UNITED STATES DEPARTMENT OF THE INTERIOR

GEOLOGICAL SURVEY

COMPUTER PROGRAMS FOR SLOPE PROFILE ANALYSIS

By

Joseph R. Davis

Open-File Report 78-907

1978

CONTENTS

Figures

Tables

INTRODUCTION

Although slope profile analysis is an established technique in physical geography, it has several potentially important applications in geologic studies which are only beginning to be recognized. As used in geographic studies, a slope profile is a line surveyed across the ground surface for which the angles of short lengths are measured (fig. 1). Slope profile analysis is the division of the profile into a number of parts, each of which possesses certain properties of form; the terminology used in discussing the properties of form in this study is similar to that used by Young (1975, p. 149):

Slope unit--a segment or an element.

Segment--a portion of a slope profile on which the angle remains approximately constant.

Element--a portion of a slope profile on which the curvature remains approximately constant.

Curvature--the rate of change of slope angle with distance downslope, expressed in degrees per 100 meters.

Convexity--positive curvature; downslope increase in angle.

Concavity--negative curvature; downslope decrease in angle.

A limitation in slope profile analysis is the subjective nature of deciding whether a convexity or concavity should be considered a smoothly curved element or a series of rectilinear segments. Strahler (1950), the first to study this problem, used statistical techniques to analyze the frequency distribution of the various possible units of the slope profile. More recently, Young (1971) suggested use of the coefficients of variation of slope angle (V_a) and slope curvature (V_c) to objectively determine slope breaks; table 1 lists the formulae necessary for computing these standard statistical coefficients.

Variability of slope angle does not have uniform significance. The effect of a change in slope angle upon geologic processes is invariably more critical for small slope angles than for large slope angles. Therefore, the coefficient of variation, which measures percentage variation, rather than the standard deviation, which measures relative variation, is employed to determine critical slope breaks. This is the same reasoning employed by many engineers, who use percentage grade rather than slope angle in many hydraulic equations.

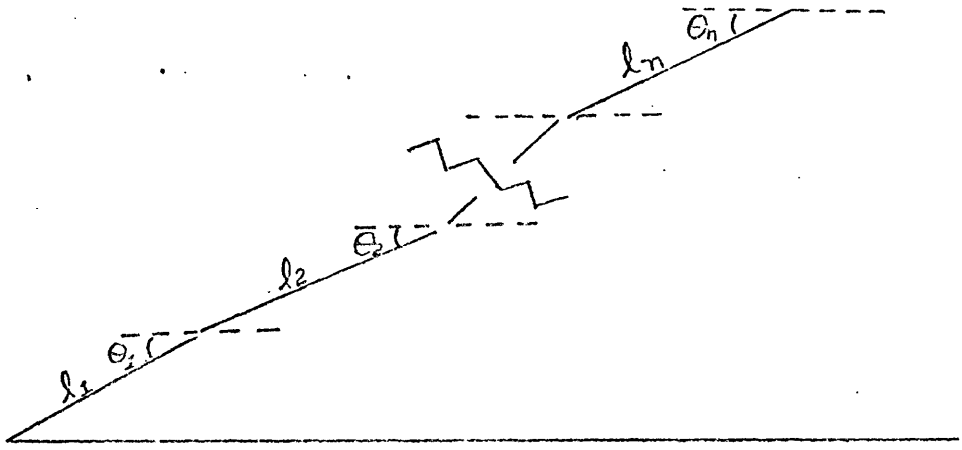

Figure 1.--Generalized slope profile showing symbols used in table 1.

Table 1.--Formulae Used[1]

Parameter	Formula
Curvature	$C_1 = 200 \times \frac{\theta_1 - \theta_2}{\ell_1 - \ell_2}$
Curvature	$C_n = 200 \times \frac{\theta_{n-1} - \theta_n}{\ell_{n-1} + \ell_n}$
Curvature[2]	$C_i = 200 \times \frac{\theta_{i-1} - \theta_{i+1}}{\ell_{i-1} + 2\ell_i + \ell_{i+1}}$
Mean angle	$\bar{\theta} = \frac{\Sigma \ell \theta}{\Sigma \ell}$
Mean curvature	$\bar{C} = \frac{\Sigma \ell C}{\Sigma \ell}$
Coefficient of variation of slope angle[3]	$V_a = 100((((\Sigma \ell \theta^2)/\Sigma \ell) - \bar{\theta}^2)\frac{1}{2}/\bar{\theta})$
Coefficient of variation of slope curvature[4]	$V_c = 100((((\Sigma \ell C^2)/\Sigma \ell) - \bar{C}^2)\frac{1}{2}/\bar{C})$

[1] From Young (1975).

[2] Used for lengths l_i, $i = 2...n-1$.

[3] if $\bar{\theta}<2$, it is replaced by 2 in the denominator.

[4] if $\bar{C}<2$, it is replaced by 2 in the denominator.

COMPUTER PROGRAMS FOR PROFILE ANALYSIS

Young (1971) designed a computer program that calculates V_a and V_c for each possible combination of contiguous measured lengths. The profile is subdivided into a set of best segments and a set of best elements such that V_a and V_c do not exceed specified maximum values; where two or more segments or elements overlap, the overlapping portion is allocated to the longest unit. Finally, the profile is subdivided into a set of best slope units, which are determined by calculating both best segments and best elements and choosing the units that have the lowest coefficients of variation.

The computer programs SLOPES and FANSEG, modified from the program written by Young (1971), allow interactive use on the USGS Honeywell Multics[1/] operating

---

[1/] Use of brand names in this report is for descriptive purposes only and does not constitute endorsement by the U.S. Geological Survey.

---

system and permit graphical presentation of the data. The program SLOPES is similar to the original program of Young (1971) in that it can be used to analyze profiles for any data obtained from field surveys. In addition to hillslope studies, this program may be useful in analyzing profiles of fault scarps and, thus, be helpful in determining the age of the faulting (Wallace, 1977). The program FANSEG computes profile analyses from topographic map data by calculating a slope angle for measured distances between contour lines. The utility of FANSEG is largely limited by the map scale and contour interval, and is probably most useful for profile studies of stream gradients, alluvial-fan segmentation, and other large-scale landform studies.

COMPUTER PROGRAMS SLOPES AND SLOPE_DATA

A sample data set for SLOPES is listed in table 2; the data file for SLOPES is prepared by using the interactive program SLOPE_DATA. Profile data should begin at the base of a slope and proceed upslope. The measured lengths should be recorded in meters and slope angles, in degrees. Suitable values for the maximum coefficients of variation are discussed in Young (1975, p. 150-151). Profile analysis of the sample data set by the program SLOPES is shown in table 3 and figures 2-4. The figures are drawn with no vertical exaggeration.

Table 2.--Sample Data Set for SLOPES

Title: Test data for slopes
Maximum V_a = 10
Maximum V_c = 25
nmax= 10

n	Angle (degrees)	Distance (meters)
1	3.5	5
2	7.0	20
3	10.0	20
4	15.0	20
5	15.5	20
6	16.0	20
7	16.5	20
8	11.0	20
9	8.0	20
10	5.0	20

Table 3.--Sample Output from SLOPES

[n--number of an individual measured length; di--length "n" (meters); a--slope angle of length "n" (degrees); c--curvature of length "n" (degrees/100 meters); unit--number of the segment or element to which length "n" is assigned; numb--the number of lengths in the unit; angle--the mean angle of the segment (degrees); curv--the mean curvature of the element (degrees/100 meters); cvar--the coefficient of variation of the unit (percent)]

test data for slopes
Best Rectilinear Segments
Maximum Coefficient of Variation= 10.00

n	di	a	c	unit	numb	angle	curv	cvar
1	5.0	3.50	-28.00	1	1	3.5000	0.0000	0.0000
2	20.0	7.00	-20.00	2	1	7.0000	0.0000	0.0000
3	20.0	10.00	-20.00	3	1	10.0000	0.0000	0.0000
4	20.0	15.00	-13.75	4	4	15.7500	0.0000	3.5493
5	20.0	15.50	-2.50	4	4	15.7500	0.0000	3.5493
6	20.0	16.00	-2.50	4	4	15.7500	0.0000	3.5493
7	20.0	16.50	12.50	4	4	15.7500	0.0000	3.5493
8	20.0	11.00	21.25	5	1	11.0000	0.0000	0.0000
9	20.0	8.00	15.00	6	1	8.0000	0.0000	0.0000
10	20.0	5.00	15.00	7	1	5.0000	0.0000	0.0000

test data for slopes
best curved elements

Maximum Coefficient of Variation= 25.00

n	di	a	c	unit	numb	angle	curv	cvar
1	5.0	3.50	-28.00	1	4	0.0000	-18.6923	20.8795
2	20.0	7.00	-20.00	1	4	0.0000	-18.6923	20.8795
3	20.0	10.00	-20.00	1	4	0.0000	-18.6923	20.8795
4	20.0	15.00	-13.75	1	4	0.0000	-18.6923	20.8795
5	20.0	15.50	-2.50	2	2	0.0000	-2.5000	0.0000
6	20.0	16.00	-2.50	2	2	0.0000	-2.5000	0.0000
7	20.0	16.50	12.50	3	1	0.0000	12.5000	0.0000
8	20.0	11.00	21.25	4	3	0.0000	17.0833	17.2465
9	20.0	8.00	15.00	4	3	0.0000	17.0833	17.2465
10	20.0	5.00	15.00	4	3	0.0000	17.0833	17.2465

test data for slopes
best slope units

maximum coefficient of variation, segments= 10.00 elements= 25.00

n	di	a	c	unit	numb	angle	curv	cvar
1	5.0	3.50	-28.00	1	3	0.0000	-20.8889	12.0359
2	20.0	7.00	-20.00	1	3	0.0000	-20.8889	12.0359
3	20.0	10.00	-20.00	1	3	0.0000	-20.8889	12.0359
4	20.0	15.00	-13.75	2	4	15.7500	0.0000	3.5493
5	20.0	15.50	-2.50	2	4	15.7500	0.0000	3.5493
6	20.0	16.00	-2.50	2	4	15.7500	0.0000	3.5493
7	20.0	16.50	12.50	2	4	15.7500	0.0000	3.5493
8	20.0	11.00	21.25	3	3	0.0000	17.0833	17.2465
9	20.0	8.00	15.00	3	3	0.0000	17.0833	17.2465
10	20.0	5.00	15.00	3	3	0.0000	17.0833	17.2465

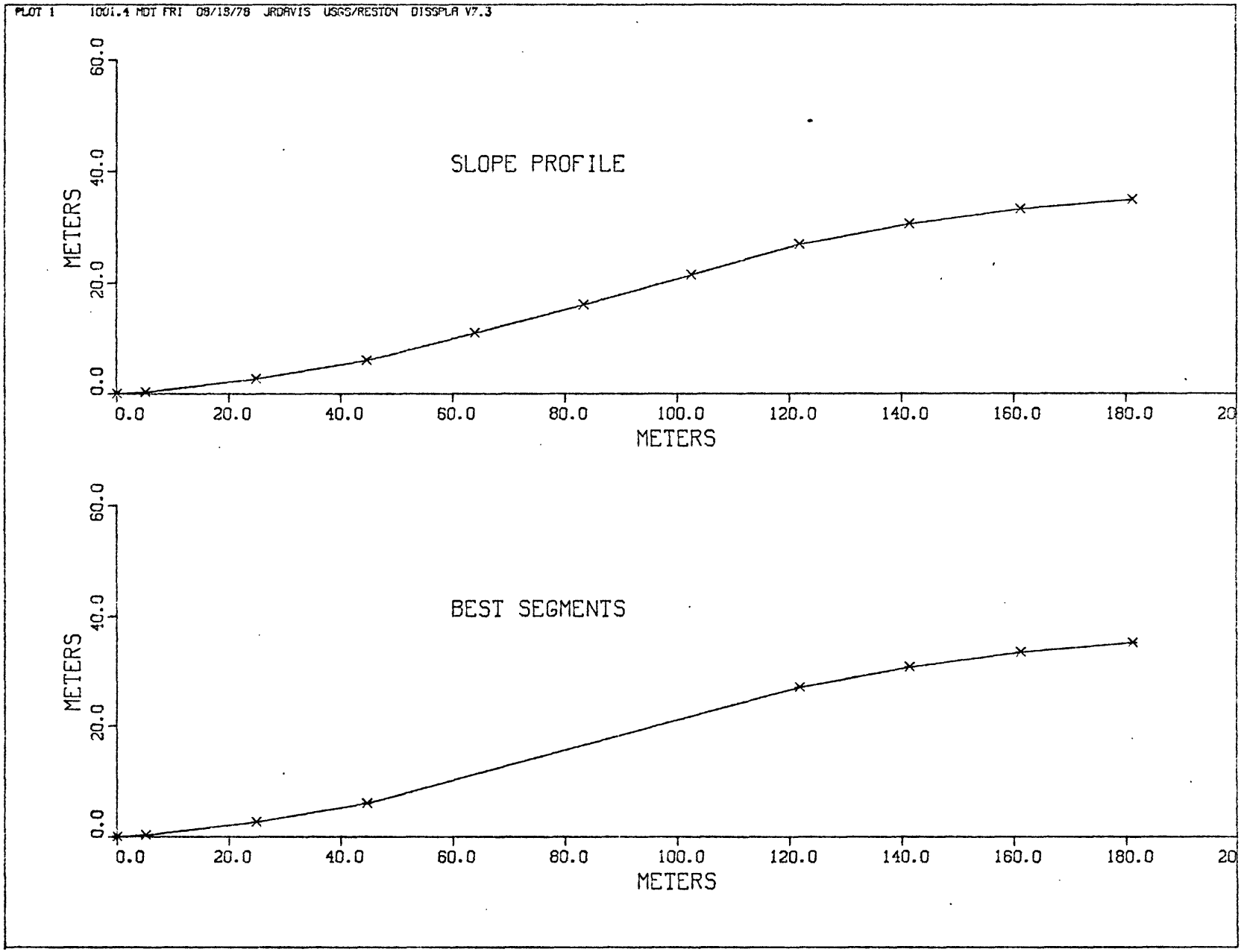

Figure 2.--Calcomp output of Best Segments analysis of sample data, program SLOPES.

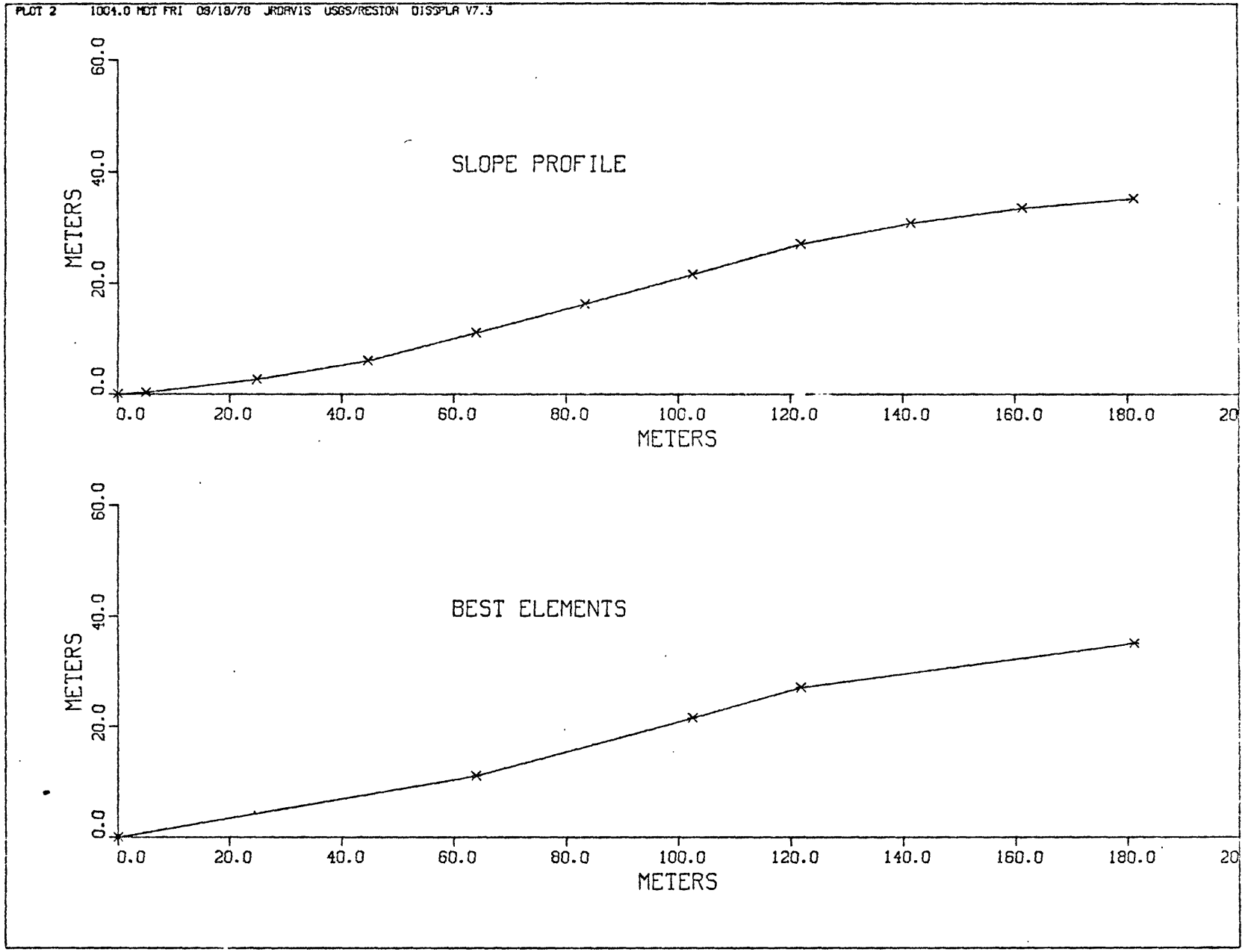

Figure 3.--Calcomp output of Best Elements analysis of sample data, program SLOPES.

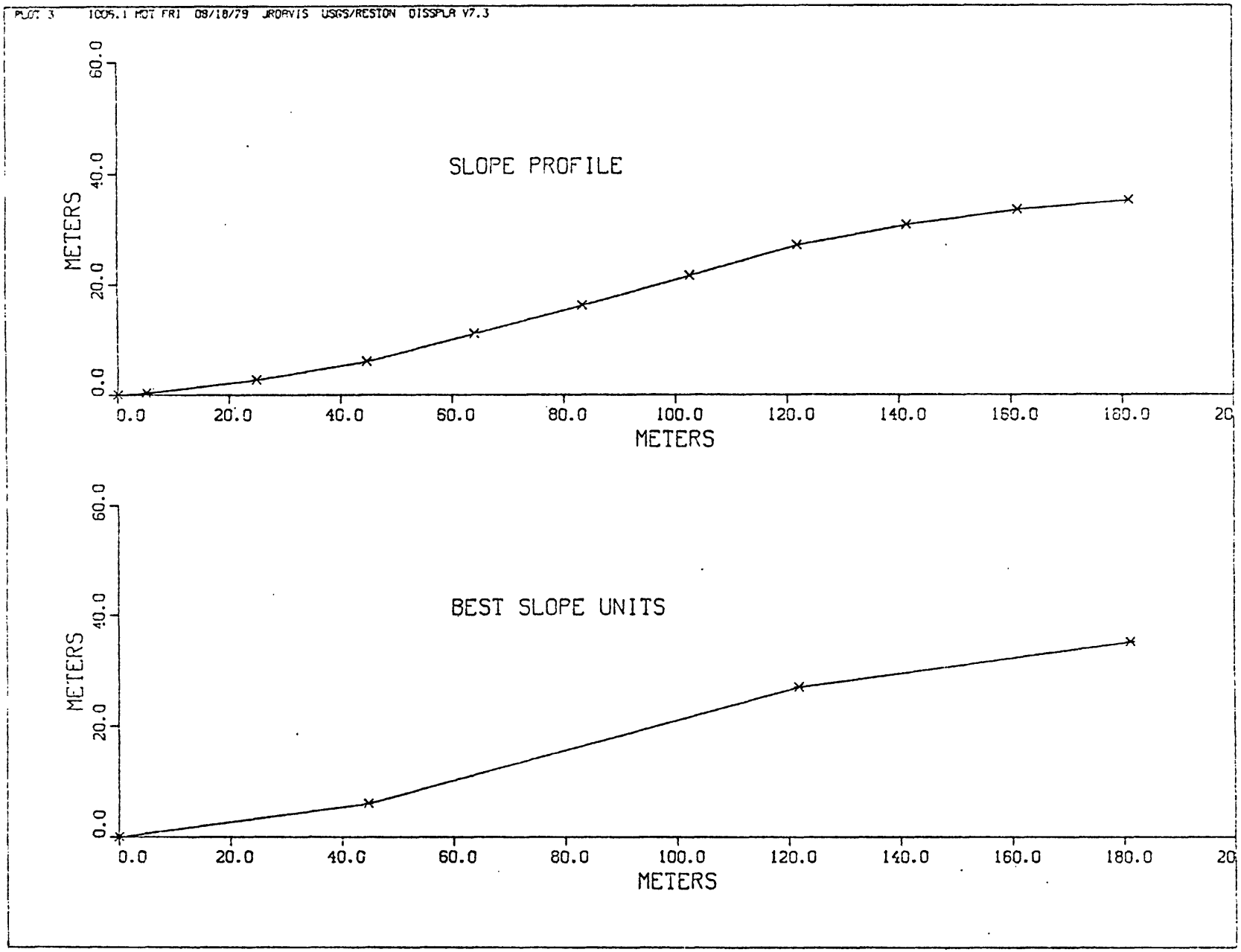

Figure 4.--Calcomp output of Best Slope Units analysis of sample data, program SLOPES

COMPUTER PROGRAMS FANSEG AND FANSEG_DATA

A sample data set for FANSEG is listed in table 4; the data file for FANSEG is prepared by using the interactive program FANSEG_DATA. Profile data should begin at the base of a slope and proceed upslope. Because most topographic maps in the United States are still published with contour intervals in English units, both the measured lengths and the contour intervals should be recorded in feet. However, for consistency with published data, curvature is still calculated by the program in units of degrees per 100 meters. Profile analysis of the sample data set by FANSEG is shown in table 5 and figures 5-7. The column headings are the same as those for output from SLOPES, except that "mpdi" is the length, in feet, of a measured distance between two contour lines. The graphical output from FANSEG has 10X vertical exaggeration.

EXECUTING THE PROGRAMS ON MULTICS

For USGS Multics users to execute one of the programs, it is necessary to first establish a link by typing the appropriate command:

```
lk > udd > Lithium > JRDavis > profile_dir > slopes
```

or

```
lk > udd > Lithium > JRDavis > profile_dir > slope_data
```

or

```
lk > udd > Lithium > JRDavis > profile_dir > fanseg
```

or

```
lk > udd > Lithium > JRDavis > profile_dir > fanseg_data
```

The user can then create a data file by executing either SLOPE_DATA or FANSEG_DATA by typing the appropriate program name in lower case letters. These programs allow the user to enter and edit the profile data required by the main programs SLOPES and FANSEG; they then ask the user for a file name; this file name is the segment name where the program output will be stored for use as input by the main programs.

After creating a data file with either SLOPE_DATA or FANSEG_DATA, it is necessary to issue two commands concerning the graphical output prior to running either SLOPES or FANSEG. First, since ISSCO Disspla System Graphics commands were used in the program, it is necessary to add the Disspla package to the user's search rules by typing:

```
asr > iml > disspla -after working_dir
```

Table 4.--Sample Data Set for FANSEG

Title: Test data for fanseg
Maximum V_a= 10
Maximum V_c= 25
nmax= 10

n	Contour interval (feet)	Map distance (feet)
1	10	650
2	10	600
3	10	400
4	20	300
5	20	250
6	20	250
7	20	300
8	20	400
9	20	500
10	20	600

Table 5.--Sample Output from FANSEG

[n--number of an individual measured length; mpdi--length, in feet, of a measured distance between two contour lines; a--slope angle of length "n" (degrees); c--curvature of length "n" (degrees/100 meters); unit--number of the segment or element to which length "n" is assigned; nump--the number of lengths in the unit; angle--the mean angle of the segment (degrees); curv--the mean curvature of the element (degrees/100 meters); cvar--the coefficient of variation of the unit (percent)]

```
test data for fanseg
Best Rectilinear Segments
 Maximum Coefficient of Variation=  10.00
```

n	mpdi	a	c	unit	nump	angle	curv	cvar
1	650.0	0.88	-0.04	1	2	0.9167	0.0000	1.8345
2	600.0	0.95	-0.16	1	2	0.9167	0.0000	1.8345
3	400.0	1.43	-1.10	2	1	1.4321	0.0000	0.0000
4	300.0	3.81	-1.65	3	4	4.1595	0.0000	9.0961
5	250.0	4.57	-0.47	3	4	4.1595	0.0000	9.0961
6	250.0	4.57	0.47	3	4	4.1595	0.0000	9.0961
7	300.0	3.81	0.90	3	4	4.1595	0.0000	9.0961
8	400.0	2.86	0.62	4	1	2.8624	0.0000	0.0000
9	500.0	2.29	0.31	5	2	2.0825	0.0000	9.1205
10	600.0	1.91	0.23	5	2	2.0825	0.0000	9.1205

```
test data for fanseg
  best curved elements

 Maximum Coefficient of Variation=  25.00
```

n	mpdi	a	c	unit	nump	angle	curv	cvar
1	650.0	0.88	-0.04	1	3	0.0000	-0.3411	21.7289
2	600.0	0.95	-0.16	1	3	0.0000	-0.3411	21.7289
3	400.0	1.43	-1.10	1	3	0.0000	-0.3411	21.7289
4	300.0	3.81	-1.65	2	1	0.0000	-1.6493	0.0000
5	250.0	4.57	-0.47	3	1	0.0000	-0.4748	0.0000
6	250.0	4.57	0.47	4	5	0.0000	0.4542	11.6753
7	300.0	3.81	0.90	4	5	0.0000	0.4542	11.6753
8	400.0	2.86	0.62	4	5	0.0000	0.4542	11.6753
9	500.0	2.29	0.31	4	5	0.0000	0.4542	11.6753
10	600.0	1.91	0.23	4	5	0.0000	0.4542	11.6753

```
test data for fanseg
  best slope units

  maximum coefficient of variation, segments=  10.00      elements=  25.00
```

n	mpdi	a	c	unit	nump	angle	curv	cvar
1	650.0	0.88	-0.04	1	3	0.0000	-0.3411	21.7289
2	600.0	0.95	-0.16	1	3	0.0000	-0.3411	21.7289
3	400.0	1.43	-1.10	1	3	0.0000	-0.3411	21.7289
4	300.0	3.81	-1.65	2	2	4.1595	0.0000	9.0961
5	250.0	4.57	-0.47	2	2	4.1595	0.0000	9.0961
6	250.0	4.57	0.47	3	5	0.0000	0.4542	11.6753
7	300.0	3.81	0.90	3	5	0.0000	0.4542	11.6753
8	400.0	2.86	0.62	3	5	0.0000	0.4542	11.6753
9	500.0	2.29	0.31	3	5	0.0000	0.4542	11.6753
10	600.0	1.91	0.23	3	5	0.0000	0.4542	11.6753

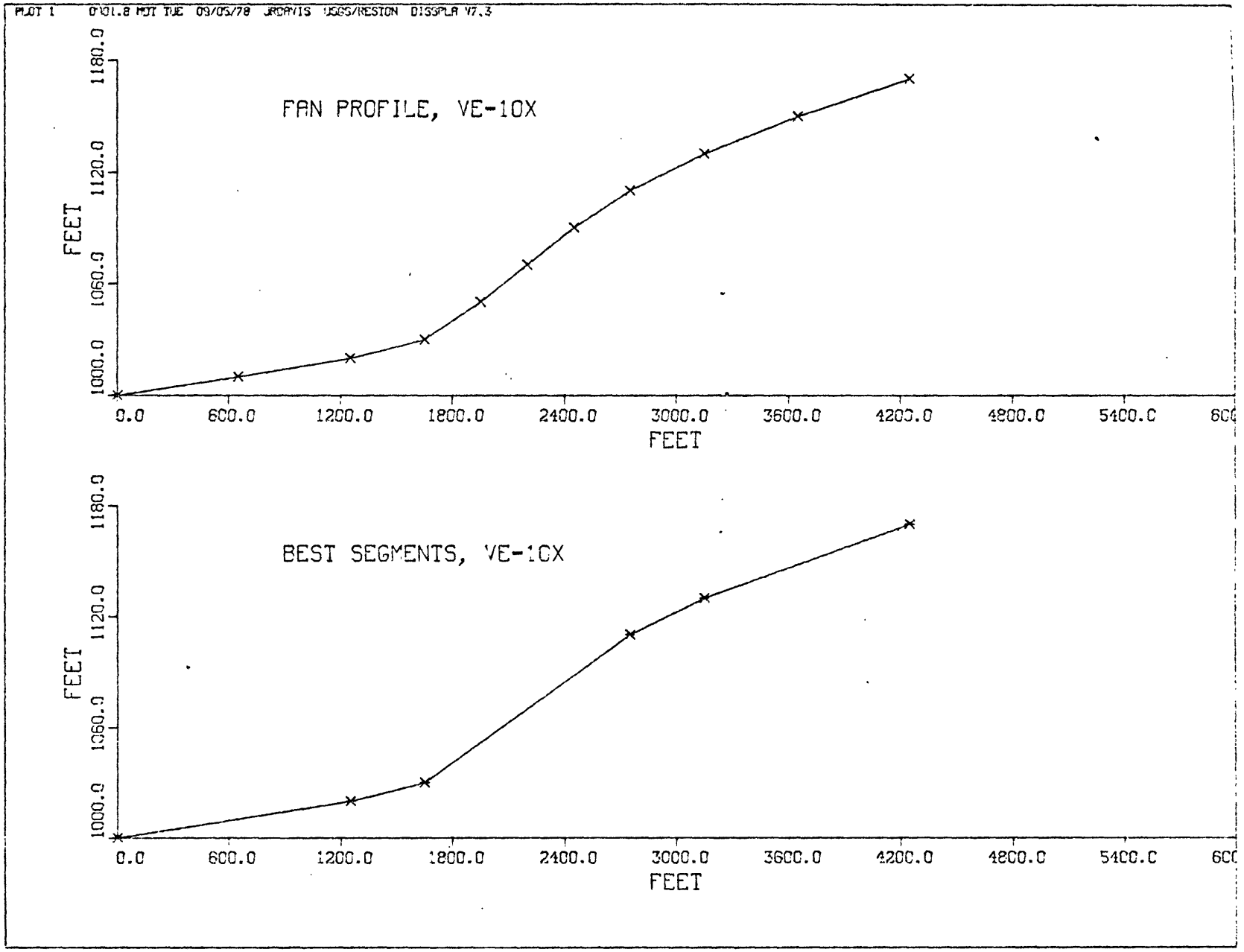

Figure 5.--Calcomp output of Best Segments analysis of sample data, program FANSEG

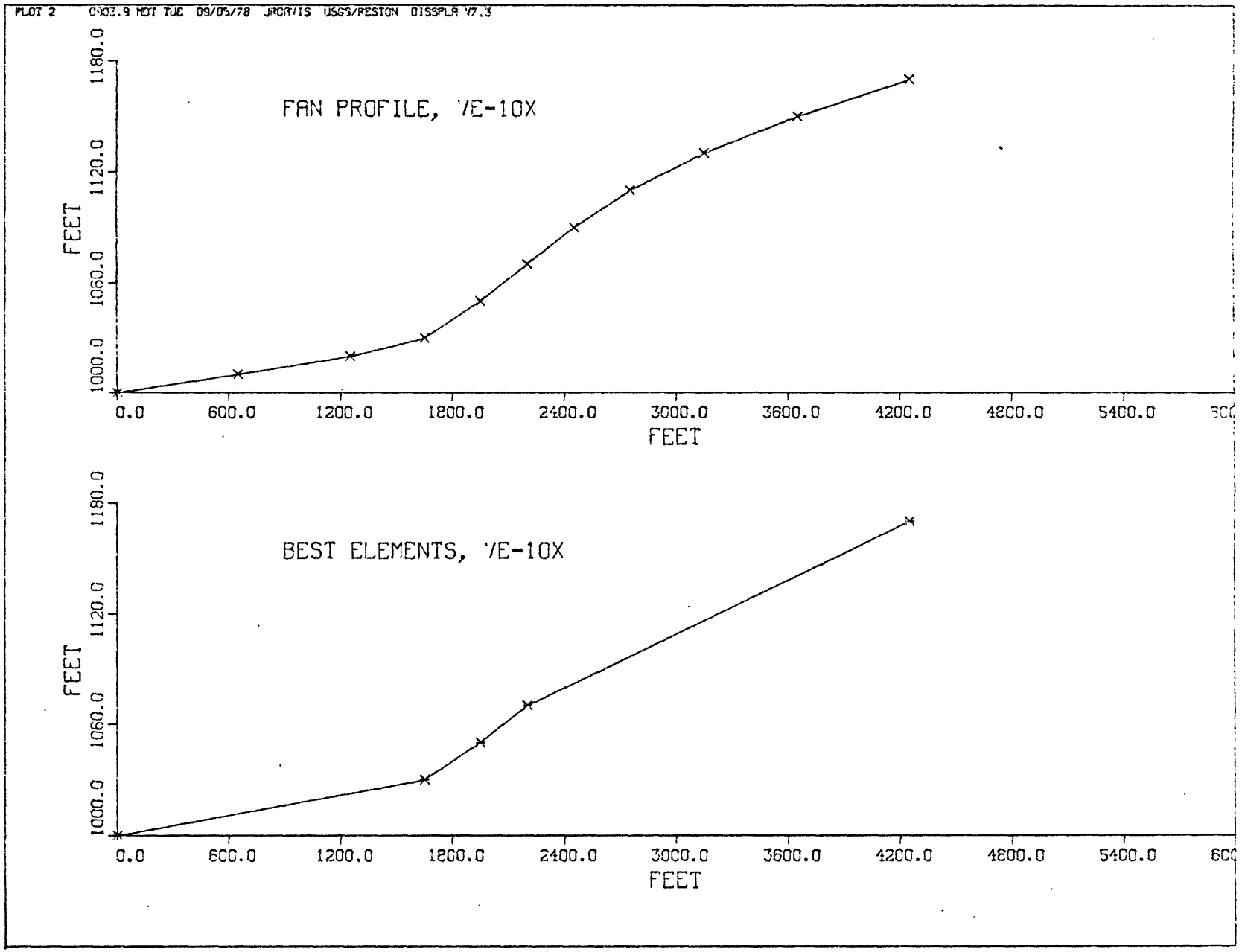

Figure 6.--Calcomp output of Best Elements analysis of sample data, program FANSEG

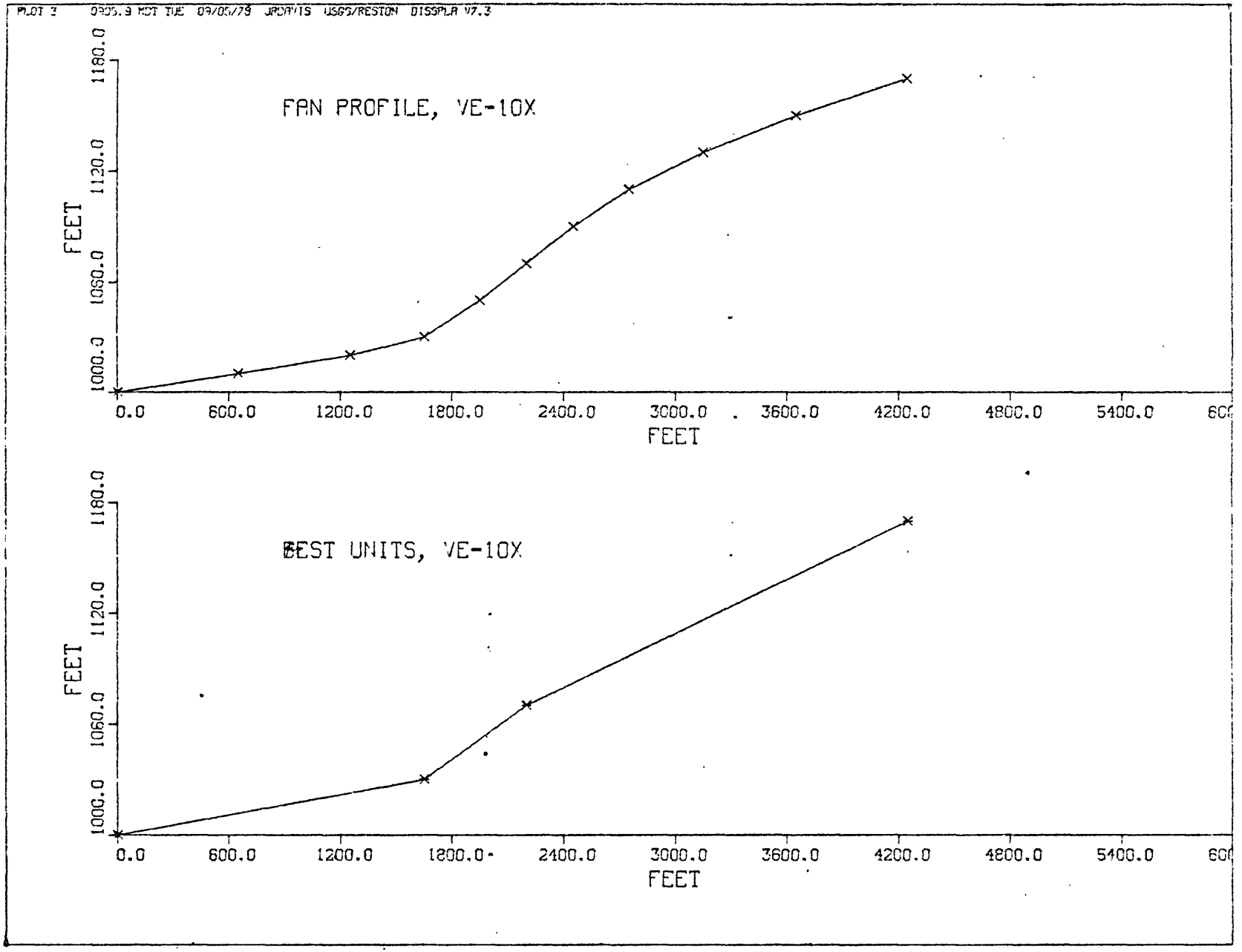

Figure 7.--Calcomp output of Best Slope Units analysis of sample data, program FANSEG

If a display terminal compatible with the USGS Tektronix software package is to be used, set the band rate at 1200 and type:

Setup_tektronix_tcs

Otherwise, a Calcomp plot can be obtained by typing:

setup_calcomp

The system operator will then respond with a message notifying the user of the tape number being used, which is necessary for the "poi" plot request. The main programs can then be run by typing "slopes" or "fanseg." The programs ask the user for the name of the data file created by either SLOPE_DATA or FANSEG_DATA and then ask for the name of an output file; this should be the segment name that identifies where the output for the printed tables will be stored. The program FANSEG also asks for the elevation of the starting point at the base of the slope, which is used to label the vertical scale of the graphical output. If only relative distances are needed, zero can be entered for this parameter.

The tabular output consists of three parts: best segments, best elements, and best slope units (tables 3 and 5). In addition to having this data stored in an output segment, the user has the option of having each part displayed at the terminal and of creating plotting vectors for each part. If Calcomp plots are desired, the user submits a request to have the data plotted by typing: poi. This command executes an interactive program which allows the user to submit the plot tape for plotting.

SOURCE CODE FOR SLOPES

The source code for SLOPES is listed below. In addition to the variables discussed previously as column headings of the output, the following variables are used:

cvmxa - maximum coefficient of variation of angle
cvmxc - maximum coefficient of variation of curvature
cvar - coefficient of variation of a length calculated by subroutine values
nmax - total number of lengths measured
xdis - array for the horizontal component of each measured length
ydis - array for the vertical component of each measured length
xfaray - array for the horizontal location of each length
yfaray - array for the vertical location of each length
xaray - array for the horizontal location of each slope unit
yaray - array for the vertical location of each slope unit

```
c       program slopes
c
          dimension n(250),dist(250),a(250),nitest(250),ip(250),ymax(250),
          &nseg(250),nump(250),initest(250),curv(250),dd(250),da(250),amean(
          &250),cmean(250),set(250),acvar(250),di(250),unit(250),xmax(250),
          &xaray(250),yaray(250),xfaray(250),yfaray(250),tstaray(250),
          &xdis(250),ydis(250)
        character nameseg*80
        external bgnpl(descriptors),calcmp(descriptors),title(descriptors)
        external messag(descriptors),graph(descriptors),curve(descriptors)
        external marker(descriptors),physor(descriptors),endgr(descriptors)
        external endpl(descriptors),donepl(descriptors),tk120(descriptors)
        character nameout*80
        character titl*80
c
c       input
c
          open(0,prompt=.true.)
    2     continue
          write(0,4)
    4     format("What is the name of the data file?")
          read(0,6)nameseg
    6     format(a80)
          open(1,file=nameseg,form='formatted',mode='in')
          read(1,680)titl
          read(1,690)cvmxa
          read(1,700)cvmxc
   10     read(1,710,end=670)nmax
          if(nmax)20,670,20
   20     read(1,720)(n(i),a(i),di(i),i=1,nmax)
          close(1)
          write(0,23)
   23     format("What name for the output file?")
          read(0,24)nameout
   24     format(a80)
          open(20,file=nameout,form='formatted',mode='out')
c
c       compute plotting coordinates
c
          do 30 i=1,nmax
          ydis(i)=(sin(a(i)/59.2958))*di(i)
          xdis(i)=(cos(a(i)/59.2958))*di(i)
       30 continue
          ymax(1) = ydis(1)
          xmax(1) = xdis(1)
          do 25 i=2,nmax
          ymax(i)=ydis(i)+ymax(i-1)
          xmax(i)=xdis(i)+xmax(i-1)
```

```
   25    continue
         xfirst=0.0
         yfirst=0.0
         xfinal=xmax(nmax)
         xfaray(1)=xfirst
         yfaray(1)=yfirst
         do 40 i=2,nmax+1
         xfaray(i)=xfaray(i-1)+xcis(i-1)
         yfaray(i)=yfaray(i-1)+ycis(i-1)
   40    continue
c
c      compute curv(i)
c
         curv(1)=200*((a(1)-a(2))/(di(1)+di(2)))
         curv(nmax)=200*((a(nmax-1)-a(nmax))/(di(nmax-1)+di(nmax)))
         icount=nmax-1
         do50 i=2,icount
         curv(i)=200*((a(i-1)-a(i+1))/(di(i-1)+2*di(i)+di(i+1)))
   50    continue
   60    nrnd=1
         go to 90
   70    nrnd=2
         go to 90
   80    nrnd=3
   90    continue
c
c      set variables to zero
c
         indsg=1
         do100 i=1,nmax
         nitest(i)=0
         amean(i)=0.0
         cmean(i)=0.0
         dist(i)=0.0
         acvar(i)=1000.
  100    continue
c
c      obtain all combinations in turn
c
  110    do370 j=1,nmax
         if(nitest(j))370,120,370
  120    nclk=1
  130    jcount=nmax-j+1
         do320 k=1,jcount
         if(nrnd-2)150,170,140
  140    if(nclk-1)170,150,170
  150    do160 i=1,nmax
         set(i)=a(i)
```

```
  160   continue
        go to 190
  170   do180 i=1,nmax
        set(i)=curv(i)
        nclk=2
  180   continue
  190   continue
        do200 i=1,k
        initest(i)=nitest(i+j-1)
        dd(i)=di(i+j-1)
        da(i)=set(i+j-1)
  200   continue
        call values(initest,dd,da,k,intsm,sumd,vmean,cvar)
c     test parameters of combination
c
        if(intsm)330,210,330
  210   if(nclk-1)220,220,230
  220   cvrmx=cvmxa
        go to 240
  230   cvrmx=cvmxc
  240   if(cvar-cvrmx)250,250,330
  250   kcount=j+k-1
        do270 i=j,kcount
        if(sumd-dist(i))320,260,270
  260   if(cvar-acvar(i))270,320,320
  270   continue
c
c     allocate new values
c
        lcount=j+k-1
        do310 i=j,lcount
        nseg(i)=indsg
        nump(i)=k
        dist(i)=sumd
        acvar(i)=cvar
        if(nrnd-2)290,300,280
  280   if(nclk-1)290,290,300
  290   amean(i)=vmean
        cmean(i)=0.0
        go to 310
  300   cmean(i)=vmean
        amean(i)=0.0
  310   continue
        indsg=indsg+1
  320   continue
  330   if(nrnd-2)370,370,340
  340   if(nclk-1)370,350,370
  350   nclk=2
```

```
        go to 130
  360   continue
  370   continue
c
c     test to see if segments cut short
c
        do480 j=1,nmax
        if(j-1)380,390,380
  380   if(nseg(j)-nseg(j-1))390,480,390
  390   continue
        if(nitest(j)-1)400,480,400
  400   if(nseg(j)-nseg(j+nump(j)-1))410,460,410
  410   nn=1
  420   if(nseg(j)-nseg(j+nump(j)-1-nn))430,440,430
  430   nn=nn+1
        go to 420
  440   kk=nump(j)-nn
        mcount=j+kk-1
        do450 i=j,mcount
        nitest(i)=0
        amean(i)=0.0
        cmean(i)=0.0
        dist(i)=0.0
        acvar(i)=1000
  450   continue
        go to 480
  460   ncount=j+nump(j)-1
        do470 i=j,ncount
        nitest(i)=1
  470   continue
  480   continue
c
c     test if all points allocated
c
        ntsum=0
        do490 i=1,nmax
        ntsum=ntsum+nitest(i)
  490   continue
        if(ntsum-nmax)110,500,500
c
c     renumber units
c
  500   nadd=1
        do530 j=2,nmax
        if(nseg(j)-nseg(j-1))520,510,520
  510   nseg(j-1)=nadd
        go to 530
  520   nseg(j-1)=nadd
```

```
      nadd=nadd+1
  530 continue
      nseg(nmax)=nadd
c
c     print results
c
      write(20,680)titl
      if(nrnd-2)540,550,560
  540 write(20,730)
      write(20,750)cvmxa
      go to 570
  550 write(20,740)
       write(20,750)cvmxc
      go to 570
  560 write(20,760)
      write(20,770)cvmxa,cvmxc
  570 continue
      write(20,780)
      do580 i=1,nmax
      write(20,790)n(i),di(i),a(i),curv(i),nseg(i),nump(i),amean(i),
      &cmean(i),acvar(i)
  580 continue
      if(nrnd-2)611,612,613
  611 write(0,615)
  615 format("Do you wish to see the best segments?  0=no, 1=yes")
      read(0,620)itest2
      if(itest2)70,70,590
  612 write(0,616)
  616 format("Do you wish to see the best elements?  0=no, 1=yes")
      read (0,620)itest3
      if(itest3)80,80,590
  613 write(0,617)
  617 format("Do you wish to see the best slope units?  0=no, 1=yes")
      read(0,620)itest4
      if(itest4)670,670,590
  590 continue
c
c     print best  segments on tty
c
      write(0,680)titl
      if(nrnd-2)591,592,593
  591 write(0,730)
      write(0,750)cvmxa
      goto594
  592 write(0,740)
      write(0,750)cvmxc
      go to 594
  593 write(0,760)
```

```
      write(0,770)cvmxa,cvmxc
  594 continue
      write(0,780)
      do 600 i=1,nmax
      write(0,790)n(i),di(i),a(i),curv(i),nseg(i),nump(i),amean(i),
      &cmean(i),acvar(i)
  600 continue
c
c     plotting routine
c
      write(0,610)
  610 format("Do you want a Disspla plot?  0=no, 1=yes")
      read(0,620)itest
  620 format(v)
      if (itest) 670,670,630
  630 nxplt=nseg(nmax)
      xaray(1)=xfaray(1)
      yaray(1)=yfaray(1)
      iicount=2
      do 650 i=1,nmax
      tstaray(i)=nseg(i)-nseg(i+1)
      if (tstaray(i)) 645,650,646
  645 xaray(iicount)=xfaray(i+1)
      yaray(iicount)=yfaray(i+1)
      iicount=iicount+1
  646 xaray(nxplt+1)=xfaray(nmax+1)
      yaray(nxplt+1)=yfaray(nmax+1)
  650 continue
      xrnd=xfinal/25.
      xlast=((aint(xrnd))+1)*25.
      xinc=xlast/10.
      yfinal=xlast*.3
      yinc=yfinal/3.
c
c     dissola commands
c
      call bgnpl(-1)
      write(0,651)
  651 format("What kind of plot?  Tektronix=0, Calcomp=1")
      read(0,620)itest7
      if(itest7)653,653,658
  653 continue
      call tk120
      go to 659
  658 continue
      call calcmp(16)
  659 continue
      call physor(1.,1.)
```

```
      call title(' ',-1,'meters',6,'meters',6,10.,3.)
      if(nrnd-2)655,660,665
655   call messag('best segments$',100,3.,2.)
      go to 676
660   call messag('best elements$',100,3.,2.)
      go to 676
665   call messag('best slope units$',100,3.,2.)
676   continue
      call graph(0.,xinc,0.,yinc)
      call marker(6)
      call curve(xaray,yaray,nxplt+1,1)
      call endgr(1)
      call physor(1.,5.)
      call title(' ',-1,'meters',6,'meters',6,10.,3.)
      call messag('slope profile$',100,3.,2.)
     call graph(0.,xinc,0.,yinc)
      call marker(4)
      call curve(xfaray,yfaray,nmax+1,1)
      call endgr(2)
      call  endpl(-nrnd)
670   continue
      if(nrnd-2)70,80,675
675   continue
680   format(a80)
690   format(v)
700   format(v)
710   format(v)
720   format(v)
730   format("Best Rectilinear Segments")
740   format(1h ,21h best curved elements//)
750   format(" Maximum Coefficient of Variation=",f7.2)
760   format(1h ,17h best slope units//)
770   format(1h ,44h maximum coefficient of variation, segments=f7.2,
     &15h      elements=f7.2//)
780   format("    n     di     a          c   unit  nump     angle
     &     curv          cvar")
790   format(1h ,i3,x,f8.1,x,f8.2,x,f8.2,x,i4,x,i3,3x,f9.4,3
     &x,f9.4,4x,f9.4)
      close(1)
      write(0,800)
      close (20)
800   format("Do you want to run the program again? 0=no, 1=yes")
      read(0,810)itest1
810   format(v)
      if(itest1)820,820,2
820   continue
      call donepl
      end
```

```
      subroutine values(nitest,f,x,m,intsm,sumd,vmean,cvar)
      dimension nitest(m),f(m),x(m)
      intsm=0
      sum1=0.0
      sum2=0.0
      sum3=0.0
      do10 i=1,m
      intsm=intsm+nitest(i)
      sum1=sum1+f(i)
      sum2=sum2+f(i)*x(i)
      sum3=sum3+f(i)*x(i)**2
10    continue
      sumd=sum1
      vmean=sum2/sum1
      if(m-1)20,60,20
20    if(sum3/sum1-(sum2/sum1)**2)60,30,30
30    if(abs(sum2/sum1)-2)50,40,40
40    cvar=abs(100*((sqrt(sum3/sum1-(sum2/sum1)**2))/(sum2/sum1)))
      go to 70
50    cvar=abs(100*((sqrt(sum3/sum1-(sum2/sum1)**2))/2))
      go to 70
60    cvar=0.0
70    return
      end
```

SOURCE CODE FOR SLOPE_DATA

The source code for SLOPE_DATA is listed below. Variable names are the same as for the program SLOPES.

```
c     This program writes data files for  ***SLOPES***
c
            dimension n(250),a(250),di(250)
            character titl*80
            character nameseg*80
            open(0,promot =.true.)
   10       continue
            write(0,20)
   20       format("Do you want to enter or edit data?  0=enter, 1=edit")
            read(0,30)itest1
   30       format(v)
            if(itest1)10,40,180
   40       continue
            write(0,60)
   60       format("input title")
            read(0,120)titl
            write(0,65)
   65       format("Data should begin at the base of the slope")
            write(0,70)
   70       format("input cvmxa--maximum coefficient of variation--angle")
            read(0,130)cvmxa
            write(0,80)
   80       format("input cvmxc--max. coef. of variation--curvature")
            read(0,140)cvmxc
            write (0,90)
   90       format("input nmax--total number of measurements")
            read(0,150)nmax
            write(0,100)
  100       format("input n, angle, measured distance")
            do 110 i=1,nmax
            read(0,160)n(i),a(i),di(i)
  110       continue
  120       format(a80)
  130       format(v)
  140       format(v)
  150       format(v)
  160       format(v)
  170       go to 205
  180       continue
c
c     edit mode
```

```
c
      write(0,190)
  190 format("Type file name")
      read(0,200)nameseg
  200 format(a80)
      open(7,file=nameseg,form='formatted',mode='in')
      read(7,120)titl
      read(7,130)cvmxa
      read(7,140)cvmxc
      read(7,150)nmax
      read(7,160)(n(i),a(i),di(i),i=1,nmax)
      close(7)
  205 continue
      write(0,210)
  210 format("do you wish to review the data?  0=no, 1=yes")
      read(0,220)itest2
  220 format(v)
      if(itest2)205,240,230
  230 continue
      write(0,120)titl
      write(0,130)cvmxa
      write(0,140)cvmxc
      write(0,150)nmax
      do 240 i=1,nmax
      write(0,160)n(i),a(i),di(i)
  240 continue
      write(0,241)
  241 format("Do you wish to change the title?  0=no, 1=yes")
      read(0,300)itest3
      if (itest3)240,230,250
  250 write(0,260)
  260 format("input new title")
      read(0,270)titl
  270 format(a80)
  280 continue
      write(0,290)
  290 format("do you wish to change cvmxa?  0=no, 1=yes")
      read(0,300)itest4
  300 format(v)
      if(itest4)280,340,310
  310 continue
      write(0,320)cvmxa
  320 format("input new cvmxa, old=",f7.2)
      read(0,330)cvmxa
  330 format(v)
  340 continue
      write(0,350)
  350 format("do you want to change cvmxc?  0=no,1=yes")
```

```
      read(0,360)itest5
  360 format(v)
      if(itest5)340,400,370
  370 continue
      write(0,380)cvmxc
  380 format("input new cvmxc, old=",f7.2)
      read(0,390)cvmxc
  390 format(v)
  400 continue
      write(0,410)
  410 format("do you wish to change nmax?  0=no, 1=yes")
      read(0,420)itest6
  420 format(v)
      if(itest6)400,460,430
  430 continue
      write(0,440)nmax
  440 format("input new nmax, old=",f6.1)
      read(0,450)nmax
  450 format(v)
  460 continue
      write(0,470)
  470 format("do you wish to change a line of data?  0=no, 1=yes")
      read(0,480)itest7
  480 format(v)
      if(itest7)460,540,490
  490 continue
      write(0,500)
  500 format("which line do you want to change, n=?")
      read(0,510)iitest
  510 format(v)
      write(0,520)n(iitest),a(iitest),di(iitest)
  520 format("input n,a,di,  old values were",i2,2f8.3)
      read(0,530)n(iitest),a(iitest),di(iitest)
  530 format(v)
      go to 460
  540 continue
      write(0,550)
  550 format("what file name do you wish for this data?")
      read(0,322)nameseg
  322 format(a80)
  323 continue
c
c     output  to segment = nameseg
c
      open(20,file=nameseg, form='formatted', mode='out')
      write(20,120)titl
      write(20,130)cvmxa
      write(20,140)cvmxc
```

```
      write(20,150)nmax
      do 570 i=1,nmax
      write(20,160)n(i),a(i),di(i)
570   continue
      close(20)
      write (0,580)
580   format("do you wish to call another file?  0=no, 1=yes")
      read(0,590)itest20
590   format(v)
      if (itest20)570,600,10
600   continue
      end
```

SOURCE CODE FOR FANSEG

The source code for FANSEG is listed below. In addition to the variable names previously discussed, the following variables are used:

di -array for the length of the measured distances converted to meters.

theta -array for the tangent of the slope angle of a measured length.

ctour -array for the contour interval, in feet, along a measured length.

```
c       program fanseg
c
          dimension n(250),dist(250),a(250),nitest(250),ctour(250),di(250),
          &nseg(250),nump(250),initest(250),curv(250),dd(250),da(250),amean(
          &250),cmean(250),set(250),acvar(250),mpdi(250),theta(250),unit(250),
          &xaray(250),yaray(250),xfaray(250),yfaray(250),tstaray(250),
          &xdis(250),ydis(250)
        real theta
        real mpdi
        external bgnpl(descriptors),calcmp(descriptors),title(descriptors)
        external messag(descriptors),graph(descriptors),curve(descriptors)
        external marker(descriptors),physor(descriptors),endgr(descriptors)
        external endpl(descriptors),donepl(descriptors),tk120(descriptors)
        character nameseg*30
        character nameout*80
        character titl*80
c
c       input
c
          open(0,prompt=.true.)
   10     continue
          write(0,20)
   20     format("What is the name of the data file?")
          read(0,30)nameseg
   30     format(a80)
          open(1,file=nameseg,form='formatted',mode='in')
          read(1,780)titl
          read(1,790)cvmxa
          read(1,800)cvmxc
   40     read(1,810,end=770)nmax
          if(nmax)50,770,50
   50     read(1,820)(n(i),ctour(i),mpdi(i),i=1,nmax)
          close(1)
          write(0,60)
   60     format("What name for the output file?")
```

```
      read(0,70)nameout
   70 format(a80)
      open(20,file=nameout,form='formatted',mode='out')
c
c     compute slope angle
c
      do100 i=1,nmax
      if(ctour(i))80,90,80
   80  theta(i)=ctour(i)/mpdi(i)
       a(i)=(atan(theta(i)))*57.2958
      go to 100
   90 a(i)=0.0
  100 continue
c
c     compute plotting coordinates
c
      write(0,110)
  110 format("Enter elevation of beginning point")
      read(0,120)yfirst
  120 format(v)
      do 130 i=1,nmax
  130 continue
      xfirst=0.0
      xfaray(1)=xfirst
      yfaray(1)=yfirst
      do 140 i=2,nmax+1
      xfaray(i)=xfaray(i-1)+mpdi(i-1)
      yfaray(i)=yfaray(i-1)+ctour(i-1)
  140 continue
      xfinal=xfaray(nmax+1)
c
c     convert map distance to meters
c
      do 145 i=1,nmax
      di(i)=mpdi(i)*.3048
  145 continue
c
c     compute curv(i)
c
      curv(1)=200*((a(1)-a(2))/(di(1)+di(2)))
      curv(nmax)=200*((a(nmax-1)-a(nmax))/(di(nmax-1)+di(nmax)))
      icount=nmax-1
      do150 i=2,icount
      curv(i)=200*((a(i-1)-a(i+1))/(di(i-1)+2*di(i)+di(i+1)))
  150 continue
  160 nrnd=1
      go to 190
  170 nrnd=2
```

```
      go to 190
  180 nrnd=3
  190 continue
c
c
c     set variables to zero
c
c
      indsg=1
      do200 i=1,nmax
      nitest(i)=0
      amean(i)=0.0
      cmean(i)=0.0
      dist(i)=0.0
      acvar(i)=1000.
  200 continue
c
c
c     obtain all combinations in turn
c
c
  210 do470 j=1,nmax
      if(nitest(j))470,220,470
  220 nclk=1
  230 jcount=nmax-j+1
      do420 k=1,jcount
      if(nrnd-2)250,270,240
  240 if(nclk-1)270,250,270
  250 do260 i=1,nmax
      set(i)=a(i)
  260 continue
      go to 290
  270 do280 i=1,nmax
      set(i)=curv(i)
      nclk=2
  280 continue
  290 continue
      do300 i=1,k
      initest(i)=nitest(i+j-1)
      dd(i)=di(i+j-1)
      da(i)=set(i+j-1)
  300 continue
      call values(initest,dd,oa,k,intsm,sumd,vmean,cvar)
c
c
c     test parameters of combination
c
c
```

```
      if(intsm)430,310,430
  310 if(nclk-1)320,320,330
  320 cvrmx=cvmxa
      go to 340
  330 cvrmx=cvmxc
  340 if(cvar-cvrmx)350,350,430
  350 kcount=j+k-1
      do370 i=j,kcount
      if(sumd-dist(i))420,360,370
  360 if(cvar-acvar(i))370,420,420
  370 continue
c
c
c     allocate new values
c
c
      lcount=j+k-1
      do410 i=j,lcount
      nseg(i)=indsg
      nump(i)=k
      dist(i)=sumd
      acvar(i)=cvar
      if(nrnd-2)390,400,380
  380 if(nclk-1)390,390,400
  390 amean(i)=vmean
      cmean(i)=0.0
      go to 410
  400 cmean(i)=vmean
      amean(i)=0.0
  410 continue
      indsg=indsg+1
  420 continue
  430 if(nrnd-2)470,470,440
  440 if(nclk-1)470,450,470
  450 nclk=2
      go to 230
  460 continue
  470 continue
c
c     test to see if segments cut short
c
      do580 j=1,nmax
      if(j-1)480,490,480
  480 if(nseg(j)-nseg(j-1))490,580,490
  490 continue
      if(nitest(j)-1)500,580,500
  500 if(nseg(j)-nseg(j+nump(j)-1))510,560,510
  510 nn=1
```

```
  520   if(nseg(j)-nseg(j+nump(j)-1-nn))530,540,530
  530   nn=nn+1
        go to 520
  540   kk=nump(j)-nn
        mcount=j+kk-1
        do550 i=j,mcount
        nitest(i)=0
        amean(i)=0.0
        cmean(i)=0.0
        dist(i)=0.0
        acvar(i)=1000
  550   continue
        go to 580
  560   ncount=j+nump(j)-1
        do570 i=j,ncount
        nitest(i)=1
  570   continue
  580   continue
c
c     test to see if all points are allocated
c
        ntsum=0
        do590 i=1,nmax
        ntsum=ntsum+nitest(i)
  590   continue
        if(ntsum-nmax)210,600,600
c
c     renumber units
c
  600   nadd=1
        do630 j=2,nmax
        if(nseg(j)-nseg(j-1))620,610,620
  610   nseg(j-1)=nadd
        go to 630
  620   nseg(j-1)=nadd
        nadd=nadd+1
  630   continue
        nseg(nmax)=nadd
c
c     print results
c
        write(20,780)titl
        if(nrnd-2)640,650,660
  640   write(20,830)
        write(20,850)cvmxa
        go to 670
  650   write(20,840)
         write(20,850)cvmxc
```

```
      go to 670
660   write(20,860)
      write(20,870)cvmxa,cvmxc
670   continue
      write(20,880)
      do680 i=1,nmax
      write(20,890)n(i),mpdi(i),a(i),curv(i),nseg(i),nump(i),amean(i),
      &cmean(i),acvar(i)
680   continue
      if(nrnd-2)711,712,713
711   continue
      write(0,715)
715   format("Do you wish to see the best segments?  0=no, 1=yes")
      read(0,720)itest2
      if(itest2)170,170,690
712   continue
      write(0,716)
716   format("Do you wish to see the best elements?  0=no, 1=yes")
      read(0,720)itest3
      if (itest3)180,180,690
713   continue
      write(0,717)
717   format("Do you wish to see the best slope units?  0=no, 1=yes")
      read(0,720)itest4
      if (itest4) 770,770,690
690   continue
c
c     print best  segments on tty
c
      write(0,780)titl
      if (nrnd-2) 691,692,693
691   write(0,830)
      write(0,850)cvmxa
      go to 694
692   write(0,840)
      write(0,850)cvmxc
      go to 694
693   write(0,860)
      write(0,870)cvmxa,cvxmc
694   continue
      write(0,880)
      do 700 i=1,nmax
      write(0,890)n(i),mpdi(i),a(i),curv(i),nseg(i),nump(i),amean(i),
      &cmean(i),acvar(i)
700   continue
c
c     compute plotting scales
c
```

```
      write(0,710)
  710 format("do you want a disspla plot?  0=no, 1=yes")
      read(0,720)itest
  720 format(v)
      if (itest) 770,770,730
  730 nxplt=nseg(nmax)
      xaray(1)=xfaray(1)
      yaray(1)=yfaray(1)
      irnd=aint((yfaray(nmax+1)-yfirst)/30.)
      ylast=(irnd+1)*30.
      ystep=ylast/3.
      xstep=ystep*10.
      iicount=2
      do 750 i=1,nmax
      tstaray(i)=nseg(i)-nseg(i+1)
      if (tstaray(i))745,750,746
  745 xaray(iicount)=xfaray(i+1)
      yaray(iicount)=yfaray(i+1)
      iicount  =iicount + 1
  746 xaray(nxplt+1)=xfaray(nmax+1)
     yaray(nxplt+1)=yfaray(nmax+1)
  750  continue
c
c     call plotting routine
c
      call bgnpl(-1)
      write(0,751)
  751 format("tektronix = 0, calcomp =1")
      read(0,720)itest7
      if(itest7)753,753,758
  753 write(0,754)
  754 format("set baud rate at 1200 cps")
      call tk120
      go to 759
  758 call calcmp(16)
  759 continue
      call physor(1.,1.)
      call title(' ',-1,'feet',4,'feet',4,10.,3.)
      if(nrnd-2)755,760,765
  755 continue
      call messag('best segments, ve=10x$',100,1.5,2.5)
      go to  776
  760 continue
      call  messag('best elements, ve=10x$',100,1.5,2.5)
      go to 776
  765 continue
      call messag('best units, ve=10x$',100,1.5,2.5)
  776 continue
```

```
      call graph(0.,xstep,yfirst,ystep)
      call marker(8)
      call curve(xaray,yaray,nxplt+1,1)
      call endgr(1)
      call physor(1.,5.)
      call title(' ',-1,'feet',4,'feet',4,10.,3.)
      call messag ('fan profile, ve=10x$',100, 1.5,2.5)
      call graph(0.,xstep,yfirst,ystep)
      call marker(4)
      call curve(xfaray,yfaray,nmax+1,1)
      call endgr(2)
      call endpl(-nrnd)
770   continue
      if(nrnd-2)170,180,775
775   continue
780   format(a80)
790   format(v)
800   format(v)
810   format(v)
820   format(v)
830   format("Best Rectilinear Segments")
840   format(1h ,21h best curved elements//)
850   format(" Maximum Coefficient of Variation=",f7.2)
860   format(1h ,17h best slope units//)
870   format(1h ,44h maximum coefficient of variation, segments=f7.2,
     &15h      elements=f7.2//)
880   format("    n    mpdi      a         c     unit   nump         angle
     &     curv          cvar")
890   format(1h ,i3,x,f8.1,x,f8.2,x,f8.2,x,i4,x,i3,4x,f9.4,3
     &x,f9.4,2x,f9.4)
      close(1)
      write(0,900)
      close(20)
900   format("Do you want to run the program again? 0=no, 1=yes")
      read(0,910)itest1
910   format(v)
      if(itest1)920,920,10
920   continue
      call donepl
      end
      subroutine values(nitest,f,x,m,intsm,sumd,vmean,cvar)
      dimension nitest(m),f(m),x(m)
      intsm=0
      sum1=0.0
      sum2=0.0
      sum3=0.0
      do10 i=1,m
      intsm=intsm+nitest(i)
```

```
      sum1=sum1+f(i)
      sum2=sum2+f(i)*x(i)
      sum3=sum3+f(i)*x(i)**2
10    continue
      sumd=sum1
      vmean=sum2/sum1
      if(m-1)20,60,20
20    if(sum3/sum1-(sum2/sum1)**2)60,30,30
30    if(abs(sum2/sum1)-2)50,40,40
40    cvar=abs(100*((sqrt(sum3/sum1-(sum2/sum1)**2))/(sum2/sum1)))
      go to 70
50    cvar=abs(100*((sqrt(sum3/sum1-(sum2/sum1)**2))/2))
      go to 70
60    cvar=0.0
70    return
      end
```

SOURCE CODE FOR FANSEG_DATA

The source code for FANSEG_DATA is listed below. Variable names are the same as those for FANSEG.

```
c     This program writes data files for  ***FANSEG***
c
      dimension n(250),ctour(250),mapdis(250)
      character titl*80
      character nameseg*80
      open(0,prompt =.true.)
   10 continue
      write(0,20)
   20 format("Do you want to enter or edit data?  0=enter, 1=edit")
      read(0,30)itest1
   30 format(v)
      if(itest1)10,40,180
   40 continue
      write(0,60)
   60 format("input title")
      read(0,120)titl
      write(0,65)
   65 format("Data should begin at the base of the slope")
      write(0,70)
   70 format("input cvmxa--maximum coefficient of variation--angle")
      read(0,130)cvmxa
      write(0,80)
   80 format("input cvmxc--max. coef. of variation--curvature")
      read(0,140)cvmxc
      write (0,90)
   90 format("input nmax--total number of measurements")
      read(0,150)nmax
      write(0,100)
  100 format("input n,  contour interval, measured distance")
      do 110 i=1,nmax
      read(0,160)n(i),ctour(i),mapdis(i)
  110 continue
  120 format(a80)
  130 format(v)
  140 format(v)
  150 format(v)
  160 format(v)
  170 go to 205
  180 continue
c
c     edit mode
```

```
c
      write(0,190)
190   format("Type file name")
      read(0,200)nameseg
200   format(a80)
      open(7,file=nameseg,form='formatted',mode='in')
      read(7,120)titl
      read(7,130)cvmxa
      read(7,140)cvmxc
      read(7,150)nmax
      read(7,160)(n(i),ctour(i),mapdis(i),i=1,nmax)
      close(7)
205   continue
      write(0,210)
210   format("do you wish to review the data?  0=no, 1=yes")
      read(0,220)itest2
220   format(v)
      if(itest2)205,240,230
230   continue
      write(0,120)titl
      write(0,130)cvmxa
      write(0,140)cvmxc
      write(0,150)nmax
      do 240 i=1,nmax
      write(0,160)n(i),ctour(i),mapdis(i)
240   continue
      write(0,241)
241   format("Do you wish to change the title?  0=no, 1=yes")
      read(0,300)itest3
      if (itest3)240,280,250
250   write(0,260)
260   format("input new title")
      read(0,270)titl
270   format(a80)
280   continue
      write(0,290)
290   format("do you wish to change cvmxa?  0=no, 1=yes")
      read(0,300)itest4
300   format(v)
      if(itest4)280,340,310
310   continue
      write(0,320)cvmxa
320   format("input new cvmxa, old=",f7.2)
      read(0,330)cvmxa
330   format(v)
340   continue
      write(0,350)
350   format("do you want to change cvmxc?  0=no,1=yes")
```

```
      read(0,360)itest5
360   format(v)
      if(itest5)340,400,370
370   continue
      write(0,380)cvmxc
380   format("input new cvmxc, old=",f7.2)
      read(0,390)cvmxc
390   format(v)
400   continue
      write(0,410)
410   format("do you wish to change nmax?  0=no, 1=yes")
      read(0,420)itest6
420   format(v)
      if(itest6)400,460,430
430   continue
      write(0,440)nmax
440   format("input new nmax, old=",f6.1)
      read(0,450)nmax
450   format(v)
460   continue
      write(0,470)
470   format("do you wish to change a line of data?  0=no, 1=yes")
      read(0,480)itest7
480   format(v)
      if(itest7)460,540,490
490   continue
      write(0,500)
500   format("which line do you want to change, n=?")
      read(0,510)iitest
510   format(v)
      write(0,520)n(iitest),ctour(iitest),mapdis(iitest)
520   format("input n,ctour,mapdis,  old values were",i2,2f8.3)
      read(0,530)n(iitest),ctour(iitest),mapdis(iitest)
530   format(v)
      go to 460
540   continue
      write(0,550)
550   format("what file name do you wish for this data?")
      read(0,322)nameseg
322   format(a80)
323   continue
c
c     output  to segment = nameseg
c
      open(20,file=nameseg, form='formatted', mode='out')
      write(20,120)titl
      write(20,130)cvmxa
      write(20,140)cvmxc
```

```
      write(20,150)nmax
      do 570 i=1,nmax
      write(20,160)n(i),ctour(i),mapdis(i)
570   continue
      close(20)
      write (0,580)
580   format("do you wish to call another file?  0=no, 1=yes")
      read(0,590)itest20
590   format(v)
      if (itest20)570,600,10
600   continue
      end
```

REFERENCES CITED

Strahler, A. N., 1950, Equilibrium theory of slopes approached by frequency distribution analysis: American Journal of Science, v. 248, p. 800-814.

Wallace, R. E., 1977, Profiles and ages of young fault scarps, north-central Nevada: Geological Society of America Bulletin, v. 88, p. 1267-1281.

Young, A., 1971, Slope profile analysis--The system of best units: Institute British Geographers Special Publication 3, p. 1-13.

_____1975, Slopes: New York, Longman, Inc., 288 p.