Progress So Far

Weeks 1 - 8

Overview

- CBC
- Gaussian processes
- Numerical continuation software
- Neurons and neural dynamics

GPR

Gaussian Process Regression

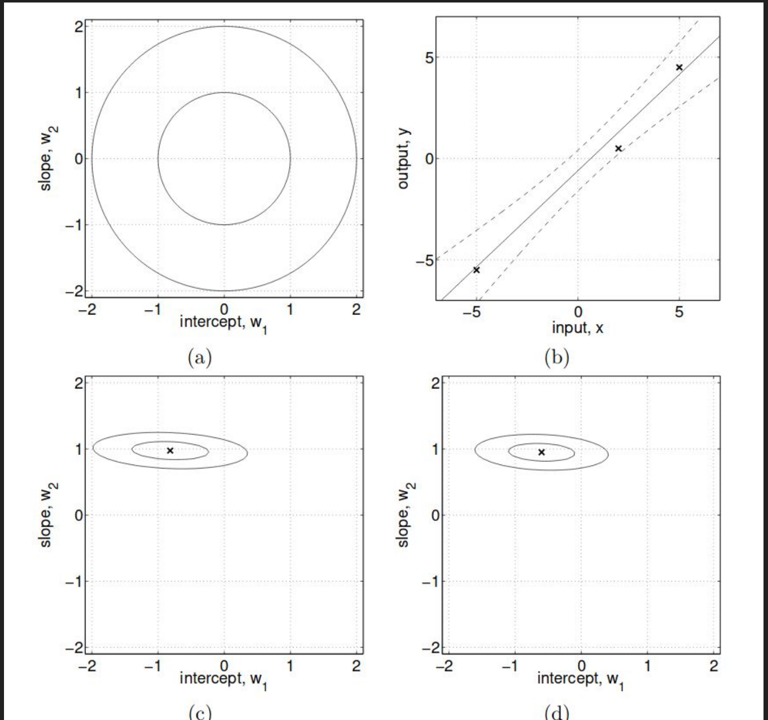
We have some data; let's assume the data came from some underlying vector function.

LSQ model fitting is equivalent to MAP fitting, under some general assumptions

Typical approach: define a model (eg. y=mx+c), and fit it.

Gaussian Process Regression

MAP fitting defines a probability distribution over a set of weights / basis functions



Gaussian Process Regression

MAP fitting defines a probability distribution over a set of weights / basis functions

These basis functions always appear in the posterior distribution as a dot product

$$
\begin{aligned}
f_*|\mathbf{x}_*, X, \mathbf{y} \sim \mathcal{N}\big(&\phi_*^\top \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \mathbf{y}, \\
&\phi_*^\top \Sigma_p \phi_* - \phi_*^\top \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \Phi^\top \Sigma_p \phi_*\big),
\end{aligned}
$$

Gaussian Process Regression

MAP fitting defines a probability distribution over a set of weights / basis functions

These basis functions always appear in the posterior distribution as a dot product

To avoid explicitly computing this dot product, we can replace it with a kernel; suitably chosen kernels can be expanded into a dot product between an infinite number of basis functions

Gaussian Process Regression

GPR lets us implicitly maintain a distribution over an infinite number of functions, rather than explicitly over finitely many

Powerful: very general, non-parametric, statistically optimal (under some general assumptions)

Gaussian Process Regression

Consistency requirement means we're able to sample at a finite number of points (sampling functions would be infinite), by using a Gaussian distribution with mean evaluated at each test point, and covariance evaluated between each test point pair

We can condition on this Gaussian distribution to sample from the posterior Gaussian process

GPR Demo

CBC

CBC - what is it and why do we do it?

- Control-based analog of numerical continuation

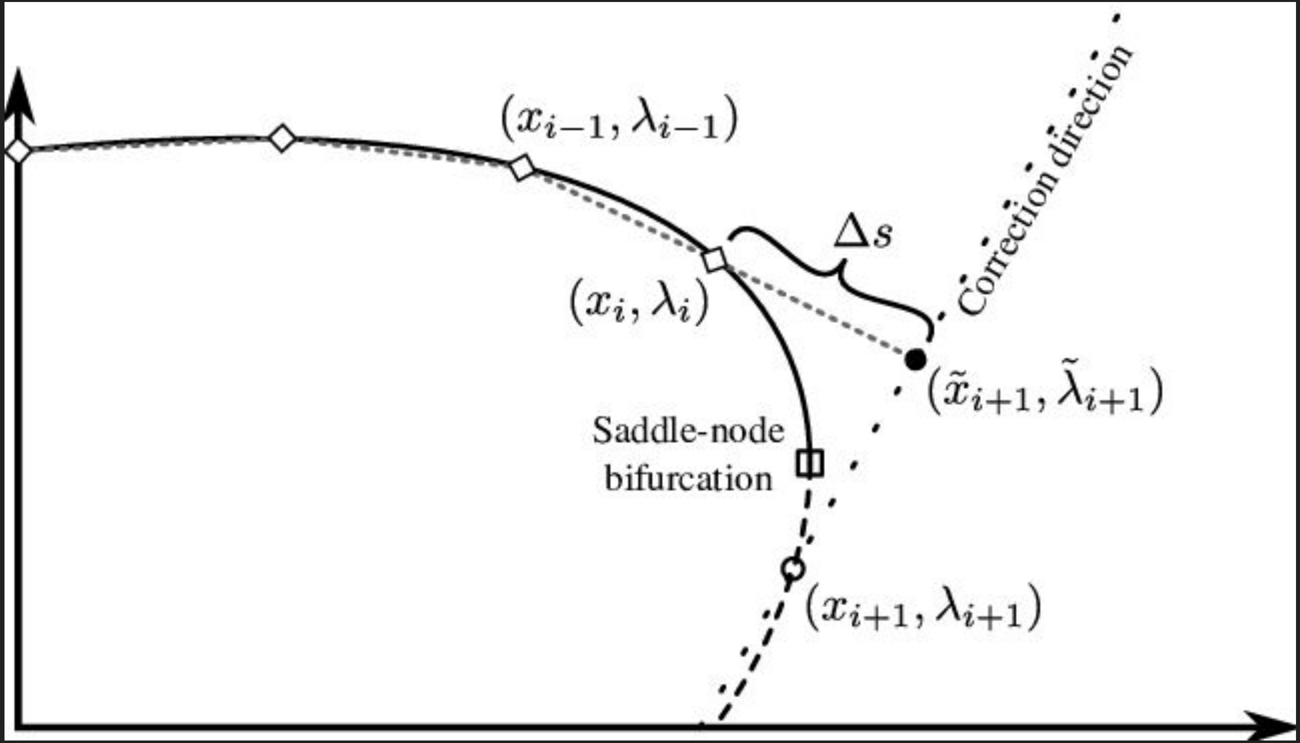
Numerical continuation - what is it and why do we do it?

- Goal - find solutions x to the zero-problem $f(x,\lambda)=0$ as λ varies

Numerical continuation - what is it and why do we do it?

- Goal - find solutions x to the zero-problem $f(x,\lambda)=0$ as λ varies
- Why? Bifurcation analysis!

Numerical continuation - how do we do it?



CBC - what is it and why do we do it?

- Control-based analog of numerical continuation

CBC - what is it and why do we do it?

- Control-based analog of numerical continuation
- Can't set up a zero-problem of form $f(x,\lambda)=0$ without some model $f(x,\lambda)$
- Models don't necessarily represent reality

How can we use control-based approaches to run continuation schemes on physical systems?

How do we implement CBC?

- Consider some controllable system $\dot{x} = f(x,\lambda) + u(x,t)$

How do we implement CBC?

- Consider some controllable system $\dot{x} = f(x,\lambda) + u(x,t)$
- When $u(x,t)=0$ the system is operating open-loop, and thus under its natural dynamics

How do we implement CBC?

- Consider some controllable system $\dot{x} = f(x,\lambda) + u(x,t)$
- When $u(x,t)=0$ the system is operating open-loop, and thus under its natural dynamics
- The construction of $u(x,t)$ requires a control target $x^*(t)$. When $x(t)=x^*(t)$, no control effort needs applying, so $u(x,t)=0$

How do we implement CBC?

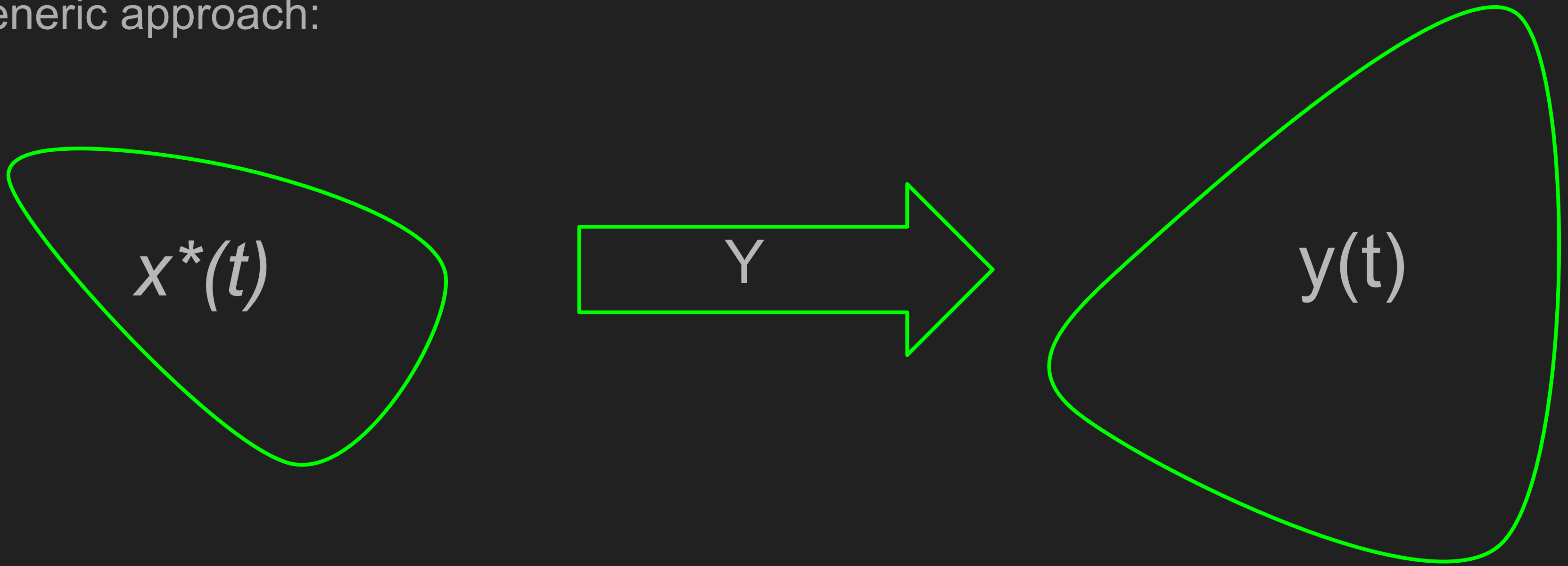
- Consider some controllable system $\dot{x} = f(x,\lambda) + u(x,t)$
- When $u(x,t)=0$ the system is operating open-loop, and thus under its natural dynamics
- The construction of $u(x,t)$ requires a control target $x^*(t)$. When $x(t)=x^*(t)$, no control effort needs applying, so $u(x,t)=0$
- Therefore, the invariant sets of the open-loop system are given by the closed-loop control targets $x^*(t)$ that drive the required control effort $u(x,t)$ to zero

How do we implement CBC?

$u(x,t)=0$ is enough to define a zero-problem, but how?

How do we implement CBC?

Generic approach:



How do we implement CBC?

Claim: a function is a solution of the free system, if and only if it is a fixed point of the input-output map Y [1].

(Question: is this actually true? Surely if the system is fully controllable, the control target will become asymptotically stable and the error will drop to zero, making every controllable target x(t) a fixed point of the I/O map)*

[1] Sieber, Jan, and Bernd Krauskopf. "Control based bifurcation analysis for experiments." *Nonlinear Dynamics* 51.3 (2008): 365-377.

How do we implement CBC?

- Continuation problem becomes one of finding a fixed point of Y

How do we implement CBC?

- Continuation problem becomes one of finding a fixed point of Y
- We can use a Galerkin projection to discretise the input and output functions

How do we implement CBC?

- Continuation problem becomes one of finding a fixed point of Y
- We can use a Galerkin projection to discretise the input and output functions
- The resulting map can then be solved using a conventional Newton or quasi-Newton solver

How do we implement CBC?

- Continuation problem becomes one of finding a fixed point of Y
- We can use a Galerkin projection to discretise the input and output functions
- The resulting map can then be solved using a conventional Newton or quasi-Newton solver
- (preferably Newton-Krylov or quasi-Newton!)

How do we implement CBC?

Is there not an easier way?

(Hint: yes)

Is there not an easier way?

Allow the forcing input to come from the controller:

1. Set the system *response* amplitude (control target fundamental harmonic)
2. Keep fundamental harmonics fixed; set higher harmonics of the control target to higher harmonics of the system output
3. Higher harmonics of control force get driven to zero

The system will now converge to an invariant set of the forced open-loop system

- System’s forcing input = control action amplitude (measured)
- System output amplitude = control target amplitude (set)

Background maths for CBC

- Galerkin projections
- Floquet theory
- Newton methods
- MIMO XAR models
- Control theory
- Gaussian processes...

Demo: CBC Duffing oscillator simulation

Numerical Continuation

(see pdf)

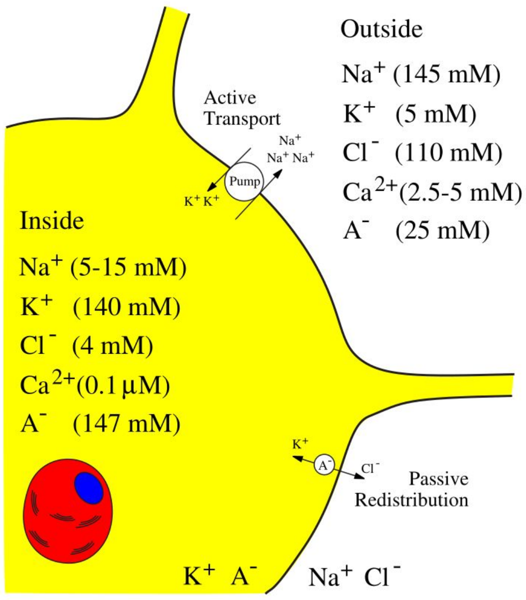
Neurons and Neural Dynamics

Neurons and neural computation

- Neural systems are capable of computation
- Computation arises from neurons deciding when to fire
- Basic description: thresholds and all-or-nothing spikes
- Biological description: gated ionic current flows
- Mathematical description: nonlinear dynamical system

Neural physiology

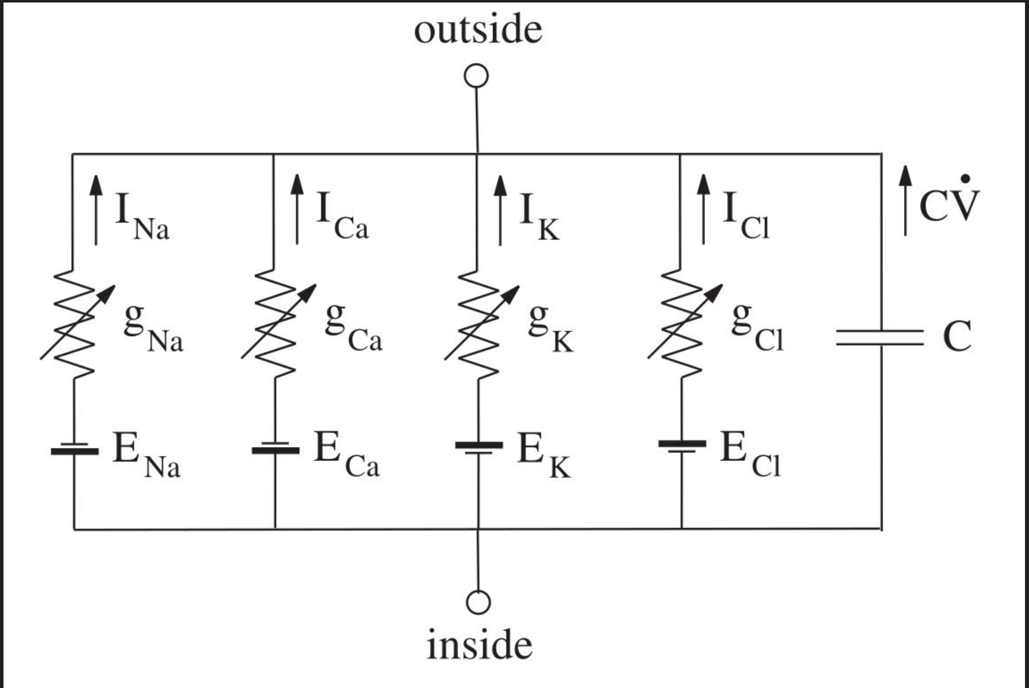
- Electrochemical equilibrium
- Resting membrane potential
- Ion channels
- Ionic currents change membrane potential



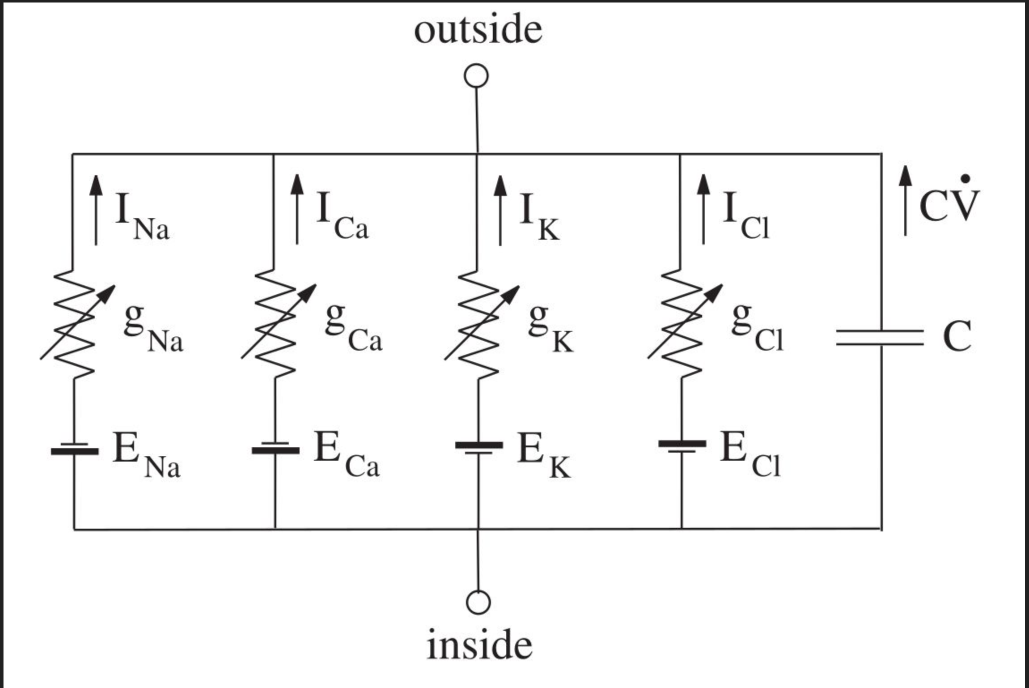
Equilibrium Potentials

$$
\begin{array}{lrcl}
Na^{+} & 62\log\frac{145}{5} & = & 90 \text{ mV} \\
 & 62\log\frac{145}{15} & = & 61 \text{ mV} \\
K^{+} & 62\log\frac{5}{140} & = & -90 \text{ mV} \\
Cl^{-} & -62\log\frac{110}{4} & = & -89 \text{ mV} \\
Ca^{2+} & 31\log\frac{2.5}{10^{-4}} & = & 136 \text{ mV} \\
 & 31\log\frac{5}{10^{-4}} & = & 146 \text{ mV}
\end{array}
$$

Neural physiology



Neural physiology



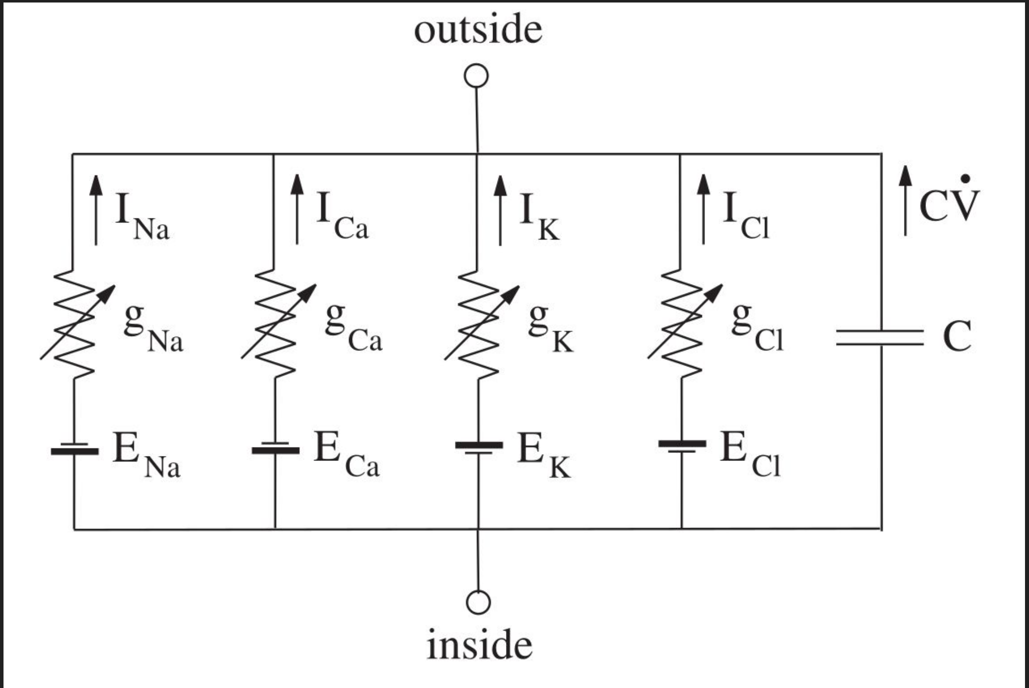
I = V/R

g = 1/R

I = gV

$$I_{\mathrm{Na}} = g_{\mathrm{Na}}\left(V - E_{\mathrm{Na}}\right), \qquad I_{\mathrm{Ca}} = g_{\mathrm{Ca}}\left(V - E_{\mathrm{Ca}}\right), \qquad I_{\mathrm{Cl}} = g_{\mathrm{Cl}}\left(V - E_{\mathrm{Cl}}\right)$$

Neural physiology



$$I = C\dot{V} + I_{\mathrm{Na}} + I_{\mathrm{Ca}} + I_{\mathrm{K}} + I_{\mathrm{Cl}}$$

Neural physiology

$$I = C\dot{V} + I_{\mathrm{Na}} + I_{\mathrm{Ca}} + I_{\mathrm{K}} + I_{\mathrm{Cl}}$$

Neural physiology

$$I = C\dot{V} + I_{\mathrm{Na}} + I_{\mathrm{Ca}} + I_{\mathrm{K}} + I_{\mathrm{Cl}}$$

$$I_{\mathrm{Na}} = g_{\mathrm{Na}}\left(V - E_{\mathrm{Na}}\right) , \qquad I_{\mathrm{Ca}} = g_{\mathrm{Ca}}\left(V - E_{\mathrm{Ca}}\right) , \qquad I_{\mathrm{Cl}} = g_{\mathrm{Cl}}\left(V - E_{\mathrm{Cl}}\right)$$

Neural physiology

$$I = C\dot{V} + I_{\mathrm{Na}} + I_{\mathrm{Ca}} + I_{\mathrm{K}} + I_{\mathrm{Cl}}$$

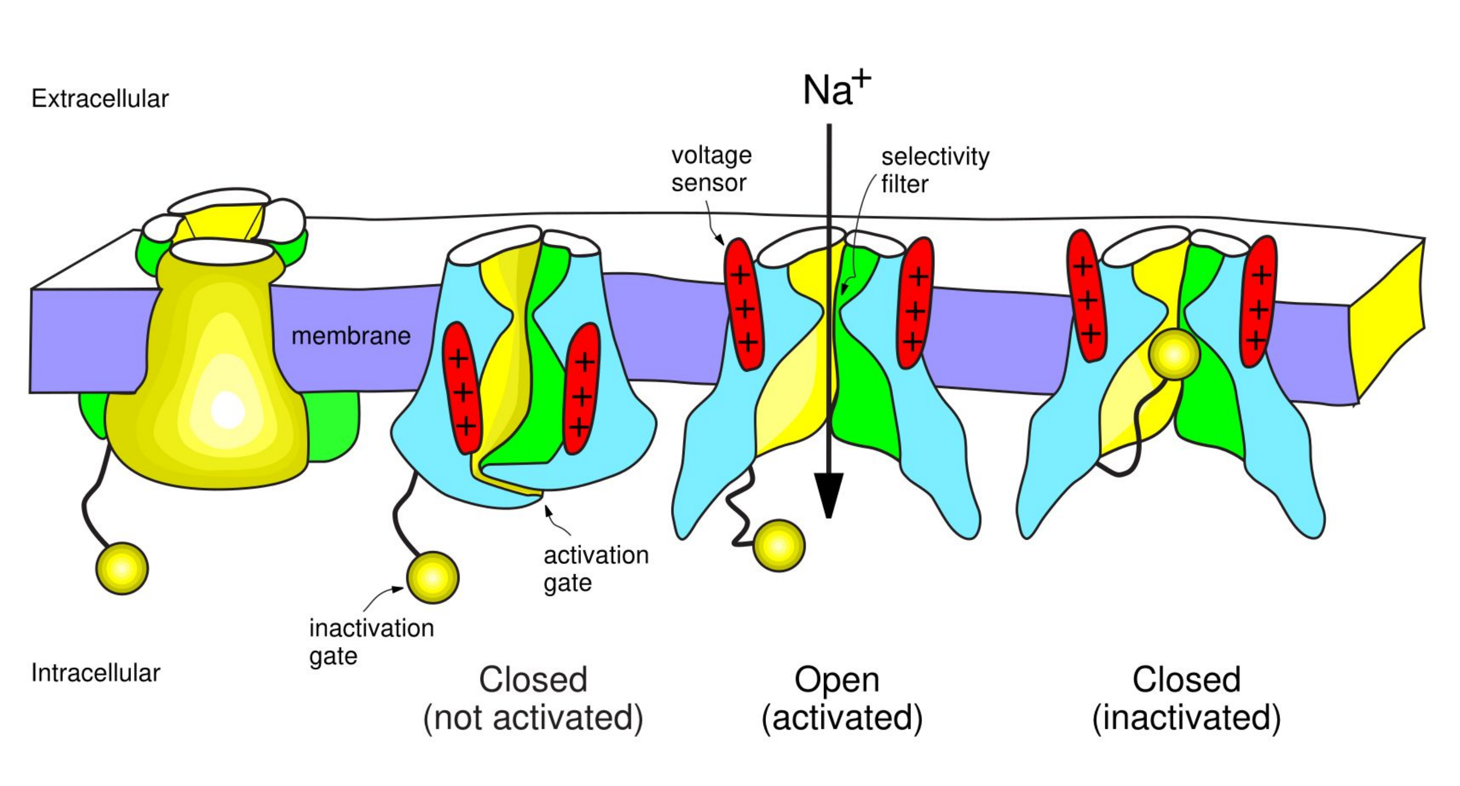
$$I_{\mathrm{Na}} = g_{\mathrm{Na}}\left(V - E_{\mathrm{Na}}\right)\,, \qquad I_{\mathrm{Ca}} = g_{\mathrm{Ca}}\left(V - E_{\mathrm{Ca}}\right)\,, \qquad I_{\mathrm{Cl}} = g_{\mathrm{Cl}}\left(V - E_{\mathrm{Cl}}\right)$$

$$C\dot{V} = I - g_{\mathrm{Na}}\left(V - E_{\mathrm{Na}}\right) - g_{\mathrm{Ca}}\left(V - E_{\mathrm{Ca}}\right) - g_{\mathrm{K}}\left(V - E_{\mathrm{K}}\right) - g_{\mathrm{Cl}}\left(V - E_{\mathrm{Cl}}\right)$$

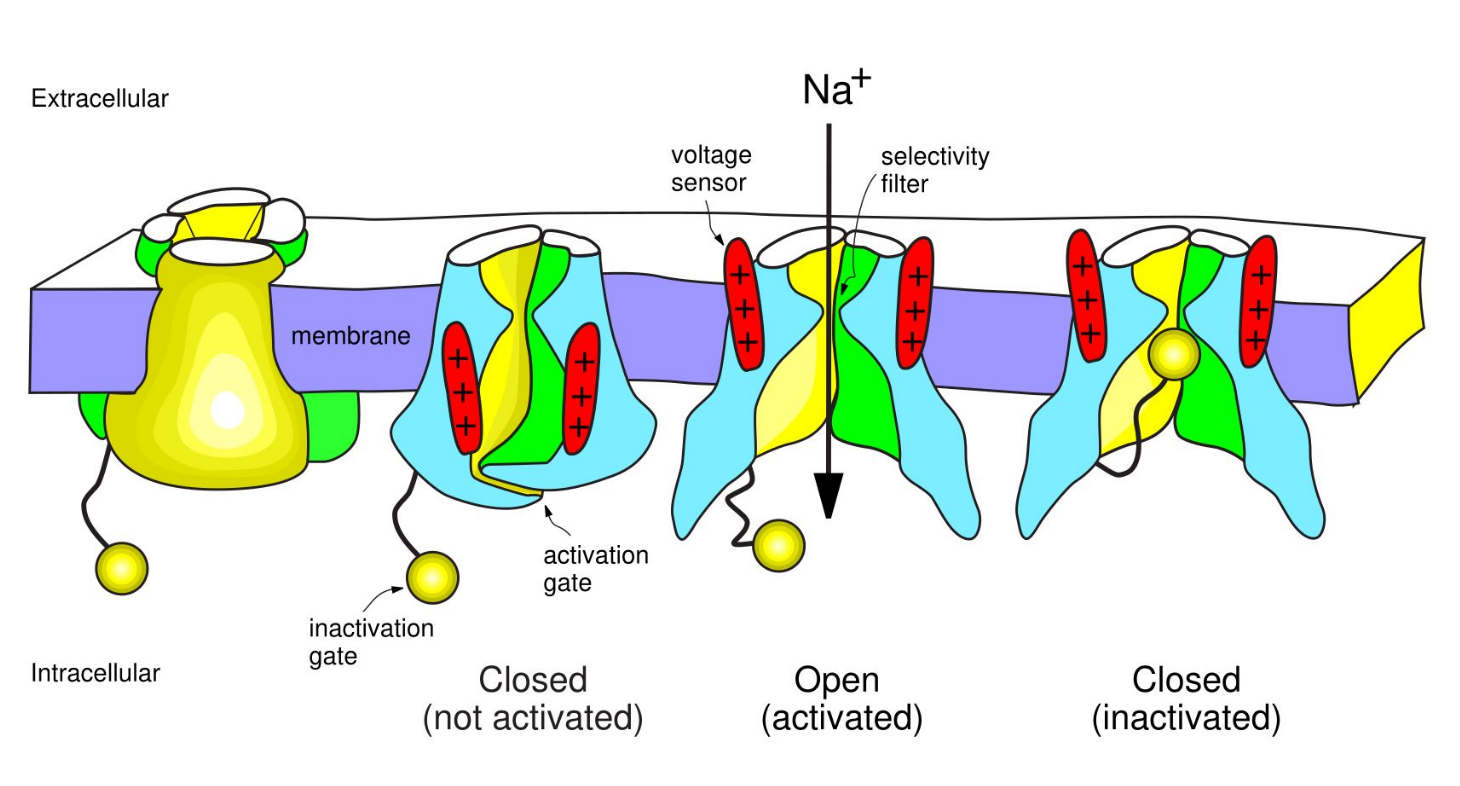
Neural physiology

So what about those conductances?

Neural physiology



Neural physiology



$$p = m^a \, h^b$$

$$I = \bar{g} \, p \, (V - E)$$

Neural Dynamics

$$C\dot{V} = I - \overbrace{\bar{g}_{\mathrm{K}} n^4 (V - E_{\mathrm{K}})}^{I_{\mathrm{K}}} - \overbrace{\bar{g}_{\mathrm{Na}} m^3 h (V - E_{\mathrm{Na}})}^{I_{\mathrm{Na}}} - \overbrace{g_{\mathrm{L}} (V - E_{\mathrm{L}})}^{I_{\mathrm{L}}}$$

Neural Dynamics

$$\dot{m} = (m_\infty(V) - m)/\tau(V)$$

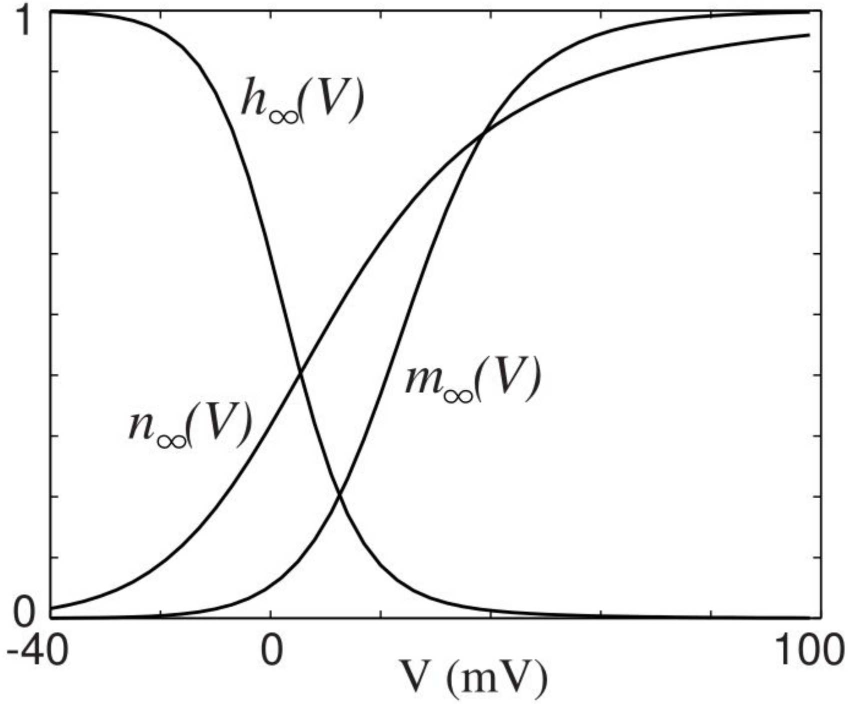
$$\dot{h} = (h_\infty(V) - h)/\tau(V)$$

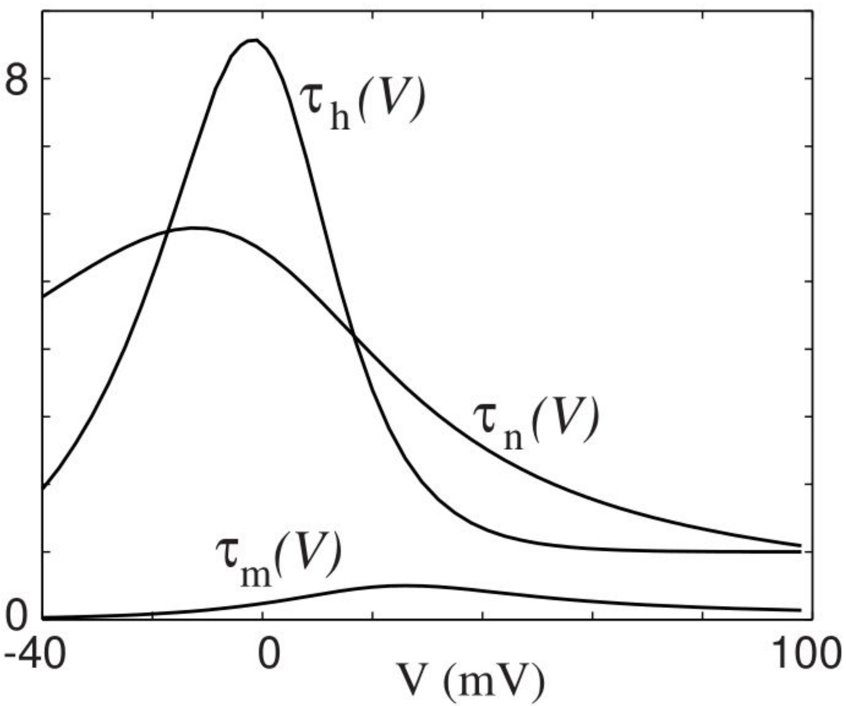
Neural Dynamics

$$
\begin{aligned}
C\dot{V} &= I - \overbrace{\bar{g}_{\mathrm{K}} n^4 (V - E_{\mathrm{K}})}^{I_{\mathrm{K}}} - \overbrace{\bar{g}_{\mathrm{Na}} m^3 h (V - E_{\mathrm{Na}})}^{I_{\mathrm{Na}}} - \overbrace{g_{\mathrm{L}} (V - E_{\mathrm{L}})}^{I_{\mathrm{L}}} \\
\dot{n} &= \alpha_n(V)(1-n) - \beta_n(V)n \\
\dot{m} &= \alpha_m(V)(1-m) - \beta_m(V)m \\
\dot{h} &= \alpha_h(V)(1-h) - \beta_h(V)h\ ,
\end{aligned}
$$

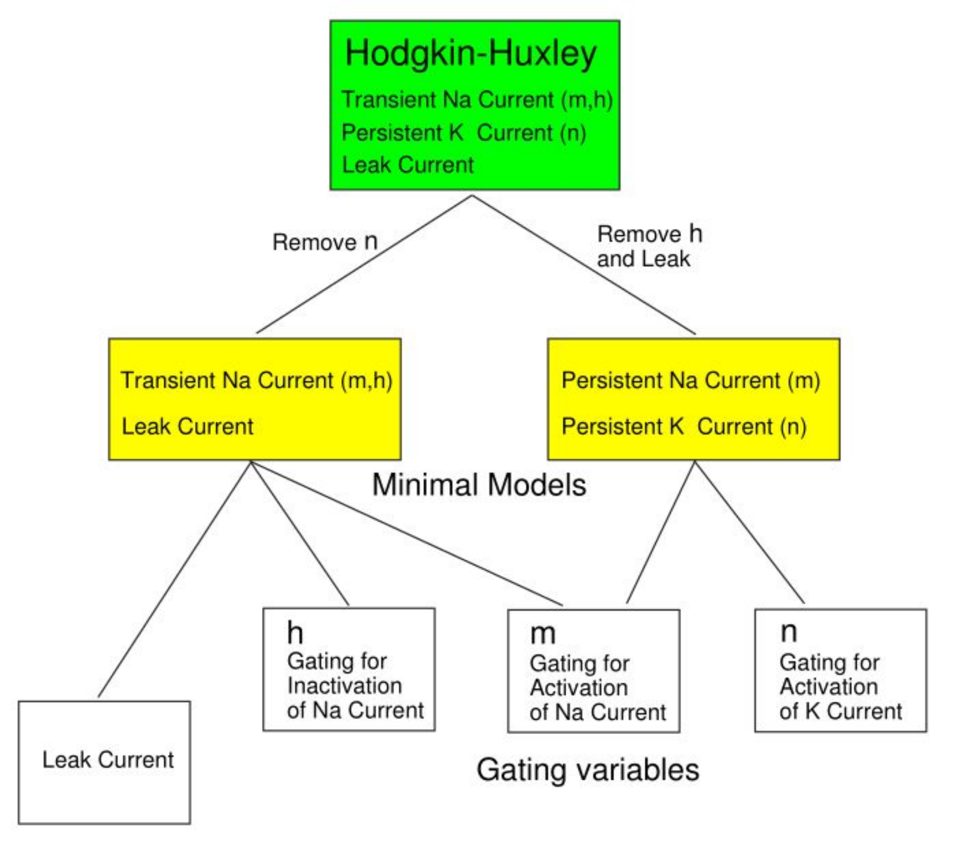
Neural Dynamics

$$C\dot{V} = I - \overbrace{\bar{g}_{\mathrm{K}} n^4 (V - E_{\mathrm{K}})}^{I_{\mathrm{K}}} - \overbrace{\bar{g}_{\mathrm{Na}} m^3 h (V - E_{\mathrm{Na}})}^{I_{\mathrm{Na}}} - \overbrace{g_{\mathrm{L}} (V - E_{\mathrm{L}})}^{I_{\mathrm{L}}}$$

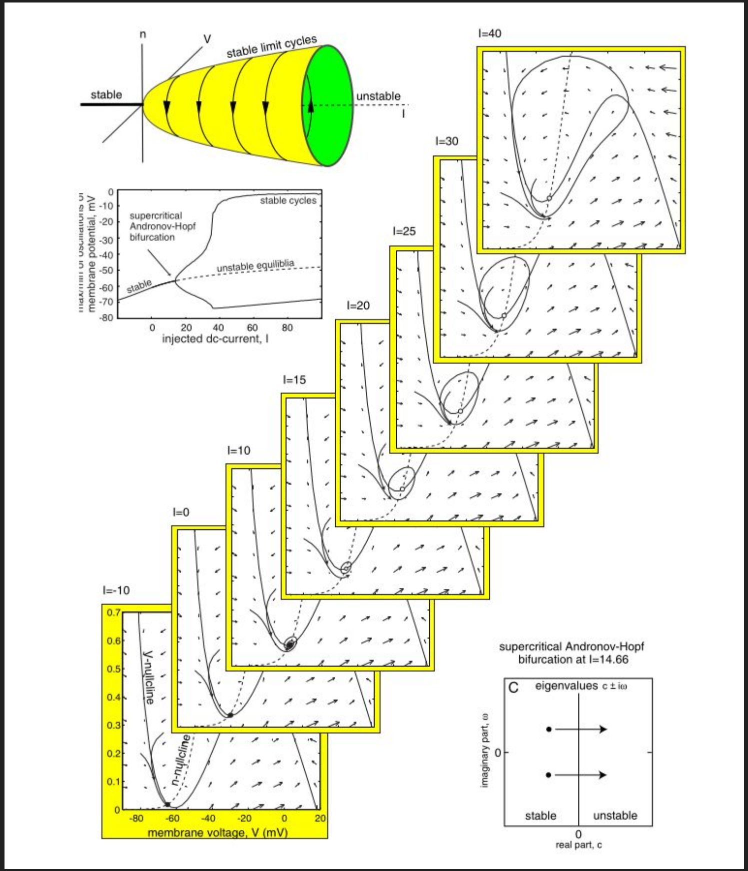




Neural Dynamics



Neural Dynamics



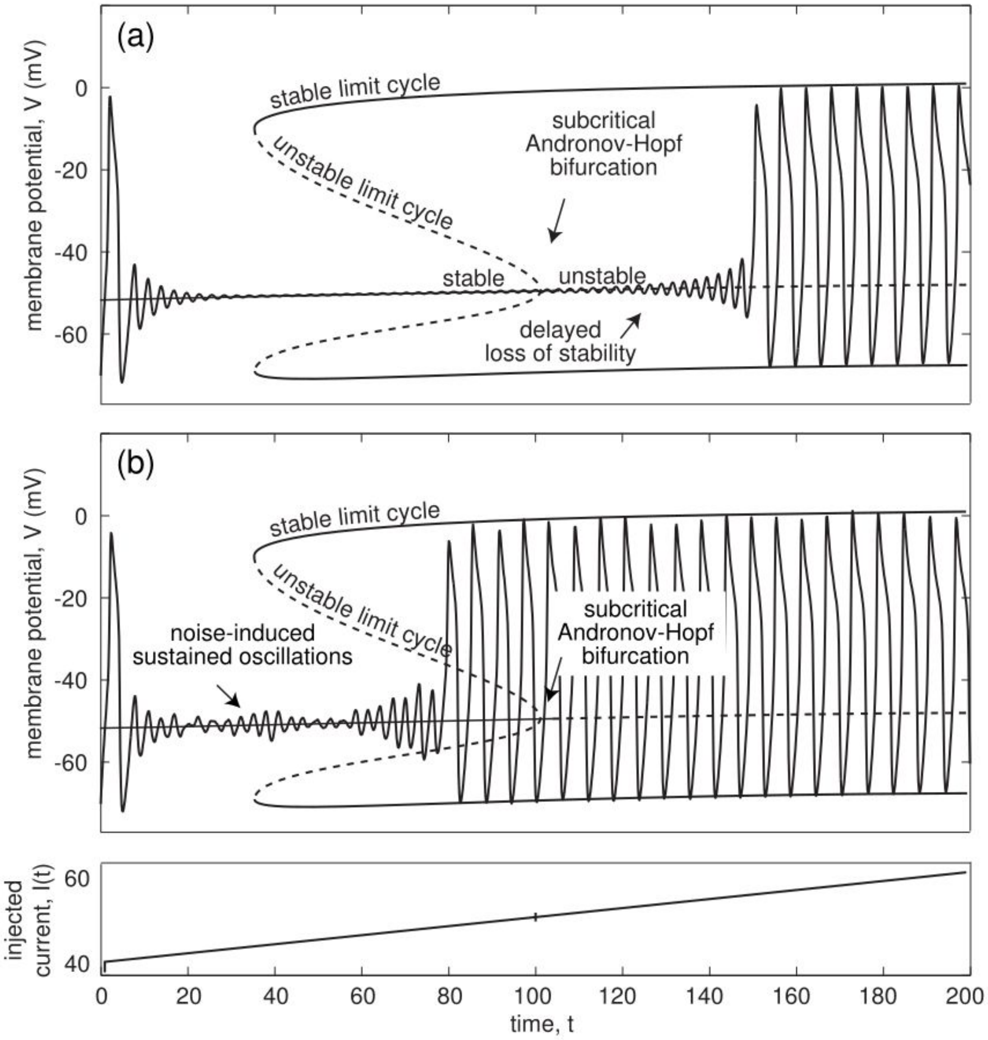
Current work

How can neural models bifurcate, and what are the physical interpretation of those bifurcations?

Future work

Short term: finish Dynamical Systems in Neuroscience textbook

Future work


(a)
stable limit cycle
unstable limit cycle
subcritical
Andronov-Hopf
bifurcation
stable
unstable
delayed
loss of stability
membrane potential, V (mV)
0
-20
-40
-60
(b)
stable limit cycle
unstable limit cycle
noise-induced
sustained oscillations
subcritical
Andronov-Hopf
bifurcation
membrane potential, V (mV)
0
-20
-40
-60
injected
current, I(t)
60
40
0
20
40
60
80
100
120
140
160
180
200
time, t

Future work

Long term:

- Formulate a theory of bifurcations in stochastic systems
- Develop an experimental method to test it
- Find neuron bifurcations