37 copies
25 to NTIS
8/22/80

MASTER

UCID-18757

Contour-Plotting Programs for Printer and Calcomp-Plotter

Peter Moller

July 1980

Lawrence
Livermore
Laboratory

This is an internal report intended primarily for internal or limited external distribution. The opinions and conclusions stated are those of the author and may or may not be those of the Laboratory.

Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore Laboratory under Contract W-7405-Eng-48.

DISTRIBUTION OF THIS DOCUMENT

Contour Plotting Programs for Printer and Calcomp Plotter

Peter Moller

DISCLAIMER

This work was prepared as an account of work sponsored by an agency of the United States Government. Therefore, the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favor by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Abstract

Contour plotting programs for plotting contour diagrams on printers or Calcomp plotters are described. The subroutines also exist in versions that are useful for the special application of finding minima and saddlepoints of nuclear potential energy surfaces generated by the subroutine PETR3 of another program package. However, for the general user the most interesting aspect of the plotting package is probably the possibility of generating printer contour plots. The plotting of printer contour plots is a very fast and convenient way of displaying two-dimensional functions.

Introduction

The contour plotting program package to be described below has been developed by the author at the Department of Mathematical Physics in Lund in 1969 for regular, equidistant grids. The package was modified by the author, during his stay at Los Alamos Scientific Laboratory in 1973, to plot rectangular grids with non-equidistant spacings between gridpoints.

The plotting package consists of the following subroutines:

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED *ESB*

RIRA, RIPA, MUCKA, POLY1), CONS, F, PLOTLN2), PLODR2),
PLINTG2), PLDUP2), PLTRNG2), BNJB CD2), and NORMO3).

The subroutines RIRA, RIPA, MUCKA, CONS and F were written by the author, the other subroutines as indicated by the references. The subroutine NORMO is called from RIPA (as explained in more detail below) only if Calcomp contour plots are wanted. This facility has not been tested in Livermore. The subroutines PLOTLN, PLODR, PLINTG, PLTRNG, and BNJB LD are called from RIRA if plots of "fission barriers" corresponding to plotted nuclear potential-energy surfaces are desired. This is a rather specialized application. Therefore, if only a printer contour plotting code is desired the call to PLOTLN and the surrounding statements could be removed from RIRA to save core space.

PROGRAM PARAMETERS AND INPUT DATA

To plot contour maps the user calls one of the two subroutines RIPA or RIRA. For RIPA the call should be CALL RIPA (E, NK, NX, NY, EA, MR, IX, IY, XVEC, YVEC, NJE, NJET, NET, DIF, LU, IYTEXT, IXTEXT, ISCALE, WIDTHX, WIDY, IDEX, IDEY, IRQ, RAT).

The program plots contour maps of NK matrices of identical dimensions (NY, NX). They should be stored in the matrix E(NK, NY, NX), which must be specified with these dimensions in the calling program. EA is a matrix used internally by RIPA. It must be dimensioned EA(NY+1, 126) in the calling program. IX is the number of x-values you wish printed or plotted along the x-axis. IY is the number of y-values you wish printed

or plotted along the y-axis. IVEC is a vector storing the grid point x-values. YVEC is a vector storing the grid point y-values. Figure 1 indicates how the input data should be stored in E, XVEC and YVEC. The x and y spacings need not be equidistant. In the plot that comes out on the printer or Calcomp, the X and Y-axis are marked at equidistant values however. This is indicated in figure 1.

If a contour plot on the printer is desired then NJE should be a multiple of 5. If a contour plot on the Calcomp plotter is desired then NJE should contain the factor 3. For each contour plot at least one data card is needed. If NJE is a multiple of 2 the text is col. 1-20 of the data card will come on one line and the text in col. 21-40 on the next line in the lower right hand corner of the plot. The scale will come on a third line. The width of a blank area or one area of identical characters (called stripe below) is equal to the number printed after the word scale on this third line. On the Calcomp the difference between function values corresponding to neighboring lines is equal to this number. On the printer the contour lines are plotted relative to the minimum value of the plot. On the Calcomp plotter the contour lines are plotted relative to zero. If NJE is a multiple of three then all text will come on one line above the plot. If NJE is a multiple of 7 which means that the Calcomp plot option has been chosen, an additional 3 data cards are needed for each plot. The card format should be FORMAT (I3, 7A8). The integer specifies how many alphanumeric characters that will follow. The text on the first card will come on top of the plot and will constitute a heading. The second card contains the x-axis text and the third card contains the y-axis text. The height of the text will

automatically be proportional to the length of the plot in the x direction. "HEIGHT" = $0.4 * WIDTX/20.0$ (inches). This usually corresponds to size standards for publications in journals.

NJET: If NJET is less than or equal to 1 then the scale will be adjusted to make the plot consist of NET "stripes". If NJET = 2 the scale of the plot will be equal to DIF. If NJET is greater than 2 then the scale will be set equal to one of the numbers 0.5, 1.0, 2.5, 5.0, 10.0, 25.0, This number is chosen so as to make the plot consist of between 15 and 30 "stripes".

LU is the logical number of the output unit for the alphanumeric plot. IYTEXT is the text at the printer plot y-axis (IYTEXT = 6HXXXXXX). IXTEXT is the text at the printer x-axis. ISCALE lets the programmer give the unit in which the data is expressed e.g. ISCALE = 5H MEV . This unit is then printed by the printer at each plot. The number of alphanumeric characters stored in IXTEXT, IYTEXT and ISCALE must be exactly 6, 6 and 5 respectively. WIDTX is the width of the plot in the x-direction in inches, not counting the text at the axis.

WIDY is the width of the plot in the y-direction in inches not counting the text at the axis. The printer plot will be the maximum size possible for a one page printout with a ratio between the length in the y-direction and the length in the x-direction equal to WIDY/WIDTX. It has been assumed that the height to width ratio on the printer of a character position is 1.66.

IDEX is the number of decimals for the numbers plotted by the Calcomp plotter at the x-axis. On the printer two decimals are always printed.

IDEX is the number of decimals for the numbers plotted at the y-axis. On the printer two decimals are always printed. If IRQ = 3 then "tic marks" are plotted at each x and y-value along the axis. If IRQ = 2 then full lines are drawn across the plot for each x and y-value. RAT is the width to height ratio for the alphanumeric characters on the plotter. It is 6.0/7.0 for the Calcomp plotter.

There exists another version of this program with slightly different input parameters and name, namely CALL RIRA (E, NK, NX, NY, EA, MR, IX, IY, XVEC, YVEC, NJE, NJET, NET, DIF, LU, GLARMM, I9, IYTEXT, IXTTEXT, ISCALE, NC, ND). This subroutine is the most convenient one to use if only printer plots are desired, since the size of the printer plot may be determined by the programmer through the parameters NC and ND. Below, only the parameters that are different from those of RIPA will be discussed.

RIRA does not plot on the Calcomp plotter. It has another feature however. It can, for each x-value, find the minimum in the contour diagram with respect to y. If NJE is a multiple of 5 these minima will be marked by * in the contour plot. If NJE is a multiple of 7 the values of these minima will be plotted as a function of x on the page following the contour plot. The largest value on the vertical axis will be equal to GLARMM. If GLARMM is too small, the program will choose a new value for GLARMM in the same way as if NJE is a multiple of 11. If

NJE is a multiple of 11 the values of the minima discussed above will also be plotted. But in this case the largest value on the vertical axis will be the smallest possible multiple of 5.

The parameter I9 allows for a change of scale. If the number of stripes turns out to be larger than I9 then the scale will be multiplied by 2^{NN} where NN is the smallest integer that will give a contour plot with less than I9 "stripes".

NC is the number of positions on the printer the plot is to occupy in the x-direction (frames and text at axis are not counted).

ND is the number of positions the plot is to occupy in the y-direction (frames and text at axis are not counted).

Limits on parameters:

$3 < NX < 29$ (1), $3 < NY < 29$ (2),

$NC < 121$ (3) and $ND < 55$ (4)

Recommendation: $IX < 15$ and $IY < 15$.

There exists at the Department of Mathematical Physics in Lund a version of the program where restrictions (1) and (2) do not apply.

PROGRAM INTERPOLATION METHODS

The contour plot exists as a matrix E1(ND,NC) in RIRA (It has been

declared DIMENSION E1(55,125)). Given the input matrix E(K,NY,NX) the matrix E1(ND,NC) is generated in the following way. The input matrix E is for each K transferred to the matrix A(30,30). Also the X and Y-values are transferred to A. How these variables are stored in A may be seen in figure 2.

By use of the subroutine MUCKA a matrix E1(ND,NC) that corresponds to a denser (or less dense) set of gridpoints is generated. First a polynomial of degree 3 is passed through A(2,2), A(2,3), A(2,4) and A(2,5). The corresponding x-values are XVEC(1), XVEC(2), XVEC(3), and XVEC(4). The coefficients are found by POLY and stored in the vector P.

The polynomial is then evaluated for the points in the denser grid corresponding to x-values in the range XVEC(1)-XVEC(3). The values of the polynomial at these points are stored in EA. A new polynomial is passed through A(2,3), A(2,4), A(2,5) and A(2,6). It is evaluated for the interval XVEC(3)-XVEC(4). The whole A matrix is covered in this manner and the generated values are stored in a matrix EA(NY+1, NC+1) which now obviously has a denser grid in the x-direction than the original E matrix (if NC > NX). Now polynomials in the y-direction are generated and the function values corresponding to the printer plot character positions are calculated. The results are stored in E1(ND,NC). Everywhere, except at the matrix end points, the generated polynomial is evaluated only between the two middle points. One would call the interpolation method, "Poor man's spline interpolation" as overlapping polynomials are used but end point derivatives are not matched. Much use indicates that the accuracy of this interpolation is excellent (and in particular sufficiently accurate for use in a plotting program.)

It can be proven⁴⁾ that the above prescription for finding the denser grid is equivalent to finding the following polynomial

$$\begin{aligned} f(x,y) = & y^3 * (AX^3 + BX^2 + CX + D) \\ & + y^2 * (EX^3 + FX^2 + GX + H) \\ & + y * (IX^3 + JX^2 + KX + L) \\ & + 8 (MX^3 + NX^2 + OX + P) \end{aligned}$$

by use of the 16 gridpoints that are closest to the point for which an interpolated value is desired. As an example consider the 6 by 10 matrix in figure 3. There the input grid points are indicated by dots. New points are desired in the cross-hatched area. Then the circled points are the ones that will be effectively used by the program code to generate the coefficients of the polynomial f . When the Calcomp contour plots are generated the program first generates a 40 by 40 matrix from the input matrix by use of the above interpolation method. The subroutine NORMD then generates and plots the Calcomp contour lines by linear interpolation in this grid. At the Department of Mathematical Physics in Lund there exists a version of RIPA which allows a matrix of any prescribed dimensions to be generated in the first step. The input matrix may in this version also be larger than 29 by 29. These two extensions of the code are only of interest for special applications and have not been made a feature of the standard version, since they increase memory requirements and the number of input parameters.

When moving the codes from one computer to another one should note that a few statements have to be changed if the wordlength of the new computer is different from that of the old computer.

The data and dimension statements in RIRA and RIPA for the vectors V and VV, which store variable format specifications may have to be changed. If this is done also a few statements of the type $VV(2) = \dots$, $V(2) = \dots$ may have to be changed to $VV(3) = \dots$ and $V(3) = \dots$ if, for example, the data that was stored in V(1) and VV(1) on the old computer is stored in two computer words (V(1), V(2) and VV(1), VV(2) respectively) on the new computer. With a knowledge of how variable formats work, it is a straightforward matter to make these changes. At the end of NORMO the FORMAT and READ statements may have to be changed so that the proper number of characters are read into each computer word of the vector INA.

For the special application of finding saddle points and minima of plotted (nuclear potential energy) surfaces the subroutines RIRASYM and RIRAASYM should be used. These will print saddle point function and coordinate values on logical file 10. These values may then be used for tabulation or for input in neutron cross-section calculations or r-process calculations. The author should be contacted if these versions are used since some cards may have to be changed for some applications. One should always check for each surface that the code did find the saddle points. This can be done just by looking at the plotted surface and at the printout, on the page before the surface, of the saddle point values. This is particularly important for RIRAASYM which is used for mass-asymmetric potential energy surfaces. It has recently been modified but may still fail on occasion. Usually it will find too many saddle points. RIRAASYM should always be called with $ND = 49$ and the input energies in $E(K,ND,NC)$ should be symmetric with respect to $(YVEC(1) +$

$YVEC(ND))/2.$

A third specialized version of the plotting program R/RA is RIRAGAM, which plots triangular plots used for displaying results of calculations corresponding to axially asymmetric shapes of the nucleus. RIRAGAM also searches for saddle points and minima.

The time for generating one printer plot is between 1 and 2 seconds on a CDC 6600. A Calcomp plot takes about 5 seconds to generate on the CDC 6600.

REFERENCES

- 1) B. Gimle, private communication (1967).
- 2) Code written by DeLos S. Detator, Dept. Chemistry: Inst. Molecular Biophysics, Florida State University, Tallahassee, Florida.
- 3) B. Rystedt, private communication (1972); S. Nordbeck, Normap manual (Student literature, Lund 1973).
- 4) I. Raynarsson, private communication (1977).

FIGURE CAPTIONS

- 1) Relation between the RIPA and RIRA input parameters E, XVEC, YVEC, IX, IY.
- 2) Storage of variables in the matrix A(30,30) in RIPA and RIRA.
- 3) Illustration of interpolation method used in RIPA and RIRA contour plotting programs. To generate interpolated points in the cross-hatched area, the sixteen circled points are used for effectively generating the polynomial $f(xy)$ discussed in the text.

Storage of input data in program code

with $IX = 5$, $IY = 4$,

$$\left\{ \begin{array}{l} XVEC(1) = 0.0 \\ XVEC(2) = 4.0 \\ XVEC(3) = 8.0 \\ XVEC(4) = 12.0 \\ XVEC(5) = 14.0 \\ XVEC(6) = 16.0 \end{array} \right. \quad \text{and} \quad \left\{ \begin{array}{l} YVEC(1) = 0.0 \\ YVEC(2) = 3.0 \\ YVEC(3) = 6.0 \\ YVEC(4) = 9.0 \end{array} \right.$$

the axis labelling will be as indicated.

Figure 1

Storage of input data in program code

X are grid coordinates in x-direction

Y are grid coordinates in y-direction

Z are function values $Z = Z(x,y)$

Figure 2

Interpolation procedure

Contours in the cross-hatched area are based on the function values of the sixteen surrounding encircled points. Specifically these points determine a function f of the following type:

$$\begin{aligned}
 f = & x^3(Ay^3 + By^2 + Cy + D) + \\
 & + x^2(Ey^3 + Fy^2 + Gy + H) + \\
 & + x(Iy^3 + Jy^2 + Ky + L) + \\
 & + (My^3 + Ny^2 + Oy + P)
 \end{aligned}$$

Figure 3