Neuroprothetics Exercise 3
Mathematical Basics 2

Dominik Scherer

19. November 2023

1 Differential equation Solvers

Many biological systems can be modelled by differential equations. Since these are not always easy or even feasible to solve analytically, a numerical approach might be the best chance to describe a biological system. In the first section, three kinds of differential equation solvers are implemented in Python to numerically approximate the solutions of differential equations. Their performance will then be evaluated by applying them to the differential equation

$$\frac{dV}{dt} = 1 - V - t\,, \tag{1}$$

for the initial condition $V_0 = V(t = -4) = -3\,\mathrm{V}$ over a range of $9\,\mathrm{s}$ with varying step sizes Δt.

1.1 Explicit Euler

The explicit Euler solver approximates the solution of an ordinary differential equation $f(V, t)$ as

$$V_{n+1} = V_n + f(V_n, t_n)\Delta t\,. \tag{2}$$

Starting with a known initial condition V_0, the solver iterates in steps of Δt over all n in a selected time range to approximate the solution. The solver takes the initial value and adds the slope at that step multiplied by the step size to obtain the next point. The term "explicit" in the method's name just implies, that each subsequent point is calculated by already known values. Figure 1 shows how well the solver performs on equation 1 with the conditions mentioned in section 1.

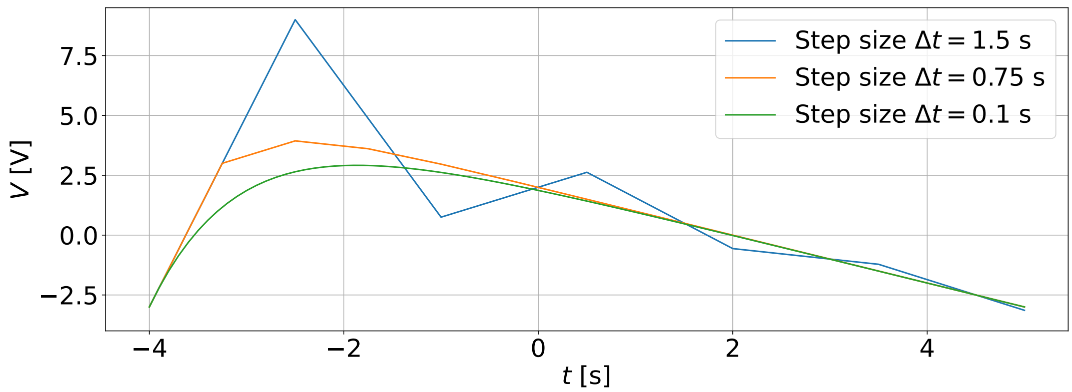

Figure 1: Simulation of the explicit Euler method on equation 1 with the initial condition $V_0 = V(t = -4) = -3\,\mathrm{V}$ over a range of 9 s with different step sizes Δt.

1.2 Heun

Unlike the Euler method, the Heun method considers two support points to approximate the next one. For that, it takes the average of the slopes at the initial point and the point that would have been the next in the Euler method, to calculate the subsequent point. With the next point in the Euler method: $\tilde{V}_n = V_n + f(V_n, t_n)\Delta t$, the actual point is then calculated by

$$V_{n+1} = V_n + \frac{f(V_n, t_n) * f(\tilde{V}_n, t_{n+1})}{2}\Delta t\,. \tag{3}$$

The solver then again iterates over all steps in the given range. Figure 2 shows the performance of the Heun method on equation 1 with the conditions of section 1.

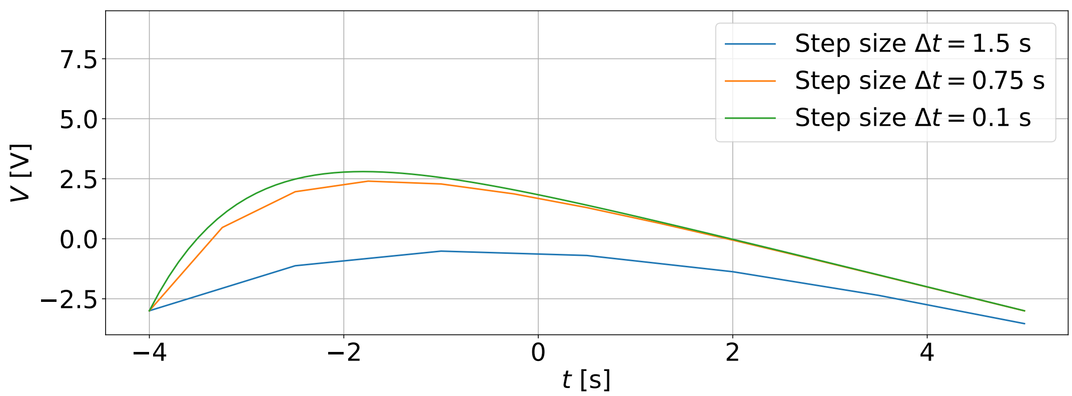

Figure 2: Simulation of the Heun method applied on equation 1 with the initial condition $V_0 = V(t = -4) = -3\,\mathrm{V}$ over a range of 9 s with different step sizes Δt.

1.3 Exponential Euler

The first-order exponential Euler takes a different approach and estimates the $n+1$ point for a differential equation of the form

$$\frac{dV}{dt} = A * V(t) + B(V,t) \tag{4}$$

in a first-order method as

$$V_{n+1} = V_n e^{A\Delta t} + \frac{B(V,t_n)}{A}(e^{A\Delta t} - 1)\,. \tag{5}$$

Figure 3 shows how the exponential Euler method performs on equation 1 with the conditions of section 1.

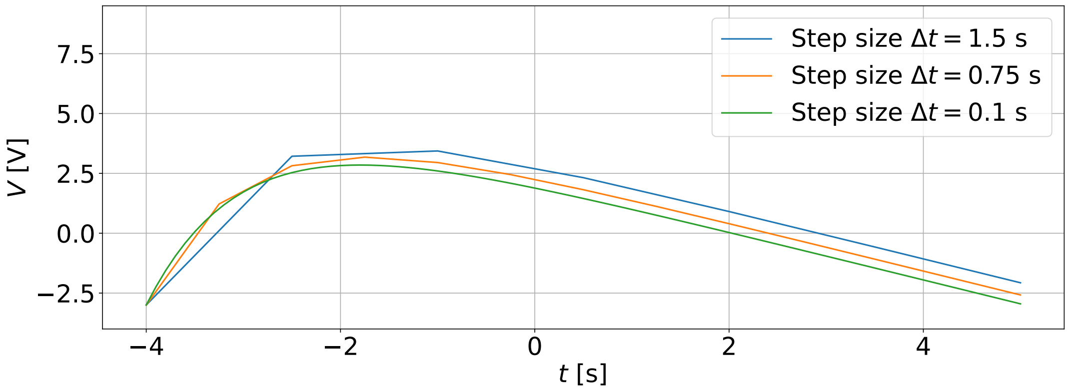

Figure 3: Simulation of the exponential Euler method on equation 1 with initial condition $V_0 = V(t = -4) = -3\,\mathrm{V}$ over a range of $9\,\mathrm{s}$ with different step sizes Δt.

2 Interpretation of solvers

2.1 Difference between solvers

The explicit Euler method only uses the function's slope at a single point n to estimate the $n+1$ point. Therefore it is a first-order solver, which can be severely affected by large deviations from the actual solution when the slope changes strongly during the first step. This is observable in the blue line in Figure 1. This overshoot can lead to the method never converging to the actual solution. In Figure 4 for example the Euler method for slightly larger step sizes will not converge to the solution but oscillate around it. The error will even increase with time when the step size is $\Delta t > 2$s.

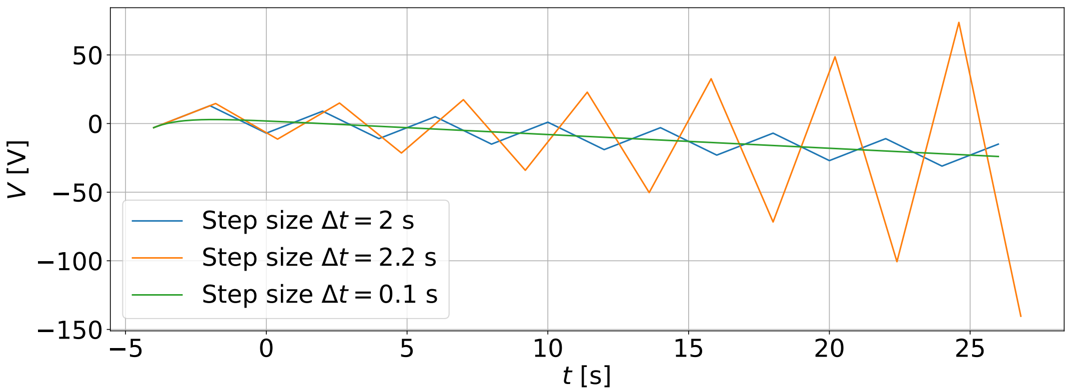

Figure 4: Simulation to show the instability of explicit Euler method for large step sizes.

The Heun method is of second order, considering two support points for each iteration, as described in section 1.2. This leads to the method approaching the real solution much "smoother" with less over- or undershooting. Even with large step sizes, the exponential Euler method is much closer to the solution in the first iterations. However, it does not seem to converge at all or much slower than the other methods with increasing iterations as shown in 3.

All models behave differently regarding stability depending on the initial value and the differential equation itself. One cannot say that there is the "one" best method since it depends on the problem.

2.2 Impact of step size

The impact of the step size varies from method to method. Besides stability considerations, the error resulting from a method depends on its order (p). The explicit Euler, for example, only considers one support point. Therefore, the method is of first order ($p = 1$), which means that the obtained error will be cut in half when halving the step size. That is because the error e scales proportionally with the step size to the power of the order: $e \propto (\Delta t)^p = \Delta t$. Respectively, the accuracy of the second-order Heun method ($p = 2$) improves quadratically with decreasing step size ($e \propto (\Delta t)^2$).

2.3 Tradeoff with small step sizes

With decreasing step size, the total number of steps required to cover the desired time interval increases. This leads to a proportional increase in the time necessary to calculate all steps. Therefore, infinitesimally small steps would require infinite computations and infinite time to calculate. In real applications, accuracy and computation time must be weighed against each other.
While increasing the accuracy of the method generally, tiny step sizes also can be a reason for inaccuracy due to the accumulation of round-off errors.

3 The Leaky Integrate and Fire Neuron model

A Leaky Integrate and Fire Neuron Model (LIF model) should be implemented in the second task. The model is described by the equation

$$V_{n+1} = \begin{cases} V_n + \frac{\Delta t}{C_m}\left(-g_{\text{leak}}\left(V_n - V_{\text{rest}}\right) + I_{\text{input}}\left(t_n\right)\right) & V_n < V_{\text{thr}} \\ V_{\text{spike}} & V_{\text{thr}} \leq V_n < V_{\text{spike}} \\ V_{\text{rest}} & V_{\text{spike}} \leq V_n \end{cases} \tag{6}$$

and the parameters:

- $C_m = 1\,\mu\text{F}:$ cell membrane capacitance
- $g_{leak} = 100\,\mu\text{S}:$ cell membrane leak conductivity
- $V_{rest} = -60\,\text{mV}:$ cell membrane resting voltage
- $V_{thr} = -20\,\text{mV}:$ cell membrane spiking threshold voltage
- $V_{spike} = 20\,\text{mV}:$ spike voltage
- $I_{input} = A_i|\sin(2\pi * 50\,\text{Hz} * t_n)|:$ Input current for $A_1 = 10\,\mu\text{A}$ and $A_2 = 30\,\mu\text{A}$

V_n (V at time t_n) is simulated for 50 ms in time steps of $\Delta t = t_{n+1} - t_n = 25\,\mu\text{s}$, starting with $V_0 = V_{rest}$ at $t_0 = 0\,s$.

Figure 5 shows the input currents for A_1 and A_2. Both are rectified sine waves with a frequency of 50 Hz but different amplitudes.

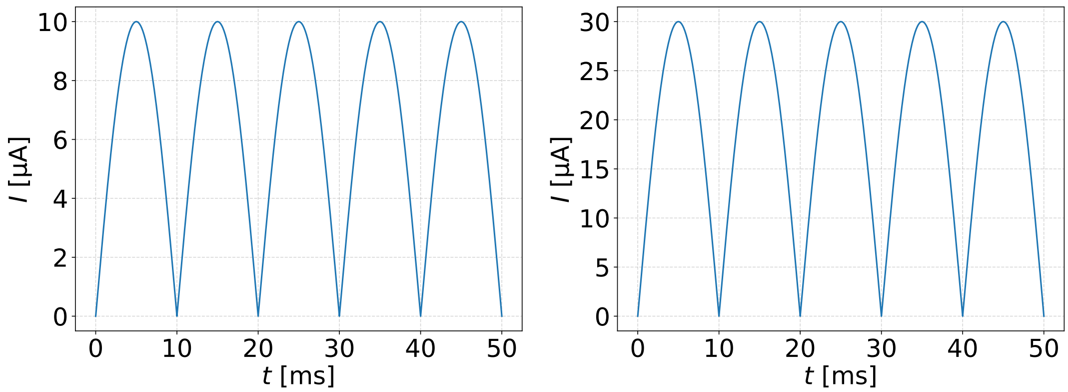

Figure 5: Rectified sine waves with a frequency of 50 Hz and amplitude $A_1 = 10\,\mu\text{A}$ (left) and respectively $A_2 = 30\,\mu\text{A}$ (right).

In Figure 6 the responses of the model to the different input amplitudes are shown. The most obvious difference between the responses is that the higher input current results in the voltage rising to the peak voltage quicker which is reached multiple times per input period instead of just once like with the lower input current.

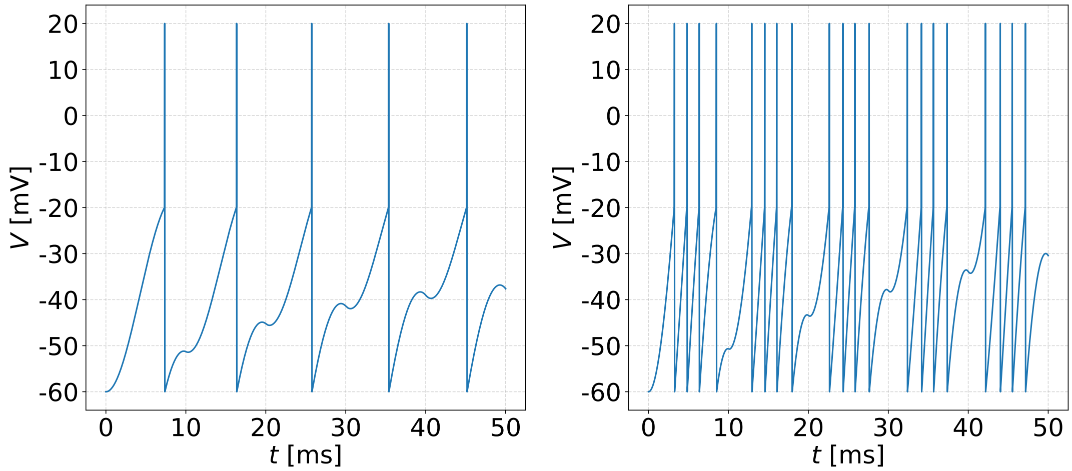

Figure 6: Responses of the LIF model to the different input amplitudes shown in 5. The left-handed plot shows the response to the input amplitude $A_1 = 10\,\mu\mathrm{A}$ and the right-handed plot to $A_2 = 30\,\mu\mathrm{A}$.

3.1 Interpretation of the LIF model's response to different input amplitudes

The LIF model aims to simulate the response of a neuronal cell to input currents from other neurons through synapses. The neuron adds up (integrates) all incoming currents over time, so the threshold is only reached when the signals of multiple preceding neurons reach the target neuron. This property constitutes the **integrating** part of the model.

Once the threshold is overcome, the model would principally remain depolarized with a membrane potential $V = V_{spike}$ without any further input currents. But here comes the **leaky** property to play. The membrane has a relatively high leak conductivity of $100\,\mu\mathrm{S} = 100\,\mu\frac{\mathrm{A}}{\mathrm{V}}$ which means that even when there is still an input current, the neuron can't hold the positive potential V and will quickly return to its resting voltage V_{rest}.

In the plots of Figure 6, you can see that right before $t = 10\,\mathrm{ms}$, where there is still a little input current, the model starts to integrate the current, resulting in a rise in output voltage. At exactly $t = 10\,\mathrm{ms}$, there is no input current, and you can see a little drop in voltage due to the leak current. Because the input period and the time necessary to reach V_{thr} are not perfectly aligned, the model starts a little earlier to integrate with each input period. This is seen in the higher voltage at times where $I_{input} = 0\,A$.

Also, the little Voltage drop due to the leak current becomes more prominent since the leak current depends on the voltage difference between V_n and V_{rest}.

The **firing** property refers to the model's output jumping to a specific voltage V_{spike} once a certain threshold voltage V_{thr} is overcome.

The model's response to fire more frequently rather than more strongly (higher V_{spike}) results from the three properties integration, leaky and firing. When a higher input current is applied, the threshold voltage is reached quickly due to integrating the input current. Since the leaky part will still bring down the voltage to V_{rest} directly, the neuron can integrate the input current again immediately to achieve multiple spikes per input period. For a lower input current, the integration to get to the threshold voltage takes too long to have multiple spikes per input period for the given input frequency.

This property is also seen in muscle stimulation in the human body. To contract a muscle more strongly, the corresponding nerve will not fire at a higher voltage but with a higher frequency.