IWR, Universität Heidelberg — Sommersemester 2011
Aufgabenblatt 3 — 11. Mai 2011

Übungen zur Vorlesung

Modellierung und Simulation in den Neurowissenschaften

http://conan.iwr.uni-heidelberg.de/teaching/numsimneuro_ss2011

Dr. S. Lang, D. Popović

Abgabe: 18. Mai 2011 in der Übung

Übung 5 Lösen von linearen Gleichungssystemen mit Octave (5 Punkte)

Octave besitzt einige built-in Integratoren zum Lösen von Differentialgleichungs-Systemen. Einer der bekanntesten ist *lsode* von Alan C. Hindmarsh.

Machen Sie sich per Hilfe-Funktion oder anderer Dokumentation mit der Funktion `lsode` vertraut und lösen Sie anschließend das folgende System gewöhnlicher Differential-Gleichungen:

$$
\begin{aligned}
\partial_t x_1 &= 5 \cdot (x_2 - x_1 \cdot x_2 + x_1 - 10^{-6} \cdot x_1^2) \\
\partial_t x_2 &= (x_3 - x_1 \cdot x_2 - x_2)/5 \\
\partial_t x_3 &= 0.1 \cdot (x_1 - x_3)
\end{aligned}
$$

mit den Startwerten $(x_1(0), x_2(0), x_3(0))^t = (0.1, 3, 0.3)^T$. Der Aufruf von `lsode` geschieht über `y = lsode ("lsode_example1", x0, t);` wobei `x0` der Startwert und `t` ein Zeitinterval ist. Der Parameter `lsode_example1` ist der Name einer Funktion, die die Differentialgleichung implementiert. Am Besten wird diese Funktion in einer Datei gleichen Namens gespeichert (also `lsode_example1.m`). Die Datei kann für dieses Beispiel dann wie folgt aussehen:

```
 1 function xdot = lsode_example1 (x, t)
 2
 3   xdot = zeros (3,1);
 4
 8   xdot(1) = x(1)*x(2);
 8   xdot(2) = ...
 8   xdot(3) = ...
11
12 endfunction
```

Die Funktion `lsode_example1` berechnet das Differentialgleichungs-System und gibt die Lösung des Systems in einem Vektor zurück.

Plotten Sie alle drei Komponenten der Lösung unseres Systems im Bereich $x = [0, 500]$ mit einem Spacing von 2. Bei ungefähr $t = 0, 125, 250$ und $t = 365$ sollten Sie Peaks der Lösung erkennen. Mit Hilfe des `logspace`-Befehls können Sie den Plotbereich um die Peaks herum verfeinern. Exportieren Sie einen Plot der Lösung, falls möglich, mit angepasstem Bereich (ansonsten mit dem uniformen).

Übung 6 Integrate-and-Fire-Neuron mit Backward Euler und Crank-Nicolson (10 Punkte)

In dieser Aufgabe wird das *Leaky Integrate and Fire*-Modell,

$$\begin{aligned} \tau_m \partial_t v(t) &= -(v(t) - v_L) + R \cdot I(t), \\ v(t) &= v_{reset}, && \text{falls } v(t) > v_{threshold}, \\ v(0) &= v_0, \end{aligned}$$

mit zwei neuen numerischen Verfahren, *Backward Euler* (BE) und *Crank-Nicolson* (CN), numerisch gelöst. Bisher hatten wir das *Forward Euler*-Verfahren (FE) verwendet. Zur Bedeutung der Parameter im Modell siehe das erste Übungsblatt.

Für eine lineare ODE

$$\begin{aligned} \partial_t v &= a \cdot v + b(t) \\ v(0) &= v_0, \end{aligned}$$

$a \in \mathbb{R}$ eine Konstante, hat das Backward Euler-Verfahren die Form

$$\begin{aligned} k_n^{-1}(v_n - v_{n-1}) &= a \cdot v_n + b(t_n) \\ v(0) &= v_0, \end{aligned}$$

wobei v_n das numerisch berechnete Potential zum Zeitpunkt t_n und $\kappa_n := t_n - t_{n-1}$ die Zeitschrittweite beim Schritt von t_{n-1} nach t_n bezeichnen, und das Crank-Nicolson-Verfahren lautet

$$\begin{aligned} k_n^{-1}(v_n - v_{n-1}) &= \tfrac{1}{2} a \cdot (v_n + v_{n-1}) + \tfrac{1}{2}(b(t_n) + b(t_{n-1})) \\ v(0) &= v_0. \end{aligned}$$

Das Leaky-integrate-and-fire-Modell ist zwar nicht-linear (durch das Rücksetzen des Potentials beim Spiken), dennoch können die Verfahren stückweise für die Inter-Spike-Intervalle verwendet werden, wenn der Spike explizit gehandelt wird – wie bisher beim Forward Euler-Verfahren.

Für den Fall eines konstanten angelegten Stroms kann eine analytische Lösung angegeben werden:

$$v(t) = v_{steady} + (v_0 - v_{steady}) \cdot \exp\{-t/\tau_m\}$$

mit der Gleichgewichtslage $v_{steady} = v_L + R \cdot I = v_{rest} + R \cdot I$. Die Zeit t in der analytischen Lösung läuft ab dem letzten Rücksetz-Zeitpunkt (zu dem ein Spike getriggert wurde) und v_0 ist der initiale Potential-Wert zu dieser Zeit. Mit Hilfe der analytischen Lösung können wir das Konvergenzverhalten der Verfahren analysieren, also wie schnell sich der Fehler $|e| = |v - v^h|$ zu einem bestimmten Zeitpunkt bei Verkleinerung der Zeitschrittweite reduziert.

1. Implementieren Sie das Forward Euler-, Backward Euler- und Crank-Nicolson-Verfahren für das Modell mit Octave und den Parametern $R = 10$, $\tau_m = 10$, $v_L = -65$, $v_{threshold} = -50$ und $v_{reset} = -65$. Der Startwert des Potentials sei $v(0) = v_0 = v_L$. Der Strom soll konstant sein, $I(t) = I_0 = 2.0$.

   Messen Sie für alle drei Verfahren die Konvergenzrate, in dem Sie den Fehler zu einem festen Zeitpunkt betrachten (absolute Differenz der analytischen Lösung und des numerischen Potentials) und gleichzeitiger die Zeitschrittweite κ halbieren, verwenden Sie $\kappa = 0.1,\ 0.05,\ 0.025,\ 0.0125,\ 0.00625,\ 0.003125$. Passen Sie auf, dass Sie immer zum selben Zeitpunkt auswerten, den Messzeitpunkt dürfen Sie aber beliebig wählen!

2. Exportieren Sie einen doppelt-logarithmische Plot „Fehler“ über „Zeitschrittweite“, in dem Sie die Werte aller drei Verfahren eintragen. Können Sie in den Plots die Konvergenzrate ablesen? Welche Konvergenzrate ergibt sich für das jeweilige Verfahren?