MLAN4 Serie 14

Aufgabe 1

Gegeben ist das Anfangswertproblem

(1) $$\ddot{y} + 200 \cdot \dot{y} + 156.25 \cdot y = 80 \cdot \cos(t) + 156.25 \qquad y(0) = 5 \qquad \dot{y}(0) = -100$$

a) Schreiben Sie (1) als System von Differentialgleichungen erster Ordnung.

b) Untersuchen Sie das System in a) auf Steifheit.

c) Bestimmen Sie numerisch die Lösung von a) mit Ihrer Schrittweitensteuerung und vergleichen Sie mit der exakten Lösung von a).

Aufgabe 2 (MATLAB)

Das folgende nicht–lineare Beispiel soll ein Test für *Ihre* Schrittweitensteuerung sein.

(2) $$\ddot{y} + \varepsilon \left(y^2 - 1\right) \dot{y} + y = \Gamma \cos(\omega t) \qquad \varepsilon > 0$$

Experimentieren Sie für (2) mit verschiedenen Anfangsbedingungen (z.B. $y(0) = -4.8$, $\dot{y}(0) = 0.2$), sowie mit verschiedenen Werten von ε, Γ und ω. (z.B. $\Gamma = 1.5$, $\omega = 0.2$, $\varepsilon = 0.1, 0.2, \ldots, 1$)

a) numerische Lösung mit Ihrer Steuerung `name23(...)`.

b) numerische Lösung mit `ode23(...)`

   mit `options = odeset('RelTol',1e-3);`, cf. `help`.

   Graphische Darstellung der numerischen Lösungen im Phasenporträt, sowie je in einem separaten plot Ort und Geschwindigkeit in Abhängigkeit von t.

c) Vergleichen Sie die verwendeten Schrittweiten. Graphische Darstellung der Schrittweiten in einem separaten plot.

d) Bestimmen Sie in jedem Zeitschritt, den Sie machen die entsprechende Steifigkeit $S(t)$.

   Stellen Sie $S(t)$ in einem separaten plot graphisch dar.

(2) ist die berühmte Gleichung von "Van der Pol"; es handelt sich hier um eine erzwungene Schwingung $g(t) \neq 0$, die wechselweise angeregt und gedämpft wird.

Aufgabe 3 (MATLAB)

Betrachten Sie das Beispiel aus der Theorie, p 31:

(3) $$\left\{\begin{array}{rcrcrcr} \dot{y}_1(t) & = & -0.01\,y_1(t) & + & 0.01\,y_2(t) & & \\ \dot{y}_2(t) & = & y_1(t) & - & y_2(t) & - & y_1(t)y_3(t) \\ \dot{y}_3(t) & = & y_1(x)y_2(t) & - & 100\,y_3(t) & & \end{array}\right. \quad \text{mit} \quad \left\{\begin{array}{rcl} y_1(0) & = & 0 \\ y_2(0) & = & 1 \\ y_3(0) & = & 1 \end{array}\right.$$

a) numerische Lösung mit Ihrer Steuerung `name23(...)`.

b) numerische Lösung mit `ode23(...)`

   mit `options = odeset('RelTol',1e-3);`, cf. `help`.

   Graphische Darstellung der numerischen Lösungen in Abhängigkeit von t.

c) Vergleichen Sie die verwendeten Schrittweiten. Graphische Darstellung der Schrittweiten in einem separaten plot.

d) Bestimmen Sie in jedem Zeitschritt, den Sie machen die entsprechende Steifheit $S(t)$.

   Stellen Sie $S(t)$ in einem separaten `plot` graphisch dar.

MLAN4 Lösungen Serie 14

Lösung 1

a) $z_1 = y$, $z_2 = \dot{y}$ und damit: $\dot{z} = Az + g(t)$, wobei
$A = \begin{pmatrix} 0 & 1 \\ -156.25 & -200 \end{pmatrix}$ und $g(t) = \begin{pmatrix} 0 \\ 80 \cdot \cos(t) + 156.25 \end{pmatrix}$

b) EW von A: $\lambda_1 = -199.2157$ und $\lambda_2 = -0.7843$, also $S(t) = S(0) = \frac{199.2157}{0.7843} = 253.9961$

c) exakte Lösung, $t \geq 0$:

$$z(t) = \begin{pmatrix} y_1(t) \\ y_2(t) \end{pmatrix} = 0.49 \cdot e^{\lambda_1 t} \cdot \begin{pmatrix} 1 \\ \lambda_1 \end{pmatrix} + 3.26 \cdot e^{\lambda_2 t} \cdot \begin{pmatrix} 1 \\ \lambda_2 \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} + 0.316 \cdot \begin{pmatrix} \cos(t - 0.9107) \\ -\sin(t - 0.9107) \end{pmatrix}$$

Lösung 2

Für dieses Beispiel sind keine analytischen Lösungen möglich. Auch macht es hier keinen Sinn, das Vektorfeld zu zeichnen, da das entsprechende Differentialgleichungssystem nicht autonom ist. (eine Trajektorie kann sich im Lauf der Zeit schneiden, Doppelpunkte)
Substitution: $x_1 = y$ und $x_2 = \dot{y} \Longrightarrow \dot{x} = f(t, x)$, wobei

$$f(t,x) = \begin{pmatrix} x_2 \\ -\varepsilon(x_1^2 - 1)x_2 - x_1 + \Gamma\cos(\omega t) \end{pmatrix}$$

Jacobi–Matrix

$$J(x) = \begin{pmatrix} 0 & 1 \\ -2\varepsilon x_1 x_2 - 1 & -\varepsilon(x_1^2 - 1) \end{pmatrix} \qquad x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \in \mathbb{R}^2.$$

`MATLAB` code

Lösung 3

Jacobi – Matrix

$$J(y) = \begin{pmatrix} -0.01 & 0.01 & 0 \\ 1 - y_3 & -1 & -y_1 \\ y_2 & y_1 & -100 \end{pmatrix} \qquad y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \in \mathbb{R}^3.$$

`MATLAB` code