D-MATH
Dr. Vasile Gradinaru
Marco Petrella

Serie 10

Abgabedatum: Fr. 10.05, in den Übungsgruppen

Koordinatoren: Marco Petrella, HG J 45 `marco.petrella@sam.math.ethz.ch`

Webpage: `http://metaphor.ethz.ch/x/2019/fs/401-1662-10L`

1. Kernaufgabe: Adaptive Methoden für steife Systeme

Aufgabenstellung

a) Implementieren Sie die Rosenbrock-Wanner Methoden der Ordnung 2 und 3. Es sollen Funktionen `row_2_step(f, Jf, yi, h)` und `row_3_step(f, Jf, yi, h)` geschrieben werden, die ausgehend vom Wert $y_i(t_i)$ genau einen Zeitschritt h der entsprechenden Methode berechnen und die Propagierte $y_{i+1}(t_i + h)$ zurück geben.

Hinweis: Die Parameter sind im Template `stiff_row_Template.py` erklärt.

b) Lösen Sie die logistische Differentialgleichung:

$$\dot{y}(t) = \lambda y(t)(1 - y(t))$$

mit dem Anfangswert $y(0) = c = 0.01$ und $\lambda = 25$ bis zum Zeitpunkt $T = 2$. Benutzen Sie $N = 100$ Zeitschritte. Plotten Sie die numerischen Lösungen $y(t)_{\texttt{ROW}}$ sowie die Fehler $y(t)_{\texttt{ROW}} - y(t)$ beider Methoden gegen die Zeit. Wie gross kann λ sein, bevor der Fehler der `ROW-2` Methode einen maximalen Wert von 0.05 überschreitet?

c) Messen Sie die Konvergenzordnung beider Methoden. Benutzen Sie hierfür obige Gleichung und Anfangswerte mit $\lambda = 10$. Wählen Sie $N = [2^4, \ldots, 2^{12}]$ und berechnen Sie den Fehler zum Endzeitpunkt $T = 2$ gegenüber der exakten Lösung:

$$y(t) = \frac{ce^{\lambda t}}{1 - c + ce^{\lambda t}}$$

Plotten Sie den Fehler gegen die Anzahl Schritte doppelt logarithmisch.

d) Implementieren Sie eine adaptive Strategie basierend auf den `ROW-2` und `ROW-3` Methoden. Verwenden Sie als Fehlerschätzer die Norm:

$$\varepsilon_i := \|y(t_i)_{\texttt{ROW-2}} - y(t_i)_{\texttt{ROW-3}}\|_2$$

Wählen Sie den initialen Zeitschritt als $h_0 = T/(100\,(\|f(y_0)\|_2 + 0.1))$ und passen Sie die Grösse des nächsten Zeitschritts durch Verkleinern ($h_{j+1} = \frac{h_j}{2}$) oder Vergrössern ($h_{j+1} = 1.1 h_j$) an.

e) Testen Sie die Implementation wiederum an der logistischen Differentialgleichung mit $\lambda = 50$. Wie viele Zeitschritte werden insgesamt zur Lösung benötigt? Plotten Sie die numerische Lösung $y(t)_{\texttt{ADA}}$ sowie die Fehler $y(t)_{\texttt{ADA}} - y(t)$ gegen die Zeit.

Bitte wenden!

f) Lösen Sie das folgende gekoppelte System:

$$\dot{y}_0(t) = -76\, y_0(t) - 25\sqrt{3}\, y_1(t)$$
$$\dot{y}_1(t) = -25\sqrt{3}\, y_0(t) - 26\, y_1(t)$$

mit Anfangswerten $y_0(0) = 1$ und $y_1(0) = 1$ bis zum Zeitpunkt $T = 1$ mit dem adaptiven Verfahren und einer Anfangsschrittweite von $h = 0.1$, plotten Sie $y(t)$.

g) Lösen Sie die folgende sehr steife Gleichung:

$$\dot{y}(t) = \lambda y^2(t)(1 - y^2(t))$$

mit dem Anfangswert $y(0) = 0.01$ und $\lambda = 500$. Plotten Sie die numerische Lösung $y(t)_{\texttt{ADA}}$ sowie die Grösse der Zeitschritte gegen die Zeit. Wie viele Zeitschritte benötigt dieses Verfahren und was ist der kleinste Zeitschritt? Wie viele Zeitschritte dieser Grösse würde ein nicht-adapives Verfahren benötigen?

2. *Broyden Verfahren*

Schreiben Sie Code, die die Plots aus dem Beispiel 3.9.2 (Broyden-Quasi-Newton-Verfahren: Konvergenz) und Beispiel 3.9.5 (Broyden-Verfahren für ein grosses nicht-lineares System), reproduzieren.
Implementieren Sie zusätzlich auch die teuere Version Broyden-Verfahren und fügen Sie sie dieden Bilder hinzu.

Hinweis: Diese Aufgabe besitzt kein Template.

3. *Gram-Schmidt-Verfahren und Householder Transformation*

In der Vorlesung haben wir die Householder Transformation verwendet um die **QR**-*Zerlegung* einer Matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ zu bestimmen. Ein weiteres, sehr intuitives Verfahren, das sukzessive die Spalten $\underline{a}_1, \ldots, \underline{a}_n$ mit $\underline{a}_i \in \mathbb{R}^m$, von $\mathbf{A}$ orthogonalisiert, ist das Gram-Schmidt-Verfahren. Das Gram-Schmidt-Verfahren ist ein Standardwerkzeug in Beweisen der Linearen Algebra. Die folgenden Algorithmen (in Pseudo-Code) liefern eine **QR**-Zerlegung nach dem *Gram-Schmidt-Verfahren* und dem *modifizierten Gram-Schmidt-Verfahren*:

Gram-Schmidt:

for $j = 1, \ldots, n$ **do**
  $\mathbf{v_j} = \mathbf{A}_{:j}$
  for $i = 1, \ldots, j-1$ **do**
    $\mathbf{R}_{ij} = \mathbf{q_i}^{\mathrm{T}} \mathbf{a_j}$
    $\mathbf{v_j} = \mathbf{v_j} - \mathbf{R}_{ij} \mathbf{q_i}$
  end for
  $\mathbf{R}_{jj} = \|\mathbf{v_j}\|_2$
  $\mathbf{Q}_{:j} = \frac{\mathbf{v_j}}{\mathbf{R}_{jj}}$
end for

Modifiziertes Gram-Schmidt:

for $i = 1, \ldots, n$ **do**
  $\mathbf{V}_{:i} = \mathbf{A}_{:i}$
end for
for $i = 1, \ldots, n$ **do**
  $\mathbf{R}_{ii} = \|\mathbf{v_i}\|_2$
  $\mathbf{q_i} = \frac{\mathbf{v_i}}{\mathbf{R}_{ii}}$
  for $j = i+1, \ldots, n$ **do**
    $\mathbf{R}_{ij} = \mathbf{q_i}^{\mathrm{T}} \mathbf{v_j}$
    $\mathbf{v_j} = \mathbf{v_j} - \mathbf{R}_{ij} \mathbf{q_i}$
  end for
end for

a) Implementieren Sie die beiden Gram-Schmidt-Verfahren in `ortho.py` und verwenden Sie beide Verfahren, um die **QR**-Zerlegung der Matrix $\mathbf{Z} \in \mathbb{R}^{50 \times 50}$ mit den Einträgen:

$$\mathbf{Z}_{ij} = 1 + \min(i, j), \quad 0 \leq i, j < 50$$

zu bestimmen.

b) Vergleichen Sie die Güte der beiden Gram-Schmidt-Verfahren in Bezug auf die Orthogonalität der Spalten von **Q**.

c) Warum sind die Gram-Schmidt-Verfahren im Gegensatz zur Householder-Transformation (siehe `ortho.py`) ungeeignete numerische Methoden zur Berechnung von **QR**-Zerlegungen?