

Technische Universität Ilmenau
Fakultät für Mathematik
und Naturwissenschaften
Institut für Mathematik

Postfach 10 0565
98684 Ilmenau
Germany
Tel.: 03677/692652
Fax: 03677/691241
Telex: 33 84 23 tuil d.
email: W.Neundorf@mathematik.tu-ilmenau.de

Preprint No. M 11/98

Gewöhnliche Differentialgleichungen

Beispiele, Modelle, Verfahren, Software

Werner Neundorf

Mai 1998

‡MSC (1991): 58F40, 65-01, 65-04, 65L05, 68Q25

Viele praktische Beispiele führen auf die Anwendung von Differentialgleichungen. Aus der Literatur wurden zahlreiche Modelle zitiert und erläutert. Ihre Analyse fällt in das Gebiet der *Theorie der gewöhnlichen Differentialgleichungen* und findet hier keinen Niederschlag. Zur Numerik der GDGL wird bis auf wenige Verfahrensvorschläge auf die einschlägige Literatur verwiesen. Zusätzlich findet man einige Hinweise auf entsprechende Software.

Many practical examples lead to the application of differential equations. Starting from the literature, it goes on to present and describe a lot of models. Their most theoretical analysis comes from the *Theory of Ordinary Differential Equations* and was not involved here. The techniques for the numerical solution of ODE are present in some problems, but the reader finds a large presentation of numerics in many textbooks. In addition, there are given some software hints.

Key words: applications for ordinary differential equations, tutorial aspects,
analysis of algorithms, programs

MSC (1991): 58F40, 65-01, 65-04, 65L05, 68Q25

1 Einleitung

Teile dieser Arbeit basieren auf einem Skriptum, daß ich in der Lehrveranstaltung *Numerik der gewöhnlichen Differentialgleichungen* für die Studenten der Mathematik im 5. Semester sowie der Allgemeinen und Theoretischen Elektrotechnik einbezogen habe. Dabei sollte ein Überblick zur Darstellung und Simulation verschiedenartiger Sachverhalte durch Differentialgleichungen gegeben werden. Entsprechende praktische Anwendungen auf den Gebieten Technik, Physik, Chemie, Mathematik, Biologie, Ökonomie, Ökologie und Medizin sind aus meistens einschlägiger Literatur sowie Sammlungen von Angehörigen des Fachgebiets Numerische Mathematik und Informationsverarbeitung zusammengetragen worden.

Die Aufgabenstellungen bieten auch eine breite Auswahl für mögliche Beleg- und Praktikumsarbeiten mit programmtechnischer Umsetzung.

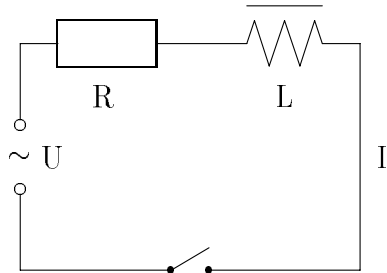
Bezüglich der Theorie der gewöhnlichen Differentialgleichungen möchte ich insbesondere auf das Skriptum von J.Knobloch und J.Steigenberger zur Vorlesung *Gewöhnliche Differentialgleichungen* für Studenten der Mathematik im 4. Semester im Rahmen der Grundausbildung in Analysis verweisen.

2 Differentialgleichungen in der Praxis

2.1 Elektrotechnik

1. Elektrische Schaltung [SCHWETLICK, KRETZSCHMAR]

Aus der Beziehung für den Spannungsabfall berechnet man die Stromstärke.



$$U(t) = U_0 \sin(\omega_0 t), \quad t_0 = 0 \leq t \leq T$$

$$I(t_0) = 0$$

$$\text{DGL: } U(t) = \underbrace{R I(t)}_{\text{Ohmscher}} + \underbrace{L \frac{dI(t)}{dt}}_{\text{Induktiver}}$$

Spannungsabfall

Allgemeine Lösung für die Stromstärke:

$$I(t) = e^{-\int \frac{R}{L} dt} \left[c_0 + \int \frac{R}{L} e^{\int \frac{R}{L} d\tau} dt \right], \quad c_0 \text{ aus Anfangsbedingung ermitteln.}$$

Lösung im speziellen Fall:

$$R = R_0 = \text{const}, \quad L = L_0 = \text{const}, \quad U(t) = U_0 \sin(\omega_0 t), \quad U_0 = \text{const}, \quad t_0 = 0$$

$$I(t) = \frac{U_0 \omega_0 L_0}{R_0^2 + \omega_0^2 L_0^2} e^{-\frac{R_0}{L_0} t} + \frac{U_0}{\sqrt{R_0^2 + \omega_0^2 L_0^2}} \sin(\omega_0 t - \alpha) \quad \text{mit } \alpha = \arctan \frac{\omega_0 L_0}{R_0}$$

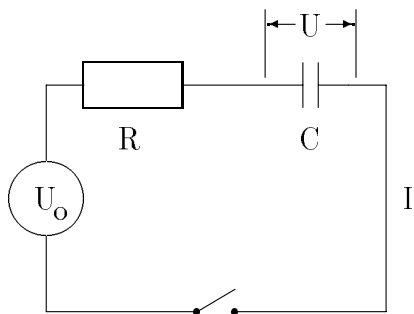
2. Einschalten eines Stromkreises [PHILIPPOW]

Anwendung der Kirchhoffschen Gesetze.

$$\text{DGL: } \frac{dU}{dt} + \frac{1}{RC} U = \frac{1}{RC} U_0, \quad I(t) = C \frac{dU(t)}{dt}$$

Transformation der Variablen liefert eine DGL 1.Ordnung

$$\dot{x} + f x = g(\tau), \quad \tau \text{ transformierte Zeit}$$



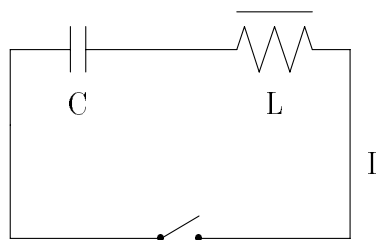
Hängen C und/oder R auch vom Strom I ab (nichtlineare Kapazitäten oder Widerstände), so gilt

$$\dot{x} + f(\dot{x})x = g(\dot{x}, \tau).$$

Diese DGL ist i.a. nicht geschlossen integrierbar.

3. Nichtlinearer ungedämpfter Schwingkreis [PHILIPPOW]

Man betrachtet man den ungedämpften Schwingkreis mit nichtlinearer Induktivität.



Wenn der Zusammenhang zwischen Strom $I(t)$ und dem Fluß $\psi(t)$ die Kennlinie $I = a\psi + b\psi^3$ besitzt, so erhält man die DGL

$$\frac{d^2\psi}{dt^2} + \frac{a}{C}\psi + \frac{b}{C}\psi^3 = 0.$$

Nach Normierung mit $\Psi = \max \psi(t) = \psi(0)$ durch

$$x(t) = \frac{\psi(t)}{\Psi}, \quad \tau = \sqrt{\frac{a}{C}}t, \quad \lambda = \Psi^2 \frac{a}{b}$$

erhält man die DGL 2.Ordnung

$$\ddot{x} + x + \lambda x^3 = 0,$$

die nicht geschlossen lösbar ist.

Das Anlegen einer sinusförmigen Erregerschwingung an diesen Schwingkreis liefert eine erzwungene Schwingung mit der normierten DGL (Duffingsche DGL)

$$\ddot{x} + x + \lambda x^3 = k \cos(\alpha\tau) .$$

4. Relaxationsschwingungen [PHILIPPOW]

Zahlreiche Probleme der nichtlinearen Elektrotechnik (ungedämpfte Schwingungen, Vibrator, Multivibrator) führen zu der normierten DGL 2.Ordnung

$$\ddot{y} - \varepsilon (1 - y^2) \dot{y} + y = 0, \quad t \geq 0, \quad \varepsilon \geq 0,$$

(Van der Pol DGL).

Für $\varepsilon = 0$ liefert sie bei Anfangsbedingungen y_0, \dot{y}_0 an der Stelle $t = 0$ die harmonische Schwingung $y(t) = \dot{y}_0 \sin t + y_0 \cos t$ mit der Periode $T = 2\pi$.

Wächst $\varepsilon > 0$ an, so verzerrt sich die Schwingung immer mehr und die Periode vergrößert sich.

Bei $\varepsilon \gg 1$ (z.B. $\varepsilon = 100, 1000$) beschreibt die DGL Relaxationsschwingungen. Dabei wechseln sich Zeiten sehr langsamer Bewegung mit solchen sehr schneller Bewegung ab.

Der Parameter ε legt auch die Einschwingdauer fest.

Bei erregten Schwingkreisen steht ein zusätzlicher Term auf der rechten Seite der DGL.

5. Mehrmaschige elektrische Netzwerke [PHILIPPOW, UHLMANN]

Das zweimaschige gekoppelte Netzwerk für die normierten Variablen $x_1(t), x_2(t)$ hat die Form

$$\begin{aligned} \ddot{x}_1 + \alpha_1 \dot{x}_1 + \varepsilon_1 (1 - x_2^2) \dot{x}_2 + x_1 &= 0 \\ \ddot{x}_2 + \alpha_2 \dot{x}_2 + \varepsilon_2 (1 - x_1^2) \dot{x}_1 + x_2 &= 0 \end{aligned}$$

Eine geschlossene Lösung ist nicht möglich. Insbesondere sind Lösungen für große Werte von $|\varepsilon_1|, |\varepsilon_2|$ gesucht.

6. Dynamische Kennlinie des Asynchronmotors [GOERING]

Bei Asynchronmotoren ist die Drehzahl stark von der Belastung abhängig. Für schwankende Motorleistung ist es zur Berechnung von Torsionsschwingungen notwendig, den Zusammenhang zwischen Motormoment und Winkelgeschwindigkeit zu bestimmen.

Mit normierten Größen $u(t), v(t)$, t dimensionslose Zeit, erhält man das nicht geschlossen lösbare DGL-System 1.Ordnung

$$\begin{aligned} \beta \dot{u} &= x - u + xv, & u(0) &= u_0, & \beta &\in \mathbb{R}^+, & x &= x(t) \\ \beta \dot{v} &= -xu - v, & v(0) &= v_0. \end{aligned}$$

7. Weitere Beispiele

(a) Spannungsstabilisatorschaltung [PHILIPPOW]

Ferroresonanzstabilisator, $y = y(t)$

$$\ddot{y} - \delta \dot{y} + ay + by^9 + \alpha y = -\Gamma \cos(t + \phi)$$

(b) Frequenzteilerschaltung

$$\ddot{y} - \varepsilon(1 - y^2 - \dot{y}^2)\dot{y} + y + 2By\dot{y} = B \sin(2t)$$

(c) System der Drei-Wellen-Wechselwirkung

$$\dot{a}_0 = a_0 + 2a_0a_1^2 \sin \vartheta$$

$$\dot{a}_1 = -a_1 - a_0^2a_1 \sin \vartheta$$

$$\dot{a}_2 = 2(a_1^2 - a_0^2) - 2\delta + 2(2a_1^2 - a_0^2) \cos \vartheta$$

(d) Rössler-System

$$\dot{X} = -Y - Z$$

$$\dot{Y} = X + aY$$

$$\dot{Z} = bX - cZ + XZ$$

2.2 Mechanik

1. Vibration von Membranen und Stäben [HOLMES, SEYDEL]

Diese führen auf DGL vom Duffingschen Typ

$$\ddot{y} - \varepsilon \dot{y} - \beta y + \alpha y^2 = f \cos(\omega t)$$

Sie stellt eine erste Näherung für die Schwingung dar und wird für verschiedene Parameterkonstellationen $\varepsilon, \beta, \alpha, f, \omega$ analysiert.

2. Schwingungen von Eisenbahnwagen [KUBICEK, MAREK]

$$I\ddot{x} + 2lf_T \left(\frac{l\dot{x}}{v} + \frac{r_1 - r_2}{2r_0} \right) + K_x x = 0$$

$$M\ddot{y} + 2f_L \left(\frac{\dot{y}}{v} - x \right) K_y y + F_g(y - \delta) = 0$$

3. Bewegungen eines Himmelskörpers [SHAMPINE, GORDON]

Die Bewegung einer Rakete oder eines Satelliten im Schwerfeld zweier großer Himmelskörper führt auf ein restringiertes Dreikörperproblem.

Bsp.: Erde - Mond - Rakete

Sonne - Erde - Rakete

Sonne - Saturn - Saturnmond

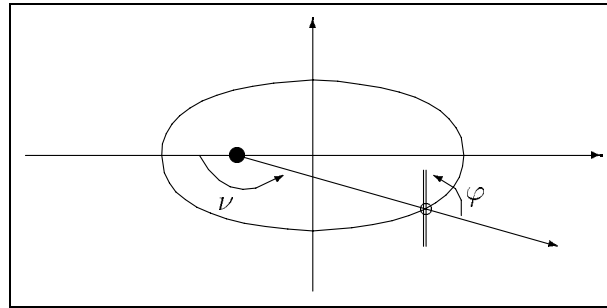
$$y_1'' = 2y_2' + y_1 - \frac{\mu^*(y_1 + \mu)}{r_1^3} - \frac{\mu(y_1 - \mu^*)}{r_2^3}$$

$$y_2'' = -2y_1' + y_2 - \frac{\mu^* y_2}{r_1^3} - \frac{\mu y_2}{r_2^3}$$

mit $\mu^* = 1 - \mu$, $0 < \mu < 1$ und $r_1 = \sqrt{(y_1 + \mu)^2 + y_2^2}$, $r_2 = \sqrt{(y_1 - \mu)^2 + y_2^2}$.

4. Orientierung von Telekommunikationssatelliten [TROGER]

Dabei geht es um die positive Orientierung der Antenne des Satelliten auf seiner elliptischen Umlaufbahn durch die Vorgabe einer geeigneten Anfangsorientierung.



$$(1 + e \cos \nu) \frac{d^2 \varphi}{d\nu^2} - 2e \sin \nu \frac{d\varphi}{d\nu} + \alpha \sin \varphi \cos \varphi = 2e \sin \nu$$

e Exzentrizität, α Konstante zur Beschreibung der Massenverteilung,
 ν Umlaufwinkel, φ gesuchte Abweichung vom Leitstrahl

5. Pendel mit Reibung [ARNOLD]

$\ddot{x} + k\dot{x} + x = 0$, $x(0)$, $\dot{x}(0)$ gegeben, $k \geq 0$ Reibungskoeffizient.

Überführung in ein DGL-System

$$\begin{cases} \dot{x} = y, & x(0) \text{ gegeben} \\ \dot{y} = \ddot{x} = -k\dot{x} - x = -x - ky, & y(0) = \dot{x}(0) \text{ gegeben} \end{cases}$$

Notation in Matrixform

$$\dot{X} = AX = \begin{pmatrix} 0 & 1 \\ -1 & -k \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}, \quad \text{trace}(A) = -k, \quad \det(A) = 1$$

Eigenwertproblem zur Matrix A

$$\lambda(A) : 0 = \det(A - \lambda I) = \lambda^2 + k\lambda + 1.$$

Falls $|k| > 2$, dann λ_i reell, negativ und verschieden.

Sei

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

Eigenvektoren

$$x_1 = \begin{pmatrix} -3 \\ 1 \end{pmatrix}, \quad x_2 = \begin{pmatrix} 1 \\ -3 \end{pmatrix}, \quad U = (x_1, x_2) \text{ regulär, } AU = U\Lambda$$

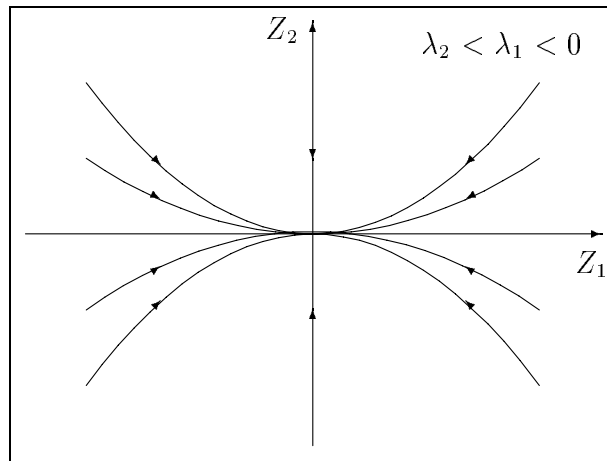
Transformation auf Diagonalform möglich.

$$\begin{aligned}
 X &= UZ \\
 \dot{X} &= U\dot{Z} = AX = AUZ \\
 U\dot{Z} &= U\Lambda Z \\
 \dot{Z} &= \Lambda Z \\
 \dot{Z}_1 &= \lambda_1 Z_1, \text{ entkoppeltes System} \\
 \dot{Z}_2 &= \lambda_2 Z_2 \\
 Z_1(t) &= e^{\lambda_1 t} Z_1(0), \text{ Lösungen} \\
 Z_2(t) &= e^{\lambda_2 t} Z_2(0)
 \end{aligned}$$

Beispiel

$$X_0 = (x(0), y(0))^T = (2, 2)^T, \quad k = \frac{10}{3}, \quad \lambda_1 = -\frac{1}{3}, \quad \lambda_2 = -3$$

1. Darstellung der Lösungen als Phasenkurven im Raum der EV

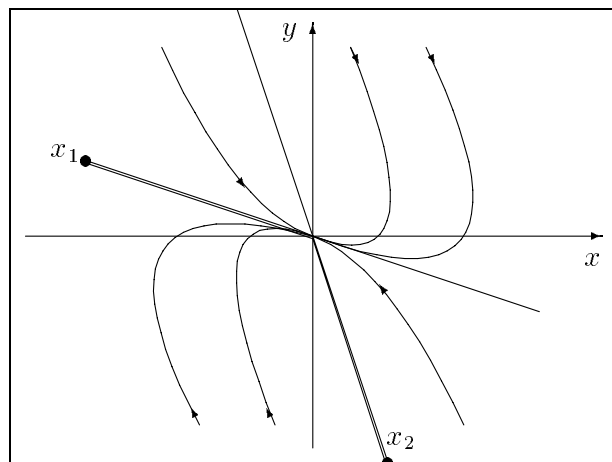


2. Darstellung der Lösungen als Phasenkurven im Raum (x_1, x_2)

$$\begin{aligned}
 U^{-1} &= \frac{1}{8} \begin{pmatrix} -3 & -1 \\ -1 & -3 \end{pmatrix}, \quad Z_0 = U^{-1}X_0 = \begin{pmatrix} -1 \\ -1 \end{pmatrix} \\
 X &= UZ = \begin{pmatrix} -3 & 1 \\ 1 & -3 \end{pmatrix} \begin{pmatrix} e^{\lambda_1 t} Z_1(0) \\ e^{\lambda_2 t} Z_2(0) \end{pmatrix} = \begin{pmatrix} 3e^{-t/3} - e^{-3t} \\ -e^{-t/3} + 3e^{-3t} \end{pmatrix}
 \end{aligned}$$

Bei gegebener Anfangsauslenkung kommt es durch den Anstoß zu einem kurzzeitigen Überschwingen des Pendels. Danach wird die Pendelbewegung wegen der Reibung jedoch stark gedämpft.

Die Lösung $x(t) = 3e^{-t/3} - e^{-3t}$ strebt mit wachsender Zeit asymptotisch gegen Null.



2.3 Thermodynamik, Chemie

1. Lorenzmodell der Wärmeströmung in einer Flüssigkeitsschicht [LORENZ, KUBICEK, STEEB]

Eine Flüssigkeitsschicht wird von einer Seite erwärmt und von der gegenüberliegenden Seite gekühlt. Die dabei resultierende konvektive Bewegung ist von Interesse.

Das DGL-System entsteht aus den Navier-Stokes-Gleichungen.

y_1 konvektive Durchmischung

y_2 horizontale Temperaturveränderung

y_3 vertikale Temperaturveränderung

$$y_1' = -\sigma y_1 + \sigma y_2, \quad y_1(0) = y_{10}$$

$$y_2' = -y_1 y_3 + r y_1 - y_2, \quad y_2(0) = y_{20}$$

$$y_3' = y_1 y_2 - b y_3, \quad y_3(0) = y_{30}$$

σ Prandtl-Zahl, r reduzierte Rayleigh-Zahl, b Wellenzahl sind positive Parameter.

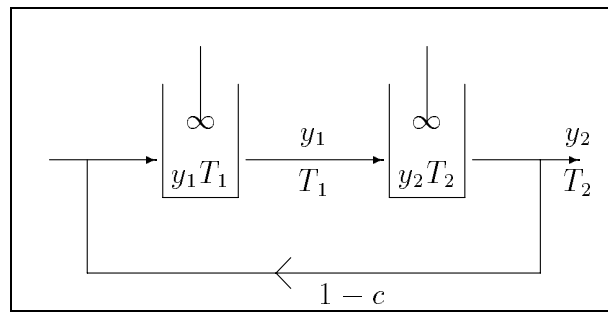
2. Schwingungen im Wärmeaustauscher [TROGER]

Durch Wärmeunterschiede in einem solchen Durchlaufsystem können Röhren unter Spannung, andere unter Druck stehen. Dadurch entstehen auch Schwingungen.

$$\ddot{y} + \varepsilon_1 \dot{y} + \varepsilon_2 y + y^3 + \dot{y}^3 = 0$$

3. Chemische Reaktoren [KUBICEK, MAREK]

Beschreibung einer Kaskade von 2 chemischen Tankreaktoren mit Rührwerk und Rückkopplung liefert ein System von 4 DGL 1.Ordnung.



y_i Konzentration, T_i Temperatur, $0 < c \leq 1$

2.4 Physik

1. Freier Fall

Angenommen, das AWP

$$y' = a\sqrt{y}, \quad y = y(t), \quad y(0) = c, \quad a = \text{const} > 0, \quad c = \text{const} \geq 0,$$

beschreibt den freien Fall eines Körpers, $y(t)$ sei also der zur Zeit t zurückgelegte Weg.

- Welche Lösung erhält man im Fall $c = 0$?
Gibt es noch eine zweite Lösung der DGL mit der AB $y(0) = 0$?
- Mit welcher AB kann ein Einschrittverfahren einen freien Fall berechnen?
- Gegeben sei $\tau \geq 0$ und die Funktion

$$z(t) = \begin{cases} 0 & , \text{ falls } 0 \leq t \leq \tau \\ \frac{1}{4}c^2(t - \tau)^2 & , \text{ falls } t > \tau. \end{cases}$$

Erfüllt $z(t)$ die DGL?

Welche Auswirkung hat die Antwort auf das Problem der Eindeutigkeit der Lösung?

Worin liegt die Ursache für diese Situation?

Hinweis: Untersuche die Glattheit der rechten Seite $f(t, y)$.

2. Bewegungsgleichung eines Körpers in laminarer Strömung

Die Bewegungsgleichung eines Körpers mit der Masse m in einer zähen Flüssigkeit (laminare Strömung) mit dem Stokes'schen Widerstandsgesetzes $R(v) = K_1 v$ (K_1 Materialeigenschaften des Mediums/Geometrie des Körpers) ergibt sich aus dem Kraftgesetz

$$m\dot{v} = A + G + R(v), \quad G \text{ Erdanziehung, } A \text{ Auftrieb.}$$

Die abgeleitete Modellgleichung für die Geschwindigkeit $v(t)$ sei

$$\dot{v} = c + k_1 v, \quad c > 0, \quad v(0) \text{ gegeben.}$$

- Diskutiere eine Lösung bei verschiedenen Startwerten $v(0)$.

- Es sei $v(0) = 0$, $k_1 < 0$, $c > 0$.

Gibt es eine Gleichgewichtsgeschwindigkeit $v_\infty = v(t = \infty) = \text{const}$?

3. Bewegungsgleichung eines Körpers in turbulenter Strömung

Die Bewegungsgleichung eines Körpers mit der Masse m in einer turbulenten Strömung (Neigung zur Wirbelbildung) mit dem Newtonschen Widerstandgesetz $R(v) = K_2 v^2$ (K_2 Materialeigenschaften/Geometrie) ergibt sich aus dem Kraftgesetz

$$m\dot{v} = A + G + R(v).$$

Die abgeleitete Modellgleichung für die Geschwindigkeit $v(t)$ sei

$$\dot{v} = c + k_2 v^2, \quad c > 0.$$

Es sei $v(0) = v_0$, $k_2 < 0$, $c > 0$.

- Diskutiere eine Lösung bei verschiedenen Startwerten.

- Gibt es eine Gleichgewichtsgeschwindigkeit $v_\infty = v(t = \infty) = \text{const}$?

4. Bewegungsgleichung für den freien Fall

Die Bewegungsgleichung für den freien Fall (ohne Reibung) mit der von der Höhe $z(t)$ abhängigen Erdbeschleunigung $g(z)$ lautet $\ddot{z} = -g(z)$.

Nach dem Newton'schen Gravitationsgesetz gilt $g(z) = \frac{g_0 z_0^2}{z^2}$, wobei $g_0 = g(z_0)$ den Wert der Erdbeschleunigung auf der Höhe z_0 bezeichnet.

Für kleine Fallhöhen kann man 3 Approximationen der rechten Seite der DGL anwenden.

$$(1) \quad g_0 = g = 9.806 \text{ ms}^{-2} \text{ Normwert, } r_E = 6.375 \cdot 10^6 \text{ m Erdradius, } z_0 = z,$$

$$\ddot{z} = -g,$$

$$(2) \quad g(z) \approx g - 2\frac{g}{r_E}(z - r_E) = 3g - 2\frac{g}{r_E}z,$$

$$\ddot{z} = -3g + 2\frac{g}{r_E}z,$$

$$(3) \quad z_0 = r_E, \quad g_0 = g,$$

$$\ddot{z} = -\frac{g r_E^2}{z^2}.$$

Die AB sind jeweils $z(0) = r_E + H$, $\dot{z}(0) = 0$, $H > 0$ Fallhöhe.

- (a) Welche konstante Funktion ist Lösung der Modellgleichung (2)?
Finde ihre allgemeine Lösung.
- (b) Berechne die Fallzeit t_F und Endgeschwindigkeit $\dot{z}(t_F)$ bei einer Fallhöhe H und gegebenen AB für die Modellgleichungen (1), (2).
- (c) Welchen Einfluß hat die Linearisierung des Modells?

5. Bewegungsgleichungen, Schwingungen, Pendel (parametrischen DGL)

- (a) Schwingungsgleichung ohne Einwirkung äußerer Kräfte

$$y'' + 2hy' + k^2y = 0$$

- (b) DGL der erzwungenen Schwingung $y(t)$ mit Einwirkung äußerer Kräfte
Untersuche $y(t)$ für $t \rightarrow \infty$.

$$y'' + 2\rho y' + \omega^2 y = a \sin(\omega_1 t), \quad y(0) = 0, \quad y'(0) = \omega$$

- (c) Van der Polsche DGL

$$\ddot{y} - \varepsilon(1 - y^2)\dot{y} + y = 0, \quad y(0) = y_0, \quad \dot{y}(0) = \dot{y}_0, \quad t \geq 0, \quad \varepsilon \geq 0$$

Der Parameter ε legt die Einschwingdauer fest.

Allgemeine Lösung für $\varepsilon = 0$: $y(t) = \dot{y}_0 \sin t + y_0 \cos t$

Van der Polsche DGL mit periodischer Erregung

$$\ddot{y} - \varepsilon(1 - y^2)\dot{y} + y = k \cos(\Omega t)$$

- (d) Rotation eines Pendels mit der Winkelgeschwindigkeit ω

$$\ddot{y} + \sin y - \frac{1}{2}\omega^2 \sin(2y) = 0, \quad y(0) = \frac{\pi}{4}, \quad \dot{y}(0) = 0, \quad \omega \in [0, 10]$$

- (e) Pendel mit Anfangsbedingung

$$y'' + k^2 y = 0, \quad k > 0, \quad y(0) = y_0, \quad y'(0) = y'_0$$

$$\text{Lösung: } y = \frac{y'_0}{k} \sin(kx) + y_0 \cos(kx)$$

$$y = \frac{y'_0}{k} \sin(kx), \quad \text{falls } y_0 = 0$$

$$y = \sin x, \quad \text{falls } y_0 = 0, \quad y'_0 = 1, \quad k = 1$$

$$y = c_1 \sin(kx) + y_0 \cos(kx), \quad \text{falls AB } y'(0) \text{ fehlt}$$

- (f) Fadenpendel

$$y'' + ay' + by = 0, \quad D = a^2 - 4b \quad \text{Diskriminante}$$

Lösung:

(1) $D > 0$

$$y = c_1 e^{k_1 x} + c_2 e^{k_2 x}$$

$$k_1 = -\frac{1}{2}(a - \sqrt{D}), \quad k_2 = -\frac{1}{2}(a + \sqrt{D})$$

z.B. falls $b < 0$, dann $D > 0$

(2) $D = 0$

$$y = c_1 e^{k_1 x} + c_2 x e^{k_1 x}, \quad k_1 = -\frac{a}{2}$$

z.B. falls $b = 0$, $a = 0$, dann $D = 0$

(3) $D < 0$

$$y = e^{-\frac{ax}{2}} (c_1 \sin(\frac{1}{2}\sqrt{|D|x}) + c_2 \cos(\frac{1}{2}\sqrt{|D|x}))$$

$$k_1 = -\frac{a}{2} + \frac{1}{2}\sqrt{-D} i, \quad k_2 = -\frac{a}{2} - \frac{1}{2}\sqrt{-D} i$$

z.B. falls $b \gg 0$, dann $D < 0$; $a > 0 \rightarrow$ gedämpftes Pendel,
 $a < 0 \rightarrow$ erregtes Pendel

(g) Bewegungsgleichung

$$\ddot{y} + f(y) = 0, \quad y(0) = 2, \quad \dot{y}(0) = 0, \quad f(z) = \begin{cases} -1 & , \text{ falls } z \leq -1 \\ z & , \text{ falls } -1 < z < 1 \\ 1 & , \text{ falls } 1 \leq z \end{cases}$$

(h) Duffingsche Bewegungsgleichungen, Vibration $y = y(t)$

$$\ddot{y} + y + \lambda y^3 = 0, \quad y(0) = y_0, \quad \dot{y}(0) = \dot{y}_0, \quad \lambda \in \mathbb{R}^+$$

$$\ddot{y} + y + \lambda y^3 = k \cos(\alpha t), \quad \lambda \in \mathbb{R}^+, \quad k, \alpha \in \mathbb{R}$$

$$\ddot{y} + \varepsilon \dot{y} - \beta y + \alpha y^3 = f \cos(\omega t)$$

$$\ddot{y} + 0.7\dot{y} + y^3 = 0.75 \cos t, \quad y(0) = \dot{y}(0) = 0$$

$$\ddot{y} + 0.2\dot{y} + y|y| = 1.5 \cos(2t) + 0.5, \quad y(0) = \dot{y}(0) = 0$$

$$\ddot{y} + \dot{y} - \frac{1}{6}y^3 = 0.8 \sin(0.27\omega t), \quad y(0) = \dot{y}(0) = 0, \quad \omega = 0.92845$$

(i) Bewegungsgleichung mit quadratischer Dämpfung

$$\ddot{y} + \frac{1}{2}\dot{y}^2 + y = 0, \quad y(0) = 1, \quad \dot{y}(0) = 0$$

(j) Emden's Bewegungsgleichungen

$$\frac{d}{dt}(t^2 \dot{y}) + t^2 y^n = 0, \quad y(0) = 1, \quad \dot{y}(0) = 0, \quad n = 2, 3, 5, \dots$$

$$\frac{1}{t^2} \frac{d}{dt}(t^2 \dot{y}) + f(y) = \ddot{y} + \frac{2}{t}\dot{y} + f(y) = 0, \quad y(0) = 1, \quad \dot{y}(0) = 0,$$

$$f(y) = \sin y \quad \text{bzw.} \quad \cos y$$

(k) Liouvillesche DGL

$$\frac{d}{dt}(t^2 \dot{y}) + t^2 e^y = 0, \quad y(0) = \dot{y}(0) = 0$$

(l) Mathieu's DGL für ein Pendel mit zeitabhängiger Länge $l(t)$

$$\ddot{y} + (a - 2q \cos(2t))y = 0, \quad y(0) = 1, \quad \dot{y}(0) = 0$$

$$a = 1.0, \quad q = 0.1 \quad \text{aufschaukelnde Schwingung}$$

$$a = 1.2, \quad q = 0.1 \quad \text{gedämpfte Schwingung}$$

Man kann einen (a, q) -Bereich finden, in dem die Lösung beschränkt bleibt (Stutt-Diagramm).

(m) Gedämpfte Schwingung einer Membran, Besselsche DGL

$$t^2 y'' + t y' + (t^2 - k^2)y = 0, \quad k \in \mathbb{Z}$$

Lösung : $y = J_0(t)$, falls $k = 0$, $y(0) = 1$, $y'(0) = 0$, Besselfunktion 1.Art

$$y = J_1(t), \quad \text{falls } k = 1, \quad y(0) = 0, \quad y'(0) = 0.5, \quad \text{Besselfunktion 1.Art}$$

(n) Resonanz zwischen Eigenschwingung und Störung

$$y'' + 6y' + 5y = 3e^{-t}$$

(o) Abrollen einer Kugel auf einer Fallinie eines parabolischen Zylinders

$$\ddot{y} + \frac{y}{2025 + y^2}(31532 + \dot{y}^2) = 0, \quad y(0) = 30, \quad \dot{y}(0) = 0$$

(p) Rollpendel

$$\left(\frac{\Theta}{m} + (r^2 + s^2 - 2rs \cos \phi) \right) \ddot{\phi} + (r\dot{\phi}^2 + g)s \sin \phi = 0$$

(q) Spannungsstabilisatorschaltung, Ferroresonanzstabilisator [PHILIPPOW]

$$\ddot{y} - \delta \dot{y} + ay + by^9 + \alpha y = -\Gamma \cos(t + \phi)$$

(r) Frequenzteilerschaltung

$$\ddot{y} - \varepsilon(1 - y^2 - \dot{y}^2)\dot{y} + y + 2By\dot{y} = B \sin(2t)$$

(s) Schwingungen im Wärmeaustauscher

$$\ddot{y} + \varepsilon_1 \dot{y} + \varepsilon_2 y + y^3 + \dot{y}^3 = 0$$

(t) Bewegungsgleichung für den freien Fall

$$\ddot{z} = -\frac{g r_E^2}{z^2}, \quad z(0) = r_E + H, \quad \dot{z}(0) = 0$$

H Fallhöhe, $g_0 = g = 9.806 \text{ m s}^{-2}$ Normwert der Erdbeschleunigung,
 $r_E = 6.375 \cdot 10^6 \text{ m}$ Erdradius

(u) Hamiltonsche Bewegungsgleichung für das Hénon Heiles-Modell [STEEB]

$$\begin{aligned}\dot{q}_1 &= p_1, & \dot{p}_1 &= -q_1 - 2q_1q_2 \\ \dot{q}_2 &= p_2, & \dot{p}_2 &= -q_2 - q_1^2 + q_2^2\end{aligned}$$

AB: z.B. $q_1(0) = 0.5, q_2(0) = p_1(0) = p_2(0) = 0.1$

(v) Bewegungsgleichung für die biquadratische Hamiltonfunktion

$$\begin{aligned}\dot{q}_1 &= p_1, & \dot{p}_1 &= -\frac{1-r}{3}q_1^3 - q_1q_2^2, & r &\in [0, 1] \\ \dot{q}_2 &= p_2, & \dot{p}_2 &= -\frac{1-r}{3}q_2^3 - q_1^2q_2\end{aligned}$$

Für wachsendes r wird das System mehr und mehr chaotisch.

(w) Bewegungsgleichung für die Hamiltonfunktion *Toda lattice*

$$\begin{aligned}\dot{a}_j &= a_j(b_j - b_{j+1}), & j &= 1, 2 & (a_3 \equiv 0) \\ \dot{b}_1 &= -2a_1^2 \\ \dot{b}_2 &= 2(a_1^2 - a_2^2) \\ \dot{b}_3 &= 2a_2^2\end{aligned}$$

(x) Schwingende Atwood-Maschine [STEEB]

$$\begin{aligned}(1 + \mu) \ddot{r} &= r(\dot{\Theta})^2 + \cos \Theta - \mu, & \mu &\text{ Masse} \\ r \ddot{\Theta} &= -2\dot{r}\dot{\Theta} - \sin \Theta\end{aligned}$$

(y) Chaotische Systeme mit Orbits (Homoklinen) [STEEB]

siehe auch Duffingsche Bewegungsgleichungen

$$\ddot{u} - \frac{1}{2}u + u^2 + u^3 = 0$$

$$\ddot{u} + u^3 = 0$$

$$\ddot{u} + a\dot{u} + bu + cu^3 = k_1 + k_2 \cos(\Omega t)$$

Parametersatz z.B. $a = 1, b = -10, c = 100, k_1 = 0, k_2 = 1.2, \Omega = 3.5$

$$\ddot{u} + a\dot{u} + b \sin u = k \cos(\Omega t)$$

$a = 0.2, b = 1, k = 1.1, \Omega = 0.8$

$$\ddot{u} + a\dot{u} + (1 + k \cos(\Omega t)) \sin u = 0$$

$a = 0.15, k = 0.94, \Omega = 1.56$

6. Fadenpendel

Wie verläuft die Bewegung eines Pendelkörpers unter den Vereinfachungen: keine Berücksichtigung des Luftwiderstandes, Masse des Fadens wird vernachlässigt, Pendelkörper ist Massepunkt (mathematisches Pendel).

Bekanntlich gilt nach NEWTON $F = m a$.

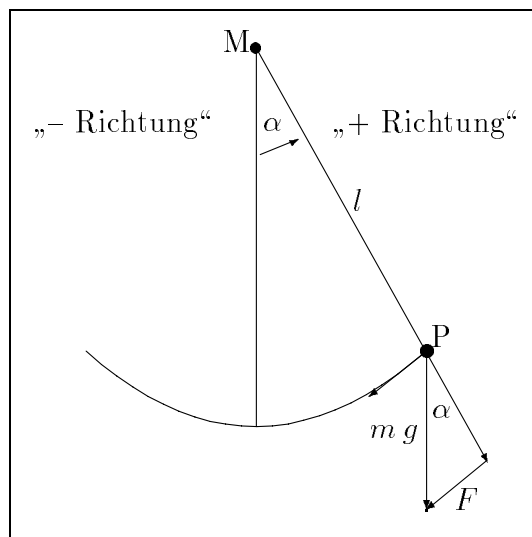
Für kleine Auslenkungen $\alpha(t)$ ergibt sich aus dem Newtonschen Grundgesetz folgende Gleichung für das mathematische Pendel (Kräfteparallelogramm)

$$m g \sin \alpha(t) = -m l \alpha''(t),$$

l	Fadenlänge,
$l \alpha(t)$	Weg, den das Pendel zurücklegt = Länge des Kreisbogens,
$a = l \alpha''(t)$	Beschleunigung des Pendelkörpers,
g	Erdbeschleunigung,
$m g$	Gewichtskraft,
F und α	haben entgegengesetzte Richtungen.

Für kleine Winkel ist $\sin \alpha \approx \alpha$, so daß eine einfache lineare DGL 2. Ordnung entsteht.

$$\alpha''(t) + \frac{g}{l} \alpha(t) = 0.$$



(a) Allgemeine Lösung der DGL

Hinweis: Finden von Basislösungen mittels Ansatz $e^{\lambda t}$.

Die beiden unabhängigen Lösungen sind $\sin(\sqrt{\frac{g}{l}}t)$, $\cos(\sqrt{\frac{g}{l}}t)$.

Die Lösungsmenge ist ein Vektorraum gebildet mittels der Linearkombination

$$\alpha(t) = c_1 \sin\left(\sqrt{\frac{g}{l}}t\right) + c_2 \cos\left(\sqrt{\frac{g}{l}}t\right), \quad c_{1,2} \in \mathbb{R}.$$

(b) Berücksichtigung von Anfangsbedingungen

Es sollen 2 Varianten unterschieden werden.

- Pendel wird nach Auslenkung um einen Winkel α_0 losgelassen.

Wenn zur Zeit $t = 0$ die AB $\alpha(0) = \alpha_0$ und $\alpha'(0) = 0$ bekannt sind, ergeben sich die Koeffizienten zu

$$c_1 = 0, \quad c_2 = \alpha_0.$$

Daraus folgt

$$\alpha(t) = \alpha_0 \cos\left(\sqrt{\frac{g}{l}} t\right).$$

- Pendel wird aus der Lage $\alpha(0) = 0$ angestoßen mit der Geschwindigkeit $v_0 = l\alpha'(0)$.

Wenn zur Zeit $t = 0$ die AB $\alpha(0) = 0$ und $\alpha'(0)$ bekannt sind, ergeben sich die Koeffizienten zu

$$c_2 = 0, \quad c_1 = \sqrt{\frac{l}{g}} \alpha'(0).$$

Daraus folgt

$$\alpha(t) = \sqrt{\frac{l}{g}} \alpha'(0) \sin\left(\sqrt{\frac{g}{l}} t\right).$$

(c) Lösung der DGL 2.Ordnung mittels Einschrittverfahren

Das AWP

$$\alpha''(t) = -\frac{g}{l} \alpha(t), \quad \alpha(0), \alpha'(0) \text{ gegeben, } t \in [0, T]$$

kann man mittels $\alpha'(t) = \beta(t)$ auf ein System von zwei DGL 1.Ordnung transformieren.

$$\begin{aligned} \alpha'(t) &= \beta(t), & \alpha(0) &\text{ gegeben} \\ \beta'(t) &= \alpha''(t) = -\frac{g}{l} \alpha(t), & \beta(0) &= \alpha'(0) \text{ gegeben.} \end{aligned}$$

Nunmehr kann man auf das allgemein notierte DGL-System

$$\begin{aligned} x' &= f(t, x, y), & x(0) &\text{ gegeben} \\ y' &= g(t, x, y), & y(0) &\text{ gegeben} \end{aligned}$$

das numerische Verfahren anwenden.

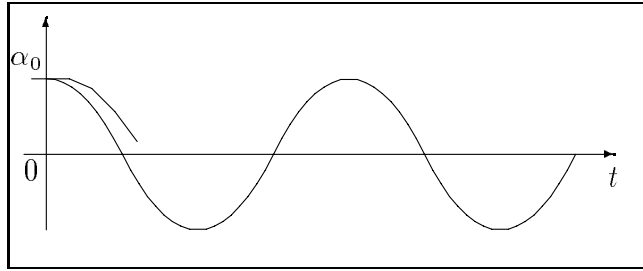
$$\begin{aligned} x_{n+1} &= x_n + hf(t_n, x_n, y_n), & x_0 &= x(0), & n &= 0, 1, \dots, & t_n &= nh, & h > 0 \\ y_{n+1} &= y_n + hg(t_n, x_n, y_n), & y_0 &= y(0). \end{aligned}$$

Vergleiche die Näherungslösung mit der exakten.

(d) 2 Varianten der graphischen Darstellung

- Vergleichende Graphik für Näherung $\{\alpha_0, \alpha_1, \alpha_2, \dots\}$ (Polygonzug) und exakte Lösung $\alpha(t)$.

Illustration am Beispiel $\alpha(t) = \alpha_0 \cos(\sqrt{\frac{g}{l}} t)$.

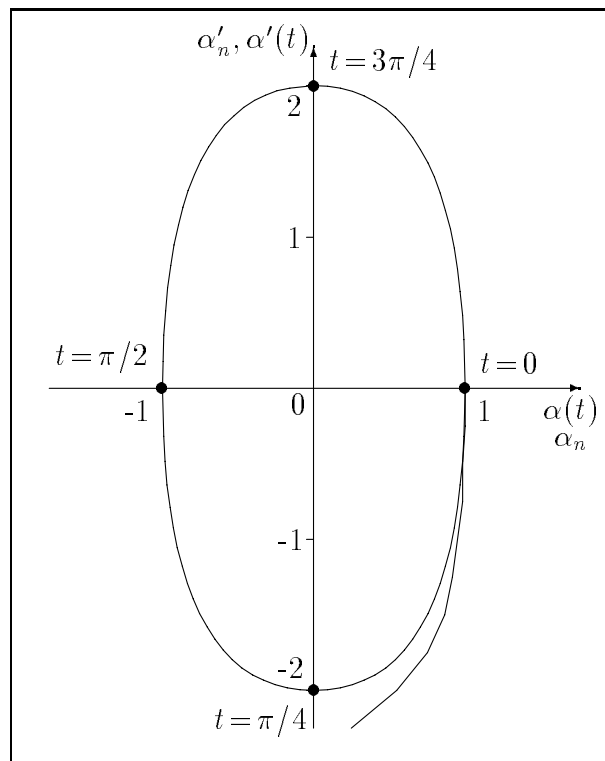


- Vergleichende Graphik für Phasenkurven
 - der Näherung $\{(\alpha_0, \alpha'_0), (\alpha_1, \alpha'_1), (\alpha_2, \alpha'_2), \dots\}$, $\alpha' = \beta$, (Polygonzug)
 - und exakten Lösung $(\alpha(t), \alpha'(t))$.

Illustration am Beispiel

$$\alpha(t) = \alpha_0 \cos\left(\sqrt{\frac{g}{l}} t\right),$$

$$\alpha'(t) = -\alpha_0 \sqrt{\frac{g}{l}} \sin\left(\sqrt{\frac{g}{l}} t\right), \quad \sqrt{\frac{g}{l}} = 2, \quad \alpha_0 = 1.$$



7. Simulation eines Doppelpendels [HENNING, KUTSCHA]

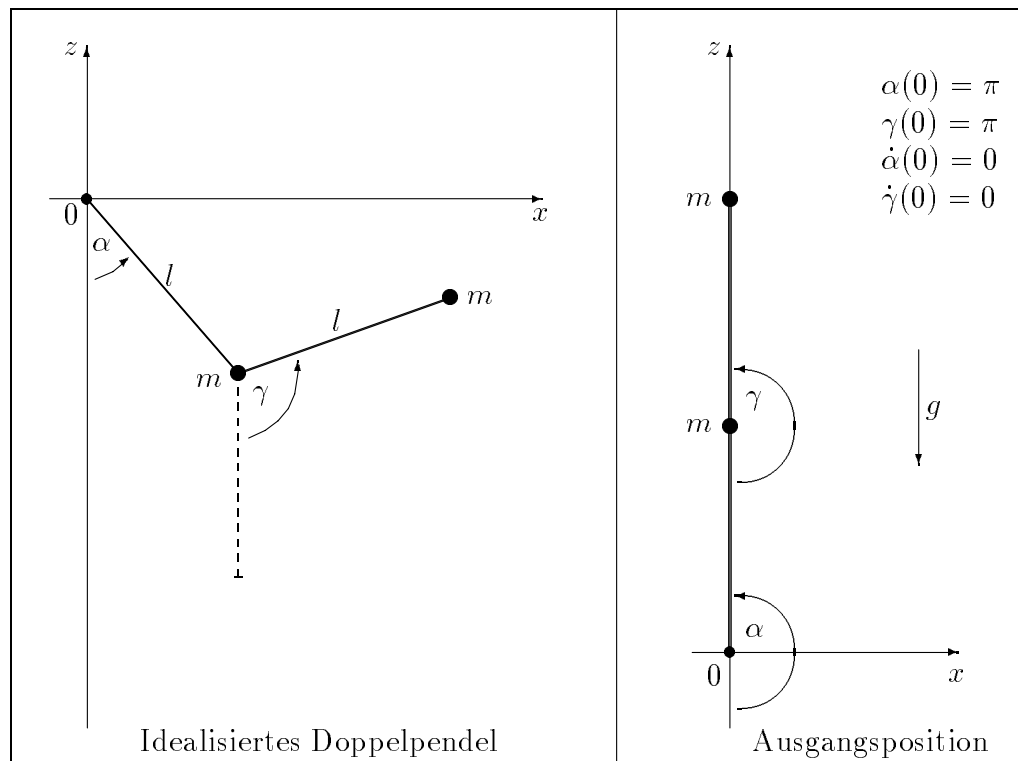
Wir betrachten ein idealisiertes Doppelpendel.

Zwei Punktmassen gleicher Masse m , die an zwei masselosen Stangen der Länge l aufgehängt sind, bewegen sich unter dem Einfluß der Schwerkraft mg . Der Luftwiderstand, die Reibung der Lager und andere Störungen werden vernachlässigt.

Der Winkel $\alpha(t)$ gibt die Lage des inneren Körpers 1 bezüglich der z -Achse an, $\gamma(t)$ die des äußeren Körpers 2 bezüglich der z -Achse an.

Ziel ist die Simulation und Beschreibung des Bewegungsablaufes mittels der Winkel $\alpha(t)$ und $\gamma(t)$ ohne bzw. mit Störgrößen.

Die Ausgangsposition sei gegeben durch $\alpha(0)$, $\gamma(0)$, $\dot{\alpha}(0)$, $\dot{\gamma}(0)$.



(a) Charakterisierung der Geometrie, Rotation und Schwingung sowie DGL

- Geometrie, Lage des inneren und äußeren Körpers im \mathbb{R}^3

$$r_1 = l \begin{pmatrix} \sin \alpha \\ 0 \\ -\cos \alpha \end{pmatrix}, \quad r_2 = l \begin{pmatrix} \sin \alpha + \sin \gamma \\ 0 \\ -\cos \alpha - \cos \gamma \end{pmatrix}$$

- Energieerhaltungssatz

$$E = \dot{\alpha}^2 + \frac{1}{2}\dot{\gamma}^2 + \dot{\alpha}\dot{\gamma} \cos(\gamma - \alpha) - 2 \cos \alpha - \cos \gamma$$

- Drallsatz

$$\frac{dL_A}{dt} = \begin{pmatrix} 0 \\ 2 \sin \alpha + \sin \gamma \\ 0 \end{pmatrix}$$

- System von Differentialgleichungen 2.Ordnung

$$\ddot{\alpha} = \frac{-2 \sin \alpha + \dot{\gamma}^2 \sin(\gamma - \alpha) + \cos(\gamma - \alpha)[\sin \gamma + \dot{\alpha}^2 \sin(\gamma - \alpha)]}{2 - \cos^2(\gamma - \alpha)}$$

$$\ddot{\gamma} = \frac{-2 \sin \gamma - 2\dot{\alpha}^2 \sin(\gamma - \alpha) + \cos(\gamma - \alpha)[2 \sin \alpha - \dot{\gamma}^2 \sin(\gamma - \alpha)]}{2 - \cos^2(\gamma - \alpha)}$$

- (b) Man löse nun die DGL 2.Ordnung mittels Polygonzugverfahren (EULER-CAUCHY-Verfahren), das für die DGL $\ddot{y} = f(t, y, \dot{y})$ mit Anfangsbedingungen die Gestalt hat

$$y_{n+1} = y_n + \Delta t \dot{y}_n, \quad n = 0, 1, \dots, N, \quad t_n = n\Delta t, \quad y_0, \dot{y}_0 \text{ gegeben,}$$

$$\dot{y}_{n+1} = \dot{y}_n + \Delta t \ddot{y}_n = \dot{y}_n + \Delta t f(t, y_n, \dot{y}_n),$$

Schrittweite $\Delta t > 0$ und Anzahl der Schritte N (Simulationsdauer).

Für obiges System arbeitet das PZV gemäß der Reihenfolge

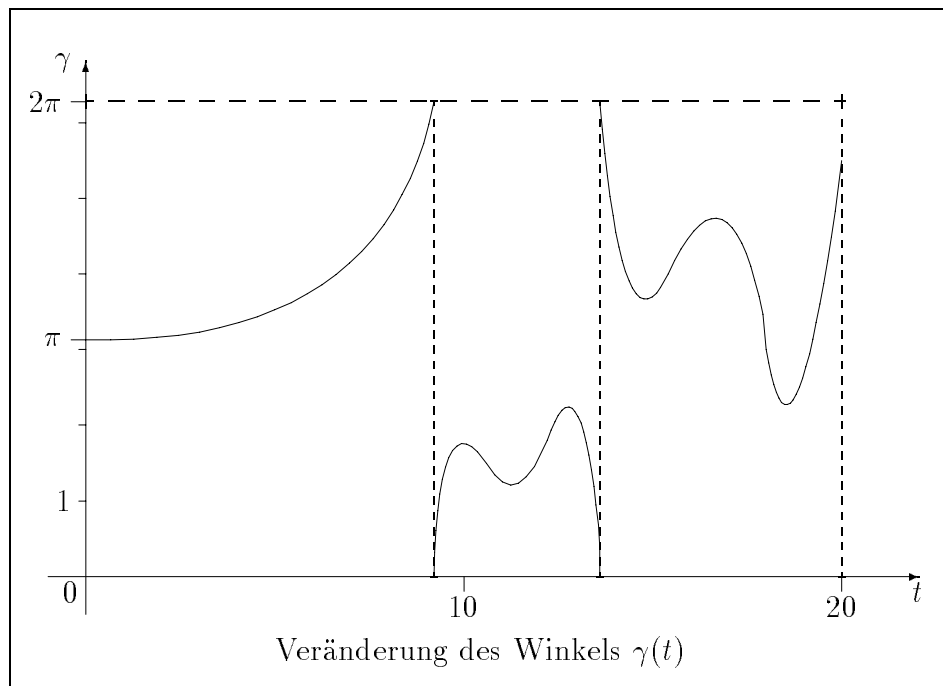
- Berechnung von $\ddot{\alpha}$, $\ddot{\gamma}$ anhand der DGL zum Zeitpunkt t_n
- $\alpha_{n+1} = \alpha_n + \Delta t \dot{\alpha}_n$,
- $\gamma_{n+1} = \gamma_n + \Delta t \dot{\gamma}_n$,
- $\dot{\alpha}_{n+1} = \dot{\alpha}_n + \Delta t \ddot{\alpha}_n$,
- $\dot{\gamma}_{n+1} = \dot{\gamma}_n + \Delta t \ddot{\gamma}_n$.

- (c) Tabellarische Ausgabe der Ergebnisse, eventuell auch als File

- (d) Darstellung der berechneten Bewegung in verschiedenen Plots

- Bewegungsablauf des Doppelpendels (der beiden Stangen) im (x, z) -Koordinatensystem (Graphik wie oben),
- Darstellung der Winkelfunktionen $\alpha(t)$, $\gamma(t)$ entsprechend der Simulationsdauer. Dabei sollen Winkelwerte, die außerhalb von $[0, 2\pi]$ liegen, mittels Modulo-Rechnung auf dieses Intervall transformiert werden.

Die nachfolgende qualitative Abbildung entstand mit den AB $\alpha(0) = \pi$, $\dot{\alpha}(0) = 0$, $\gamma(0) = \pi$, $\dot{\gamma}(0) > 0$.



- (e) Man kann in die Betrachtungen noch eine kleine Störung s einbeziehen (Masse auf der x -Achse, die eine geringe Beschleunigung verursacht). Größenordnungen für s sind $10^{-5} \dots 10^{-10}$. In den Beziehungen wirkt sich diese wie folgt aus.

Energieerhaltungssatz

$$E = \dot{\alpha}^2 + \frac{1}{2}\dot{\gamma}^2 + \dot{\alpha}\dot{\gamma} \cos(\gamma - \alpha) - 2 \cos \alpha - \cos \gamma + s(2 \sin \alpha - \sin \gamma)$$

Drallsatz

$$\frac{dL_A}{dt} = \begin{pmatrix} 0 \\ 2 \sin \alpha + \sin \gamma \\ 0 \end{pmatrix} - s \begin{pmatrix} 0 \\ 2 \cos \alpha + \cos \gamma \\ 0 \end{pmatrix}$$

Gestörtes System von Differentialgleichungen 2.Ordnung

$$\begin{aligned} \ddot{\alpha} &= \frac{-2 \sin \alpha + \dot{\gamma}^2 \sin(\gamma - \alpha) + \cos(\gamma - \alpha)[\sin \gamma + \dot{\alpha}^2 \sin(\gamma - \alpha)]}{2 - \cos^2(\gamma - \alpha)} \\ &+ s \frac{\cos \gamma \cos(\gamma - \alpha) - 2 \cos \alpha}{2 - \cos^2(\gamma - \alpha)} \\ \ddot{\gamma} &= \frac{-2 \sin \gamma - 2\dot{\alpha}^2 \sin(\gamma - \alpha) + \cos(\gamma - \alpha)[2 \sin \alpha - \dot{\gamma}^2 \sin(\gamma - \alpha)]}{2 - \cos^2(\gamma - \alpha)} \\ &+ s \frac{2 \cos(\gamma - \alpha)(\cos \alpha - 1)}{2 - \cos^2(\gamma - \alpha)} \end{aligned}$$

Analog kann man die gestörten Winkelfunktionen mittels PZV berechnen, die Ergebnisse tabellarisch ausgeben sowie den Bewegungsablauf graphisch darstellen.

2.5 Mathematik

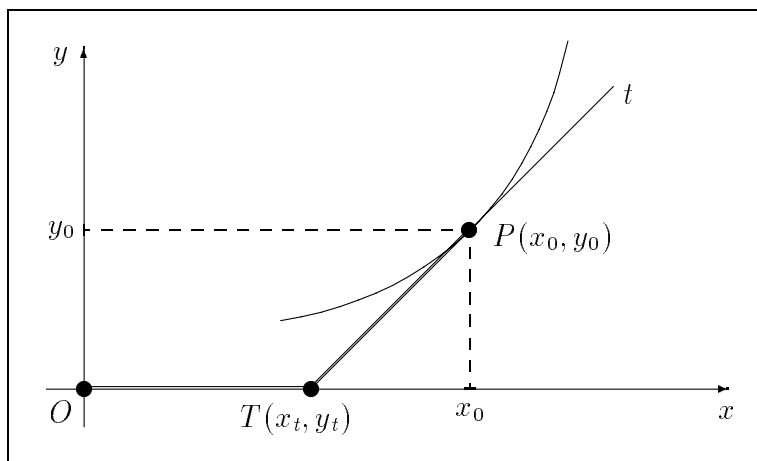
1. Bestimmung von Kurven mittels Abstandsbedingung I

Gesucht sind alle Kurven mit folgenden Eigenschaften:

Die Tangente an einen Kurvenpunkt P schneide die x -Achse in T . Wie lautet die Gleichung der Kurve, wenn der Abstand zwischen P und T gleich dem Abstand zwischen T und dem Koordinatenursprung O ist für alle Punkte P der Kurve?

Aus der Abstandsbedingung $|OT|^2 = |TP|^2$ und der Tangentengleichung $y - y(x_0) = y'(x_0)(x - x_0)$ am Punkt $P(x_0, y_0)$ einer Kurve $y = y(x)$ folgt die DGL

$$y' = \frac{2xy}{x^2 - y^2}.$$



(a) Allgemeine Lösung der DGL

Es ist eine homogene DGL, die unter Nutzung der Substitution $u = \frac{y}{x}$ sowie mittels Partialbruchzerlegung gelöst werden kann.

Mit $y' = u'x + u$ erhält man

$$\begin{aligned} y' &= \frac{2\frac{y}{x}}{1 - \left(\frac{y}{x}\right)^2} \\ u'x + u &= \frac{2u}{1 - u^2} \\ u'x &= \frac{u(1 + u^2)}{1 - u^2} \\ \frac{1}{x} &= \frac{1 - u^2}{u(1 + u^2)} u' = \frac{u'}{u} - \frac{2uu'}{1 + u^2} \end{aligned}$$

$$\ln |u| - \ln |1 + u^2| = \ln |x| + c$$

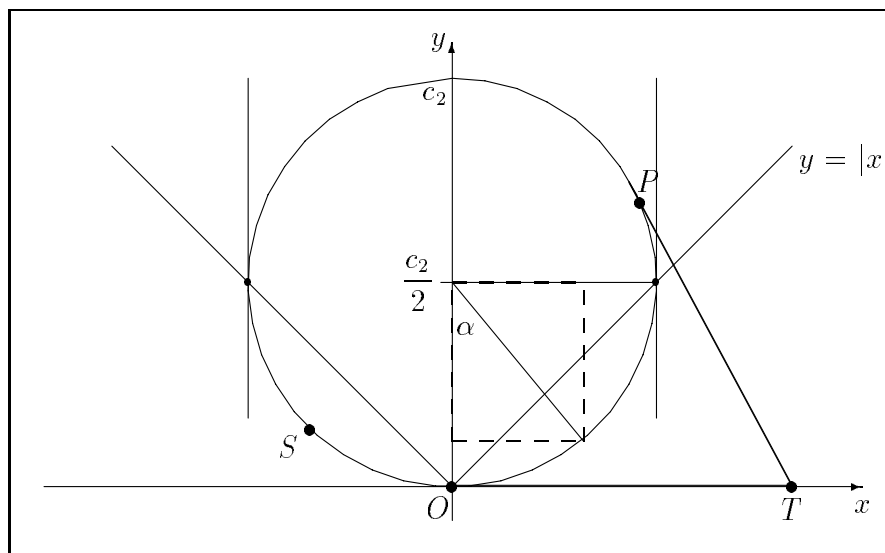
$$\left| \frac{u}{1+u^2} \right| = c_1|x|, \quad c_1 \in \mathbb{R}^+.$$

Mit $u = y/x$ erfolgt nach Umformung

$$y = c_1(y^2 + x^2)$$

$$\left(y - \frac{c_2}{2}\right)^2 + x^2 = \left(\frac{c_2}{2}\right)^2, \quad c_2 = 1/c_1.$$

Das sind offenbar Kreisgleichungen. Die Mittelpunkte der Kreise liegen auf der y -Achse.



- (b) Man löse nun die DGL $y' = \frac{2xy}{x^2 - y^2} = f(x, y)$ mittels Polygonzugverfahren (EULER-CAUCHY-Verfahren)

$$y_{n+1} = y_n + hf(x_n, y_n), \quad n = 0, 1, \dots, \quad x_n = x_0 + nh, \quad y_0 = y(x_0),$$

Schrittweite $h > 0$ gegeben.

Jedoch sind zunächst einige Einschränkungen und Besonderheiten zu berücksichtigen.

- Im kartesischen Koordinatensystem kann man mit der DGL wegen der Eindeutigkeit der Lösungskurve maximal den unteren oder oberen Halbkreis beschreiben. Ansonsten ist die DGL zu transformieren, indem die Koordinaten in Abhängigkeit vom Winkel α geschrieben werden, z.B. $x(\alpha) = \frac{c_2}{2} \sin \alpha$, $y(\alpha) = \frac{c_2}{2}(1 - \cos \alpha)$, $\alpha \geq 0$.
- Die AB kann nicht $(x_0, 0)$ sein. Dann ist $y'(x_0) = f(x_0, 0) = 0$ für $x_0 \neq 0$ oder man würde $y'(0) = f(0, 0)$ per Definition zu Null setzen. Das PZV $y_{n+1} = y_n(1 + h \frac{2x_n}{x_n^2 - y_n^2})$ liefert in beiden Fällen die bekannte Nulllösung der DGL (Grenzfall = unendlicher Kreis).

- Als AB ist $y(x_0) = |x_0|$, $x_0 \neq 0$, nicht zugelassen, weil dann die Funktion $f(x, y)$ für den Startwert nicht definiert ist. Diese AB liegen dort, wo die Tangenten an den Kreis vertikal sind.
- Man wähle also geeignete AB $S(x_0, y_0)$, berechne mittels PZV die Näherungslösung und kontrolliere ihren Verlauf.
Mache einen graphischen Vergleich der exakten Lösung (Kreisbogen) mit dem Lösungsverlauf des Näherungsverfahrens (Polygonzug).
Teste verschiedene Schrittweiten h .

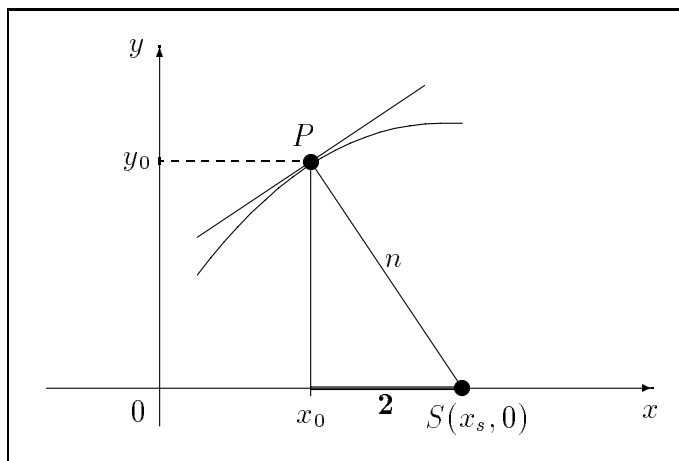
2. Bestimmung von Kurven mittels Abstandsbedingung II

Gesucht sind alle Kurven mit folgender Eigenschaft:

Der Schnittpunkt S der Kurvennormalen n in einem Kurvenpunkt P mit der x -Achse soll von der Abszisse von P stets den Abstand 2 haben.

Aus der Normalengleichung $y - y(x_0) = -\frac{1}{y'(x_0)}(x - x_0)$ im Punkt $P(x_0, y_0)$ einer Kurve $y = y(x)$, ihrem Schnittpunkt $S(x_s, 0)$ mit der x -Achse sowie der Abstandsbedingung $x_s - x_0 = 2$ folgt die DGL

$$y y' = 2.$$



(a) Allgemeine Lösung der DGL

Es gilt

$$\begin{aligned} \frac{1}{2}(y^2)' &= 2 \\ y^2 &= 4x + c \\ y &= 2\sqrt{x + c_1}, \quad x \geq -c_1. \end{aligned}$$

- (b) Charakterisiere das Richtungsfeld der DGL $y' = \frac{2}{y}$.

In jedem Punkt $P(x, y)$ des Gebietes $\Omega \in \mathbb{R}^2$ kann man die Steigung y' durch Eintragen eines kleinen „Strichs“ für die Tangentenrichtung andeuten. Ein solcher Strich heißt *Linienelement* oder *Richtungselement* (x, y, y') der Lösungskurve. Die Gesamtheit aller Linienelemente heißt *Richtungsfeld*. Was passiert, wenn $y = 0$?

- (c) Bestimme die orthogonalen Trajektorien der Kurvenschar.

Definition: Alle Kurven, die in jedem Punkt eine vorgegebene Kurvenschar orthogonal schneiden, heißen *orthogonale Trajektorien*.

Gegeben sei die explizite DGL 1.Ordnung $y' = f(x, y)$.

Für ihre Lösungsschar bestimmt man die orthogonale Kurvenschar, indem man lediglich y' durch $-\frac{1}{y'}$ ersetzt. Die Vektoren $(1, y')$ und $(1, -1/y')$ sind nämlich orthogonal.

Somit lautet deren DGL $\frac{y'}{y} = -\frac{1}{2}$, ihre allgemeine Lösung ist $y = c_1 e^{-\frac{x}{2}}$.

Zeichne ebenfalls das Richtungsfeld der orthogonalen Trajektorien.

- (d) Man löse nun die DGL $y' = \frac{2}{y} = f(x, y)$ sowie die der orthogonalen Trajektorien $y' = -\frac{y}{2}$ mittels Polygonzugverfahren (EULER-CAUCHY-Verfahren)

$$y_{n+1} = y_n + hf(x_n, y_n), \quad n = 0, 1, \dots, \quad x_n = x_0 + nh, \quad y_0 = y(x_0),$$

Schrittweite $h > 0$ gegeben.

Man wähle geeignete AB (x_0, y_0) , berechne mittels PZV die Näherungslösung und kontrolliere ihren Verlauf.

Mache einen graphischen Vergleich der exakten Lösung mit dem Lösungsverlauf des Näherungsverfahrens (Polygonzug).

Teste verschiedene Schrittweiten h .

3. Phasenkurve

Für die Lösung $y = y(x)$ der DGL $y' = f(x, y)$ ist manchmal auch eine andere Darstellung zu empfehlen. Das trifft zum Beispiel auf Bewegungsdifferentialgleichungen bei Abhängigkeit von der Zeit t zu. Formal kann man $x = t$ setzen und erhält das DGL-System

$$\dot{x} = 1, \quad \dot{y} = f(t, y).$$

Die Lösung $(x(t), y(t))$ wird *Phasenkurve* genannt und ihre Darstellung kann im Phasenraum (x, y) in Abhängigkeit von der Zeit t erfolgen.

Mit einer gegebenen Anfangsbedingung $(y(t_0), x(t_0)) = (y_0, x_0)$ löst man das AWP. Allgemein lautet das DGL-System $\dot{x} = f_1(t, x, y)$, $\dot{y} = f_2(t, x, y)$.

Der eigentliche Typ von Phasenkurven entsteht bei der DGL 2. Ordnung

$y'' = f(t, y, y')$, wenn man diese durch die Definition $x = y' = \dot{y}$ in das System von zwei Gleichungen

$$\dot{y} = x = f_1(t, y, x), \quad \dot{x} = \ddot{y} = f_2(t, y, x) = f_2(t, y, x)$$

überführt und somit die Phasenkurve $(y, x) = (y, \dot{y})$ erhält.

Konstruiere für folgende DGL die Phasenkurven.

a) $y' = 1 + y, y(0) = 1/2.$

b) $y'' + y = 0, y(0) = 0, y'(0) = 1,$ Pendel ohne Reibung.

c) $y'' + ky' + y = 0, y(0) = y_0, y'(0) = y'_0,$ Pendel mit Reibung $k > 0,$
z.B. $k = \frac{10}{3}.$

d) $y'' - (2 + e^t)y' + y = 0, y(0) = 2, y'(0) = -1.$

Liefert $(1,0)$ als Grenzwert der Phasenkurve eine stationäre Lösung?

4. Van der Polsche DGL [ARNOLD]

Normierte DGL 2.Ordnung

$$\ddot{y} - \varepsilon(1 - y^2)\dot{y} + y = 0, \quad y(0) = y_0, \dot{y}(0) = \dot{y}_0, \quad t \geq 0, \quad \varepsilon \geq 0.$$

Parameter ε legt die Einschwingdauer fest.

(a) Notation als Phasenkurve

$$\ddot{y} = \varepsilon(1 - y^2)\dot{y} - y = f(t, y, \dot{y}), \quad x \equiv \dot{y}$$

System von DGL 1.Ordnung

$$\dot{y} = x = f_1(t, y, x), \quad y(t_0) = y_0, \quad t_0 = 0$$

$$\dot{x} = f(t, y, x) = f_2(t, y, x), \quad x(t_0) = \dot{y}_0$$

Phasenkurve: $(y, x) = (y, \dot{y}).$

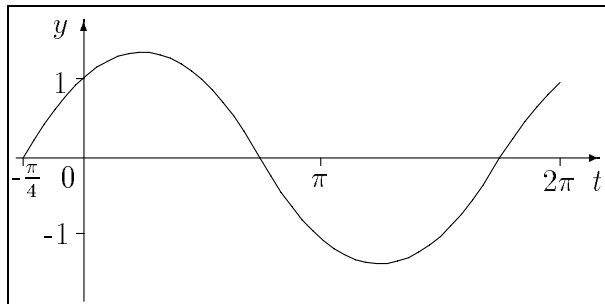
(b) Van der Pol DGL und Lösung für $\varepsilon = 0$

$$\ddot{y} + y = 0$$

$$y(t) = \dot{y}_0 \sin t + y_0 \cos t$$

$$= A \sin(\omega t + \varphi), \quad \text{harmonische Schwingung}$$

$$y(t) = \sqrt{2} \sin\left(t + \frac{\pi}{4}\right), \quad \text{Lösung für AB } y_0 = \dot{y}_0 = 1$$



Amplitude $A = \sqrt{2}$

Frequenz $\omega = 1$

Periode $\frac{2\pi}{\omega} = 2\pi$

(c) Lösung der DGL $\ddot{y} + \omega^2 y = 0$, $\omega > 0$, $\varepsilon = 0$

$$\begin{aligned}y(t) &= A \sin(\omega t + \varphi), \text{ Ansatz zur Bestimmung von } A, \varphi \\ \dot{y}(t) &= A\omega \cos(\omega t + \varphi) \\ \ddot{y}(t) &= -A\omega^2 \sin(\omega t + \varphi)\end{aligned}$$

Berücksichtigung der AB ohne Betrachtung von Sonderfällen

$$\begin{aligned}y_0 &= A \sin(\omega t_0 + \varphi) \\ \dot{y}_0 &= A\omega \cos(\omega t_0 + \varphi) \\ \tan(\omega t_0 + \varphi) &= \frac{\omega y_0}{\dot{y}_0} \\ \varphi &= \arctan \frac{\omega y_0}{\dot{y}_0} - \omega t_0 \\ \omega^2 y_0^2 + \dot{y}_0^2 &= A^2 \omega^2 \sin^2(\omega t_0 + \varphi) + A^2 \omega^2 \cos^2(\omega t_0 + \varphi) \\ A^2 &= y_0^2 + \frac{\dot{y}_0^2}{\omega^2} \\ A &= \frac{1}{\omega} \sqrt{(\omega y_0)^2 + \dot{y}_0^2}\end{aligned}$$

Lösung

$$y(t) = \frac{1}{\omega} \sqrt{(\omega y_0)^2 + \dot{y}_0^2} \sin\left(\omega(t - t_0) + \arctan \frac{\omega y_0}{\dot{y}_0}\right)$$

(d) DGL $\ddot{y} + \omega^2 y = 0$, $\omega = 1$, $\varepsilon = 0$ als DGL-System

$$\begin{aligned}x &\equiv \dot{y} \\ \dot{x} &= \ddot{y} = -y\end{aligned}$$

Somit

$$\begin{cases} \dot{y} = x, & y(t_0) = y_0 \\ \dot{x} = -y, & x(t_0) = \dot{y}_0 \end{cases}$$

bzw.

$$\begin{cases} \dot{x}_1 = x_2, & x_1(t_0) = x_{10} = y_0 \\ \dot{x}_2 = -x_1, & x_2(t_0) = x_{20} = \dot{y}_0 \end{cases}$$

Phasenkurve: $(y, x) = (y, \dot{y}) = (x_1, x_2)$.

Lösungsansatz mit Bestimmung von A aus AB ohne Betrachtung von Sonderfällen

$$\begin{aligned}x_1(t) &= A \cos(t + \varphi) \\ x_2(t) &= -A \sin(t + \varphi)\end{aligned}$$

$$x_1(t_0) = A \cos(t_0 + \varphi) = y_0$$

$$x_2(t_0) = -A \sin(t_0 + \varphi) = \dot{y}_0$$

$$\phi = -\arctan \frac{\dot{y}_0}{y_0} - t_0 = -\operatorname{arccot} \frac{y_0}{\dot{y}_0} - t_0$$

$$A = \sqrt{y_0^2 + \dot{y}_0^2}$$

$$\begin{aligned} y(t) = x_1(t) &= \sqrt{y_0^2 + \dot{y}_0^2} \cos\left(t - t_0 - \operatorname{arccot} \frac{y_0}{\dot{y}_0}\right) \\ &= \sqrt{y_0^2 + \dot{y}_0^2} \cos\left(t - t_0 - \left(\frac{\pi}{2} - \arctan \frac{y_0}{\dot{y}_0}\right)\right) \\ &= \sqrt{y_0^2 + \dot{y}_0^2} \sin\left(t - t_0 + \arctan \frac{y_0}{\dot{y}_0}\right) \end{aligned}$$

Spezialfälle:

$$1. y_0 = A, \dot{y}_0 = 0 \rightarrow \arctan \infty = \frac{\pi}{2}$$

$$y(t) = A \sin\left(t - t_0 + \frac{\pi}{2}\right) = A \cos(t - t_0)$$

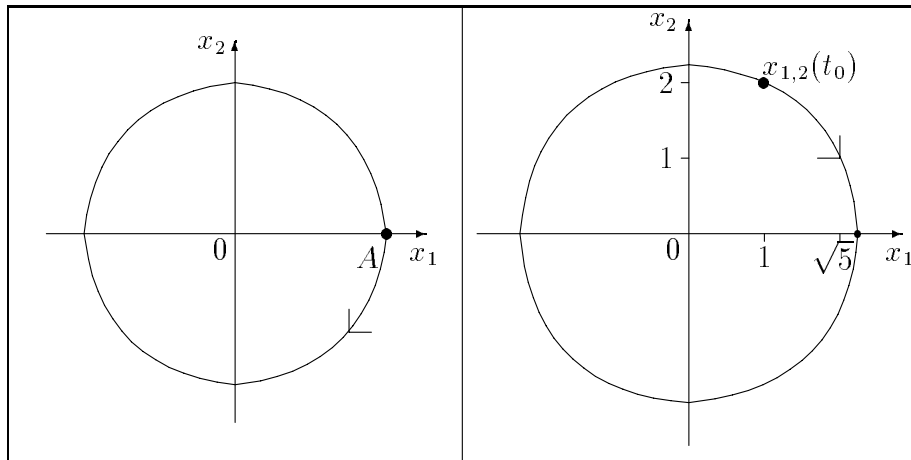
$$2. y_0 = 1, \dot{y}_0 = 2 \rightarrow A = \sqrt{5}, \varphi = -\arctan 2 - t_0 \approx -1.1 - t_0$$

$$y(t) = \sqrt{5} \sin(t - t_0 + \arctan 0.5) \approx \sqrt{5} \sin(t - t_0 + 0.46)$$

$$y(t_0) = \sqrt{5} \sin(0.46) \approx 1$$

$$\dot{y}(t) = \sqrt{5} \cos(t - t_0 + \arctan 0.5) \approx \sqrt{5} \cos(t - t_0 + 0.46)$$

$$\dot{y}(t_0) = \sqrt{5} \cos(0.46) \approx 2$$



$$1. \begin{aligned} x_1(t) &= A \cos(t - t_0) \\ x_2(t) &= -A \sin(t - t_0) \end{aligned} \quad 2. \begin{aligned} x_1(t) &= \sqrt{5} \cos(t - t_0 - 1.1) \\ x_2(t) &= -\sqrt{5} \sin(t - t_0 - 1.1) \end{aligned}$$

(e) Van der Polsche DGL

$$\ddot{y} = \varepsilon(1 - y^2)\dot{y} - y = f(t, y, \dot{y}), \quad x_1 \equiv \dot{y},$$

als System von DGL 1.Ordnung

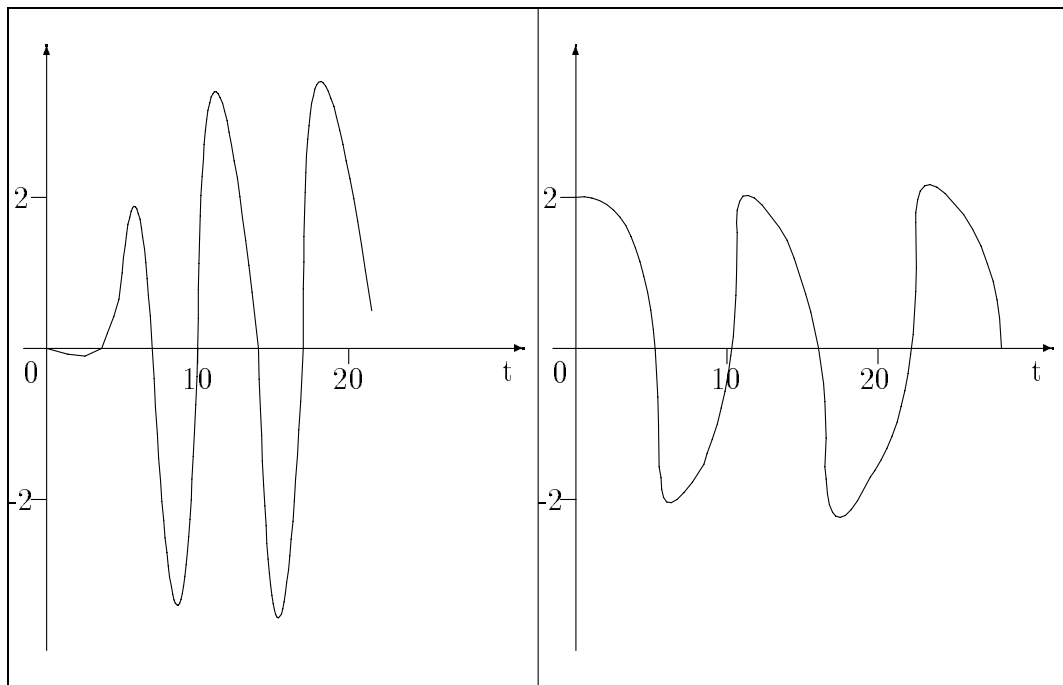
$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1 + \varepsilon(1 - x_1^2)x_2 \end{aligned}$$

Damit erhalten wir eine besondere Form der allgemeinen Störungsgleichung

$$\begin{aligned} \dot{x}_1 &= x_2 + \varepsilon f_1(x_1, x_2) \\ \dot{x}_2 &= -x_1 + \varepsilon f_2(x_1, x_2) \end{aligned}$$

mit entsprechenden Funktionen f_1, f_2 .

(f) Beispiele zum Schwingungsverhalten, qualitative Verläufe $y(t)$



$$\begin{aligned} 1. \quad & \ddot{y} - (1 - y^2)\dot{y} + y = 0 \\ & y(0) = 0, \quad \dot{y}(0) = -0.05 \end{aligned}$$

$$\begin{aligned} 2. \quad & \ddot{y} - 5(1 - y^2)\dot{y} + y = 0 \\ & y(0) = 2, \quad \dot{y}(0) = 0 \end{aligned}$$

Wächst $\varepsilon > 0$ an, so verzerrt sich die Schwingung immer mehr und die Periode vergrößert sich.

Bei $\varepsilon \gg 1$ (z.B. $\varepsilon = 100, 1000$) beschreibt die DGL Relaxationsschwingungen. Dabei wechseln sich Zeiten sehr langsamer Bewegung mit solchen sehr schneller Bewegung ab.

Der Parameter ε legt auch die Einschwingdauer fest.

2.6 Biologie und Populationsdynamik

1. Zweispezimodell - ökologisches Gleichgewicht bei Populationen

Dies wird beschrieben durch das Volterrasche System von zwei DGL 1.Ordnung

$$\begin{aligned}\dot{x} &= ax(1-y), & x &= x(t), \\ \dot{y} &= -cy(1-x), & y &= y(t),\end{aligned}$$

bzw. entkoppelt in der Form

$$\begin{aligned}x\ddot{x} &= \dot{x}^2 + acx^2 - cx\dot{x} + cx^2\dot{x} - acx^3, \\ y\ddot{y} &= \dot{y}^2 + acy^2 + ay\dot{y} - ay^2\dot{y} - acy^3.\end{aligned}$$

Löse die DGL für $a = 2$, $c = 1$ mit den AB $x(0) = 1$, $y(0) = 3$.

2. Lotka-Volterra-Modell [BRAUN]

Das originale Lotka-Volterra-Modell

$$\begin{aligned}\dot{x} &= x(1-y), \\ \dot{y} &= -y(1-x)\end{aligned}$$

ist ein Räuber-Beute-System mit der Beutepopulation/Futternvorrat $x(t)$ und der Räuberpopulation/Verbraucher $y(t)$ und kann auf eine höhere Dimension verallgemeinert werden.

Die 3D-Verallgemeinerung hat die Form

$$\begin{aligned}\dot{x}_1 &= x_1 - x_1x_2 + cx_1^2 - ax_3x_1^2 = x_1(1-x_2) + x_1^2(c-ax_3), \\ \dot{x}_2 &= -x_2 + x_1x_2 = -x_2(1-x_1), \\ \dot{x}_3 &= -bx_3 + ax_1x_3^2 = -x_3(b-ax_1x_3)\end{aligned}$$

mit $a, b, c \geq 0$ als Bifurkationsparameter.

3. Konkurrenz zweier Arten, die sich aus derselben beschränkten Resource ernähren [BRAUN]

a) $\dot{u} = u[-a_1 + c_1(1 - b_1u - b_2v)]$, $u(0) = u_0$, $c_1 > a_1$

$$\dot{v} = v[-a_2 + c_2(1 - b_1u - b_2v)], \quad v(0) = v_0, \quad c_2 > a_2$$

b) $\dot{u} = u[b_1 - \lambda_1(h_1u + h_2v)]$, $u(0) = u_0$

$$\dot{v} = v[b_2 - \lambda_2(h_1u + h_2v)], \quad v(0) = v_0$$

4. Jäger-Beute-Modell [KUBICEK, MAREK, BRAUN]

$$\text{a) } \dot{u} = u \left[R \left(1 - \frac{u}{K} \right) - \frac{Rv}{u + D} \right], \quad u(0) = u_0$$

$$\dot{v} = v \left[-S \left(1 - \frac{\gamma v}{u} \right) \right], \quad v(0) = v_0$$

$$\text{b) } \dot{x} = ax - bxy - ex^2, \quad x(0) = x_0$$

$$\dot{y} = -cy + dxy - fy^2, \quad y(0) = y_0$$

$$\text{c) } \dot{x} = ax - b\sqrt{x}y, \quad x(0) = x_0 > 0$$

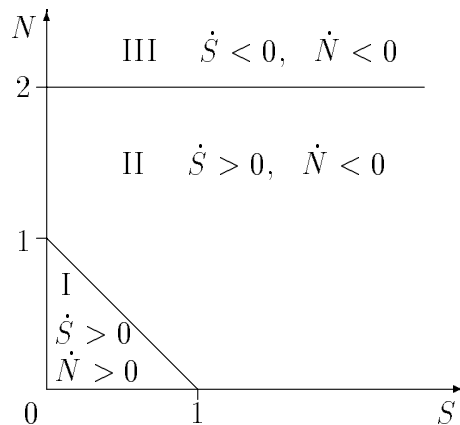
$$\dot{y} = \begin{cases} d\sqrt{x}y, & \text{falls } x \neq 0 \\ -cy, & \text{falls } x = 0 \end{cases}, \quad y(0) = y_0 > 0$$

5. Ameisen-Kampf [BRAUN]

$$\dot{S} = \frac{1}{10}S - \frac{1}{20}SN, \quad S(t_0) = S_0 > 0$$

$$\dot{N} = \frac{1}{100}N - \frac{1}{100}N^2 - \frac{1}{100}SN, \quad N(t_0) = N_0 > 0$$

- a) Man zeige, daß die Geraden $N = 2$ und $N + S = 1$ den ersten Quadranten in drei Gebiete aufteilen, in denen dS/dt und dN/dt feste Vorzeichen haben.



- b) Man verifiziere, daß jede Lösung $S(t), N(t)$, die in den Gebieten I oder III startet, schließlich nach II geht.
- c) Man beweise, daß jede im Gebiet II beginnende Lösung $S(t), N(t)$ für alle späteren Zeitpunkte dort bleibt.
- d) Man schließe aus c), daß für alle Lösungen $S(t), N(t)$ mit positiven $S(t_0), N(t_0)$ gilt $\lim_{t \rightarrow \infty} S(t) = \infty$, $\lim_{t \rightarrow \infty} N(t)$ endlich mit Grenzwert ≤ 2 .
- e) Man beachte, daß ein t_0 existiert, so daß $dN/dt \leq -N$ für $t \geq t_0$ gilt. Man schließe aus dieser Ungleichung auf $\lim_{t \rightarrow \infty} N(t) = 0$.

6. Veränderungen bei Populationen - Wachstumsmodell

Unter der Annahme, daß es eine maximale Population m gibt, die z.B. aus Gründen des Lebensraumes, nicht überschritten werden kann, sei die Bestandsänderung der Population proportional sowohl zum Bestand als auch zur jeweiligen Abweichung vom möglichen Höchststand. Dies ergibt die DGL

$$y' = f(t, y) = p y(m - y), \quad y = y(t) \geq 0, \quad \text{AB } y(0), \quad \text{Proportionalitätsfaktor } p > 0.$$

Sie wird auch als Wachstumsgleichung oder logistische Gleichung bezeichnet.

(a) Lösung des Anfangswertproblems

Hinweis: Trennung der Variablen, Idee der Partialbruchzerlegung.

$$\begin{aligned} p &= \frac{1}{y(m-y)} y' = \left[\frac{1}{my} + \frac{1}{m(m-y)} \right] y' \\ mp &= \frac{y'}{y} + \frac{y'}{m-y} \\ \int mp \, dt &= \int \frac{y'}{y} \, dt + \int \frac{y'}{m-y} \, dt \\ mpt + c &= \ln y - \ln(m-y) = \ln \frac{y}{m-y} \\ c_1 e^{mpt} &= \frac{y}{m-y}, \quad c_1 = e^c > 0 \end{aligned}$$

Somit haben wir eine Bedingung für das Wachstum, die nach y aufzulösen ist.

$$y(t) = \frac{c_1 m e^{mpt}}{1 + c_1 e^{mpt}}, \quad c_1 \in \mathbb{R}^+$$

Unter Berücksichtigung der Anfangsbedingung ergibt sich folgende Gleichung für c_1 .

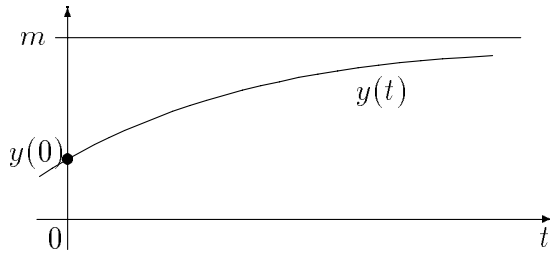
$$y(0) = \frac{c_1 m}{1 + c_1} \quad \text{und somit} \quad c_1 = \frac{y(0)}{m - y(0)}$$

Die Wachstumsgleichung hat die endgültige Form

$$y(t) = \frac{y(0) m e^{mpt}}{m - y(0) + y(0) e^{mpt}}.$$

(b) Eigenschaften der DGL und Lösung

$$y(t) = \frac{y(0) m e^{mpt}}{m - y(0)(e^{mpt} - 1)}, \quad t \in [0, T], \quad \text{AB } y(0) \in (0, m), \quad p > 0$$



- Überprüfe folgende Aussagen zur Lösung.

$$y(0) \leq y(t) < m, \quad m = \lim_{t \rightarrow \infty} y(t)$$

$$y(t_1) < y(t_2) \quad \text{für } t_1 < t_2$$

$$y'(t) > 0, \quad y'(0) = p(m - y(0))y(0)$$

- Was passiert, wenn die AB $y(0) = 0$ bzw. $y(0) = m$ ist?
- Charakterisiere das Richtungsfeld der DGL.

In jedem Punkt $P(t, y)$ des Streifens $\Omega = [0, T] \times [0, m] \in \mathbb{R}^2$ kann man die Steigung y' durch Eintragen eines kleinen „Strichs“ für die Tangentenrichtung andeuten. Ein solcher Strich heißt *Linienelement* oder *Richtungselement* (t, y, y') der Lösungskurve. Die Gesamtheit aller Linienelemente heißt *Richtungsfeld*.

Nimm zwecks Vereinfachung die Parameter $p = m = 1$.

Betrachte auch das Richtungsfeld außerhalb des Streifens.

- (c) Löse das AWP im Bereich mit zulässigem Startpunkt $y(0)$ mittels Polygonzugverfahren (EULER-CAUCHY-Verfahren)

$$y_{n+1} = y_n + hf(t_n, y_n), \quad n = 0, 1, \dots, \quad t_n = nh, \quad y_0 = y(0),$$

Schrittweite $h > 0$ gegeben.

Mache einen graphischen Vergleich der exakten Lösung mit dem Lösungsverlauf des Näherungsverfahrens (Polygonzug). Teste verschiedene Schrittweiten h .

Wie ist h zu wählen, damit z.B. analog zu den Eigenschaften der exakten Lösung die Bedingungen

$$y_n < y_{n+1} < \dots < m$$

erfüllt werden.

Rechne das PZV auch mit Anfangsbedingungen, die außerhalb des Streifens liegen.

7. Modellierung eines Ökosystems

Das typische Räuber-Beute-Modell kann auf der Grundlage von gewöhnlichen Differentialgleichungen beschrieben werden. Dazu nimmt man an, daß

- die räumliche Verteilung der Spezies keine Rolle spielt,
- die Modellierung mit reellen Zahlen aufgrund einer hohen Anzahl von Individuen sinnvoll ist,
- eine Normierung der Anzahlen auf 1 vorgenommen wird.

Bezeichnet man die Menge Gras, die Zahl der Hasen und die Zahl der Füchse mit g, h bzw. f , so kann man folgendes Modell (System von DGL) aufstellen.

$$\begin{aligned}g'(t) &= 1 - h(t)g(t), \\h'(t) &= h(t)(g(t) - f(t)), \\f'(t) &= f(t)h(t) - c_1f(t) - c_2\sqrt{f(t)}, \quad c_1 + c_2 = c = \text{const} > 0, \quad c_i \geq 0.\end{aligned}$$

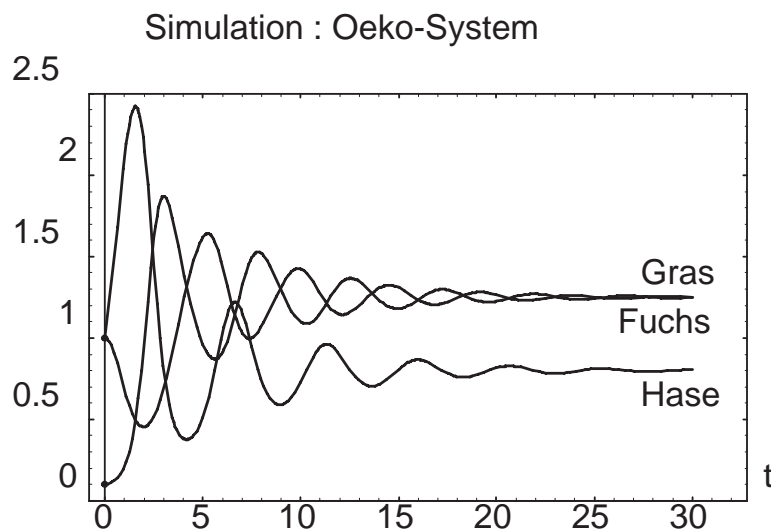
Mögliche Anfangsbedingungen (AB) sind $g(0) = h(0) = 1, f(0) = 0.1$.

Die erste Gleichung beschreibt das Wachstum der Grasmenge und das "Gefressenwerden von Gras" mit einer Rate, die proportional zum Produkt aus Hasen und Gras ist. In der zweiten Gleichung für Hasen ist die Wachstumsrate ebenfalls mit $h(t)g(t)$ gegeben. Zusätzlich geht die Zahl der Füchse "hasenmindernd" mit $-h(t)f(t)$ ein. In der Gleichung der Füchse ist ihr Wachstum von den vorhandenen Hasen abhängig, während der negative Term das Jägerverhalten simulieren soll. Der Abschub erfolgt proportional zur Anzahl der Füchse.

(a) Simulationslauf

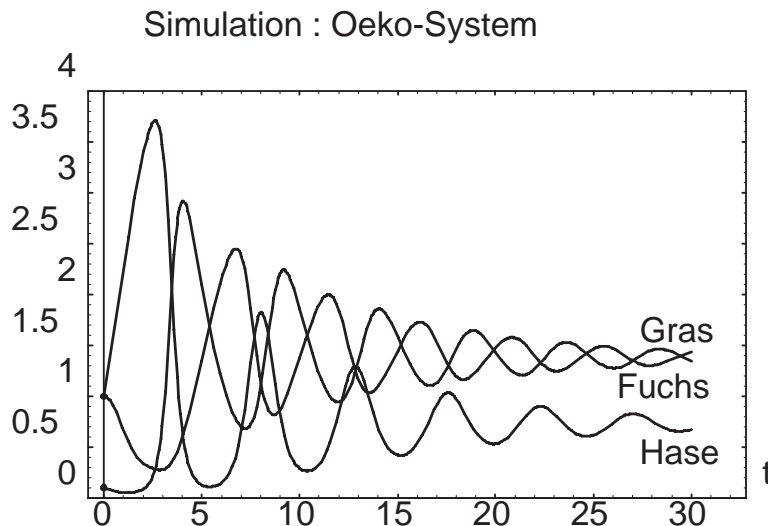
Das System tendiert zu Schwingungen. Dies werden jedoch mit der Zeit herausgedämpft, und es stellt sich ein Gleichgewicht ein.

Beispiel 1: $c = c_1 = 0.8, c_2 = 0$



Beispiel 2: $c = 0.8$, $c_1 = 0.3$, $c_2 = 0.5$

Wegen $1 > \sqrt{f(t)} > f(t)$ am Anfang haben wir eine höhere Abschußquote der Füchse als im ersten Beispiel. Die Zahl der Füchse wird so stark dezimiert, so daß es zu einer richtigen Hasenplage kommt. Die Folge dazu ist wieder ein starker Rückgang der Grasmenge. Insgesamt sind die Ausschläge größer.



(b) Stationäres Gleichgewicht

Bei hinreichend großer Zeit stellt sich für die Anzahlen der Populationen ein Gleichgewicht ein. Berechne die Gleichgewichtsgrößen $g_\infty = g(t = \infty)$, h_∞ und f_∞ aus den Bedingungen des stationären Verhaltens $g' = h' = f' = 0$.

(c) Numerik

Die numerische Lösung des Systems der DGL 1.Ordnung

$$y' = F(t, y), \quad y : \mathbb{R} \rightarrow \mathbb{R}^3,$$

erfolgt mittels Polygonzugverfahren (EULER-CAUCHY-Verfahren)

$$\begin{aligned} y_{n+1} &= y_n + \Delta t F(t_n, y_n), \quad n = 0, 1, \dots, N, \\ t_n &= n\Delta t, \quad \text{Schrittweite } \Delta t > 0 \text{ gegeben,} \\ \text{AB } y_0 &= y(0) \text{ gegeben.} \end{aligned}$$

Man wähle geeignete AB y_0 , berechne mittels PZV die Näherungslösungen und kontrolliere ihren Verlauf.

Stelle die Lösungen graphisch dar. Teste verschiedene Schrittweiten Δt .

Die Abbildungen sind mit *Mathematica* erstellt.

2.7 Medizin

1. Modell der Diabeteserkennung [BRAUN]

$$\dot{u} = F_1(u, v) + J(t), \quad u(t) \text{ Konzentration der Glukose im Blut}$$

$$\dot{v} = F_2(u, v), \quad v(t) \text{ Insulinkonzentration}$$

$J(t)$ externe Rate des Anstiegs der Blutzuckerkonzentration

2. Epidemiologie [BRAUN]

Einfaches Modell zur zeitlichen Verbreitung von Epidemien ohne Betrachtung von Migrationen der Gesamtpopulation.

Fragen: Entsteht eine Epidemie oder nicht? Wann stirbt sie aus?

$$\dot{y}_1 = -ry_1y_2, \quad y_1 \text{ aufnahmefähige Population}$$

$$\dot{y}_2 = ry_1y_2 - \gamma y_2, \quad y_2 \text{ infizierte Population}$$

$$\dot{y}_3 = \gamma y_2, \quad y_3 \text{ resistente bzw. gestorbene Population}$$

3. Modell eines Immunsystems [KUBICEK, MAREK]

Das Modell der Abwehr des Immunsystems gegen Krebs beschreibt die zeitliche und räumliche Entwicklung der Zellular-Population $X(t, r)$, $M_0(t, r)$, $M_1(t, r)$.

$X(t, r)$ Konzentration der normalen Zellen

$M_{0,1}(t, r)$ Konzentrationen der zytotoxischen Zellen

4. Zellwachstum

Die DGL für explosionsartiges Wachstum von Zellen habe die Form

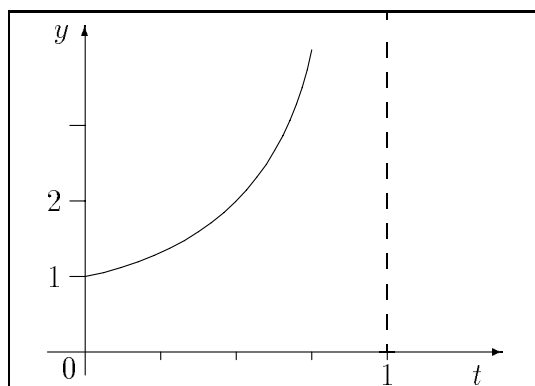
$$y' = y^2, \quad y(0) = c > 0, \quad t \geq 0.$$

Lösung

$$\frac{y'}{y^2} = -\left(\frac{1}{y}\right)' = 1, \quad y \neq 0$$

$$\frac{1}{y} = -t + C, \quad \text{AB beachten}$$

$$y(t) = \frac{1}{-t + 1/c}$$



$$\text{Bsp.: } c = 1, \quad y(t) = \frac{1}{-t+1}$$

Analytische Eigenschaften der Lösung beachten, hier z.B. Polstelle bei $t = 1$.

5. Tumorwachstum [BRAUN]

Die einfache Dynamik des Zellwachstums kann beschrieben werden durch die DGL

$$y' = \lambda y, \quad y(t_0) = y_0, \quad \lambda > 0 \text{ Wachstumsrate.}$$

Gemäß ihrer Lösung $y(t) = y_0 \exp(\lambda(t - t_0))$ wachsen die sich teilenden Zellen exponentiell an, wobei sich das Zellvolumen $y(t)$ während eines jeden Zeitintervalls der Länge $\ln 2/\lambda$ verdoppelt.

Empirische Untersuchungen haben gezeigt, daß die Zunahme des Zellvolumens eher der Bedingung

$$y(t) = y_0 e^{\frac{\lambda}{\alpha}(1 - e^{-\alpha t})}, \quad \lambda, \alpha > 0$$

entspricht (Gompertzsche Beziehung), die für $t \rightarrow \infty$ den Grenzwert $y_0 \exp(\lambda/\alpha)$ hat.

Die dazugehörige DGL ist

$$y' = \lambda e^{-\alpha t} y,$$

die zweierlei interpretiert werden kann.

a) $y' = (\lambda e^{-\alpha t}) y$

Die Hemmung des Zellwachstums hat ihre Ursache in der mit der Zeit fallenden Wachstumsrate $\lambda \exp(-\alpha t)$.

b) $y' = \lambda (e^{-\alpha t} y)$

Die durchschnittlich zur Zellbildung nötige Zeit bleibt konstant, aber die Wachstumshemmung basiert auf dem Verlust sich reproduzierender Zellen $\exp(-\alpha t)y$.

3 Implementierung der Modelle

1. Allgemeine Hinweise

Die zahlreichen Modelle eignen sich auch für Belegaufgaben und/oder ein Computerpraktikum. Dabei können sowohl Untersuchungen zum Problem als auch die entsprechende Algorithmen- und Programmentwicklung unter Anwendung der Kenntnisse über Programmierkonzepte wie Unterprogrammtechnik, Modulkonzept, Rekursion, Programm-Tools, Units, statische und dynamische Datenstrukturen, graphische Bausteine, objektorientierte Programmierung, Entwurfsoberfläche Turbo Vision, Menütechnik, Klassenbibliotheken, Einbinden von Programmbausteinen, Hilfen u.ä. erfolgen.

Bei Vergleich von verschiedenen numerischen Methoden, wobei oft mit einfachen Verfahren begonnen wird, ist eine Aufwandsanalyse bzgl. Arithmetik, Zeit und Speicherbedarf möglich.

Empfehlenswert ist eine Aufwärtskompatibilität der Programme, d.h. zusätzliche gleichartige Aufgabenstellungen sollten ohne größeren Aufwand integrierbar sein.

Bezüglich des Programmentwurfs orientiere man sich soweit wie möglich an der Struktur des Problems. Dabei sind **3 Entwurfsmethoden** hervorzuheben.

- Funktionsorientierter Entwurf, auch schrittweise Verfeinerung oder algorithmische Dekomposition genannt.
Ausgangspunkt sind Funktionen bzw. Operationen des Systems. Implementierungsdetails, insbesondere Einzelheiten der Realisierung auf dem Rechner, werden auf die „untersten“ Module verschoben.
- Datenorientierter Entwurf.
Ausgangspunkt sind Operanden, und man orientiert sich an den Datenstrukturen.
- Objektorientierter Entwurf.
Wichtiges Element ist dabei die Verknüpfung von Operanden und Operationen zu Objekten.

2. Hinweise zur Menütechnik

Die Struktur und der Inhalt der Programmenüs können mehr oder weniger elegant gestaltet werden. Sie werden auf dem Textbildschirm oft bescheidener ausfallen als auf graphischen Oberflächen.

Der Aufbau einer hierarchischen Menüstruktur, manchmal noch mit zahlreichen Querverweisen, führt meistens zu einer erheblichen Vergrößerung des Programmcodes. Wichtig ist jedoch, daß der Softwareanwender zielgerichtet auf die eigentliche Lösung des Problems geführt wird und dabei die notwendigen Hilfestellungen erhalten kann. Oft richtet man Menü- und Statuszeilen, Anzeige-, Dialog-, Schalter- Hilfe- und Infofenster ein.

Empfohlen wird zum Beispiel, folgende Teilaufgaben des Algorithmus zu einer Einheit zusammenzufassen.

- **Initialisierung/Eingabe - Init**
 - Standardvorgaben übernehmen
 - Tastatureingabe
 - Dateieingabe
 - Auswahl von Funktionen im Quelltext
 - Funktionseingabe zur Laufzeit
 - Schalterwahl, Tastenwahl, Hotkeys
 - Korrektur und Modifikation der Daten
- **Verfahrens- und Methodenauswahl - Proc**
 - Klassifizierung der Verfahren
 - Eingabe spezieller Verfahrensparameter
- **Rechnen - Calc**
 - Speichern von Zwischenergebnissen und Ergebnissen
 - Ablaufsteuerung
 - Unterbrechungsmöglichkeiten mit Datenrettung
 - Aktivitätsanzeige
- **Anzeigen - Show**
 - Ergebnisdarstellung
 - Graphische Komponenten
 - Ausblenden von Teilen der Ergebnisse
 - Wiederholung der Anzeige/Ausgabe
- **Druck - Print**
 - Druckerauswahl und -einstellungen
- **Dateiarbeit - File**
 - Festlegung von Dateitypen
- **Einstellungen - Options**
 - Einstellungen für Text- und Graphikbildschirm
 - Cursor- und Mausbedienung
 - Rechnerspezifische Einstellungen
- **Wiederholungen - Restart**
 - Neues Beispiel rechnen
 - Anzeige der zuletzt abgespeicherten Ergebnisse
- **Rückkehr zu anderen Menüs/Programmende - Exit**

3. Illustration an einem Beispiel

Bezüglich der Programmoberfläche ist das nachfolgende Beispiel als Anregung zu sehen. Weitere Möglichkeiten für Menütechniken kann der Leser dem Skript “Praktikumsaufgaben für die Fachgebiete *Algorithmen und Programmierung*, *Grundlagen der Informatik* und *Informatik und Programmierung*“ [NEUNDORF, WALTHER] entnehmen.

Simulation des Doppelpendels aus Abschnitt 2.4.

Autor: C.Reimann, Matrikel 96 Studiengang *Technische Physik*.

Die Implementierung des Pendels ohne/mit Störung erfolgte mittels einem modularen Konzept mit objektorientierten Komponenten in der Programmiersprache *Turbo Pascal*. Es läuft auf DOS-Ebene von 80387-PC und höher. Die hardwaremäßige VESA-Unterstützung ist empfohlen, ansonsten ist das Programm UNIVBE.EXE zu laden.

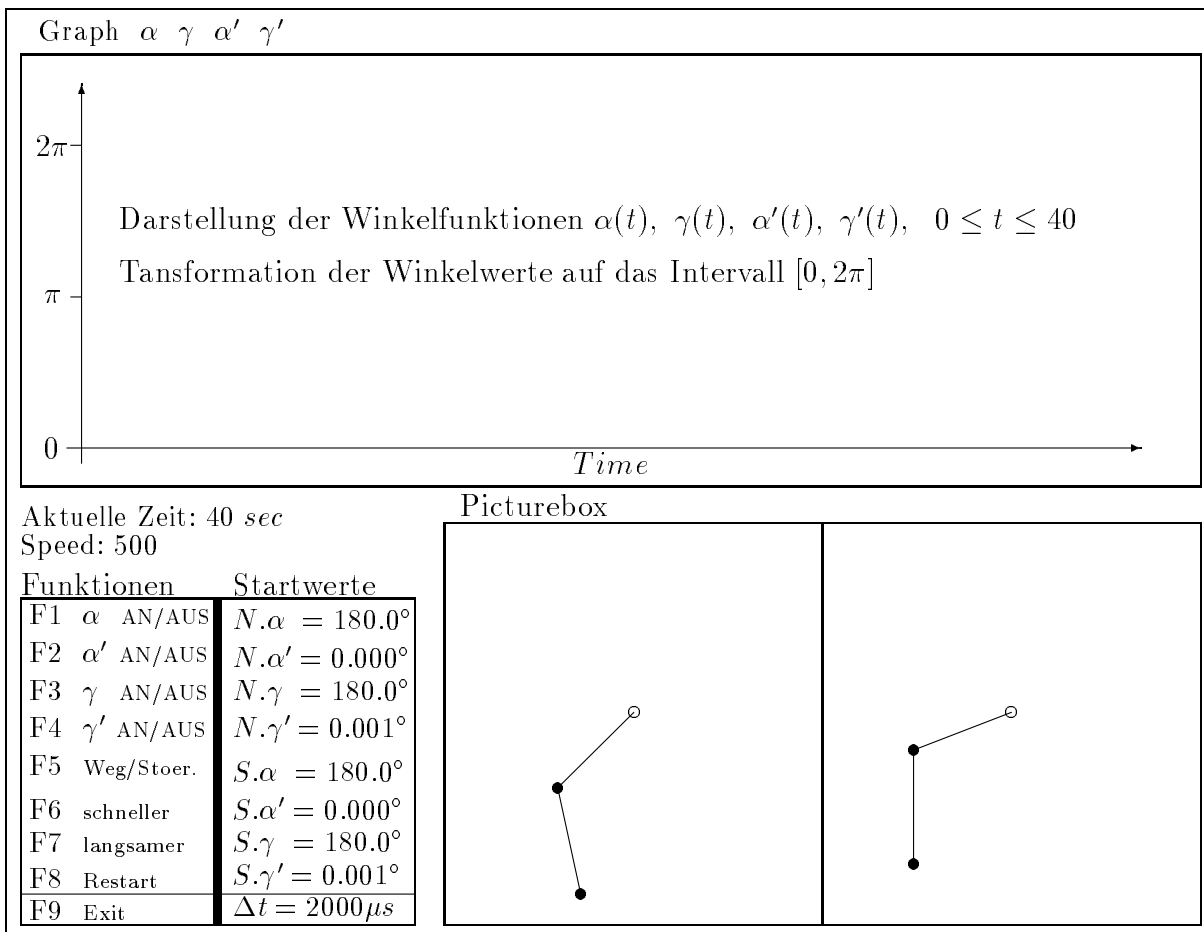
Teile des Programmsystems sind:

- DOPPEL.PAS Hauptprogramm (Rahmen)
- VESA.PAS Routinen zur Programmierung des VESA-Standards

Include-Files

- TYPE.INC Globale Typdeklarationen
- CONST.INC Globale Konstanten
- VAR.INC Globale Variablen
- FORWARD.INC Forward-Deklarationen
- INIT.INC Grundinitialisierungen
- UTILS.INC Utilities
- GRAPH.INC Graphische Oberflächenroutinen
- AUSGABE.INC Anzeigeroutinen
- PROG.INC Tastenabfrage und Programmkern
- DIFF.INC Beschreibung der Differentialgleichungen
- CALC.INC DGL-Löser (PZV)
- FILE.INC Filehandlingroutinen

Im Programm gibt es mehrere Menüs. Die Auswahl der Menüpunkte erfolgt mittels Tasten. Nach Eingabe der Problemparameter sowie Berechnung und Speicherung der Lösung kann z.B. mittels Menüpunkt **F5 Show** folgende graphische Lösungsdarstellung aufgerufen werden.



Das System von DGLn 2.Ordnung für die zeitabhängigen Winkelfunktionen ist

$$\ddot{\alpha} = \frac{-2 \sin \alpha + \dot{\gamma}^2 \sin(\gamma - \alpha) + \cos(\gamma - \alpha)[\sin \gamma + \dot{\alpha}^2 \sin(\gamma - \alpha)]}{2 - \cos^2(\gamma - \alpha)} + s \frac{\cos \gamma \cos(\gamma - \alpha) - 2 \cos \alpha}{2 - \cos^2(\gamma - \alpha)}$$

$$\ddot{\gamma} = \frac{-2 \sin \gamma - 2\dot{\alpha}^2 \sin(\gamma - \alpha) + \cos(\gamma - \alpha)[2 \sin \alpha - \dot{\gamma}^2 \sin(\gamma - \alpha)]}{2 - \cos^2(\gamma - \alpha)} + s \frac{2 \cos(\gamma - \alpha)(\cos \alpha - 1)}{2 - \cos^2(\gamma - \alpha)}, \quad s \text{ Störungsparameter}$$

Die numerische Lösung des Anfangswertproblems wird mittels Polygonzugverfahren (EULER-CAUCHY-Verfahren) sowohl für den ungestörten als auch gestörten Fall ($N.\alpha(t), \dots$ bzw $S.\alpha(t), \dots$) durchgeführt.

Die Darstellung der berechneten Bewegung erfolgt in verschiedenen Plots

- als Bewegungsablauf der beiden Stangen im (x, z) -Koordinatensystem,
- wahlweise Darstellung der Winkelfunktionen $\alpha(t)$, $\alpha'(t)$, $\gamma(t)$, $\gamma'(t)$.

Die Geschwindigkeit des Ablaufs kann durch Steuerparameter geändert werden.

4 Allgemeine Softwarehinweise

Aus der umfangreichen Menge von Software zu DGL sollen hier nur einige Programme, Programmsysteme sowie strukturierte Modul- und Textbook-Bibliotheken ausgewählt werden. Die Auswahl erfolgte unter dem Gesichtspunkt der lokalen Verfügbarkeit.

Die vielfältigen Möglichkeiten in Computeralgebrasystemen seien nur erwähnt.

4.1 Programme

1 S_GEBIET

- **Kurzcharakteristik zum Programm S_GEBIET.PAS**
Stabilitätsgebiet von MSV für gewöhnliche DGL 1.Ordnung

- (C) W.Neundorf IfMath TUIlmenau 1995
- MS DOS
- *Programmiersprache*: Turbo Pascal V.6.0,7.0, Borland Pascal V.7.0
- *Aufruf*: S_GEBIET.EXE
- *Dokumentation*: Quelltext, selbstdokumentierend
- Freeware
- Inhalt
Graphik des Stabilitätsgebiets
eines r -stufigen BDF-Verfahrens oder Mehrschrittverfahrens n -ter Ordnung für eine gewöhnliche Differentialgleichung 1.Ordnung $y' = f(x, y)$

Das MSV hat die allgemeine Form

$$\begin{aligned}\alpha_r y_r + \alpha_{r-1} y_{r-1} + \dots + \alpha_0 y_0 &= h(\beta_r f_r + \beta_{r-1} f_{r-1} + \dots + \beta_0 f_0) \\ f_k &= f(x_k, y_k), \quad y_0, y_1, \dots, y_{r-1} \text{ gegeben}\end{aligned}$$

Spezielle einbezogene explizite und implizite MSV sind:

BDF, GEAR

$$\alpha_r y_r + \alpha_{r-1} y_{r-1} + \dots + \alpha_0 y_0 = h \beta_r f_r, \quad \beta_r = 1, \text{ falls normiert}$$

ADAMS-BASHFORTH

$$\alpha_r y_r = -\alpha_{r-1} y_{r-1} + h(\beta_{r-1} f_{r-1} + \dots + \beta_0 f_0), \quad \beta_r = 0$$

ADAMS-MOULTON

$$\alpha_r y_r = -\alpha_{r-1} y_{r-1} + h(\beta_r f_r + \beta_{r-1} f_{r-1} + \dots + \beta_0 f_0)$$

NYSTRÖM

$$\alpha_r y_r = -\alpha_{r-2} y_{r-2} + h(\beta_{r-1} f_{r-1} + \dots + \beta_0 f_0), \quad \beta_r = 0$$

MILNE-SIMPSON

$$\alpha_r y_r = -\alpha_{r-2} y_{r-2} + h(\beta_r f_r + \beta_{r-1} f_{r-1} + \dots + \beta_0 f_0)$$

• Kurzcharakteristik der Oberfläche

- Menüorientiert, wahlweise Ablaufsteuerung
- Auswahl zwischen vorgegebenen MSV und BDF (meist bis zur Ordnung 6), berechneten BDF (maximal bis zur Ordnung 20) und anderen MSV mit Koeffizienteneingabe
- Graphik des Stabilitätsgebiets des MSV mit Auswahl des Gebiets und der Skalierung
- Dateiarbeit, Stabilitätsgebiet (Rand) als Datei verwaltet
- Vergleichende Graphik aller Gebiete einer Klasse von Verfahren, soweit dazu die Dateien vorher erstellt wurden

2 RICHT_DG

• Kurzcharakteristik zum Programm RICHT_DG.PAS Richtungsfeld und AWP für gewöhnliche DGL 1.Ordnung

- (C) W.Neundorf IfMath TUIlmenau 1995
- MS DOS
- *Programmiersprache:* Turbo Pascal V.6.0,7.0, Borland Pascal V.7.0
- *Aufruf:* RICHT_DG.EXE
- *Modul:* Unit FOR_COM8.PAS für Funktionsparser $f(x, y)$
- *Dokumentation:* Quelltext, selbstdokumentierend
- Freeware
- Inhalt
Graphik des Richtungsfeldes
einer gewöhnlichen Differentialgleichung 1.Ordnung $y' = f(x, y)$ im auswählbaren Darstellungsbereich $[x_a, x_a + (j - 1)\Delta x] \times [y_a, y_a + (n - 1)\Delta y]$.
Nach Wahl Lösung eines AWP im Bereich mit zulässigem Startpunkt $y(x_0) = y_0$.
Druck von ausgewählten Bereichen des Richtungsfeldes.

- **Kurzcharakteristik der Oberfläche**

- Menüorientiert, wahlweise Ablaufsteuerung, benutzerfreundliche Oberfläche, Fenstertechnik (4 kleine oder 1 großes Fenster), Ausdünnen des Punktgitters für Darstellung (fein, normal, grob)
- Demonstrationsvariante
- Auswahl der DGL (Anzahl=20) oder Funktionseingabe und -parser für rechte Seite
- Dateiarbeit, Datei für (x, y, y') , vorhandene Beispieldateien
- Tabellarische und graphische Ausgaben
- Druck von ausgewählten Bereichen des Richtungsfeldes
- Lösung des AWP mittels klassischen RKV mit dualer Schrittweitensteuerung, Auswahl der Parameter, Informationen zum Lösungsablauf und Ergebnissen, Graph der Lösung

3 STAB_ESV

- **Kurzcharakteristik zum Programm STAB_ESV.PAS**

Stabilität von ESV bei AWP für gewöhnliche DGL 1.Ordnung

- (C) W.Neundorf IfMath TU Ilmenau 1997
- MS DOS
- *Programmiersprache:* Turbo Pascal V.6.0,7.0, Borland Pascal V.7.0
- *Aufruf:* STAB_ESV.EXE
- *Modul:* Funktion $f(x, y)$ im Quelltext definiert
- *Dokumentation:* Quelltext, selbstdokumentierend
- Freeware
- Inhalt
 - Graphik des Lösungsverhaltens von einfachen ESV für die gewöhnliche Differentialgleichung 1.Ordnung $y' = f(x, y) = y(1 - y)$ (Wachstumsgleichung) im auswählbaren Argumentbereich $[0, xmax]$.
 - Test von expliziten Einschrittverfahren
 - EULER-CAUCHY-Verfahren (PZV)
 - Klassisches RK-Verfahren (RKV)
$$y_{n+1} = y_n + h\Phi(x_n, y_n; h), \quad n = 0, 1, 2, \dots,$$
mit $y_0 \geq 0$ und verschiedenen Schrittweiten $h > 0$.
- Nach Wahl
 - Lösung eines AWP im Bereich mit zulässigem Startpunkt $y(x_0) = y_0$,

- Lösungsverlauf und Stabilität
- Bifurkationsdiagramm und stationäre Punkte für ESV im Schrittweitenbereich $0 < h_{min} \leq h \leq h_{max}$, für jedes h Berechnung hinreichend vieler Werte y_n bis Erreichen des stationären Zustands, letzte Werte als Datei speicherbar

- **Kurzcharakteristik der Oberfläche**

- Menüorientiert, wahlweise Ablaufsteuerung
- Auswahl zwischen vorgegebenen ESV und Darstellungsvarianten, Auswahl der entsprechenden Parameter
- Graphik der Näherungslösung bzw. des Bifurkationsdiagramms
- Infos zum Lösungsablauf und Ergebnissen, Dateiarbeit für Bifurkationsdiagramm

4 DifEqu

- **Kurzcharakteristik zum Programm DifEqu zur Lösung von gewöhnlichen DGL**

- (C) Géza Makay Ungarn
- Download from the WWW server
<http://www.mcs.dundee.ac.uk:8080/software/index.html#DEs>
siehe auch *<http://www.math.u-szeged.hu/%7Emakay>*
- MS DOS oder Windows
- *Aufruf:* difequ.exe
- *Dokumentation:* Hilfen, selbstdokumentierend
- Freeware

- **Inhalt**

DifEqu is a programm designed for numerically solving ordinary, functional and partial differential equations, difference equations and do many more things. It can be used for solving problems arising in mathematics, physics, chemistry, biology. The program ist most usefull for teaching, doing research and creating simulation. To this end DifEqu has

- A built in programming language in which the user can write a program to be executed by DifEqu.
The programming language contains:
 - * Many different variable types (long, float, text, program, file, etc.);
 - * Usual operators (+, -, *, / <, >, etc.);

- * User defined variables, vector variables, functions;
 - * Built in functions:
 - Mathematical functions (Sin, Cos, Log, Exp, Min, Max, etc.);
Solving more complicated equations faster than *Phaser* by Kocak;
 - Text handling functions (StrLen, StrMid, Asc, Chr, etc.);
 - File handling functions (OpenFile, CloseFile, Fseek, FPrint, ReadChar, ReadText, ReadLong, etc.)
 - Printing functions (Print, Printf, AnyToS, Pos, etc.);
 - Drawing functions (MoveTo, LineTo, PenSize, FgrColor, BgrColor, ChangeFont, etc.);
 - Program control functions (Execute, If, While, GetConfig, SetConfig, Abort, etc.);
 - Miscellaneous functions (Push, Pop, Clear, SaveValue, etc.)
 - * Descriptive messages and exact location of the error in a program written in then programming language;
 - * Program status button to check the values of variables or run a program by hand.
- Full context sensitive help;
 - Many ways of saving your work (position and contents of the windows on the screen, graphics as Windows BMP file, text as text file);
 - Capability of print previewing and printing your work,
 - Some examples shipped together with the program.

5 DG, DG87

- **Kurzcharakteristik zu DG Version 1.1 1994**
Numerische Lösung von GDGL und DGL-Systemen 1.Ordnung
 - (C) Frank Schilder IfMath TUI 1994
 - POWER-Basic V. 3.0 MS DOS
 - *Aufruf:* DG.EXE Standardversion
DG87.EXE mit mathematischen Koprozessor
DGL.EXE selbstextrahierende Datei
 - *Dokumentation:* DG.DOC
 - Shareware
- **Integrierte Verfahren für eine GDGL 1.Ordnung**
 - DORMAND-PRINCE-VERFAHREN der Ordnung 5(4) und 8(7)
 - ROSENBROCK-WANNER-VERFAHREN der Ordnung 4(3) und 6(7)
 - GRAGG-BURLISCH-STOER-VERFAHREN der maximalen Ordnung 16
 - SEULEX-EXTRAPOLATIONSVERFAHREN mittels linear impliziten EULER-VERFAHREN der maximalen Ordnung 8
- **Kurzcharakteristik der Oberfläche**
 - Integration kleiner und mittlerer AWP
 - Menüorientiert, wahlweise Ablaufsteuerung, benutzerfreundliche Oberfläche
 - On-Line-Hilfesystem
 - Funktionseingabe und -parser, Dateiarbeit, vorhandene Beispieldateien
 - Tabellarische und graphische Ausgaben

6 APPLI unter Windows

- **Kurzcharakteristik zu APPLI**
Gewöhnliche Differentialgleichungen
 - (C) Falk Nedwal IfMath TUI 1993, 94
 - TP für Windows 1.5 oder BP
 - *Aufruf:* APPLI.EXE
 - *Dokumentation:* README8.TXT
 - Shareware

- **Integrierte Verfahren für GDGL**

- RUNGE-KUTTA-VERFAHREN der Ordnung 8 (RKV(8))
- Eingebettetes RKV(8,7)
- ROSENBRock-WANNER-VERFAHREN der Ordnung 5(6) (ROW(5,6))
- Eingebettetes ROW(3,4)
- SEULEX-EXTRAPOLATIONSVERFAHREN mittels linear impliziten EULER-VERFAHREN

- **Kurzbeschreibung des AWP**

- DGL oder DGL-System 1.Ordnung, $n \geq 1$, $t \in [t_0, t_1]$, $x_i = x_i(t)$

$$x'_1 = f_1(t, x_1, x_2, \dots, x_n), \quad x_1(t_0) = x_{1,0}$$

$$x'_2 = f_2(t, x_1, x_2, \dots, x_n), \quad x_2(t_0) = x_{2,0}$$

.....

$$x'_n = f_n(t, x_1, x_2, \dots, x_n), \quad x_n(t_0) = x_{n,0}$$

Variablen können auch anders benannt sein, z.B. $y'_1(x), \dots$

- Fortsetzung der Integration ab t_1 möglich
- Zusätzlich Anzeige von gewünschten 'Observationen',
z.B. $O_1(t) = x_1^2 + x_2^2$ bei Phasenkurven, $O_2(t) = \sum_{i=1}^n x_i$
- Tabellarische Ausgabe

7 DELTA_60

- **Kurzcharakteristik zu DELTA_60**

**Dynamische Analyse von gewöhnlichen Differentialgleichungen
2. und 3.Ordnung, Software zur angegebenen Literatur**

- (C) W. Büntig Fak. EI TUI 1992
- TP 6.0 für MS DOS
- *Aufruf:* DELTA.EXE
- *Dokumentation:* DELTA1.DOC
- *Literatur:* E.PHILIPPOW, W.BÜNTIG: Analyse nichtlinearer dynamischer Systeme der Elektrotechnik. Carl Hanser Verlag München 1992.
- Shareware

- **Hauptmenü**

- Wahl der DGL
- Integration der DGL
- Vorbereitung der Fixpunktsuche
- Suche der Fixpunkte
- Fixpunktverfolgung bei Parameteränderung
- Harmonische Analyse
- α - und ω - Invarianzkurven bestimmen
- Leistungsdichtespektrum bestimmen

- **Kurzcharakteristik der Oberfläche**

- Text- und Graphikbildschirme
- Graphik, Speichern und Laden von Bildern
- Wahlweise Ablaufsteuerung
- Auswahlkatalog von vorhandenen DGL, aktuell 16 DGL

z.B. Problem Nr.1 : Van der Polsche DGL

$\ddot{y} - \varepsilon(1 - y^2)\dot{y} + y = 0$, $y(0) = y_0$, $\dot{y}(0) = \dot{y}_0$, Parameter ε legt die Einschwingdauer fest.

Als DGL-System

$$\begin{aligned} dx_0/dt &= x_1, \\ dx_1/dt - p_1(1 - x_0^2)x_1 + x_0 &= 0. \end{aligned}$$

8 BVPS unter Unix

- **Kurzcharakteristik zu BVPS**

Numerische Lösung von GDGL und DGL-Systemen 1.Ordnung.

Objektorientiertes C++ Programm zur numerischen Lösung von AWP gewöhnlicher DGL-Systeme

$$\dot{x} = f(x, t), \quad x(t_0) = x_0, \quad f : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$$

- Lit.: F.SCHILDER: Ein portables C++ Programm zu numerischen Integration von Anfangswertproblemen. Preprint M 1/97 TUI März 1997.
- (C) Frank Schilder IfMath TUI 1996
e-mail: F.Schilder@Mathematik.TU-Ilmenau.de
- C++ in Unix, auch für Gnu-C++, Borland-C++ Compiler

- *Aufruf*: bvps dateiname
- *Dokumentation*: bvps.txt
- Freeware
- siehe auch
<http://imath.mathematik.tu-ilmenau.de/~fschild/>

- **Integrierte Verfahren**

- 14 eingebettete explizite ESV hoher Konvergenzordnung
 - 4 Verfahren von DORMAND-PRINCE
 - 4 Verfahren von FEHLBERG, (RKF-Verfahren)
 - 4 Verfahren von VERNER
 - Verfahren von MERSON
 - Verfahren von ENGLAND
- 2 robuste eingebettete linear implizite ROSENBROCK-WANNER-VERFAHREN insbesondere für steife DGL
- 3 Extrapolationsverfahren mit Ordnungs- und Schrittweitensteuerung
 - LIMEX mittels linear impliziter Mittelpunktsregel
 - SEULEX mittels linear impliziten EULER-VERFAHREN
 - GRAGG-BURLISCH-STOER (GBS)

- **Kurzcharakteristik der Oberfläche**

- Integration mittlerer und größerer Systeme, Grenze für n durch Hardware
- Einfache Bedienbarkeit
- Unkomplizierte plattformunabhängige Oberfläche
- Ein- und Ausgaben in ASCII-Format, Dateiarbeit, vorhandene Beispieldateien
- Integrierter komfortabler Funktionsparser für $f(x, t)$
- Tabellarische und graphische Ausgaben, Ausgabedateien können mit Graphikprogramm *GnuPlot* verarbeitet werden.

4.2 Programmsysteme

1 MAYA

- **Kurzcharakteristik zu MAYA Version 1.0**

- Numerische Methoden**

- Lit.: K.KOSE, R.SCHRÖDER, K.WIELICZEK:
Numerik sehen und verstehen. Vieweg Braunschweig 1992
 - (C) Frank Schilder IfMath TUI 1994
 - MS DOS
 - *Aufruf*: MAYA.EXE
 - *Dokumentation*: Selbstdokumentierend
 - Shareware
 - Inhalt
 - Funktionen, Interpolation, Konstruktion von Bézier-Polynomen
 - Ausgleichsrechnung
 - Chaos bei Differenzgleichungen
 - **Anfangswertaufgaben**
 - Nullstellenaufgaben, Nichtlineare Gleichungssysteme

- **Anfangswertaufgaben**

- Lösungsschar einer Differentialgleichung
 - Funktionsweise verschiedener Verfahren
 - Stabilität von Einschrittverfahren
 - Vergleich der Verfahren
 - Abhängigkeit der Lösung von den Anfangswerten
 - Zweidimensionale Anfangswertaufgaben (Systeme 1.Ordnung mit 2 DGL)
 - Einfluß der Anfangswerte bei zweidimensionalen DGL

- **Integrierte Verfahren für GDGL**

- Explizites EULER-CAUCHY-VERFAHREN (Polygonzugverfahren)
 - Implizites EULER-VERFAHREN
 - Verbessertes EULER-VERFAHREN
 - HEUN-VERFAHREN
 - TRAPEZREGEL
 - Klassisches RUNGE-KUTTA-VERFAHREN

- **Kurzbeschreibung des AWP**

- DGL oder DGL-System 1.Ordnung, $n = 2$, $x \in [x_0, x_1]$

$$y' = f(x, y), \quad y(x_0) = y_0 \quad \text{bzw.} \quad \begin{cases} u' = f(x, u, v), & u(x_0) = u_0 \\ v' = g(x, u, v), & v(x_0) = v_0 \end{cases}$$

- Verschiedene Graphikkomponenten

2 NUMI2

- **Kurzcharakteristik zu NUMI2 Version 1.0 1993**

Numerische Methoden

- (C) Frank Schilder IfMath TUI, M.Jungierek, S.Thiede, M.Kramer Merseburg 1993

- POWER-Basic V. 3.0 MS DOS

- *Aufruf*: NUMI2.EXE

- *Dokumentation*: HANDBUCH.TXT

- Shareware

- Inhalt

- Gaußscher Algorithmus, LU-Zerlegung, Cholesky-Zerlegung

- Iterationsverfahren für lineare GS, QR-Zerlegung

- Hessenberg-Verfahren und Nullstellen des charakteristischen Polynoms

- Newton-Verfahren im \mathbb{R} und \mathbb{R}^n

- Interpolationsverfahren, Spline-Interpolation

- Fourier-Interpolation/Entwicklung

- Tschebyscheff-Approximation

- Differenzenquotient-Verfahren für Ableitungsbestimmung

- Newton-Cotes-Formeln, Gauß-Quadratur

- **Verbessertes Polygonzugverfahren** für DGL 1.Ordnung

- **Klassisches Runge-Kutta-Verfahren** für DGL 1.Ordnung

- Jacobi-Verfahren für EWP

- **Integrierte Verfahren für eine GDGL 1.Ordnung**

- Verbessertes POLYGONZUGVERFAHREN

- Klassisches RUNGE-KUTTA-VERFAHREN

- **Kurzcharakteristik der Oberfläche**

- Menüorientiert, wahlweise Ablaufsteuerung, NUMI-Arbeitsblatt

- Funktionseingabe und -parser, Dateiarbeit

- Graphische Darstellungen

4.3 Strukturierte Modulbibliotheken

A. Turbo Pascal Numerical Methods Toolbox

Borland International, Inc. 1986, 87

D:\USER\NEUNDORF\MAT123 (lokal)

Q:\STUD_M93\MAT123 (Novell-Netz MATH1 des IfMath)

H:\LANG\TP70\NEUNDORF\MAT123 (RZ-Novell-Netz BLANC)

1. Lineare Algebra
2. Nullstellen von Funktionen
3. Interpolation von Funktionen
4. Approximation von Funktionen
5. Numerische Differentiation und Integration
6. Minimierung und Maximierung von Funktionen

7. Gewöhnliche Differentialgleichungen und -systeme

- **Anfangswertprobleme** (*Unit: INITVAL.PAS*)
 - **DGL 1.Ordnung mit Runge-Kutta-Verfahren**
UP: InitialCond1stOrder, Programm: RUNGE_1.PAS
 - **DGL 1.Ordnung mit Runge-Kutta-Fehlberg-Verfahren**
UP: RungeKuttaFehlberg, Programm: RKF_1.PAS
 - **DGL 1.Ordnung mit Adams-Bashforth-Moulton(PECE)-Verfahren**
UP: Adams, Programm: ADAMS_1.PAS
 - **DGL 2.Ordnung mit Runge-Kutta**
UP: InitialCond2ndOrder, Programm: RUNGE_2.PAS
 - **DGL n-ter Ordnung mit Runge-Kutta**
UP: InitialCondition, Programm: RUNGE_N.PAS
 - **DGL-System 1.Ordnung mit Runge-Kutta**
UP: InitialConditionSystem, Programm: RUNGE_S1.PAS
 - **DGL-System 2.Ordnung mit Runge-Kutta**
UP: InitialCondition System2, Programm: RUNGE_S1.PAS
- **Randwertprobleme** (*Unit: INITVAL.PAS*)
 - **Lineares Schießverfahren für DGL 2.Ordnung**
UP: LinearShooting, Programm: LINSHOT2.PAS
 - **Nichtlineares Schießverfahren für DGL 2.Ordnung**
UP: Shooting, Programm: SHOOT2.PAS
- **Eigenwertprobleme**

8. Partielle Differentialgleichungen und -systeme
9. Integralgleichungen
10. Darstellung spezieller Funktionen

B. MATHPAK 87

Source Code for Numerical Software

Turbo Pascal 5.0/5.5/6.0 1986-90

Precision Plus Software

939 Griffith Street London, Ontario Canada N6K 352

Literatur: MATHPAK 87 Version 3.0

High Performance Software für den 80x87

Benutzer- und Referenzhandbuch

SOS Software Service GmbH

Alter Postweg 10, Augsburg

O:\MATPAK (Novell-Netz MATH1 des IfMath)

H:\LANG\TP70\NEUNDORF\MATPAK (RZ-Novell-Netz BLANC)

C. TPNUM 3.21 (1993)

Tests, Unterprogramme und Units zur Numerischen Mathematik

Turbo Pascal, Borland Pascal 7.0

Literatur: DIETEL, J. (Rechenzentrum der RWTH Aachen):

Formelsammlung zur Numerischen Mathematik

mit Turbo Pascal-Programmen (3. Auflage),

BI-Wissenschaftsverlag 1991

Q:\STUD_M93\FORMELN\TPNUM (Novell-Netz MATH1 des IfMath)

O:\FORMELN\TPNUM (Novell-Netz MATH1 des IfMath)

H:\LANG\TP70\NEUNDORF\FORMELN\TPNUM (RZ-Novell-Netz BLANC)

* Gewöhnliche Differentialgleichungen und -systeme

- DGLESV : Einschrittverfahren von Euler-Cauchy, von Heun und von Runge-Kutta für eine einzelne Differentialgleichung
- EINBRUKU : Runge-Kutta-Einbettungsformeln mit Schrittweitensteuerung
- IMPLRUKU : Implizite Runge-Kutta-Verfahren vom Gaußtyp
- Adaptive Anfangswertproblemlöser
 - AWP : Einbettungsformel (Runge-Kutta 2. und 3. Ordnung oder Formel von Englund 4. und 5. Ordnung)
 - RK_FEHL : Runge-Kutta-Fehlberg-Verfahren
- PRAE_KO : Prädiktor-Korrektor-Verfahren nach Adams-Bashforth-Moulton
- BUL_STOER : Extrapolationsverfahren von Bulirsch-Stoer-Gragg
- RWP : Zurückführung des Randwertproblems auf ein Anfangswertproblem

4.4 Textbook-Bibliotheken

Köckler, N.:

**Numerische Algorithmen in Softwaresystemen
unter besonderer Berücksichtigung der NAG-Bibliothek.
Teubner Stuttgart 1990.**

- 1 Lineare Gleichungssysteme
- 2 Lineare Optimierung
- 3 Interpolation und Approximation
- 4 Nichtlineare Gleichungen
- 5 Eigenwertprobleme
- 6 Numerische Integration

7 Anfangswertaufgaben

- Fehler, Konsistenz, Konvergenz, Stabilität, steife DGL-Systeme
- Einschrittverfahren, Runge-Kutta-Verfahren
NAG-Routine: D02PAF, D02BDF
- Mehrschrittverfahren, Adams-Verfahren, PECE
NAG-Routine: D02QAF
- Differentiationsverfahren, BDF
NAG-Routine: D02QDF
- Anfangswertaufgaben bei IMSL

8 Rand- und Eigenwertprobleme

- Differenzenverfahren
NAG-Routine: D02GAF
- Schießverfahren, Einfach-Schießen, Mehrzielmethode
NAG-Routine: D02HAF
- Randwertaufgaben bei IMSL

9 Partielle Differentialgleichungen

A Die FORTRAN-Programme

Anfangswertaufgaben *Routine: AWA*

Rand- und Eigenwertaufgaben *Routine: RWA*

B Die NAG-GS Graphik Bibliothek

C PAN - der schnelle Zugriff zu NAG

D Der Aufbau der NAG-Bibliothek

Rice, J.R.:

Numerical Methods, Software and Analysis. 2nd Edition. Academic Press Inc. Boston 1993.

- 1 Mathematics and computer science background
- 2 Numerical software
- 3 Errors, round-off, and stability
- 4 Models and formulas for numerical computation
- 5 Interpolation
- 6 Matrices and linear equations
- 7 Differentiation and integration
- 8 Nonlinear equations

9 Ordinary differential equations

- Introduction, difference equations, stability
- Basic method for IVP, EULER, RUNGE-KUTTA, Multistep, PECE
- Polyalgorithms, step size, order and error control
- Systems of ODE
- Software and programmes for stiff problems
ACM Algorithms, Software Packages IMSL
(IVP, Two-Point BVP), DEPAC, EPISODE
- Analysis of methods for ODE, one-step and multistep methods

- 10 Partial differential equations
- 11 Approximation of functions and data
- 12 Software practice, costs, and engineering
- 13 Software performance evaluation
- 14 The validation of numerical computations
- 15 FORTRAN

4.5 Software im Unix-Netz

Im Netz des Instituts für Mathematik der TU Ilmenau ist durch DM F.Schilder Software installiert, die unter anderem für numerische Berechnungen eingesetzt werden kann. Informationen dazu bietet die WWW-Seite

<http://www.mathematik.tu-ilmenau.de/~develop>

Dort ist auch eine Klassenbibliothek für Numerik gewöhnlicher Differentialgleichungen in C++ : *fs - classlib* sowie die Numerikprogramme

- *ivps* Numerische Approximation zu GDGL,
- *torus* Approximation invarianter 2-Tori.

Literatur

- [1] Knobloch, J.; Steigenberger, J.: Gewöhnliche Differentialgleichungen. Skriptum 100 S. TU Ilmenau 1996.
- [2] Schwetlick, H.; Kretzschmar, H.: Numerische Verfahren für Naturwissenschaftler und Ingenieure. Fachbuchverlag Leipzig Köln 1991.
- [3] Henning, K.; Kutscha, S.: Informatik im Maschinenbau, 4.Auflage. Springer-Verlag Berlin 1994.
- [4] Braun, M.: Differentialgleichungen und ihre Anwendungen. Springer-Verlag Berlin 1979, 1991, 1994.
- [5] Holmes, P.; Guckenheimer, J.: Nonlinear oscillations, dynamical systems, and bifurcation of vector fields. Springer-Verlag 1983, 1990.
- [6] Troger, H.; Steindl, A.: Nonlinear stability and bifurcation theory. Springer-Verlag 1991.
- [7] Seydel, R.: Practical bifurcation and stability analysis. Springer-Verlag New York 1994.
- [8] Seydel, R.: From equilibrium to chaos. Elsevier New York 1988.
- [9] Goering, H.; Felgenhauer, A.; Lube, G.; Roos, H.-G.; Tobiska, L.: Singularly perturbed differential equations. Mathematical Research 13. Akademie-Verlag Berlin 1983.
- [10] Lorenz, E.N.: The essence of chaos. Paperback University of Washington Press 1995.
- [11] Steeb, W.-H.: Chaos and Fractals. Algorithms and Computation. B.I. Wissenschaftsverlag Mannheim 1992.
- [12] Philippow, E.: Grundlagen der Elektrotechnik. Verlag Technik 1988.
- [13] Philippow, E.; Büntig, W.G.: Analyse nichtlinearer dynamischer Systeme der ET: Einführung in die numerische Untersuchung einfacher Systeme. Hanser München 1992.
- [14] Uhlmann, H.: Grundlagen der elektrischen Modellierung und Simulationstechnik. Akad. Verlagsgesellschaft Geest & Portig. Wiss. Monographien der ET, Bd.5 1977.
- [15] Kubicek, M.; Marek, M.: Computational Methods in Bifurcation Theory and Dissipative Structures. Springer Series in Comput. Physics 1983.
- [16] Arnold, V.I.: Obyknovennye differentsialnye uravnenia. Izd. Nauka Moskva 1971. Dt. Ausgabe : Gewöhnliche Differentialgleichungen. Springer-Verlag Berlin 1980.
- [17] Hairer, E.; Norsett, S.P.; Wanner, G.: Solving Ordinary Differential Equations, Vol. 1,2. Springer-Verlag Berlin 1987, 1991. (*Lehrbuchsammlung*)
- [18] Golub, G.H.; Ortega, J.M.: Wissenschaftliches Rechnen & Differentialgleichungen. Heldermann Verlag Lemgo 1995. Hrsg. der dt. Übersetzung: R.D.Grigorieff.

- [19] Albrecht, P.: Die numerische Behandlung gewöhnlicher Differentialgleichungen. Akademie-Verlag Berlin 1979.
- [20] Shampine, L.F.; Gordon, M.K.: Computerlösung gewöhnlicher Differentialgleichungen. Vieweg & Sohn Braunschweig 1984.
- [21] Fried, I.: Numerical Solution of Differential Equations. Computer Science an Applied Mathematics. Academic Press New York 1979.
- [22] Miranker, W.L.: Numerical Methods for Stiff Equations and Singular Perturbation Problems. Mathematics and Its Applications 5. D. Reidel Publishing Comp. Dordrecht 1981.
- [23] Collatz, L.: The Numerical Treatment of Differential Equations, 3rd Edition. Springer-Verlag Berlin 1960.
- [24] Abraham, R.H.; Shaw, C.D.: Dynamics - The Geometry of Behavior, Vol. 1, 2, 3 (Periodic, Chaotic, Global Behavior) The Visual Mathematics Library. Aerial Press, Inc. Santa Cruz 1983
- [25] Parker, T.S.; Chua, L.O.: Practical Numerical Algorithms for Chaotic Systems. Springer-Verlag New York Berlin 1989.
- [26] Stuart, A.; Humphries, A.: Dynamical Systems and Numerical Analysis. Cambridge Monographs on Applied and Computational Mathematics 2. Cambridge Univ. Press 1996.
- [27] Grigorieff, R.D.: Numerik gewöhnlicher Differentialgleichungen. Bd. I : Einschrittverfahren. Bd.II : Mehrschrittverfahren. Teubner Stuttgart 1972, 1977.
- [28] Demailly, J.-P.: Gewöhnliche Differentialgleichungen. Theoretische und numerische Aspekte. Vieweg Lehrbuch Angewandte Mathematik Braunschweig 1994.
- [29] Strehmel, K.; Weiner, R.: Numerik gewöhnlicher Differentialgleichungen. Teubner Studienbücher Mathematik. B.G. Teubner Stuttgart 1995.
- [30] Burrage, K.: Parallel and Sequential Methods for Ordinary Differential Equations. Oxford University Press Oxford 1995.
- [31] Ainsworth, M.; Levesly, J.; Light, W.A.; Marletta, M.: Theory of Numerics of Ordinary and Partial Differential Equations. Oxford University Press Oxford 1995.
- [32] Korsch, H.J.: Chaos: a program collection for the PC. Springer-Verlag Berlin 1998.
- [33] Korsch, H.J.: Chaos: A program collection for the PC. With 250 fig., many numerical experiments, and two 3 1/2MS-DOS-diskettes. Springer-Verlag Berlin 1994.
- [34] Bräuning, G.: Gewöhnliche Differentialgleichungen. Fachbuchverlag Leipzig 1964.
- [35] Werner, H.; Arndt, H.: Gewöhnliche Differentialgleichungen. Eine Einführung in Theorie und Praxis. Springer-Verlag Berlin 1986.

- [36] Wenzel, H.: MINÖL 7/1,2. Gewöhnliche Differentialgleichungen, Band 1, 2. B.G. Teubner Verlagsgesellschaft 1973.
- [37] Ross, S.L.: Differential equations, 3 Ed. John Wiley & Sons, Inc. New York 1984.
- [38] Furlan, P.: Das gelbe Rechenbuch 3 für Ingenieure, Naturwissenschaftler und Mathematiker. Verlag Martina Furlan Dortmund 1997.
- [39] Köckler, N.: Numerische Algorithmen in Softwaresystemen : unter besonderer Berücksichtigung der NAG-Bibliothek. B.G. Teubner Stuttgart 1990.
- [40] Rice, J.R.: Numerical Methods, Software and Analysis. 2nd Edition. Academic Press Inc. Boston 1993.
- [41] Engeln-Müllges, G.; Reutter, F.: Formelsammlung zur Numerischen Mathematik mit FORTRAN 77- bzw. Turbo Pascal-Programmen. B.I. Wissenschaftsverlag Mannheim 1988, 1991.
- [42] Engeln-Müllges, G.; Reutter, F.: Numerik-Algorithmen mit ANSI C-Programmen. (auch für Turbo Pascal, FORTRAN). B.I. Wissenschaftsverlag Mannheim 1993.
- [43] Kose, K.; Schröder, R.; Wieliczek, K.: Numerik sehen und verstehen. Ein kombiniertes Lehr- und Arbeitsbuch mit Visualisierungssoftware. Vieweg Braunschweig 1992.
- [44] Dietel, J.: Formelsammlung zu Numerischen Mathematik mit Turbo Pascal-Programmen (TPNUM). Rechenzentrum der RWTH Aachen 1993.
- [45] Neundorf, W.; Walther, B.: Praktikumsaufgaben für die Fachgebiete *Algorithmen und Programmierung*, *Grundlagen der Informatik* und *Informatik und Programmierung*. Skriptum TU Ilmenau 1997.
- [46] Schilder, F.: Ein portables C++ Programm zu numerischen Integration von Anfangswertproblemen. Preprint M 1/97 TUI März 1997.

Anschrift:

Dr. Werner Neundorf
 Technische Universität Ilmenau, Institut für Mathematik
 PF 10 0565
 D - 98684 Ilmenau

e-mail : neundorf@mathematik.tu-ilmenau.de