IDENTIFICATION

SPLINE CURVE FITTING - MU KUFIT
704 FORTRAN SUBROUTINE
Richard S. Christian, Terry W. Edwards - Revised October 20, 1964
Midwestern Universities Research Association, Madison, Wisconsin

PURPOSE

To fit a plane curve, y(x), either of open or closed form, to an unlimited set of tabular coordinates, x_i and y_i such that the function itself, and the first and perhaps the second derivatives are continuous at all tabular points; and to compute arc lenghts along the curve.

METHOD

MU KUFIT generates a table of coefficients C_{1i}, C_{2i}, and C_{3i}, valid in the interval $(x,y)_i$ to $(x,y)_{i+1}$ such that

$$\Delta y = C_{1i}\Delta x + C_{2i}\Delta x^2 + C_{3i}\Delta x^3 \quad \text{if } \Delta y_{i+1} \leq \Delta x_{i+1},$$

or $$\Delta x = C_{1i}\Delta y + C_{2i}\Delta y^2 + C_{3i}\Delta y^3 \quad \text{if } \Delta y_{i+1} > \Delta x_{i+1},$$

where $\Delta x = (x - x_i)$ and $\Delta y = (y - y_i)$.

An accumulated arc length is also computed for each $(x,y)_i$ such that $S(x,y)_1 = 0$.

The coefficients C_{1i}, C_{2i}, and C_{3i} are generated by an iteration process involving sucessively overlapping sub groups of coordinate pairs.

It is possible to specify first derivatives (either y'_i or x'_i depending on whether $\Delta y_{i+1} \leq \Delta x_{i+1}$ or $\Delta y_{i+1} > \Delta x_{i+1}$, respectively) at any point or points along the curve. MU KUFIT will attempt to match these first derivatives at the points indicated, and will, in this event, generate discontinuous second derivatives at the points indicated.

If no first derivatives are specified, MU KUFIT will choose the best values it can (via Lagrangian methods) for the end points of an open curve, and will make all derivatives continuous (through the ends of the specified table) in the event of a closed curve. This latter result is obtained by making two passes over the curve, the second pass being made over rotated arrays of coordinates.

USAGE

MU KUFIT may be entered via the 11 arguement calling sequence:

CALL KUFIT (X,Y,CO,M,KIND,LT,FAC,NGR,INBR,DYPN1, DYPPN2)

where the various arguements have the following significance:

X: One dimensional array of x coordinates spacifing curve.
Y: One dimensional array of y coordinates specifing curve.

CO: A two dimensional array, CO(4,...), used for both input and output as follows:

INPUT:

CO(4,n)≠0: signal that first derivative is specified for the n^{th} coordinate point.

CO(1,n): value of first derivative so specified for the n^{th} coordinate point.

CO(2,n) and CO(3,n): not used

OUTPUT:

CO(1,n), CO(2,n), CO(3,n) are the resulting coefficients for the n^{th} coordinate point.

CO(4,n) is the accumulated arc length for the n^{th} coordinate point.

M: Number of coordinate points comprising curve. For a closed curve the first and last coordinates must be identical, and hence M ust be one larger than the total number of distinct points.

KIND: Type of curve involved:

-1: Open curve.

0: Closed curve with discontinuous first derivative at the joint, i.e. at $(x,y)_1 = (x,y)_M$.

+1: Closed curve with continuous first derivatives everywhere, but with straight sections, hence discontinuous second derivatives.

2: Closed curves with first and second derivatives continuous everywhere.

LT: Number of allowed iterations over a single subgroup of points. (Nominally LT=30)*

FAC: Factor by which iteration parameter is to be modified by between iterations. (Nominally FAC = 1.04)*

NGR: Maximum number of points allowed in one subgroup. (Nominally NGR = 8)*

INBR: Number of itermediate points used for arc length calculations. (Nominally INBR = 20)*

*Note: The above four arguements, if not specified, or if specified outside the ranges

$$10 \leq LT \leq 30, \quad 1.01 \leq FAC \leq 1.10, \quad 4 \leq NGR \leq 10, \quad 4 \leq INBR \leq 20$$

will be set to their nominal value.

DYPN1: Returned as $(\Delta y')_{n1}$ if a subgroup of points can not be made to converge in LT itevations.**

DYPPN2: Returned as $(\Delta y'')_{n2}$ if a subgroup of points can not be made to converge in LT itevations.**

**Note: These last two arguements are indicative of the quality of the fit achieved by MU KUFIT over the specified curve. Both are returned as zero if all subgroups converged in the fitting process.

AUXILIARY ROUTINES

MU KUFIT requires the following subroutines:

1. Floating point SQRT (Floating point arguement).

CODING INFORMATION

MU KUFIT is available in both FORTRAN source card and binary card forms. Approximately 3300 locations of core storage are required.

```
C       SUBROUTINE KUFIT - CHRISTIAN, EDWARDS
C
C       TWO DIMENSIONAL SPLINE CURVE FITTING
C
C       PROGRAM WILL FIT ANY TWO DIMENSIONAL CURVE,
C       Y=Y(X),OF EITHER OPEN OR CLOSED FORM, WITH
C       A SEQUENCE OF CUBIC POLYNOMIALS VALID BETWEEN
C       SUCCESSIVE COORDINATE POINTS. THE FITTING IS
C       DONE SUCH THAT BOTH FIRST AND SECOND DERIVATIVES
C       ARE CONTINUOUS OVER EACH PORTION OF THE CURVE
C       SO SPECIFIED. PROGRAM ALSO CALCULATES ARC LENGTHS
C       ALONG CURVE.
C
C       TWO DIFFERENT CALLING SEQUENCES ARE POSSIBLE.
C       METHOD 1 REQUIRES LESS PROGRAM STORAGE, METHOD 2 REQUIRES
C       LESS WORK.
C
C       CALLING SEQUENCE 1 - USER MAY ALTER COMMON STORAGE.
C
C       SUBROUTINE KUFIT (MM,KKIND,LLT,FFAC,NNGR,IINBR,DYPN1,DYPPN2)
C       DIMENSION X(100),Y(100),CO(4,100)
C       COMMON X,Y,CO
C
C       CALLING SEQUENCE 2 - THIS IS NORMAL SEQUENCE.
C
        SUBROUTINE KUFIT(X,Y,CO,MM,KKIND,LLT,FFAC,NNGR,IINBR,DYPN1,DYPPN2)
        DIMENSION X(1),Y(1),CO(4,1)
C
        DIMENSION YP1(30),CN2(30),IC(2,10)
        M=MM
        KIND=KKIND
        LC=0
        LRC=0
        IF (KIND) 10,2,2
 2      IF (X(1)-X(M)) 4,3,4
 3      IF (Y(1)-Y(M)) 4,5,4
 4      KIND=-1
        GO TO 10
 5      LC=1
        IF (KIND-1) 10,10,6
 6      DO 7 I=1,M
        IF (CO(4,I)) 8,7,8
 7      CONTINUE
        GO TO 10
 8      KIND=1
 10     LT=30
        IF (LLT-10) 16,14,12
 12     IF (LLT-30) 14,16,16
 14     LT=LLT
 16     FAC=1.04
        IF (FFAC-1.01) 22,20,18
 18     IF (FFAC-1.10) 20,20,22
 20     FAC=FFAC
 22     MAXGR=10
        NGR=8
        IF (NNGR-4) 28,26,24
 24     IF (NNGR-10) 26,26,28
 26     NGR=NNGR
 28     INBR=20
        IF (IINBR-4) 34,32,30
```

```
30    IF (IINBR-20) 32,34,34
32    INBR=IINBR
34    DYPN1=0.0
      DYPPN2=0.0
36    DO 38 I=1,LT
      YP1(I)=0.0
38    CN2(I)=0.0
40    N1=1
      NDGR=0
      CVFAC=1.0
      LM=1
      NEST1=0
      NESTM=0
      IF (LC-1) 60,60,42
42    MS=M/2
      LRC=LRC+MS
44    DO 52 I=1,MS
      XX=X(1)
      YY=Y(1)
      CO1=CO(1,1)
      CO2=CO(2,1)
      CO3=CO(3,1)
      CO4=CO(4,1)
      M2=M-1
46    DO 50 J=2,M2
      X(J-1)=X(J)
      Y(J-1)=Y(J)
      DO 48 K=1,4
      CO(K,J-1)=CO(K,J)
48    CONTINUE
50    CONTINUE
      X(M2)=XX
      Y(M2)=YY
      CO(1,M2)=CO1
      CO(2,M2)=CO2
      CO(3,M2)=CO3
      CO(4,M2)=CO4
52    CONTINUE
      X(M)=X(1)
      Y(M)=Y(1)
      DO 55 K=1,4
      CO(K,M)=CO(K,1)
55    CONTINUE
      GO TO 102
60    IF (KIND-1) 62,75,75
62    IF (CO(4,1)) 67,65,67
65    NEST1=1
67    IF (CO(4,M)) 72,70,72
70    NESTM=M
72    GO TO 92
75    IF (CO(4,1)) 77,85,77
77    IF (CO(4,M)) 82,80,82
80    CO(1,M)=CO(1,1)
      CO(4,M)=CO(4,1)
82    GO TO 92
85    IF (CO(4,M)) 87,90,87
87    CO(1,1)=CO(1,M)
      CO(4,1)=CO(4,M)
      GO TO 92
90    NEST1=1
```

```
      NESTM=M
92    LM=LM+1
      GO TO 100
95    LM=LM+1
97    NEST1=0
100   CO(2,N1)=0.0
      CO(3,N1)=0.0
      IF (N1-M) 102,490,490
102   DO 106 I=LM,M
      IF (CO(4,I)) 104,106,104
104   LM=I
      GO TO 108
106   CONTINUE
      LM=M
108   IF (LM-N1-1) 110,110,145
110   DXN=X(N1+1)-X(N1)
      DYN=Y(N1+1)-Y(N1)
      YPN=CO(1,N1)
      IF (M-N1-1) 112,112,118
112   IF (KIND) 114,114,116
114   YPM=CO(1,N1+1)
      IF (ABSF(DYN)-ABSF(DXN)) 135,135,137
116   DXM=X(2)-X(1)
      DYM=Y(2)-Y(1)
      GO TO 120
118   DXM=X(N1+2)-X(N1+1)
      DYM=Y(N1+2)-Y(N1+1)
120   IF (ABSF(DYN)-ABSF(DXN)) 122,122,128
122   IF (ABSF(DYM)-ABSF(DXM)) 124,124,126
124   YPM=CO(1,N1+1)
      GO TO 135
126   YPM=1.0/CO(1,N1+1)
      GO TO 135
128   IF (ABSF(DYM)-ABSF(DXM)) 130,130,132
130   YPM=1.0/CO(1,N1+1)
      GO TO 137
132   YPM=CO(1,N1+1)
      GO TO 137
135   CO(2,N1)=(3.0*DYN/DXN-(YPM+YPN+YPN))/DXN
      CO(3,N1)=(YPN+YPM-2.0*DYN/DXN)/(DXN*DXN)
      GO TO 140
137   CO(2,N1)=(3.0*DXN/DYN-(YPM+YPN+YPN))/DYN
      CO(3,N1)=(YPN+YPM-2.0*DXN/DYN)/(DYN*DYN)
140   N1=N1+1
      GO TO 95
145   IF (NDGR) 150,150,147
147   NDGR=NDGR/2
150   N2=N1+NGR-NDGR-1
      IF (N2-LM) 165,165,152
152   N2=LM
      IF (N2-N1-2) 155,160,160
155   N1=N2-2
      LM=N1+1
      GO TO 97
160   NDGR=NGR-(N2-N1+1)
165   IF (NEST1) 190,190,167
167   IF (LC-1) 170,170,200
170   DXI=X(N1+1)-X(N1)
      DYI=Y(N1+1)-Y(N1)
      DXJ=X(N1+2)-X(N1)
```

```
      DYJ=Y(N1+2)-Y(N1)
      IF (N2-N1-2) 172,172,180
172   IF (ABSF(DYI)-ABSF(DXI)) 175,178,178
175   DJI=DXJ-DXI
      CO(1,N1)=DXJ/DXI*DYI/DJI-DXI/DXJ*DYJ/DJI
      GO TO 190
178   DJI=DYJ-DYI
      CO(1,N1)=DYJ/DYI*DXI/DJI-DYI/DYJ*DXJ/DJI
      GO TO 190
180   DXK=X(N1+3)-X(N1)
      DYK=Y(N1+3)-Y(N1)
      IF (ABSF(DYI)-ABSF(DXI)) 182,182,185
182   DJI=DXJ-DXI
      DKJ=DXK-DXJ
      DIK=DXI-DXK
      CO(1,N1)=-DXK/DXI*DXJ/DJI*DYI/DIK-DXI/DXJ*DXK/DJI*DYJ/DKJ-DXJ/DXK*
     1DXI/DKJ*DYK/DIK
      GO TO 190
185   DJI=DYJ-DYI
      DKJ=DYK-DYJ
      DIK=DYI-DYK
      CO(1,N1)=-DYK/DYI*DYJ/DJI*DXI/DIK-DYI/DYJ*DYK/DJI*DXJ/DKJ-DYJ/DYK*
     1DYI/DKJ*DXK/DIK
190   IF (N1-1) 192,192,200
192   IF (KIND-1) 200,195,195
195   CO(1,M)=CO(1,1)
200   YP1M=CO(1,N1)
      IREV=0
      IGR=0
      DO 202 I=1,MAXGR
      IC(1,I)=0
202   IC(2,I)=0
      IF (N2-LM) 206,204,204
204   IF (N2-M) 220,205,205
205   IF (NESTM) 220,220,206
206   DXI=X(N2-1)-X(N2)
      DYI=Y(N2-1)-Y(N2)
      DXJ=X(N2-2)-X(N2)
      DYJ=Y(N2-2)-Y(N2)
      IF (N2-N1-2) 208,208,214
208   IF (ABSF(DYI)-ABSF(DXI)) 210,210,212
210   DJI=DXJ-DXI
      CO(1,N2)=DXJ/DXI*DYI/DJI-DXI/DXJ*DYJ/DJI
      GO TO 220
212   DJI=DYJ-DYI
      CO(1,N2)=DYJ/DYI*DXI/DJI-DYI/DYJ*DXJ/DJI
      GO TO 220
214   DXK=X(N2-3)-X(N2)
      DYK=Y(N2-3)-Y(N2)
      IF (ABSF(DYI)-ABSF(DXI)) 216,216,218
216   DJI=DXJ-DXI
      DKJ=DXK-DXJ
      DIK=DXI-DXK
      CO(1,N2)=-DXK/DXI*DXJ/DJI*DYI/DIK-DXI/DXJ*DXK/DJI*DYJ/DKJ-DXJ/DXK*
     1DXI/DKJ*DYK/DIK
      GO TO 220
218   DJI=DYJ-DYI
      DKJ=DYK-DYJ
      DIK=DYI-DYK
      CO(1,N2)=-DYK/DYI*DYJ/DJI*DXI/DIK-DYI/DYJ*DYK/DJI*DXJ/DKJ-DYJ/DYK*
```

```
     1DYI/DKJ*DXK/DIK
 220  L=1
      FACT=FAC
      FACT2=(FACT-1.0)/4.0
      TEST=(ABSF(YP1M)+1.0)*FACT2/250.0
      DXI=X(N2-1)-X(N2)
      DYI=Y(N2-1)-Y(N2)
      DXJ=X(N2-2)-X(N2)
      DYJ=Y(N2-2)-Y(N2)
      IF (N2-N1-2) 222,222,228
 222  IF (ABSF(DYI)-ABSF(DXI)) 224,224,226
 224  DJI=DXJ-DXI
      CC=-2.0/DXI*DYI/DJI+2.0/DXJ*DYJ/DJI
      GO TO 231
 226  DJI=DYJ-DYI
      CC=-2.0/DYI*DXI/DJI+2.0/DYJ*DXJ/DJI
      GO TO 233
 228  DXK=X(N2-3)-X(N2)
      DYK=Y(N2-3)-Y(N2)
      IF (ABSF(DYI)-ABSF(DXI)) 230,230,232
 230  DJI=DXJ-DXI
      DKJ=DXK-DXJ
      DIK=DXI-DXK
      CC=2.0/DXI*(DXK+DXJ)/DJI*DYI/DIK+2.0/DXJ*(DXI+DXK)/DJI*DYJ/DKJ+2.0
     1/DXK*(DXJ+DXI)/DKJ*DYK/DIK
 231  KASE=2
      DN=-DXI
      AA=-DYI
      BB=CO(1,N2)
      GO TO 235
 232  DJI=DYJ-DYI
      DKJ=DYK-DYJ
      DIK=DYI-DYK
      CC=2.0/DYI*(DYK+DYJ)/DJI*DXI/DIK+2.0/DYJ*(DYI+DYK)/DJI*DXJ/DKJ+2.0
     1/DYK*(DYJ+DYI)/DKJ*DXK/DIK
 233  KASE=1
      DN=-DYI
      AA=-DXI
      BB=CO(1,N2)
 235  CN2(1)=CC
 240  MODE=0
      IF (ABSF(CC)-0.05) 245,245,250
 245  MODE=1
 250  CO(2,N2)=CC/2.0
 254  DO 295 KK=N1,N2
      K=N2+N1-KK
      IF (KK-N1-1) 295,290,256
 256  DXN=X(K+1)-X(K)
      DYN=Y(K+1)-Y(K)
      IF (ABSF(DYN)-ABSF(DXN)) 270,270,260
 260  DN=DYN
      AA=DXN
      IF (KASE-1) 285,285,265
 265  KASE=1
      GO TO 280
 270  DN=DXN
      AA=DYN
      IF (KASE-1) 275,275,285
 275  KASE=2
 280  BB=1.0/CO(1,K+1)
```

```
      CC=-2.0*CO(2,K+1)/(CO(1,K+1)**3)
      GO TO 290
285   BB=CO(1,K+1)
      CC=2.0*CO(2,K+1)
290   DN2=DN*DN
      DN3=DN*DN2
      CO(1,K)=0.5*(6.0*AA-4.0*DN*BB+DN2*CC)/DN
      CO(2,K)=0.5*(-6.0*AA+6.0*DN*BB-2.0*DN2*CC)/DN2
      CO(3,K)=0.5*(2.0*AA-2.0*DN*BB+DN2*CC)/DN3
295   CONTINUE
      YP1(L)=CO(1,N1)
      IGR=IGR+1
      IF (ABSF(YP1(L)-YP1M)-TEST*CVFAC*ABSF((YP1(L)+YP1M)/2.0))
     1 300,300,302
300   NDONE=1
      GO TO 378
302   NDONE=0
      NOUT=0
      IF (L-2) 304,304,378
304   DO 309 KK=N1,N2
      K=N2+N1-KK
      KL=N2-KK+1
      IF (KL-1) 305,305,306
305   IF (CO(1,K)*YP1M) 307,309,308
306   IF (CO(1,K)*CO(1,K-1)) 307,307,308
307   IC(L,KL)=-1
      GO TO 309
308   IC(L,KL)=+1
309   CONTINUE
      IF (L-2) 310,330,378
310   L=2
      NOKL=N2-N1+1
      DO 314 KK=N1,N2
      KL=N2-KK+1
      IF (IC(1,KL)) 311,311,313
311   IF (KL-2) 315,312,312
312   IF (IC(1,KL-1)) 315,315,313
313   NOKL=KL
314   CONTINUE
315   DXN=X(N2)-X(N2-1)
      DYN=Y(N2)-Y(N2-1)
      IF (ABSF(DYN)-ABSF(DXN)) 318,318,316
316   KASE=1
      DN=DYN
      AA=DXN
      GO TO 320
318   KASE=2
      DN=DXN
      AA=DYN
320   BB=CO(1,N2)
      IF (MODE) 327,327,322
322   IF (FACT-1.0) 323,323,324
323   IF (CN2(1)) 326,325,325
324   IF (CN2(1)) 325,326,326
325   CC=CN2(1)-FACT2
      GO TO 328
326   CC=CN2(1)+FACT2
      GO TO 328
327   CC=CN2(1)*FACT
328   CN2(2)=CC
```

```
      GO TO 378
330   DO 332 KK=N1,N2
      KL=N2-KK+1
      IF (IC(1,KL)-IC(2,KL)) 334,332,334
332   CONTINUE
      NOK=NOKL
      GO TO 344
334   NOK=N2-N1+1
      DO 342 KK=N1,N2
      KL=N2-KK+1
      IF (IC(2,KL)) 336,340,340
336   IF (KL-2) 344,338,338
338   IF (IC(2,KL-1)) 344,340,340
340   NOK=KL
342   CONTINUE
344   IF (NOK-NOKL) 352,348,346
346   IF (NOK-3) 358,358,364
348   DO 350 KK=N1,N2
      KL=N2-KK+1
      IF (IC(2,KL)) 350,352,352
350   CONTINUE
      NOUT=1
      GO TO 378
352   IF (NOKL-3) 358,358,368
358   X1=YP1(1)-YP1M
      X2=YP1(2)-YP1M
      IF (X1) 360,362,361
360   IF (X2) 363,362,375
361   IF (X2) 375,362,363
362   NDONE=1
      GO TO 378
363   IF (ABSF(X2)-ABSF(X1)) 368,364,364
364   IF (IREV) 365,365,366
365   FACT=1.0/FACT
      GO TO 367
366   FACT=2.0/(FACT+1.0)
367   FACT2=ABSF(FACT-1.0)/4.0
      IREV=IREV+1
368   YP1(1)=YP1(2)
      CN2(1)=CN2(2)
      DO 370 KK=N1,N2
      KL=N2-KK+1
370   IC(1,KL)=IC(2,KL)
      NOKL=NOK
372   IF (IGR-300/(N2-N1+1)) 315,374,374
374   NOUT=1
      GO TO 378
375   L=3
      L1=1
      L2=2
      LBRKT=0
      DO 376 I=1,MAXGR
      IC(1,I)=0
376   IC(2,I)=0
378   CONTINUE
385   IF (NDONE) 386,386,450
386   IF (L-2) 388,388,390
388   IF (NOUT) 240,240,425
390   IF (LBRKT) 405,405,392
392   IF (L-LT) 394,425,425
```

```
394   X1=YP1(L1)-YP1M
      X2=YP1(L2)-YP1M
      X3=YP1(L)-YP1M
      IF (X1*X3) 396,396,398
396   IF (X2*X3) 402,402,400
398   L1=L
      GO TO 402
400   L2=L
402   L=L+1
405   LBRKT=LBRKT+1
      DXN=X(N2)-X(N2-1)
      DYN=Y(N2)-Y(N2-1)
      IF (ABSF(DYN)-ABSF(DXN)) 415,415,410
410   KASE=1
      DN=DYN
      AA=DXN
      GO TO 420
415   KASE=2
      DN=DXN
      AA=DYN
420   BB=CO(1,N2)
      ALFA=(YP1M-YP1(L1))/(YP1(L2)-YP1(L1))
      CC=((1.0-ALFA)*CN2(L1)+ALFA*CN2(L2))
     1*0.90+((CN2(L1)+CN2(L2))/2.0)*0.10
      CN2(L)=CC
      DO 423 I=1,L
      IF (I-L) 422,423,423
422   IF (CC-CN2(I)) 423,425,423
423   CONTINUE
      GO TO 240
425   NOUT=0
      IF (N2-N1-2) 430,430,427
427   CO(1,N1)=YP1M
      NDGR=(N2-N1+1)-((N2-N1+1)*5)/6+NDGR
      CVFAC=CVFAC*5.0
      GO TO 470
430   IF (ABSF(CO(1,N1)-YP1M)-DYPN1) 440,440,435
435   DYPN1=ABSF(CO(1,N1)-YP1M)
440   IF (ABSF(CN2(L-1)-CN2(L))-DYPPN2) 450,450,445
445   DYPPN2=ABSF(CN2(L-1)-CN2(L))
450   IF (N2-M) 455,490,490
455   CVFAC=1.0
      IF (N2-LM) 460,485,485
460   N1=N1+(N2-N1+1)/2
      IF (NDGR-1) 470,465,465
465   NDGR=NDGR-1
470   N2=N1+NGR-NDGR-1
      IF (N2-LM) 480,480,472
472   N2=LM
      IF (N2-N1-2) 475,477,477
475   N1=N2-2
477   NDGR=NGR-(N2-N1+1)
480   GO TO 200
485   N1=N2
      GO TO 95
490   IF (LC-1) 660,495,495
495   NEND=0
      IF (KIND-1) 500,505,505
500   NEND=1
      GO TO 520
```

```
505   IF (CO(4,M-1)) 515,510,515
510   IF (CO(4,1)) 515,520,515
515   CO(1,M)=CO(1,1)
      CO(2,M)=CO(2,1)
      CO(3,M)=CO(3,1)
      GO TO 600
520   CONTINUE
550   DXN=X(M)-X(M-1)
      DYN=Y(M)-Y(M-1)
      IF (ABSF(DYN)-ABSF(DXN)) 560,560,555
555   KASE=1
      DN=DYN
      GO TO 570
560   KASE=2
      DN=DXN
570   DXN=X(2)-X(1)
      DYN=Y(2)-Y(1)
      IF (ABSF(DYN)-ABSF(DXN)) 585,585,580
580   IF (KASE-1) 595,595,590
585   IF (KASE-1) 590,590,595
590   CO(1,M)=1.0/CO(1,M)
      CO(2,M)=-CO(2,M)*CO(1,M)**3
      CO(3,M)=CO(3,1)
595   IF (LC-1) 660,596,600
596   IF (NEND) 597,597,600
597   LC=LC+1
      GO TO 40
600   IF (LRC) 660,660,610
610   M3=M-1
      DO 650 I=1,LRC
      XX=X(M3)
      YY=Y(M3)
      CO1=CO(1,M3)
      CO2=CO(2,M3)
      CO3=CO(3,M3)
      CO4=CO(4,M3)
620   DO 640 J=2,M3
      M2=M3+2-J
      X(M2)=X(M2-1)
      Y(M2)=Y(M2-1)
      DO 630 K=1,4
      CO(K,M2)=CO(K,M2-1)
630   CONTINUE
640   CONTINUE
      X(1)=XX
      Y(1)=YY
      CO(1,1)=CO1
      CO(2,1)=CO2
      CO(3,1)=CO3
      CO(4,1)=CO4
650   CONTINUE
      X(M)=X(1)
      Y(M)=Y(1)
      DO 655 K=1,4
      CO(K,M)=CO(K,1)
655   CONTINUE
660   CO(4,1)=0.0
      XNBR=INBR
670   DO 720 M2=2,M
      S=0.0
```

```
      DXN=X(M2)-X(M2-1)
      DYN=Y(M2)-Y(M2-1)
      IF (ABSF(DYN)-ABSF(DXN)) 680,680,690
680   DN=0.5*DXN/XNBR
      GO TO 700
690   DN=0.5*DYN/XNBR
700   DO 710 M3=1,INBR
      DXM=2*M3-1
      DXM=DXM*DN
710   S=S+SQRTF(1.0+(CO(1,M2-1)+(2.0*CO(2,M2-1)+3.0*CO(3,M2-1)*DXM)*DXM)
     1**2)
720   CO(4,M2)=CO(4,M2-1)+2.0*ABSF(DN)*S
750   RETURN
      END (0,1,1,1,0)
```