

Homework 1

Introduction to Matlab / Python

Stefan Röhl

Technische Universität München, Arcisstraße 21, Munich, Germany

Email: stefan.roehl@tum.de

I. IMPLEMENTATIONS IN MATLAB

Die Verfahren wurden alle einzeln in separaten Matlab Funktionen implementiert und können mit einem function-handle (*fkt*) aufgerufen werden, um eine höhere Flexibilität zu bieten. (vgl. code/ [expEuler.m, forwardEuler.m, heun.m]). Die Kernbestandteile der Funktionen sind nachfolgend aufgelistet

```
%forwardEuler
V(k+1) = V(k) + dt*fkt(V(k),t(k));

%expEuler
V(k+1) = V(k) * exp(A(t(k))*dt) - ...
(B(V(k),t(k)) / A(t(k))) * ...
(1-exp(A(t(k))*dt));

%heun
A = fkt(V(k),t(k));
V_temp = V(k) + A*dt;
B = fkt(V_temp,t(k+1));
V(k+1) = V(k) + (A + B)/2*dt;
```

II. SOLVE FUNCTIONS

Die drei verschiedenen Verfahren zur Lösung von DGLn wurden im folgenden mit der DGL (1) getestet.

$$\frac{dV}{dt} = 1 - V - t \quad (1)$$

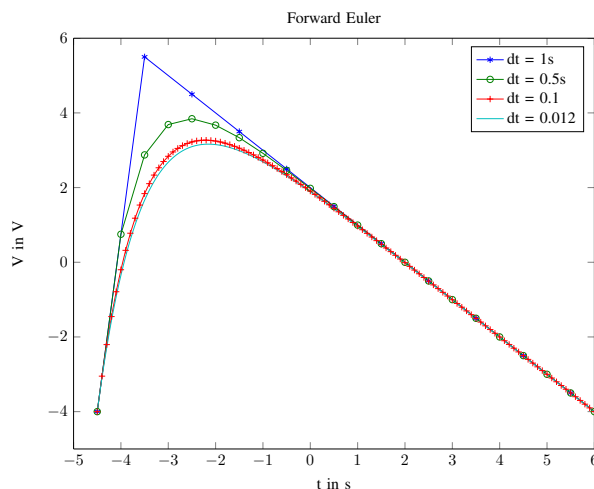


Figure 1. Forward Euler mit versch. Schrittgrößen *dt*

Beim Forward Euler ist bei großen Schrittweiten ein starkes überschießen zu beobachten (vgl. Abb. 1), was durch die große Steigung im Startpunkt verursacht wird. Bei kleineren Schrittweiten erreicht man eine bessere Näherung.

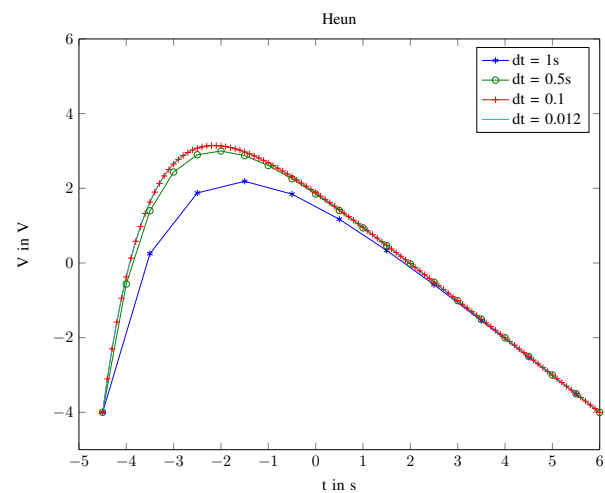


Figure 2. Verf. v. Heun mit versch. Schrittgrößen *dt*

Beim Verfahren von Heun ist genau das Gegenteil der Fall. Hier führt eine große Schrittweite dazu, dass die Näherung unterhalb der gesuchten Lösung liegt.

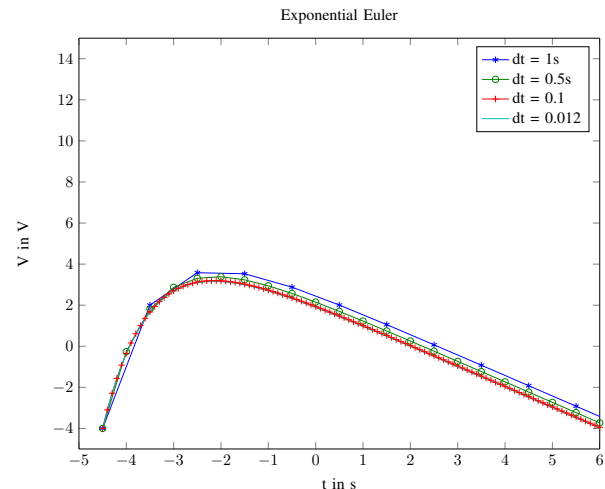


Figure 3. Exponential Euler mit versch. Schrittgröße *dt*

Fehler???

Generell ist bei der Wahl von kleinen Schrittweiten eine bessere Annäherung des Solvers an die tatsächliche Lösung der DGL zu erwarten. Das heißt die numerische Lösung stimmt besser mit der analytischen Lösung überein, denn je nach der Ordnung des Verfahrens sinkt der Fehler mit der Verkleinerung der Schrittweite umso mehr. Jedoch ist abzuwägen wie genau die Lösung für die Anwendung sein muss. Denn die Wahl einer kleinen Schrittweite bringt eine immer längere Rechenzeit mit sich. Somit ist eine generelle Wahl einer sehr kleinen Schrittweite nicht zu empfehlen.

III. THE LEAKY INTEGRATE AND FIRE NEURON

Der eingeprägte Strom wird durch das Neuron auf integriert. Das heißt die Spannung steigt immer weiter an bis sie den Threshold Wert V_{thr} erreicht hat. Bei diesem Wert wird die Zelle zu einem Spike angeregt und die komplette Energie fast instantan abgegeben. Das Potential erhöht sich auch V_{spike} und fällt sogleich auf Ruhepotential V_{rest} ab. (vgl. Abb. 4)

Im zweiten Szenario mit einem höheren (doppelt so hohem)

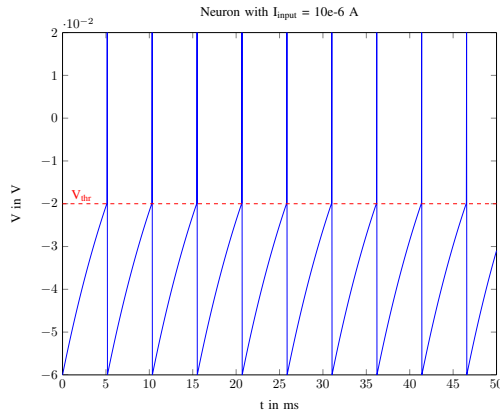


Figure 4. Potentialverlauf mit $I_{input} = 10\mu A$

eingepprägten Strom von $20\mu A$ wird V_{thr} doppelt so schnell erreicht. Daher feuert das Neuron auch mit der doppelten Frequenz. (vgl. Abb. 5)

Die Aufladung des Neurons erfolgt durch den nicht mehr konstanten Strom nun unregelmäßiger. Daher erreicht das Potential nicht mehr so schnell V_{thr} , weswegen das Neuron mit einer geringer Frequenz feuert, als mit konstantem $10\mu A$ Strom. (vgl. Abb. 6)

Durch die Amplitude von $30\mu A$ ist der Strom in der richtigen Phasenlage ($\sin \rightarrow 1$) so groß, dass das Neuron nach dem feuern und einem Abfall auf V_{rest} sehr schnell wieder die Threshold-Spannung erreicht und somit erneut feuern kann. Somit kommt es zu gehäuftem Feuern (4 Spikes) unterbrochen von kurzen Pausen. (vgl. Abb. 7)

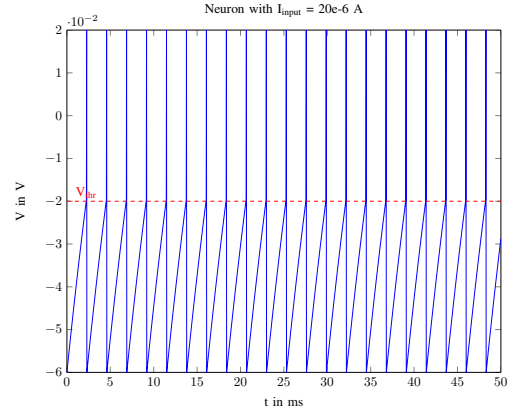


Figure 5. Potentialverlauf mit $I_{input} = 20\mu A$

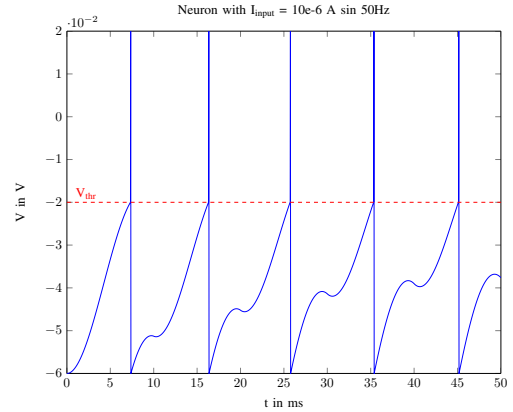


Figure 6. Potentialverlauf mit $I_{input} = 10\mu A \cdot |\sin(2\pi 50t)|$

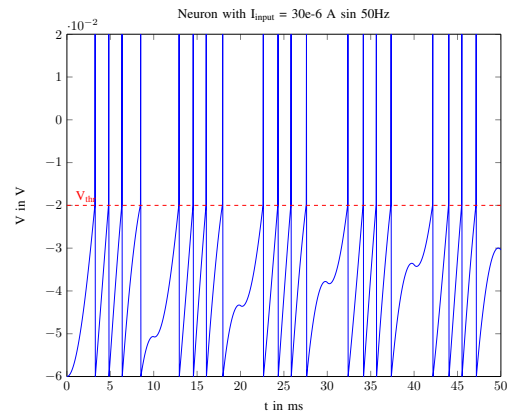


Figure 7. Potentialverlauf mit $I_{input} = 30\mu A \cdot |\sin(2\pi 50t)|$