

Kernforschungszentrum Karlsruhe

Oktober 1971

Externer Bericht

8/71-6

Institut für Reaktorentwicklung

Fortran Programme zur analytischen Beschreibung der
Anlaufströmung in eindimensionalen Geometrien

G.Enderle
U.Schumann

Gesellschaft für Kernforschung m.b.H., Karlsruhe

Als Manuskript vervielfältigt

Für diesen Bericht behalten wir uns alle Rechte vor

**GESELLSCHAFT FÜR KERNFORSCHUNG M. B. H.
KARLSRUHE**

Kernforschungszentrum Karlsruhe

Oktober 1971

Externer Bericht

8/71-6

Institut für Reaktorentwicklung

Fortran Programme zur analytischen Beschreibung der
Anlaufströmung in eindimensionalen Geometrien

G.Enderle
U.Schumann

Gesellschaft für Kernforschung m.b.H., Karlsruhe



Zusammenfassung

Zur Kontrolle der Genauigkeit von numerischen Verfahren zur Lösung instationärer Probleme der Hydrodynamik sowie des Massen- und Energieaustausches ist es wünschenswert, exakte Vergleichslösungen verfügbar zu haben. Es werden hier zu diesem Zweck die Lösungen des Anlaufproblems

$$\frac{\partial v}{\partial t} = \frac{1}{Re} \nabla^2 v + p; v(t=0, r) = v_0; v(t, \text{Rand}) = 0$$

in Platten-, Zylinder- und Hohlzylindergeometrie bereitgestellt. Dieser Bericht enthält die Theorie der analytischen Methode sowie die Beschreibung und Listen der FORTRAN-IV-Unterprogramme ANLAUF, ANPRIN und ANPLOT zum Berechnen, Ausdrucken und Zeichnen der Lösungen. Die Zeichnungen werden mit der CALCOMP-Software erzeugt.

Abstract

In computing non stationary problems of hydrodynamics or mass and heat exchange it is desirable to have the exact solutions of some particular problems available against which numerical procedures can be checked for accuracy.

For this purpose, the solutions of the transient problems

$$\frac{\partial v}{\partial t} = \frac{1}{Re} \nabla^2 v + p; v(t=0, r) = v_0; v(t, \text{boundary}) = 0$$

for slab, cylinder and hollow cylinder geometries are derived. This report contains the theory of the analytical method and the description and listings of the FORTRAN IV subroutines ANLAUF, ANPRIN and ANPLOT which calculate, print and plot the solutions. The plots are generated by using CALCOMP software.



Inhaltsverzeichnis

Zusammenfassung

I. Theorie

1. Einleitung
2. Bestimmung von V_{∞} für Ringspaltgeometrie
3. Bestimmung von V_a für Ringspaltgeometrie
4. Lösungen für Platte und Zylinder
5. Bemerkungen zum Unterschied zwischen Anlauf- und Einlaufströmungen
6. Äquivalenz des Quellterms p mit einem konstant beschleunigten Rand

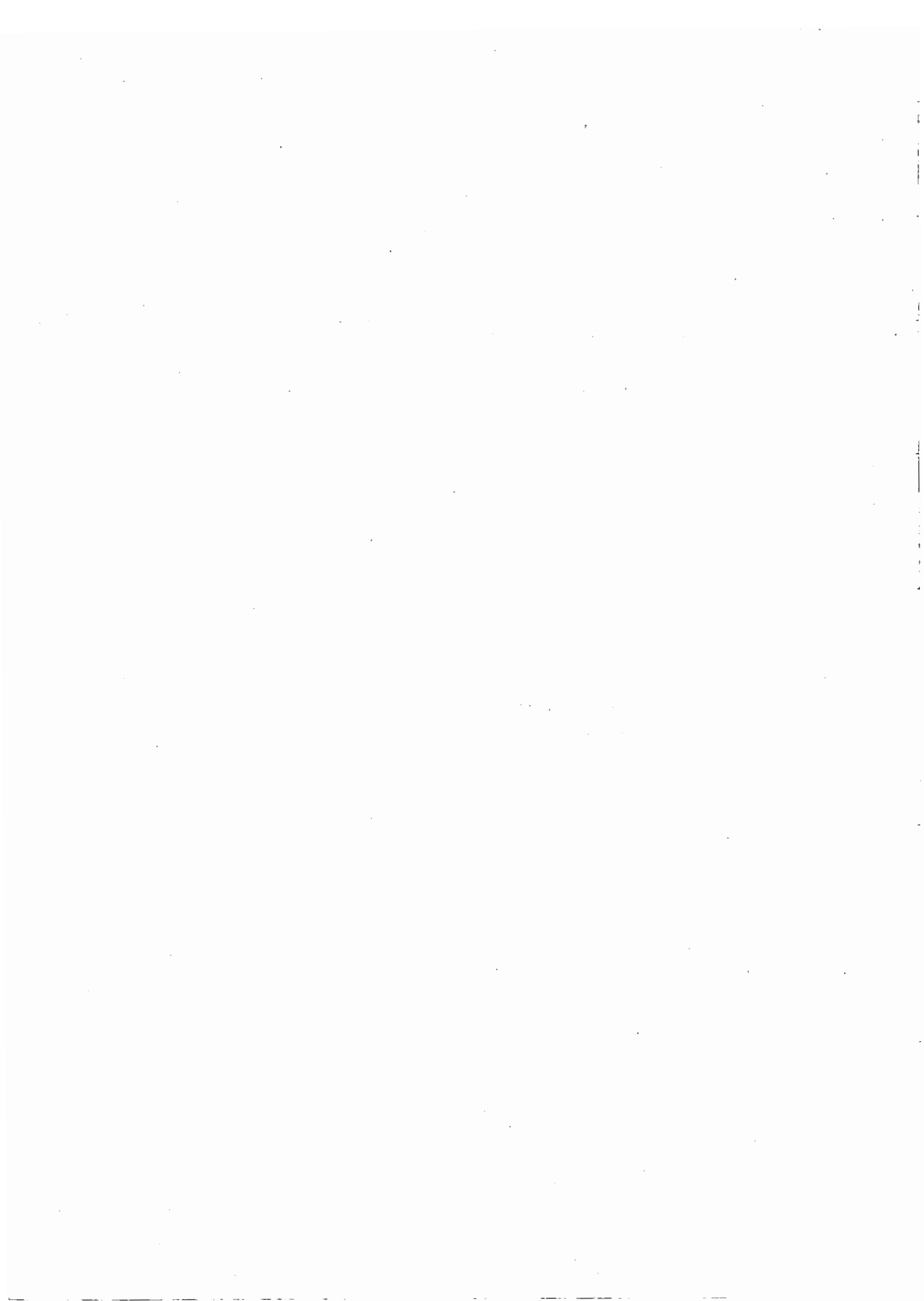
II. ANLAUF, ANPRIN, ANPLØT

Fortran-Programme zum Berechnen, Ausdrucken und Zeichnen der Lösungen des Anlaufproblems

1. Zweck
2. Anwendung
3. Beispiele
4. Rechengenauigkeit
5. Speicherbelegung, Rechenzeit
6. Zur Programmlogik

Literatur

Abbildungen



I. Theorie

1. Einleitung

Bei zahlreichen Problemen des Impuls, Massen- und Energieaustausches treten Differentialgleichungen der Form

$$\frac{\partial v}{\partial t} = \frac{1}{Re} \nabla^2 v + p, \quad v = v(t, r) \quad (1)$$

auf. Die verwendeten Symbole sind hier der Strömungsmechanik entnommen. v entspricht einer dimensionslosen Geschwindigkeit, p einem dimensionslosen Druckgradienten in Strömungsrichtung, t ist die dimensionslose Zeit, ∇^2 ist der dimensionslose Laplace-Operator und Re eine Konstante (Reynold-Zahl). Formal tritt die selbe Gleichung in allen drei genannten Problembereichen auf. Aufgrund komplizierter Rand- und Anfangsbedingungen sowie variablem p sind für die Lösung dieser Differentialgleichung komplizierte numerische Verfahren erforderlich. Zur Kontrolle dieser Verfahren ist es wünschenswert, analytische Vergleichslösungen verfügbar zu haben. Für das folgende vereinfachende Problem sind analytische Lösungsmethoden bekannt:

- eindimensionale Geometrie (siehe unten)
 - $p = \text{const}$
 - Anfangsbedingung $v(t = 0, r) = 0$
 - Randbedingung $v(t, \text{Rand}) = 0$
(Dirichletsche Randbedingung)
- (2)

Es werden hier die analytischen Lösungen bereitgestellt, nach denen die im II. Abschnitt beschriebenen Rechenprogramme die Lösungen $v(t, r)$ berechnen.

Dazu werden die drei folgenden speziellen eindimensionalen Geometrien betrachtet (vergl. Abb. 1):

- 1) unendlich ausgedehnte Platte
Rand: $r = \pm R$
- 2) unendlich langer Zylinder
Rand: $r = \pm R$
- 3) unendlich langer Hohlzylinder
Rand: $r = R_0$ und $r = R_1$ $0 < R_0 < R_1$

Hierfür ist die Differentialgleichung (1) mit (2) lösbar mit folgendem Ansatz (Separation zulässig, da (1) und (2) linear):

$$v(t, r) = v_{\infty}(r) + v_a(t, r) \quad (3)$$

$$\frac{\partial v_a}{\partial t} = \frac{1}{Re} \nabla^2 v_a \quad (4)$$

$$\sigma = \frac{1}{Re} \nabla^2 v_{\infty} + p \quad (5)$$

$$v_{\infty}(\text{Rand}) = \sigma \quad (6)$$

$$v_a(\text{Rand}) = 0 \quad (7)$$

$$v_a(t=0, r) = -v_{\infty}(r) \quad (8)$$

Da die Lösungen der Gleichung (3) und (4) für die Geometrien 1) und 2) an verschiedenen Stellen (z.B. [1, 2, 3]) angegeben sind, wird hier im Detail nur auf die Geometrie 3) eingegangen. Das mathematische Vorgehen ist in den anderen Fällen gleichartig, es werden hierfür lediglich die Ergebnisse angegeben. Für die Geometrie 3) lauten die Gleichungen (4) und (5) in Zylinderkoordinaten:

$$\frac{\partial v_a}{\partial t} = \frac{1}{Re} \left[\frac{\partial^2 v_a}{\partial r^2} + \frac{1}{r} \frac{\partial v_a}{\partial r} \right] \quad (4')$$

$$\sigma = \frac{1}{Re} \left[\frac{dv_{\infty}}{dr^2} + \frac{1}{r} \frac{dv_{\infty}}{dr} \right] + p \quad (5')$$

2. Bestimmung von $v_{\infty}(r)$ für Ringspaltgeometrie [1]

Mit der Substitution $u = \frac{dv_{\infty}}{dr}$ folgt aus (5')

$$r \, du + u \, dr + p \cdot Re \cdot r \, dr = 0$$

$$d(u \cdot r) = - p \cdot Re \cdot r \, dr$$

$$u r = - p Re \frac{r^2}{2} + \beta$$

$$\frac{dv_{\infty}}{dr} = - \frac{p Re}{2} r + \frac{\beta}{r}$$

$$v_{\infty} = - \frac{p \cdot Re}{4} r^2 + \beta \ln r + \gamma$$

aus (6) folgt

$$0 = - \frac{p \cdot Re}{4} R_0^2 + \beta \ln R_0 + \gamma$$

$$0 = - \frac{p \cdot Re}{4} R_1^2 + \beta \ln R_1 + \gamma$$

↘

$$\beta = \frac{p \cdot Re}{4} (R_0^2 - R_1^2) / \ln(R_0/R_1)$$

$$\gamma = \frac{p \cdot Re}{4} (R_1^2 \ln R_0 - R_0^2 \ln R_1) / \ln(R_0/R_1)$$

mit $\alpha = - \frac{p Re}{4}$ lautet die Lösung

$$v_{\infty}(r) = \alpha r^2 + \beta \ln r + \gamma \quad (9)$$

3. Bestimmung von $v_a(r, t)$

Zur Lösung der Gleichung (4') gelangt man mit dem Ansatz:

$$v_a(t, r) = D e^{-q^2 t / Re} \cdot \psi(r) \quad (10)$$

Einsetzen dieses Ansatz führt auf die Besselsche Differentialgleichung der Ordnung Null für $\psi(r)$:

$$\frac{d^2 \psi}{dr^2} + \frac{1}{r} \frac{d\psi}{dr} + q^2 \psi = 0 \quad (11)$$

Deren allgemeine Lösung für $q^2 > 0$ ($q^2 < 0$ scheidet aus [4]) wird mit den Besselfunktionen erster und zweiter Art der Ordnung Null beschrieben:

$$\psi(r) = A J_0(qr) + B Y_0(qr)$$

Aus den Bedingungen am Rand (7) folgt:

$$0 = A J_0(qR_0) + B Y_0(qR_0) \quad (12)$$

$$0 = A J_0(qR_1) + B Y_0(qR_1)$$

Dies führt auf eine Bestimmungsgleichung für q :

$$J_0(qR_0) \cdot Y_0(qR_1) - Y_0(qR_0) \cdot J_0(qR_1) = 0 \quad (13)$$

Diese Gleichung liefert unendlichviele Lösungen

$$q_k, \quad k = 1, 2, 3,$$

Außerdem wird durch die Gleichung (12) eine der beiden Konstanten A, B bestimmt:

$$\begin{aligned} \psi(r) &= A_k J_0(q_k r) - \frac{J_0(q_k R_1)}{Y_0(q_k R_1)} \cdot Y_0(q_k r) \\ &= (A_k / Y_0(q_k R_1)) \cdot U_0(q_k r) \end{aligned} \quad (14)$$

$$U_0(q_k r) \equiv J_0(q_k r) \cdot Y_0(q_k R_1) - Y_0(q_k r) \cdot J_0(q_k R_1)$$

Die Konstante D aus (10) und $(A_k / Y_0(q_k R_1))$ lassen sich zur Konstante C_k zusammenfassen, sodaß die allgemeine Lösung von (4') lautet:

$$v_a(t, r) = \sum_{k=1}^{\infty} e^{-q_k^2 \cdot t / Re} \cdot C_k U_0(q_k r) \quad (15)$$

Die Konstanten C_k folgen aus der Anfangsbedingung (8) unter Ausnutzung der Orthogonalität der Funktionen $\sqrt{r} \cdot U_0(q_k r)$ im Intervall $R_0 \leq r \leq R_1$. Hierfür sind zunächst einige Hilfsbeziehungen erforderlich, die man zum Teil in [2] findet:

3.1 Integrale

$$I_1 = \int_{R_0}^{R_1} r U_0(q_j r) \cdot U_0(q_k r) dr$$

U_0 ist Lösung der Besselgleichung (11). Aus ihr folgt:

$$U_0(qr) = -\frac{1}{q^2} \frac{1}{r} \frac{d}{dr} \left(r \frac{d}{dr} U_0(qr) \right) \quad (16)$$

Mit dieser Angabe erhält man nach partieller Integration und mit Gleichung (13)

$$\int_{R_0}^{R_1} r U_0(q_j r) \cdot U_0(q_k r) dr = 0, \quad j \neq k$$

und

$$\int_{R_0}^{R_1} r U_0(q_j r) \cdot U_0(q_k r) dr = \frac{1}{2q_k^2} \left[\left(r \frac{d}{dr} U_0(q_k r) \right) \right]_{R_0}^{R_1}, \quad j = k$$

$$\left[r \frac{d}{dr} U_0(q_k r) \right]_{r=R_1} = q_k R_1 \left[J_0'(q_k R_1) Y_0(q_k R_1) - Y_0'(q_k R_1) J_0(q_k R_1) \right]$$

dies ist die Wronskische Determinante $[J_0, Y_0]$ deren Wert nach [2.5] gleich $-\frac{2}{\pi}$ ist.

$$\left[r \frac{d}{dr} U_0(q_k r) \right]_{r=R_0} = q_k R_0 \left[J_0'(q_k R_0) Y_0(q_k R_0) - Y_0'(q_k R_0) J_0(q_k R_0) \right]$$

hierfür gilt wegen (13):

$$\begin{aligned} &= \frac{J_0(q_k R_1)}{J_0(q_k R_0)} \cdot q_k R_0 \left[J_0'(q_k R_0) Y_0(q_k R_0) - Y_0'(q_k R_0) J_0(q_k R_0) \right] \\ &= -\frac{2}{\pi} \frac{J_0(q_k R_1)}{J_0(q_k R_0)} \end{aligned}$$

Damit haben wir als Ergebnis:

$$I_1 = \int_{R_0}^{R_1} r U_0(q_j r) \cdot U_0(q_k r) dr = \begin{cases} 0 & \text{für } j \neq k \\ \frac{2 \{ J_0^2(q_k R_0) - J_0^2(q_k R_1) \}}{\pi^2 q_k^2 J_0^2(q_k R_0)} & \text{für } j = k \end{cases}$$

Analog lassen sich mit partieller Integration unter Beachtung von (16) ableiten:

$$I_2 \equiv \int_{R_0}^{R_1} \tau U_0(q_R \tau) d\tau = \frac{2 \{ J_0(q_R R_0) - J_0(q_R R_1) \}}{\pi q_R^2 J_0(q_R R_0)}$$

$$I_3 \equiv \int_{R_0}^{R_1} \tau^3 U_0(q_R \tau) d\tau = \frac{2 \left\{ \left(R_1^2 - \frac{4}{q_R^2} \right) J_0(q_R R_0) - \left(R_0^2 - \frac{4}{q_R^2} \right) J_0(q_R R_1) \right\}}{\pi q_R^2 J_0(q_R R_0)}$$

$$I_4 \equiv \int_{R_0}^{R_1} \tau \cdot \ln \tau \cdot U_0(q_R \tau) d\tau = \frac{2 \{ J_0(q_R R_0) \ln R_1 - J_0(q_R R_1) \cdot \ln R_0 \}}{\pi q_R^2 J_0(q_R R_0)}$$

3.2 Vollständige Lösung

Die Koeffizienten C_k folgen aus (15 und 8) durch Multiplikation mit $\tau U_0(q_j \tau)$ und Integration von R_0 bis R_1 :

$$- \int_{R_0}^{R_1} \tau U_0 v_{\infty}(\tau) d\tau = \int_{R_0}^{R_1} \tau U_0(q_j \tau) \cdot \sum_{k=0}^{\infty} C_k U_0(q_k \tau) d\tau$$

Unter Beachtung der Ergebnisse von Kap. 3.1 ergibt sich:

$$C_k = -\pi \left[\left(\gamma \{ J_0(q_k R_0) - J_0(q_k R_1) \} + \beta \{ J_0(q_k R_0) \ln R_1 - J_0(q_k R_1) \ln R_0 \} + \alpha \left\{ \left(R_1^2 - \frac{4}{q_k^2} \right) J_0(q_k R_0) - \left(R_0^2 - \frac{4}{q_k^2} \right) J_0(q_k R_1) \right\} \right) / \left\{ J_0^2(q_k R_0) - J_0^2(q_k R_1) \right\} \right] \cdot J_0(q_k R_0) \quad (17)$$

wobei α, β und γ Kap. 2 und q_k Gleichung (13) zu entnehmen sind.

$$v(t, \tau) = \alpha \tau^2 + \beta \ln \tau + \gamma + \sum_{k=1}^{\infty} C_k e^{-q_k^2 \frac{t}{Re}} \left(J_0(q_k \tau) \cdot Y_0(q_k R_1) - Y_0(q_k \tau) J_0(q_k R_1) \right) \quad (18)$$

3.3 Über die Lösungen q_k der Gleichung (13)

Mit $x_R \equiv q_R \cdot R_0$ und $y \equiv R_1/R_0$ lautet Gl. (13):

$$f(x_R, y) \equiv J_0(x_R) \cdot Y_0(yx_R) - Y_0(x_R) \cdot J_0(yx_R) \stackrel{!}{=} 0$$

Für $y = 1.2, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0$ sind [2] die

ersten fünf Wurzeln zu entnehmen:

$y = 2 \curvearrowright$

x_1	x_2	x_3	x_4	x_5
3.1230	6.2734	9.4182	12.5614	15.7040

Die Wurzeln lassen sich iterativ nach dem Newton-Verfahren bestimmen:

$$x_R^{(n+1)} = x_R^{(n)} - \frac{f(x_R^{(n)}, y)}{f'(x_R^{(n)}, y)} ; \quad x_R^{(0)} = \frac{k \cdot \pi}{y-1} ; \quad n=0,1,2,\dots \quad (20)$$

mit

$$f'(x_R, y) = J_1(yx_R) \cdot Y_0(x_R) \cdot y + J_0(yx_R) \cdot Y_1(x_R) - J_1(x_R) \cdot Y_0(yx_R) - J_0(x_R) \cdot Y_1(yx_R) \cdot y \quad (21)$$

Für $x \gg 1$ lassen sich die Wurzeln analytisch bestimmen:

$$x \gg 1 \curvearrowright \quad \begin{aligned} J_0(x) &\approx \sqrt{\frac{2}{\pi x}} \sin\left(x + \frac{\pi}{4}\right) \\ Y_0(x) &\approx -\sqrt{\frac{2}{\pi x}} \cos\left(x + \frac{\pi}{4}\right) \end{aligned} \quad [4] \quad (22)$$

$$\begin{aligned} \frac{\pi x}{2} \cdot f(x, y) &\approx -\sin\left(x + \frac{\pi}{4}\right) \cdot \cos\left(yx + \frac{\pi}{4}\right) \\ &\quad + \sin\left(yx + \frac{\pi}{4}\right) \cdot \cos\left(x + \frac{\pi}{4}\right) \\ &= \sin(x \cdot (y-1)) \stackrel{!}{=} 0 \end{aligned}$$

$$\curvearrowright \quad x_j (y-1) = j \cdot \pi$$

$$x_j = \frac{j \cdot \pi}{y-1} \quad , \quad j \gg 1$$

4. Lösungen für Platte und Zylinder

4.1 Platte

$$v(t, r) = \alpha + \gamma r^2 + \sum_{k=1}^{\infty} C_k e^{-q_k^2 \frac{t}{Re}} \cos(q_k r) \quad (24)$$

$$q_k = (k - \frac{1}{2}) \pi / R_1, \quad k = 1, 2, \dots \quad (25)$$

$$\alpha = p Re R^2 / 2, \quad R \equiv R_1 \quad (26)$$

$$\gamma = -p Re / 2$$

$$C_k = \frac{-\alpha \sin(q_k R) - \gamma (R^2 - \frac{2}{q_k^2}) \sin(q_k R)}{\frac{1}{2} q_k (R + \frac{1}{2q_k} \sin(2q_k R))} \quad (27)$$

4.2 Zylinder

$$v(t, r) = \alpha + \gamma r^2 + \sum_{k=1}^{\infty} C_k e^{-q_k^2 \frac{t}{Re}} J_0(q_k r) \quad (28)$$

$$J_0(q_k R) = 0 \quad \rightarrow q_k, \quad k = 1, 2, \dots \quad (29)$$

$$\alpha = p \cdot Re \cdot R^2 / 4 \quad (30)$$

$$\gamma = -p \cdot Re / 4$$

$$C_k = - \left\{ \gamma (R^2 - 4/q_k^2) - \alpha \right\} / \left\{ \frac{R}{2} q_k J_1(q_k R) \right\} \quad (31)$$

Über die Lösungen q_k der Gleichung (29) $J_0(x_k) = 0$:
Die ersten sechs Wurzeln sind [2] zu entnehmen:

x_1	x_2	x_3	x_4	x_5	x_6
2,4048	5.5201	8.6537	11.7915	14.9309	18.0711

Die Wurzeln lassen sich iterativ nach dem Newton-Verfahren entsprechend (26) mit

$$f(x) = J_0(x), \quad f'(x) = -J_1(x), \quad x_k^{(0)} = \pi \left(k - \frac{1}{4}\right) \quad (32)$$

errechnen.

Für $x \gg 1$ lassen sich die Wurzeln analytisch bestimmen:

$$\begin{aligned} f(x) = J_0(x) &\approx \sqrt{\frac{2}{\pi x}} \sin\left(x + \frac{\pi}{4}\right) \quad [4] \\ \leadsto x_j &\approx \pi \left(j - \frac{1}{4}\right) \quad j \gg 1 \end{aligned} \quad (33)$$

5. Bemerkung zum Unterschied zwischen Anlauf- und Einlaufströmungen

Die in den vorhergehenden Kapiteln beschriebene Anlaufströmung tritt auf, wenn die genannten Körper mit einer Flüssigkeit gefüllt sind, die sich im Ruhezustand befindet und auf die vom Zeitpunkt $t = 0$ an ein konstanter Druckgradient p wirkt. Die Analogie aus der Wärmeübertragung ist ein Körper mit konstanter Temperatur in dem vom Zeitpunkt $t = 0$ eine konstante Wärmequellendichte induziert wird.

Mit den gleichen Methoden lassen sich auch Anlaufströmungen berechnen, bei denen $v(t = 0, r)$ ein beliebiges von nullverschiedenes Anfangsströmungsprofil (Temperaturprofil) beschreibt; beispielsweise ein "slug flow", $v(t = 0, r) = \text{const.} = v_0$. Für diesen Fall werden die analytischen Lösungen mit Konstanten C_k beschrieben in deren Bestimmungsgleichungen γ durch $(\gamma - v_0)$ ersetzt ist.

Dieser letzte Fall hat Ähnlichkeit mit der stationären Einlaufströmung, die durch folgende Gleichungen (in Zylindergeometrie) beschrieben wird:

$$v \frac{\partial v}{\partial z} + u \frac{\partial v}{\partial r} = p + \frac{1}{Re} \left\{ \frac{\partial^2 v}{\partial r^2} + \frac{1}{r} \frac{\partial v}{\partial r} + \frac{\partial^2 v}{\partial z^2} \right\} \quad (34)$$

$$v \frac{\partial u}{\partial z} + u \frac{\partial u}{\partial r} = \frac{1}{Re} \left\{ \frac{\partial^2 u}{\partial r^2} + \frac{1}{r} \frac{\partial u}{\partial r} + \frac{\partial^2 u}{\partial z^2} \right\} \quad (35)$$

$$\frac{\partial v}{\partial z} + \frac{\partial u}{\partial r} + \frac{u}{r} = 0 \quad (36)$$

Hierbei ist v die Geschwindigkeit in der axialen (z)-Richtung und u die Geschwindigkeit in der radialen Richtung.

Randbedingung $v(z, \text{Rand}) = 0, \quad u(z, \text{Rand}) = 0$

Anfangsbedingung $v(z=0, r) = \overline{v_\infty}$

Für $z \rightarrow \infty$ und $t \rightarrow \infty, u(z=0, r) = 0$ haben beide Probleme die gleichen

Lösungen $v(r) = v_{\infty}(r)$

$$u(r) = 0$$

Für endliches z bestehen jedoch Unterschiede zu der Anlaufströmung mit endlichem t , deren Größe (nach meinem Wissen) nur durch eine numerische Lösung der Einlaufströmung bestimmt werden kann.

6. Äquivalenz des Quellterms p mit einem konstant beschleunigten Rand.

Friedland [7] hat für Wärmeleitungsvorgänge gezeigt, daß folgende Probleme die gleiche Lösung besitzen:

a) $\frac{\partial v}{\partial t} = \frac{1}{Re} \nabla^2 v + p(t), \quad v(t, \text{Rand}) = 0 \quad (37)$

b) $\frac{\partial v}{\partial t} = \frac{1}{Re} \nabla^2 v, \quad v(t, \text{Rand}) = - \int_0^t p(t) dt \quad (38)$

Hierbei ist mit $v(t, r)$ die Differenz zwischen dem Feldwert im Inneren am Ort r und am Rand zur Zeit t bezeichnet. Wie leicht einsichtig ist, führt dieses Äquivalenzprinzip hier zu der Aussage, daß die oben berechneten Lösungen $v(t, r)$ auch Lösungen des Problems

$$\frac{\partial v}{\partial t} = \frac{1}{Re} \nabla^2 v, \quad v(t, \text{Rand}) = - pt, \quad v(t=0, r) = 0 \quad (39)$$

sind. D.h., eine konstant beschleunigte Oberfläche erzeugt das gleiche Geschwindigkeitsprofil wie ein konstanter Druck.

II. ANLAUF, ANPRIN, ANPLØT

1. Zweck

Zur Kontrolle von numerischen Verfahren zur Lösung instationärer Probleme aus dem Gebiet der Hydrodynamik ist es oft wünschenswert, analytische Vergleichslösungen verfügbar zu haben. ANLAUF berechnet eine analytische Lösung für folgendes Problem:

Differentialgleichung:

$$\frac{\partial V}{\partial t} = \frac{1}{Re} \nabla^2 V + P \quad (1)$$

Anfangsbedingung $V(t=0, r) = V_0 = \text{const} |_r$

Randbedingung $V(t, \text{Rand}) = 0$

Es werden die folgenden Geometrien betrachtet:

- 1.) unendlich ausgedehnte Platte,
- 2.) unendlich langer Zylinder,
- 3.) unendlich langer Hohlzylinder,

Das Programm ANLAUF berechnet zu einem als Argument übergebenen Ortsvektor R den Geschwindigkeitsvektor V(R) zur Zeit T.

ANPRIN druckt darüber hinaus das Ergebnis der Berechnung aus.

ANPLØT zeichnet den Verlauf der Geschwindigkeit V(R) über dem Ort auf dem CALCOMP-Plotter zu verschiedenen Zeitpunkten T_i , die durch einen übergebenen Zeitvektor T gegeben sind.

2. Anwendung

2.1 ANLAUF

Aufruf: CALL ANLAUF (T, NR, R, RO, RI, P, RE, VO, IGEØ, V)

(alle physikalischen Größen sind dimensionslos)

Dabei bedeuten:

- T Zeitpunkt, für den der Geschwindigkeitsvektor berechnet werden soll
- NR Zahl der Ortspunkte Ri des Ortsvektors
- R Vektor der Ortspunkte
- RO innerer Randflächenradius für Hohlzylindergeometrie, für Platten- und Zylindergeometrie ohne Bedeutung, RO wird dann auf -RI gesetzt.
- RI Äußerer Randflächenradius für Hohlzylindergeometrie bzw. Radius für Platten- oder Zylindergeometrie
- P Druck = - grad p
- RE Reynoldszahl
- VO Anfangswert der Geschwindigkeit.
- IGEØ Geometrie-Parameter
- IGEØ = 1 Plattengeometrie
- IGEØ = 2 Zylindergeometrie
- IGEØ = 3 Hohlzylindergeometrie
- V Ergebnisvektor der Geschwindigkeitswerte

Soweit möglich, werden die Eingabeparameter zu ANLAUF auf ihre Richtigkeit überprüft. Es erfolgt eine Fehlermeldung, wenn:

1. IGEO NICHT GLEICH 1,2 ODER 3
2. NR KLEINER 1
3. T KLEINER 0
4. P KLEINER 0
5. R1 NICHT GROESSER 0
6. RE NICHT GROESSER 0
7. R1 NICHT GROESSER RO FUER IGEO=3
8. WENN BEI IGEO = 3 DER DGL-TERM, DER DAS HOHLZYLINDERPROBLEM VOM PLATTENPROBLEM UNTERSCHIEDET, KLEINER IST ALS E-5, WIRD MIT PLATTENGEOMETRIE GERECHNET, ES ERFOLGT MELDUNG: ERR=.... < E-5, GESCHWINDIGKEITSVERTEILUNG GERECHNET MIT FORMELN FUER PLATTENGEOMETRIE
9. WENN NACH 10000 ITERATIONSSCHRITTEN DIE ABRUCHBEDINGUNG NOCH NICHT ERFÜLLT IST

Die Differentialgleichungen für das Hohlzylinder- und das Plattenproblem unterscheiden sich durch den Term:

$$\frac{1}{Re \cdot r} \cdot \frac{\partial v}{\partial r}, \text{ Für } (R1 - R0) \ll R1 \text{ ergeben sich}$$

für die Berechnung des Geschwindigkeitsvektors. große numerische Ungenauigkeiten. Jedoch ist für $(R1 - R0) \ll R1$ $v(r)$

Hohlzylinder
 $R0 \leq r \leq R1$

$$\approx v(r) \text{ Platten mit } R = \frac{R0 - R1}{2} \text{ Um abschätzen zu können, } -R \leq r \leq +R$$

wann mit Plattengeometrie gerechnet werden darf, wird der Term

$$\frac{1}{Re \cdot r} \frac{\partial v}{\partial r} \text{ angenähert durch } ERR = \frac{1}{Re \cdot R0} \cdot \frac{V_{\text{max}}((R0 + R1)/2)}{(R1 - R0)}$$

Ist $ERR < 10^{-5}$, wird mit Plattengeometrie gerechnet und es erfolgt die Meldung:

ERR=..... < E-5, GESCHWINDIGKEITSVERTEILUNG GERECHNET MIT FORMELN FUER PLATTENGEOMETRIE

Der Wert 10^{-5} wurde aufgrund von numerischen Rechnungen gefunden. Für $ERR < 10^{-5}$ ist die Ungenauigkeit, die durch die Rechnung mit den Plattenformeln entsteht, geringer als die numerischen Ungenauigkeiten, die sich bei Rechnung mit den Hohlzylinderformeln ergeben würden.

2.2 ANPRIN

Aufruf: CALL ANPRIN (T, NR, R, RO, RI, P, RE, VO, IGEØ, V)

Die Parameter haben dieselbe Bedeutung wie in der Subroutine ANLAUF.

ANPRIN berechnet das Geschwindigkeitsfeld V durch Aufruf von ANLAUF und druckt das Ergebnis in folgender Form aus:

	T	NR	RO	RI	P
	0.9999996E 01	11	-0.1000000E 01	0.1000000E 01	0.2000000E 01
I	R(I)		V(I)		
1	-0.1000000E 01	01	-0.6925921E-26		
2	-0.8000000E 00	00	0.3600000E 00		
3	-0.6000000E 00	00	0.6399999E 00		
4	-0.4000000E 00	00	0.8399999E 00		
5	-0.2000000E 00	00	0.9600000E 00		
6	0.0		0.9999999E 00		
7	0.1999998E 00	00	0.9600000E 00		
8	0.3999996E 00	00	0.8400003E 00		
9	0.5999994E 00	00	0.6400006E 00		
10	0.7999992E 00	00	0.3600012E 00		
11	0.1000000E 01	01	-0.6925921E-26		

(Die ausgedruckten Werte der Reynoldszahl RE, der Anfangsgeschwindigkeit VO und des Geometriefaktors IGEØ sind hier aus Platzgründen abgeschnitten.)

2.3 ANPLØT

Aufruf: CALL ANPLØT (T, NT, RO, RI, P, RE, VO, IGEØ, IDPLØT)

Bedeutung der Argumente, die von ANLAUF abweichen:

T Vektor von Zeitpunkten T_i , $T_i \geq 0$

NT Anzahl der Zeitpunkte T_i , $NT \leq 16$

ANPLØT liefert eine Fehlermeldung falls $NT > 16$ ist.

In diesem Fall wird NT auf 16 gesetzt.

IDPLØT Nummer, die die Zeichnung erhalten soll.

ANPLOT ruft für jeden Zeitpunkt T_i die Routine ANLAUF auf. Der Ortsvektor R besteht dabei aus den Punkten

$$R_i = R_0 + i \cdot \frac{R_1 - R_0}{20}, \quad i = 0, 1, \dots, 20.$$

Die Geschwindigkeitswerte werden linear interpoliert. Zum Zeichnen wird die CALCOMP-Software und der CALCOMP-Plotter 763 benutzt [8].

Dazu werden folgende JCL-Karten benötigt: *)

1. eine SETUP-Karte (nach der JOB-Karte)

```
/*SETUP DDNAME=PLOTTAPE,DEVICE=TAPE9,ID=(T147AN,,,NL)
```

T = Tuscheplot

P = Kugelschreiberplot

hier muß die DVZ-

Benutzernummer und die Jobkennzeichnung stehen, übereinstimmend mit der JOB-Karte.

2. eine DD-Karte für das Plotband
(im GØ-step)

```
//G.PLOTTAPE DD UNIT=TAPE9,LABEL=(,NL)
```

Das Zeichnen mit der Calcomp-Software erfordert eine Initialisierung vor dem ersten Plot und ein Schließen nach dem letzten Plot eines Jobs.

ANPLOT initialisiert das Plotband, wenn $IDPLOT=1$ gesetzt wird. Aus diesem Grunde muß im ersten Aufruf von ANPLOT in einem Job, in dem nicht schon vorher CALCOMP-Zeichnungen erzeugt wurden, $IDPLOT=1$ sein. Das Schließen des Plotbandes kann entweder durch ANPLOT geschehen oder durch ein explizites Schließen durch Aufruf der Routine PLOTTC. Wenn beim letzten Aufruf von ANPLOT in einem Job $IDPLOT=999$ gesetzt wird, wird das Plotband von ANPLOT geschlossen. Andernfalls muß nach dem letzten Plot der Subroutinenaufruf:

```
CALL PLOTTC (12.0, 0.0, 999)
```

durch den Benutzer erfolgen.

*) Diese Angaben beziehen sich auf die im Kernforschungszentrum vorhandene Maschinenkonfiguration (IBM 360/65 und 360/85).

3. Beispiele

3.1 ANPRIN für Plattengeometrie

```
DIMENSION R(11), V(11)
R1=10.
DØI=1,11
R(I)=-R1+ 2*R1/11*(I-1)
1 CØNTINUE
  :
  CALL ANPRIN (10.,11,R,0.,R1,2.,1.,0.,1,V)
  :
```

	T	NR	RO	R1	P
0.9999995E	02	11	-0.1000000E 02	0.1000000E 02	0.2000000E 01
I		R(I)	V(I)		
1		-0.1000000E 02	-0.4579030E-14		
2		-0.8000000E 01	0.3329538E 02		
3		-0.6000000E 01	0.5885551E 02		
4		-0.4000000E 01	0.7691924E 02		
5		-0.2000000E 01	0.8767606E 02		
6		0.0	0.9124770E 02		
7		0.2000000E 01	0.8767606E 02		
8		0.4000000E 01	0.7691924E 02		
9		0.6000000E 01	0.5885551E 02		
10		0.8000000E 01	0.3329538E 02		
11		0.1000000E 02	-0.4579030E-14		

3.2 ANPRIN für Zylindergeometrie

```
R1=1.
DØ 2 I=1,11
R(I)=-R1+ 2*R1/11.*(I-1)
2 CØNTINUE
  :
  CALL ANPRIN (0.01,11,R,0.,R1,2.,1.,0.,2,V)
  :
```


	T	NR	RO	R1	P
	0.9999998E-02	11	-0.1000000E 01	0.1000000E 01	0.2000000E 01
I		R(I)		V(I)	
1	-0.1000000E 01	01	0.0		
2	-0.8000000E 00	00	0.1872708E-01		
3	-0.6000000E 00	00	0.1997928E-01		
4	-0.4000000E 00	00	0.2000020E-01		
5	-0.2000000E 00	00	0.2000174E-01		
6	0.0		0.2000241E-01		
7	0.1999998E 00	00	0.2000174E-01		
8	0.3999996E 00	00	0.2000020E-01		
9	0.5999994E 00	00	0.1997928E-01		
10	0.7999992E 00	00	0.1872709E-01		
11	0.1000000E 01	01	0.0		

3.3 ANPRIN für Hohlzylindergeometrie

RO=10.

R1=11.

DØ 3 I=1,11

$R(I)=RO+(R1-RO)/11.*(I-1)$

3 CØNTINUE

CALL ANPRIN (1.0E6,11,R,RO,R1,2.,1.,2.,3,V)

	T	NR	RO	R1	P
	0.1000000E 07	11	0.1000000E 02	0.1100000E 02	0.2000000E 01
I		R(I)		V(I)	
1	0.1000000E 02	02	0.5684342E-13		
2	0.1010000E 02	02	0.9119380E-01		
3	0.1020000E 02	02	0.1615883E 00		
4	0.1030000E 02	02	0.2113929E 00		
5	0.1040000E 02	02	0.2408131E 00		
6	0.1050000E 02	02	0.2500473E 00		
7	0.1060000E 02	02	0.2392885E 00		
8	0.1070000E 02	02	0.2087244E 00		
9	0.1080000E 02	02	0.1585379E 00		
10	0.1090000E 02	02	0.8890545E-01		
11	0.1100000E 02	02	0.1776357E-13		

3.4 ANPRIN für Hohlzylindergeometrie, gerechnet mit Formeln für Plattengeometrie

```

:
:
RO=1000.
Rl=1001.
DØ 4 I=1,11
R(I)= RO + (Rl-RO)/11.*(I-1)
4 CØNTINUE
:
:
CALL ANPRIN (0.1,11,R,RO,Rl,2.,1.,0.,3,V)
:
:
```

	T	NR	RO	Rl	P
	0.9999998E-02	11	0.1000000E 04	0.1001000E 04	0.2000000E 01
I	R(I)		V(I)		
1	0.1000000E 04		-0.8153372E-16		
2	0.1000100E 04		0.1439111E-01		
3	0.1000200E 04		0.1886321E-01		
4	0.1000300E 04		0.1983879E-01		
5	0.1000400E 04		0.1998460E-01		
6	0.1000500E 04		0.1999807E-01		
7	0.1000600E 04		0.1998470E-01		
8	0.1000700E 04		0.1983964E-01		
9	0.1000800E 04		0.1886811E-01		
10	0.1000900E 04		0.1441061E-01		
11	0.1001000E 04		-0.8153372E-16		

ERR= 0.249875D-06 <E-5, GESCHWINDIGKEITSVERTEILUNG GERECHNET MIT FORMELN FUER PLATTENGEOMETRIE

3.5 ANPLØT

```

:
:
T(1)=0.0
T(2)=0.01
T(3)=0.1
T(4)=1.0
T(5)=10.0
T(6)=100.0
T(7)=1000.0
T(8)=1000000.0
CALL ANPLØT(T,8,0,1,1.0,2.,1.,0.,3,1)
CALL ANPLØT(T,8,0.5,1.0,2.,1.,0.,3,1)
:
:
CALL ANPLØT(T,8,0.,11.,2.,1.,0.,1,8)
:
:
```

siehe Abb. 2
" Abb. 3
" Abb. 4

```
CALL ANPLOT(T,8,0.,10.,2.,1.,2.,2,2)   siehe Abb. 5  
CALL PLOTTC (12.0,0.0,999)
```

Der Knick im Geschwindigkeitsverlauf für $T=0$ und $V_0 \neq 0$ rührt daher, daß die Randbedingung $V(\text{Rand}, t)=0$ stärker ist als die Anfangsbedingung $V(t=0)=V_0$ und zwischen den einzelnen Stützpunkten linear interpoliert wird. Die an den Kurven stehenden Texte "N=1", "N=2", usw. bezeichnen das Element des Zeitvektors, zu dem die betreffende Kurve gehört.

4. Rechengenauigkeit

Da ANLAUF in erster Linie dazu dienen soll, numerische Verfahren zu kontrollieren, wurde Wert gelegt auf größtmögliche Genauigkeit. ANLAUF rechnet daher intern doppelt genau. Die Genauigkeit des Ergebnisses ist abhängig von $IGEO$, T , RO , RI und VO . Die Geschwindigkeitswerte ergeben sich aus einer konvergierenden Reihe. Je weniger Summationen nötig sind, bis die Differenz zwischen i -ter und $(i+1)$ -ter Zwischensumme kleiner als eine Schranke ist, desto größer ist die numerische Genauigkeit. Für $T=0$ werden die meisten Summationen benötigt. Andererseits ist aber $V(t=0) = V_0$. Man kann deshalb die Genauigkeit eines Ergebnisses für T , RO , RI , VO , $IGEO$ abschätzen, indem man sich den Geschwindigkeitsvektor für die gleichen Werte RO , RI , VO , $IGEO$, jedoch zum Zeitpunkt $T=0$ berechnen läßt und die Abweichung der errechneten Geschwindigkeitswerte von V_0 betrachtet. Die Werte $|V(t=0) - V_0|$ sind obere Grenzen für den tatsächlichen Fehler zu Zeiten $t > 0$.

5. Speicherbelegung, Rechenzeit

ANLAUF, ANPRIN und ANPLOT, sowie die dazugehörenden Subroutinen und Functions benötigen zusammen 100 098 Bytes (= 97,75 K Bytes).

Die Rechenzeit ist stark abhängig von der Zeit T , von RO und RI , von $IGEO$, NR (bzw. bei ANPLOT NT), sowie von VO .

Beispiele:

1. $IGEO=3$, $RO=0.5$, $RI=10.0$, $NR=11$, $VO=2.0$, $NT=8$

Rechenzeit $t_{CPU} \approx 1 \text{ min } 16 \text{ sec}$ pro Aufruf
von ANLAUF oder ANPRIN,

$$t_{CPU} \approx 4 \text{ min } 49 \text{ sec} \text{ pro Aufruf}$$

von ANPLOT (M65-Zeit) *)

2. $IGEO=3$, $RO=10.0$, $RI=11.0$, $NR=11$, $VO=0.0$, $NT=8$

$$t_{CPU} \approx 2 \text{ sec} \text{ pro Aufruf ANLAUF oder ANPRIN}$$

$$t_{CPU} \approx 31,5 \text{ sec} \text{ pro Aufruf ANPLOT (M65-Zeit)}$$

*) M65 = IBM 360/65

3. IGEØ=1, R1=300.0, NR=11, VO=2.0, NT=8

$t_{\text{CPU}} \approx 6$ sec pro Aufruf ANLAUF oder
ANPRIN

$t_{\text{CPU}} \approx 1$ min 31 sec pro Aufruf
(M65-Zeit) ANPLØT

6. Zur Programmlogik

Die Programme ANLAUF, ANPRIN und ANPLØT benutzen zur Berechnung des Geschwindigkeitsprofils die in Abschnitt I beschriebenen Verfahren. Die Summation über die unendlichen Reihen in den Ausdrücken (18), (24) und (28) wird abgebrochen, wenn die Differenz zwischen i -ter und $(i+1)$ -ter Zwischensumme, bezogen auf $V_{\text{max}}(t=\infty)$, kleiner wird als 10^{-12} . Der Wert 10^{-12} wurde als Abbruchschranke gewählt, weil die numerischen Ungenauigkeiten stets größer sind als 10^{-12} , so daß die Berücksichtigung weiterer Summanden die Genauigkeit des Ergebnisses nicht mehr verbessern könnte. Zur Berechnung der Nullstellen der Funktionen (13) und (19) nach dem Newton-Verfahren wird die Subroutine NEWTØN verwendet. Genauigkeit der Nullstellenberechnung:

X_0 = exakte Nullstelle

X_{ON} = errechnete Nullstelle

$$\frac{|X_0 - X_{\text{ON}}|}{X_{\text{ON}}} \leq 10^{-12} \quad \text{für } |X_{\text{ON}}| \geq 10^{-12}$$

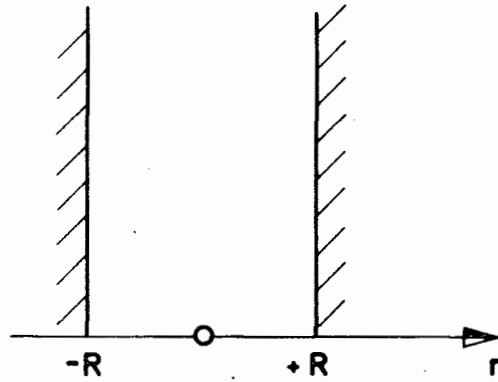
$$|X_0 - X_{\text{ON}}| \leq 10^{-12} \quad \text{für } |X_{\text{ON}}| < 10^{-12}$$

Für die Zylinder- und Hohlzylindergeometrie werden die Besselfunktionen J_0 , J_1 , Y_0 , Y_1 benötigt. Die im SSP [9] befindlichen Subroutinen BESJ und BESY erwiesen sich als zu ungenau. Es wurden deshalb die double precision functions BEFJO, BEFJ1, BEFY0 und BEFY1 erstellt. Sie berechnen den Wert der Besselfunktionen J_0 , J_1 , Y_0 und Y_1 nach den in [6] angegebenen Approximationspolynomen.

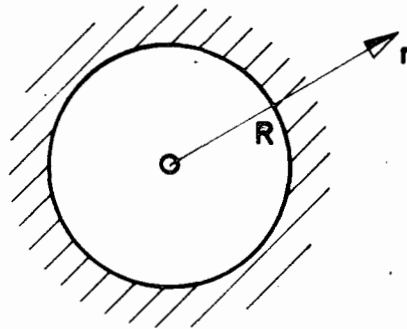
Literatur:

- [1_] Gröber, Erk, Grigull:
Wärmeübertragung
3-Aufl. (1963)
- [2_] H.S.Carslaw, J.C.Jaeger
Conduction of Heat in Solids
2. Aufl. (1959)
- [3_] A.V.Luikov
Analytical Heat Diffusion Theory
(1968)
- [4_] Glasstone, Edlund:
Kernreaktortheorie
(1961)
- [5_] R.Sauer, I.Szabo
Mathematische Hilfsmittel des Ingenieurs
Teil I, S. 116
- [6_] M.Abramowitz, I.A.Stegun (ed.):
Handbook of Mathematical Functions
Dover Publ., New York 1965
- [7_] Friedland, A.J.:
The Equivalence of Transient Heat Generation and
Transient Surface Temperature Boundary Conditions
in Heat Conduction
Nucl. Eng. Design 16 (1971), 151-152
- [8_] Calcomp GmbH, 4000 Düsseldorf-Oberkassel
Programme für Calcomp-Plotter der Serie 500,
600 und 700
- [9_] IBM Publikation Nr. H20-0205-3
System/360 Scientific Subroutine
Package (360A-CM-03X) Version III
Programmers Manual

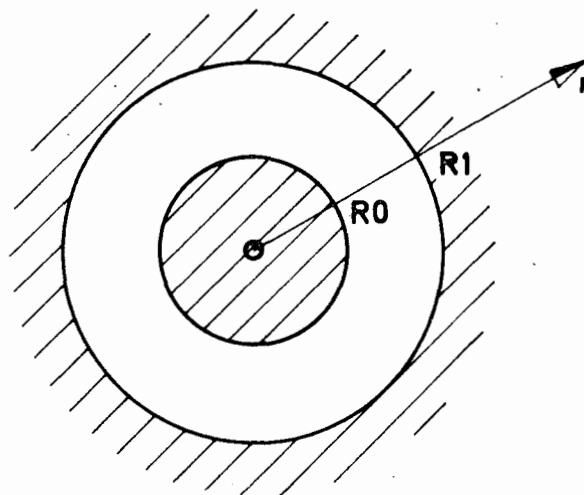
Geometrie 1): Unendlich ausgedehnte Platte
(IGE ϕ =1)



Geometrie 2): Unendlich langer Zylinder
(IGE ϕ =2)



Geometrie 3): Unendlich langer Hohlzylinder
(IGE ϕ =3)



(Das Medium strömt senkrecht zur Zeichenebene)

Abb. 1 Die betrachteten eindimensionalen Geometrien

IGEO= RING

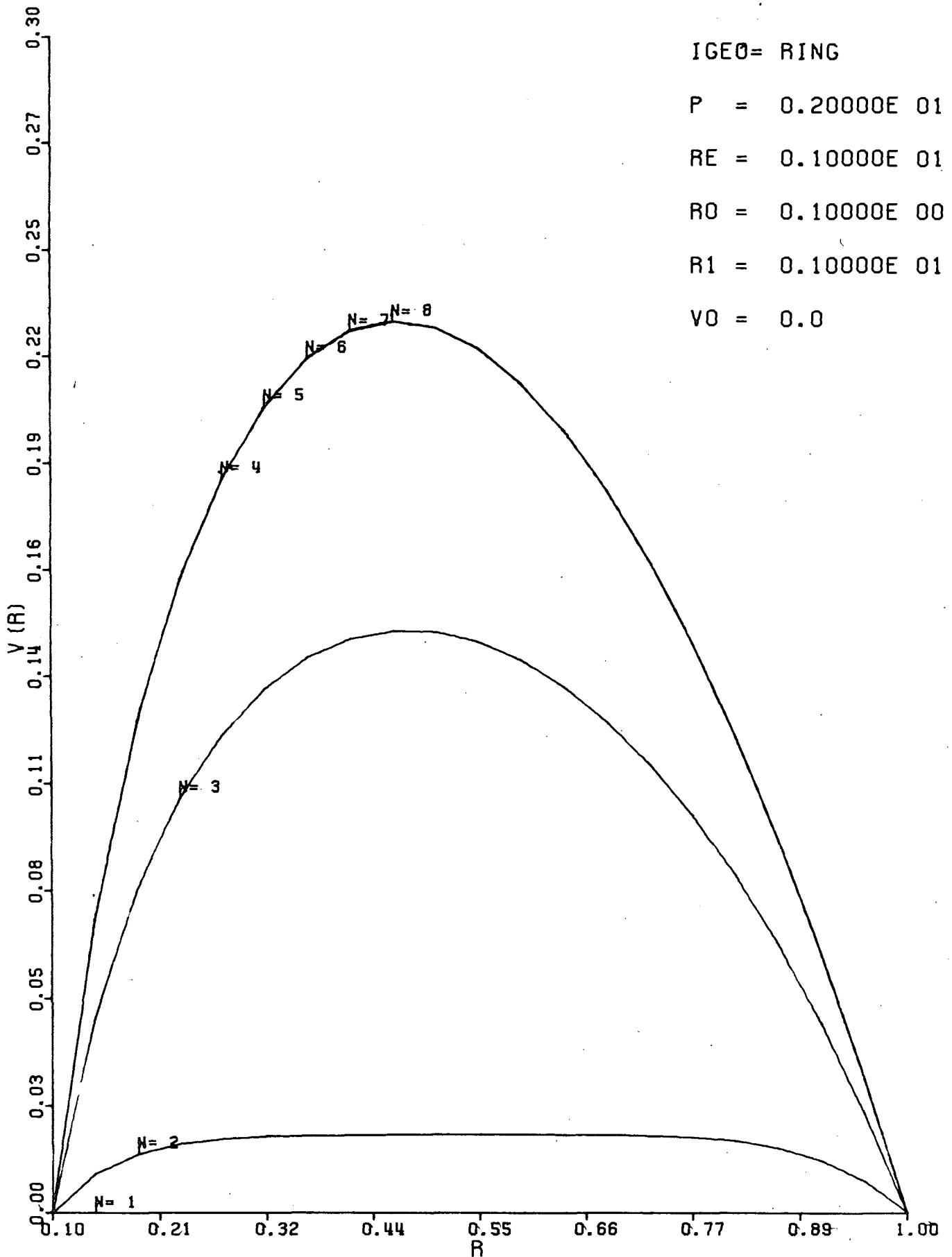
P = 0.20000E 01

RE = 0.10000E 01

RO = 0.10000E 00

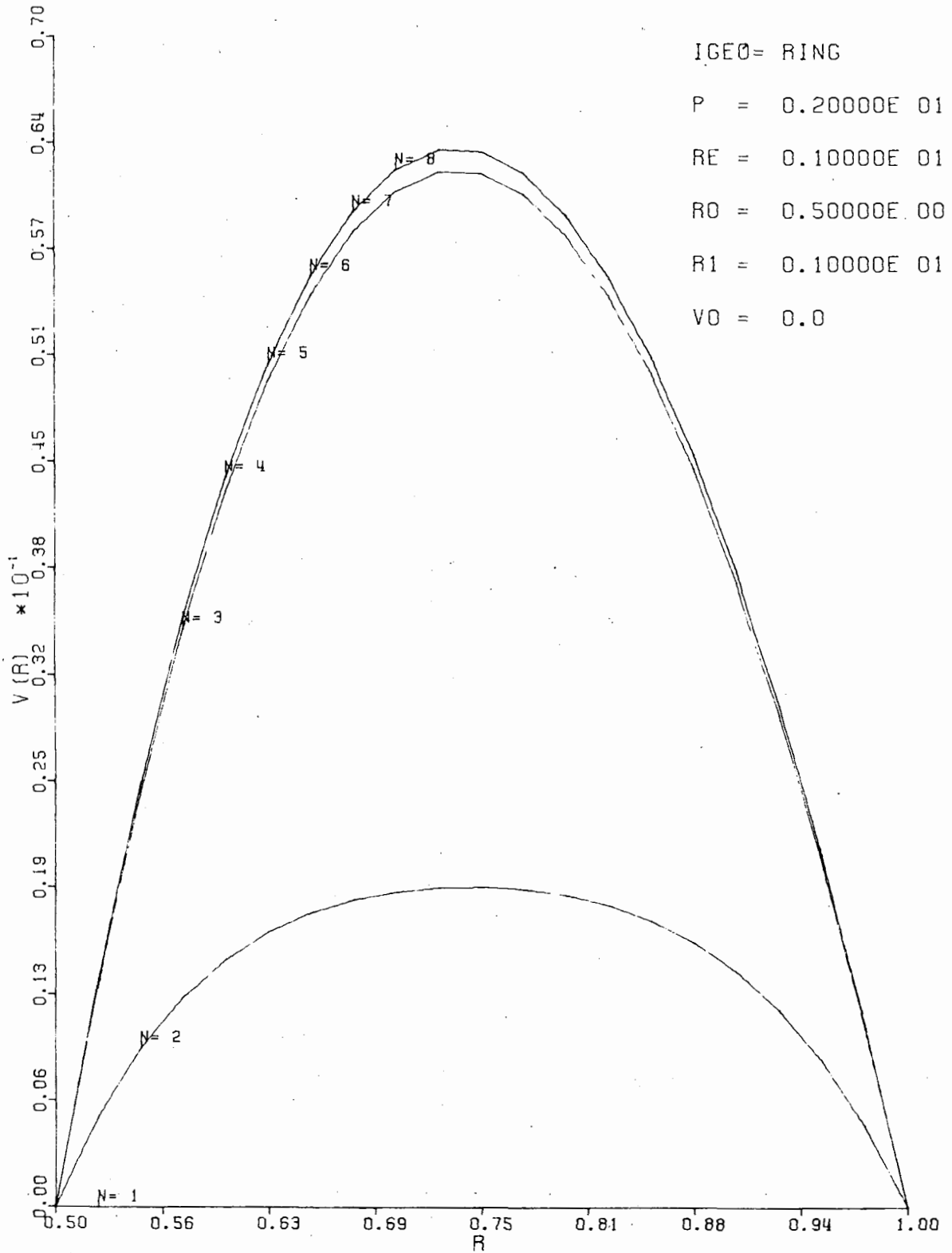
R1 = 0.10000E 01

VO = 0.0



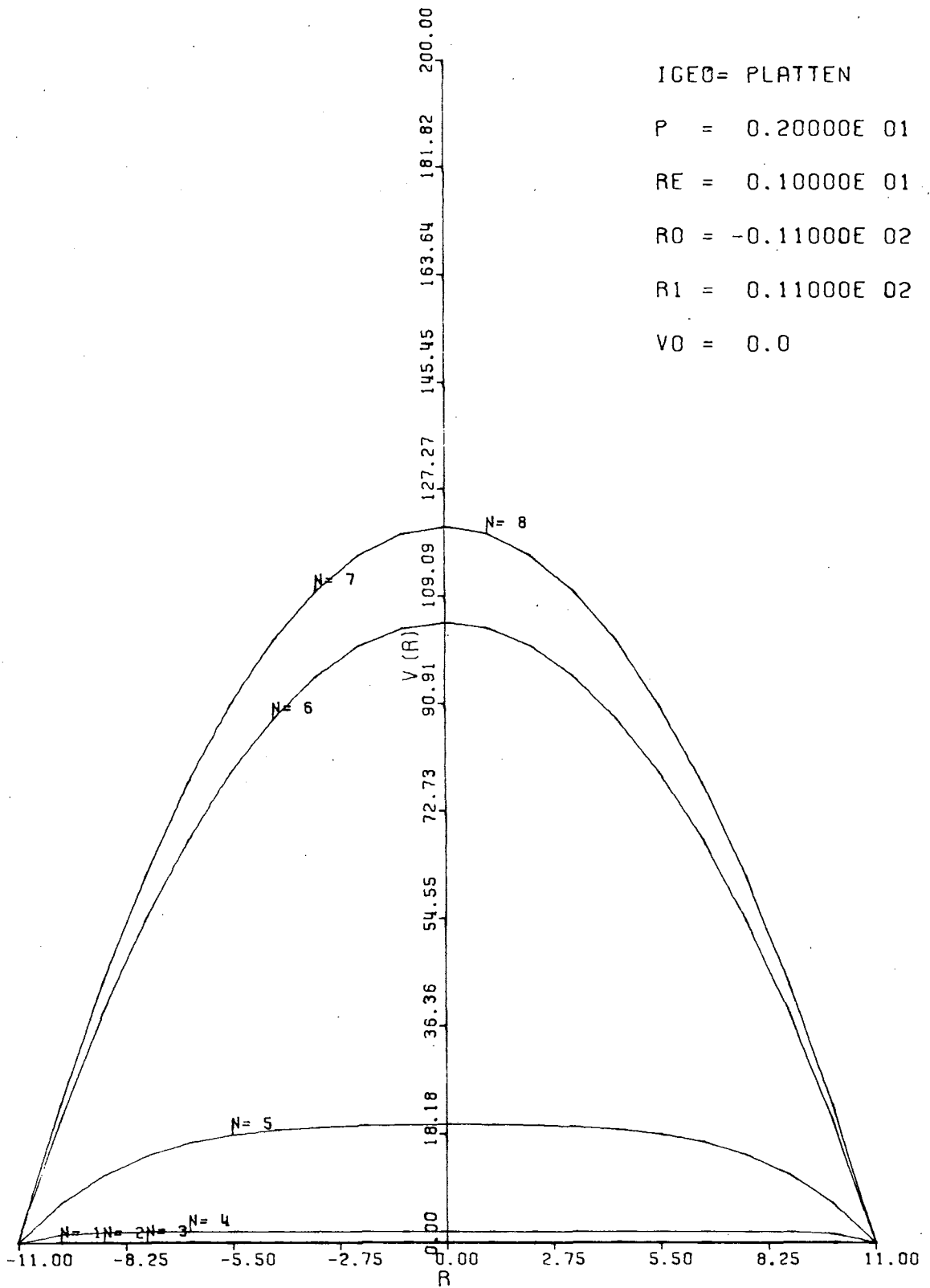
IDPLOT = 1

Abb. 2 Durch ANPLOT erzeugte Zeichnung bei Hohlzylinder-
geometrie mit $R1/RO = 10$



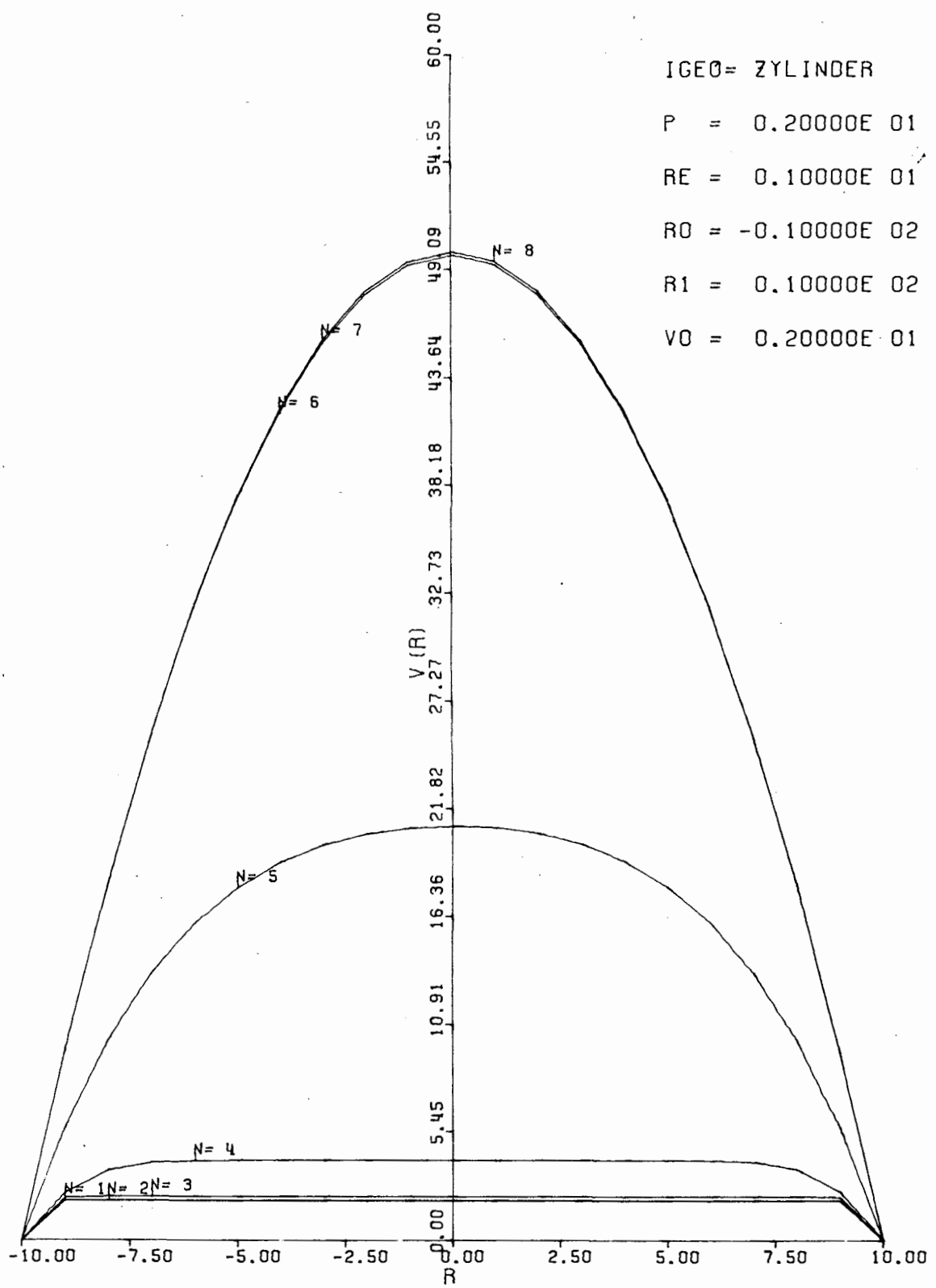
IDPLOT = 1

Abb. 3 Durch ANPLOT erzeugte Zeichnung bei Hohlzylinder-
geometrie mit $R1/RO=2$



IDPLOT = 8

Abb. 4 Durch ANPLOT erzeugte Zeichnung bei Plattengeometrie



IDPLOT = 2

Abb. 5 Durch ANPLOT erzeugte Zeichnung bei Zylindergeometrie

A N H A N G 1

LISTE DER PROGRAMME ANLAUF, ANPRIN UND ANPLOT

SUBROUTINE ANLAUF (\$T, NR, \$R, \$RO, \$R1, \$P, \$RE, \$VO, IGEO, \$V)

```

C
CN ANLAUF
C
CA ENDERLE
C
CD 15.7.71
C
CB BERECHNUNG VON ANLAUFSTROEMUNGEN IN PLATTEN-, ZYLINDER- UND
CB HOHLZYLINDERGEOMETRIE
CB ZU GEGEBENEM VEKTOR VON ORTSPUNKTEN WIRD DER VEKTOR DER
CB GESCHWINDIGKEITSWERTE BERECHNET
C
CP T ZEIT
CP NR ZAHL DER ORTSPUNKTE R(I), (I=1, NR) FUER WELCHE DIE
C GESCHWINDIGKEIT V(I) BERECHNET WERDEN SOLL
CP R VEKTOR DER ORTSPUNKTE
CP RO INNERER RANDFLAECHENRADIUS (WIRD NUR FUER HOHLZYLINDER-
C GEOMETRIE BENUTZT, MUSS JEDOCH AUCH BEI PLATTEN- UND
C ZYLINDERGEOMETRIE ALS DUMMY UEBERGEHEN WERDEN)
CP R1 AEUSSERER RANDFLAECHENRADIUS
CP P DRUCK=-DELTA(DRUCK)/DELTA(Z)
CP RE REYNOLDS-ZAHL
CP VO ANFANGSWERT VON V
CP IGEO = 1 PLATTE /
CP IGEO = 2 ZYLINDER / GEOMETRIE-PARAMETER
CP IGEO = 3 HOHLZYLINDER /
CP V VEKTOR DER GERECHNETEN GESCHWINDIGKEITSWERTE
C
IMPLICIT REAL*8(A-H,O-Z)
CE ES ERFOLGT EINE MELDUNG, WENN:
CE 1. IGEO NICHT GLEICH 1,2 ODER 3
CE 2. NR KLEINER 1
CE 3. T KLEINER 0
CE 4. P KLEINER 0
CE 5. R1 NICHT GROESSER 0
CE 6. RE NICHT GROESSER 0
CE 7. R1 NICHT GROESSER RO FUER IGEO=3
CE 8. WENN BEI IGEO = 3 DER DGL-TERM, DER DAS HOHLZYLINDERPROBLEM VOM
CE PLATTENPROBLEM UNTERSCHIEDET, KLEINER IST ALS E-3, WIRD MIT
CE PLATTENGEOMETRIE GERECHNET, ES ERFOLGT MELDUNG: ERR=..... <E-3,
CE GESCHWINDIGKEITSVERTeilUNG GERECHNET MIT FORMELN FUER PLATTEN-
CE GEOMETRIE
CE 9. WENN NACH 10000 ITERATIONSSCHRITTEN DIE ABRUCHBEDINGUNG
CE NOCH NICHT ERFUELLT IST
C
EXTERNAL F,DF,BEFJO,DBES

```

```

COMMON Y
DIMENSION $R(1),$V(1),QKI(5000),CKI(5000)
DATA NOUT/6/
CALL FSPIE
KMAX=0
IF ((IGEO.EQ.1).OR.(IGEO.EQ.2).OR.(IGEO.EQ.3)) GO TO 80
WRITE (NOUT,1000)
1000 FORMAT (' IGEO NICHT 1,2 ODER 3')
GO TO 999
80 IF (NR.GE.1) GO TO 90
WRITE (NOUT,1001)
1001 FORMAT (' NR KLEINER ALS 1')
GO TO 999
90 IF (($T.GE.0.0).AND.($R1.GT.0.0).AND.($P.GE.0.0).AND.
1($RE.GT.0.0)) GO TO 111
WRITE (NOUT,1003)
1003 FORMAT (' T ODER P NICHT GROESSER GLEICH 0.0 ODER R1 ODER RE NIC
1HT GROESSER 0.0')
GO TO 999
111 IF ($R1.GT.$R0) GO TO 91
WRITE (NOUT,1004)
1004 FORMAT (' R1 NICHT GROESSER R0')
GO TO 999
91 IF (IGEO.NE.3) GO TO 100
IF ($R0.GT.0.0) GO TO 100
WRITE (NOUT,1005)
1005 FORMAT (' R0 NICHT GROESSER NULL')
GO TO 999
100 PI=3.141592653589793238
T=$T
R0=$R0
R1=$R1
P=$P
RE=$RE
V0=$V0
IF (IGEO.EQ.1) GO TO 110
IF (IGEO.EQ.2) GO TO 200
Y=R1/R0
ALPHA=-P*RE/4.0
BETA=(P*RE*0.25)*(R0*R0-R1*R1)/DLOG(R0/R1)
GAMMA=(P*RE*0.25)*(R1*R1*DLOG(R0)-R0*R0*DLOG(R1))/DLOG(R0/R1)
ASSIGN 11 TO MARCA
ASSIGN 10 TO MARKE
VUNEND=ALPHA*((R0+R1)/2.0)**2+BETA*DLOG((R1+R0)/2.0)+GAMMA
ERR=2./((RE*R0)*VUNEND/(R0+R1))
IF (ERR.GT.1.0E-5) GO TO 52
WRITE (NOUT,1006) ERR
1006 FORMAT(' ERR=',E13.6,' <E-5, GESCHWINDIGKEITSVERTEILUNG GERECHNET
1MIT FORMELN FUER PLATTENGEOMETRIE')
R1ALT=R1
R0ALT=R0
R1=(R1ALT-R0ALT)/2.0
R0=-R1
DO 53 I=1,NR
53 $R(I)=$R(I)-R0ALT-R1
GO TO 110
52 DO 1 I=1,NR

```

```

R=$R(I)
V=ALPHA*R*R+BETA*DLOG(R)+GAMMA
DO 2 K=1,10000
IF (K.GT.5000) GO TO 30
IF (K.LE.KMAX) GO TO 32
KMAX=K
X0=(K*PI)/(Y-1)
X01=X0
GO TO MARCA,(11,21)
11 CALL NEWTON (X0,F,DF)
IF (X0.LT.1.0E-12) GO TO 41
IF (DABS((X0-X01)/X0).GT.1.0E-12) GO TO 21
GO TO 51
41 IF (DABS(X0-X01).GT.1.0E-12) GO TO 21
51 ASSIGN 21 TO MARCA
ASSIGN 20 TO MARKE
21 QKI(K)=X0/R0
QK=QKI(K)
QKR1=QK*R1
QKQK=QK*QK
CKI(K)=-PI*((GAMMA-V0)*(BEFJO(X0)-BEFJO(QKR1))+BETA*(BEFJO(X0)*
1DLOG(R1)-BEFJO(QKR1)*DLOG(R0))+ALPHA*((R1*R1-4.0/(QKQK))*BEFJO
1(X0)-(R0*R0-4/(QKQK))*BEFJO(QKR1)))/(BEFJO(X0)**2-BEFJO(QKR1)
1**2))*BEFJO(QK*R0)
CK=CKI(K)
GO TO 31
32 QK=QKI(K)
QKR1=QK*R1
CK=CKI(K)
GO TO 31
30 X0=(K*PI)/(Y-1)
X01=X0
GO TO MARKE, (10,20)
10 CALL NEWTON (X0,F,DF)
IF (X0.LT.1.0E-12) GO TO 40
IF (DABS((X0-X01)/X0).GT.1.0E-12) GO TO 20
GO TO 50
40 IF (DABS(X0-X01).GT.1.0E-12) GO TO 20
50 ASSIGN 20 TO MARKE
20 QK=X0/R0
QKR1=QK*R1
QKQK=QK*QK
CK=-PI*((GAMMA-V0)*(BEFJO(X0)-BEFJO(QKR1))+BETA*(BEFJO(X0)*
1DLOG(R1)-BEFJO(QKR1)*DLOG(R0))+ALPHA*((R1*R1-4.0/(QKQK))*BEFJO
1(X0)-(R0*R0-4/(QKQK))*BEFJO(QKR1)))/(BEFJO(X0)**2-BEFJO(QKR1)
1**2))*BEFJO(QK*R0)
31 VERGL=V
QKR=QK*R
EXPO=-QK*QK*T/RE
IF (EXPO.LE.-100.0) GO TO 33
EXPT=DEXP(EXPO)*(BEFJO(QKR)*BEFY0(QK*R1)-BEFY0(QKR)*BEFJO(QK*R1)
1)
GO TO 34
33 EXPT=0.0
34 V=V+CK*EXPT
IF (VUNEND.LT.1.0E-12) GO TO 60
IF (DABS((V-VERGL)/VUNEND).LT.1.0E-12) GO TO 12

```

```

GO TO 2
60 IF (DABS(V-VERGL).LT.1.0E-12) GO TO 12
WRITE (NOUT,1007)
1007 FORMAT(' GENAUIGKEIT NACH 10000 ITERATIONSSCHRITTEN NOCH NICHT ERR
1EICHT')
2 CONTINUE
12 $V(I)=V
1 CONTINUE
RETURN

```

C
CPLATTE
C

```

110 ALPHA=P*RE*R1*R1*0.5
GAMMA=-0.5*P*RE
DO 101 I=1,NR
R=$R(I)
IF (R.LT.0.0) R=DABS(R)
V=ALPHA+GAMMA*R*R
DO 102 K=1,10000
IF (K.GT.5000) GO TO 130
IF (K.LE.KMAX) GO TO 132
KMAX=K
QKI(K)=(K-0.5)*PI/R1
QK=QKI(K)
QKR1=QK*R1
CKI(K)=((-ALPHA+V0)*DSIN(QKR1)-GAMMA*(R1*R1-2.0/(QK*QK))*DSIN(QKR1
1))/(0.5*QK*(R1+(1.0/(2.0*QK))*DSIN(2.0*QKR1)))
CK=CKI(K)
GO TO 131
132 QK=QKI(K)
CK=CKI(K)
GO TO 131
130 QK=(K-0.5)*PI/R1
QKR1=QK*R1
CK=((-ALPHA+V0)*DSIN(QKR1)-GAMMA*(R1*R1-2.0/(QK*QK))*DSIN(QKR1
1))/(0.5*QK*(R1+(1.0/(2.0*QK))*DSIN(2.0*QKR1)))
131 VERGL=V
EXPO=-QK*QK*T/RE
IF (EXPO.LE.-100.0) GO TO 133
EXPT=DEXP(EXPO)*DCOS(QK*R)
GO TO 134
133 EXPT=0.0
134 V=V+CK*EXPT
IF (ALPHA.LT.1.0E-12) GO TO 160
IF (DABS((V-VERGL)/ALPHA).LT.1.0E-12) GO TO 170
GO TO 102
160 IF (DABS(V-VERGL).LT.1.0E-12) GO TO 170
WRITE (NOUT,1007)
102 CONTINUE
170 $V(I)=V
101 CONTINUE
IF (IGEO.NE.3) RETURN
DO 178 I=1,NR
178 $R(I)=$R(I)+ROALT+R1
RO=ROALT
R1=R1ALT
RETURN

```


C
CZYLINDER

```
C
200 ALPHA=P*RE*R1*R1/4.0000000
    GAMMA=-0.25*P*RE
    ASSIGN 211 TO MARCA
    ASSIGN 210 TO MARKE
    DO 201 I=1,NR
    R=$R(I)
    IF (R.LT.0.0) R=-R
    V=ALPHA+GAMMA*R*R
    DO 202 K=1,10000
    IF (K.GT.5000) GO TO 230
    IF (K.LE.KMAX) GO TO 232
    KMAX=K
    XO=PI*(K-0.25)
    X01=XO
    GO TO MARCA,(211,221)
211 CALL NEWTON (XO,BEFJO,DBES)
    IF (XO.LT.1.0E-12) GO TO 241
    IF (DABS((XO-X01)/XO).GT.1.0E-12) GO TO 221
    GO TO 251
241 IF (DABS(XO-X01).GT.1.0E-12) GO TO 221
251 ASSIGN 220 TO MARKE
    ASSIGN 221 TO MARCA
221 QKI(K)=XO/R1
    QK=QKI(K)
    CKI(K)=(-P*RE/(QK*QK)+V0)/(R1/2.0*QK*BEFJ1(QK*R1))
    CK=CKI(K)
    GO TO 231
232 QK=QKI(K)
    CK=CKI(K)
    GO TO 231
230 XO=PI*(K-0.25)
    X01=XO
    GO TO MARKE,(210,220)
210 CALL NEWTON (XO,BEFJO,DBES)
    IF (XO.LT.1.0E-12) GO TO 240
    IF (DABS((XO-X01)/XO).GT.1.0E-12) GO TO 220
    GO TO 250
240 IF (DABS(XO-X01).GT.1.0E-12) GO TO 220
250 ASSIGN 220 TO MARKE
220 QK=XO/R1
    CK=(-P*RE/(QK*QK)+V0)/(R1/2.0*QK*BEFJ1(QK*R1))
231 VERGL=V
    EXPO=-QK*QK*T/RE
    IF (EXPO.LE.-100.0) GO TO 233
    EXPT=DEXP(EXPO)*BEFJO(QK*R)
    GO TO 234
233 EXPT=0.0
234 V=V+CK*EXPT
    IF (ALPHA.LT.1.0E-12) GO TO 260
    IF (DABS((V-VERGL)/ALPHA).LT.1.0E-12) GO TO 270
    GO TO 202
260 IF (DABS(V-VERGL).LT.1.0E-12) GO TO 270
    WRITE (NOUT,1007)
202 CONTINUE
```

270 \$V(I)=V
201 CONTINUE
999 RETURN
· END

```

SUBROUTINE ANPRIN (T,NR,R,RO,R1,P,RE,VO,IGEO,V)
C
CN ANPRIN
C
CA ENDERLE
C
CD 15.7.71
C
CB BERECHNUNG EINER ANLAUFSTROEMUNG MITTELS DER SUBROUTINE ANLAUF
CB UND AUSDRUCKEN DER ERGEBNISSE
C
CP PARAMETER WIE IN ANLAUF
C
DIMENSION R(1),V(1)
DATA NOUT/6/
CALL ANLAUF(T,NR,R,RO,R1,P,RE,VO,IGEO,V)
WRITE (NOUT,1002)
1002 FORMAT ('1',13X,'T',4X,'NR',15X,'RO',15X,'R1',16X,'P',15X,
1'RE',15X,'VO',3X,'IGEO')
WRITE (NOUT,1000) T,NR,RO,R1,P,RE,VO,IGEO
1000 FORMAT ('0',E14.7,3X,I3,3X,5(E14.7,3X),I4)
WRITE (NOUT,1003)
1003 FORMAT ('0',2X,'I',13X,'R(I)',13X,'V(I)')
WRITE (NOUT,1004)
1004 FORMAT (' ')
DO 4 I=1,NR
4 WRITE (NOUT,1001) I,R(I),V(I)
1001 FORMAT (' ',I3,3X,2(E14.7,3X))
RETURN
END

```

```

SUBROUTINE ANPLOT(T,NT,RO,R1,P,RE,VO,IGEO,IDPLOT)
C
CN ANPLOT
C
CA ENDERLE
C
CD 15.7.71
C
CB PROGRAMM ZUM PLOTTEN DES GESCHWINDIGKEITSPROFILES EINER ANLAUF-
CB STROEMUNG IN PLATTEN-, ZYLINDER- UND HOHLZYLINDERGEOMETRIE
C
CP T VEKTOR DER ZEITPUNKTE, ZU DENEN DAS GESCHWINDIGKEITS-
C PROFIL ERRECHNET UND GEZEICHNET WERDEN SOLL
CP NT ANZAHL DER ZEITPUNKTE, NT KLEINER GLEICH 16
CP RO INNERER RANDFLAECHENRADIUS (NUR ABGEFRAGT BEI HOHLZYLINDER)
CP R1 AUESSENER RANDFLAECHENRADIUS
CP P DRUCK=-DELTA(DRUCK)/DELTA(Z)
CP RE REYNOLDS-ZAHL
CP VO ANFANGSWERT DER GESCHWINDIGKEIT
CP IGEO = 1 PLATTEN /
CP IGEO = 2 ZYLINDER /GEOMETRIE-PARAMETER
CP IGEO = 3 HOHLZYLINDER/
CP IDPLOT NUMMER, DIE DIE ZEICHNUNG ERHALTEN SOLL, BEIM ERSTEN
C AUFRUF VON ANPLOT IN EINEM JOB MUSS IDPLOT=1 SEIN
C
CF PLOTTAPE PLOTDATEI
C
CS ANLAUF BERECHNET DIE GESCHWINDIGKEITSWERTE
IMPLICIT REAL*8($)
C
CE MELDUNGEN WIE IN SUBROUTINE ANLAUF
CE FUER NT > 16 ERFOLGT MELDUNG UND NT WIRD AUF 16 GESETZT
C
DIMENSION T(1),R(23),V(23)
DATA NOUT/6/
DIMENSION IBUF(1000)
INTEGER NPTEX(03),NRETEX(03),NROTEX(03),NRITEX(03),NVOTEX(03)
DIMENSION XB(4),YB(4),NTEXT3(4)
INTEGER NTEXT1(4)/'IGEO', '= ZY', 'LIND', 'ER..'/,NTEXT2(4)/'IGEO',
1 '= RI', 'NG ', ' ..'/,NTEXT4(4)/'IGEO', '= PL', 'ATTE', 'N ..'/
INTEGER*2 NTTEX(2),IH2/'N=' /
INTEGER BLANK/' /
INTEGER IDTEXT(3)/'IDPL', 'OT =', ' /
CALL CONVX(IDPLOT,NPTEX,'I4 ')
IDTEXT(3)=NPTEX(1)
IF (IDPLOT.NE.1) GO TO 111
CALL CALCIN(IBUF,1000,1)
CALL PLOT(0.0,2.0,-3)
111 DO 131 I=1,3
NPTEX(I)=BLANK
NRETEX(I)=BLANK
NROTEX(I)=BLANK
NRITEX(I)=BLANK
131 NVOTEX(I)=BLANK
IF (NT.LE.16) GO TO 3
NT=16
WRITE (NOUT,1000)

```

```

1000 FORMAT (' NT GROESSER 16')
3 IF (IGEO.NE.3) R0=-R1
DO 1 I=1,21
1 R(I)=R0+(R1-R0)*(I-1.0)/20.0
DO 2 J=1,NT
CALL ANLAUF(T(J),21,R,RO,R1,P,RE,VO,IGEO,V)
IF (J.NE.1) GO TO 100
$P=P
$RE=RE
$RO=RO
$R1=R1
IF(IGEO.EQ.1) GO TO 101
IF(IGEO.EQ.2) GO TO 102
VMAX=(0.25*$P*$RE)*(( $RO*$RO-$R1*$R1)/DLOG($RO/$R1)*DLOG(0.5*($RO
1+$R1))
1+($R1*$R1*DLOG($RO)-$RO*$RO*DLOG($R1))/DLOG($RO/$R1)-(0.5*($R1+$RO
3))*2)
GO TO 103
101 VMAX= 0.5*$P*$RE*$R1*$R1
GO TO 103
102 VMAX=0.25*$P*$RE*$R1*$R1
103 VMAX=AMAX1(VMAX,VO)
POT=ALOG10(VMAX)
IF (POT.LT.0.0) POT=POT-1.0
IPOT=POT
RUND=10.0**IPOT
GANZ=VMAX/RUND+1
IGANZ=GANZ
YMAX=IGANZ*RUND
EINH=YMAX/10.0
YMIN=0.0
XMIN=RO
XMAX=R1
EINHX=(XMAX-XMIN)/10.0
IF (IGEO.EQ.1) GO TO 121
IF (IGEO.EQ.2) GO TO 122
IF (IGEO.EQ.3) GO TO 123
121 DO 124 K=1,4
124 NTEXT3(K)=NTEXT4(K)
XPAGE=4.0
GO TO 127
122 DO 125 K=1,4
125 NTEXT3(K)=NTEXT1(K)
XPAGE=4.0
GO TO 127
123 DO 126 K=1,4
126 NTEXT3(K)=NTEXT2(K)
XPAGE=0.0
127 CALL CONVX(P,NPTEX,'E12.5 ')
CALL CONVX(RE,NRETEX,'E12.5 ')
CALL CONVX(RO,NROTEX,'E12.5 ')
CALL CONVX(R1,NRITEX,'E12.5 ')
CALL CONVX(VO,NVOTEX,'E12.5 ')
DELTXV=(XMAX-XMIN)/8.0
DELTYV=(YMAX-YMIN)/11.0
CALL AXIS(0.0,0.0,'R',-1,8.0,0.0,XMIN,DELTXV)
CALL AXIS(XPAGE,0.0,'V(R)',4,11.0,90.0,0.0,DELTYV)

```

```

CALL SYMBOL(6.0,10.8,0.14,NTTEXT3,0.0,14)
CALL SYMBOL(6.0,10.3,0.14,4HP =,0.0,4)
CALL SYMBOL(6.0,09.8,0.14,4HRE =,0.0,4)
CALL SYMBOL(6.0,09.3,0.14,4HRO =,0.0,4)
CALL SYMBOL(6.0,08.8,0.14,4HR1 =,0.0,4)
CALL SYMBOL(6.0,08.3,0.14,4HVO =,0.0,4)
CALL SYMBOL(6.7,10.3,0.14,NPTEX,0.0,12)
CALL SYMBOL(6.7,09.8,0.14,NRETEX,0.0,12)
CALL SYMBOL(6.7,09.3,0.14,NROTEX,0.0,12)
CALL SYMBOL(6.7,08.8,0.14,NR1TEX,0.0,12)
CALL SYMBOL(6.7,08.3,0.14,NVOTEX,0.0,12)
CALL SYMBOL(1.6,-1.0,0.4,IDTEXT,0.0,12)
100 JP=J
R(22)=XMIN
R(23)=DELTXV
V(22)=0.0
V(23)=DELTYV
IF (IGED.NE.3.AND.JP.GE.8) JP=JP+3
VJP=V(JP+1)
XB(1)=XMIN+JP*0.5*EINHX
YB(1)=VJP
XB(2)=XB(1)
YB(2)=YB(1)+0.05*EINH
129 CALL CONVX(J,NTTEX,'I2 ')
NTTEX(2)=NTTEX(1)
NTTEX(1)=IH2
CALL LINE(R,V,21,1,0,0)
XB(3)=XMIN
XB(4)=DELTXV
YB(3)=0.0
YB(4)=DELTYV
CALL LINE(XB,YB,2,1,0,0)
CALL WHERE(RXPAGE,RYPAGE,RFAKT)
CALL SYMBOL(RXPAGE,RYPAGE,0.10,NTTEX,0.0,4)
2 CONTINUE
IF (IDPLOT.NE.999) GO TO 998
CALL PLOTTC(12.0,0.0,999)
GO TO 999
998 CALL PLOTTC(12.0,0.0,-3)
999 CONTINUE
RETURN
END

```

```
SUBROUTINE NEWTON (X0,F,DF)
  IMPLICIT REAL*8 (A-H,O-Z)
  XN1=X0
  I=0
1  XN=XN1-F(XN1)/DF(XN1)
  IF (DABS(XN).LT.1.0E-12) GO TO 4
  IF (DABS((XN1-XN)/XN).LE.1.0E-12) GO TO 2
  GO TO 5
4  IF (DABS(XN1-XN).LT.1.0E-12) GO TO 2
5  I=I+1
  IF (I-1000) 3,2,3
3  XN1=XN
  GO TO 1
2  X0=XN
  RETURN
  END
```

```
DOUBLE PRECISION FUNCTION F(X)
IMPLICIT REAL*8 (A-H,O-Z)
COMMON Y
F=BEFJO(X)*BEFYO(X*Y)-BEFYO(X) *BEFJO(Y*X)
RETURN
END
```

```
DOUBLE PRECISION FUNCTION DF(X)
IMPLICIT REAL*8 (A-H,O-Z)
COMMON Y
DF=BEFJ1(Y*X)*BEFYO(X)*Y+BEFJO(Y*X)*BEFY1(X)-BEFJ1(X)*BEFYO(X*Y)
1-BEFJO(X)*BEFY1(X*Y)*Y
RETURN
END
```

```
DOUBLE PRECISION FUNCTION DBES(X)
REAL*8 X
DBES=-BEFJ1(X)
RETURN
END
```



```

DOUBLE PRECISION FUNCTION BEFJO(X)
REAL*8 X,XX3,X3X,TETA
IF (X.EQ.0.0) GO TO 4
IF (X.LT.0.0) GO TO 3
1 FORMAT (' X= ',E13.6,' NEGATIV IN JO')
IF (X.GT.3.0) GO TO 2
XX3=X*X/9.0000000
BEFJO= (((((0.0002100*XX3-0.0039444)*XX3+0.0444479)*XX3-0.3163866
1)*XX3+1.2656208)*XX3-2.2499997)*XX3+1
RETURN
2 X3X=3.0/X
TETA= (((((0.00013558*X3X-0.00029333)*X3X-0.00054125)*X3X+0.002625
173)*X3X-0.00003954)*X3X-0.04166397)*X3X-0.78539816+X
BEFJO=1.0000000/DSQRT(X)* (((((0.00014476*X3X-0.00072805)*X3X+
10.00137237)*X3X-0.00009512)*X3X-0.00552740)*X3X-0.00000077)*X3X
2+0.79788456)*DCOS(TETA)
RETURN
3 WRITE (6,1) X
4 BEFJO=1.0
RETURN
END

```

```

DOUBLE PRECISION FUNCTION BEFJ1(X)
REAL*8 X,XX3,TETA,X3X
IF (X.EQ.0.0) GO TO 4
IF (X.LT.0.0) GO TO 3
1 FORMAT (' X= ',E13.6,' NEGATIV IN J1')
IF (X.GT.3.0) GO TO 2
XX3=X*X/9.0000000
BEFJ1=X* (((((0.00001109*XX3-.00031761)*XX3+.00443319)*XX3
1-.03954289)*XX3+0.21093573)*XX3-0.56249985)*XX3+0.5)
RETURN
2 X3X=3.0/X
TETA=(((((-0.00029166*X3X+0.00079824)*X3X+0.00074348)*X3X
1-0.00637879)*X3X+0.00005650)*X3X+0.12499612)*X3X-2.35619449+X
BEFJ1=1.0/DSQRT(X)*(((((-0.00020033*X3X+0.00113653)*X3X
1-0.00294511)*X3X+0.00017105)*X3X+0.01659667)*X3X+0.00000156)*X3X
2+0.79788456)*DCOS(TETA)
RETURN
3 WRITE (6,1) X
4 BEFJ1=0.0
RETURN
END

```

```

DOUBLE PRECISION FUNCTION BEFY0(X)
REAL*8 X,XX3,PI,TETA,X3X
IF (X.LE.0.0) GO TO 3
IF (X.GT.3.0) GO TO 2
1 FORMAT (' X= ',E13.6,' NULL ODER NEGATIV IN Y0')
XX3=X*X/9.0000000
PI=3.141592653589793238
BEFY0=(2.0000000/PI)*DLOG(X/2.0000000)*BEFJ0(X)+((((
1-0.00024846*XX3+0.00427916)*XX3-0.04261214)*XX3+0.25300117)*XX3
2-0.74350384)*XX3+0.60559366)*XX3+0.36746691)
RETURN
2 X3X=3.0000000/X
TETA= (((((0.00013558*X3X-0.00029333)*X3X-0.00054125)*X3X
1+0.00262573)*X3X-0.00003954)*X3X-0.04166397)*X3X-0.78539816+X
BEFY0=1.0000000/DSQRT(X)*(((
1+0.00137237)*X3X-0.00009512)*X3X-0.00552740)*X3X-0.00000077)*X3X
2+0.79788456)*DSIN(TETA)
RETURN
3 WRITE (6,1) X
RETURN
END

```

```

DOUBLE PRECISION FUNCTION BEFY1(X)
REAL*8 X,XX3,PI,X3X,TETA
IF (X.LE.0.0) GO TO 3
1 FORMAT (' X= ',E13.6,' NEGATIV ODER 0 IN Y1')
IF (X.GT.3.0) GO TO 2
XX3=X*X/9.00000000
PI=3.141592653589793238
BEFY1=(1.0000000/X)*(((((0.0027873*XX3-0.0400976)*XX3
1+0.31239)*XX3-1.3164827)*XX3+2.1682709)*XX3+0.22120910)*XX3
2-0.63661980+((2.00000000/PI)*X*DLOG(X/2.0000000)*BESJ1(X))
RETURN
2 X3X=3.00000000/X
TETA= (((((-0.00029166*X3X+0.00079824)*X3X+0.00074348)*X3X
1-0.00637879)*X3X+0.00005650)*X3X+0.12499612)*X3X-2.35619449+X
BEFY1=(1.0000000/DSQRT(X))*((( (-0.00020033*X3X+0.00113653)
1*X3X-0.00249511)*X3X+0.00017105)*X3X+0.01659667)*X3X+0.00000156)
2*X3X+0.79788456)*DSIN(TETA)
RETURN
3 WRITE (6,1) X
RETURN
END

```