C
C----- WRITE 1000 DATA POINTS ONTO TAPE 1
       DO 10 I = 1, 1000
         X = ...
         Y = ...
         Z = ...
         CALL QQWXYZ(1   )
  10 CONTINUE
C
C----- END FILE TAPE 1
       CALL QQWEOF(1   )
```

```
C
C----- REWIND TAPE 1, AND PRESET EOFT TO FALSE
       CALL QQREW(1    )
       EOFT =  .FALSE.
C
C----- READ BACK THE 1000 POINTS
       DO 20 I = 1, 2000
       CALL QQRXYZ(1    )
       IF (EOFT) GOTO 30
       ... = X
       ... = Y
       ... = Z
    20 CONTINUE
C
    30 CONTINUE
```

Structure

The overall structure of PLOT3D is displayed in the following flow chart:



Subroutines

The following is a list and brief description of all subroutines contained in PLOT3D in alphabetical order:

QQ2D: Called when two-dimensional interpolation (enrichment) is selected. It transfers all points from file DTF to DTF1 and, after every third point, it calls QQTRI which adds large numbers of interpolated points to DTF1. QQ2D assumes that three consecutive points on DTF are the vertex points of a single triangle.

QQCK: Checks the input parameters NX, NY, XSIZE, YSIZE, XC, and YC for errors in value.

QQCSIZ: Calculates the size of characters and the distance between text lines, for the four title lines, so as to insure that they will properly fit in the space allocated within, or above, the plot.

QQER: Writes out error messages on file ERF and sets ERR to an appropriate nonzero value.

QQF: Reads data from a file and maps each data point within the x and y axes of the current window onto a cell in the image space F. It also accumulates the number of points mapped to each image cell, in the array NP, and tallies the number of z values which lie outside the ZMIN to ZMAX limits.

QQNCHR: Calculates the maximum number of characters in the title lines, minus all trailing blanks.

QQPAK: Packs text consisting of one character per word into 10 characters per word (for the CDC 6400 computer).

QQPLOT: Calls subroutine PICTURE which generates the actual CalComp plotter commands to produce a three-dimensional hidden line plot.

QQPLT2: Calls the CalComp subroutine PLOT to initialize the plotter so that the origin is set at a specified point.

QQPTTL: Calls QQNCHR, QQCSIZ, and QQPAK to prepare the plot titles for plotting.

QQPTXT: Plots the titles prepared by QQPTTL by calling the CalComp plotter routine SYMBOL which generates the plotting instructions.

QQREW: Rewinds a file.

QQRXYZ: Reads one x,y,z coordinate triple from the binary data file and stores it in common block /QQXYZ/. If X = Y = Z = FINI, it sets EOFT = .TRUE.

QQTRI: Accepts the x, y, z coordinates of three vertex points of a triangular region and generates a series of equispaced linearly interpolated points along the plane defined by the three points. The points are written on DTF1.

QQWEOF: Writes the point (FINI, FINI, FINI) on a file. This point establishes the end-of-file.

QQWIND: Computes the x, y, and z coordinates of a cubical window which encloses the data. It is called if:

$$XMIN \geq YMAX, \text{ or}$$

$$YMIN \geq YMAX, \text{ or}$$

$$ZMIN \geq ZMAX.$$

If any one or more of the above conditions is encountered it computes new values for the x, y, or z axes which completely contain the data.

QQWXYZ: Writes the x, y, z coordinates for the point currently stored in /QQXYZ/ on a file.

External Subroutines

The following is a list and brief description of all subroutines called by PLOT3D which are not contained in the source deck.

PICTURE: This is a very large and complex plotting routine, adapted from the program developed by Melvin L. Prueitt of Los Alamos Scientific Laboratory, which generates commands for the CalComp plotter to produce three-dimensional hidden line displays. This routine is non-ANSI and rather machine dependent.

PLOT: This is a CalComp plotter routine which is used to set the origin of a plot.

PUTC: This is a small machine-dependent subroutine which packs one character-per-word text arrays into n characters-per-word, where n is the maximum number of characters which can be stored in a computer word (n equals 10 on the CDC 6400).

SYMBOL: This is a CalComp plotter command which is used to plot characters.

Coding Information

- Literals and constants:

  The literals used in PLOT3D can be divided into seven classes:

  1) the integers 0, 1, 2 and the real numbers 0.0 and 1.0 used as initial values or offsets;

  2) the integer 1 and the real number 1.0 used as an increment/decrement;

  3) the integers 0, 1, 2, and 3 and the real number 0.0 used in arithmetic tests;

  4) the integers 1, 2, and 3 used as subscripts;

5) the integer -3 and real number 0.0 used as arguments;

6) the logical constants .TRUE. and .FALSE.;

7) the integers 1 through 13 used as error numbers.

All constants are assigned values in the "CONSTANTS" section of each routine and are described in the following table:

CONSTANTS

NAME	VALUE	SUB	BLOCK	DESCRIPTION
NXD	30	QQPL3D	/QQPL2/	Maximum dimension of the image space along the x axis
NYD	30	QQPL3D	/QQPL2/	Maximum dimension of the image space along the y axis
PAGEW	8.5	QQPL3D	/QQPL2/	Width of a plotter page, in inches
EXTRME	10^{30}	QQPL3D	/QQPL2/	An extremely large value
EPS	10^{-30}	QQPL3D	/QQPL2/	An extremely small value
MINXS	0.1	QQCK	---	Minimum allowed value of XSIZE
MAXXCS	8.001	QQCK	---	Maximum allowed value of XC + XSIZE
MINYS	0.1	QQCK	---	Minimum allowed value of YSIZE
MAXYCS	10.501	QQCK	---	Maximum allowed value of YC + YSIZE
MINXC	0.499	QQCK	---	Minimum allowed value of XC
MAXXC	8.001	QQCK	---	Maximum allowed value of XC
MINYC	0.499	QQCK	---	Minimum allowed value of YC
MAXYC	10.501	QQCK	---	Maximum allowed value of YC
STDCSZ	0.27	QQCSIZ	---	Size of characters, in inches, in the titles for a standard size plot
STDDY1	0.36	QQCSIZ	---	Vertical distance, in inches, between title lines for a standard size plot

NAME	VALUE	SUB.	BLOCK	DESCRIPTION
STDDY2	1.4	QQCSIZ	---	Vertical distance, in inches, between title lines on a standard size plot when titles only are plotted, without the plot itself
STDSZ	6.0	QQCSIZ	---	The size, in inches, of a standard size plot along the x and y axes
ROUND	1.5	QQF	---	A rounding factor used when locating the grid cell into which an x, y, z point is stored
BLANK	1H	QQNCHR	---	The blank character
LINE	0	QQPLOT	---	A parameter used by subroutine PICTURE
SC	1.0	QQPLOT	---	A parameter used by subroutine PICTURE
STDSZ	6.0	QQPLOT	---	The size, in inches, of a standard size plot
TITSZ	4.2	QQPLOT	---	The size, in inches, of a standard size plot when titles only are plotted
XCDSPL	0.9	QQPLOT	---	The standard size of XC when titles only are to be plotted
YC1	0.0	QQPLOT	---	The YC displacement when titles only are to be plotted
NCL	66	QQPTTL	---	The maximum number of characters in a title line
NLINE	4	QQPTTL	---	The maximum number of title lines

- Word Lengths:

All values in PLOT3D are assumed to be stored in full length single precision words. All alphanumeric values are stored one character per word, except text strings packed by QQPAK, which are stored 10 characters per word.

• Naming Conventions:

All subroutines and common blocks start with the letters "QQ" in order to insure uniqueness. In addition, all variables in a common block which are not used in a given subroutine are represented by dummy variables of the form ZZnn or ZZZn where n is a digit. These dummy variables may span more than one array or set of names. For example, assume subroutine SB1 only makes use of the variable FINI in /QQXYZ/. Then /QQXYZ/ is declared in SB1 as follows:

```
      SUBROUTINE SB1
           ...
      COMMON /QQXYZ/ ZZZ1(3), FINI, ZZZ2
      REAL           ZZZ1,    FINI
      LOGICAL        ZZZ2
           ...
```

Limitations

PLOT3D checks for 11 error conditions. If any of these conditions occurs, it outputs an appropriate error message and returns to the calling program. For a description of the error conditions and messages refer to the user's guide.

PLOT3D does not check to see if the files are correctly formatted or if unit numbers are valid. Such errors are left to the computer system.

Extensions

PLOT3D can be expensive to run if large numbers of data points are to be read, generated, or displayed. Much of the execution time is consumed in the input and output of data points using binary read/write operations (formatted read/write operations are even slower). If sufficient computer memory is available, however, one can drastically reduce this time by storing the data points directly in memory

and accessing them through simple memory reference operations. The four file manipulation routines discussed above were written with just this possibility in mind. It is only necessary to alter these routines in an appropriate manner in order to simulate the binary read/write operations with retrieve/store operations. The latter are logically identical to the former but are an order of magnitude faster.

Computer Resources

- Storage:

The object deck for PLOT3D occupies 4142 (octal) words of storage when compiled under the Minnesota Fortran compiler on the CDC 6400 computer. To this must be added the 3600 (decimal) words in blank common used for the image space, plus buffer areas needed for the four files TLF, ERF, DTF, and DTF1. On the CDC 6400, this amounts to approximately 4242 + 7020 + 4000 = 17162 (octal) words of storage, some of which may be shared by the calling program.*

- Execution Time:

The execution time for PLOT3D depends primarily on the number of grid cells in the image space and the number of data points generated by two-dimensional interpolation. The following table gives the approximate execution times in CPU (central processing unit) seconds on the CDC 6400 computer as a function of grid size and number of data points:

*Note that the CDC 6400 has up to four instructions per word. Hence, other computers may require substantially more space for the computer code.

PLOT OPTION	GRID SIZE	APPROX. NO. POINTS	CPU SECONDS
No interpolation	30 by 30	24	4.0
No interpolation	5 by 5	24	0.42
2D interpolation	30 by 30	1200	6.72
2D interpolation	5 by 5	1200	3.53

Sample Run

The annotated listing on the next few pages illustrates the control, program, and input data cards for a sample run, along with the associated output. The output consists of a set of plots and annotation on one page which show the effects of various plot options applied to a single file of x, y, z coordinates. Note that the input data cards are processed by a free form input system (Gales and Anderson 1978; Anderson and Gales 1978).

```
XPL3D,CM50000,T35,P2.          PLOT3D TEST
ACCOUNT,XXXXXXXX,XXXXXX.
COMMENT.
COMMENT.*************************************************************
COMMENT.* THE ABOVE CARDS IDENTIFY THE JOB, SPECIFY THE MEMORY       *
COMMENT.* REQUIREMENTS (50000 OCTAL), ESTIMATE THE CENTRAL PROCESSOR *
COMMENT.* TIME (35 SECONDS), AND SPECIFY THE USERS ACCOUNT           *
COMMENT.*************************************************************
COMMENT.
MNF,E=0,R=7,J.
COMMENT.
COMMENT.*************************************************************
COMMENT.* THE ABOVE CARD CALLS THE MINNESOTA FORTRAN COMPILER TO     *
COMMENT.* COMPILE THE EXECUTION PROGRAM FOR PLOT3D                   *
COMMENT.*************************************************************
COMMENT.
ATTACH,BPL3D,ID=BPL3D.
COMMENT.
COMMENT.*************************************************************
COMMENT.* THE ABOVE CARD ATTACHES THE PLOTTING ROUTINE IN BINARY FORM *
COMMENT.*************************************************************
COMMENT.
ATTACH,BFF,ID=BFF.
COMMENT.
COMMENT.*************************************************************
COMMENT.* THE ABOVE CARD ATTACHES THE FREE-FORMAT INPUT ROUTINES IN   *
COMMENT.* BINARY FORM.                                               *
COMMENT.*************************************************************
COMMENT.
ATTACH,GRAFTN.
LIBRARY,GRAFTN.
COMMENT.
COMMENT.*****************************************
COMMENT.* THE ABOVE CARDS ATTACH THE LIBRARY   *
COMMENT.* ROUTINES FOR PL3D AND DESIGNATE THAT *
COMMENT.* FILE AS THE CURRENT LIBRARY.         *
COMMENT.*****************************************
COMMENT.
LOAD,BFF.
LOAD,BPL3D.
LOAD,LGO.
EXECUTE,XPL3D.
COMMENT.
COMMENT.*************************************************************
COMMENT.* THE ABOVE CARDS LOAD THE EXECUTION, INPUT AND PLOT ROUTINES *
COMMENT.* AND CAUSE CONTROL TO BE PASSED TO XPL3D FOR EXECUTION      *
COMMENT.*************************************************************
CATALOG,TAPE99,RP=5.
COMMENT.
COMMENT.*****************************************
COMMENT.* THE ABOVE CARD SAVES THE PLOTTING    *
COMMENT.* TAPE,-TAPE99-, SO THAT THE PLOTS MAY *
COMMENT.* BE PREVIEWED BEFORE BEING SENT TO    *
COMMENT.* THE PLOT QUEUE.                      *
COMMENT.*****************************************
COMMENT.
COMMENT.*************************************************************
COMMENT.* THE BELOW CARDS ARE THE EXECUTION PROGRAM XPL3D            *
COMMENT.*************************************************************
COMMENT.
```

```
*EOR
      PROGRAM XPL3D(INPUT,OUTPUT,TAPE1,TAPE2,TAPE3,TAPE5=INPUT,
     .  TAPE6=OUTPUT)
C
C
C-PURPOSE----------------------------------------------------------------
C
C     DEMONSTRATES THE USE OF THE PLOTTING PACKAGE PLOT3D
C
C
C-IDENTIFICATION---------------------------------------------------------
C
C     DATE:          AUGUST 1977
C     LOCATION:      CENTER FOR QUANTITATIVE SCIENCE
C                        IN FORESTRIES, FISHERIES, AND WILDLIFE
C                    UNIVERSITY OF WASHINGTON
C                    SEATTLE, WASHINGTON 98195
C     LANGUAGE:      ANSI STANDARD FORTRAN
C     INSTALLATION: CDC 6400/CYBER 73 UNDER SCOPE 3.4 O.S.
C     PROGRAMMERS:   L.GALES, M.BAILEY, J. SANISLO
C
C
C-GLOBAL VARIABLES-------------------------------------------------------
C
C                    ************************************************
C                    *              BLANK COMMON                    *
C                    * USED FOR STORAGE OF INPUT VARIABLES READ     *
C                    * BY INPUT PACKAGE.                            *
C                    * -WSPACE- IS USED BY THE PLOT ROUTINES.       *
C                    ************************************************
C
      COMMON / /     WSPACE(3600),
     .               NX,     NY,     LBOX,   LHIDE,  XMIN,   XMAX,
     .               YMIN,   YMAX,   ZMIN,   ZMAX,   XRICH,  YRICH,
     .               DFAULT, XVIEW,  YVIEW,  ZVIEW,  XC,     YC,
     .               XSIZE,  YSIZE,  PAGE,   TITLE,  LOCTIT
      REAL           WSPACE
      INTEGER        NX,     NY,     LBOX,   LHIDE
      REAL           XMIN,   XMAX,   YMIN,   YMAX,   ZMIN,   ZMAX,
     .               XRICH,  YRICH,  DFAULT, XVIEW,  YVIEW,  ZVIEW,
     .               XC,     YC
      LOGICAL        PAGE,   TITLE,  LOCTIT
C
C     DEFINITIONS
C
C     WSPACE(*) =    STORAGE AREA FOR PLOT ROUTINES
C     NX,NY =        NUMBER OF X AND Y CELLS IN THE IMAGE SPACE
C     LBOX =         BOX CONTROL FOR SUBROUTINE -PICTURE-
C     LHIDE =        HIDDEN LINE REMOVAL FLAG FOR SUBROUTINE -PICTURE-
C     XMIN,XMAX,
C     YMIN,YMAX,
C     ZMIN,ZMAX =    DEFINE A RECTANGULAR PARALLELOPIPED WHICH ENCLOSES
C                        THE DATA TO BE PLOTTED
C     XRICH,YRICH = THE DELTA X AND DELTA Y INCREMENTS USED IN TWO-
C                        DIMENSIONAL ENRICHMENT
C     DFAULT =       THE DEFAULT VALUES ASSIGNED TO ALL POINTS
C     XVIEW,YVIEW,
C     ZVIEW =        THE POINT OF VIEW FROM WHICH THE PLOT IS OBSERVED
C                        (IN INCHES)
C     XC,YC =        LOCATION ON PAGE OF LOWER LEFT CORNER OF PLOT
```

```
C                        (IN INCHES)
C      XSIZE,YSIZE = PHYSICAL SIZE OF PLOT (IN INCHES)
C      PAGE =        .TRUE. IFF THE CURRENT PLOT IS TO APPEAR ON THE NEXT
C                        PAGE
C      TITLE =       .TRUE. IFF THE CURRENT PLOT CONSISTS OF ONLY TEXT
C
C                    ************************************************
C                    *                                              *
C                    *            DATA COMMON PL3D1                 *
C                    * USED FOR STORAGE OF THE DATA POINTS          *
C                    *                                              *
C                    ************************************************
C
      COMMON/PL3D1/ X(8),    Y(8),    Z(8)
      REAL          X,       Y,       Z
C
C     DEFINITIONS
C
C     X(*),Y(*),Z(*) = THE LOCATION OF EACH DATA POINT
C
C
C                    ************************************************
C                    *                                              *
C                    *            VERTEX COMMON PL3D2               *
C                    * CONTAINS THE LOCATION OF THE VERTICES        *
C                    * FOR EACH TRIANGULAR REGION                   *
C                    *                                              *
C                    ************************************************
C
      COMMON/PL3D2/ EX(8,3),EY(8,3),EZ(8,3)
      REAL          EX,     EY,     EZ
C
C     DEFINITIONS
C
C     EX(*,*) =     THE X COORDINATE OF EACH VERTEX
C     EY(*,*) =     THE Y COORDINATE OF EACH VERTEX
C     EZ(*,*) =     THE Z COORDINATE OF EACH VERTEX
C
C
C                    ************************************************
C                    *                                              *
C                    *                COMMON QQXYZ                  *
C                    * USED IN QQPL3D FOR ACCESSING THE DATA        *
C                    * POINTS AND CHECKING FOR AN END-OF-FILE       *
C                    *                                              *
C                    ************************************************
C
      COMMON/QQXYZ/ ZZZ1(3),FINI,
     .              ZZZ2
      REAL          ZZZ1,   FINI
      LOGICAL       ZZZ2
C
C     DEFINITIONS
C
C     FINI =        -99999.0, AN END-OF-FILE MARK
C
C
C-LOCAL VARIABLES-----------------------------------------------------
C
      INTEGER       TLF,    ERF,    DTF,    DTF1,   INF,    ECF,
```

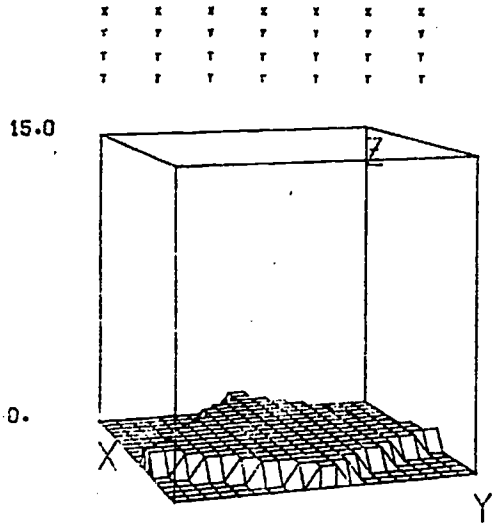
```
     .                DCF,    ERR,    C(48),  I,      J,      K
      REAL            PXC,    PYC
C
C     DEFINITIONS
C
C     TLF =           TITLE FILE CONTAINING FOUR TITLES WHICH LABEL PLOT
C     ERF =           THE ERROR MESSAGE FILE
C     DTF =           DATA FILE OF X, Y, AND Z COORDINATES
C     DTF1 =          SCRATCH DATA FILE
C     INF =           INPUT FILE CONTAINING VARIABLE VALUES
C     ECF =           FILE USED TO ECHO INPUT
C     DCF =           DECLARATION FILE USED TO DECLARE INPUT VARIABLES
C     ERR =           ERROR NUMBER
C     C(*) =          THE TITLE TEXT
C     I,J,K =         LOOP INDICES
C     PXC,PYC =       LOCATION OF THE LOWER LEFT CORNER OF THE PREVIOUS
C                         PLOT (IN INCHES)
C
C
C-SUBROUTINES-----------------------------------------------------------
C
C                     EL
C                     NMLIST
C
C
C-EXTERNALS-------------------------------------------------------------
C
C     QQPL3D          THE PLOTTING ROUTINE
C     PLOT,PLOTS      THE INITIALIZATION AND TERMINATION ROUTINES FOR
C                     THE PLOTTER (LOCATED IN THE GRAPHICS LIBRARY)
C     QQWEOF          WRITES AN END-OF-FILE MARK ON THE DESIGNATED FILE
C                     (LOCATED IN THE PLOTTING PACKAGE)
C     QQREAD          READS INPUT VARIABLES IN FREE FORMAT
C
C-FILES-----------------------------------------------------------------
C
C     TLF             INPUT,  FORMATTED,      TEXTFILE
C     ERF             OUTPUT, FORMATTED,      ERROR MESSAGE FILE
C     DTF             INPUT,  BINARY,         DATA FILE OF COORDINATES
C     DTF1            IN/OUT, BINARY,         SCRATCH DATA FILE
C     INF             INPUT,  FORMATTED,      INPUT FILE
C     DCF             OUT/IN, FORMATTED,      DECLARATION FILE
C     ECF             OUTPUT, FORMATTED,      ECHO FILE
C
C
C-CONSTANTS-------------------------------------------------------------
C
      DATA            X/  1.0,  3.0,  2.0,  3.0,  4.2,  5.0,  4.5,  3.0  /
      DATA            Y/  3.0,  4.0,  2.0,  2.4,  2.8,  3.0,  2.5,  1.0  /
      DATA            TLF,    ERF,    DTF,    DTF1,   INF,    ECF      /
     .                 3,      6,      1,      2,      5,      6        /
     .                DCF,    FINI      /
     .                 1,    -99999.0   /
     .                C/      8*1HX,  8*1HY,  32*1HT  /
C
C
C-INITIALIZATIONS-------------------------------------------------------
C
C                        /* THE CALL TO -PLOTS- INITIALIZES THE PLOTTER
                      CALL PLOTS
```

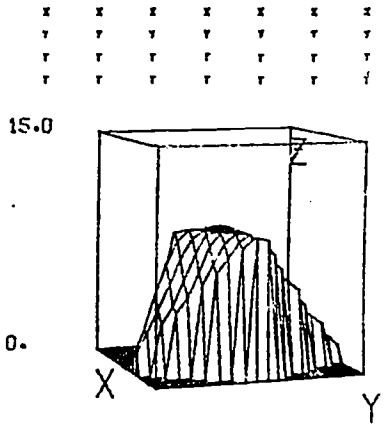
```
                    PXC = 0.0
                    PYC = 0.0
C                       /* THE CALL TO -NMLIST- DECLARES INPUT VARIABLES
                    CALL NMLIST(3600,1H ,DCF,ERF,  ERR)
C                       /* THE CALL TO -QQREAD- READS INITIAL VALUES
                    CALL QQREAD(INF,ECF,ERF,  ERR)
C
C
C-START-------------------------------------------------------------------
C
                    DO 701 I=1,4
                        GOTO(10,?0,30,40),I
C
   10                   DO 702 J=1,8
                            Z(J) = 1.0
  702                   CONTINUE
C                           /* READ PLOT PARAMETERS
                        CALL QQREAD(INF,ECF,ERF,  ERR)
                        GOTO 999
C
   20                   Z(2) = 9.0
                        Z(4) = 9.0
                        Z(5) = 9.0
C                           /* READ PLOT PARAMETERS
                        CALL QQREAD(INF,ECF,ERF,  ERR)
                        GOTO 999
C
   30                   DO 703 J=1,8
                            Z(J) = 9.0
  703                   CONTINUE
                        Z(5) = 0.C
C                           /* READ PLOT PARAMETERS
                        CALL QQREAD(INF,ECF,ERF,  ERR)
                        GOTO 999
C
C                           /* READ PLOT PARAMETERS
   40                   CALL QQREAD(INF,ECF,ERF,  ERR)
                        GOTO 998
C
  999                   CALL FL(1, 1,3,2)
                        CALL EL(2, 2,3,4)
                        CALL EL(3, 2,4,5)
                        CALL EL(4, 2,5,6)
                        CALL EL(5, 5,6,7)
                        CALL EL(6, 5,8,7)
                        CALL EL(7, 5,4,8)
                        CALL EL(8, 4,3,8)
                        DO 704 J=1,8
                            DO 705 K=1,3
                                WRITE(DTF) EX(J,K),EY(J,K),EZ(J,K)
  705                       CONTINUE
  704                   CONTINUE
                        CALL QQWEOF(DTF  )
  998                   WRITE(TLF,1) C
    1 FORMAT            (8A10)
                        CALL QQPL3D(TLF,ERF,DTF,DTF1,NX,NY,LBOX,LHIDF,
                                    XMIN,XMAX,YMIN,YMAX,ZMIN,ZMAX,
                                    XRICH,YRICH,DFAULT,XVIEW,YVIEW,
                                    ZVIEW,XC,YC,XSIZE,YSIZE,PAGE,TITLE,
                                    LOCTIT,  PXC,PYC,  ERR)
```

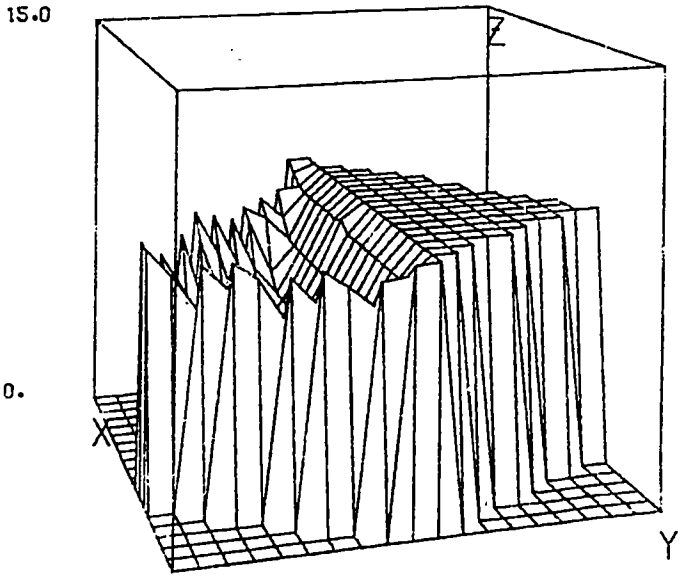
```
  701               CONTINUE
C
C                       /* THE CALL TO -PLOT(0,0,999)- TERMINATES THE
C                       /* PLOTTER
                    CALL PLOT(0,0,999)
                    STOP
                    END
      SUBROUTINE FL(I, I1,I2,I3)
C
C
C-PURPOSE----------------------------------------------------------
C
C     CALCULATES THE VERTICES  OF EACH TRIANGULAR REGION
C
C
C-ARGUMENTS--------------------------------------------------------
C
      INTEGER       I,      I1,     I2,     I3
C
C     DEFINITIONS
C
C     I,I1,I2,I3 =  INDICES  FOR THE CALULATION OF THE VERTICES
C
C
C-GLOBAL VARIABLES-------------------------------------------------
C
C     NOTE: FOR DEFINITIONS REFER TO MAIN PROGRAM
C
      COMMON/PL3D1/ X(8),   Y(8),   Z(8)
      REAL          X,      Y,      Z
C
C
      COMMON/PL3D2/ EX(8,3),EY(8,3),EZ(8,3)
      REAL          EX,     EY,     EZ
C
C
C-START------------------------------------------------------------
C
                    EX(I,1) = X(I1)
                    EX(I,2) = X(I2)
                    EX(I,3) = X(I3)
                    EY(I,1) = Y(I1)
                    EY(I,2) = Y(I2)
                    EY(I,3) = Y(I3)
                    EZ(I,1) = Z(I1)
                    EZ(I,2) = Z(I2)
                    EZ(I,3) = Z(I3)
                    RETURN
                    END
      SUBROUTINE NMLIST(INDX,NAME,DCF,ERF,  ERR)
C
C
C-PURPOSE----------------------------------------------------------
C
C     WRITE DECLARATION FILE USED BY FREE-FORMAT INPUT ROUTINES
C
C
C-ARGUMENTS--------------------------------------------------------
C
      INTEGER       INDX,   NAME,   DCF,    ERF,    ERR
```

```
C
C     DEFINITIONS
C
C     INDX =          START OF VARIABLE STORAGE IN BLANK COMMON
C     NAME =          I/O LIST IDENTIFIER
C     DCF  =          DECLARATION FILE
C     ERF  =          ERROR FILE
C     ERR  =          ERROR FLAG (REMAINS 0 IF NO ERRORS)
C
C
C-EXTERNALS-----------------------------------------------------------
C
C     QQINTL          FREE-FORMAT INPUT ROUTINE TO PROCESS VARIABLES LIST
C
C
C-START---------------------------------------------------------------
C
                      REWIND DCF
                      WRITE(DCF,1)
                      REWIND DCF
                      CALL QQINTL(INDX,NAME,DCF,ERF,   ERR)
                      REWIND DCF
                      RETURN
    1 FORMAT(
     .62H        INTEGER      NX,      NY,      LBOX,    LHIDE
     ./62H        REAL         XMIN,    XMAX,    YMIN,    YMAX,    ZMIN,
     ./62H                     ZMAX,    XRICH,   YRICH,   DFAULT,  XVIEW,
     ./62H                     YVIEW,   ZVIEW,   XC,      YC,      XSIZE,
     ./62H                     YSIZE
     ./62H        LOGICAL      PAGE,    TITLE,   LOCTIT
     ./62H        $
     ./62H        $
     .)
                      END
*EOR
```

```
 /*
  /* INITIAL PLOT PARAMETERS */
 /*
  XMIN = 1.0,  XMAX = 5.0,  YMIN = 1.0, YMAX = 4.0,
  ZMIN = 0.0,  ZMAX = 15.0,  XRICH = 0.05,  YRICH = 0.05, DEAULT = C.0,
  XVIEW = 5.0,  YVIEW = 13.228, NX = 20, NY = 20,
  ZVIEW = 30.0,
  LBOX = 3,  LHIDE = 2      $
 /*
 /* PARAMETERS FOR PLOT 1 */
 /*
   XC = 1.5,  YC = 6.0,  XSIZE = 2.0, YSIZE = 2.0,
   TITLE = .F.,  LOCTIT = .F.     $
 /*
 /* PARAMETERS FOR PLOT 2 */
 /*
    XC = 5.0,  LOCTIT = .T.    $
 /*
 /* PARAMETERS FOR PLOT 3 */
 /*
    XC = 3.0,  YC = 3.5,  XSIZE = 3.0,  YSIZE = 3.0,  LOCTIT = .F.    $
 /*
 /* PARAMETERS FOR PLOT 4 */
 /*
    XC = 3.0,  YC = 1.5, XSIZE = 5.0,  YSIZE = 1.0,
    TITLE = .T.      $
*EOR
*EOF
```


15.0
0.
Z
X
Y


15.0
0.
Z
X
Y


15.0
0.
Z
X
Y

Acknowledgments

We are indebted to Dr. Melvin L. Prueitt of the Los Alamos Scientific Laboratory for the original PICTURE program which is used in PLOT3D as a subroutine.

References

Anderson, L. and L. Gales. 1978. Programmer's guide for FFORM: a format free input system. Center for Quantitative Science in Forestry, Fisheries, and Wildlife, University of Washington, Seattle, Washington.

Gales, L. and L. Anderson. 1978. User's guide for FFORM: a format free input system. Center for Quantitative Science in Forestry, Fisheries and Wildlife, University of Washington, Seattle, Washington.

Gales, L. 1978. User's guide for subroutine PLOT3D. Center for Quantitative Science in Forestry, Fisheries, and Wildlife, University of Washington, Seattle, Washington.