HUMBOLDT-UNIVERSITÄT ZU BERLIN

BERNSTEIN CENTRE FOR COMPUTATIONAL NEUROSCIENCE

Bernstein Center for Computational Neuroscience
Humboldt-Universität zu Berlin
Philippstr. 13 House 6

Phone: 030/2093-9110
Fax: 030/2093-6771
webpage: http://www.bccn-berlin.de/

Models of Neural Systems I, WS 2008/09
Computer Practical 7

Ordinary Differential Equations (ODEs)

Differential equations describe the evolution of systems in continous time and are widely used in science and engineering. Here, we will focus on the equations of the following form:

$$\dot{x}(t) = f\left(x(t), t\right), \tag{1}$$

where $\dot{x}(t) \equiv \frac{dx}{dt}$ is the first derivative of $x(t)$ with respect to time t. The solution to this equation is a function (dynamical variable) which satisfies the above relation. This equation involves only first derivative with respect to a single independent variable and therefore it is called first-order ordinary differential equation. Here, you will revise how to find analytical solutions to such equations and learn solving them numerically. Finally, you will apply the methods to model the potential across a lipid membrane.

Exercises

1. **Analytical solution to ODEs**

   Solve the following differential equations with the initial condition $x(0) = x_0$ (without using the computer).

   (a) $\frac{dx}{dt} = -x$; $x_0 = 1$

   (b) $\frac{dx}{dt} = x^{-1}$; $x_0 = 1$

   (c) $\frac{dx}{dt} = 1 - x$; $x_0 = 0$

   (d) (optional) $\frac{dx}{dt} = x(1-x)$; $x_0 = \frac{1}{2}$

   Plot the results.

2. **Numerical solutions to ODEs**

   The simplest numerical method of solving the equation (1) is by discretization. Lets assume that the function $x(t)$ is constant over short time interval Δt. We can then rewrite the equation using finite steps:

$$x_{i+1} = x_i + f(x_i, t_i)\Delta t \tag{2}$$

In order to find the solution, we start with the value x_0 given by initial condition and then proceed recursively by adding small increments to the function according to the equation (2). This algorithm is called Euler method.

(a) Solve one of the problems from Exercise 1 with Euler method. Compare the analytical and numerical solutions.

(b) (Optional) Define a Python function which takes as an argument the function $f(x, t)$, initial condition, stop time and the intergration step Δt:

```
def logistic(x,t):
    return x*(x-1);
def euler(f_func,x_0, t_max,dt):
    ...

#Solve logistic ODE
x_t=euler(logistic, 0.5, 5., 0.01)
```

(c) (Optional) Compare the results obtained with Euler method and fourth-order Runge-Kutta method implemented in `scipy.integrate` package (see help for details). The standard usage case is given by the following example:

```
from scipy import integrate, arange
def f(x, t): return -x
x_0 = 0; t = arange(0, 5, 0.01)
x = integrate.odeint(f, x_0, t)
```

3. **Passive membrane**

The following equation describes a passive membrane, which is subject to a current injection:

$$\tau_m \frac{dV(t)}{dt} = -V(t) + E_m + R_m I(t) \tag{3}$$

At time $t = 0$ membrane voltage is at rest $V(t = 0) = E_m$.

(a) Build a model neuron from equation (**??**). Use $R_m = 10^7\,\Omega$, $I(t) = I_0 = 10^{-5}\,\text{mA}$, $\tau_m = 10\,\text{ms}$, $E = -80\,\text{mV}$. Using Euler method find $V(t)$ as a function of time.

(b) Consider the following time-dependent current injection:

$$I(t) = \begin{cases} 10^{-5}\,\text{mA} & \text{if } 100 \geq t \geq 10, \\ 0 & \text{otherwise.} \end{cases}$$

Find a numerical solution for $V(t)$. Repeat the calculations for several values of the membrane time constant τ_m.

Contact

Robert Schmidt (ITB, r. 2316) Phone: 2093-8926 Email: r.schmidt@biologie.hu-berlin.de

Bartosz Telenczuk (ITB, r. 1309) Phone: 2093-8838 Email: b.telenczuk@biologie.hu-berlin.de