Part 1: Neurodynamics

Outline:
• Introduction
• Review of Dynamical Systems
• Linear Systems (Example: negative feedback in retina)
• Numerical solution methods
• Nonlinear oscillations
• Competitive dynamics
• Associative memory
Last time: levels of abstraction

\[
\begin{align*}
\tau_r \frac{dv(t)}{dt} &= -v(t) + F(I_s(t)) \\
\tau_s \frac{dI_s(t)}{dt} &= -I_s(t) + w \cdot u
\end{align*}
\]

\[
\begin{align*}
\tau_r &\ll \tau_s \\
\tau_r &\gg \tau_s
\end{align*}
\]

\[
\begin{align*}
v(t) &= F(I_s(t)), \text{ with} \\
\tau_s \frac{dI_s(t)}{dt} &= -I_s(t) + w \cdot u \\
\tau_r \frac{dv(t)}{dt} &= -v(t) + F(w \cdot u)
\end{align*}
\]
Networks of such units:
• main distinction: feedforward vs. recurrent

Feedforward networks:
• can in principle do practically everything you may want to do, i.e. approximate all “nice” functions
• E.g., coordinate transformations, object recognition(?), etc.
**But:** it may not be economical to do everything with feedforward networks
Review of Dynamical Systems

Natural processes unfold over time:
• swinging of a pendulum
• decay of radioactive material
• a chemical reaction
• growth of a plant
• formation of a Tornado
• galloping of a horse
• reaching for a cup of tea
• action potential traveling down an axon
• remembering an event

A universal mathematical language for describing processes unfolding in time is dynamical systems theory.

1. Continuous time systems: differential equations
2. Discrete time systems: difference equations, iterated maps
Differential Equations

**General Problem:** can be very hard or even impossible to solve analytically

**Three Approaches:**
- *Numerical simulation:* generate approximate solutions with computer
- *Local stability analysis:* study fixed points and surrounding areas
- *Global stability analysis:* look for an “energy” function governing global behavior

Consider these ordinary first order DEs:

\[ \dot{x}(t) = f(x(t), t) \quad \text{non-autonomous} \]
\[ \dot{x}(t) = f(x(t)) \quad \text{autonomous} \]

**Note:** non-autonomous systems can always be turned into autonomous systems by adding a new variable.

**Example:**

\[ \dot{x}(t) = \frac{1}{1+t} x(t) \]

is equivalent to the system

\[ \dot{x}(t) = y(t)x(t) \]
\[ \dot{y}(t) = -y(t)^2 \]
Graphical Interpretations of a DE

\[ \dot{x}(t) = -\alpha x(t) \quad x(t) = A_0 \exp(-\alpha t) \]

**Idea 1:** read this as a prescription of how to choose \( \frac{dx(t)}{dt} \) (the slope of \( x(t) \)) as a function of \( x(t) \):

Solving the DE means to find smooth curves that have these line segments as tangents, there’s one curve through any point.
\[ \dot{x}(t) = -\alpha x(t) \quad x(t) = A_0 \exp(-\alpha t) \]

**Idea 2:** read this as a prescription of how \( x(t) \) “flows” as a function of \( x(t) \)

This picture is the 1-D flow field of the DE.
If \( x \) is positive (right side) \( x \) will be shrinking (left arrow).
If \( x \) is negative (left side) \( x \) will be growing (right arrow).
At \( x=0 \), \( \frac{dx}{dt}=0 \), i.e. \( x \) does not change at all: steady state, fixed point.

**Note:** this qualitative analysis is the 1-D version of a “phase space analysis”.
It allows to read off the long term behavior easily. It can always be done, even for DEs where solution cannot easily be found analytically.
Linear Systems

Motivations:
• linear systems are easy to work out analytically: the solution essentially reduces to solving a linear algebra problem
• few systems are linear, but what we will use what we learn here for the stability analysis of non-linear systems

Consider the following general linear system:

\[ \dot{x}(t) = Ax(t) + b \]

\[ x: N\text{-dim. vector, } A: N\times N \text{ matrix, } b: N\text{-dim. vector} \]

Note: all essential aspects of linear systems can be studied in 2 dim. systems.
Stationary state, homogeneous and inhomogeneous system

Inhomogeneous system: \( \dot{x}(t) = Ax(t) + b \)

Homogeneous system: \( \dot{x}(t) = Ax(t) \)

Stationary state of inhomogeneous system:

\[
0 = \dot{x}_0 = Ax_0 + b \iff x_0 = -A^{-1}b \quad \text{(if } A \text{ invertible)}
\]

Introduce deviation from stationary state, \( y(t) \):

\[
y(t) = x(t) - x_0 \quad \Rightarrow \quad \dot{y}(t) = \dot{x}(t) \quad \mid \text{plug into DE}
\]

\[
\dot{y}(t) = A(y(t) + x_0(t)) + b = Ay(t) + Ax_0(t) + b = Ay(t)
\]

If we find a solution to the homogeneous DE for \( y(t) \), the solution for \( x(t) \) is obtained by simply adding \( x_0 \).
Solution of homogeneous system

Homogeneous system: \( \dot{x}(t) = Ax(t) \)

Ansatz: \( x(t) = a \exp(\lambda t) \Rightarrow \dot{x}(t) = a \lambda \exp(\lambda t) = \lambda x(t) \)

Put back in DE: \( Ax(t) = \lambda x(t) \)  
Eigenvalue problem

Besides the trivial solution \( x(t) = 0 \), a solution requires this determinant to be zero:

\[ |A - \lambda I| = 0 \]

This so-called *characteristic equation* determines the Eigenvalues or roots of the system. The Eigenvalues determine the character of the solution:

- all real parts negative: asymptotic stability
- highest real part equal to zero: marginal stability
- at least one real part positive: unstable
- non-zero imaginary parts: oscillatory components
Box 5.1  Summary of the Solution of $\dot{x} = Ax$

Solve for the eigenvalues of

$$|\lambda I - A| = 0$$

Taking the determinant results in an $n$th-order characteristic equation in $\lambda$:

$$\lambda^n + a_1\lambda^{n-1} + \cdots + a_{n-2}\lambda + a_{n-1} = 0$$

Solve this for $\lambda_1, \ldots, \lambda_n$. For each $\lambda_i$ determine the eigenvector $v_i$ from

$$[A - \lambda_i I]v_i = 0$$

At the end of the solution process, the answer is given by

$$x(t) = \sum_{i=1}^{n} a_i v_i e^{\lambda_i t}$$

where $\lambda_i$ and $v_i$ are the eigenvalues and eigenvectors of $A$ and the $a_i$ are determined by the initial conditions.

Notes:

1. Since the determinant $|A - \lambda_i I|$ is zero, the eigenvectors can be determined only up to a scale factor.

2. The solution method outlined here only works for distinct eigenvalues, that is, $\lambda_i \neq \lambda_j \forall i, j$. For multiple repeated roots, the solution method is more complex and results in expressions of the form $t^{n_i}e^{\lambda_i t}$. 
Some phase portraits of 2-D systems

- Spiral Point
- Node
- Saddle Point
- Center
Physical analogs for different kinds of stability

Consider object sliding across a hilly landscape. Stationary point: \( \frac{dx}{dt}=0 \)
What happens if system in stationary state gets a small “nudge”?

- System returns to stationary point: point is stable (asymptotically stable)
- System rests at neighboring point: marginally stable
- System runs away from stationary point: point is unstable

**saddle point:** direction of small nudge matters
Negative Feedback in the Retina

A

rod and cone receptors (R)
horizontal (H)
bipolar (B)
amacrine (A)
retinal ganglion (G)

B

+ +
R

- +
H

- +
B

- +
A

- +
G_1

to optic nerve

G_2

light
Model negative feedback to cone from horizontal cells by the following linear system:

\[ \dot{c} = \frac{1}{\tau_c} (-c - kh + l) \]
\[ \dot{h} = \frac{1}{\tau_h} (-h + c) \]

Rewrite in matrix form to better see linear structure:

\[
\begin{pmatrix}
\dot{c} \\
\dot{h}
\end{pmatrix}
= \begin{pmatrix}
-\frac{1}{\tau_c} & -\frac{k}{\tau_c} \\
\frac{1}{\tau_h} & -\frac{1}{\tau_h}
\end{pmatrix}
\begin{pmatrix}
c \\
h
\end{pmatrix}
+ \begin{pmatrix}
l \\
0
\end{pmatrix}
\]

is of the form \( \dot{\mathbf{x}}(t) = A\mathbf{x}(t) + \mathbf{b} \)

Stationary state:

\[ \dot{c}_0 = \dot{h}_0 = 0 \Rightarrow h_0 = c_0 = \frac{l}{k + 1} \]
Specific case: $\tau_c = 0.025s$, $\tau_h = 0.08s$, $k = 4$

For $L=10$ the stationary point is $c_0 = h_0 = 2$, leading to the homogeneous equation:

$$\begin{bmatrix}
\dot{c}' \\
\dot{h}'
\end{bmatrix} =
\begin{bmatrix}
-40 & -160 \\
12.5 & -12.5
\end{bmatrix}
\begin{bmatrix}
c' \\
h'
\end{bmatrix}$$

The Eigenvalues of this matrix are:

$$\lambda = -26.25 \pm 42.56i$$

Both Eigenvalues have negative real parts and non-zero imaginary parts, from which we see that the solution has the character of a spiral point (damped oscillation), in particular, the stationary point is stable.
Numerical Solution Methods

Motivation:
In only few instances can we easily obtain analytical solutions to our problems. But we can always simulate.

But:
This does not mean that we should run simulations only. Whenever we can get analytical answers, they typically provide interesting insights.

Note:
We will consider the 1-dim. case, but all methods we discuss (Euler method, improved Euler method, fourth order Runge Kutta) generalize to $N$ dimensions easily.
Consider the ordinary first order DE: \( \dot{x}(t) = f(x(t)) \)

**Idea:**
approximate derivative with finite ratio: \( \dot{x}(t) \approx \frac{\Delta x}{\Delta t} = \frac{x(t + \Delta t) - x(t)}{\Delta t} \)

\[
\frac{x(t + \Delta t) - x(t)}{\Delta t} \approx f(x(t))
\]

Now solve for \( x(t+\Delta t) \): \( x(t + \Delta t) \approx x(t) + f(x(t))\Delta t \)

Given an initial condition \( x(0) \) we can select a “suitable” \( \Delta t \) and compute \( x_n = x(t+n\Delta t) \) in an iterative manner, \( n=1,2,3,... \)

This is the so-called **Euler method**, a first-order method.
Graphical Interpretation

\[ x(t + \Delta t) \approx x(t) + f(x(t), t)\Delta t \]

**Idea:** extrapolate tangents to \( x(t) \) for finite distance
Improved Euler Method
(2nd order Runge Kutta method)

Euler Method can be viewed as first order Taylor approximation:

\[ x(t + \Delta t) \approx x(t) + f(x(t))\Delta t \]

Other methods can be derived by considering higher order approximations.

\[ x(t + \Delta t) \approx x(t) + f(x(t))\Delta t + \frac{1}{2} \frac{df(x(t))}{dt} \Delta t^2 \]

Improved Euler Method works in two steps:

\[ \tilde{x}_{n+1} = x_n + f(x_n)\Delta t \]  
“trial step”

\[ x_{n+1} = x_n + \frac{1}{2} [f(x_n) + f(\tilde{x}_{n+1})]\Delta t \]  
“real step”

Intuition: average slopes at beginning and end of interval.
Motivation: higher order methods make more accurate steps but are not necessarily better. Important is also the number of computer operations necessary per step. The 4th order Runge Kutta method is good compromise:

\[
\begin{align*}
  k_1 &= f(x_n) \Delta t \\
  k_2 &= f(x_n + \frac{1}{2} k_1) \Delta t \\
  k_3 &= f(x_n + \frac{1}{2} k_2) \Delta t \\
  k_4 &= f(x_n + k_3) \Delta t \\
  x_{n+1} &= x_n + \frac{1}{6} \left( k_1 + 2k_2 + 2k_3 + k_4 \right)
\end{align*}
\]

Notes: The right choice of \( \Delta t \) is problematic: \( \Delta t \) too small: simulations take long, \( \Delta t \) too big: simulations can be grossly incorrect
- good heuristic is to halve \( \Delta t \) and see if result remains the same
- more advanced: variable step size methods
Local Stability Analysis

**Step One:** find stationary point(s)

**Step Two:** linearize around all stationary points (using Taylor expansion), the Eigenvalues of the linearized problem determine nature of stationary point:

Real parts:
- positive: growth of fluctuations, instability
- negative: decay of fluctuations, stability

Imaginary parts:
- if present, solutions are oscillatory (spiraling)
  - spiraling inward or outward if non-zero real parts

Overall: point (asymptotically) stable if all real parts negative
Interaction of Excitatory and Inhibitory Neuronal Populations

Motivations:

• understand the emergence of oscillations in excitatory-inhibitory networks
• learn about local stability analysis

Consider 2 populations of excitatory and inhibitory neurons with firing rates \( v \):

Dale’s law: every neuron is either excitatory or inhibitory, never both
Mathematical formulation:

\[ \tau_E \frac{dv_E}{dt} = -v_E + [M_{EE} v_E + M_{EI} v_I - \gamma_E]_+ \]

\[ \tau_I \frac{dv_I}{dt} = -v_I + [M_{II} v_I + M_{IE} v_E - \gamma_I]_+ . \]

Parameters: \( M_{EE} = 1.25, M_{EI} = -1, \gamma_E = -10 \text{Hz}, \tau_E = 10 \text{ms} \)
\( M_{II} = 0, M_{IE} = 1, \gamma_I = 10 \text{ Hz}, \tau_I = \text{varying} \)

Stationary point:

\( v_E^0 = 26.67, \ v_I^0 = 16.67 \)
Linearization around stationary point gives the following matrix $\mathbf{A}$ with these Eigenvalues:

\[
\begin{pmatrix}
\frac{(M_{EE} - 1)}{\tau_E} & \frac{M_{EI}}{\tau_E} \\
\frac{M_{IE}}{\tau_I} & \frac{(M_{II} - 1)}{\tau_I}
\end{pmatrix}.
\]

As discussed in the Mathematical Appendix, the stability of the fixed point is determined by the real parts of the eigenvalues of this matrix. The eigenvalues are given by

\[
\lambda = \frac{1}{2} \left( \frac{M_{EE} - 1}{\tau_E} + \frac{M_{II} - 1}{\tau_I} \pm \sqrt{\left( \frac{M_{EE} - 1}{\tau_E} - \frac{M_{II} - 1}{\tau_I} \right)^2 + \frac{4M_{EI}M_{IE}}{\tau_E \tau_I}} \right).
\]
**Phase Portrait**

**A:** Stationary point is intersection of the nullclines. Arrows indicate direction of flow in different area of the phase space (state space).

**B:** real and imaginary part of Eigenvalue as a function of $\tau_i$. 

Jochen Triesch, UC San Diego, http://cogsci.ucsd.edu/~triesch
For $\tau_I$ below critical value of 40ms, Eigenvalues have negative real parts: we see damped oscillations. Trajectory spirals to stable fixed point.
When $\tau_I$ grows beyond critical value of 40ms, a *Hopf bifurcation* occurs (here $\tau_I = 50\text{ms}$): stable fixed point $\rightarrow$ unstable fixed point $+\text{ limit cycle}$

Here, the amplitude of the oscillation grows until the non-linearity “clips” it.
Neural Oscillations

- interaction of excitatory and inhibitory neuron populations can lead to oscillations

- very important in, e.g. locomotion: rhythmic walking and swimming motions: Central Pattern Generators (CPGs)

- also very important in olfactory system (selective amplification)

- also oscillations in visual system: functional role hotly debated. Proposed as solution to binding problem:
  - Idea: neural populations that represent features of the same object synchronize their firing
Binding Problem

- what and where (how) pathways in visual system
- how do you know what is where?

<table>
<thead>
<tr>
<th>Visual Field</th>
<th>Neural Representation</th>
<th>Spike Trains</th>
</tr>
</thead>
<tbody>
<tr>
<td>circle</td>
<td></td>
<td></td>
</tr>
<tr>
<td>triangle</td>
<td></td>
<td></td>
</tr>
<tr>
<td>up</td>
<td></td>
<td></td>
</tr>
<tr>
<td>down</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Synchronization

- no
- yes