Jürgen Schwarz

Behandlung von Polynomen – Teil 1

Stichwörter:

Polynom, HP-BASIC, Laplace-Transformation, Approximation, Ausgleichsrechnung

Hardware:

HP 9845 B mit SP ROM

1 Einleitung

Gemessene Vorgänge werden vor der Weiterverarbeitung im Computer oft durch geeignete Approximationen oder Ausgleichskurven ersetzt. Als Ansatz für die Approximationsfunktion werden oft Polynome [1] oder Exponentialsummen [2] verwendet. Sollen die Exponentialsummen weiter verarbeitet werden, so ist es zweckmäßig, diese in den Bildbereich der Laplace-Transformation zu transformieren. Dabei erhält man gebrochenrationale Funktionen, also ebenfalls Polynome.

Hier sollen jetzt einige SUB-Programme zur Behandlung von Polynomen abgeleitet und beschrieben werden. Dazu ist eine Vereinbarung über die Darstellung der Polynome im Rechner erforderlich.

2 Darstellung von Polynomen

Ein Polynom n-ten Grades mit reellen Koeffizienten

$$f(x) = a_n x^n + a_{n-1} \cdot x^{n-1} + \dots + a_2 x^2 + a_1 x + a_0;\ a_n \neq 0;\ a_i, x \in \mathbb{R} \qquad (1)$$

wird am einfachsten durch ein eindimensionales Feld A (0:n) repräsentiert, A(0) speichert den Wert von a_0, A(1) den Wert von a_1 usw.

Polynome mit komplexen Koeffizienten

$$f(z) = (a_n + i\, b_n) z^n + (a_{n-1} + i\, b_{n-1}) z^{n-1} + \dots + a_0 + i\, b_0;$$
$$a_n + i\, b_n \neq 0;\ a_i, b_i \in \mathbb{R};\ z \in \mathbb{C} \qquad (2)$$

könnten durch ein zweidimensionales Feld repräsentiert werden. Als zweckmäßiger hat sich aber die Vereinbarung von zwei Feldern A(0 : n) und B(0 : n) erwiesen.

3 Addition und Subtraktion

Die einfachsten Rechenarten sind die Addition und die Subtraktion. Das Listing 1 zeigt ein SUB-Programm zur Ausführung dieser Rechenarten. Gegeben sind zwei Polynome mit reellen Koeffizienten n-ten Grades f(x) und m-ten Grades g(x) und berechnet werden soll die bewertete Summe dieser Polynome

$$h(x) = k_f \cdot f(x) + k_g \cdot g(x) \qquad (3)$$
$$= k_f \cdot (a_n x^n + a_{n-1} x^{n-1} + \dots + a_0) + k_g \cdot (b_m x^m + b_{m-1} x^{m-1} + \dots + b_0).$$

Für die einzelnen Elemente von h(x) gilt dann

$$c_i = k_f \cdot a_i + k_g \cdot b_i, \qquad (4)$$

wenn die c_i die Koeffizienten von h(x) und die a_i bzw. b_i die Koeffizienten von f(x) bzw. g(x) sind. Der Grad des Polynoms h(x) ist im allgemeinen* das Maximum von n und m.

Nun noch ein paar Bemerkungen zu dem Programm. In Zeile 180 werden zwei Felder zur Zwischenspeicherung von Ergebnissen vereinbart. Diese müssen so groß wie das größte verwendete Polynom sein. Durch die Verwendung des Befehls ROW (Feld), welcher die aktuelle Anzahl des Elementes des Feldes bereitstellt, wird dies gewährleistet.

Durch entsprechende Wahl von k_f und k_g kann das Programm sowohl zur Addition als auch zur Subtraktion verwendet werden. Für Polynome mit komplexen Koeffizienten kann das Programm auch Anwendung finden, in dem es zweimal, zuerst für die Realteile und dann für die Imaginärteile, angewandt wird.

4 Multiplikation und Division

Listing 2 zeigt ein SUB-Programm zur Multiplikation zweier Polynome mit reellen Koeffizienten nach der Gleichung

$$h(x) = f(x) \cdot g(x) \qquad (5)$$
$$= a_n b_m x^{n+m} + (a_{n-1} \cdot b_m + a_n \cdot b_{m-1}) x^{n+m-1} + \dots + a_0 b_0.$$

Die Koeffizienten c_i des Polynoms h(x) ergeben sich hier aus der Gleichung

$$c_i = a_0 \cdot b_i + a_1 \cdot b_{i-1} + a_2 \cdot b_{i-2} + \dots + a_i \cdot b_0, \qquad (6)$$

sofern alle Elemente existieren. Diese Einschränkung wird durch die Laufanweisung auf Zeile 330 realisiert.

In diesem SUB-Programm sind die Polynomgrade n und m in den Kopf aufgenommen worden. Will man sich aber die Mühe der Bereitstellung dieser Zahlen sparen (dies kann bei längeren Rechnungen sehr zweckmäßig sein), so erfolgt die Berechnung dieser Werte bei einem Aufruf des SUB-Programms mit $m = n = 0$ automatisch. Diesen Teil des Programmes hätte man aber auch genauso wie beim Additionsprogramm gestalten können. Auch bei diesem Programm erfolgt ein Schutz der Ausgangsdaten, der einen Aufruf der Art CALL Produkt (0, 0, A(X*), B(*), A(*)) ermöglicht.

Jürgen Schwarz, Wiesbadener Str. 59 C,
1000 Berlin 33

* Nur für $n = m$ und $k_f \cdot a_n + k_g \cdot b_n = 0$ gilt dies nicht.

```
10    SUB Addition(REAL K_f,A(*),K_g,B(*),C(*))
20    !
30    ! SUB-Programm zur bewerteten Addition und Subtraktion von zwei Polynomen
40    ! mit reellen Koeffizienten:  h(x) = k_f * f(x) + k_g * g(x)
50    !
60    ! Eingabedaten: k_f             ... Bewertungsfaktor für das Polynom f(x)
70    !               a(0:n)          ... Koeffizienten des Polynoms f(x)
80    !               k_g             ... Bewertungsfaktor für das Polynom g(x)
90    !               b(0:m)          ... Koeffizienten des Polynoms g(x)
100   ! Ergebnis:     c(0:max(n,m)) ... Koeffizienten des Polynoms h(x)
110   !
120   !               Programmierer:   Jürgen Schwarz
130   !               Programm-Name:           Additn
140   !               Datum/Variante:  05.11.84 / 03
150   !               Speichermedium: Kassetten 57/58
160   !
170     INTEGER N,M,P
180     REAL A_stern(0:MAX(ROW(A),ROW(B))-1),B_stern(0:MAX(ROW(A),ROW(B))-1)
190     !
200     N=ROW(A)
210     REPEAT
220       N=N-1
230     UNTIL (A(N)<>0) OR (N=0)              ! Grad von f(x)
240     M=ROW(B)
250     REPEAT
260       M=M-1
270     UNTIL (B(M)<>0) OR (M=0)              ! Grad von g(x)
280     P=MAX(N,M)                            ! Grad von h(x)
290     !
300     MAT A_stern=(K_f)*A
310     MAT B_stern=(K_g)*B
320     !
330     MAT C=ZER
340     REDIM A_stern(0:P),B_stern(0:P),C(0:P)
350     MAT C=A_stern+B_stern
360     !
370     WHILE (C(P)=0) AND (P>0)
380       P=P-1
390     END WHILE                             ! Grad von h(x)
400     REDIM C(0:P)
410     !
420   SUBEND
```

Listing 1: Addition und Subtraktion von zwei Polynomen mit reellen Koeffizienten.

Dieser Befehl wird auch in den Zeilen 200 und 240 zur Bereitstellung der Polynomgrade verwendet. In den dann folgenden Zeilen (210 – 230 und 250 – 270) wird die Bedingung $a_n \neq 0$ geprüft und falls sie nicht erfüllt ist, wird der Grad der Polynome entsprechend gesenkt. Die Befehle in den Zeilen 300 und 310 bewirkten die Multiplikation jedes Elementes der Felder mit dem davor in Klammer gesetzten Faktor. Der Befehl MAT Feld = ZER setzt alle gerade benutzten Elemente des Feldes zu null, d. h. die Elemente, die der aktuellen Größe des Feldes entsprechen. Diese Größe kann mit dem Befehl REDIM (Zeile 340) verändert werden. In Zeile 350 werden die korrespondierenden Feldelemente von A_stern und B_stern addiert. Die folgenden Zeilen dienen zur Prüfung des Grades des Ergebnispolynoms.

Durch die Zwischenspeicherung der Ergebnisse wird ein Schutz der Ausgangsdaten erreicht. Dadurch ist auch ein Aufruf der Art CALL Addition (1, A(*), 1, B(*), A(*)) zulässig, wobei in A(*) nach Abarbeitung die Summe von dem ursprünglichen A(*) und B(*) steht.

Enthält eine Computersprache zu den Befehlen ROW (...) und REDIM äquivalente Befehle nicht, so müssen die Gradzahlen der Polynome ähnlich wie in Listing 4 in den SUB-Programm-Kopf aufgenommen werden.

```
10    SUB Produkt(INTEGER N,M,REAL A(*),B(*),C(*))
20    !
30    ! SUB-Programm zur Durchführung einer Multiplikation von zwei Polynomen
40    ! mit reellen Koeffizienten:  h(x) = f(x) * g(x)
50    !
60    ! Eingabedaten: n          ... Grad              des Polynoms f(x)
70    !               m          ... Grad              des Polynoms g(x)
80    !               a(0:n)     ... Koeffizienten des Polynoms f(x)
90    !               b(0:m)     ... Koeffizienten des Polynoms g(x)
100   ! Ergebnis:     n+m        ... Grad              des Polynoms h(x)
110   !               c(0:n+m) ... Koeffizienten des Polynoms h(x)
120   !
130   !               Programmierer:    Jürgen Schwarz
140   !               Programm-Name:            Produk
150   !               Datum/Variante:    05.11.84 / 04
160   !               Speichermedium: Kassetten 57/58
170   !
180     INTEGER I,J
190     REAL A_stern(0:ROW(A)-1),B_stern(0:ROW(B)-1)
200     !
210     IF (N=0) AND (M=0) THEN
220       N=ROW(A)-1
230       M=ROW(B)-1
240     END IF
250     !
260     REDIM A(0:N),B(0:M)
270     MAT A_stern=A
280     MAT B_stern=B
290     MAT C=ZER
300     REDIM C(0:N+M)
310     !
320     FOR I=0 TO N+M
330       FOR J=MAX(0,I-M) TO MIN(I,N)
340         C(I)=C(I)+A_stern(J)*B_stern(I-J)
350       NEXT J
360     NEXT I
370     !
380   SUBEND
```

Listing 2: Multiplikation von zwei Polynomen mit reellen Koeffizienten.

Ein SUB-Programm zur Multiplikation zweier Polynome mit komplexen Koeffizienten zeigt das Listing 3. Die Berechnung der Grade der Polynome erfolgt analog dem Additionsprogramm und auch hier erfolgt ein Schutz der Ausgangsdaten. Für die Koeffizienten $e_i + if_i$ des Polynoms

$$\begin{aligned} h(z) &= f(z) \cdot g(z) \\ &= [(a_n + ib_n) z^n + (a_{n-1} + ib_{n-1}) z^{n-1} + \ldots + a_0 + ib_0] \cdot [(c_m + id_m) z^m + (c_{m-1} + id_{m-1}) z^{m-1} + \ldots + c_0 + id_0] \\ &= [(a_n \cdot c_m - b_n \cdot d_m) + i(b_n \cdot c_m + a_n \cdot d_m)] \cdot z^{n+m} + [(a_n \cdot c_{m-1} - b_n \cdot d_{m-1} + a_{n-1} \cdot c_m - b_{n-1} \cdot d_m) + i(b_n \cdot c_{m-1} + a_n \cdot c_{m-1} + b_{n-1} \cdot c_m + a_{n-1} \cdot d_m)] \cdot z^{n+m-1} + \ldots + [(a_0 \cdot c_0 - b_0 \cdot d_0) + i(b_0 \cdot c_0 + a_0 \cdot d_0)] \end{aligned} \tag{7}$$

gilt hier im einzelnen

$$\begin{aligned} e_i &= a_0 \cdot c_i - b_0 \cdot d_i + a_1 \cdot c_{i-1} - b_1 \cdot d_{i-1} + \ldots + a_i \cdot c_0 - b_i \cdot d_0 \\ f_i &= b_0 \cdot c_i + a_0 \cdot d_i + b_1 \cdot c_{i-1} + a_1 \cdot d_{i-1} + \ldots + b_i \cdot c_0 + a_i \cdot d_0, \end{aligned} \tag{8}$$

wenn die a_k und c_k die Realteile und b_k und d_k die Imaginärteile der Koeffizienten der Polynome f(z) und g(z) sind.

Zur Durchführung der Division von zwei Polynomen mit reellen Koeffizienten dient das SUB-Programm Division auf Listing 4. Hier wurde der bekannte formale Vorgang der Division von zwei Polynomen

$$(a_n \cdot x^n + a_{n-1} \cdot x^{n-1} + a_{n-2} \cdot x^{n-2} + \ldots) : (b_m x^m + b_{m-1} \cdot x^{m-1} + \ldots) =$$

$$\frac{-\ (a_n x^n + b_{m-1} \cdot \frac{a_n}{b_m} x^{n-1} + b_{m-2} \cdot \frac{a_n}{b_m} x^{n-2} + \ldots)}{0 \cdot x^n + (a_{n-1} - b_{m-1} \cdot \frac{a_n}{b_m}) x^{n-1} + (a_{n-2} - b_{m-2} \cdot \frac{a_n}{b_m}) \cdot x^{n-2} + \ldots}$$

$$= \frac{a_n}{b_m} x^{n-m} + \left(\frac{a_{n-1}}{b_m} - \frac{b_{m-1} \cdot a_n}{{b_m}^2}\right) \cdot x^{n-m-1} + \ldots \tag{9}$$

automatisiert.

```
10    SUB Komplex_produkt(REAL A_r(*),A_i(*),B_r(*),B_i(*),C_r(*),C_i(*))
20    !
30    ! SUB-Programm zur Durchführung einer Multiplikation von zwei Polynomen
40    ! mit komplexen Koeffizienten:  h(z) = f(z) * g(z)
50    !
60    ! Eingabedaten:
70    !          a_r(0:n)   ... Realteil      der Koeffizienten des Polynoms f(z)
80    !          a_i(0:n)   ... Imaginärteil der Koeffizienten des Polynoms f(z)
90    !          b_r(0:m)   ... Realteil      der Koeffizienten des Polynoms g(z)
100   !          b_i(0:m)   ... Imaginärteil der Koeffizienten des Polynoms g(z)
110   ! Ergebnis:
120   !          c_r(0:n+m) ... Realteil      der Koeffizienten des Polynoms h(z)
130   !          c_i(0:n+m) ... Imaginärteil der Koeffizienten des Polynoms h(z)
140   !
150   !          Programmierer:   Jürgen Schwarz
160   !          Programm-Name:           kmpl_p
170   !          Datum/Variante:   05.11.84 / 02
180   !          Speichermedium: Kassetten 57/58
190   !
200     INTEGER I,J,K,N,M
210     REAL A_r_st(0:MAX(ROW(A_r),ROW(A_i))-1)
220     REAL A_i_st(0:MAX(ROW(A_r),ROW(A_i))-1)
230     REAL B_r_st(0:MAX(ROW(B_r),ROW(B_i))-1)
240     REAL B_i_st(0:MAX(ROW(B_r),ROW(B_i))-1)
250     !
260     N=MAX(ROW(A_r),ROW(A_i))
270     M=MAX(ROW(B_r),ROW(B_i))
280     REDIM A_r(0:N-1),A_i(0:N-1),B_r(0:M-1),B_i(0:M-1)
290     REPEAT
300       N=N-1
310     UNTIL (A_r(N)<>0) OR (A_i(N)<>0) OR (N=0)    ! Grad von f(z)
320     REPEAT
330       M=M-1
340     UNTIL (B_r(M)<>0) OR (B_i(M)<>0) OR (M=0)    ! Grad von g(z)
350     REDIM A_r(0:N),A_i(0:N),B_r(0:M),B_i(0:M)
360     !
370     MAT A_r_st=A_r
380     MAT A_i_st=A_i
390     MAT B_r_st=B_r
400     MAT B_i_st=B_i
410     !
420     MAT C_r=ZER
430     MAT C_i=ZER
440     REDIM C_r(0:N+M),C_i(0:N+M)
450     FOR I=0 TO N+M
460       FOR J=MAX(0,I-M) TO MIN(I,N)
470         K=I-J
480         C_r(I)=C_r(I)+A_r_st(J)*B_r_st(K)-A_i_st(J)*B_i_st(K)
490         C_i(I)=C_i(I)+A_i_st(J)*B_r_st(K)+A_r_st(J)*B_i_st(K)
500       NEXT J
510     NEXT I
520     !
530   SUBEND
```

Listing 3: Multiplikation von zwei Polynomen mit komplexen Koeffizienten.

Am Ende der Division wird geprüft, ob der sich ergebende Rest wirklich null ist. Ist dies nicht der Fall, dann erfolgt eine Fehlermeldung.

Dieses Programm arbeitet ohne den ROW-Befehl und macht diese auch auf die anderen Programme übertragbare Variante deutlich. Ein Programm zur Division von Polynomen mit komplexen Koeffizienten wird hier nicht explizit angegeben, ist aber für $m = 1$ Bestandteil des Nullstellenbestimmungsprogramms Siljak, das in Teil 2 dieser Veröffentlichung vorgestellt wird.

Literatur

[1] M. PAUL: Digitale Meßwertverarbeitung. VDE-Verlag Berlin, Offenbach (1982).

[2] G.H. GOLUB, V. PEREYRA: „The differentiation of pseudoinverses and nonlinear least squares problems where variables separate." SIAM J. Numer. Anal. **10** (1973) 413 – 432.

CAL

```
10    SUB Division(INTEGER N,M,REAL A(*),B(*),C(*))
20    !
30    ! SUB-Programm zur Durchführung einer Division von zwei Polynomen
40    ! mit reellen Koeffizienten:  h(x) = f(x) / g(x)
50    !
60    ! Eingabedaten: n         ... Grad         des Polynoms f(x)
70    !               m         ... Grad         des Polynoms g(x)
80    !               a(0:n)    ... Koeffizienten des Polynoms f(x)
90    !               b(0:m)    ... Koeffizienten des Polynoms g(x)
100   ! Ergebnis:     n-m       ... Grad         des Polynoms h(x)
110   !               c(0:n-m)  ... Koeffizienten des Polynoms h(x)
120   !
130   !               Programmierer:   Jürgen Schwarz
140   !               Programm-Name:           Divvek
150   !               Datum/Variante:   05.11.84 / 02
160   !               Speichermedium: Kassetten 57/58
170   !
180     INTEGER I,J,Fehler
190     REAL A,A_stern(0:N),B_stern(0:M)
200 Z:  IMAGE "A_stern(",DD,") =",SD.DDDXDDDE,20X,"A =",SD.DDDXDDDE
210     !
220     IF (N<M) OR (M<0) THEN PAUSE   ! widersprüchliche Eingabedaten
230     !
240     REDIM A(0:N),B(0:M)
250     MAT A_stern=A
260     MAT B_stern=B
270     MAT C=ZER
280     REDIM C(0:N-M)
290     !
300     Fehler=0
310     FOR I=N-M TO 0 STEP -1
320       C(I)=A_stern(I+M)/B_stern(M)
330       FOR J=I+M TO I STEP -1
340         A=A_stern(J)
350         A_stern(J)=A-C(I)*B_stern(J-I)
360         IF I=0 THEN
370           IF Fehler=1 THEN
380             GOSUB Fehler
390           ELSE
400             IF ABS(A)>0 THEN
410               IF ABS(A_stern(J)/A)>1E-11 THEN GOSUB Fehler
420             ELSE
430               IF ABS(A_stern(J))>1E-11*ABS(SUM(C)) THEN GOSUB Fehler
440             END IF
450           END IF
460         END IF
470       NEXT J
480     NEXT I
490     SUBEXIT
500     !
510 Fehler:   ! Fehler bei der Division
520     !     ! Der Rest ist nicht null bzw. die Rundungsfehler sind zu groß
530     IF Fehler=0 THEN PRINT LIN(2)
540     Fehler=1
550     PRINT USING Z;J,A_stern(J),A
560     IF J>I THEN RETURN
570     !
580     PRINT LIN(1),"Fehler im SUB-Programm Division!"
590     PRINT "Division geht nicht ohne Rest auf!",LIN(2)
600     PRINT "A(*) =";LIN(2),A(*)
610     PRINT "B(*) =";LIN(2),B(*)
620     PRINT "C(*) = [ A(*) / B(*) = ]";LIN(2),C(*)
630     PRINT "Weiter mit CONT!"
640     PAUSE
650     SUBEXIT
660     !
670   SUBEND
```

Listing 4: Division von zwei Polynomen mit reellen Koeffizienten.

Abstract

Jürgen Schwarz

45 Behandlung von Polynomen Teil 1

Gemessene Vorgänge werden vor der Weiterverarbeitung im Computer oft durch geeignete Approximationen ersetzt. Als Ansatz für die Approximationsfunktion dienen oft Polynome oder Exponentialsummen, wobei letztere zweckmäßigerweise in den Bildbereich der Laplace-Transformation transformiert werden. Dabei entstehen ebenfalls Polynome. Dieser Artikel enthält einige Subprogramme zur Behandlung von Polynomen.