437 | Oktober 1983

SCHRIFTENREIHE SCHIFFBAU

J.-S. Kouh

Routinen zur Kurvendarstellung mit kubischen Splines

TUHH
Technische Universität Hamburg-Harburg

Routinen zur Kurvendarstellung mit kubischen Splines

J.-S.Kouh., Hamburg, Technische Universität Hamburg-Harburg, 1983

Schriftenreihe Schiffbau
Schwarzenbergstraße 95c
D-21073 Hamburg

http://www.tuhh.de/vss

Institut für Schiffbau der Universität Hamburg

Routinen zur Kurvendarstellung mit kubischen Splines

von

J.-S. Kouh

Oktober 1983

Bericht Nr. 437

Inhaltsverzeichnis

Zusammenfassung

Kubische Splines werden wegen ihrer Einfachheit häufig zur numerischen Beschreibung "empirischer" Kurven zwecks Verarbeitung im Computer verwendet. In schiffbaulichen und ähnlichen Anwendungen haben sie eine große Verbreitung und sich gut bewährt. Der vorliegende Bericht umfaßt drei Pakete von Fortran-Unterprogrammen zur Kurvendarstellung mit kubischen Splines, die sich durch den Formeltyp zur Beschreibung der Kurven unterscheiden. Die Pakete dienen dazu, durch eine Folge von Stützpunkten eine Kurve zu interpolieren oder graphisch darzustellen. Die verschiedenen Pakete haben spezifische Vor- und Nachteile; deshalb wird es dem Benutzer der Routinen überlassen, ein geeignetes Paket für seine Aufgabe zu wählen.

1. Allgemeines

Aufgabe der hier beschriebenen Software ist es, durch in der Ebene oder im Raum vorgegebenen Punkte Kurven zu legen, die einen glatten, "natürlichen" Verlauf haben. Bei Verarbeitung mit dem Computer werden die Kurven durch Formeln beschrieben. Einfach läßt sich eine Kurve durch N gegebene Punkte z.B. mit einem Polynom N-1-ten Grades darstellen. Da Polynome höheren Grades wellig werden und z.B. Kurven, die sowohl gekrümmte als auch gerade Stücke enthalten, nicht durch ein einziges Polynom beschrieben werden können, erweist sich insbesondere die Methode der stückweisen Interpolation als zweckmäßig. Dabei wird eine Kurve aus mehreren Kurvenstücken zusammengesetzt. Jedes Kurvenstück wird durch ein Polynom niedrigen Grades beschrieben. Die in dieser Weise dargestellten Kurven nennt man Splines. Man spricht von kubischen Splines, wenn die Polynome vom 3. Grad sind. Kubische Splines werden meistens zur Darstellung empirischer Kurven gewählt, weil sie die einfachsten Kurven sind, die einen glatten Verlauf ergeben. Physikalisch entspricht ein kubisches Spline einer Straklatte, die an den Stützstellen festgehalten ist. Analog zur Straklatte kann ein glatter Kurvenverlauf durch folgende Bedingungen gewährleistet werden:

1. Die Polynome müssen an ihren Verbindungsstellen durch den gleichen Punkt gehen, d.h. keine Sprungstellen haben.
2. Die Polynome müssen an ihren Verbindungsstellen gleiche Tangenten, d.h. keine Knickstellen haben.
3. Die Polynome müssen an ihren Verbindungsstellen im Regelfall auch gleiche Krümmung haben.

Die Koeffizienten der einzelnen Polynome sind aus diesen Bedingungen zu bestimmen.

Es gibt verschiedene Arten von kubischen Splines, z.B.:

Darstellungsart	Ebene Kurve	Raumkurve
explizit	Y = f(X)	Y = f(X) Z = g(X)
in Parameterform	X = X(u) Y = Y(u)	X = X(u) Y = Y(u) Z = Z(u)

Jede dieser Darstellungsarten hat ihre Vor- und Nachteile. Deshalb sind drei Softwarepakete erstellt worden:

1. für ebene Kurven in expliziter Form
2. für ebene Kurven in Parameterform
3. sowohl für ebene als auch für Raumkurven in Parameterform mit gebrochenen Polynomen.

Die hier entwickelten Kurvendarstellungen in Parameterform sind invariant gegenüber Verschiebungen und Drehungen des Koordinatensystems sowie gegenüber Maßstabsänderungen mit gleichem Maßstabsfaktor in allen Koordinatenrichtungen. Dagegen ist die Kurvendarstellung in expliziter Form invariant gegenüber Verschiebungen und gegenüber Maßstabsänderungen auch mit verschiedenen Maßstabsfaktoren in den einzelnen Koordinatenrichtungen, dafür aber nicht invariant gegenüber Koordinatensystemdrehungen. Dementsprechend sollte auch die Anwendung der Softwarepakete erfolgen. Will man z.B. eine mathematische Funktion Y einer reellen Variablen X plotten, so ist die Forderung nach Unabhängigkeit der Kurve vom Maßstab der Ordinaten (auch bei konstantem Abszissenmaßstab) zu stellen und daher das Softwarepaket für explizite Darstellung zu wählen; soll dagegen ein krummlinig begrenzter Gegenstand dargestellt werden, so ist Unabhängigkeit von der Orientierung des Koordinatensystems zu fordern und daher die Parameterdarstellung zu bevorzugen. Die Parameterdarstellung ist z.B. für geschlossene Kurven geeignet; die explizite Darstellung setzt voraus, daß zu jedem Abszissenwert X des Definitionsbereichs des Splines genau eine Ordinate Y gehört, und daß dY/dX an allen Punkten des Definitionsbereichs existiert (also insbesondere keine vertikalen Tangenten auftreten).

Zur expliziten Darstellung räumlicher Kurven wird empfohlen, das Paket für die explizite Darstellung ebener Kurven zu benutzen und damit getrennt die zwei Projektionen Y(X) und Z(X) darzustellen. Entsprechend kann auch das Paket 2 für die Darstellung von Raumkurven in zwei Projektionen verwendet werden. Das Paket 3, das gebrochene Polynome verwendet, ist umfangreicher und rechenaufwendiger, bietet dafür aber z.B. die Möglichkeit, beliebige Kegelschnitte exakt darzustellen (also insbesondere auch Kreise) und den Kurvenverlauf auch bei unveränderten Stützpunkten und Tangentenrichtungen in allen Stützpunkten noch vielfältig zu beeinflussen.

Die Softwarepakete bestehen aus Unterprogrammen für folgende Aufgaben:

- Einlesen bzw. Aufbereiten der Daten, die eine Kurve definieren
- Bestimmung der Polynomkoeffizienten
- Interpolation von Koordinaten
- Zeichnen von Kurven

Im folgenden wird zunächst auf die einzelnen Verfahren eingegangen. Anschließend werden ein Beispielprogramm, seine Ergebnisse und die zugehörigen Routinen nach der o.a. Einteilung aufgelistet und ihre Zwecke und Funktionsweisen erläutert. Bei der Erstellung der Software wurde eine modulare Programmierung angestrebt, so daß zur Lösung einzelner Aufgaben jeweils nur ein Teil der Routinen benötigt wird.

Die Benutzung der Softwarepakete setzt voraus, daß N Stützpunkte $\vec{P}_i = (X_i, Y_i)$ bzw. $\vec{P}_i = (X_i, Y_i, Z_i)$, i = 1 bis N gegeben sind, und und daß einige weitere Angaben (End- bzw. Nebenbedingungen) gemacht werden.

-

2. Ebene kubische Splines in expliziter Form

Ein Kurvenstück wird in einzelnen Abszissenintervallen $X_i \leqslant X \leqslant X_{i+1}$, i = 1 bis N jeweils zwischen 2 Stützpunkten explizit dargestellt durch

$$Y(X) = a_i + b_i X + c_i X^2 + d_i X^3 \tag{1}$$

Die Koeffizienten a_i, b_i, c_i und d_i jedes Kurvenstücks werden so bestimmt, daß die eingangs angegebenen Bedingungen für einen glatten Kurvenverlauf erfüllt sind. Dabei kann die Formel (1) durch eine Koordinatentranslation vereinfacht werden:

$$Y(X) = Y_i + e_i(X-X_i) + f_i(X-X_i)^3 + g_i(X-X_i)^3 \tag{2}$$

Es bleiben also drei Koeffizienten pro Kurvenstück zu bestimmen. Eine ausführliche Erläuterung der Theorie und Algorithmen zur Bestimmung der Koeffizienten ist in [1] angegeben. Zu dem Paket gehören 8 Routinen. Die Routinen VKTABC, VEKTEQ, SUBTRV und MAMU sind problemunabhängige Hilfsroutinen.

1) Eingangsdaten

- Die Stützpunkte und zwei Kennzahlen zur Angabe der Endbedingungen werden vom Benutzerprogramm (siehe Beispielprogramm) eingelesen oder berechnet.

2) Bestimmung von Koeffizienten

- Die Subroutine KOEF ermittelt die zu einem Kurvenstück gehörigen Koeffizienten e_i, f_i und g_i der Formel (2) und speichert sie in einem Feld XI.

  Die Parameter der Subroutine
  KOEF(X,Y,N,IB1,IBN,XI,HF)
  haben folgende Bedeutung:
  Eingangswerte:

  X - Feld mit Dim. N; enthält X-Werte (Abszissen) der Stützpunkte

  Y - Feld mit Dim. N; enthält Y-Werte (Ordinaten) der Stützpunkte

N - Anzahl der Stützpunkte ($N \geqslant 2$)

IB1 - Kennzahl für Anfangsbedingung

= 1 für Y''' = O, d.h. die Krümmung ist im ersten Abschnitt etwa konstant

= 2 für Y'' = O, d.h. die Krümmung ist im ersten Stützpunkt gleich O.

= 3 für vorgegebenen Tangentenwinkel am ersten Stützpunkt. Der Winkel ist in Grad in das Feldelement XI(1,1) einzugeben. Er wird von der X-Achse gemessen (+ im entgegengesetzten Sinne des Uhrzeigers, sonst -).

IBN - Kennzahl für Endbedingung am letzten Stützpunkt (1, 2 oder 3 wie bei IB1).

Ergebnisse:

XI - enthält die Koeffizienten e_i, f_i und g_i der Formel (2) für i = 1 bis N - 1. Das Feld ist mit 3 * (N-1) zu dimensionieren.

Hilfsfeld:

HF - mindestens mit Dimension 2 * N

3) Interpolation

- Die Subroutine INTPL1 berechnet Y- oder dY/dX-Werte oder Krümmungen zu gegebenen X-Werten. Dabei wird zunächst das Intervall festgestellt, in dem X liegt. Mit den zugehörigen Koeffizienten wird der gewählte o.a. Wert berechnet.

Die Parameter der Subroutine
INTPL1(X,Y,N,U,V,NI,XI,MOD)
haben folgende Bedeutung:
Eingangsparameter:

X - siehe die Subroutine KOEF

Y - siehe die Subroutine KOEF

N - siehe die Subroutine KOEF

U - Feld mit Dim. NI enthält die zu interpolierenden X-Werte

NI - Anzahl der zu interpolierenden Punkte

XI - siehe die Subroutine KOEF

MOD - Kennzahl zu den zuberechnenden Daten
1, Y-Werte, d.h. Ordinaten
2, dY/dX, d.h. Steigung
3, $Y''/(1+Y'^2)^{3/2}$, d.h. Krümmung

Ausgangsparameter:

V - Feld mit Dim. NI enthält die MOD entsprechenden Daten.

4) Zeichnen

- Die Subroutine IDRAW1 erzeugt abhängig von einem Maßstab Z (Z = wahre Länge in cm/gezeichnete Länge in cm) aus einer Kurve einen Polygonzug, dessen Abweichungen von der Kurve, ein gegebenes Maß a (z.B. a = 0.05 mm), nicht überschreiten darf.

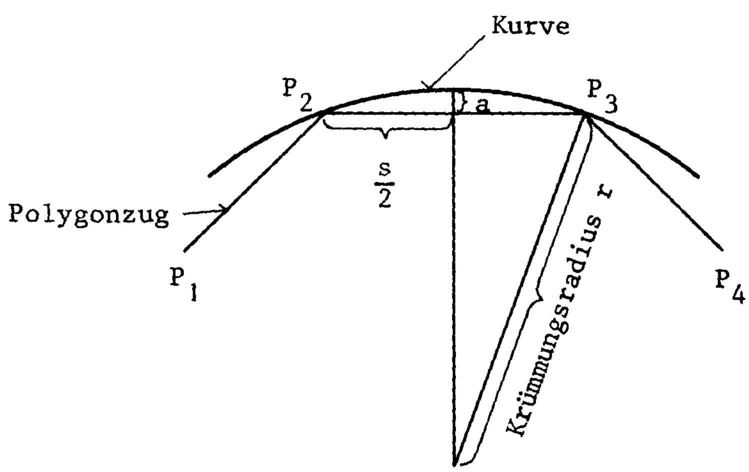


Bild 1

Wird der Krümmungsradius bei Z = 1 mit r bezeichnet, so ergibt sich der Abstand zwischen zwei benachbarten Polygonecken (P_2 und P_3 im Bild 1) etwa nach folgender Formel:

$$s = \sqrt{8 \times a \times r \times Z}$$

Der erzeugte Polygonzug wird dann von der Subroutine LINE geplottet.

Die Parameter der Subroutine
IDRAW1(X,Y,XI,N,ZM,AC)
haben folgende Bedeutung:

X bis N siehe die Subroutine KOEF

ZM - Zeichenmaßstab, entspricht Z

AC - gegebenes Maß a in cm

```
      DIMENSION X(100),Y(100),XI(300)
      DIMENSION U(100),V(100),HF(200)
C   ANZAHL VON STUETZPUNKTEN N, KENNZAHL FUER ENDBEDING. AM 1. UND N.
C   STUETZPUNKT IBC1 UND IBCN, SOWIE ANZAHL VON ZU INTERP. PUNKTEN NI
      READ(5,*)N,IB1,IBN,NI
      WRITE(6,150)N,IB1,IBN,NI
C   X-WERTE DER STUETZPUNKTE EINLESEN
      READ(5,*)(X(I),I=1,N)
      WRITE(6,200)(X(I),I=1,N)
C   Y-WERTE DER STUETZPUNKTE EINLESEN
      READ(5,*)(Y(I),I=1,N)
      WRITE(6,200)(Y(I),I=1,N)
C   GGF. TANGENTENWINKEL AM 1. PKT. EINLESEN
      IF(IB1.EQ.3)READ(5,*)XI(1)
      IF(IB1.EQ.3)WRITE(6,200)XI(1)
C   GGF. TANGENTENWINKEL AM N. PKT. EINLESEN
      IF(IBN.EQ.3)READ(5,*)XI(2)
      IF(IBN.EQ.3)WRITE(6,200)XI(2)
C   KOEFFIZIENTEN FUER KUBISCHE SPLINES BESTIMMEN
      CALL KOEF(X,Y,N,IB1,IBN,XI,HF)
      NI1=NI-1
      DU=(X(N)-X(1))/NI1
      U(1)=X(1)
      DO 10 J=1,NI1
      U(J+1)=U(J)+DU
   10 CONTINUE
C   ORDINATEN BERECHNEN
      CALL INTPL1(X,Y,N,U,V,NI,XI,1)
      WRITE(6,200)(U(I),I=1,NI)
      WRITE(6,200)(V(I),I=1,NI)
C   STEIGUNGEN BERECHNEN
      CALL INTPL1(X,Y,N,U,V,NI,XI,2)
      WRITE(6,200)(U(I),I=1,NI)
      WRITE(6,200)(V(I),I=1,NI)
C   KRUEMMUNGEN BERECHNEN
      CALL INTPL1(X,Y,N,U,V,NI,XI,3)
      WRITE(6,200)(U(I),I=1,NI)
      WRITE(6,200)(V(I),I=1,NI)
C   ZEICHNENMASSSTAB ZM, GENAUIKEIT AC FUER PLOT EINLESEN
      READ(5,*)ZM,AC
      WRITE(6,200)ZM,AC
C   PLOT INITIIREN
      CALL PLOTS(0,0,11)
C   URSPRUNG DEFINIEREN
      CALL PLOT(3.,5.,-3)
C   URSPRUNG MIT SYMBOL + MARKIEREN
      CALL SYMBOL(0.,0.,0.7,3,0.,-1)
C   KURVE ZEICHNEN
      CALL IDRAW1(X,Y,XI,N,ZM,AC)
C   PLOT ABSCHLIESSEN
      CALL PLOT(20.,25.,999)
  150 FORMAT(1X,6I10)
  200 FORMAT(1X,6F11.5)
      STOP
      END
```

5	3	1	21			
0.00000	1.00000	2.00000	3.00000	4.00000	Eingabedaten	
1.00000	0.00000	0.00000	0.00000	0.00000		
0.00000						
0.00000	0.20000	0.40000	0.60000	0.80000	1.00000	Ab-
1.20000	1.40000	1.60000	1.80000	2.00000	2.20000	szissen
2.40000	2.60000	2.80000	3.00000	3.20000	3.40000	
3.60000	3.80000	4.00000				
1.00000	0.92176	0.72527	0.46790	0.20702	0.00000	Ordi-
-0.11005	-0.13698	-0.10888	-0.05385	0.00000	0.03044	naten
0.03863	0.03161	0.01639	0.00000	-0.01171	-0.01756	
-0.01756	-0.01171	0.00000				
0.00000	0.20000	0.40000	0.60000	0.80000	1.00000	Ab-
1.20000	1.40000	1.60000	1.80000	2.00000	2.20000	szissen
2.40000	2.60000	2.80000	3.00000	3.20000	3.40000	
3.60000	3.80000	4.00000				
0.00000	-0.73463	-1.18244	-1.34341	-1.21756	-0.80488	Stei-
-0.31902	0.02634	0.23122	0.29561	0.21951	0.09073	gungen
-0.00293	-0.06146	-0.08488	-0.07317	-0.04390	-0.01463	
0.01463	0.04390	0.07317				
0.00000	0.20000	0.40000	0.60000	0.80000	1.00000	Ab-
1.20000	1.40000	1.60000	1.80000	2.00000	2.20000	szissen
2.40000	2.60000	2.80000	3.00000	3.20000	3.40000	
3.60000	3.80000	4.00000				
-4.39024	-1.54728	-0.40981	-0.01869	0.34422	1.31448	Krüm-
1.79686	1.37418	0.62258	-0.02581	-0.68183	-0.54930	mungen
-0.38048	-0.20373	-0.02896	0.14517	0.14592	0.14629	
0.14629	0.14592	0.14517				
0.25000	0.00100					

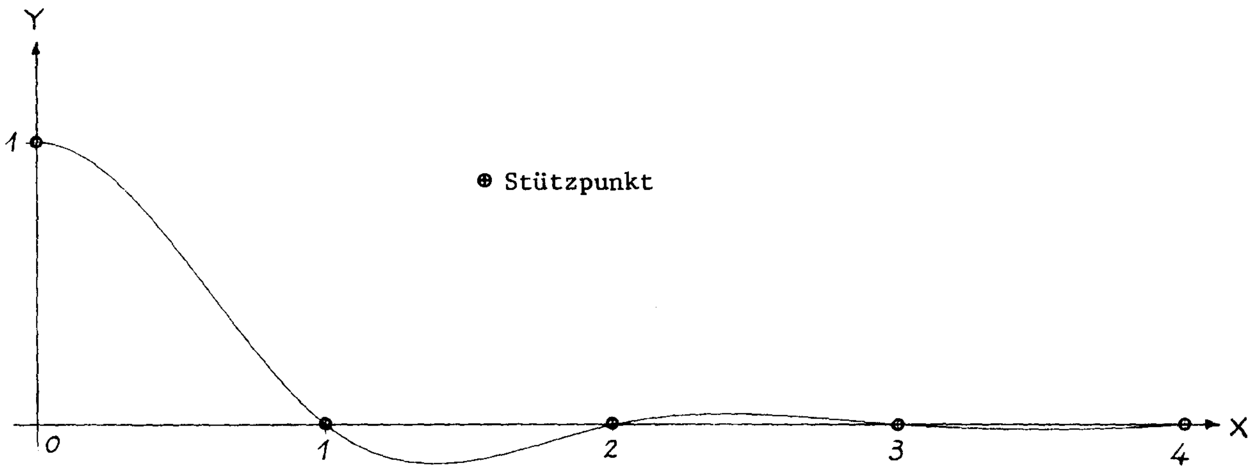


Bild 2

```
      SUBROUTINE KOEF(X,Y,N,IB1,IBN,XI,HF)
C  ZUM ERMITTELN KOEFFIZIENTE POLYNOM 3. GRADES
C   BESCHREIBUNG DER PARAMETER:
C    EINGANG:
C     X  -FELD MIT DIM. N ENTHAELT X-WERTE (ABSZISSE)
C     Y  -FELD MIT DIM. N ENTHAELT Y-WERTE (ORDINATE)
C     N  -ANZAHL DER GEGEB. PUNKTE
C     IB1-KENNZAHL FUER ANFANGSBEDINGUNG
C     IBN-KENNZAHL FUER ENDEBEDINGUNG
C          IB1, IBN=1 ODER 2 ODER 3
C          1, Y'''=0
C          2, Y''=0
C          3, TANGENTENWINKEL GEGEB.
C             FUER IB1 WINKEL IN XI(1,1), IBN IN XI(2,1) EINGEB.
C    AUSGANG:
C     XI -FELD MIT DIM. 3*(N-1) ENTHAELT KOEFF. POLYN. 3. GRADES
C    HILFSFELD:
C     HF -FELD MIT MIND. DIM. 2*N
C   BENOETIGTEN ROUTINEN: MATX, VKTABC, VEKTEQ, SUBTRV, MAMU
C  LIT: VORLESUNGSMANUSKRIPT KGS VON SOEDING,
C       ESS-BERICHT NR. 22, 1977, S. 5.6
C
      DIMENSION X(N),Y(N),XI(3,1),HF(1)
      DIMENSION XI0(3),XI1(3),XIO(3),XI1(3),XN0(3),XN1(3)
      DIMENSION XR(3),XZ1(3),XZ2(3),XZ3(3),CM(3,3)
      DIMENSION XIC0(3),XIC1(3)
      DATA RAD/0.0174532925/,XI0/2*0.,1./,XI1/1.,2*0./
      DATA CM(1,1),CM(2,2),CM(3,1),CM(3,2),CM(3,3)/2*-2.,2*0.,1./
C
C  1. SCHRITT
      XI0(3)=1.
      XI1(3)=0.
      DX=X(2)-X(1)
      DY=Y(2)-Y(1)
      GOTO(110,120,130),IB1
  110 XI0(1)=DY/DX
      XI0(2)=0.
      XI1(1)=1.
      XI1(2)=-1./DX
      GOTO 150
  120 XI0(1)=0.
      XI0(2)=0.
      XI1(1)=1.
      XI1(2)=0.
      GOTO 150
  130 XI0(1)=TAN(RAD*XI(1,1))
      XI0(2)=0.
      XI1(1)=0.
      XI1(2)=1.
C
  150 N1=N-1
      N2=N-2
      XN0(3)=1.
      XN1(3)=0.
      DX=X(N)-X(N1)
      DY=Y(N)-Y(N1)
      GOTO(160,170,180),IBN
  160 XN0(1)=DY/DX
      XN0(2)=0.
      XN1(1)=1.
      XN1(2)=-1./DX
```

```
      GOTO 190
  170 XN0(1)=DY/DX
      XN0(2)=0.
      XN1(1)=1.
      XN1(2)=-1.5/DX
      GOTO 190
  180 XN0(1)=1.5*DY/DX
      XN0(2)=-TAN(RAD*XI(2,1))/DX
      XN1(1)=1.
      XN1(2)=-2./DX
C
C  2. SCHRITT
  190 CALL VEKTEQ(X10,XI(1,1),3)
      CALL VEKTEQ(X11,XZ1,3)
      BS=0.
      DO 200 I=2,N1
      DX=X(I)-X(I-1)
      DY=Y(I)-Y(I-1)
      CALL MATX(DX,DY,CM)
      CALL MAMU(CM,XZ1,XZ2,3,3,1)
      CALL MAMU(CM,XI(1,I-1),XZ3,3,3,1)
      IF(I.EQ.N1)GOTO 192
      CALL VEKTEQ(XI0,XIC0,3)
      CALL VEKTEQ(XI1,XIC1,3)
      GOTO 194
  192 CALL VEKTEQ(XN0,XIC0,3)
      CALL VEKTEQ(XN1,XIC1,3)
  194 CALL SUBTRV(XIC0,XZ3,XR,3)
      ZR=-XZ2(1)*XIC1(2)+XZ2(2)*XIC1(1)
      HF(I)=(-XR(1)*XIC1(2)+XR(2)*XIC1(1))/ZR
      HF(N+I)=(XZ2(1)*XR(2)-XZ2(2)*XR(1))/ZR
      BS=BS+HF(I)
      CALL VKTABC(XIC0,XIC1,HF(N+I),XI(1,I),3)
      CALL VEKTEQ(XZ2,XZ1,3)
  200 CONTINUE
C
C  3. SCHRITT
      CALL VEKTEQ(X11,XZ1,3)
      CALL VKTABC(XI(1,1),XZ1,BS,XI(1,1),3)
      DO 300 I=2,N2
      DX=X(I)-X(I-1)
      DY=Y(I)-Y(I-1)
      CALL MATX(DX,DY,CM)
      CALL MAMU(CM,XZ1,XZ2,3,3,1)
      BS=BS-HF(I)
      CALL VKTABC(XI(1,I),XZ2,BS,XI(1,I),3)
      DX=X(I+1)-X(I)
      DY=Y(I+1)-Y(I)
      XI(3,I)=((DY/DX-XI(1,I))/DX-XI(2,I))/DX
      CALL VEKTEQ(XZ2,XZ1,3)
  300 CONTINUE
      DX=X(2)-X(1)
      XI(3,1)=(((Y(2)-Y(1))/DX-XI(1,1))/DX-XI(2,1))/DX
      DX=X(N)-X(N1)
      XI(3,N1)=(((Y(N)-Y(N1))/DX-XI(1,N1))/DX-XI(2,N1))/DX
      RETURN
      END
```

```
      SUBROUTINE INTPL1(X,Y,N,U,V,NI,XI,MOD)
C  ZUM INTERPOLIEREN MIT ERMITTELTEN KOEFFIZIENTEN
C   BESCHREIBUNG DER PARAMETER:
C    EINGANG:
C     X   -FELD MIT DIM. N ENTHAELT X-WERTE (ABSZISSE)
C     Y   -FELD MIT DIM. N ENTHAELT Y-WERTE (ORDINATE)
C     N   -ANZAHL DER GEGEB. PUNKTE
C     U   -FELD MIT DIM. NI ENTHAELT ZU INTERPOL. X-WERTE (ABSZISSE)
C     NI  -ANZAHL DER ZU INTERPOL. PUNKTE
C     XI  -FELD MIT DIM. 3*(N-1) ENTHAELT KOEFF. POLYN. 3. GRADES
C     MOD-KENNZAHL DER ZU ERMITTELNDEN DATEN
C         1; ORDINATEN
C         3; KRUEMMUNGEN
C    AUSGANG:
C     V   -FELD MIT DIM. NI ENTHAELT NACH MOD BESTIMMTEN DATEN
C
      DIMENSION X(N),Y(N),U(NI),V(NI),XI(3,1)
C
      N1=N-1
      DO 300 I=1,NI
      DO 100 J=1,N1
      IF((U(I).GE.X(J).AND.U(I).LT.X(J+1)).OR.U(I).LT.X(1))GOTO 200
  100 CONTINUE
      J=J-1
  200 DX=U(I)-X(J)
      DX=U(I)-X(J)
      GOTO (220,240,260),MOD
  220 V(I)=Y(J)+((XI(3,J)*DX+XI(2,J))*DX+XI(1,J))*DX
      GOTO 300
  240 V(I)=(3.*XI(3,J)*DX+2.*XI(2,J))*DX+XI(1,J)
      GOTO 300
  260 V(I)=(6.*XI(3,J)*DX+2.*XI(2,J))/
     *       (1.+((3.*XI(3,J)*DX+2.*XI(2,J))*DX+XI(1,J))**2)**1.5
  300 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE IDRAW1(X,Y,XI,N,ZM,AC)
C  ZUM ZEICHNEN EINER KURVE, DIE DURCH KUBISCHE SPLINE
C  DARGESTELLT IST.
C  BESCHREIBUNG DER PARAMETER:
C    X BIS N-SIEHE SUBROUTINE KOEF
C    ZM      -ZEICHENMASSSTAB
C    AC      -GENAUIKEIT ZUM ZEICHENE EINER KURVE DURCH GERAUEN
C
      DIMENSION X(N),Y(N),XI(3,1),XZ(202),YZ(202),DQY(4)
      DATA DTT/0.1/,NTT/11/
C
      N1=N-1
      DO 300 I=1,N1
      DX=X(I+1)-X(I)
      RL=SQRT(DX**2+(Y(I+1)-Y(I))**2)
C  KRUEMMM. AN NTT STELLEN BERECHNEN, DIE GROESTE AUSSUCHEN
      RKM=0.
      T=0.
      DO 100 J=1,NTT
      DDX=T*DX
      RKM=AMAX1(RKM,(6.*XI(3,I)*DDX+2.*XI(2,I))/(1.+
     *  ((3.*XI(3,I)*DDX+2.*XI(2,I))*DDX+XI(1,I))**2)**1.5)
      T=T+DTT
  100 CONTINUE
C  DIE MASSSTAB UND GENAUIGKEIT ENTSP. PUNKTZAHL BESTIMMEN
      NP=IFIX(AMAX1(1.,RL/SQRT(8.*ZM*AC/AMAX1(RKM,1.E-8)))+2.)
      DT=1./(NP-1)
C  PUNKTKOORDINATEN ZUM ZEICHNEN BERECHNEN MIT DIFFERERZENVERF.
      XZ(1)=X(I)
      YZ(1)=Y(I)
      XZ(NP)=X(I+1)
      YZ(NP)=Y(I+1)
      IF(NP.LE.2)GOTO 250
      DDX=DT*DX
      DQY(1)=Y(I)
      DQY(2)=((XI(3,I)*DDX+XI(2,I))*DDX+XI(1,I))*DDX
      DQY(3)=2.*(3.*XI(3,I)*DDX+XI(2,I))*DDX**2
      DQY(4)=6.*XI(3,I)*DDX**3
      NP1=NP-1
      DO 200 J=2,NP1
      DO 150 K=1,3
      DQY(K)=DQY(K)+DQY(K+1)
  150 CONTINUE
      XZ(J)=XZ(J-1)+DDX
      YZ(J)=DQY(1)
  200 CONTINUE
  250 CONTINUE
      XZ(NP+1)=0.
      YZ(NP+1)=0.
      XZ(NP+2)=ZM
      YZ(NP+2)=ZM
C  ZEICHNEN EIN KURVENSTUECK
      CALL LINE(XZ,YZ,NP,1,0,0)
  300 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MATX(DX,DY,CM)
C  ZUM ZUWEISEN MATRIX CM
      DIMENSICN CM(3,3)
C
      CM(2,1)=-3./DX
      CM(1,2)=-DX
      CM(1,3)=3.*DY/DX
      CM(2,3)=3.*DY/DX**2
      RETURN
      END
```

```
      SUBROUTINE VKTABC(A,B,C,D,N)
C  A, B, C SIND VEKTOREN; C SKALAR.  D = A + C*B
      DIMENSION A(N),B(N),D(N)
      DO 100 I=1,N
      D(I)=A(I)+C*B(I)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE VEKTEQ(A,B,N)
C  ZUM GLEICHSETZEN ZWEIER VEKTOREN
      DIMENSION A(N),B(N)
      DO 100 I=1,N
      B(I)=A(I)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SUBTRV(A,B,C,N)
C  VEKTORSUBTRAKTION A - B = C
      DIMENSION A(N),B(N),C(N)
      DO 100 I=1,N
      C(I)=A(I)-B(I)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MAMU(X,Y,Z,IX,JX,JY)
C  MATRIXMULTIPLIKATION Z = X * Y
      DIMENSION X(IX,JX),Y(JX,JY),Z(IX,JY)
      DO 200 I=1,IX
      DO 200 K=1,JY
      SUM=0.
      DO 100 J=1,JX
  100 SUM=SUM+X(I,J)*Y(J,K)
  200 Z(I,K)=SUM
      RETURN
      END
```

3. Ebene kubische Splines in Parameterform

Ein Kurvenstück zwischen zwei Stützpunkten $\vec{P}_i = (X_i, Y_i)$ und $\vec{P}_{i+1} = (X_{i+1}, Y_{i+1})$ wird abhängig von einem Parameter u dargestellt durch

$$X(u) = a_{xi} + b_{xi}u + c_{xi}u^2 + d_{xi}u^3$$

$$Y(u) = a_{yi} + b_{yi}u + c_{yi}u^2 + d_{yi}u^3 \qquad (3)$$

Die eingangs genannten drei Bedingungen (gegebene Stützpunkte sowie stetige Steigung und Krümmung der Kurve in den Stützpunkten) reichen nicht aus, um die 8 Koeffizienten von (3) eindeutig zu bestimmen. Deshalb wird die zusätzliche Forderung aufgestellt, daß in dem lokalen Koordinatensystem ξ, η (siehe Bild 3) η ein Polynom 3. Grades von ξ sein soll.

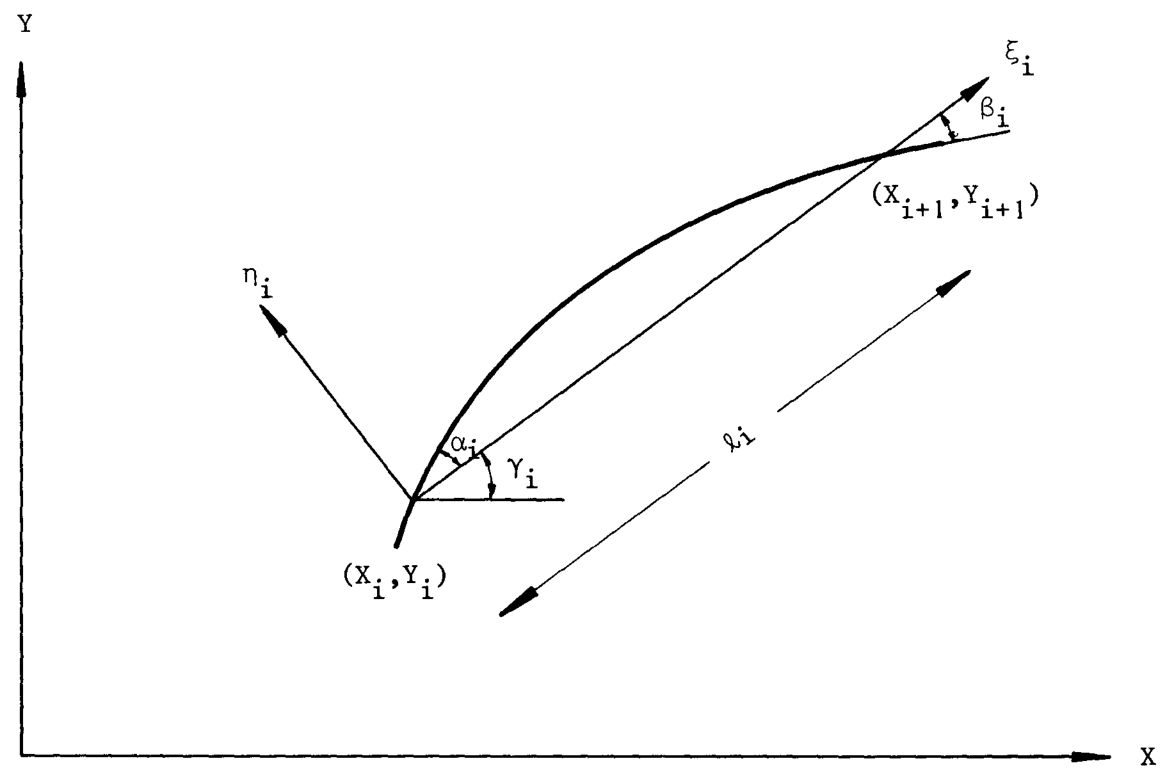


Bild 3

Mit Hilfe einer Koordinatentransformation kann das Kurvenstück durch folgende Formel im $\xi_i\eta_i$-Koordinatensystem dargestellt werden

$$\xi_i = \ell_i \times u$$

$$\eta_i = u(1-u)\left[e_i(1-u)-f_i(u)\right]$$

$$\text{mit} \quad \ell_i = \sqrt{(X_{i+1}-X_i)^2 + (Y_{i+1}-Y_i)^2} \tag{4}$$

Die Zusammenhänge zwischen dem XY-und $\xi\eta$-Koordinatensystem sind:

$$X = X_i + \xi_i \cos\gamma_i - \eta_i \sin\gamma_i$$

$$Y = Y_i + \xi_i \sin\gamma_i + \eta_i \cos\gamma_i$$

$$\text{mit} \quad \gamma_i = \text{arctg}\,\frac{Y_{i+1} - Y_i}{X_{i+1} - X_i} \tag{5}$$

Während die Winkel γ_i in (5) aus den gegebenen Stützpunkten ermittelt werden, sind die Koeffizienten e_i und f_i in (4) für jedes Kurvenstück aus der Bedingung für einen glatten Kurvenverlauf zu bestimmen.

Die Interpolation, d.h. die Berechnung von Y zu gegebenen X, erfordert größeren Rechenaufwand als bei der expliziten Kurvendarstellung, weil der Parameter u in (3) zu gegebenem X durch Lösung einer kubischen Gleichung zu bestimmen ist.

7 Routinen sind für das Verfahren in Paket 2 geschreiben worden:

1) Eingangsdaten

- Die Stützpunkte und die Angaben zu End- und Nebenbedingungen werden vom Benutzerprogramm (vgl. das Hauptprogramm des Beispiels) eingelesen oder berechnet.

2) Bestimmung von Koeffizienten

– Die Subroutine INANLE berechnet die Sehnenlänge ℓ_i und die Winkel γ_i. Dabei wird eine Rekursionsformel für die Berechnung der Winkel γ_i aus [2] übernommen.

Die Parameter der Subroutine
INANLE(X,Y,N,RL,GAMA)
haben folgende Bedeutung:
Eingangsparameter:

X - Feld mit Dim. N enthält X-Werte (Abszisse) der Stützpunkte

Y - Feld mit Dim. N enthält Y-Werte (Ordinate) der der Stützpunkte

N - Anzahl der Stützpunkte ($N \geqslant 2$)

Ausgangsparameter:

RL - Feld für die berechneten Sehnenlängen ℓ_i, $i = 1$ bis $N - 1$.

GAMA - Feld für die berechneten Winkel γ_i (in Rad.), $i = 1$ bis N.

Die Subroutine INTANG baut ein Gleichungssystem für die Winkel α_i, $i = 1$ bis N auf und löst das System mit der Subroutine GELB. Dabei müssen Kennzahlen für Endbedingungen, Nebenbedingungen und ggf. die Informationswerte zu den Bedingungen eingegeben werden.

Die Möglichkeiten zur Angabe der End- und Nebenbedingungen sind:

Kennzahl	1	2	3
Bedeutung	Krümmung am Endpunkt = ein Faktor a mal Krümmung am Nachbarpunkt	Krümmung k am Endpunkt wird vorgegeben	Tangentenwinkel ω am Endpunkt wird vorgegeben
zu der Kennzahl zugehöriger Informationswert	der Faktor a bei geradem Auslauf a = 0 bei krummem Auslauf a = 1 Bild 4	die Krümmung k $(k = \frac{1}{r})$; r ist Krümmungsradius in Längen-Einheit, die der der Koordinaten entspricht Bild 5	der Tangentenwinkel ω in Grad Bild 5

Tabelle 1

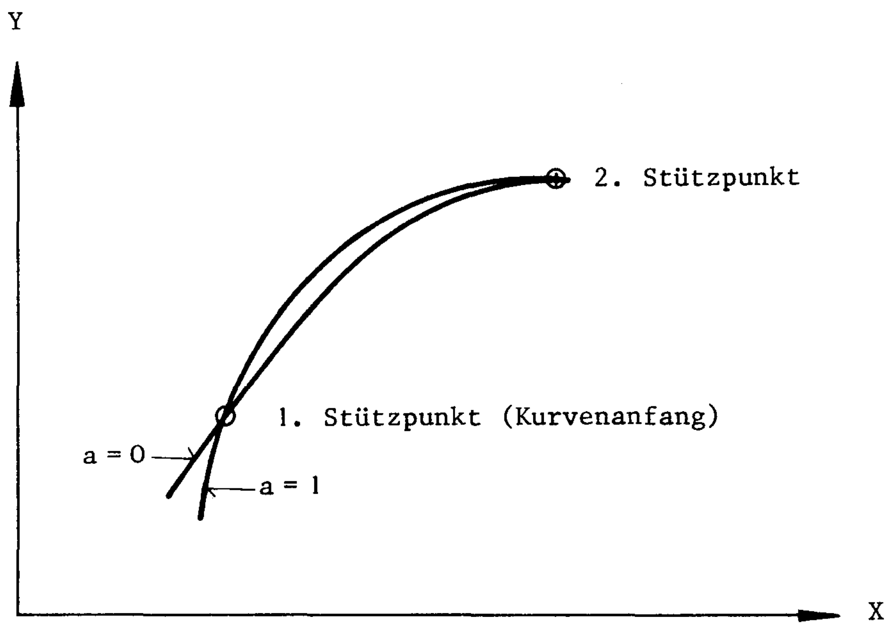


Bild 4

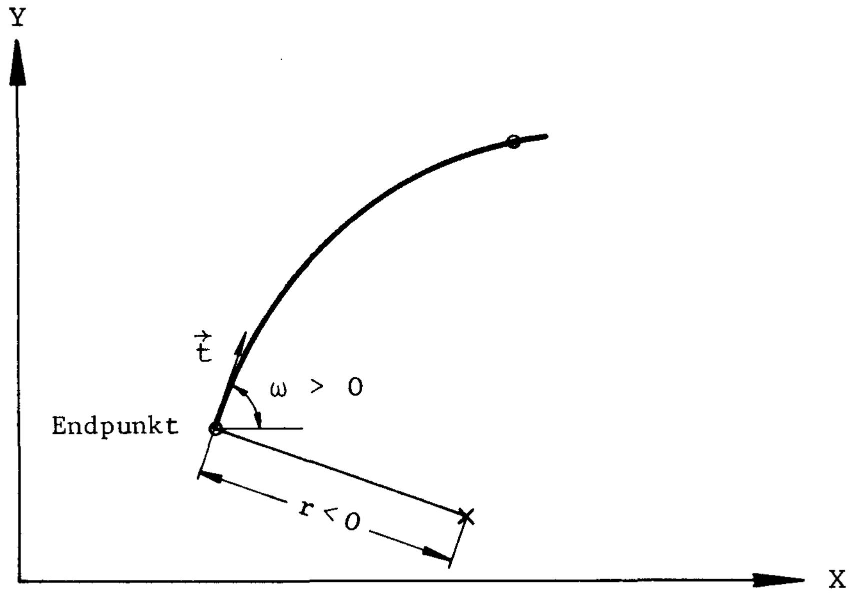


Bild 5

Die Tangentenwinkel und das Vorzeichen der Krümmung (auch Krümmungsradius) werden durch folgende Regel festgelegt:

- Am ersten Punkt der Kurven ist der Tangentenwinkel ω wie in Bild 5 angegeben definiert. An den folgenden Stützpunkten der Kurven sind die Tangentenwinkel größer als am jeweils vorhergehenden Punkt, wenn sich die Kurve bei Fortschreiten vom ersten zum letzten Stützpunkt nach links krümmt.
- Die Krümmung ist positiv, wenn sich die Kurve nach links krümmt.

b) an den Stützpunkten zwischen den Endpunkten

Kennzahl	0	1	2	3
Bedeutung	Krümmung stetig	Krümmung unstetig Der Kurvenbereich nach dem Stützpunkt beeinflußt nicht den Kurvenverlauf vor dem Stützpunkt	Krümmung unstetig Der Kurvenbereich vor dem Stützpunkt beeinflußt nicht den Kurvenverlauf nach dem Stützpunkt	vorgegebener Tangentenwinkel ω
Informationswert	entfällt Bild 6	entfällt Bild 6	entfällt Bild 6	der Tangentenwinkel ω in Grad Bild 7

Tabelle 2

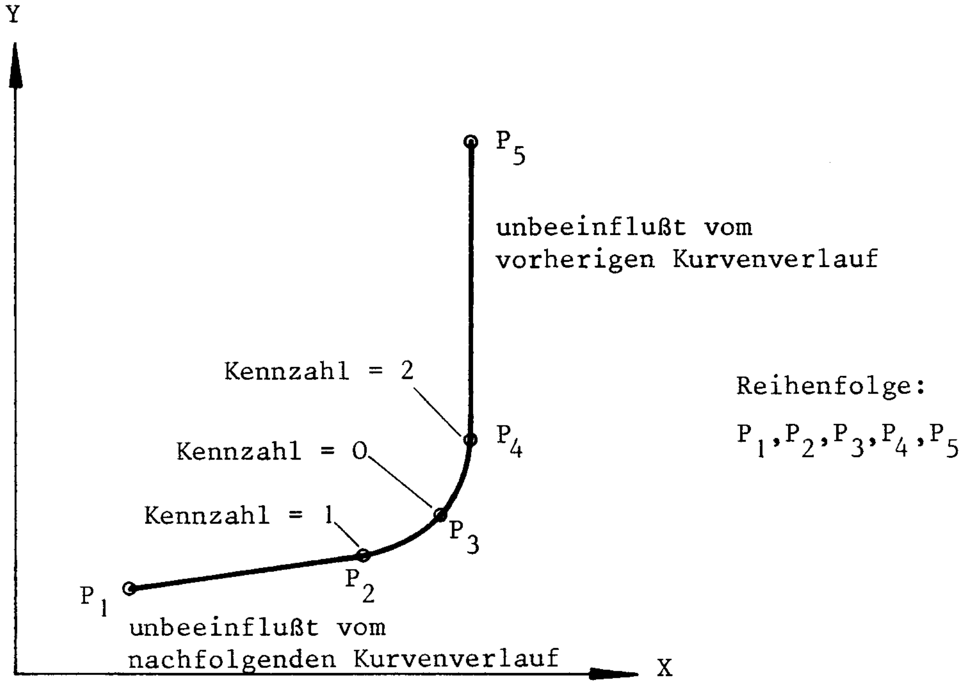

Y
P5
unbeeinflußt vom
vorherigen Kurvenverlauf
Kennzahl = 2
Reihenfolge:
P1,P2,P3,P4,P5
P4
Kennzahl = 0
Kennzahl = 1
P3
P1
P2
unbeeinflußt vom
nachfolgenden Kurvenverlauf
X

Bild 6

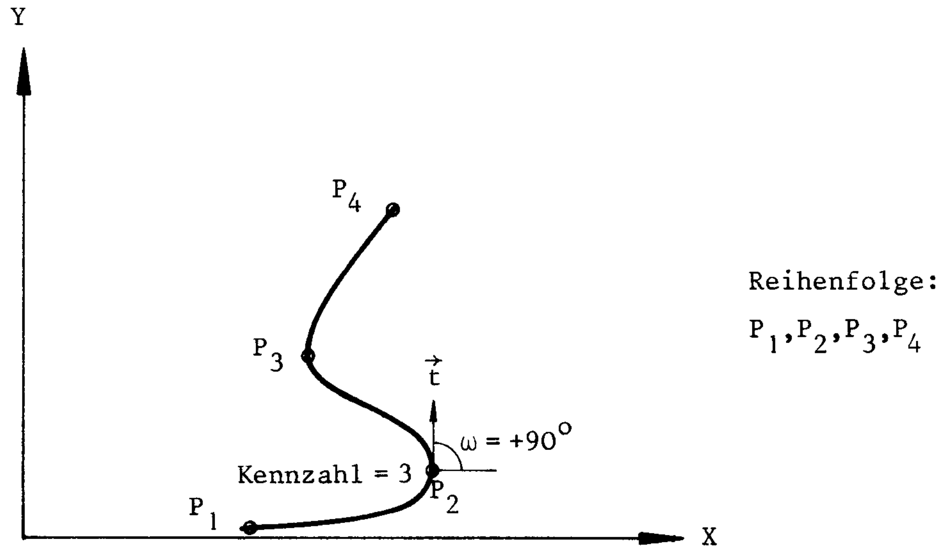

Y
P4
Reihenfolge:
P1,P2,P3,P4
P3
t
ω = +90°
Kennzahl = 3
P1
P2
X

Bild 7

Die Parameter der Subroutine
INTANG(N,IBC,RBC,RL,GAMA,ALFA,AA)
haben folgende Bedeutung:

Eingangsparameter:

N - Anzahl der Stützpunkte

IBC - Feld mit Dim. N für die Kennzahlen zu den End- bzs. Nebenbedingungen

RBC - Feld mit Dim. N für die den Kennzahlen entsprechenden Informationswerte

RL - siehe die Subroutine INANLE

GAMA - siehe die Subroutine INANLE

Ausgangsparameter:

ALFA - Feld mit Dim. N für die Winkel α_i (Bild 3), $i = 1$ bis N

Hilfsfeld:

AA - mindestens mit Dim. 3 * N

Die Subroutine GELB löst ein lineares Gleichungssystem mit Bandstruktur. Die Routine gehört zum Scientific Subroutine Package der IBM. Ihre Parameter sind mit Kommentarkarten ausführlich beschrieben.

3) Interpolation

- Die Subroutine INTPL2 berechnet die Ordinaten Y oder Steigungen dY/dX oder Krümmungen. Dabei wird das erste Intervall in der Laufrichtung der Kurve festgestellt, in dem die gegebene Abszisse X liegt. Danach werden der Parameter u mit Hilfe der Funktion PT ermittelt und die oben genannten geometrischen Größen berechnet.

  Die Parameter der Subroutine
  INTPL2(X,Y,ALFA,GAMA,N,XI,YI,NI,MOD)
  haben folgende Bedeutung:

  Eingangsparameter:

  X - siehe die Subroutine INANLE

Y - siehe die Subroutine INANLE

ALFA - siehe die Subroutine INTANG

GAMA - siehe die Subroutine INANLE

N - Anzahl der Stützpunkte

XI - Feld mit Dim. NI für die gegebenen X-Werte

NI - Anzahl der zu interpolierenden Punkte

MOD - Kennzahl der zu berechnenden geometrischen Größen
1. Ordinaten Y
2. Steigungen dY/dX
3. Krümmungen

Ausgangsparameter:

YI - Feld mit Dim. NI für die dem Eingangsparameter MOD entsprechenden Größen.

- Die Funktion PT löst eine kubische Gleichung für u iterativ mit dem Newtonschen Näherungsverfahren auf. Maximale Anzahl von Iterationsschritten ist auf 30 angesetzt.

  Die Parameter der Funktion
  PT(X,Y,AI,BI,XI)
  haben folgende Bedeutung:

  Eingangsparameter:

  X - X-Werte der zwei Endpunkte eines Kurvenstücks

  Y - Y-Werte der zwei Endpunkte eines Kurvenstücks

  AI - tg α_i (α_i siehe Bild 3)

  BI - tg β_i (β_i siehe Bild 3)

  XI - gegebener X-Wert, zu dem der Parameter u zu ermitteln ist.

4) Zeichnen

- Die Subroutine IDRAW2 (X,Y,ALFA,GAMA,N,ZM,AC) zeichnet eine Kurve entsprechend den zu IDRAW1 gemachten Angaben. Die Parameter X bis N sind in den Subroutinen INANLE und INTANG beschrieben. Die Parameter ZM und AC haben die gleichen Bedeutungen wie die der Subroutine IDRAW1.

- Die Funktion RKRMM berechnet die Krümmung zu einem gegebenen Parameter u bei einem Kurvenstück zwischen zwei Stützpunkten.

```
C   TESTPROGRAMM FUER DURCH KUBISCHE SPLINES IN
C   PARAMETERFORM DARGESTELLTEN KURVEN
      DIMENSION X(100),Y(100),ALFA(100),GAMA(100),IBC(100),RBC(100)
      DIMENSION RL(100),AA(300),XI(100),YI(100)
C
C   ANZAHL VON STUETZPUNKTEN N, ANZAHL VON ZU INTERP. PUNKTEN NI,
C   ANZAHL VON ANZUGEBENDEN NEBENBEDING. AN ZWISCHENPUNKTEN NBCH
      READ(5,*)N,NI,NBCH
      WRITE(6,*)N,NI,NBCH
C   X-WERTE DER STUETZPUNKTE EINLESEN
      READ(5,*)(X(I),I=1,N)
      WRITE(6,200)(X(I),I=1,N)
C   Y-WERTE DER STUETZPUNKTE EINLESEN
      READ(5,*)(Y(I),I=1,N)
      WRITE(6,200)(Y(I),I=1,N)
C   DATEN FUER END- BZW. NEBENBEDING. VOREINSTELLEN
      DO 40 I=1,N
      RBC(I)=0.
   40 IBC(I)=0
C   DATEN FUER DIE BEIDEN ENDBEDING. EINLESEN
      READ(5,*)IBC(1),IBC(N),RBC(1),RBC(N)
      WRITE(6,*)IBC(1),IBC(N),RBC(1),RBC(N)
      IF(NBCH.LT.1)GOTO 48
C   DATEN FUER NEBENDING. AN ZWISCHENPUNKTEN EINLESEN, NP=NUMMER
      DO 45 I=1,NBCH
      READ(5,*)NP,IBC(NP),RBC(NP)
      WRITE(6,*)NP,IBC(NP),RBC(NP)
   45 CONTINUE
   48 CONTINUE
      NI1=NI-1
      DXI=(X(N)-X(1)-0.01)/NI1
      XI(1)=X(1)
      DO 50 I=1,NI1
      XI(I+1)=XI(I)+DXI
   50 CONTINUE
C   SEHNELAENGEN UND WINKEL GAMA BERECHNEN
      CALL INANLE(X,Y,N,RL,GAMA)
C   WINKEL ALFA BESTIMMEN
      CALL INTANG(N,IBC,RBC,RL,GAMA,ALFA,AA)
C   ORDINATEN BERECHNEN
      CALL INTPL2(X,Y,ALFA,GAMA,N,XI,YI,NI,1)
      WRITE(6,200)(XI(I),I=1,NI)
      WRITE(6,200)(YI(I),I=1,NI)
C   STEIGUNGEN BERECHNEN
      CALL INTPL2(X,Y,ALFA,GAMA,N,XI,YI,NI,2)
      WRITE(6,200)(XI(I),I=1,NI)
      WRITE(6,200)(YI(I),I=1,NI)
C   KRUEMMUNGEN BERECHNEN
      CALL INTPL2(X,Y,ALFA,GAMA,N,XI,YI,NI,3)
      WRITE(6,200)(XI(I),I=1,NI)
      WRITE(6,200)(YI(I),I=1,NI)
      READ(5,*)ZM,AC
      WRITE(6,200)ZM,AC
C   PLOT INITIIEREN
      CALL PLOTS(0,0,12)
C   URSPRUNG DEFINIEREN
      CALL PLOT(3.,2.,-3)
C   URSPRUNG MIT SYMBOL + MARKIEREN
      CALL SYMBOL(0.,0.,0.7,3,0.,-1)
C   KURVE ZEICHNEN
      CALL IDRAW2(X,Y,ALFA,GAMA,N,ZM,AC)
C   PLOT ABSCHLIESSEN
      CALL PLOT(20.,30.,999)
  200 FORMAT(1X,6G11.4)
      STOP
      END
```

```
          4           21           2
0.0000E+00  1.250      1.500       1.500
0.0000E+00 0.0000E+00 0.2500       2.000                     Eingabedaten
          3            3  0.0000000E+00    90.00000
          2            1  0.0000000E+00
          3            2  0.0000000E+00
0.0000E+00 0.7450E-01 0.1490      0.2235      0.2980      0.3725       Ab-
0.4470     0.5215     0.5960      0.6705      0.7450      0.8195       szissen
0.8940     0.9685      1.043       1.118       1.192       1.266
 1.341      1.415      1.490
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00  Ordi-
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00  naten
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.2816E-03
0.1025E-01 0.4381E-01 0.1600
0.0000E+00 0.7450E-01 0.1490      0.2235      0.2980      0.3725       Ab-
0.4470     0.5215     0.5960      0.6705      0.7450      0.8195       szissen
0.8940     0.9685      1.043       1.118       1.192       1.266
 1.341      1.415      1.490
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00  Stei-
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00  gungen
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.3473E-01
0.2539     0.7201      4.000
0.0000E+00 0.7450E-01 0.1490      0.2235      0.2980      0.3725       Ab-
0.4470     0.5215     0.5960      0.6705      0.7450      0.8195       szissen
0.8940     0.9685      1.043       1.118       1.192       1.266
 1.341      1.415      1.490
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00  Krüm-
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00  mungen
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00  2.212
 3.590      3.439      3.567
0.2000     0.1000E-02
```

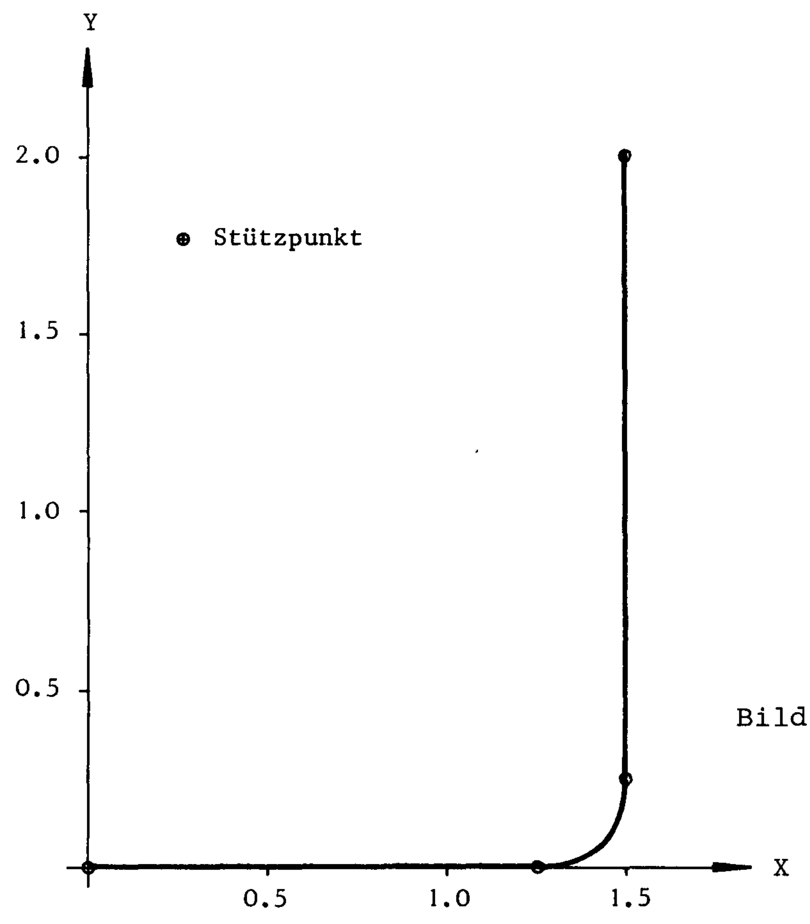


Bild 8

```
      SUBROUTINE INANLE(X,Y,N,RL,GAMA)
C  ZUR BERECHNUNG VON SEHNENLAENGEN UND WINKELN
C  ZW. SEHNEN UND X-ACHSE
C  BESCHREIBUNG DER PARAMETER:
C    EINGANG:
C      X    -FELD MIT X-WERTE
C      Y    -FELD MIT Y-WERTE
C      N    -ANZAHL DER PUNKTE
C    AUSGANG:
C      RL   -FELD MIT SEHNENLAENGEN
C      GAMA-FELD MIT WINKELN ZW. SEHNEN UND X-ACHSE
C
      DIMENSION X(N),Y(N),GAMA(N),RL(1)
      DATA PI/3.141592654/
C
      N1=N-1
      DX1=X(2)-X(1)
      DY1=Y(2)-Y(1)
      RL(1)=SQRT(DX1**2+DY1**2)
      GAMA(1)=ARSIN(DY1/RL(1))
      IF(DX1.LT.0.)GAMA(1)=SIGN(-1.,GAMA(1))*PI+GAMA(1)
      IF(N1.LE.1)GOTO 200
      DO 150 I=2,N1
      I1=I-1
      I2=I+1
      DX2=X(I2)-X(I)
      DY2=Y(I2)-Y(I)
      RL(I)=SQRT(DX2**2+DY2**2)
      CRPM=DX1*DX2+DY1*DY2
      ARG=(DY2*DX1-DX2*DY1)/RL(I1)/RL(I)
      IF(ABS(ARG).GT.1.)ARG=SIGN(1.,ARG)
      DIF=ARSIN(ARG)
      IF(CRPM.LT.0.)DIF=SIGN(1.,DIF)*(PI-ABS(DIF))
      GAMA(I)=GAMA(I1)+DIF
      DX1=DX2
      DY1=DY2
  150 CONTINUE
  200 CONTINUE
      GAMA(N)=PI/2.
      RETURN
      END
```

```
      SUBROUTINE INTANG(N,IBC,RBC,RL,GAMA,ALFA,AA)
C
C  ZUR BERECHNUNG DER TANGENTWINKEL EINER KURVE MIT
C  VORGEGEBENEN STUETZPUNKTEN UND RANDBEDINGUNGEN
C
C  BESCHREIBUNG DER PARAMETER:
C    EINGANG:
C       N    -ANZAHL DER STUETZPUNKTE
C       IBC  -FELD DER KENNZAHL DER NEBENBEDINGUNGEN
C             IBC(1),IBC(N)=1: F''(1)=RBC(1)*F''(2)
C                              F''(N)=RBC(N)*F''(N-1)
C             BEI KA(KRUEMMER AUSLAUF), RBC(1) BZW. RBC(N)=1.
C             BEI GA(GERADER AUSLAUF), RBC(1) BZW. RBC(N)=0.
C             IBC(1),IBC(N)=2: KRUEMMUNG AN ENDPKT. GEG.
C                              BETRAG MUSS KLEINER ALS 1.E6
C             IBC(1),IBC(N)=3: TANGENTWINKEL AN ENDPKT. GEG.
C             FUER I=2,N-1
C             IBC(I)=0: F'' STETIG
C             IBC(I)=1: F'' BETRAG UNSTETIG, MIT 'WL'
C             IBC(I)=2: F'' BETRAG UNSTETIG, MIT 'HL'
C             IBC(I)=3: RBC(I)=GEGEBENE TANGENTWINKEL
C       RBC  -FELD MIT DER NEBENBEDINGUNG ENTSP. GROESSEN
C       RL   -FELD MIT SEHNELAENGEN
C       GAMA-FELD MIT WINKELN ZW. SEHNEN UND X-ACHSE
C    AUSGANG:
C       ALFA-FELD MIT WINKELN ZW. TANGENTEN UND SEHNEN
C    HILFSPARAMETER:
C       AA   -HILFSFELD MIT MINDESTER DIMENSION 3*N
C
C     VERWENDETE UNTERPROGRAMME
C      SUBROUTINE GELB
C
      DIMENSION ALFA(1),IBC(1),RBC(1),GAMA(1),RL(1),AA(1)
      DATA MLD,MUD/2*1/,EPS/1.E-7/,RAD/0.0174532925/
      DATA IOUT/6/
C
      N1=N-1
      NEA=N1*3
C  FUER DEN ANFANGSPUNKT
      IB=IBC(1)
      GOTO(20,30,40),IB
   20 ALFA(1)=(2.*RBC(1)+1.)*(GAMA(1)-GAMA(2))
      AA(1)=RBC(1)+2.
      AA(2)=2.*RBC(1)+1.
      GOTO 50
   30 ALFA(1)=GAMA(1)-GAMA(2)-0.5*RL(1)*RBC(1)
      AA(1)=2.
      AA(2)=1.
      GOTO 50
   40 ALFA(1)=RBC(1)*RAD-GAMA(1)
      AA(1)=1.
      AA(2)=0.
   50 CONTINUE
C
C  FUER DEN ENDPUNKT
      IB=IBC(N)
      GOTO(60,70,80),IB
   60 ALFA(N)=(RBC(N)+2.)*(GAMA(N1)-GAMA(N))
```

```
      AA(NEA)=2.*RBC(N)+1.
      AA(NEA+1)=RBC(N)+2.
      GOTO 90
   70 ALFA(N)=2.*(GAMA(N1)-GAMA(N))+0.5*RL(N1)*RBC(N)
      AA(NEA)=1.
      AA(NEA+1)=2.
      GOTO 90
   80 ALFA(N)=RBC(N)*RAD-GAMA(N)
      AA(NEA)=0.
      AA(NEA+1)=1.
C  FUER ZWISCHENPUNKTE
   90 IF(N.EQ.2)GOTO 110
      G1=GAMA(1)-GAMA(2)
      DO 100 I=2,N1
      I1=I-1
      J=I1*3
      IB=IBC(I)+1
      G2=GAMA(I)-GAMA(I+1)
      GOTO(91,92,93,94),IB
C
C  KRUEMMUNG STETIG
   91 ALFA(I)=2.*RL(I)*G1+RL(I1)*G2
      AA(J)=RL(I)
      AA(J+1)=2.*(RL(I1)+RL(I))
      AA(J+2)=RL(I1)
      GOTO 95
C
C  KRUEMMUNG UNSTETIG, MIT 'WL'
   92 ALFA(I)=G1
      AA(J)=1.
      AA(J+1)=1.
      AA(J+2)=0.
      GOTO 95
C
C  KRUEMMUNG UNSTETIG, MIT 'HL'
   93 ALFA(I)=G2
      AA(J)=0.
      AA(J+1)=1.
      AA(J+2)=1.
      GOTO 95
C
C  TANGENTWINKEL RBC(I) VORGEGEBEN
   94 ALFA(I)=RBC(I)*RAD-GAMA(I)
      AA(J)=0.
      AA(J+1)=1.
      AA(J+2)=0.
   95 G1=G2
  100 CONTINUE
  110 CONTINUE
      CALL GELB(ALFA,AA,N,1,MUD,MLD,EPS,IER)
      IF(IER)140,120,140
  120 RETURN
  140 WRITE(IOUT,1002)IER
 1002 FORMAT(1H ,'IER=',I5,3X,25HFEHLER SIEHE BESCHREIBUNG,
     *  22H DER SUBROUTINE *GELB*)
      RETURN
      END
```

```
C      ..............................................................
C
C         SUBROUTINE GELB
C
C         PURPOSE
C            TO SOLVE A SYSTEM OF SIMULTANEOUS LINEAR EQUATIONS WITH A
C            COEFFICIENT MATRIX OF BAND STRUCTURE
C
C         USAGE
C            CALL GELB(R,A,M,N,MUD,MLD,EPS,IER)
C
C         DESCRIPTION OF PARAMETERS
C            R       -M BY N RIGHT HAND SIDE MATRIX (DESTORED).
C                     ON RETURN R CONTAINS THE SOLUTION OF THE EQUATIONS.
C            A       -M BY M COEFFICIENT MATRIX WITH BAND STRUCTURE
C                     (DESTORED)
C            M       -THE NUMBER OF EQUATIONS IN THE SYSTEM.
C            N       -THE NUMBER OF RIGHT HAND SIDE VEKTORS.
C            MUD     -THE NUMBER OF UPPER CODIAGONALS (THAT MEANS
C                     CODIAGONALS ABOVE MAIN DIAGONAL).
C            MLD     -THE NUMBER OF LOWER CODIAGONALS (THAT MEANS
C                     CODIAGONALS BELOW MAIN DIAGONAL).
C            EPS     -AN INPUT CONSTANT WHICH IS USED AS RELATIVE
C                     TOLERANCE FOR TEST ON LOSS OF SIGNIFICANCE.
C            IER     -RESULTING ERROR PARAMETER CODED AS FOLLOWS
C                     IER=0  -NO ERROR.
C                     IER=-1 -NO RESULT BECAUSE OF WRONG INPUT PARAME-
C                             TERS M,MUD,MLD OR BECAUSE OF PIVOT ELEMENT
C                             AT ANY ELIMINATION STEP EQUAL TO 0.
C                     IER=K  -WARNING DUE TO POSSIBLE LOSS OF SIGNIFI-
C                             CANCE INDICATED AT ELIMINATION STEP K+1,
C                             WHERE PIVOT ELEMENT WAS LESS THAN OR
C                             EQUAL TO THE INTERNAL TOLERANCE EPS TIMES
C                             ABSOLUTELY GREATEST ELEMENT OF MATRIX A.
C      ..............................................................
      SUBROUTINE GELB(R,A,M,N,MUD,MLD,EPS,IER)
C
C
      DIMENSION R(1),A(1)
C
C      TEST ON WRONG INPUT PARAMETERS
      IF(MLD)47,1,1
    1 IF(MUD)47,2,2
    2 MC=1+MLD+MUD
      IF(MC+1-M-M)3,3,47
C
C      PREPARE INTEGER PARAMETERS
C         MC=NUMBER OF COLUMNS IN MATRIX A
C         MU=NUMBER OF ZEROS TO BE INSERTED IN FIRST ROW OF MATRIX A
C         ML=NUMBER OF MISSING ELEMENTS IN LAST ROW OF MATRIX A
C         MR=INDEX OF LAST ROW IN MATRIX A WITH MC ELEMENTS
C         MZ=TOTAL NUMBERS OF ZEROS TO BE INSERTED IN MATRIX A
C         MA=TOTAL NUMBERS OF STORAGE LOCATIONS NECESSARY FOR MATRIX A
C         NM=NUMBER OF ELEMENTS IN MATRIX R
    3 IF(MC-M)5,5,4
    4 MC=M
    5 MU=MC-MLD-1
      ML=MC-MLD-1
      MR=M-ML
      MZ=(MU*(MU+1))/2
```

```
      MA=M*MC-(ML*(ML+1))/2
      NM=N*M
C
C     MOVE ELEMENTS BACKWARD AND SEARCH FOR ABSOLUTELY GREATEST ELEMENT
C     (NOT NECESSARY IN CASE OF A MATRIX WITHOUT LOWER CODIAGONALS)
      IER=0
      PIV=0.
      IF(MLD)14,14,6
    6 JJ=MA
      J=MA-MZ
      KST=J
      DO 9 K=1,KST
      TB=A(J)
      A(JJ)=TB
      TB=ABS(TB)
      IF(TB-PIV)8,8,7
    7 PIV=TB
    8 J=J-1
    9 JJ=JJ-1
C
C     INSERT ZEROS IN FIRST MU ROWS (NOT NESSARY IN CASE MZ=0)
      IF(MZ)14,14,10
   10 JJ=1
      J=1+MZ
      IC=1+MUD
      DO 13 I=1,MU
      DO 12 K=1,MC
      A(JJ)=0.
      IF(K-IC)11,11,12
   11 A(JJ)=A(J)
      J=J+1
   12 JJ=JJ+1
   13 IC=IC+1
C
C     GENERATE TEST VALUE FOR SINGULARITY
   14 TOL=EPS*PIV
C
C
C     START DECOMPOSITION LOOP
      KST=1
      IDST=MC
      IC=MC-1
      DO 38 K=1,M
      IF(K-MR-1)16,16,15
   15 IDST=IDST-1
   16 ID=IDST
      ILR=K+MLD
      IF(ILR-M)18,18,17
   17 ILR=M
   18 II=KST
C
C     PIVOT SEARCH IN FIRST COLUMN (ROW INDEXES FROM I=K UP TO I=ILR)
      PIV=0.
      DO 22 I=K,ILR
      TB=ABS(A(II))
      IF(TB-PIV)20,20,19
   19 PIV=TB
      J=I
      JJ=II
   20 IF(I-MR)22,22,21
```

```
   21 ID=ID-1
   22 II=II+ID
C
C     TEST ON SINGULARITY
      IF(PIV)47,47,23
   23 IF(IER)26,24,26
   24 IF(PIV-TOL)25,25,26
   25 IER=K-1
   26 PIV=1./A(JJ)
C
C     PIVOT ROW REDUCTION AND ROW INTERCHANGE IN RIGHT HAND SIDE R
      ID=J-K
      DO 27 I=K,NM,M
      II=I+ID
      TB=PIV*R(II)
      R(II)=R(I)
   27 R(I)=TB
C
C     PIVOT ROW REDUCTION AND ROW INTERCHANGE IN COEFFICIENT MATRIX A
      II=KST
      J=JJ+IC
      DO 28 I=JJ,J
      TB=PIV*A(I)
      A(I)=A(II)
      A(II)=TB
   28 II=II+1
C
C     ELEMENT REDUCTION
      IF(K-ILR)29,34,34
   29 ID=KST
      II=K+1
      MU=KST+1
      MZ=KST+IC
      DO 33 I=II,ILR
C
C     IN MATRIX A
      ID=ID+MC
      JJ=I-MR-1
      IF(JJ)31,31,30
   30 ID=ID-JJ
   31 PIV=-A(ID)
      J=ID+1
      DO 32 JJ=MU,MZ
      A(J-1)=A(J)+PIV*A(JJ)
   32 J=J+1
      A(J-1)=0.
C
C     IN MATRIX R
      J=K
      DO 33 JJ=I,NM,M
      R(JJ)=R(JJ)+PIV*R(J)
   33 J=J+M
   34 KST=KST+MC
      IF(ILR-MR)36,35,35
   35 IC=IC-1
   36 ID=K-MR
      IF(ID)38,38,37
   37 KST=KST-ID
   38 CONTINUE
C     END OF DECOMPOSITION LOOP
```

```
C
C
C       BACK SUBSTITUTION
        IF(MC-1)46,46,39
   39   IC=2
        KST=MA+ML-MC+2
        II=M
        DO 45 I=2,M
        KST=KST-MC
        II=II-1
        J=II-MR
        IF(J)41,41,40
   40   KST=KST+J
   41   DO 43 J=II,NM,M
        TB=R(J)
        MZ=KST+IC-2
        ID=J
        DO 42 JJ=KST,MZ
        ID=ID+1
   42   TB=TB-A(JJ)*R(ID)
   43   R(J)=TB
        IF(IC-MC)44,45,45
   44   IC=IC+1
   45   CONTINUE
   46   RETURN
C
C
C       ERROR RETURN
   47   IER=-1
        RETURN
        END
```

```
      SUBROUTINE INTPL2(X,Y,ALFA,GAMA,N,XI,YI,NI,MOD)
C  ZUR INTERPOLATION MIT ERMITTELTEN WINKELN ALFA UND GAMA
C  BESCHREIBUNG DER PARAMETER:
C    EINGANG:
C     X    -FELD MIT DIM. N ENTHAELT X-WERTE (ABSZISSE)
C     Y    -FELD MIT DIM. N ENTHAELT Y-WERTE (ORDINATE)
C     ALFA-FELD MIT WINKELN ZW. TANGENTEN UND SEHNEN, DIM. N
C     GAMA-FELD MIT WINKELN ZW. SEHNEN UND X-ACHSE, DIM. N
C     N    -ANZAHL DER GEGEB. PUNKTE
C     XI   -FELD MIT DIM. NI ENTHAELT ZU INTERPOL. X-WERTE (ABSZISSE)
C     NI   -ANZAHL DER ZU INTERPOL. PUNKTE
C     MOD -KENNZAHL DER ZU ERMITTELNDEN DATEN
C          1, ORDINATEN
C          2, STEIGUNGEN
C          3, KRUEMMUNGEN
C    AUSGANG:
C     YI   -FELD MIT DIM. NI ENTHAELT NACH MOD BESTIMMTEN DATEN
C
      DIMENSION X(N),Y(N),ALFA(N),GAMA(N),XI(NI),YI(NI)
C
      N1=N-1
      DO 300 I=1,NI
      DO 100 J=1,N1
      IF(XI(I).GE.X(J).AND.XI(I).LT.X(J+1))GOTO 200
  100 CONTINUE
      J=J-1
  200 BETA=ALFA(J+1)+GAMA(J+1)-GAMA(J)
      AI=TAN(ALFA(J))
      BI=TAN(BETA)
      T=PT(X(J),Y(J),AI,BI,XI(I))
      DX=X(J+1)-X(J)
      DY=Y(J+1)-Y(J)
      ET=1.-T
      GOTO(220,240,240),MOD
  220 YI(I)=Y(J)+DY*T+DX*T*ET*(AI*ET-BI*T)
      GOTO 300
  240 DXT=DX-DY*((ET-T)*(AI*ET-BI*T)-T*ET*(AI+BI))
      DYT=DY+DX*((ET-T)*(AI*ET-BI*T)-T*ET*(AI+BI))
      IF(MOD.EQ.3)GOTO 260
      YI(I)=SIGN(1.,DXT)*DYT/(ABS(DXT)+1.E-30)
      GOTO 300
  260 DXTT=2.*DY*(AI*(2.-3.*T)+BI*(1.-3.*T))
      DYTT=-2.*DX*(AI*(2.-3.*T)+BI*(1.-3.*T))
      YI(I)=(DXT*DYTT-DYT*DXTT)/((DXT**2+DYT**2)**1.5+1.E-20)
  300 CONTINUE
      RETURN
      END
```

```
      FUNCTION PT(X,Y,AI,BI,XI)
C  ZUM ERMITTELN DER PARAMETER T BEI GEGEB. X-WERT XI
C  BESCHREIBUNG DER PARAMETER:
C    EINGANG:
C      X    -X-ORDINATEN DER ZWEI ENDPUNKTE
C      Y    -Y-ORDINATEN DER ZWEI ENDPUNKTE
C      AI    TANGENS DES WINKELS ZW. TANGENTE UND SEHNE AM L. ENDE
C      BI    TANGENS DES WINKELS ZW. TANGENTE UND SEHNE AM R. ENDE
C      XI   -GEGEB. X-WERT, ZU DEM PAR. T ERMITTELT WIRD
C    AUSGANG:
C      PT   -ZU ERMITTELNDER PARAMETER
C
      DIMENSION X(2),Y(2)
      DATA IOUT/6/,MIT/30/,EPS/1.E-6/
C
      IF(ABS(XI-X(1)).GT.1.E-6)GOTO 100
      PT=0.
      RETURN
  100 IF(ABS(XI-X(2)).GT.1.E-6)GOTO 200
      PT=1.
      RETURN
C  KOEFF. DES POLYNOM 3. GRADES
  200 A=-(Y(2)-Y(1))*(AI+BI)
      B=(Y(2)-Y(1))*(AI-BI)*C.5
      C=-C.25*A+(X(2)-X(1))
      D=-C.25*B+(X(2)-X(1))*0.5+X(1)-XI
      S=0.
      IT=C
  300 IT=IT+1
      IF(IT.GT.MIT)GOTO 400
      F=(((A*S+B)*S+C)*S+D)/((3.*A*S+2.*B)*S+C)
      S=S-F
      IF(ABS(F).GE.EPS)GOTO 300
      IF(S.LE.-0.5-EPS.OR.S.GE.0.5+EPS)GOTO 500
      PT=S+0.5
      RETURN
  400 WRITE(IOUT,410)
  410 FORMAT(37H ***NO CONVERG., MAX. ITERATIONS USED)
      RETURN
  500 WRITE(IOUT,510)
  510 FORMAT(23H ***NO SOLUTION FOUNDED)
      RETURN
      END
```

```
      FUNCTION RKRMM(DX,DY,AI,BI,T)
C  ZUR BERECHNUNG DER KRUEMMUNG EINES KURVENSTUECKS
C  BEI GEGEBENEM PARAMETER T
      ET=1.-T
      DXT=DX-DY*((ET-T)*(AI*ET-BI*T)-T*ET*(AI+BI))
      DYT=DY+DX*((ET-T)*(AI*ET-BI*T)-T*ET*(AI+BI))
      DXTT=2.*DY*(AI*(2.-3.*T)+BI*(1.-3.*T))
      DYTT=-2.*DX*(AI*(2.-3.*T)+BI*(1.-3.*T))
      RKRMM=ABS(DXT*DYTT-DYT*DXTT)/((DXT**2+DYT**2)**1.5+1.E-20)
      RETURN
      END
```

```
      SUBROUTINE IDRAW2(X,Y,ALFA,GAMA,N,ZM,AC)
C  ZUM ZEICHNEN EINER KURVE, DIE DURCH KUBISCHE SPLINE
C  IN PARAMETERFORM DARGESTELLT IST.
C  BESCHREIBUNG DER PARAMETER:
C    X BIS N-SIEHE SUBROUTINE INTPL2
C    ZM      -ZEICHENMASSSTAB
C    AC      -GENAUIGKEIT ZUM ZEICHNEN EINER KURVE DURCH GERADEN
C
      DIMENSION X(N),Y(N),ALFA(N),GAMA(N)
      DIMENSION XZ(202),YZ(202),DQX(4),DQY(4)
      DATA DTT/0.1/,NTT/11/
C
      N1=N-1
      DO 300 I=1,N1
      DX=X(I+1)-X(I)
      DY=Y(I+1)-Y(I)
      RL=SQRT(DX**2+DY**2)
      BETA=ALFA(I+1)+GAMA(I+1)-GAMA(I)
      AI=TAN(ALFA(I))
      BI=TAN(BETA)
C  KRUEMMM. AN NTT STELLEN BERECHNEN, DIE GROESTE AUSSUCHEN
      RKM=0.
      T=0.
      DO 100 J=1,NTT
      RKM=AMAX1(RKM,RKRMM(DX,DY,AI,BI,T))
  100 T=T+DTT
C  DIE MASSSTAB UND GENAUIGKEIT ENTSP. PUNKTZAHL BESTIMMEN
      NP=IFIX(AMAX1(1.,RL/SQRT(8.*ZM*AC/AMAX1(RKM,1.E-8)))+2.)
      DT=1./(NP-1)
C  PUNKTKOORDINATEN ZUM ZEICHNEN BERECHNEN MIT DIFFERERZENVERF.
      XZ(1)=X(I)
      YZ(1)=Y(I)
      XZ(NP)=X(I+1)
      YZ(NP)=Y(I+1)
      IF(NP.LE.2)GOTO 250
      C1=(AI+BI)*DT**3
      C2=(2.*AI+BI)*DT**2
      DQX(1)=X(I)
      DQX(2)=(DX-AI*DY)*DT+DY*(C2-C1)
      DQX(3)=2.*DY*(C2-3.*C1)
      DQX(4)=-6.*DY*C1
      DQY(1)=Y(I)
      DQY(2)=(DY+AI*DX)*DT-DX*(C2-C1)
      DQY(3)=-2.*DX*(C2-3.*C1)
      DQY(4)=6.*DX*C1
      NP1=NP-1
      DO 200 J=2,NP1
      DO 150 K=1,3
      DQX(K)=DQX(K)+DQX(K+1)
  150 DQY(K)=DQY(K)+DQY(K+1)
      XZ(J)=DQX(1)
  200 YZ(J)=DQY(1)
  250 CONTINUE
      XZ(NP+1)=0.
      YZ(NP+1)=0.
      XZ(NP+2)=ZM
      YZ(NP+2)=ZM
C  ZEICHNEN EIN KURVENSTUECK
  300 CALL LINE(XZ,YZ,NP,1,0,0)
      RETURN
      END
```

4. Rationale kubische Splines in Parameterform für ebene Kurven und Raumkurven

Im folgenden wird nur auf die Raumkurven eingegangen; für ebene Kurven ist eine der drei Koordinaten X,Y,Z wegzulassen.

Zwischen zwei Stützpunkten (X_i, Y_i, Z_i) und $(X_{i+1}, Y_{i+1}, Z_{i+1})$ wird ein Kurvenstück definiert durch gebrochene kubische Polynome:

$$X(u) = \frac{a_{xi} + b_{xi}\,u + c_{xi}\,u^2 + d_{xi}\,u^3}{a_{hi} + b_{hi}\,u + c_{hi}\,u^2 + d_{hi}\,u^3}$$

$$Y(u) = \frac{a_{yi} + b_{yi}\,u + c_{yi}\,u^2 ++ d_{yi}\,u^3}{a_{hi} + b_{hi}\,u + c_{hi}\,u^2 + d_{hi}\,u^3}$$

$$Z(u) = \frac{a_{zi} + b_{zi}\,u + c_{zi}\,u^2 + d_{zi}\,u^3}{a_{hi} + b_{hi}\,u + c_{hi}\,u^2 + d_{hi}\,u^3} \tag{6}$$

mit $0 \leqslant u \leqslant 1$. Während die Zähler drei verschiedene Polynome sind, ist der Nenner in allen drei Ausdrücken dasselbe kubische Polynom. Bezeichnet man die Zähler mit kleinen Buchstaben x(u), y(u) und z(u) und den Nenner mit h(u), so hat man dieselbe Darstellung wie in den Gleichungen (3), nur jetzt im 4-dimensionalen Raum mit den Koordinaten x,y,z und h. Man nennt diese homogene Koordinaten. Die kartesischen Koordinaten X,Y,und Z hängen mit den homogenen Koordinaten also durch die Gleichungen

$$X(u) = \frac{x(u)}{h(u)}$$

$$Y(u) = \frac{y(u)}{h(u)}$$

$$Z(u) = \frac{z(u)}{h(u)} \tag{7}$$

zusammen.

Im folgenden werden die kartesischen Koordinaten eines Kurvenpunktes zum Parameter u in Vektorschreibweise mit $\vec{P}(u)$ bezeichnet; die Ableitung nach u ist $\vec{P}_u(u)$; und die homogenen Koordinaten und ihre u-Ableitungen werden mit $\vec{p}(u)$ bzw. $\vec{p}_u(u)$ bezeichnet. So kann der Vektor $\vec{p}(u)$ in Matrizenschreibweise nach folgender Formel berechnet werden:

$$\vec{p}(u) = \begin{pmatrix} x(u) \\ y(u) \\ z(u) \\ h(u) \end{pmatrix} = \begin{pmatrix} a_{xi} & b_{xi} & c_{xi} & d_{xi} \\ a_{yi} & b_{yi} & c_{yi} & d_{yi} \\ a_{zi} & b_{zi} & c_{zi} & d_{zi} \\ a_{hi} & b_{hi} & c_{hi} & d_{hi} \end{pmatrix} \begin{pmatrix} 1 \\ u \\ u^2 \\ u^3 \end{pmatrix} \qquad (8)$$

Um zu sehen, wie die Koeffizienten dieser Matrix mit den Kennwerten des betreffenden Kurvenstücks zusammenhängen, ist die folgende Umformung nützlich:

$$\vec{p}(u) = [\vec{p}(0) \quad \vec{p}(1) \quad \vec{p}_u(0) \quad \vec{p}_u(1)] \begin{pmatrix} F(u) \\ F(1-u) \\ G(u) \\ -G(1-u) \end{pmatrix} \qquad (9)$$

Dabei sind F(u) und G(u) die Hermiteschen Polynome 3. Grades,

$$F(u) = 2u^3 - 3u^2 + 1$$

$$G(u) = u^3 - 2u^2 + u \quad , \qquad (10)$$

und $\vec{p}(0)$, $\vec{p}(1)$ sind die homogenen Vektoren der beiden Endpunkte des Kurvenstücks, die den Parameter u = 0 bzw. u = 1 entsprechen. $\vec{p}(0)$ und $\vec{p}(1)$ werden aus den kartesischen Koordinaten der Endpunkte mit der Festsetzung bestimmt, daß h(0) und h(1) gleich 1 sind:

$$\vec{p}(0) = \begin{pmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{pmatrix} \qquad \vec{p}(1) = \begin{pmatrix} X_{i+1} \\ Y_{i+1} \\ Z_{i+1} \\ 1 \end{pmatrix} \qquad (11)$$

$\vec{p}_u(0)$ und $\vec{p}_u(1)$ werden aus den Tangentenvektoren $\vec{t}(0)$ und $\vec{t}(1)$ an den Endpunkten des betreffenden Kurvenstücks und aus vier zusätzlichen einzugebenden Kurvenparameter A(0), C(0), A(1), und C(1) berechnet (Bild 9). Bei Raumkurven können die Winkel α_i und β_i in verschiedenen Ebenen liegen.

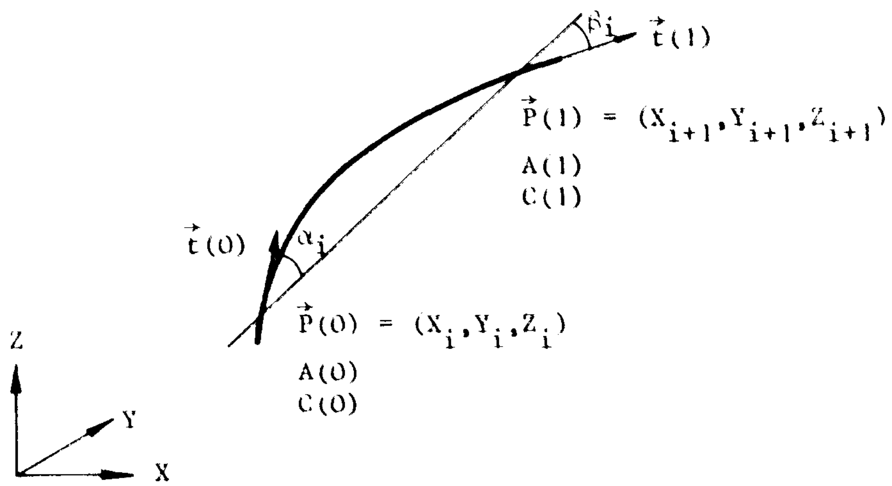


Bild 9

Diese Parameter können benutzt werden, um den Verlauf der Kurve zwischen den gegebenen Stützpunkten bei unveränderten Tangentenrichtungen in den Stützpunkten zu beeinflussen. Will man dies nicht, wird empfohlen, die "Standardwerte" 1 für alle 4 Parameter zu wählen. Die Parameter sind definiert durch die Gleichungen

$$\vec{P}_u(0) = A(0)\ \ell_i\ \vec{t}(0)$$

$$\vec{P}_u(1) = A(1)\ \ell_i\ \vec{t}(1)$$

$$h_u(0) = C(0)\left[\,|A(0)\ \vec{t}(0) + A(1)\ \vec{t}(1)| - 2\right]$$

$$h_u(1) = -C(1)\left[\,|A(0)\ \vec{t}(0) + A(1)\ \vec{t}(1)| - 2\right] \quad \text{mit}$$

$$\ell_i = |\vec{P}(1) - \vec{P}(0)| = \sqrt{(X_{i+1} - X_i)^2 - (Y_{i+1} - Y_i)^2 - (Z_{i+1} - Z_i)^2}$$

(12)

Wird die Gleichung

$$\begin{pmatrix} x(u) \\ y(u) \\ z(u) \end{pmatrix} = h(u)\vec{P}(u) \tag{13}$$

nach u abgeleitet, ergibt sich

$$\begin{pmatrix} x_u(u) \\ y_u(u) \\ z_u(u) \end{pmatrix} = h(u)\vec{P}_u(u) + h_u(u)\vec{P}(u)\,. \tag{14}$$

An den Endpunkten bei u=0 und u=1 gilt dann mit (12):

$$\begin{pmatrix} x_u(0) \\ y_u(0) \\ z_u(0) \end{pmatrix} = A(0)\ \ell_i\ \vec{t}(0) + C(0)\left[\,|A(0)\ \vec{t}(0) + A(1)\ \vec{t}(1)| - 2\right]\vec{P}(0)$$

$$\begin{pmatrix} x_u(1) \\ y_u(1) \\ z_u(1) \end{pmatrix} = A(1)\ \ell_i\ \vec{t}(1) - C(1)\left[\,|A(0)\ \vec{t}(0) + A(1)\ \vec{t}(1)| - 2\right]\vec{P}(1)$$

(15)

Damit lassen sich alle Komponenten der Vektoren $\vec{p}_u(0)$ und $\vec{p}_u(1)$ bestimmen, wenn an den beiden Endpunkten die Koordinaten $\vec{P}(0)$ und $\vec{P}(1)$, die Tangenteneinheitsvektoren $\vec{t}(0)$ und $\vec{t}(1)$ sowie die Kurvenparameter A(0), C(0), A(1) und C(1) gegeben sind. Die zugehörigen Krümmungsvektoren an den Endpunkten $\vec{K}(0)$ und $\vec{K}(1)$ sind

$$\vec{K}(0) = \frac{2}{A(0)^2 \ell_i} \Big\{ \big[3 - h_u(1)\big]\big[\vec{s}_i - (\vec{s}_i \cdot \vec{t}(0))\vec{t}(0)\big] + A(1)\big[(\vec{t}(0) \cdot \vec{t}(1))\vec{t}(0) - \vec{t}(1)\big]\Big\}$$

$$\vec{K}(1) = \frac{2}{A(1)^2 \ell_i} \Big\{ \big[3 + h_u(0)\big]\big[-\vec{s}_i + (\vec{s}_i \cdot \vec{t}(1))\vec{t}(1)\big] + A(0)\big[-(\vec{t}(0) \cdot \vec{t}(1))\vec{t}(1) + \vec{t}(0)\big]\Big\} \tag{16}$$

mit $\vec{s}_i = (\vec{P}(1) - \vec{P}(0))/\ell_i$. $\vec{s}_i$ ist also ein Einheitsvektor. In (16) besteht ein linearer Zusammenhang zwischen $\vec{K}(0)$ und $h_u(1)$ bzw. $\vec{K}(1)$ und $h_u(0)$ und damit auch zwischen $\vec{K}(0)$ und C(1) bzw. $\vec{K}(1)$ und C(0) (siehe (12)). Bei der Anforderung eines gegebenen Krümmungsvektors an einem Endpunkt ist es daher einfacher, C(0) bzw. C(1) als A(0) bzw. A(1) zu ändern. Eine besondere Eigenschaft des rationalen kubischen Splines bei ebener Kurve ist, daß Kegelschnitte exakt dargestellt werden können. Sie sind bei einem Kurvenstück durch folgende Bedingungen zu erreichen:

a) $C(0) = C(1) = 1$

b) $\mathrm{sign}(\alpha_i) = -\mathrm{sign}(\beta_i) \qquad |\alpha| \leq 90^\circ,\ |\beta_i| \leq 90^\circ$

c) $A(0)/A(1) = -\sin\beta_i/\sin\alpha_i$ (17)

wobei α_i der Winkel zwischen $\vec{s}_i$ und $\vec{t}(0)$ sowie β_i zwischen $\vec{s}_i$ und $\vec{t}(1)$ ist (Bild 9). α_i und β_i haben verschiedene Vorzeichen, wenn die Tangenten nach verschiedenen Seiten von $\vec{s}_i$ abweichen. Zum Erzeugen eines Kegelschnitts wird A(0) bzw. A(1) beim Festhalten von C(0) und C(1) auf 1 nach dem Verhältnis (17) c) geändert.

Bei Kurven, die aus mehreren Kurvenstücken bestehen, werden die Tangenteneinheitsvektoren an den Stützpunkten, wenn sie nicht gegeben sind, so bestimmt, daß der Krümmungsvektor in den Stützpunkten stetig ist (siehe hierzu [4]). Durch die gebrochenen Funktionen erfordert dieses Verfahren größeren Rechenaufwand als die 2. Methode. Dagegen hat die Software des Pakets 3 folgende Vorteile:

- Das Verfahren kann Kreise, Ellipsen und Hyperbeln exakt darstellen.
- Die Kurvenkrümmung verläuft auch bei starken Winkeländerungen zwischen zwei Stützpunkten gleichmäßig.
- Auch bei gegebenen Tangentenrichtungen in den Stützpunkten kann die Kurvenkrümmung stetig verlaufen.
- Außer Stützpunkten und ggf. Kurventangentenvektoren können weitere Kurvenparameter zur Beeinflussung des Kurvenverlaufs angegeben werden.
- Projektive Transformation der Raumkurven (perspektivische Darstellungen) ergeben Splines desselben Typs und können daher exakt ausgeführt werden.

Es folgt die Beschreibung der zu dieser Methode gehörigen Routinen:

1) Einlesen und Aufbereitung der Daten
- Die Subroutine ISPLND liest alle nötigen Daten ein, die eine Kurve definieren. Dabei wird die Subroutine ISTDP aufgerufen, um Kennzahlen und Informationswerte zu den End- bzw. Nebenbedingungen sowie die Kurvenparameter mit Standardwerten (siehe folgende Tabellen) voreinzustellen. Davon abweichende Werte können durch noch einzulesende Daten geändert werden. Die Möglichkeiten zur Angabe der End- bzw. Nebenbedingungen sind:

a) an den Endpunkten

Kennzahl	Bedeutung	zugehörige Informationswerte	
		ebene Kurve	Raumkurve
1 Vorein- stellung	Krümmung am Endpunkt = ein Faktor a mal Krümmung am Nachbarpunkt Bild 4	der Faktor a (a=1, Voreinstellung)	der Faktor a (a=1, Voreinstellung)
2	Die Ableitung der Krümmung k am Endpunkt nach dem Parameter u $\frac{dk}{du} = 0$	entfällt	entfällt
3	Tangentenrichtung vorgegeben	ein Tangentenwinkel (in Grad) zw. Abszisse und Tangentenvektor Bild 5	zwei Tangentenwinkel (in Grad) ω_1 in der XY-Ebene und ω_2 in der XZ-Ebene Bild 10
4	wie bei Kennzahl 3	wie bei Kennzahl 3	zwei Tangentenwinkel (in Grad) ω_1 in der XY-Ebene und ω_3 in der YZ-Ebene Bild 10
5	wie bei Kennzahl 3	wie bei Kennzahl 3	zwei Tangentenwinkel (in Grad) ω_2 in der XZ-Ebene und ω_3 in der YZ-Ebene Bild 10
6	Tangentenvektor vorgegeben	zwei Komponenten des Tangentenvektors	drei Komponenten des Tangentenvektors
23-26 nur für 1. Punkt	entsprechend 3 - 6 zusätzlich: Kegelschnitt zw. 1. und 2. Punkt *	wie 3 - 6	wie 3 - 6
13-16 nur für letzten Punkt	entsprechend 3 - 6 zusätzlich: Kegelschnitt zw. letztem und vor- letztem Punkt **	wie 3 - 6	wie 3 - 6

* der Kurvenparameter A(0) des ersten Kurvenstücks wird geändert
** der Kurvenparameter A(1) des letzten Kurvenstücks wird geändert

Tabelle 3

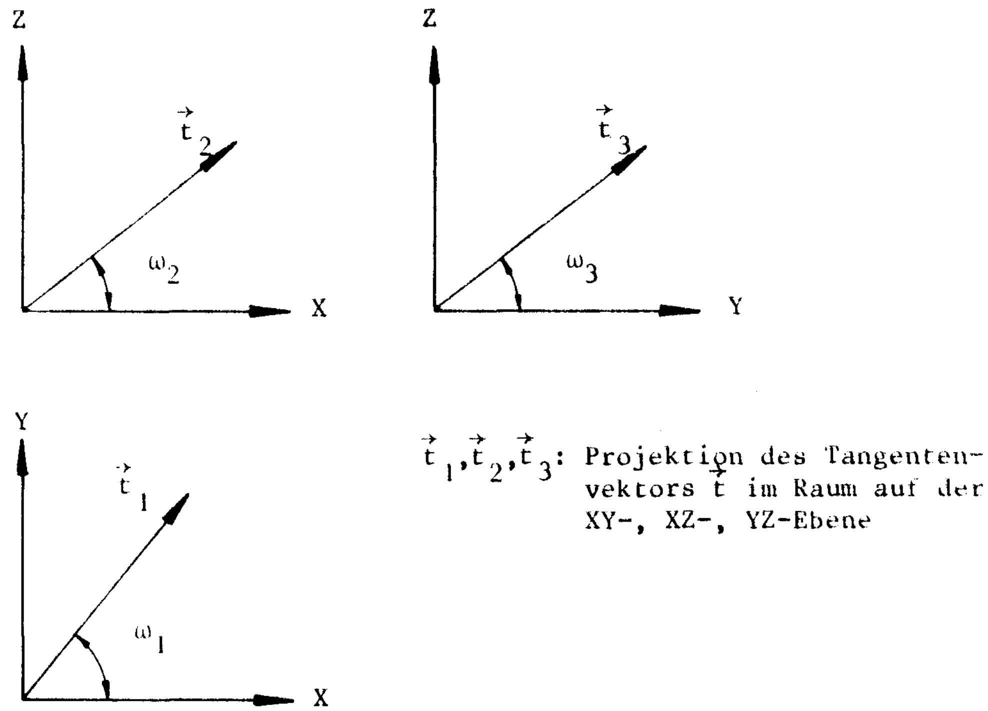
b) an den Stützpunkten zwischen den Endpunkten

Kennzahl	Bedeutung	zugehörige Informationswerte	
		ebene Kurve	Raumkurve
0 Vorein-stellung	Krümmung stetig	entfällt	entfällt
1	Krümmung unstetig Der Kurvenbereich nach dem Stützpunkt beeinflußt den Kurvenverlauf vor dem Stützpunkt	entfällt	entfällt
2	Krümmung unstetig Der Kurvenbereich vor dem Stützpunkt beeinflußt den Kurvenverlauf nach dem Stützpunkt	entfällt	entfällt
3 - 6	wie an Endpunkten	wie an Endpunkten	wie an Endpunkten
-3 bis -6	entsprechend 3 - 6 zusätzlich mit stetiger Krümmung	wie an Endpunkten	wie an Endpunkten
11 bis 16	entsprechend 1 - 6 zusätzlich mit Kegelschnitt zw. den Stützpunkten $\vec{P}_{i-1}$ und $\vec{P}_i$ *	entsprechend 1 - 6	entsprechend 1 - 6
21 bis 26	entsprechend 1 - 6 zusätzlich mit Kegelschnitt zw. den Stützpunkten $\vec{P}_i$ und $\vec{P}_{i+1}$ **	entsprechend 1 - 6	entsprechend 1 - 6

* der Kurvenparameter A(1) des Kurvenstücks zw. $\vec{P}_{i-1}$ und $\vec{P}_i$ wird geändert

** der Kurvenparameter A(0) des Kurvenstücks zw. $\vec{P}_i$ und $\vec{P}_{i+1}$ wird geändert

Tabelle 4



$\vec{t}_1, \vec{t}_2, \vec{t}_3$: Projektion des Tangentenvektors $\vec{t}$ im Raum auf der XY-, XZ-, YZ-Ebene

Bild 10

Die Vorzeichen der Winkel ω_1, ω_2 und ω_3 sind nach der in Paket 2 angegebenen Regel festzulegen.

Die Kurvenparameter A(O) und A(1) sowie C(O) und C(1) für alle Kurvenstücke haben 1 als Standardwert. Zum Beeinflussen des Kurvenverlaufs werden sie ggf. anders gewählt. Die Änderung erfolgt entweder direkt durch Wertzuweisung oder indirekt durch gegebene End- bzw. Nebenbedinungen. Bei der direkten Änderung steht für C(O) und C(1) nur der Wert O zur Auswahl. Für A(O) und A(1) können Werte s im Bereich $0.5 \leq s \leq 1.5$ gewählt werden. Die Kurve mit C(O) = C(1) = O für alle Kurvenstücke ist eine gewöhnliche kubische Splinekurve, nicht eine gebrochene kubische Splinekurve. Die Unter- und Obergrenzen für s sind nach Erfahrung angesetzt, damit keine scharfe Änderung des Kurverlaufs auftritt. Die End- bzw. Nebenbedingungen, die zur vom Programm durchgeführten indirekten Änderung der Kurvenparameter führen, sind:

a) Wenn stetige Krümmung an einem Zwischenpunkt bei einer gegebenen Tangentenrichtung gefordert wird, werden die C-Parameter zweier benachbarter Kurvenstücke so bestimmt, daß die Bedingung erfüllt wird.

b) Wenn ein Kurvenstück als Kegelschnitt verlangt wird, werden die A-Parameter des betreffenden Kurvenstücks entsprechend (17 c) geändert. Da ein Kegelschnitt vorausgetzt, daß bei dem Kurvenstück C(O) = C(1) = 1 sein muß, ergibt sich bei Änderung von C(O) und C(1) zu O nicht exakt ein Kegelschnitt.

Die einzulesenden Daten sind:

- Eine Karte für die Variabelen ND,N,IBC1,IBCN,NB,NCHAB,ICD, die folgende Bedeutung haben:

  ND - Anzahl der Dimensionen einer darzustellenden Kurve: bei ebener Kurve ND = 2, bei einer Raumkurve ND = 3

  N - Anzahl der Stützpunkte

  IBC1 - O, wenn sowohl die Kennzahl als auch die Informationswerte für den 1, Punkt den voreingestellten entsprechen
  1, sonst

  IBCN - O, wenn sowohl die Kennzahl als auch die Informationswerte für den N-ten Punkt den voreingestellten entsprechen
  1, sonst

  NB - Anzahl der anzugebenden Nebenbedingungen für die Zwischenpunkte, deren zugehörige Kennzahlen oder Informationswerte von der Voreinstellung abweichen

  NCHAB - Anzahl der zu ändernden Kurvenparameterpaare A(O) und A(1). Ihre voreingestellten Werte sind 1

  ICD - O, die Kurvenparameter C(O) und C(1) für alle Kurvenstücke werden zu O geändert.
  sonst, C(O) und C(1) besetzen Standardwert 1

- N Karten für die Stützpunkte. Jede Karte enthält die Koordinaten eines Stützpunktes, also bei ebener Kurve 2 Werte, bei Raumkurve 3 Werte.

- Wenn IBC1 $\neq$ O: eine Karte mit einer Integer-Zahl für die abzuändernde Kennzahl am 1. Stützpunkt IBC(1).

- Wenn IBCN $\neq$ O: eine Karte mit einer Integer-Zahl für die abzuändernde Kennzahl am N. Stützpunkt IBC(N).

- Wenn IBC1 $\neq$ O und IBC(1) $\neq$ 2 ist: eine Karte mit einer Anzahl von Werten, die nach Tabelle 3 einzugeben sind.

- Wenn IBCN $\neq$ O und IBC(N) $\neq$ 2 ist: eine Karte mit einer Anzahl von Werten, die nach Tabelle 3 einzugeben sind.

- Wenn NB $\neq$ O: NB mal folgende Karten:
  - eine Karte für INDP,IBC(INDP)

    INDP-Nummer des Zwischenpunktes, an dem die Kennzahl der Nebenbedingung der Voreinstellung nicht entspricht.

    IBC(INDP) - die anzugebende Kennzahl.

- Wenn eine Tangentenrichtung vorgeschrieben wird.
  (d.h. die letzte Ziffer von IBC(INDP) größer als oder gleich 3 ist; siehe die Tabelle 4) eine Karte mit einer Anzahl von Werten, die der Tabelle 4 entspricht.

- Wenn NCHAB $\neq$ O: NCHAB Karten für IAB,AB(I,IAB),AB(2,IAB)

  IAB - Nummer des ersten Punktes des betreffenden Kurvenstücks

  AB(1,IAB) - entspricht dem Kurvenparameter A(O)

  AB(2,IAB) - entspricht dem Kurvenparameter A(1)

Alle Daten werden mit der FORTRAN 77-Anweisung
READ(IN,*) Eingabeliste
gelesen, die zur formatfreien Übertragung von Daten dient.

- Die Subroutine ISTDP besetzt die Kennzahlen und die Informationswerte, die End- und Nebenbedingungen spezifizieren, sowie die Kurvenparameter mit den Standardwerten. Die Parameter der Subroutine
  ISTDP(IBC,RBC,AB,CD,CDSTD,ND,N)
  haben folgende Bedeutung:

  IBC - Feld mit Dim. N für die Kennzahlen

  RBC - 2D-Feld mit Dim. (ND,N) für die Informationswerte

  AB - 2D-Feld für die Kurvenparameter A(O) und A(1), das mit (2,N-1) dimensioniert sein muß.
  Index 1. für A(O), 2 für A(1)

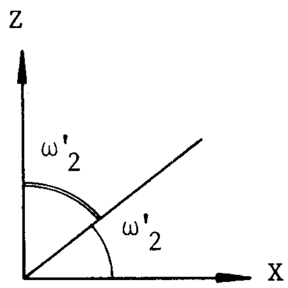
CD - entsprechend dem Feld AB für die Kurvenparameter C(O) und C(1).

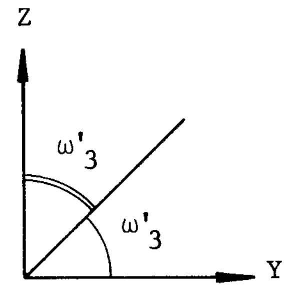
CDSTD - festzusetzender Wert für C(O) und C(1): 1. oder O.

ND - Anzahl der Dimensionen der darzustellenden Kurve

N - Anzahl der Stützpunkte

- Die Subroutine IWITV berechnet aus gegebenen Tangentenwinkeln bzw. Tangentenvektoren die Tangenteneinheitsvektoren.
- Die Subroutine IBEDUM stellt die zwei Tangentenwinkel in den zwei Projektionsebenen zur Angabe der Tangentenrichtung bei Raumkurven um, um daraus die Tangenteneinheitsvektoren zu bestimmen. Beispiel: Sind die Tangentenwinkel ω_2 in der XZ-Ebene und ω_3 in der YZ-Ebene gegeben, so





werden die Winkel ω_2' und ω_3' in Bezug auf die Z-Achse berechnet

$$\omega_2' = 90^\circ - \omega_2$$

$$\omega_3' = 90^\circ - \omega_3$$

- Die Subroutine IPWTV berechnet aus den von der Subroutine IBEDUM eingestellten Winkeln den Tangenteneinheitsvektor. In dem vorhergegenden Beispiel werden seine Komponente wie folgt berechnet

$$e_z = \frac{1}{\sqrt{1 + tg^2\omega_2' + tg^2\omega_3'}}$$

$$e_z = e_z \times tg\,\omega_2'$$

$$e_x = e_z \times tg\,\omega_3'$$

2) Bestimmung der Tangenteneinheitsvektoren und ggf. Änderung der Kurvenparameter

- Die Subroutine ISPLTP bestimmt mit Hilfe der Subroutine ISPLNT die Tangenteneinheitsvektoren. Anschließend führt sie in den oben angegebenen Fällen eine Änderung der Kurvenparameter durch. Die Parameter der Subroutine ISPLTP(X,IBC,RBC,AB,CD,ND,N,SL,SV,TANV) haben folgende Bedeutung

  Eingangsparameter

  X - 2D-Feld mit Dim. (ND,N) für die Koordinaten der Stützpunkte der Kurve

  IBC bis N - siehe die Subroutine ISTDT. Die Ausgangswerte von Feld AB bzw. CD können ggf. durch gegebene Bedingungen von ihren Eingangswerten abweichen

  Ausgangsparameter

  SL - Feld für Sehnelängen zwichen den Stützpunkten Dim. N-1

  SV - 2D-Feld für Sehnevektoren mit Einheitslängen, also $(\vec{P}_{i+1} - \vec{P}_i)/|\vec{P}_{i+1} - \vec{P}_i|$ für i = 1 bis N - 1. Dim. (ND,N)

  TANV - 2D-Feld für die zu ermittelnden Tangenteneinheitsvektoren $\vec{t}_i$, i = 1 bis N

- Die Subroutine ISPLNT berechnet iterativ die Tangenteneinheitsvektoren aus der Bedingung, daß die Krümmung der Kurve, wenn nichts anderes angegeben ist, an den Stützpunkten stetig ist. Nebenbedingungen mit unstetigen Krümmungen, gegebenen

Tangentenrichtungen oder gegebenem Krümmungsverhältnis an den Kurvenenden erfordern eine besondere Behandlung. Die iterative Auflösung des nichtlinearen Gleichungssystems für die Tangenteneinheitsvektoren erfolgt mit Hilfe der Subroutine ITANV. Die Parameter der Subroutine
ISPLNT(X,IBC,AB,CD,SL,SV,ND,N,MAXIT,EPS,TANV)
haben folgende Bedeutung

X bis N - siehe die Subroutine ISPLTP

MAXIT - maximale Anzahl der Iterationsschritte
EPS - geforderte Genauigkeit
TANV - siehe die Subroutine ISPLTP

- Die Subroutine ISLSV berechnet die Sehnelängen und Sehnevektoren (Einheitsvektoren) zwischen den Stützpunkten. Ihre Parameter ISLSV(X,ND,N,SL,SV) sind in den oben stehenden Routinen beschrieben.
- Die Subroutine ITANV(IBC,RBC,AB,CD,SL,SV,ND,N,TANV,AA) baut ein lineares Gleichungssystem für die zu ermittelnden Tangenteneinheitsvektoren auf und löst das System mit der Subroutine GELB, die im 2. Paket beschrieben ist.
- Die Subroutine IKREND(SV,SL,TV,AB,CD,RKR,ND,LR) berechnet den Krümmungsvektor an einem Endpunkt eines Kurvenstücks. Ihre Parameter haben folgende Bedeutung:

  Eingangsparameter

  SV - Sehnevektoren zwischen den Endpunkten

  SL - Sehnelängen zwischen den Punkten

  TV - Tangenteneinheitsvektoren an den Endpunkten

  AB - die Kurvenparameter A(0) und A(1) des Kurvenstücks

  CD - die Kurvenparameter C(0) und C(1) des Kurvenstücks

  ND - Anzahl der Dimensionen der Kurve

  LR - gibt an, ob der Krümmungsvektor am 1. (LR = 1) oder 2. (LR = 2) Endpunkt berechnet wird

Ausgangsparameter

RKR - der zu berechnende Krümmungsvektor. Dim. 2 oder 3.

- Die Funktion HABL(SV,SL,TV,AB,RKR,ND,LR) berechnet $h_u(0)$ bzw. $h_u(1)$ in den Formeln (12) zu gegebenem Krümmungsvektor RKR.
- Die Funktion CDK(SV,SL,TV,AB,RKR,ND,LR) berechnet den zu $h_u(0)$ und $h_u(1)$ gehörigen Kurvenparameter C(0) bzw. C(1).
- Die Subroutine IEINHV(VEKT,ND) rechnet einen Vektor zu einem parallelen Einheitsvektor um. Der Parameter VEKT enthält beim Eingang den Vektor und beim Ausgang den Einheitsvektor. ND ist Anzahl der Dimension des Vektors.

3) Interpolation

- Die Subroutine INTPL3 berechnet zu einer gegebenen Koordinate (X,Y oder Z) die zwei anderen Koordinaten. Sind mehrere Punkte einem Wert der gegebenen Koordinaten zugeordnet, wird nur der erste Punkt in der Laufrichtung der Kurve ermittelt. Die Parameter der Subroutine
  INTPL3(X,AB,CD,TANV,ND,N,XI,IORD,NI)
  haben folgende Bedeutung

  Eingangsparameter

  X bis N - siehe die Subroutine ISPLNT

  NI - Anzahl der zu interpolierenden Punkte

  IORD - Kennzahl zu den gegebenen Koordinaten
  1 : X; 2 : Y; 3 : Z.

  XI(IORD,J) - für J = 1 bis NI, enthalten die Werte der gegebenen Koordinate

  Ausgangsparameter

  XI(I,J) - für I = 1 bis ND und J = 1 bis NI, enthalten die gegebenen und die interpolierten Koordinaten

- Die Subroutine ICURVI(CM,AB,CD,XI,IORD,ND,N) interpoliert zu einer gegebenen Koordinate eines Kurvenstücks die zwei übrigen Koordinaten. Ist das Kurvenstück zu dem Wert der gegebenen

Koordinate mehrdeutig, wird nur ein Punkt berechnet. Die Parameter AB,CD sind die Kurvenparameter A(0) und A(1) sowie C(0) und C(1) des betreffenden Kurvenstücks. XI,IORD,ND haben die gleiche Bedeutung wie in der Subroutine INTPL3, N ist hier die Anzahl der zu interpolierenden Punkte bei dem Kurvenstück. CM ist eine ND×4 Matrix, deren 1. Spalte die Koordinaten des 1. Punktes, 2. Spalte die Koordinaten des 2. Punktes, 3. Spalte den Tangenteneinheitsvektor am 1. Punkt und 4. Spalte den Tangenteneinheitsvektor am 2. Punkt enthält.

- Die Subroutine IHMATX(CM,AB,CD,ND,NH,HM) berechnet die NH × 4 Matrix HM in (9) für ein Kurvenstück aus bekannten Endpunkten, Tangenteneinheitsvektoren, die in der Matrix CM enthalten sind, und den 4 Kurvenparametern, die in den Feldern AB und CD enthalten sind. NH = ND + 1. Die Routine wird auch beim Zeichnen einer Kurve gebraucht.

- Die Subroutine ICXH(HM,NH,U,XH) berechnet die homogenen Koordinaten XH($\vec{p}(u)$ in (8)) des Punktes auf einem Kurvenstück mit einem gegebenen Parameter U.

- Die Funktion PU(HM,NH,IZ,XYZ) ermittelt den Parameter U zu einer gegebenen Koordinate XYZ(IZ). Zur Lösung der Gleichung 3. Grades für u wird die Newtonsche Näherungsmethode verwendet.

- Die Subroutine IHZUK(XH,NH,XK,ND) berechnet die den homogenen Koordinaten XH ($\vec{p}(u)$) entsprechenden kartesischen Koordinaten XK ($\vec{P}(u)$).

- Die Subroutine ICXH(HM,NH,U,DXH) berechnet den nach u abgeleiteten homogenen Vektor DXH ($\vec{p}_u(u)$) zu einem gegebenen Parameter U.

- Die Subroutine IRCOS(HM,NH,U,RCOS,ND) berechnet Richtungscosinus RCOS ($\vec{t}(u)$) zu einem gegebenen Parameter U.

- Die Subroutine INFG(U,FG) berechnet die Hermiteschen Polynome 3. Grades FG zu einem gegebenen U (wird auch beim Zeichnen benutzt).

- Die Subroutine INDDFG(U,DDFG) berechnet die 2. Ableitungen der Hermiteschen Polynome 3. Grades DDFG zu einem gegebenen U (wird auch beim Zeichnen benutzt).

4) Zeichnen

- Die Subroutine IDRAW3(X,AB,CD,TANV,ND,N) ruft ICDRAW auf, um eine Kurve zu zeichnen, die aus mehreren Kurvenstücken zusammengesetzt ist. Ihre Parameter haben gleiche Bedeutung wie die in INTPL3.

- Die Subroutine ICDRAW(HM,NH) zeichnet ein Kurvenstück, das durch die NH $\times$ 4 - Matrix HM in (9) definiert ist. Bei Raumkurven werden folgende Fälle unterschieden:

  a) Zeichnen von 3 Projektionen auf die Koordinatenebenen XY, XZ und YZ

  b) Zeichnen einer perspektivischen Darstellung (Zentralprojektion auf eine Koordinatenebene)

  c) Zeichnen von Kurvenstücken nach a) und b).

  Zur perspektivischen Darstellung wird zunächst die homogene Matrix HM mit einer Transformationsmatrix multipliziert. Die Produktmatrix wird zur Berechnung der zum Zeichnen benötigten Punkte in der Subroutine INAXHP benutzt.

- Die Subroutine ISEGM(HM,NH,U,NS,HMN) teilt ein Kurvenstück in NS Segmente auf. HMN ist ein 3D-Feld, das die homogenen Matrizen aller Segmente enthält.

- Die Subroutine IDWPRJ(XYZ,NP) zeichnet 3 Projektionen eines Raumkurvenstücks. Der Parameter XYZ ist ein Feld für die Koordinaten der NP zum Zeichnen benötigten Punkte.

  Beim 1. Aufruf wird in der Reihenfolge XY,XZ,YZ gezeichnet; dann YZ,XZ,XY; dann wieder XY,XZ,YZ, damit der Plotter optimal arbeitet.

- Die Subroutine INPDU(HM,NH,NP,DU) berechnet die Anzahl NP der zum Zeichnen benötigten Punkte und das Inkrement DU des

Parameters U zu einem gegebenen Maßstab Z und der Genauigkeit a wie bei der Subroutine IDRAW1.

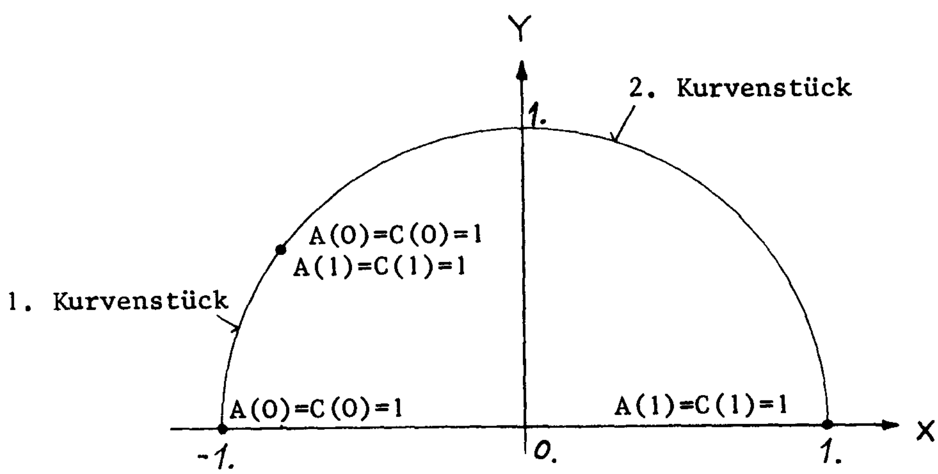
- Die Subroutine INAXHP(HM,NH,NP,DU,XHP) berechnet die zum Zeichnen benötigten Punkte mit dem sogenannten Differenzverfahren (siehe [5]). Die Koordinaten der Punkte sind im Feld XHP gespeichert.

- Die Subroutine INDQ(DU,DQ) liefert mit dem konstanten Inkrement DU die Anfangsmatrix DQ für das Differenzverfahren.

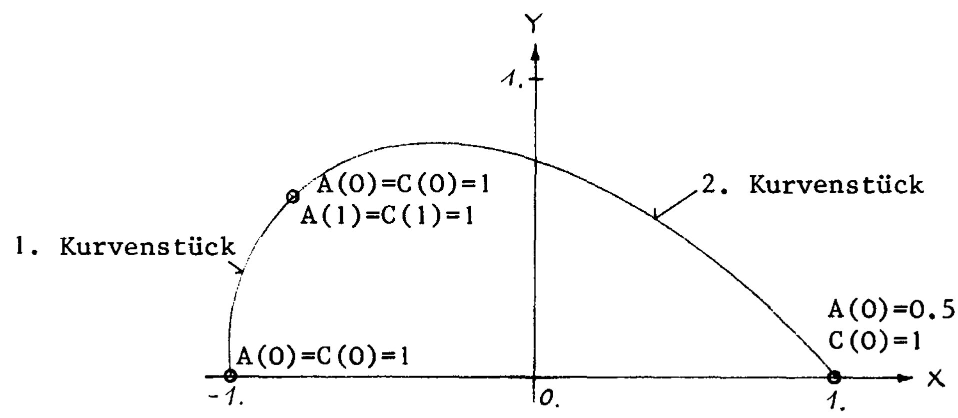
Zusätzlich werden noch 10 allgemeine Hilfsroutinen benutzt: POTP, ICROSS, SQT, WENN, IWENN, IVONZU, IVMALK, IVADD, IVSUBT, MAMU. Ihre Zwecke sind in den Quellprogrammen mit Kommentarzeilen beschrieben.

```
C   TESTPROGRAMM FUER DURCH GEBROCHENE KUBISCHE SPLINES
C   IN PARAMETERFORM DARGESTELLTE KURVE
      DIMENSION X(150),IBC(50),RBC(150),AB(2,50),CD(2,50)
      DIMENSION SL(50),SV(150),TANV(150),XI(550)
      COMMON /PROJ/LPROJ,ISEQ,XS,YS
      COMMON /PERS/LPERS,TRM(4,4),NE
      COMMON /ZMAC/ZM,AC
C
C   DATEN ZUR DARSTELLUNG EINER KURVE EINLESEN
      CALL ISPLND(X,IBC,RBC,AB,CD,ND,N)
C   TANGENTENEINHEITSVEKTOREN AN DEN STUETZPUNKTEN BESTIMMEN
      CALL ISPLTP(X,IBC,RBC,AB,CD,ND,N,SL,SV,TANV)
C
      READ(5,*)NI,IORD
      WRITE(6,1500)NI,IORD
      NI1=NI-1
      DX=(X(ND*(N-1)+IORD)-X(IORD)-0.01)/FLOAT(NI1)
      XI(IORD)=X(IORD)
      DO 100 I=1,NI1
      XI(ND*I+IORD)=XI(ND*(I-1)+IORD)+DX
  100 CONTINUE
C   INTERPOLATION DURCHFUEHREN
      CALL INTPL3(X,AB,CD,TANV,ND,N,XI,IORD,NI)
      NDNI=ND*NI
      WRITE(6,2000)(XI(I),I=1,NDNI)
C   ZEICHENMASSSTAB ZM UND GENAUIKEIT AC
      ZM=0.25
      AC=0.001
C   STEUERUNG FUER ZEICHNEN VON PROJEKTIONEN LPROJ UND
C   PERSPEKTIVISCHE DARSTELLUNG LPERS (NUR BEI RAUMKURVE)
      LPROJ=1
      LPERS=0
C   ABSTAND YS ZW. DEN X-ACHSEN DER XY- UND XZ-EBENEN UND XS
C   ZW. DEN Z-ACHSEN DER XZ- UND YZ-EBENEN (NUR BEI RAUMKURVE)
      XS=6.5
      YS=5.
C   PLOT INITIIEREN
      CALL PLOTS(0,0,13)
C   URSPRUNG DEFINIEREN
      CALL PLOT(8.,5.,-3)
C   URSPRUNG MIT SYMBOL + MARKIEREN
      CALL SYMBOL(0.,0.,0.7,3,0.,-1)
      IF(ND.EQ.2)GOTO 200
      CALL SYMBOL(0.,YS,0.7,3,0.,-1)
      CALL SYMBOL(XS,YS,0.7,3,0.,-1)
  200 CONTINUE
C   KURVE ZEICHNEN
      CALL IDRAW3(X,AB,CD,TANV,ND,N)
C   PLOT ABSCHLIESSEN
      CALL PLOT(20.,20.,999)
 1500 FORMAT(1X,5I5)
 2000 FORMAT(1X,6F11.5)
      STOP
      END
```

```
DATEN FUER KURVE
    2    3    1    1    0    0    1
-1.0000     0.00000E+00
-.80000     0.60000
 1.0000     0.00000E+00
    2
    2
   51    1
   -1.00000     0.00000    -0.96020     0.27931    -0.92040     0.39098
   -0.88060     0.47386    -0.84080     0.54135    -0.80100     0.59866
   -0.76120     0.64852    -0.72140     0.69252    -0.68160     0.73173
   -0.64180     0.76687    -0.60200     0.79850    -0.56220     0.82700
   -0.52240     0.85270    -0.48260     0.87584    -0.44280     0.89662
   -0.40300     0.91520    -0.36320     0.93171    -0.32340     0.94626
   -0.28360     0.95894    -0.24380     0.96983    -0.20400     0.97897
   -0.16420     0.98643    -0.12440     0.99223    -0.08460     0.99642
   -0.04480     0.99900    -0.00500     0.99999     0.03480     0.99939
    0.07460     0.99721     0.11440     0.99343     0.15420     0.98804
    0.19400     0.98100     0.23380     0.97228     0.27360     0.96184
    0.31340     0.94962     0.35320     0.93555     0.39300     0.91954
    0.43280     0.90149     0.47260     0.88128     0.51240     0.85875
    0.55220     0.83371     0.59200     0.80594     0.63180     0.77513
    0.67160     0.74091     0.71140     0.70279     0.75120     0.66008
    0.79100     0.61182     0.83080     0.55657     0.87060     0.49199
    0.91040     0.41373     0.95020     0.31164     0.99000     0.14107
```

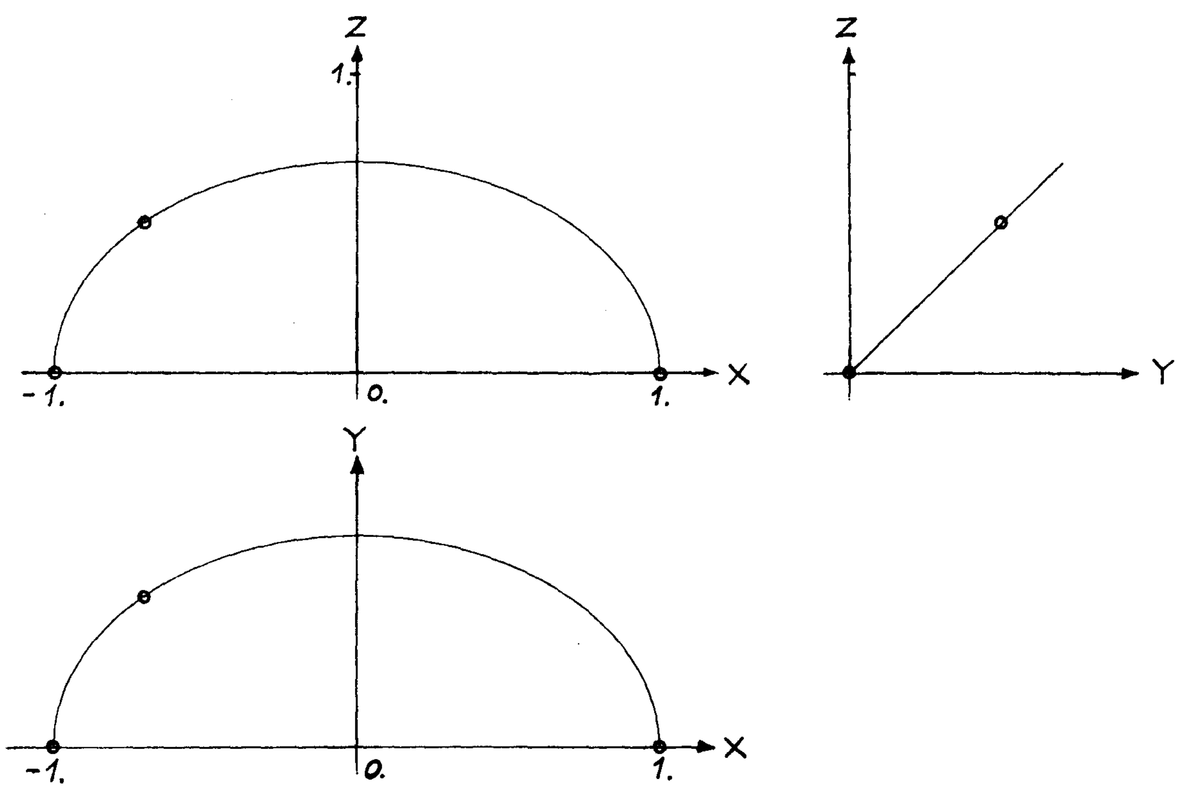


```
DATEN FUER KURVE
    2    3     1     1     0     1     1
-1.0000      0.00000E+00
-.80000      0.60000
 1.0000      0.00000E+00
    2
    2
           2    1.000000       0.5000000
   51    1
   -1.00000     0.00000    -0.96020    0.35576    -0.92040    0.44088
   -0.88060     0.50459    -0.84080    0.55603    -0.80100    0.59901
   -0.76120     0.63554    -0.72140    0.66665    -0.68160    0.69308
   -0.64180     0.71539    -0.60200    0.73404    -0.56220    0.74939
   -0.52240     0.76176    -0.48260    0.77140    -0.44280    0.77852
   -0.40300     0.78331    -0.36320    0.78594    -0.32340    0.78653
   -0.28360     0.78521    -0.24380    0.78208    -0.20400    0.77724
   -0.16420     0.77076    -0.12440    0.76271    -0.08460    0.75317
   -0.04480     0.74218    -0.00500    0.72979     0.03480    0.71605
    0.07460     0.70100     0.11440    0.68466     0.15420    0.66707
    0.19400     0.64825     0.23380    0.62822     0.27360    0.60700
    0.31340     0.58459     0.35320    0.56102     0.39300    0.53628
    0.43280     0.51037     0.47260    0.48330     0.51240    0.45505
    0.55220     0.42562     0.59200    0.39498     0.63180    0.36312
    0.67160     0.33000     0.71140    0.29558     0.75120    0.25982
    0.79100     0.22265     0.83080    0.18400     0.87060    0.14376
    0.91040     0.10180     0.95020    0.05794     0.99000    0.01193
```



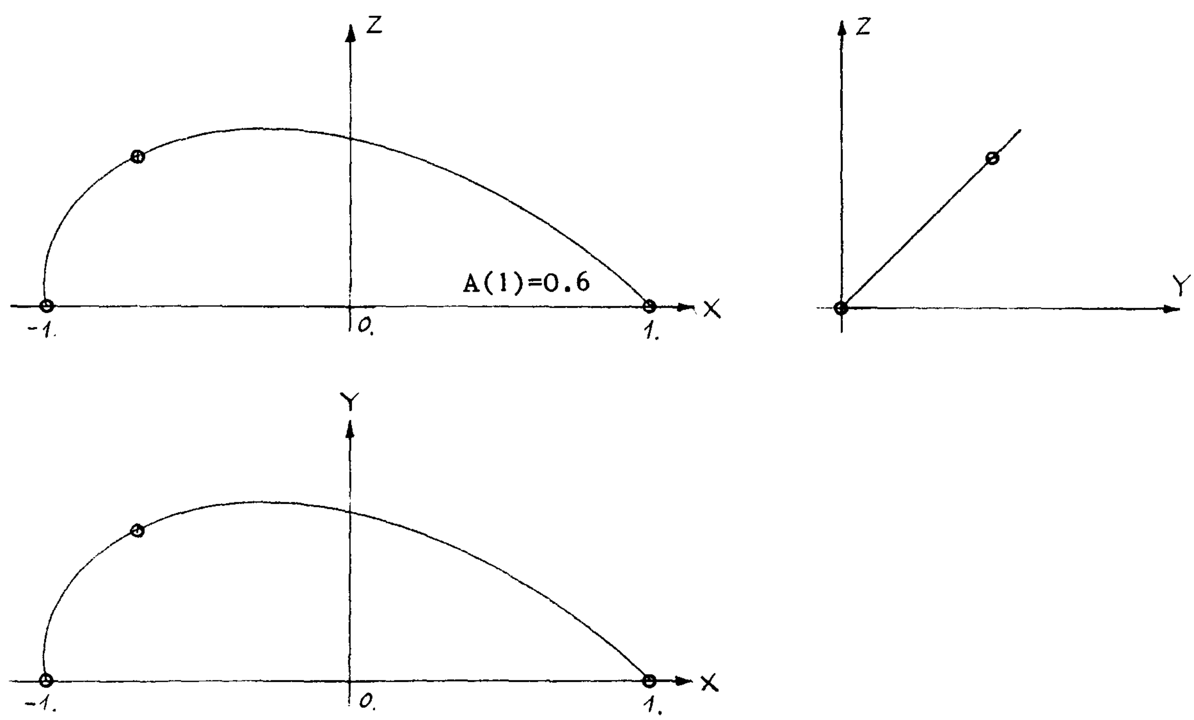
```
DATEN FUER KURVE
    3    3    1    1    0    0    1
-1.0000     0.00000E+000.00000E+00
-.70000     0.50498     0.50498
 1.0000     0.00000E+000.00000E+00
    2
    2
   51    1
```

-1.00000	0.00000	0.00000	-0.96020	0.19751	0.19751
-0.92040	0.27647	0.27647	-0.88060	0.33507	0.33507
-0.84080	0.38279	0.38279	-0.80100	0.42332	0.42332
-0.76120	0.45858	0.45858	-0.72140	0.48969	0.48969
-0.68160	0.51741	0.51741	-0.64180	0.54227	0.54227
-0.60200	0.56463	0.56463	-0.56220	0.58478	0.58478
-0.52240	0.60296	0.60296	-0.48260	0.61932	0.61932
-0.44280	0.63401	0.63401	-0.40300	0.64715	0.64715
-0.36320	0.65882	0.65882	-0.32340	0.66911	0.66911
-0.28360	0.67808	0.67808	-0.24380	0.68577	0.68577
-0.20400	0.69224	0.69224	-0.16420	0.69751	0.69751
-0.12440	0.70162	0.70162	-0.08460	0.70458	0.70458
-0.04480	0.70640	0.70640	-0.00500	0.70710	0.70710
0.03480	0.70668	0.70668	0.07460	0.70514	0.70514
0.11440	0.70247	0.70247	0.15420	0.69865	0.69865
0.19400	0.69368	0.69368	0.23380	0.68751	0.68751
0.27360	0.68013	0.68013	0.31340	0.67149	0.67149
0.35320	0.66154	0.66154	0.39300	0.65022	0.65022
0.43280	0.63745	0.63745	0.47260	0.62316	0.62316
0.51240	0.60723	0.60723	0.55220	0.58953	0.58953
0.59200	0.56989	0.56989	0.63180	0.54811	0.54811
0.67160	0.52391	0.52391	0.71140	0.49695	0.49695
0.75120	0.46675	0.46675	0.79100	0.43262	0.43262
0.83080	0.39356	0.39356	0.87060	0.34790	0.34790
0.91040	0.29256	0.29256	0.95020	0.22037	0.22037
0.99000	0.09976	0.09976			



```
DATEN FUER KURVE
    3     3     1     1     0     1     1
-1.0000     0.00000E+000.00000E+00
-.70000     0.50498     0.50498
 1.0000     0.00000E+000.00000E+00
    2
    2
          2    1.000000        0.6000000
  51    1
```

-1.00000	0.00000	0.00000	-0.96020	0.26463	0.26463
-0.92040	0.32672	0.32672	-0.88060	0.37341	0.37341
-0.84080	0.41126	0.41126	-0.80100	0.44302	0.44302
-0.76120	0.47019	0.47019	-0.72140	0.49366	0.49366
-0.68160	0.51405	0.51405	-0.64180	0.53171	0.53171
-0.60200	0.54689	0.54689	-0.56220	0.55978	0.55978
-0.52240	0.57055	0.57055	-0.48260	0.57936	0.57936
-0.44280	0.58631	0.58631	-0.40300	0.59154	0.59154
-0.36320	0.59513	0.59513	-0.32340	0.59716	0.59716
-0.28360	0.59773	0.59773	-0.24380	0.59688	0.59688
-0.20400	0.59469	0.59469	-0.16420	0.59120	0.59120
-0.12440	0.58647	0.58647	-0.08460	0.58052	0.58052
-0.04480	0.57341	0.57341	-0.00500	0.56515	0.56515
0.03480	0.55579	0.55579	0.07460	0.54534	0.54534
0.11440	0.53382	0.53382	0.15420	0.52126	0.52126
0.19400	0.50766	0.50766	0.23380	0.49305	0.49305
0.27360	0.47742	0.47742	0.31340	0.46079	0.46079
0.35320	0.44316	0.44316	0.39300	0.42452	0.42452
0.43280	0.40487	0.40487	0.47260	0.38421	0.38421
0.51240	0.36253	0.36253	0.55220	0.33981	0.33981
0.59200	0.31603	0.31603	0.63180	0.29116	0.29116
0.67160	0.26518	0.26518	0.71140	0.23805	0.23805
0.75120	0.20971	0.20971	0.79100	0.18011	0.18011
0.83080	0.14917	0.14917	0.87060	0.11681	0.11681
0.91040	0.08289	0.08289	0.95020	0.04727	0.04727
0.99000	0.00976	0.00976			



```
      SUBROUTINE ISPLND(X,IBC,RBC,AB,CD,ND,N)
C  ZUM EINLESEN VON DATEN, DIE EINE SPLINEKURVE DEFINIEREN
C  BESCHREIBUNG DER PARAMETER:
C    SIEHE DIE SUBROUTINEN 'ISPLNT' UND 'ITANV'
C  FUER CDSTD STEHT 1. ODER 0. ZUR AUSWAHL
C
      DIMENSION X(150),IBC(50),RBC(150),AB(2,50),CD(2,50)
      DIMENSION SL(50),SV(150),TANV(150)
C
      WRITE(6,550)
C  ANZAHL DER DIMENSION, PUNKTE, VON STANDARD ABWEICHENDEN
C  NEBENBEDINGUNGEN UND PARAMETER EINLESEN
      READ(5,*)ND,N,IBC1,IBCN,NB,NCHAB,ICD
      WRITE(6,750)ND,N,IBC1,IBCN,NB,NCHAB,ICD
      DO 200 J=1,N
      I1=(J-1)*ND+1
      I2=I1+ND-1
      READ(5,*)(X(I),I=I1,I2)
      WRITE(6,700)(X(I),I=I1,I2)
  200 CONTINUE
C  STANDARDWERTE EINSETZEN
      CDSTD=1.
      IF(ICD.EQ.0)CDSTD=0.
      CALL ISTDP(IBC,RBC,AB,CD,CDSTD,ND,N)
C  KENNZAHL FUER ENDBEDINGUNGEN BZW. AUCH DIE WERTE
C  (BEI VORGEGEB. TANGENTENWINKEL) EINLESEN
      IF(IBC1.NE.0)READ(5,*)IBC(1)
      IF(IBCN.NE.0)READ(5,*)IBC(N)
      WRITE(6,750)IBC(1)
      WRITE(6,750)IBC(N)
      IF(IBC1.EQ.0)GOTO 310
      IB=IBB(IBC(1))
      IF(IB.EQ.2)GOTO 310
      ID1=1
      IF(IB.NE.1)ID1=IWENN(IB.LT.6,ND-1,ND)
      READ(5,*)(RBC(I),I=1,ID1)
      WRITE(6,700)(RBC(I),I=1,ID1)
      IF(IB.NE.1)CALL IWITV(RBC(1),IBC(1),ND)
  310 CONTINUE
      IF(IBCN.EQ.0)GOTO 360
      IF(IB.EQ.2)GOTO 360
      ID1=1
      IB=IBB(IBC(N))
      IF(IB.NE.1)ID1=IWENN(IB.LT.6,ND-1,ND)
      I1=(N-1)*ND+1
      I2=I1+ID1-1
      READ(5,*)(RBC(I),I=I1,I2)
      WRITE(6,700)(RBC(I),I=I1,I2)
      IF(IB.NE.1)CALL IWITV(RBC(I1),IBC(N),ND)
  360 CONTINUE
C
C  GGF. KENNZAHL FUER NEBENBEDINGUNGEN BZW. AUCH DIE WERTE
C  (BEI VORGEGEB. TANGENTENWINKEL) EINLESEN
      IF(NB.EQ.0)GOTO 450
      DO 400 L=1,NB
      READ(5,*)INDP,IBC(INDP)
      WRITE(6,750)INDP,IBC(INDP)
      IB=IBB(IBC(INDP))
      IF(IB.LT.3)GOTO 400
      ID1=IWENN(IB.LT.6,ND-1,ND)
      I1=(INDP-1)*ND+1
```

```
      I2=I1+ID1-1
      READ(5,*)(RBC(I),I=I1,I2)
      WRITE(6,700)(RBC(I),I=I1,I2)
      CALL IWITV(RBC(I1),IBC(INDP),ND)
  400 CONTINUE
  450 CONTINUE
C
C  GGF. NICHT-STANDARDE PARAMETER EINLESEN
      IF(NCHAB.EQ.0)GOTO 510
      DO 500 I=1,NCHAB
      READ(5,*)IAB,AB(1,IAB),AB(2,IAB)
      WRITE(6,*)IAB,AB(1,IAB),AB(2,IAB)
  500 CONTINUE
  510 CONTINUE
  550 FORMAT(17HODATEN FUER KURVE)
  700 FORMAT(1X,10G11.5)
  750 FORMAT(1X,10I5)
      RETURN
      END
```

```
      SUBROUTINE ISTDP(IBC,RBC,AB,CD,CDSTD,ND,N)
C  ZUR EINSETZUNG STANDARDPARAMETER FUER 2D- BZW. 3D-KURVEN
C  BESCHREIBUNG DER PARAMETER:
C    SIEHE DIE SUBROUTINEN 'ISPLNT' UND 'ITANV'
C  FUER PARAMETER CDSTD STEHT 1. ODER 0. ZUR AUSWAHL
C
      DIMENSION IBC(N),RBC(ND,N),AB(2,1),CD(2,1)
C
      N1=N-1
      DO 100 I=1,N
      IBC(I)=0
      DO 100 J=1,ND
      RBC(J,I)=0.
  100 CONTINUE
      IBC(1)=1
      RBC(1,1)=1.
      IBC(N)=1
      RBC(1,N)=1.
      DO 200 I=1,N1
      DO 200 J=1,2
      AB(J,I)=1.
      CD(J,I)=CDSTD
  200 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE IWITV(RBC,IBC,ND)
C  ZUM BERECHNEN TANGENTENEINHEITSVEKTOR AUS
C  GEGEB. WINKELN BZW. TANGENTENVEKTOR
      DIMENSION RBC(ND)
      DATA RAD/0.01745329/
      I1=IABS(IBC)
      I2=I1/10
      IF(I1-I2*10.GE.6)GOTO 200
      IF(ND.EQ.3)GOTO 100
      RB=RBC(1)*RAD
      RBC(1)=COS(RB)
      RBC(2)=SIN(RB)
      GOTO 350
  100 CALL IBEDUM(RBC,IBC,ND)
      GOTO 350
  200 BETR=SQT(RBC,ND)
      DO 300 I=1,ND
      RBC(I)=RBC(I)/BETR
  300 CONTINUE
  350 IBC=ISIGN(1,IBC)*(I2*10+3)
      RETURN
      END



      SUBROUTINE IBEDUM(RBC,IBC,ND)
C  ZUM UMSTELLEN DER WINKELN FUER RAUMKURVEN
C
      DIMENSION RBC(ND),IUM(6)
      DATA IUM/0,0, 1,0, 1,1/
C
      I1=IABS(IBC)/10
      IND=(IABS(IBC)-I1*10-3)*2
      DO 100 I=1,2
      JN=IUM(IND+I)
      IF(JN.EQ.1)RBC(I)=90.-RBC(I)
  100 CONTINUE
      CALL IPWTV(RBC,IBC,ND)
      RETURN
      END



      FUNCTION IBB(IBC)
C  ZUM UMRECHNEN DER KENNZAHL ZU END- BZW. NEBENBEDING.
      IA=IABS(IBC)
      IC=IA/10
      IBB=IA-IC*10
      RETURN
      END
```

```
      SUBROUTINE IPWTV(TV,IBC,ND)
C   ZUM UMRECHNEN TANGENTENRICHTUNG IM RAUM
C   MIT ZWEI WINKELN IN ZWEI PROJEKTIONSEBENEN
C
      DIMENSION TV(ND),ISP(9)
      DATA RAD/0.01745329/
      DATA ISP/1,2,3, 2,1,3, 3,1,2/
C
      TAN1=TAN(TV(1)*RAD)
      TAN2=TAN(TV(2)*RAD)
      SGN=SIGN(1.,COS(TV(1)*RAD))
      I1=IABS(IBC)/10
      IND=(IABS(IBC)-I1*10-3)*3+1
      I1=ISP(IND)
      TV(I1)=SGN*SQRT(1./(1.+TAN1**2+TAN2**2))
      I2=ISP(IND+1)
      TV(I2)=TV(I1)*TAN1
      I2=ISP(IND+2)
      TV(I2)=TV(I1)*TAN2
      RETURN
      END




      FUNCTION SQT(X,N)
      DIMENSION X(N)
      SQT=0.
      DO 100 I=1,N
      SQT=SQT+X(I)**2
  100 CONTINUE
      SQT=SQRT(SQT)
      RETURN
      END




      FUNCTION IWENN(L,IA,IB)
C     =====================
      LOGICAL L
      IF(L)GOTO 1
      IWENN=IB
      RETURN
    1 IWENN=IA
      RETURN
      END
```

```
      SUBROUTINE ISPLTP(X,IBC,RBC,AB,CD,ND,N,SL,SV,TANV)
C  ZUR BESTIMMUNG TANGENTENV. UND EVENTL. VON NEBENBEDING.
C  ABHAENGIGER AENDERUNG DER PARAMETER AB, CD
C  BESCHREIBUNG DER PARAMETER:
C    SIEHE DIE SUBROUTINEN 'ISPLNT' UND 'ITANV'
C
      DIMENSION X(ND,N),IBC(N),RBC(ND,N),AB(2,1)
      DIMENSION TANV(ND,N),SL(1),SV(ND,1),CD(2,1)
      DIMENSION SA(3),SB(3),RKRL(3),RKRR(3)
      DATA EPS/1.E-5/
C
C  ERMITTELN TANGENTENVEKTOREN AN STUETZPUNKTEN
      MAXIT=20
      CALL ISPLNT(X,IBC,RBC,AB,CD,SL,SV,ND,N,MAXIT,EPS,TANV)
C  FESTSTELLEN, OB PARAMETER GEAENDERT WERDEN MUESSEN
      N1=N-1
      DO 600 I=1,N1
      DO 500 J=1,2
      IJ=I+J-1
C  TANGENTENRICHTUNG UND STETIGER KRUEMMUNG
      IF(IBC(IJ).EQ.-3)GOTO 300
      IB=IABS(IBC(IJ))/10
      IF(IB.NE.3-J)GOTO 500
C  ABSCHNITT ZU KEGELSCHNITT
      CALL ICROSS(TANV(1,I),SV(1,I),SA,ND)
      CALL ICROSS(TANV(1,I+1),SV(1,I),SB,ND)
      AZUB=(SQT(SB,ND)+1.E-30)/(SQT(SA,ND)+1.E-30)
      GOTO(260,230),J
  230 AB(2,I)=AB(1,I)/AZUB
      GOTO 500
  260 AB(1,I)=AB(2,I)*AZUB
      GOTO 500
C  NEBENBEDINGUNG MIT GEGEB. TANGENTENRICHT. UND STETIGER KRUEMM.
  300 IBC(IJ)=0
      I1=IJ-1
      CALL IKREND(SV(1,I1),SL(I1),TANV(1,I1),
     *            AB(1,I1),CD(1,I1),RKRL,ND,2)
      CALL IKREND(SV(1,IJ),SL(IJ),TANV(1,IJ),
     *            AB(1,IJ),CD(1,IJ),RKRR,ND,1)
      IF(SQT(RKRL,ND)-SQT(RKRR,ND))330,500,360
  330 CD(2,IJ)=CDK(SV(1,IJ),SL(IJ),TANV(1,IJ),AB(1,IJ),RKRL,ND,1)
      GOTO 500
  360 CD(1,I1)=CDK(SV(1,I1),SL(I1),TANV(1,I1),AB(1,I1),RKRR,ND,2)
C  ENDBEDINGUNG MIT GEGEB. TANGENTENRICHT. UND KRUEMM.
  500 CONTINUE
  600 CONTINUE
      RETURN
```

```
      SUBROUTINE IKREND(SV,SL,TV,AB,CD,RKR,ND,LR)
C  ZUR BERECHNUNG KRUEMMUNGSV. AN EINEM ENDE EINES KURVENST.
C
      DIMENSION SV(ND),TV(ND,2),AB(2),CD(2),RKR(ND)
      DIMENSICN ZV1(3),ZV2(3),ZV3(3)
C
      CALL IVMALK(AB(1),TV(1,1),ZV1,ND)
      CALL IVMALK(AB(2),TV(1,2),ZV2,ND)
      CALL IVADD(ZV1,ZV2,ZV3,ND)
      SKP=DOTP(TV(1,1),TV(1,2),ND)
      GW=SQT(ZV3,ND)-2.
      GOTO(100,200),LR
  100 HU=3.+CD(2)*GW
      CALL IVMALK(HU,SV,ZV1,ND)
      CALL IVMALK(-HU*DOTP(TV(1,1),SV,ND)+AB(2)*SKP,TV(1,1),ZV2,ND)
      CALL IVADD(ZV1,ZV2,ZV3,ND)
      CALL IVMALK(-AB(2),TV(1,2),ZV1,ND)
      CALL IVADD(ZV3,ZV1,ZV2,ND)
      CALL IVMALK(2./SL/AB(1)**2,ZV2,RKR,ND)
      RETURN
  200 HU=3.+CD(1)*GW
      CALL IVMALK(-HU,SV,ZV1,ND)
      CALL IVMALK(HU*DOTP(TV(1,2),SV,ND)-AB(1)*SKP,TV(1,2),ZV2,ND)
      CALL IVADD(ZV1,ZV2,ZV3,ND)
      CALL IVMALK(HU*DOTP(TV(1,2),SV,ND)-AB(1)*SKP,TV(1,2),ZV2,ND)
      CALL IVMALK(HU*DOTP(TV(1,2),SV,ND)-AB(1)*SKP,TV(1,2),ZV2,ND)
      CALL IVADD(ZV1,ZV2,ZV3,ND)
      CALL IVMALK(AB(1),TV(1,1),ZV1,ND)
      CALL IVADD(ZV3,ZV1,ZV2,ND)
      CALL IVMALK(2./SL/AB(2)**2,ZV2,RKR,ND)
      RETURN
      END

      FUNCTICN HABL(SV,SL,TV,AB,RKR,ND,LR)
C  ZUR BERECHNUNG DER ABLEIT. VON H MIT GEGEB. KRUEMM.
      DIMENSICN SV(ND),TV(ND,2),RKR(ND),AB(2)
      SKP=DOTP(TV(1,1),TV(1,2),ND)
      GOTO(100,200),LR
  100 HABL=(-AB(2)*(SKP**2-1.)+AB(1)**2*SL*DOTP(TV(1,2),RKR,ND)*0.5)
     *      /(DOTP(TV(1,2),SV,ND)-DOTP(TV(1,1),SV,ND)*SKP)-3.
      RETURN
  200 HABL=(AB(1)*(SKP**2-1.)+AB(2)**2*SL*DOTP(TV(1,1),RKR,ND)*0.5)
     *      /(-DOTP(TV(1,1),SV,ND)+DOTP(TV(1,2),SV,ND)*SKP)-3.
      RETURN
      END

      FUNCTICN CDK(SV,SL,TV,AB,RKR,ND,LR)
C  ZUR BERECHNUNG C ODER D MIT GEGEB. KRUEMM.
C  LR=1: CD(2);   LR=2: CD(1)
C
      DIMENSICN SV(ND),TV(ND,2),RKR(ND),AB(2),SA(3),SB(3),SC(3)
      DATA EFZEHN/1.E-15/
C
      CALL IVMALK(AB(1),TV(1,1),SA,ND)
      CALL IVMALK(AB(2),TV(1,2),SB,ND)
      CALL IVADD(SA,SB,SC,ND)
      GW=SQT(SC,ND)-2.
      CDK=HABL(SV,SL,TV,AB,RKR,NC,LR)/(GW+SIGN(1.,GW)*EFZEHN)
      RETURN
      END
```

```
      SUBROUTINE ISPLNT(X,IBC,RBC,AB,CD,SL,SV,ND,N,MAXIT,EPS,TANV)
C  ZUR ITERATIVEN BESTIMMUNG DER TANGENTENVEKTOREN
C  MIT GEGEBENEN PARAMETERN
C
C     EINGANG:
C       X      -FELD MIT KOORD. GEGEB. STUETZPKT. MIT DIM. ND*N
C       IBC..N-SIEHE SUBROUTINE 'ITANV'
C       MAXIT -MAX. ANZAHL ZUGELASSENER ITERATIONSSCHRITTE
C       EPS    -GEFORDERTE GENAUIGKEIT
C     AUSGANG:
C       TANV   -ZU ERMITTELNDE TANGENTENVEKTOREN
C
      DIMENSION X(ND,N),IBC(N),RBC(ND,N),AB(2,1)
      DIMENSION TANV(ND,N),SL(1),SV(ND,1)
      DIMENSION TANVA(150),AA(150),FAKT(6)
      DATA FAKT/1.,1., 1.,0., 0.,1./,VK/1.5/
      DATA IOUT/6/
C
      N1=N-1
C  SEHNELAENGE UND SEHNEVEKTOR
      CALL ISLSV(X,ND,N,SL,SV)
C  ANFANGSVEKTOREN ZUWEISEN
      DO 150 J=2,N1
      IB=IBB(IBC(J))
      IF(IB.EQ.3)GOTO 110
      IZ=IB*2+1
      DO 100 I=1,ND
      TANV(I,J)=FAKT(IZ)*SV(I,J-1)*SL(J)
     *          +FAKT(IZ+1)*SV(I,J)*SL(J-1)
  100 CONTINUE
      CALL IEINHV(TANV(1,J),ND)
      GOTO 130
  110 DO 120 I=1,ND
      TANV(I,J)=RBC(I,J)
  120 CONTINUE
  130 DO 140 I=1,ND
      TANVA((I-1)*N+J)=TANV(I,J)
  140 CONTINUE
  150 CONTINUE
      IF(IBB(IBC(1)).EQ.3)GOTO 160
      DOT=(1.+VK-(VK-1.)*RBC(1,1))*DOTP(TANV(1,2),SV(1,1),ND)
      DO 151 I=1,ND
      TANV(I,1)=DOT*SV(I,1)-TANV(I,2)
  151 CONTINUE
      CALL IEINHV(TANV(1,1),ND)
      DO 155 I=1,ND
      TANVA((I-1)*N+1)=TANV(I,1)
  155 CONTINUE
      GOTO 180
  160 DO 170 I=1,ND
      TANV(I,1)=RBC(I,1)
      TANVA((I-1)*N+1)=TANV(I,1)
  170 CONTINUE
  180 IF(IBB(IBC(N)).EQ.3)GOTO 200
      DOT=(1.+VK-(VK-1.)*RBC(1,N))*DOTP(TANV(1,N1),SV(1,N1),ND)
      DO 185 I=1,ND
      TANV(I,N)=DOT*SV(I,N1)-TANV(I,N1)
  185 CONTINUE
      CALL IEINHV(TANV(1,N),ND)
```

```
      DO 190 I=1,ND
      TANVA(I*N)=TANV(I,N)
  190 CONTINUE
      GOTO 220
  200 DO 210 I=1,ND
      TANV(I,N)=RBC(I,N)
      TANVA(I*N)=TANV(I,N)
  210 CONTINUE
  220 M1=0
  250 M1=M1+1
      IF(M1.GT.MAXIT)GOTO 400
      CALL ITANV(IBC,RBC,AB,CD,SL,SV,ND,N,TANVA,AA)
C   TANGENTENVEKTOREN NORMIEREN
      DO 280 J=1,N
      VL=0.
      DO 260 I=1,ND
      II=(I-1)*N
      VL=VL+TANVA(II+J)**2
  260 CONTINUE
      VL=SQRT(VL)
      DO 270 I=1,ND
      II=(I-1)*N
      TANVA(II+J)=TANVA(II+J)/VL
  270 CONTINUE
  280 CONTINUE
CC
C   GENAUIGKEIT PRUEFEN
      IFLAG=0
      DO 290 J=1,N
      DO 290 I=1,ND
      II=(I-1)*N
      IF(ABS(TANV(I,J)-TANVA(II+J)).GT.EPS)GOTO 300
  290 CONTINUE
      MAXIT=M1
      IFLAG=1
  300 DO 350 I=1,ND
      II=(I-1)*N
      DO 350 J=1,N
      TANV(I,J)=TANVA(II+J)
  350 CONTINUE
      IF(IFLAG.NE.1)GOTO 220
      RETURN
  400 WRITE(IOUT,1000)
 1000 FORMAT(44H ***NO CONVERG., MAX. NUMBER OF ITERAT. USED)
      RETURN
      END




      SUBROUTINE IEINHV(VEKT,ND)
C   ZUM BILDEN EINHEITSVEKTOR
      DIMENSION VEKT(ND)
      SQV=SQT(VEKT,ND)+1.E-30
      DO 100 I=1,ND
      VEKT(I)=VEKT(I)/SQV
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ISLSV(X,ND,N,SL,SV)
C  ZUR BERECHNUNG SEHNELAENGE UND VEKTOREN, DIE
C  STUETZPUNKTE VERBINDEN
C
C    EINGANG:
C      X....N-SIEHE SUBROUTINEN 'ISPLNT' UND 'ITANV'
C    AUSGANG:
C      SL..SV-SIEHE SUBROUTINE 'ITANV'
C
      DIMENSION X(ND,N),SL(1),SV(ND,1),EPS(3)
      DATA EPS/1.E-8,2*1.E-9/
C
      N1=N-1
      DO 200 I=1,N1
      DO 100 J=1,ND
      SV(J,I)=X(J,I+1)-X(J,I)+EPS(J)
  100 CONTINUE
      SL(I)=SQT(SV(1,I),ND)
      DO 150 J=1,ND
      SV(J,I)=SV(J,I)/SL(I)
  150 CONTINUE
  200 CONTINUE
      RETURN
      END


      FUNCTION DOTP(U,V,ND)
C  ZUM BILDEN SKALARPRODUKT
      DIMENSION U(ND),V(ND)
      DOTP=0.
      DO 100 I=1,ND
      DOTP=DOTP+U(I)*V(I)
  100 CONTINUE
      RETURN
      END

      SUBROUTINE ICROSS(U,V,W,ND)
C  ZUM BILDEN VEKTORPRODUKT
      DIMENSION U(ND),V(ND),W(ND)
      IF(ND.EQ.3)GOTO 100
      W(1)=U(1)*V(2)-U(2)*V(1)
      W(2)=0.
      RETURN
  100 W(1)=U(2)*V(3)-U(3)*V(2)
      W(2)=U(3)*V(1)-U(1)*V(3)
      W(3)=U(1)*V(2)-U(2)*V(1)
      RETURN
      END

      SUBROUTINE MAMU(X,Y,Z,IX,JX,JY)
C  MATRIXMULTIPLIKATION Z = X * Y
      DIMENSION X(IX,JX),Y(JX,JY),Z(IX,JY)
      DO 20 I=1,IX
      DO 20 K=1,JY
      SUM=0.
      DO 10 J=1,JX
   10 SUM=SUM+X(I,J)*Y(J,K)
   20 Z(I,K)=SUM
      RETURN
      END
```

```
      SUBROUTINE ITANV(IBC,RBC,AB,CD,SL,SV,ND,N,TANV,AA)
C  ZUR ERMITTELUNG TANGENTENVEKTOREN EINER KURVE MIT
C  GEGEB. STUETZPUNKTEN RANDBEDINGUNGEN BZW. NEBENBEDING.
C     EINGABE:
C       IBC -FELD FUER KENNZAHL DER NEBENBEDINGUNG
C             IBC=2: K'(1,0)=0. BZW. K'(N-1,1)=0.
C                    K IST HIER KRUEMMUNG
C             IBC=3 BIS 5:
C                 BEI RAUMKURVEN, 2 WINKELN IN 2 PROJEKT.EBENEN GEGEB.
C                 3(XY-XZ); 4(XY-YZ); 5(XZ-YZ).
C                 BEI EBENEKURVEN, 1 WINKEL ZW. ABSZ. UND TANG. GEGEB.
C             IBC=6: P'(1,0) BZW. P'(N-1,1) D.H. TANGENTENV. GEGEB.
C             FUER I=1, IBC=23 BIS 26: WIE IBC=3 BIS 6 ZUSAETZL.
C                       MIT KEGELSCHNITT ZW. P(1) UND P(2).
C             FUER I=N, IBC=13 BIS 16: WIE IBC=3 BIS 6 ZUSAETZL.
C                       MIT KEGELSCHNITT ZW. P(N-1) UND P(N).
C             FUER I=2,N-1,1
C             IBC= 0: K(I-1,1)=K(I,0)
C                     K IST HIER KRUEMMUNG
C             IBC= 1: ENTSPR. WL, K UNSTETIG
C             IBC= 2: ENTSPR. HL, K UNSTETIG
C             IBC= 3 BIS 5: WIE OBEN BEI I=1 BZW. I=N
C             IBC= 6: P'(I-1,1)=P'(I,0) D.H. TANGENTENV. GEGEB.
C             IBC=-3 BIS -6: WIE 3 BIS 6, ZUSAETZL. MIT STETIGEM KR.V.
C             IBC=11 BIS 16: WIE IBC=1 BIS 6, ZUSAETZL. MIT
C                            KEGELSCHNITT ZW. P(I-1) UND P(I).
C             IBC=21 BIS 26: WIE IBC=1 BIS 6, ZUSAETZL. MIT
C                            KEGELSCHNITT ZW. P(I) UND P(I+1).
C       RBC -FELD FUER WERTE ZUR GEGEB. NEBEN- BZW. ENDBEDING.
C       AB  -GEGEBENE PARAMETER (STANDARDWERT 1.)
C       CD  -GEGEBENE PARAMETER (1. ODER 0.)
C       SL  -SEHNELAENGE
C       SV  -SEHNEVEKTOR, (P(I+1)-P(I))/ABS(P(I+1)-P(I))
C       ND  -2 BEI 2D KURVE; 3 BEI 3D KURVE
C       N   -ANZAHL DER STUETZPUNKTE
C     AUSGABE:
C       TANV-TANGENTENVEKTOR AN DEN STUETZPUNKTEN
C     HILFSFELD:
C       AA  -HILFSFELD FUER KOEFFIZIENTEMATRIX, MIND. 3*N
C
      DIMENSION IBC(N),RBC(ND,N),AB(2,1),SL(1),SV(ND,1),TANV(N,ND)
      DIMENSION AA(1),V1(3),V2(3),V3(3)
      DATA MLD,MUD/2*1/,EPS/1.E-7/
      DATA ESECHS/1.E-6/,HSECHS/0.5E-6/
      DATA IOUT/6/
C
      N1=N-1
      NEA=N1*3
C  DATEN FUER ANFANGPUNKT BERECHNEN
      CALL IVONZU(TANV(1,1),N,V1,1,ND)
      CALL IVONZU(TANV(2,1),N,V2,1,ND)
      SK12=DOTP(V1,V2,ND)
      SKVS12=DOTP(V1,SV(1,1),ND)
      SKVS21=DOTP(V2,SV(1,1),ND)
      MIN=(SQRT(ABS(AB(1,1)**2+AB(2,1)**2
     *   +2.*AB(1,1)*AB(2,1)*SK12))-2.)*CD
      DDH=3.+MIN
      ZDH=DDH-1.
      EDH=DDH-2.
```

```
      IB=IABS(IBC(1))
      IC=IB/10
      IB=IB-IC*10
      GOTO(150,150,400),IB
  150 XX=DDH*SKVS12-ZDH*AB(1,1)-AB(2,1)*SK12
      GOTO(200,300),IB
C
C   MIT RANDBEDINGUNG K(1,0)=RBC(1,1)*K(1,1)
  200 AA(2)=1.5*AB(2,1)**5*(AB(1,1)*EDH+2.*XX)+ESECHS
      AA(1)=AB(2,1)**4*(-1.5*AB(1,1)*((-ZDH-HIN*DDH)*SKVS12
     *        +AB(2,1)*EDH*SK12)+XX*(3.*AB(1,1)*ZDH+4.*XX)
     *        -(DDH**2+(ZDH*AB(1,1))**2+AB(2,1)**2-2.*DDH*ZDH
     *        *AB(1,1)*SKVS12-2.*DDH*AB(2,1)*SKVS21+2.*ZDH
     *        *AB(1,1)*AB(2,1)*SK12))+RBC(1,1)**2*AB(1,1)**4
     *        *(DDH**2*(1.-SKVS21**2)+2.*DDH*AB(1,1)*(-SKVS12
     *        +SKVS21*SK12)+AB(1,1)**2*(1.-SK12**2))
      SQHH=-1.5*AB(2,1)**4*(-DDH*(2.*XX+AB(1,1)*HIN)
     *      -AB(1,1)*ZDH)+HSECHS
      DO 210 K=1,ND
      TANV(1,K)=SQHH*SV(K,1)
  210 CONTINUE
      GOTO 500
C
C   MIT RANDBEDINGUNG K'(1,0)=0.
  300 AA(2)=AB(2,1)*(EDH*AB(1,1)+2.*XX)
      SQHH=DDH*(2.*XX+HIN*AB(1,1))+ZDH*AB(1,1)
      AA(1)=-AA(2)*SK12+SQHH*SKVS12
      DO 310 K=1,ND
      TANV(1,K)=SQHH*SV(K,1)
  310 CONTINUE
      GOTO 500
C
C   TANGENTENRICHTUNG GEGEBEN
  400 AA(1)=1.
      AA(2)=0.
      DO 410 K=1,ND
      TANV(1,K)=RBC(K,1)
  410 CONTINUE
  500 CONTINUE
C
C   DATEN FUER ZWISCHENPUNKTEN
      IF(N.EQ.2)GOTO 1050
      DO 1000 I=2,N1
      I1=I-1
      I2=I+1
      J=I1*3
      CALL IVONZU(TANV(I2,1),N,V3,1,ND)
      SK23=DOTP(V2,V3,ND)
      SKVS23=DOTP(V2,SV(1,I),ND)
      SKVS32=DOTP(V3,SV(1,I),ND)
      HIN1=(SQRT(ABS(AB(1,I)**2+AB(2,I)**2
     *      +2.*AB(1,I)*AB(2,I)*SK23))-2.)*CD
      DDH1=3.+HIN1
      ZDH1=DDH1-1.
      EDH1=DDH1-2.
C
      IB=IABS(IBC(I))
      IC=IB/10
      IB=IB-IC*10+1
      GOTO(960,970,980,990),IB
C
```

```
C  MIT STETIGER KRUEMMUNG
 960 AALI=AB(1,I)**2*SL(I)
     BBLI=AB(2,I1)**2*SL(I1)
     AA(J)=AB(1,I1)*AALI
     AA(J+1)=AALI*(DDH*SKVS21-AB(1,I1)*SK12)
    *        +BBLI*(DDH1*SKVS23-AB(2,I)*SK23)
     AA(J+2)=AB(2,I)*BBLI
     AALI=AALI*DDH
     BBLI=BBLI*DDH1
     DO 965 K=1,ND
     TANV(I,K)=AALI*SV(K,I1)+BBLI*SV(K,I)
 965 CONTINUE
     GOTO 997
C
C  MIT WL-BEDINGUNG
 970 XX=-DDH*SKVS12+ZDH*AB(2,I1)+AB(1,I1)*SK12
     AA(J)=1.5*AB(1,I1)**5*(AB(2,I1)*EDH-2.*XX)+ESECHS
     AA(J+1)=AB(1,I1)**4*(-1.5*AB(2,I1)*((-ZDH-HIN*DDH)*SKVS21
    *        +AB(1,I1)*EDH*SK12)+XX*(-3.*AB(2,I1)*ZDH+4.*XX)
    *        -(DDH**2+(ZDH*AB(2,I1))**2+AB(1,I1)**2-2.*DDH
    *        *ZDH*AB(2,I1)*SKVS21-2.*DDH*AB(1,I1)*SKVS12
    *        +2.*ZDH*AB(1,I1)*AB(2,I1)*SK12))+AB(2,I1)**4
    *        *(DDH**2*(1.-SKVS12**2)-2.*DDH*AB(2,I1)*(SKVS21
    *        -SKVS12*SK12)+AB(2,I1)**2*(1.-SK12**2))
     AA(J+2)=0.
     SQHH=-1.5*AB(1,I1)**4*(DDH*(2.*XX-AB(2,I1)*HIN)
    *      -AB(2,I1)*ZDH)+HSECHS
     DO 975 K=1,ND
     TANV(I,K)=SQHH*SV(K,I1)
 975 CONTINUE
     GOTO 997
C
C  MIT HL-BEDINGUNG
 980 XX=DDH1*SKVS23-ZDH1*AB(1,I)-AB(2,I)*SK23
     AA(J)=0.
     AA(J+1)=AB(2,I)**4*(-1.5*AB(1,I)*((-ZDH1-HIN1*DDH1)*SKVS23
    *    +AB(2,I)*EDH1*SK23)+XX*(3.*AB(1,I)*ZDH1+4.*XX)-(DDH1**2
    *        +(ZDH1*AB(1,I))**2+AB(2,I)**2-2.*DDH1*ZDH1*AB(1,I)
    *        *SKVS23-2.*DDH1*AB(2,I)*SKVS32+2.*ZDH1*AB(1,I)*AB(2,I)
    *        *SK23))+AB(1,I)**4*(DDH1**2*(1.-SKVS32**2)+2.*DDH1
    *    *AB(1,I)*(-SKVS23+SKVS32*SK23)+AB(1,I)**2*(1.-SK23**2))
     AA(J+2)=1.5*AB(2,I)**5*(AB(1,I)*EDH1+2.*XX)+ESECHS
     SQHH=-1.5*AB(2,I)**4*(-DDH1*(2.*XX+AB(1,I)*HIN1)
    *      -AB(1,I)*ZDH)+HSECHS
     DO 985 K=1,ND
     TANV(I,K)=SQHH*SV(K,I)
 985 CONTINUE
     GOTO 997
C
C  TANGENTENRICHTUNG GEGEBEN
 990 AA(J)=0.
     AA(J+1)=1.
     AA(J+2)=0.
     DO 995 K=1,ND
     TANV(I,K)=RBC(K,I)
 995 CONTINUE
 997 CALL IVONZU(V3,1,V2,1,ND)
     SK12=SK23
     SKVS12=SKVS23
     SKVS21=SKVS32
     HIN=HIN1
```

```
      DDH=DDH1
      ZDH=ZDH1
      EDH=EDH1
 1000 CONTINUE
 1050 CONTINUE
C
C  DATEN FUER ENDPUNKT BERECHNEN
      IB=IABS(IBC(N))
      IC=IB/10
      IB=IB-IC*10
      GOTO(570,570,800),IB
  570 XX=-DDH*SKVS21+ZDH1*AB(2,N1)+AB(1,N1)*SK12
      GOTO(600,700),IB
C
C  MIT RANDBEDINGUNG K(N,1)=RBC(1,N)*K(N,0)
  600 AA(NEA)=1.5*AB(1,N1)**5*(AB(2,N1)*EDH-2.*XX)+ESECHS
      AA(NEA+1)=AB(1,N1)**4*(-1.5*AB(2,N1)*((-ZDH-HIN*DDH)
     *        *SKVS21+AB(1,N1)*EDH*SK12)+XX*(-3.*AB(2,N1)*ZDH+4.*XX)
     *        -(DDH**2+(ZDH*AB(2,N1))**2+AB(1,N1)**2-2.*DDH*ZDH
     *        *AB(2,N1)*SKVS21-2.*DDH*AB(1,N1)*SKVS12+2.*ZDH
     *        *AB(1,N1)*AB(2,N1)*SK12))+RBC(1,N)**2*AB(2,N1)**4
     *        *(DDH**2*(1.-SKVS12**2)-2.*DDH*AB(2,N1)*(SKVS21
     *        -SKVS12*SK12)+AB(2,N1)**2*(1.-SK12**2))
      SQHH=-1.5*AB(1,N1)**4*(DDH*(2.*XX-AB(2,N1)*HIN)
     *      -AB(2,N1)*ZDH)+HSECHS
      DO 610 K=1,ND
      TANV(N,K)=SQHH*SV(K,N1)
  610 CONTINUE
      GOTO 900
C
C  MIT RANDBEDINGUNG K'(N-1,1)=0
  700 AA(NEA)=AB(1,N1)*(-2.*XX+EDH*AB(2,N1))
      SQHH=-DDH*(2.*XX-HIN*AB(2,N1))+ZDH*AB(2,N1)
      AA(NEA+1)=-AA(NEA)*SK12+SQHH*SKVS21
      DO 710 K=1,ND
      TANV(N,K)=SQHH*SV(K,N1)
  710 CONTINUE
      GOTO 900
C
C  TANGENTENRICHTUNG GEGEBEN
  800 AA(NEA)=0.
      AA(NEA+1)=1.
      DO 810 K=1,ND
      TANV(N,K)=RBC(K,N)
  810 CONTINUE
  900 CONTINUE
C
C  ROUTINE ZUM LOESEN GLEICHUNGSSYSTEM AUFRUFEN
      CALL GELB(TANV,AA,N,ND,MUD,MLD,EPS,IER)
      IF(IER)1200,1100,1200
 1100 RETURN
 1200 WRITE(IOUT,1250)IER
 1250 FORMAT(1H ,4HIER=,I5,3X,12HFEHLER SIEHE,
     * 35H BESCHREIBUNG DER SUBROUTINE *GELB*)
      RETURN
      END
```

```
      SUBROUTINE IVONZU(X,ICRX,Y,ICRY,N)
C     ================================
C  ZUM AUSHOLEN DATENELEMENTE AUS EINEM FELD X,
C  UND ZUWEISEN DIESE IN EIN ANDERES FELD Y
C   BESCHREIBUNG DER PARAMETER:
C     X   -FELD, AUS DEM ELEMENTE AUSHOLT WERDEN
C     ICRX-INDEXINCREMENT IM X-FELD
C     Y   -FELD, IN DAS ELEMENTE ZUGEWIESEN WERDEN
C     ICRY-INDEXINCREMENT IM Y-FELD
C     N   -ANZAHL DER ZU UEBERTRAGENDEN ELEMENTE
C
      DIMENSION X(1),Y(1)
      INDX=1
      INDY=1
      DO 1000 I=1,N
      Y(INDY)=X(INDX)
      INDX=INDX+ICRX
      INDY=INDY+ICRY
 1000 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE IVMALK(C,X,Y,N)
C  MULTIPLK. DES VEKTORS X MIT SKALAR C, ERGIBT Y
      DIMENSION X(N),Y(N)
      DO 10 I=1,N
      Y(I)=C*X(I)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE IVADD(X,Y,Z,N)
C  VEKTORADDITION X + Y = Z
      DIMENSION X(N),Y(N),Z(N)
      DO 10 I=1,N
      Z(I)=X(I)+Y(I)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INTPL3(X,AB,CD,TANV,ND,N,XI,IORD,NI)
C  ZUR KURVENINTERPOLATION MIT GEGEB. ORDINATEN
C  BESCHREIBUNG DER PARAMETER:
C    EINGANG:
C      X BIS N    -SIEHE SUBROUTINEN 'ISPLNT' UND 'ITANV'
C      NI         -ANZAHL DER ZU INTERPOLIERENDEN PUNKTE
C      IORD       -KENNZAHL DER GEGEB. ORDINATEN
C                  1:X;  2:Y;  3:Z.
C      XI(IORD,J)-ENTHAELT DIE WERTE DER GEGEB. ORDINATEN F. J=1,NI
C    AUSGANG:
C      XI(I,J)    -DIE INTERPOLIERTEN KOORDINATEN F. I=1,ND, J=1,NI
C
      DIMENSION X(ND,N),AB(2,1),CD(2,1),TANV(ND,N)
      DIMENSION XI(ND,NI),CM(12)
C
      N1=N-1
      DO 300 I=1,NI
      DO 100 J=1,N1
      IF(XI(IORD,I).GE.X(IORD,J).AND.XI(IORD,I).LT.X(IORD,J+1))
     *  GOTO 200
  100 CONTINUE
      J=J-1
  200 DO 250 K=1,2
      JK=J+K-1
      JC=1+(K-1)*ND
      CALL IVONZU(X(1,JK),1,CM(JC),1,ND)
      CALL IVONZU(TANV(1,JK),1,CM(JC+2*ND),1,ND)
  250 CONTINUE
      CALL ICURVI(CM,AB(1,J),CD(1,J),XI(1,I),IORD,ND,1)
  300 CONTINUE
      RETURN
      END



      SUBROUTINE ICURVI(CM,AB,CD,XI,IORD,ND,N)
C  ZUR INTERPOLATION MIT EINER GEGEB. ORDINATE
      DIMENSION CM(ND,4),XI(ND,N),HM(16),XH(4),AB(2),CD(2)
C
      NH=ND+1
      CALL IHMATX(CM,AB,CD,ND,NH,HM)
      DO 100 I=1,N
      CALL ICXH(HM,NH,PU(HM,NH,IORD,XI(IORD,I)),XH)
      CALL IHZUK(XH,NH,XI(1,I),ND)
  100 CONTINUE
      RETURN
      END



      SUBROUTINE ICDXH(HM,NH,U,DXH)
C  ZUM BERECHNEN 1. ABLEIT. DER HOMOG. KOORD. NACH U
      DIMENSION HM(NH,4),DXH(NH),DFG(4)
C
      CALL INDFG(U,DFG)
      CALL MAMU(HM,DFG,DXH,NH,4,1)
      RETURN
      END
```

```
      SUBROUTINE IHMATX(CM,AB,CD,ND,NH,HM)
C   BERECHNET AUS KART. KURVENMATRIX CM DIE HOMOGENE HM
C   BESCHREIBUNG DER PARAMETER:
C    EINGANG:
C     CM-KART. KURVENMATRIX EINES KURVENSTUECKS
C        1. SPALT ENTH. ORTSVEKTOR 1. PKT.
C        2. SPALT ENTH. ORTSVEKTOR 2. PKT.
C        3. SPALT ENTH. EINHEITSTANGENTENV. AM 1. PKT.
C        4. SPALT ENTH. EINHEITSTANGENTENV. AM 2. PKT.
C     AB-KURVENPARAMETER
C     CD-KURVENPARAMETER
C     ND-ANZAHL DER DIM.
C     NH-GLEICH ND+1
C    AUSGANG:
C     HM-HOMOGENE KURVENMATRIX EINES KURVENSTUECKS
C
      DIMENSION CM(ND,4),HM(NH,4),AB(2),CD(2)
      DIMENSION SEHN(3),QU0(3),QU1(3)
C
      CALL IVSUBT(CM(1,2),CM(1,1),SEHN,ND)
      RL=SQT(SEHN,ND)
      CALL IVMALK(AB(1)/SQT(CM(1,3),ND),CM(1,3),QU0,ND)
      CALL IVMALK(AB(2)/SQT(CM(1,4),ND),CM(1,4),QU1,ND)
      CALL IVADD(QU0,QU1,SEHN,ND)
      GW=SQT(SEHN,ND)-2.
      HM(NH,3)=CD(1)*GW
      HM(NH,4)=-CD(2)*GW
      DO 100 I=1,ND
      HM(I,1)=CM(I,1)
      HM(I,2)=CM(I,2)
      HM(I,3)=QU0(I)*RL+CM(I,1)*HM(NH,3)
      HM(I,4)=QU1(I)*RL+CM(I,2)*HM(NH,4)
  100 CONTINUE
      HM(NH,1)=1.
      HM(NH,2)=1.
      RETURN
      END

      SUBROUTINE ICXH(HM,NH,U,XH)
C   ZUM BERECHNEN DER HOMOG. KOORD. ZU GEGEB. U
      DIMENSION HM(NH,4),XH(NH),FG(4)
      CALL INFG(U,FG)
      CALL MAMU(HM,FG,XH,NH,4,1)
      RETURN
      END

      SUBROUTINE IRCOS(HM,NH,U,RCOS,ND)
C   ZUM BERECHNEN RICHTUNGSCOSINUS
      DIMENSION HM(NH,4),RCOS(ND),XH(4),DXH(4)
C
      CALL ICXH(HM,NH,U,XH)
      CALL ICDXH(HM,NH,U,DXH)
      DO 100 I=1,ND
      RCOS(I)=(DXH(I)-XH(I)*DXH(NH)/XH(NH))/XH(NH)
  100 CONTINUE
      CALL IEINHV(RCOS,ND)
      RETURN
      END
```

```
      FUNCTION PU(HM,NH,IZ,XYZ)
C  ZUM ERMITTELN PARAMETER U BEI GEGEB.
C  X- BZW. Y- BZW. Z-WERT
C
C   BESCHREIBUNG DER PARAMETER:
C    EINGANG:
C     HM -INTERPOLATIONSMATRIX
C     NH -ANZAHL DER ZEILE VON HM
C     IZ -1, WENN X GEGEB.
C         2, WENN Y GEGEB.
C         3, WENN Z GEGEB. (NUR BEI RAUMKURVE)
C     XYZ-GEGEB. X- BZW. Y- (BZW. Z-)WERT
C    AUSGANG:
C     PU -ZU ERMITTELNDER PARAMETER
C
      DIMENSION HM(NH,4),CST(4,4),XX(16),A(4)
      DATA CST/2.,-2.,2*1.,2*0.,-0.5,0.5,-1.5,1.5,
     *     2*-0.25,2*0.5,0.125,-0.125/,MIT/30/
C
      IF(ABS(XYZ-HM(IZ,1)/HM(NH,1)).GT.1.E-6)GOTO 80
      PU=0.
      RETURN
   80 IF(ABS(XYZ-HM(IZ,2)/HM(NH,2)).GT.1.E-6)GOTO 90
      PU=1.
      RETURN
   90 CALL MAMU(HM,CST,XX,NH,4,4)
      DO 100 J=1,4
  100 A(J)=XX((J-1)*NH+IZ)-XX(J*NH)*XYZ
      U=0.
      IT=0
  110 IT=IT+1
      IF(IT.GT.MIT)GOTO 150
      F=(((A(1)*U+A(2))*U+A(3))*U+A(4))
     *  /((3.*A(1)*U+2.*A(2))*U+A(3)+1.E-6)
      U=U-F
      IF(ABS(F).GT.1.E-6)GOTO 110
      PU=U+0.5
      RETURN
  150 PU=-1.
      RETURN
      END

      SUBROUTINE IVSUBT(X,Y,Z,N)
C  VEKTORSUBTRAKTION X - Y = Z
      DIMENSION X(N),Y(N),Z(N)
      DO 10 I=1,N
      Z(I)=X(I)-Y(I)
   10 CONTINUE
      RETURN
      END

      SUBROUTINE IHZUK(XH,NH,XK,ND)
C  ZUR UMRECHNUNG VON HOMOG. KOORD. ZU KART. KOORD.
      DIMENSION XH(NH),XK(ND)
C
      DO 100 I=1,ND
      XK(I)=XH(I)/XH(NH)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE IDRAW3(X,AB,CD,TANV,ND,N)
C  ZUM ZEICHNEN EINER KURVE, DIE DURCH GEBROCHENE KUBISCHE
C  SPLINES IN PARAMETERFORM DARGESTELLT IST.
C  BESCHREIBUNG DER PARAMETER:
C    SIEHE SUBROUTINEN 'ISPLNT' UND 'ITANV'
C  BESCHREIBUNG DER COMMOMBLOECKE:
C    SIEHE SUBROUTINE 'SPLINE'
C
      DIMENSION X(ND,N),AB(2,1),CD(2,1),TANV(ND,N)
      DIMENSION CM(12),HM(16),FAKTR(4)
      COMMON /PROJ/LPROJ,ISEQ,XS,YS
      COMMON /PERS/LPERS,TRM(4,4),NE
      DATA FAKTR/-1.,-1.,0.,-3./
C
      ISEQ=1
      N1=N-1
      NH=ND+1
      DO 200 I=1,N1
      DO 100 J=1,2
      IJ=I+J-1
      IC=(J-1)*ND+1
      CALL IVONZU(X(1,IJ),1,CM(IC),1,ND)
      CALL IVONZU(TANV(1,IJ),1,CM(IC+2*ND),1,ND)
  100 CONTINUE
      CALL IHMATX(CM,AB(1,I),CD(1,I),ND,NH,HM)
      CALL ICDRAW(HM,NH)
      ISEQ=-ISEQ
  200 CONTINUE
      IF(ND.EQ.2.OR.ISEQ.EQ.1)RETURN
      IR=(-ISEQ+LPERS)*2-1
      CALL PLOT(FAKTR(IR)*XS,FAKTR(IR+1)*YS,-3)
      RETURN
      END
```

```
      SUBROUTINE INDQ(DU,DQ)
C  ZUR BERECHNUNG DER ANFANGSWERTEN ZU DIFFERENZVERF.
      DIMENSION DQ(4,4)
      DQ(1,1)=1.
      DQ(2,1)=0.
      DQ(3,1)=0.
      DQ(4,1)=0.
      DUQ=DU**2
      DUC=DU*DUQ
      DQ(1,2)=2.*DUC-3.*DUQ
      DQ(2,2)=-DQ(1,2)
      DQ(3,2)=DUC-2.*DUQ+DU
      DQ(4,2)=DUC-DUQ
      DQ(1,3)=12.*DUC-6.*DUQ
      DQ(2,3)=-DQ(1,3)
      DQ(3,3)=6.*DUC-4.*DUQ
      DQ(4,3)=6.*DUC-2.*DUQ
      DQ(1,4)=12.*DUC
      DQ(2,4)=-DQ(1,4)
      DQ(3,4)=6.*DUC
      DQ(4,4)=DQ(3,4)
      RETURN
      END
```

```
      SUBROUTINE ICDRAW(HM,NH)
C     ======================
C  ZUM ZEICHNEN EINES KURVENSTUECKS, DAS DURCH
C  HOMOGENE MATRIX DARGESTELLT IST.
C  BESCHREIBUNG DES COMMOMBLOCKS /PERS/:
C    COMMON /PERS/LPERS,TRM(4,4),NE
C    LPERS-STEUERT AUSGABE PERSPEKT. DARSTELLUNG
C     BEI LPERS=1: TRM MUSS GEGEBEN WERDEN,
C               PERSPEKTIV. DARSTELLUNG WIRD DURCHGEF.
C     BEI LPERS=0: TRM WIRD NICHT GEGEBEN,
C               PERSPEKTIV. DARSTELLUNG WIRD NICHT DURCHGEF.
C    TRM  -TRANSFORMATIONSMATRIX
C    NE   -KENNZAHL DER PROJEKTIONSEBENE (BEI LPERS=1)
C          NE=1, XY-EBENE;  2, YZ-EBENE;  3, ZX-EBENE
C  BESCHREIBUNG DER COMMOMBLOECKE /PROJ/ UND /ZMAC/:
C    SIEHE SUBROUTINEN 'IDWPRJ' UND 'INPDU'
C
      DIMENSION HM(NH,4),XHP(2004),HMPER(16),INDP(6)
      DIMENSION NPMAX(2),U(6),HMN(80),NP(5),DU(5)
      COMMON /PERS/LPERS,TRM(4,4),NE
      COMMON /PROJ/LPROJ,ISEQ,XS,YS
      COMMON /ZMAC/ZM,AC
      DATA NPMAX/1000,666/,INDP/1,2,2,3,1,3/,FX,FY/0.5,1.5/
      DATA NSEG/5/,U/0.,0.2,0.4,0.6,0.8,1./
      DATA IOUT/6/
C
      ND=NH-1
      NM=NH*4
C  ANZAHL DER PUNKTE ZUM ZEICHNEN BERECHNEN
      CALL ISEGM(HM,NH,U,NSEG,HMN)
      NPG=0
      DO 100 I=1,NSEG
      CALL INPDU(HMN(NM*(I-1)+1),NH,NP(I),DU(I))
      NPG=NPG+NP(I)
      IF(NPG.GT.NPMAX(NH-2))GOTO 500
  100 CONTINUE
      NG=NPG*ND
      DO 150 I=1,ND
      XHP(NG+I)=0.
      XHP(NG+ND+I)=ZM
  150 CONTINUE
      IF(ND.EQ.3)GOTO 300
C  EBENEKURVE ZEICHNEN
      NPG=0
      DO 200 I=1,NSEG
      CALL INAXHP(HMN(NM*(I-1)+1),NH,NP(I),DU(I),XHP(NPG*ND+1))
      NPG=NPG+NP(I)
  200 CONTINUE
      CALL LINE(XHP(1),XHP(2),NPG,ND,0,0)
      RETURN
C  RAUMKURVE ZEICHNEN
  300 IGOTO=LPERS*2+LPROJ+1
      GOTO(600,400,450,350),IGOTO
C  SOWOHL PROJEKTIONEN ALS AUCH PERSPEKT. DARSTELLUNG
  350 IF(ISEQ.EQ.-1)GOTO 450
C  NUR PROJEKTIONEN
  400 NPG=0
      DO 410 I=1,NSEG
      CALL INAXHP(HMN(NM*(I-1)+1),NH,NP(I),DU(I),XHP(NPG*ND+1))
      NPG=NPG+NP(I)
  410 CONTINUE
      CALL IDWPRJ(XHP,NPG)
      IF(IGOTO.EQ.2)RETURN
```

```
C  NUR PERSPEKT. DARSTELLUNG
  450 NPG=0
      DO 460 I=1,NSEG
      CALL MAMU(HMN(NM*(I-1)+1),TRM,HMPER,NH,NH,NH)
      CALL INAXHP(HMPER,NH,NP(I),DU(I),XHP(NPG*ND+1))
      NPG=NPG+NP(I)
  460 CONTINUE
      I=(NE-1)*2+1
      IND1=INDP(I)
      IND2=INDP(I+1)
      XHP(NG+IND1)=-FX*XS*ZM
      XHP(NG+IND2)=-FY*YS*ZM
      CALL LINE(XHP(IND1),XHP(IND2),NPG,ND,0,0)
      IF(IGOTO.EQ.3)RETURN
      IGOTO=2
      GOTO 400
  500 WRITE(IOUT,510)NPMAX(NH-2)
  510 FORMAT(44H ***DIE ANZAHL DER ZUM ZEICHNEN BENOET. PKT.,
     *        33H IST GROESSER ALS DIE MAX. ANZAHL,I5)
  600 RETURN
      END


      SUBROUTINE ISEGM(HM,NH,U,NS,HMN)
C     ================================
C  ZUR SEGMENTIERUNG EINES KURVENSTUECKS
C
C   BESCHREIBUNG DER PARAMETER:
C    EINGANG:
C     HM  -INTERPOLATIONSMATRIX, DIE EIN KURVENSTUECK DEF.
C     NH  -ANZAHL DER ZEILE DER MATRIX HM
C     U   -VEKTOR, DER NS+1 U-WERTE ENTHAELT, DURCH
C          DIE EIN KURVENSTUECK SEGMENTIERT WIRD
C     NS  -ANZAHL DER SEGMENTE
C    AUSGANG:
C     HMN -NEUE INTERPOLATIONSMATRIZEN, DIE SEGMENTE DEF.
C
      DIMENSION HM(NH,4),HMN(NH,4,NS),U(1)
      DIMENSION B(4,4),A(4,4),AA(4,4),XX(16),XY(16)
      DATA B/15*0.,1./
      DATA A/2.,-2.,2*1.,-3.,3.,-2.,-1.,2*0.,1.,0.,1.,3*0./
      DATA AA/3*0.,1.,4*1.,2*0.,1.,0.,3.,2.,1.,0./
C
      DO 100 K=1,NS
      DU=U(K+1)-U(K)
      B(3,4)=U(K)
      B(2,4)=U(K)**2
      B(1,4)=U(K)**3
      B(3,3)=DU
      B(2,3)=2.*DU*U(K)
      B(1,3)=3.*DU*B(2,4)
      B(2,2)=DU**2
      B(1,2)=3.*B(2,2)*U(K)
      B(1,1)=DU**3
      CALL MAMU(HM,A,XX,NH,4,4)
      CALL MAMU(XX,B,XY,NH,4,4)
      CALL MAMU(XY,AA,HMN(1,1,K),NH,4,4)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE IDWPRJ(XYZ,NP)
C     =========================
C  ZUM ZEICHNEN DREI PROJEKTIONEN EINER RAUMKURVE
C
C  BESCHREIBUNG DER PARAMETER:
C    XYZ  -FELD ENTHAELT KOORDINATEN MIT DIM. 3*(NP+2)
C    NP   -ANZAHL DER PUNKTE
C  BESCHREIBUNG DES COMMOMBLOCKS /PROJ/:
C    COMMON /PROJ/LPROJ,ISEQ,XS,YS
C    LPROJ-STEUERT ZEICHNEN VON PROJEKTIONEN
C     BEI LPROJ=1: PROJEKTIONEN WERDEN GEZEICHNET
C     BEI LPROJ=0: PROJEKTIONEN WERDEN NICHT GEZEICHNET
C    ISEQ -STEUERT REIHENFOLGE DER PROJEKTIONEN
C     BEI ISEQ= 1:XY,XZ,YZ
C     BEI ISEQ=-1:REIHENFOLGE UMGEKEHERT WIE BEI ISEQ=1
C    YS    -ABSTAND ZW. DER X-ACHSE DER XY-PROJEKTION UND DER
C           X-ACHSE DER XZ-PROJEKTION
C    XS    -ABSTAND ZW. DER Z-ACHSE DER XZ-PROJEKTION UND DER
C           Z-ACHSE DER YZ-PROJEKTION
C  DIE ANORDNUNG DER PROJEKTIONEN SIEHT WIE FOLGT AUS:
C       Z             Z
C       I             I
C       I             I
C       I             I
C       I             I
C       I             I
C       ---------X    ---------Y
C
C       Y
C       I
C       I
C       I
C       I
C       I
C       ---------X
C
      DIMENSION XYZ(1),IND(6),FAKT(6)
      COMMON /PROJ/LPROJ,ISEQ,XS,YS
      DATA ND/3/,NPROJ/3/,IND/1,2, 1,3, 2,3/
      DATA FAKT/0.,0., 0.,1., 1.,1./
C
      I1=3-ISEQ*2
      NG=NP*ND
      DO 100 I=1,NPROJ
      I3=(I-1)*ISEQ*2+I1
      IND1=IND(I3)
      IND2=IND(I3+1)
      XYZ(NG+IND1)=-XYZ(NG+ND+IND1)*FAKT(I3)*XS
      XYZ(NG+IND2)=-XYZ(NG+ND+IND2)*FAKT(I3+1)*YS
      CALL LINE(XYZ(IND1),XYZ(IND2),NP,ND,0,0)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INPDU(HM,NH,NP,DU)
C     =============================
C  ZUR BERECHNUNG DER ANZAHL DER ZUM ZEICHNEN BENOETIGEN
C  PUNKTE EINES KURVENSTUECKS MIT GEGEB. GENAUIGKEIT
C  UND GEGEB. MASSSTAB
C
C  BESCHREIBUNG DER PARAMETER:
C    EINGANG:
C      HM  -INTERPOLATIONSMATRIX
C      NH  -ANZAHL DER ZEILE HM
C    AUSGANG:
C      NP  -ZU ERMITTELNDE ANZAHL DER PKT.
C      DU  -INKREMENT DES PARAMETERS U
C  BESCHREIBUNG DES COMMOMBLOCKS:
C    COMMON /ZMAC/ZM,AC
C      ZM  -ZEICHENMASSSTAB
C      AC  -GEFORDERTE GENAUIGKEIT
C
      DIMENSION HM(NH,4)
      COMMON /ZMAC/ZM,AC
      DATA NC/5/
C
      ND=NH-1
      RL=0.
      DO 100 I=1,ND
        RL=(HM(I,1)/HM(NH,1)-HM(I,2)/HM(NH,2))**2+RL
  100 CONTINUE
      RL=SQRT(RL)
      CR=1./(NC-1)
      RKM=0.
      DO 200 I=1,NC
        RKM=AMAX1(RKM,ABS(HKRM(HM,NH,(I-1)*0.25)))
  200 CONTINUE
      RP=AMAX1(1.,RL/SQRT(8.*ZM*AC/AMAX1(RKM,1.E-8)))
      NP=IFIX(RP+2.)
      DU=1./(NP-1)
      RETURN
      END
```

```
      SUBROUTINE INFG(U,FG)
C  ZUR BERECHNUNG HERMITESCHEN POLYNOME 3. GRADES
C
      DIMENSION FG(4)
      FG(2)=-2.*U**3+3.*U**2
      FG(1)=-FG(2)+1.
      FG(4)=U**3-U**2
      FG(3)=FG(4)-U**2+U
      RETURN
      END
```

```
      SUBROUTINE INAXHP(HM,NH,NP,DU,XHP)
C     ==================================
C   ZUM BECHNEN ZUM ZEICHNEN BENOETIGEN PUNKTE
C    HM -INTERPOLATIONSMATRIX
C    NP -ANZAHL DER PUNKTE
C    DU -INTERVALL DES PARAMETERS U
C    NH -ANZAHL DER ZEILE HM
C    XHP-KOORDINATEN
C
      ND=NH-1
      NP1=NP-1
      IND=NP1*ND+1
      CALL IHZUK(HM(1,1),NH,XHP(1),ND)
      CALL IHZUK(HM(1,2),NH,XHP(IND),ND)
      IF(NP.LE.2)GOTO 66666
      CALL INDQ(DU,DQ)
      CALL MAMU(HM,DQ,SK,NH,4,4)
      IND=ND+1
      DO 200 I=2,NP1
      DO 100 J=1,NH
      DO 100 K=1,3
      IZ=(K-1)*NH+J
      SK(IZ)=SK(IZ)+SK(IZ+NH)
  100 CONTINUE
      CALL IHZUK(SK(1),NH,XHP(IND),ND)
      IND=IND+ND
  200 CONTINUE
66666 RETURN
      END
```

```
      SUBROUTINE INDFG(U,DFG)
C   ZUR BERECHNUNG DER ABLEIT. HERMIT. POLYNOME 3. GRADES
C
      DIMENSION DFG(4)
      DFG(1)=6.*U**2-6.*U
      DFG(2)=-DFG(1)
      DFG(4)=3.*U**2-2.*U
      DFG(3)=DFG(4)-2.*U+1.
      RETURN
      END
```

```
      FUNCTION HKRM(HM,NH,U)
C     ======================
C     ZUR BERECHNUNG DER KRUEMMUNG EINES KURVENSTUECKS
C
C     BESCHREIBUNG DER PARAMETER:
C     HM   -INTERPOLATIONSMATRIX EINES KURVENSTUECKS
C     NH   -ANZAHL DER ZEILE HM
C     U    -GEGEBENER PARAMETER (0.LE.U.GE.1.)
C     HKRM-ZU BERECHNENDE KRUEMMUNG ZU U
C
      DIMENSION HM(NH,4),FG(4)
      DIMENSION XH(4),DXH(4),DDXH(4),DX(3),DDX(3)
C
      CALL INFG(U,FG)
      CALL MAMU(HM,FG,XH,NH,4,1)
      CALL INDFG(U,FG)
      CALL MAMU(HM,FG,DXH,NH,4,1)
      CALL INDDFG(U,FG)
      CALL MAMU(HM,FG,DDXH,NH,4,1)
      RN=0.
      ND=NH-1
      DO 10 I=1,ND
      DX(I)=XH(NH)*DXH(I)-XH(I)*DXH(NH)
      DDX(I)=XH(NH)*DDXH(I)-XH(I)*DDXH(NH)
      RN=RN+DX(I)**2
   10 CONTINUE
      IF(NH.EQ.4)GOTO 20
C  ZWISCHENWERT FUER EBENEKURVE BEI NH=3
      RZ=DX(1)*DDX(2)-DX(2)*DDX(1)
        GOTO 30
C  ZWISCHENWERT FUER RAUMKURVE BEI NH=4
   20 RZ=SQRT((DX(1)*DDX(2)-DX(2)*DDX(1))**2
     *        +(DX(2)*DDX(3)-DX(3)*DDX(2))**2
     *        +(DX(3)*DDX(1)-DX(1)*DDX(3))**2)
   30 HKRM=XH(NH)**2*RZ/(RN+1.E-6)**2
      RETURN
      END




      SUBROUTINE INDDFG(U,DDFG)
C  ZUR BERECHNUNG DER 2. ABLEIT. HERMIT. POLYNOME 3. GRADES
C
      DIMENSION DDFG(4)
      DDFG(1)=12.*U-6.
      DDFG(2)=-DDFG(1)
      DDFG(3)=6.*U-4.
      DDFG(4)=DDFG(3)+2.
      RETURN
      END
```

5. Literatur

[1] Söding, H.: Rechnergestützter Schiffsentwurf. Vorlesungsmanuskript. ESS-Bericht Nr. 22. Januar 1977.

[2] Söding, H.: Das Straken von Schiffslinien mit Digitalrechnern. Hansa 1967 Nr. 16, S. 1386-1393.

[3] Kouh, J.-S.: Kurven- und Flächendarstellung in homogenen Koordinaten. 17. Kolloquium des SFB 98, Januar 1983.

[4] Bär, G.: Parametrische Interpolation empirischer Raumkurven. ZAMM 57, 305-311, 1977.

[5] Kestner, W.; Saniter, J.; Strasser, W.; Trambacz, U.: Einführung in Computer Graphics. Informationsforschungsgruppe Computer Graphics, TU Berlin. Eigenverlag Brennpunkt Kybernetik, Berlin, 1973, S. 7-85ff.