Multi-scale modelling of neural circuits

BRANDY School

Val di Sole

Prof. Thomas Nowotny (@drtnowotny)

CCNR and Sussex Neuroscience, School of Engineering and Informatics

UNIVERSITY OF SUSSEX

Also known as

Comp Neuro 101

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Why do we need models?

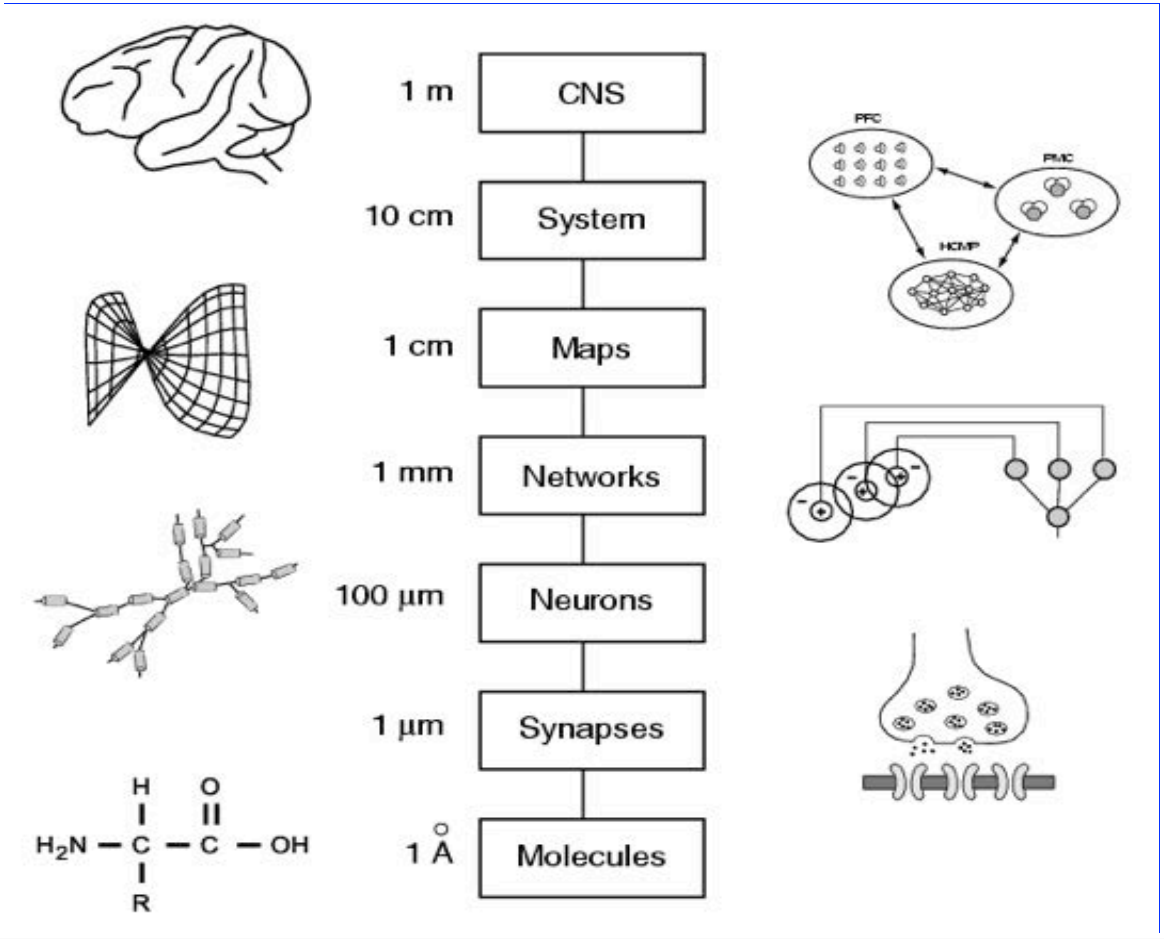
- Common misunderstanding: Modelling is **not** a form of hypothesis testing: **“garbage in, garbage out”**
- Forces us to make assumptions explicit.
- Enables many “virtual" experiments to be done, can pinpoint the one that is most crucial.
- Can lead to unexpected predictions.
- Often much quicker/easier to try out ideas, e.g. blocking a connection or channel type

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

What makes a good model?

- However, it's easy to make a bad alifey or physicsy model
- Good to have close contact with neuroscientists
- Model should not only replicate existing data but must also make new predictions about the biological system

Scales in the nervous system



Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Levels (scales) of modelling

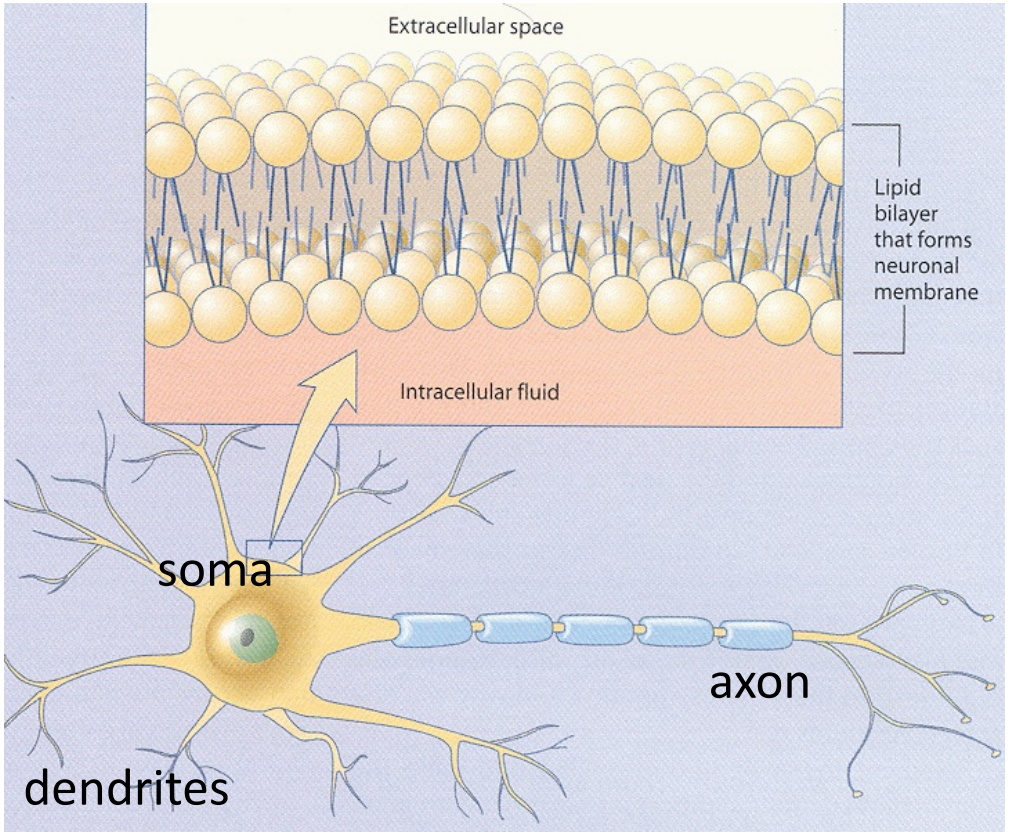
- Many different types of models: There is a continuum from very realistic to very abstract. All models must make simplifications to be useful.
- For instance, a model of a single neuron:
  - Binary threshold unit, continuous unit
  - Rate model, Integrate-and-fire model, Spiking neuron
  - A few compartments, many compartments
  - Individual channels, detailed model of channel dynamics
  - …

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Which model to use?

- Depends on purpose of the model! Different types are appropriate for different sorts of questions.
- Two mainstream approaches: Top-down vs bottom up:
  - **Top-down:** based on rational theoretical hypothesis to infer the mechanism of brain functions, i.e., start with an idea about abstract task / problem, figure out a good way to solve it and see if that's what the nervous system does.
  - **Bottom-up:** based on experimental data, i.e., look closely at the nervous system, try and figure out what it’s doing, derive the task/problem from there.

Modelling neurons

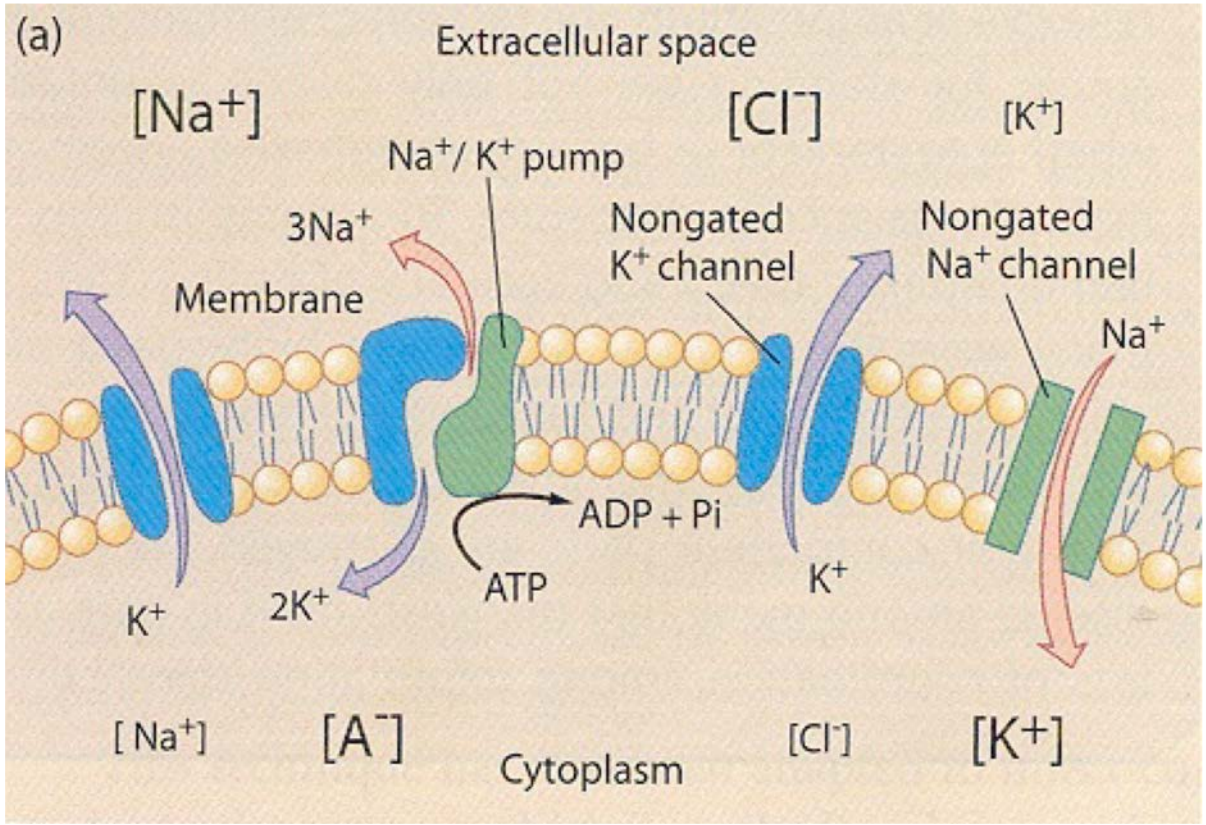


Neurons have a soma (cell body), dendrites (input) and axon (output).

We will abstract this to ”single compartment” or “point” neurons

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

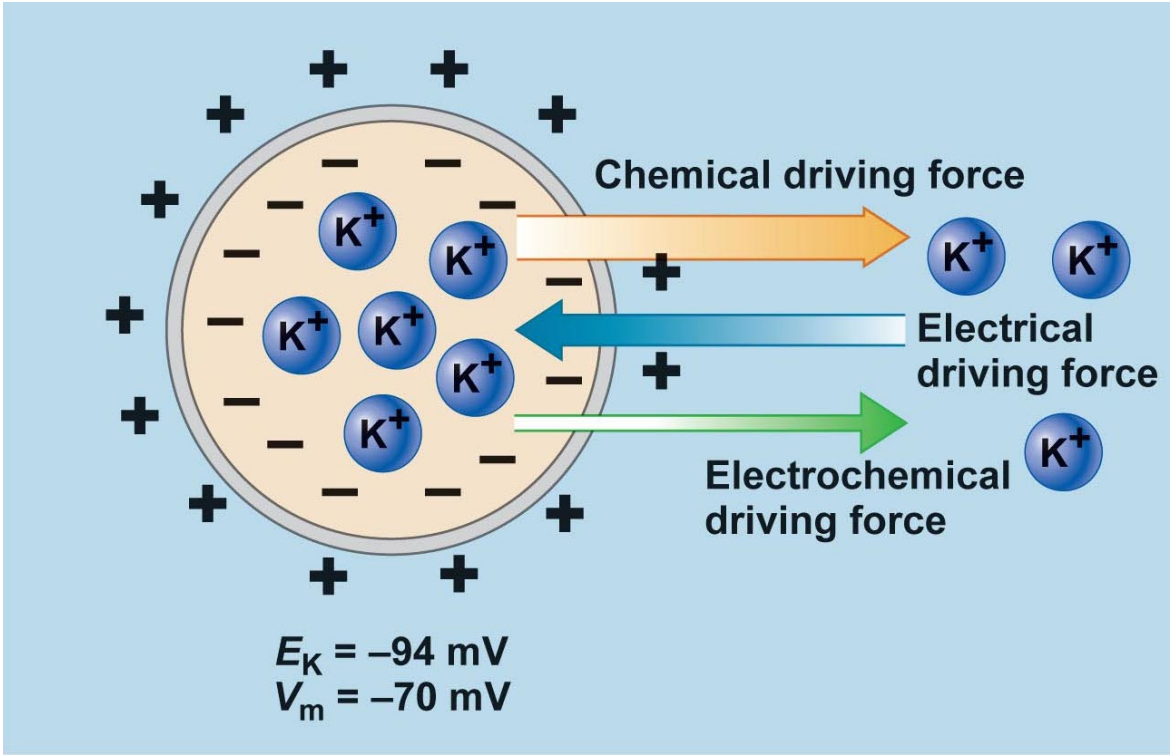
Ion channels



Ion channels in the membrane lower the effective membrane resistance by a factor of 10,000 (depending on density, type, etc.) Ion pumps maintain the differences in concentrations inside and outside by expending energy.

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

The membrane potential



Exterior potential 0 by convention

Excess of positive charge outside -> resting membrane potential is negative.

Difference in concentration: ions diffuse through channels (when open)

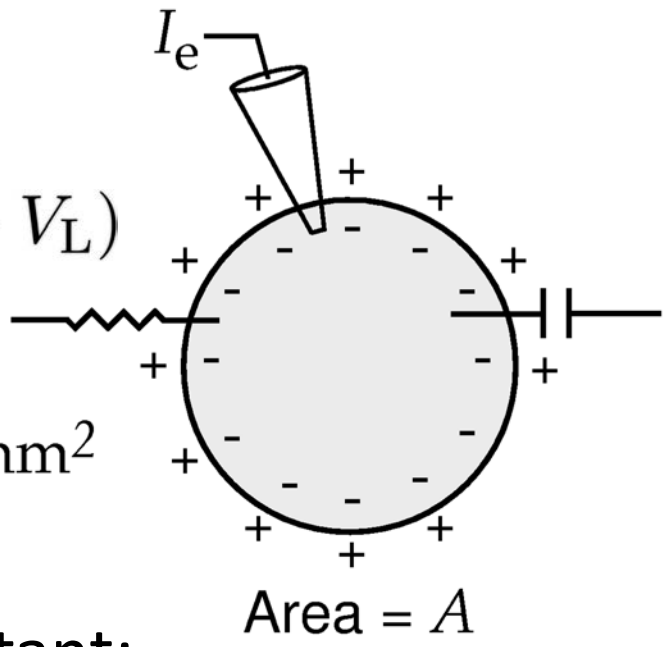
Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Conductance based model

$$I_{\mathrm{L}} = g_{\mathrm{L}}(V - V_{\mathrm{L}})$$

$$g_{\mathrm{L}} = A/r_m$$

$$r_{\mathrm{m}} \approx 1\ \mathrm{M\Omega\ mm^2}$$



$$Q = C_{\mathrm{m}}V$$

$$C_{\mathrm{m}} = c_{\mathrm{m}}A$$

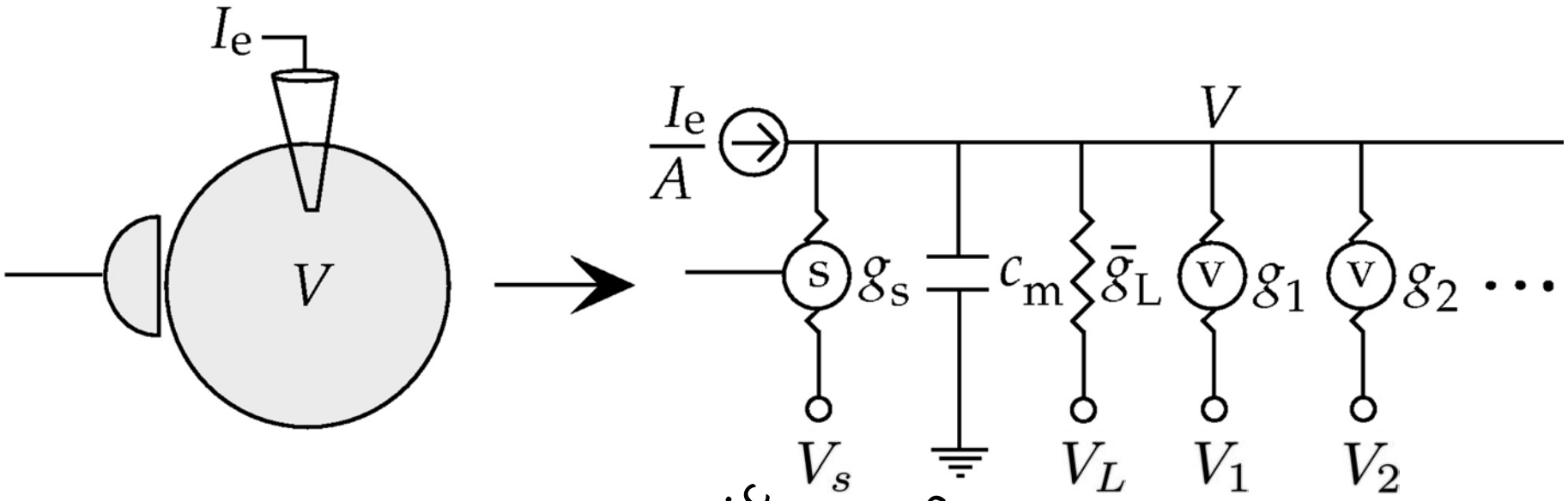
$$c_{\mathrm{m}} \approx 10\ \mathrm{nF/mm^2}$$

If C_m is constant:

$$C_m \frac{dV}{dt} = \frac{dQ}{dt} = I = I_e - g_{\mathrm{L}}(V - V_{\mathrm{L}})$$

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Adding ion channels



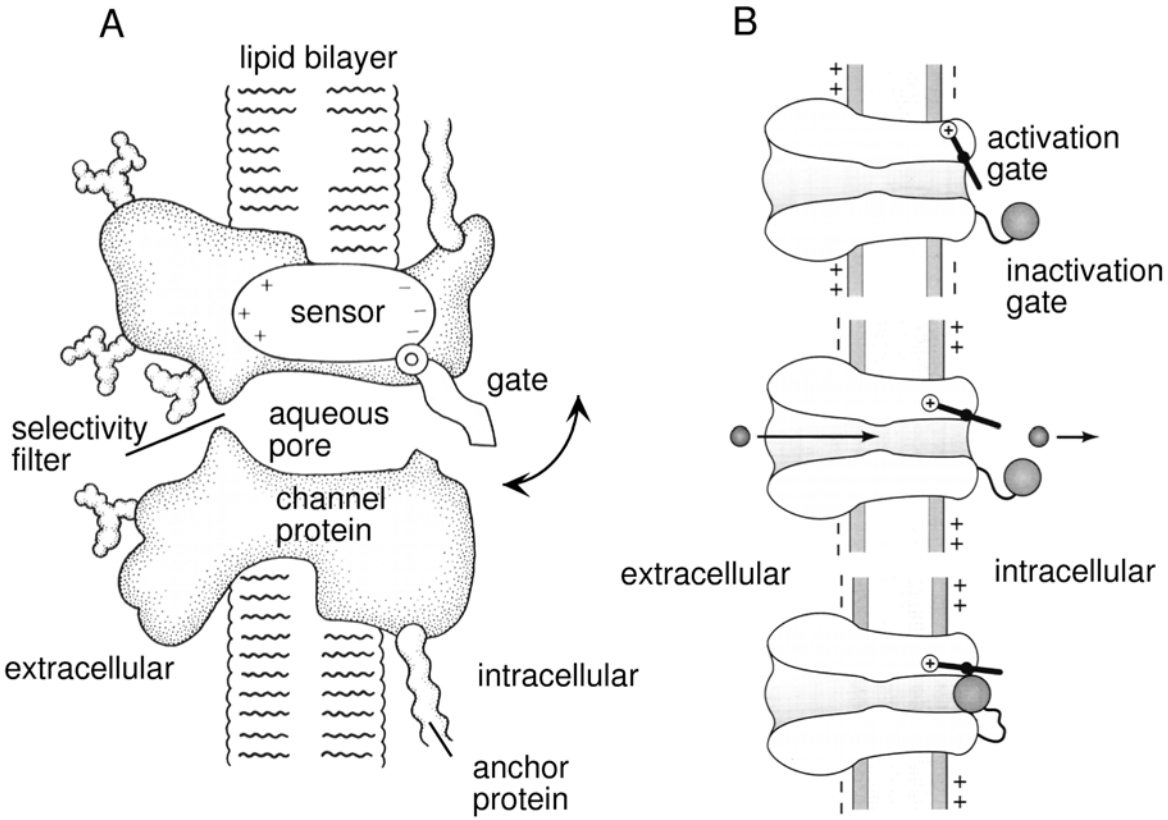
maximal conductance

$$I_x = -g_x a_x (V - V_x)$$

ionic current activation reversal potential

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Persistent and transient currents



Delayed rectifier Sodium current

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Hodgkin-Huxley model

Non-inactivating delayed rectifier current

$$I_{\mathrm{Kd}} = g_{\mathrm{Kd}} n^4 (V_K - V)$$

$$\frac{dn}{dt} = \alpha_n (1 - n) - \beta_n n$$

$$\alpha_n = \frac{0.01(10 - V)}{\exp(\frac{10-V}{10}) - 1} \qquad \beta_n = 0.125 \exp\left(\frac{-V}{80}\right)$$

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Hodgkin-Huxley model

Sodium current

$$I_{\mathrm{Na}} = g_{\mathrm{Na}} m^3 h (V_{\mathrm{Na}} - V)$$

$$\frac{dm}{dt} = \alpha_m (1 - m) - \beta_m m \qquad \text{“activation”}$$

$$\frac{dh}{dt} = \alpha_h (1 - h) - \beta_h h \qquad \text{“inactivation”}$$

$$\alpha_m = \ldots \quad \beta_m = \ldots \quad \alpha_h = \ldots \quad \beta_h = \ldots$$

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Hodgkin-Huxley model

Is often rewritten in terms of

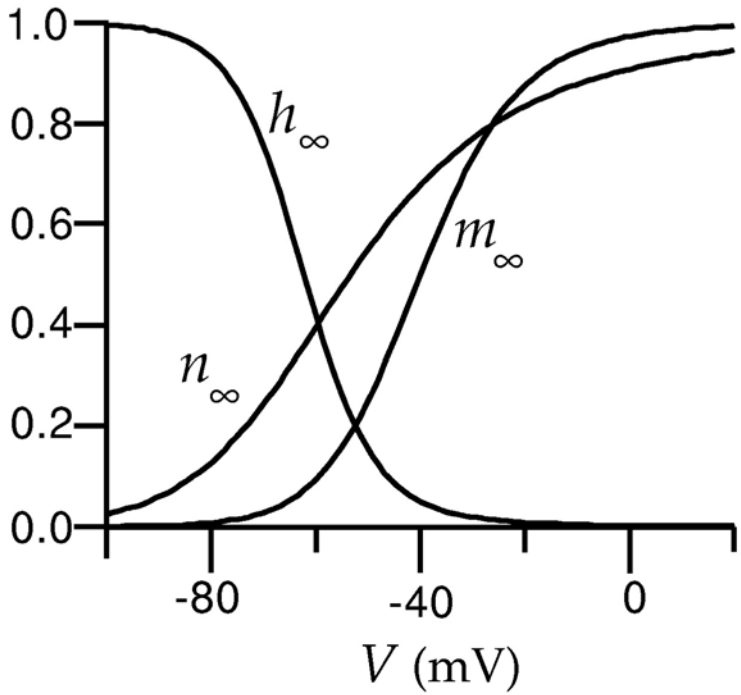
$$n_\infty = \frac{\alpha_n}{\alpha_n + \beta_n}$$

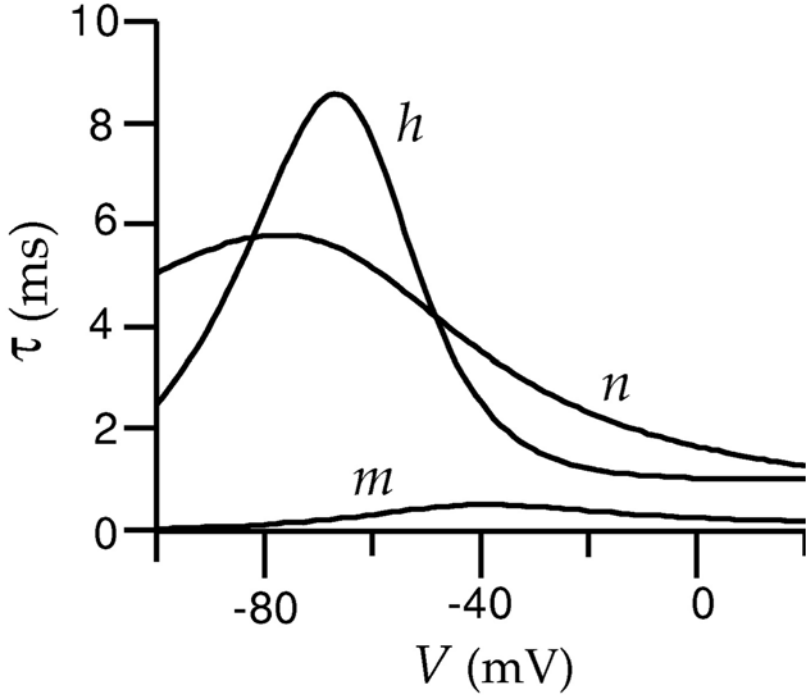
$$\tau_n = \frac{1}{\alpha_n + \beta_n}$$

$$\Rightarrow \frac{dn}{dt} = \frac{1}{\tau_n}(n_\infty - n)$$

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Hodgkin-Huxley model

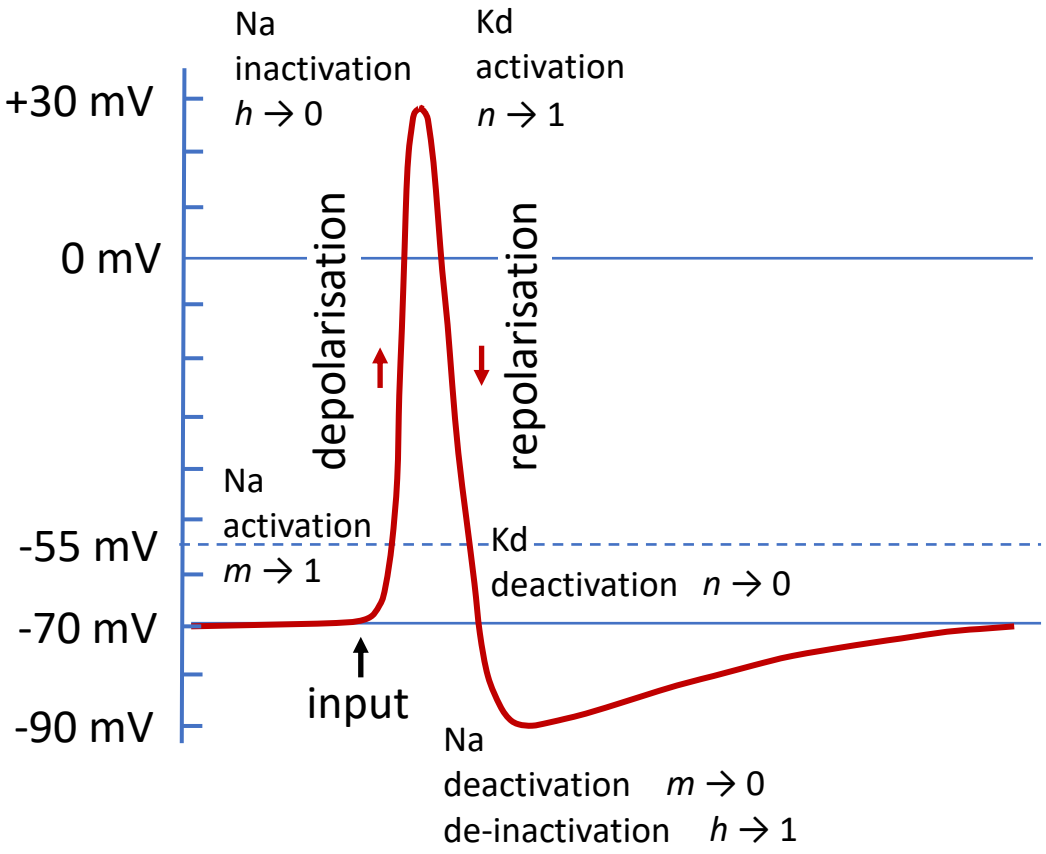




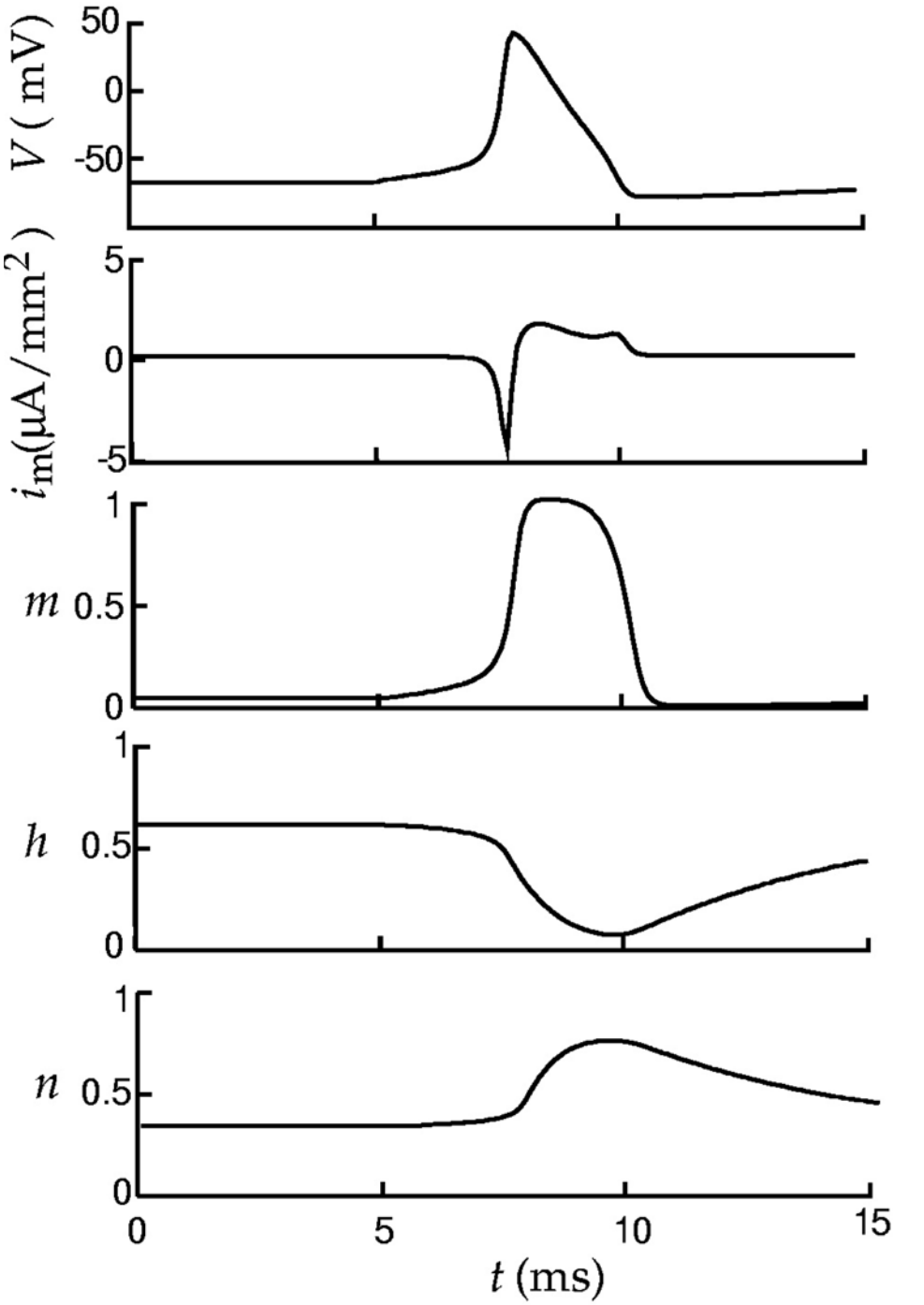
$$\frac{dV}{dt} = \frac{1}{C}(I_e + I_{\mathrm{Na}} + I_{\mathrm{Kd}} + I_L)$$

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Action potential



Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX



Action Potential

Hodgkin-Huxley model

- The original model is “numerically not very nice”
- Is a type II neuron (Hopf bifurcation)
- Today, not used much
- One of the most-used models is the HH-like model by Traub & Miles (1991), which has type I behaviour (saddle-node bifurcation)

Traub-Miles model

- **In practice, it works very well**
- Surprisingly, it has numerical instabilities:

$$\alpha_n = 0.032 \frac{-50 - V}{\exp\left(\frac{-50-V}{5}\right) - 1}$$

  Unclear at *V=-50*.

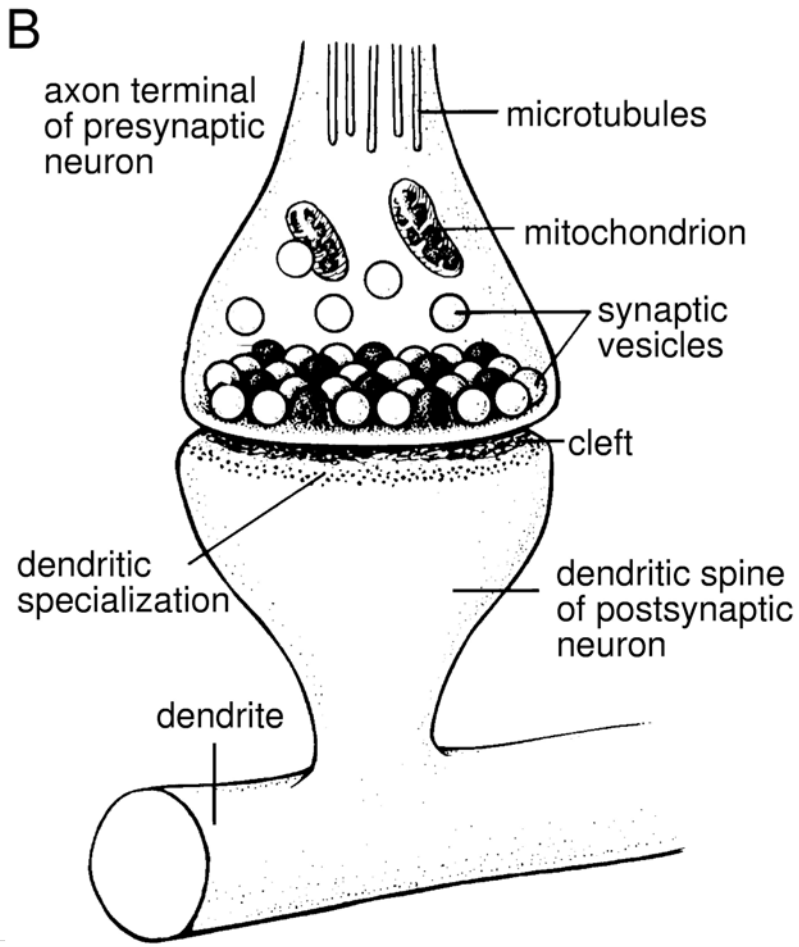
- L'Hôpital shows that it's well-defined and continuous

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Modelling networks

- Action potential travel down axons (mostly not modeled explicitly)
- Axons make contacts with target neuron's dendrites at Synapses
- Synapses can – depending on transmitter – be excitatory (depolarize target neuron) or inhibitory (hyper-polarize)

Synapses



Synaptic transmission is by chemical transmitters for most synapses

Depolarization of the pre-synaptic terminal leads to transmitter release across the cleft

Post-synaptic channels are opened by the transmitter

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Modeling synapses

Fraction *s* of released transmitter:

$$\frac{ds}{dt} = -\beta s + \alpha(1-s)\mathbf{1}_{[t,t+t_r]}$$

Synaptic current into post-synaptic neuron:

$$I_{\text{syn}} = g_{\text{syn}} s (V_{\text{rev}} - V_{\text{post}})$$

How to use HH models

- Used for numerical simulations:
  - Each neuron:

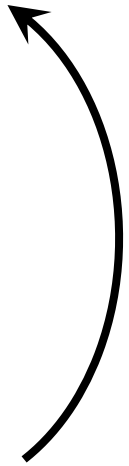
$$\frac{dV_j}{dt} = \dots$$

  - Each pre-synapse:

$$\frac{ds_j}{dt} = \dots$$

  - Each post-synapse:

$$I_{i,\mathrm{syn}} = \sum_j g_{ij} s_j (V_{\mathrm{rev}} - V_i)$$



Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

How to simulate …

- Euler Algorithm

$$V_i(t + \Delta t) = V(t) + \frac{dV_i}{dt}\Delta t + \mathcal{O}(\Delta t^2)$$

- Good if equations not too stiff, small time steps
- Otherwise: Runge-Kutta algorithms, implicit algorithms, …

Simplified models

- One can work with *V* as the main variable (and many people/studies do)
- But for larger scale/ different analysis one can use simplified models

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

A zoo of models …

- Morris-Lecar
- Fitzhugh-Nagumo
- (leaky) integrate-and-fire (IF/ LIF), exponential, adExp, GIF

Reduced HH

- McCulloch-Pitts ← discrete time/ probabilistic
- Hindmarsh-Rose
- Izhikevich, Rulkov

"abstract spiking"

- Wilson-Cowan ← rate models
- Kuramoto ← phase oscillators

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Integrate-and-Fire

- Simple membrane equation – passive only $\dot{V} = -g_L(V_L - V)$

- Explicit spiking mechanism:
  if $V > V_{\mathrm{th}}$: spike fired,
  reset: $V \to V_{\mathrm{reset}}$

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

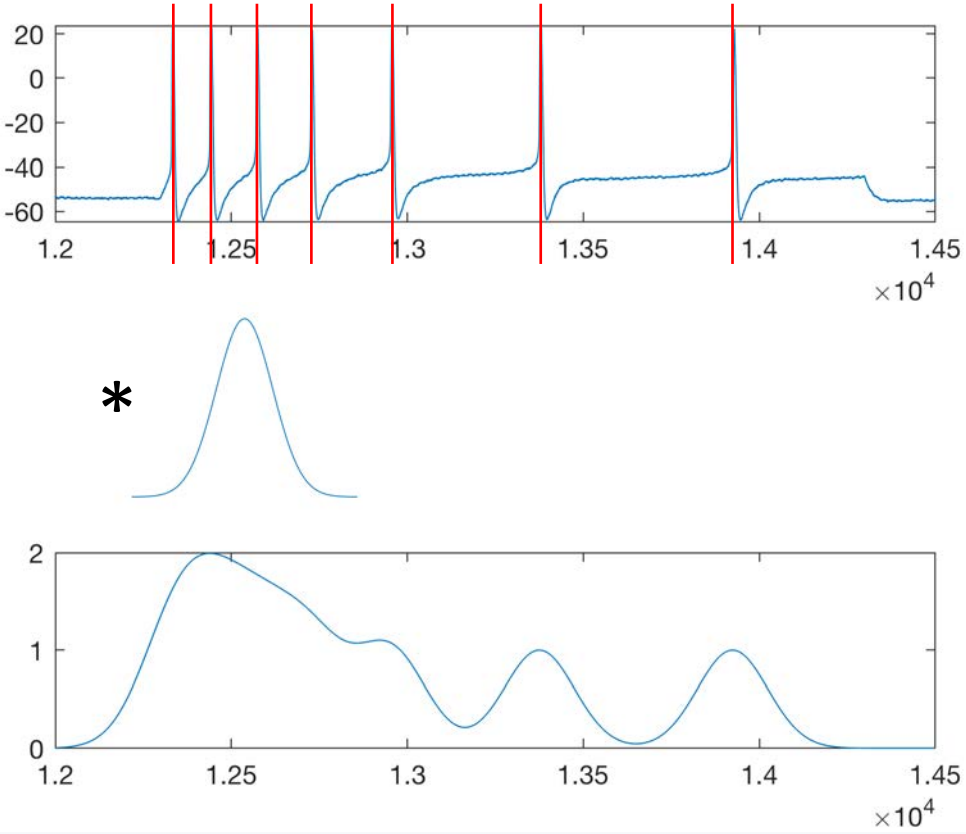
Integrate-and-Fire

- Faster to simulate
- Allows some analytical work:
  - Event - based simulation
  - General analysis with theory of stochastic processes
- Suitable for neuromorphic devices

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Different views of the same activity

- For synaptic transmission only spikes matter



Trace of a B1 neuron in the pond snail

Detect spikes

Convolve with a kernel

Equivalent “spike density function” – a measure of spike rate

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Quantitative rate reduction

- We start from a full, conductance-based model:

$$C\dot{V}_i = -I_{\mathrm{Na}} - I_K - I_L - I_M - I_{i,\mathrm{ext}} - I_{i,\mathrm{syn}}$$

- Synapses are modelled by

$$\dot{s}_i = -\beta s_i + \alpha \sum_k \mathbf{1}_{[t_k, t_k + t_r]}$$

$$I_{i,\mathrm{syn}} = \sum_j g_{ij} s_j (V_i - V_{\mathrm{rev}})$$

Buckley & Nowotny, PRL 106, 238109 (2011)

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Step 1: f-I curve

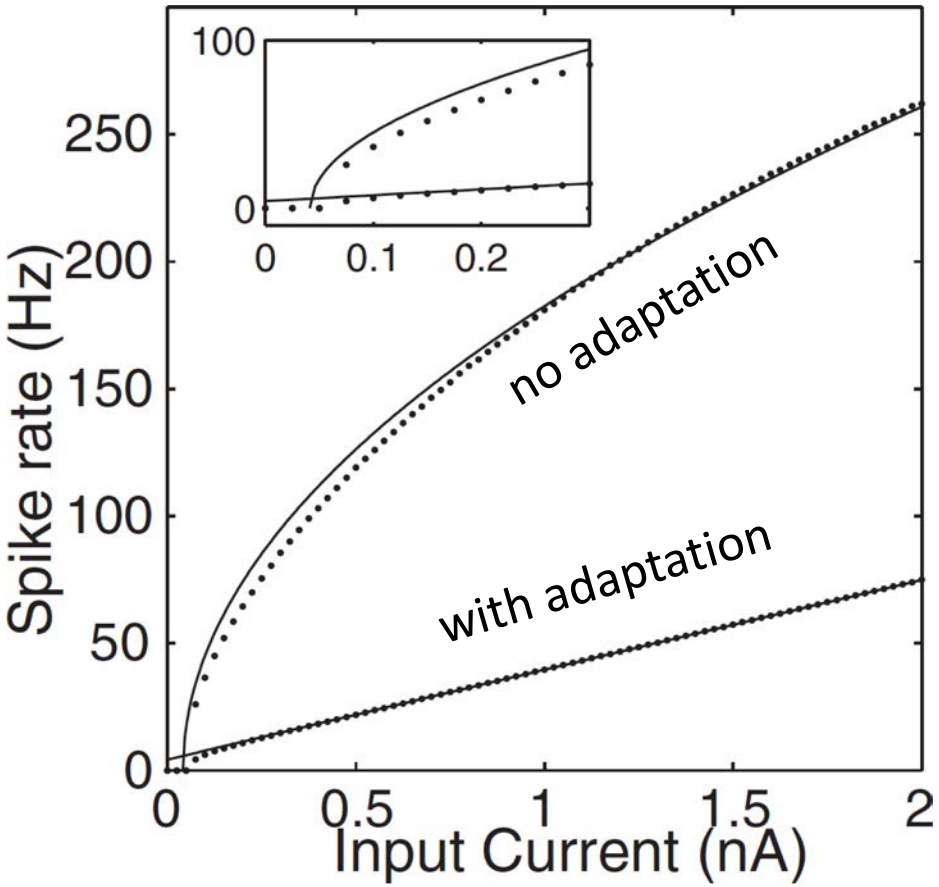
It is common to characterize neurons by their so-called F-I curve:

- For type 1 (saddle-node bifurcation) neurons:

$$F(I) \sim \sqrt{I - I_0}$$

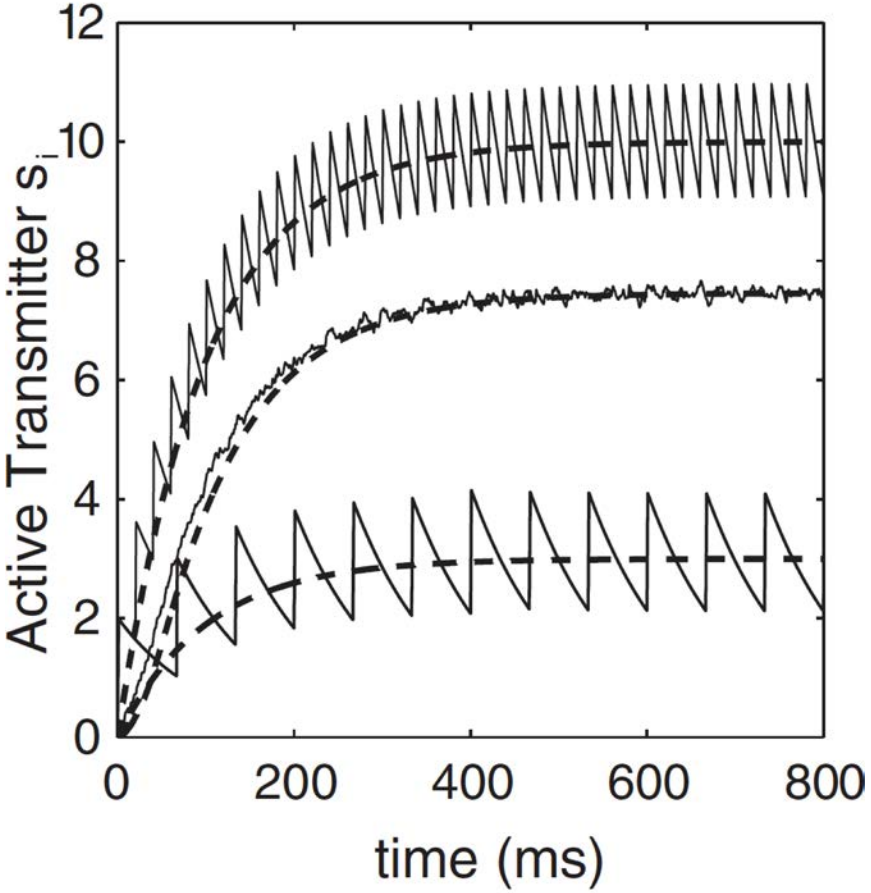
- For type 1 with adaptation: linear

$$F(I) = [mI + C]_+$$



Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Step 2: Effective synapse activation



Synapse activation driven by a constant spike train with frequency F:

$$\dot{s}_i = -\beta s_i + \alpha t_r F$$

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Step 3: Putting it all together

Insert the F-I curve into the ***s*** equation:

$$\dot{\mathbf{s}} = -\beta\mathbf{s} + \alpha m t_r[-G\mathbf{s} + \theta + \mathbf{I}]_+$$

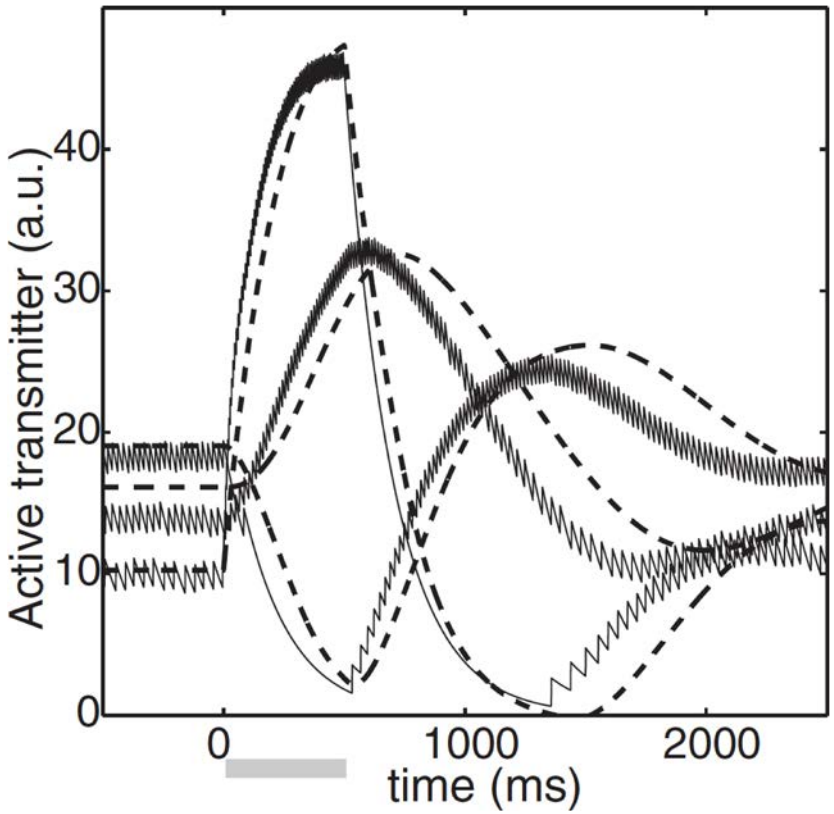
$$\dot{\mathbf{s}} = -\beta\mathbf{s} + \gamma[-G\mathbf{s} + \theta + \mathbf{I}]_+$$

$$\mathbf{F} = m[-G\mathbf{s} + \theta + \mathbf{I}]_+$$

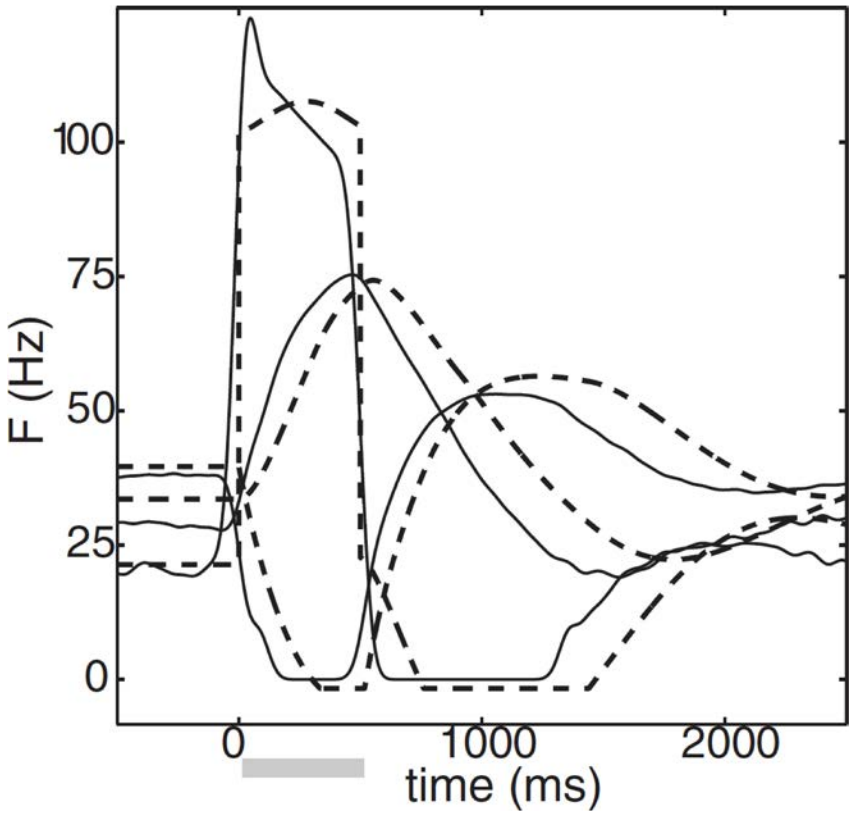
Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Examples

s



F



Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

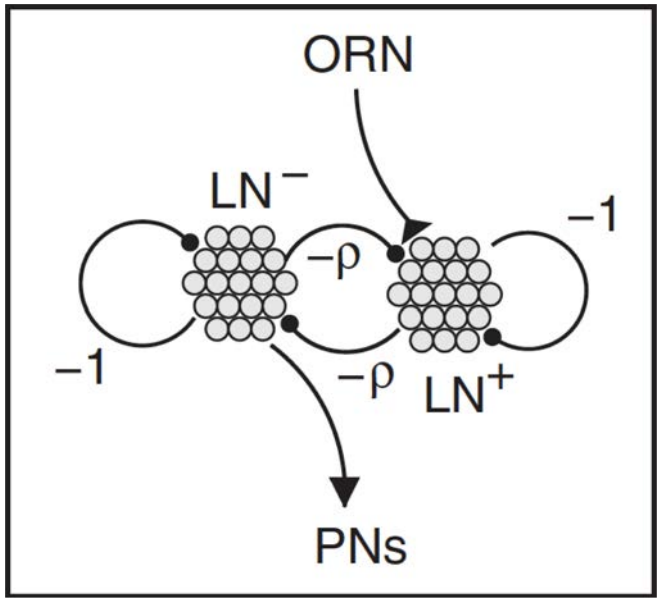
So what?

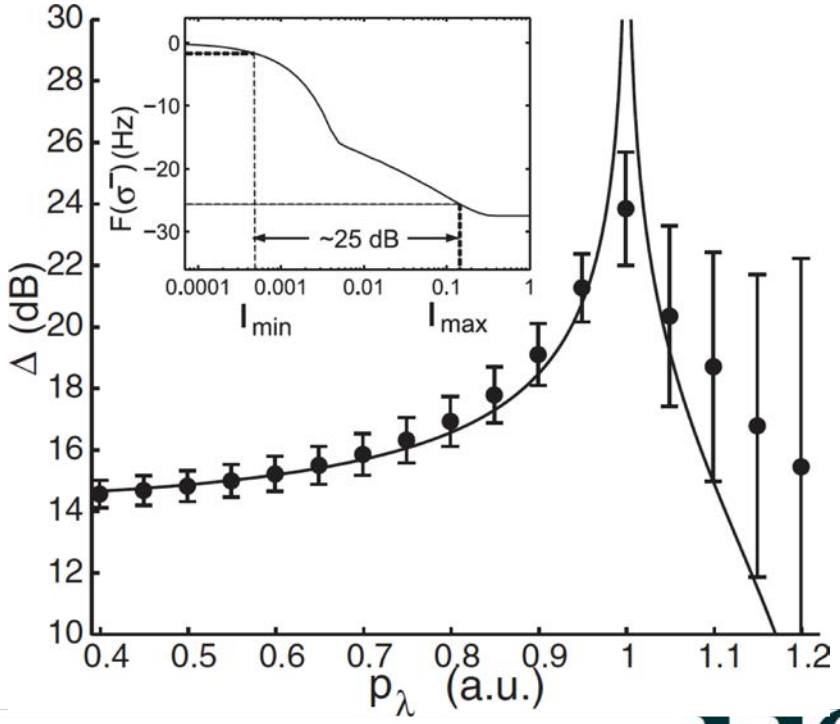
- The rate equations can be further reduced to mean field models
- These can be analyzed analytically
- E.g. investigate global stability / "dynamical systems notion" of criticality – see Buckley & Nowotny, PRL 106, 238109 (2011)

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Example result

- Maximal dynamic range close to a bifurcation ("minimally stable global fixed point")





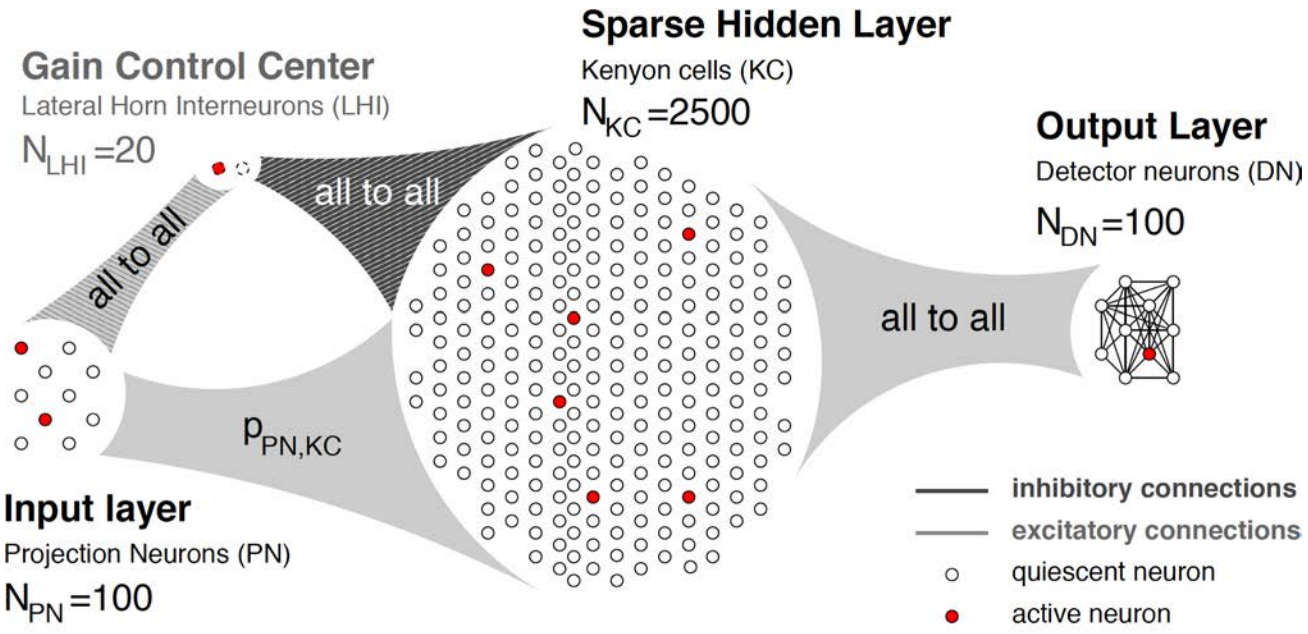
Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Software

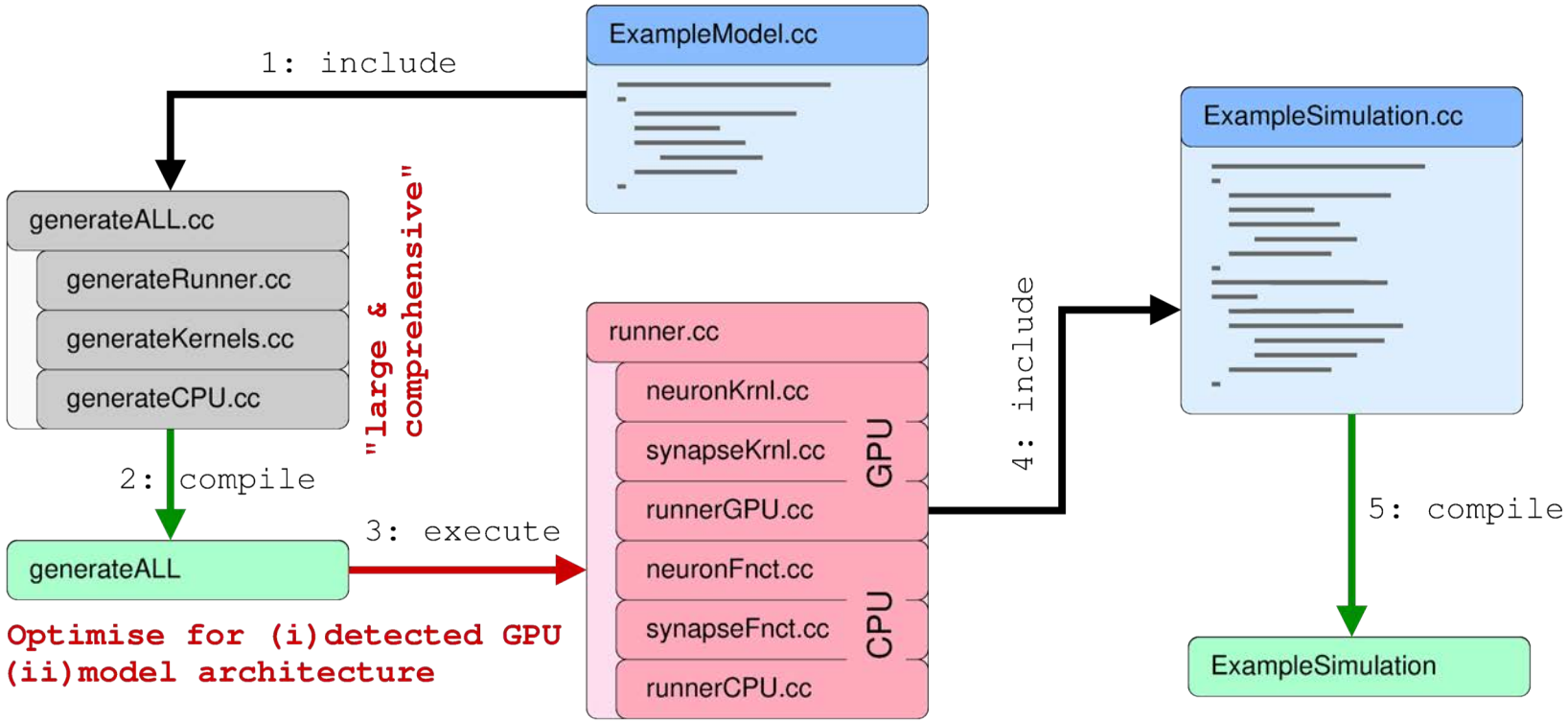
- There are many packages for neuronal network simulation:
  - Neuron, Genesis
  - Brian/ Brian 2
  - NEST
  - GeNN (GPU enhanced neuronal networks), Brian2GeNN, SpineML2GeNN

Origins of GeNN

- In 2010 I tried to implement a network on CUDA from scratch
- Achieved a 24x speed up over CPU
- It took a month to implement an existing model (after learning how to use CUDA)
- The program was optimised for a particular GPU
- It was designed for one size of the simulation

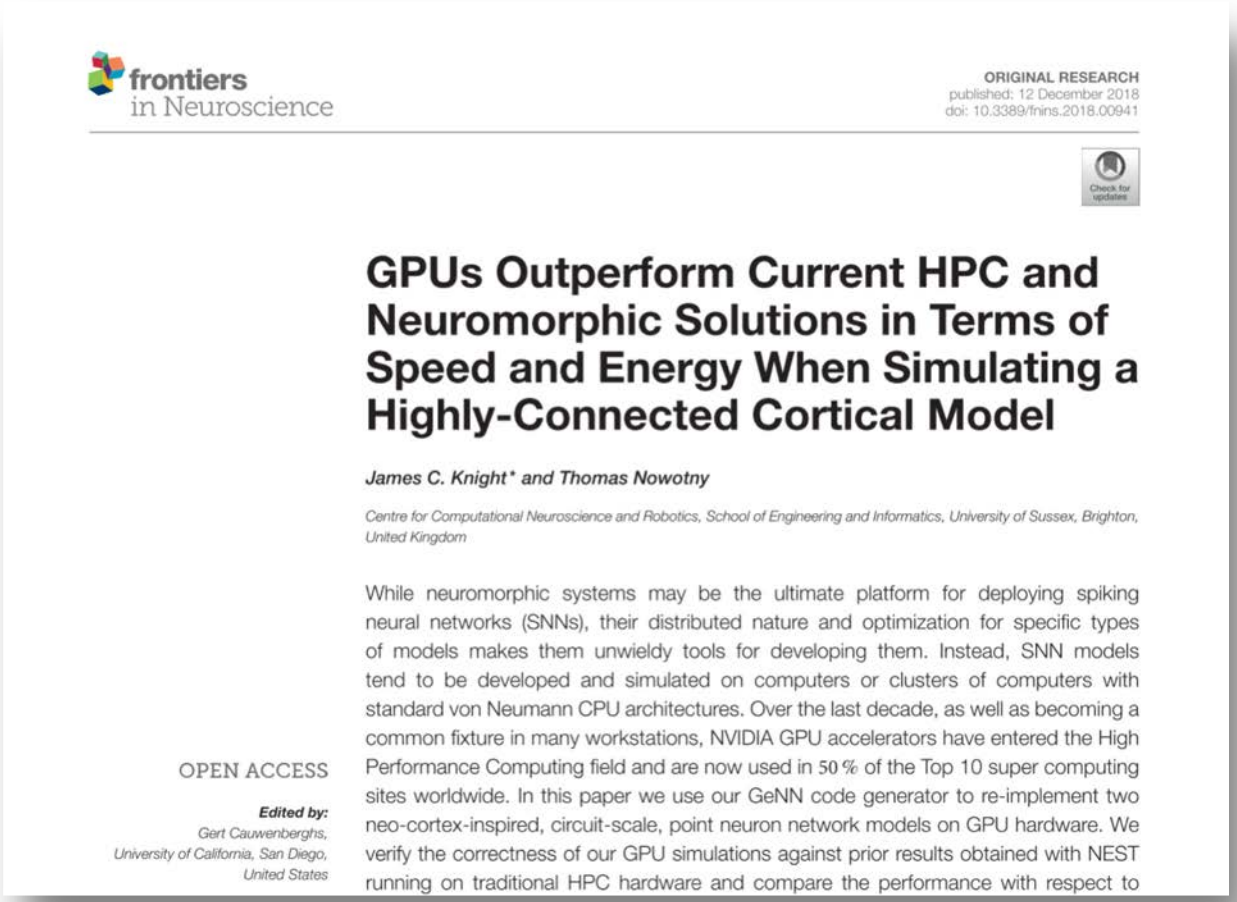


How GeNN works



Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Latest version outperforms HPC



frontiers in Neuroscience

ORIGINAL RESEARCH
published: 12 December 2018
doi: 10.3389/fnins.2018.00941

Check for updates

GPUs Outperform Current HPC and Neuromorphic Solutions in Terms of Speed and Energy When Simulating a Highly-Connected Cortical Model

James C. Knight and Thomas Nowotny*

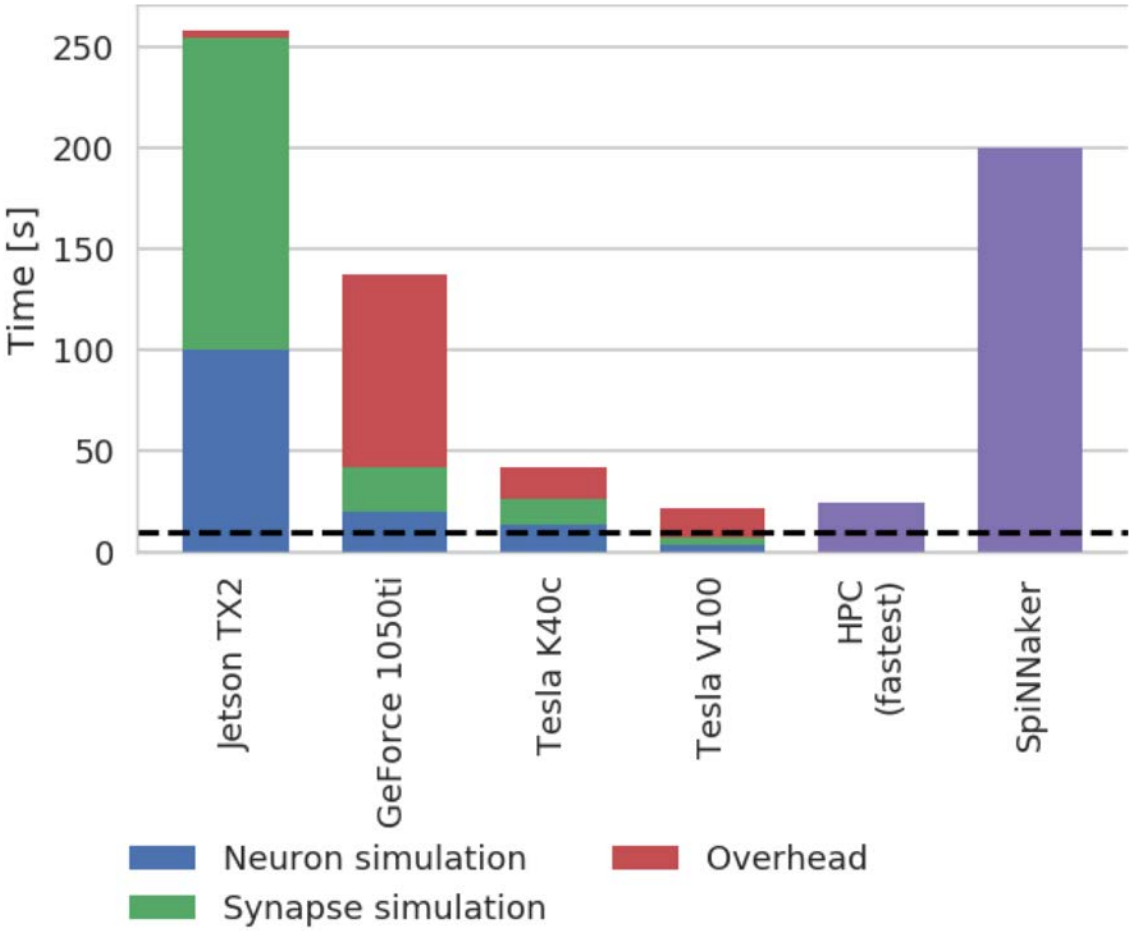
Centre for Computational Neuroscience and Robotics, School of Engineering and Informatics, University of Sussex, Brighton, United Kingdom

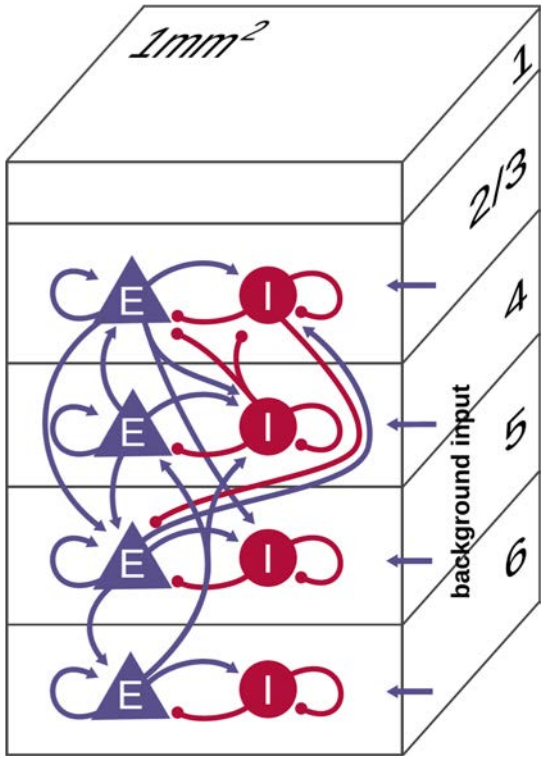
While neuromorphic systems may be the ultimate platform for deploying spiking neural networks (SNNs), their distributed nature and optimization for specific types of models makes them unwieldy tools for developing them. Instead, SNN models tend to be developed and simulated on computers or clusters of computers with standard von Neumann CPU architectures. Over the last decade, as well as becoming a common fixture in many workstations, NVIDIA GPU accelerators have entered the High Performance Computing field and are now used in 50 % of the Top 10 super computing sites worldwide. In this paper we use our GeNN code generator to re-implement two neo-cortex-inspired, circuit-scale, point neuron network models on GPU hardware. We verify the correctness of our GPU simulations against prior results obtained with NEST running on traditional HPC hardware and compare the performance with respect to

OPEN ACCESS

Edited by:
Gert Cauwenberghs,
University of California, San Diego,
United States

Simulation performance

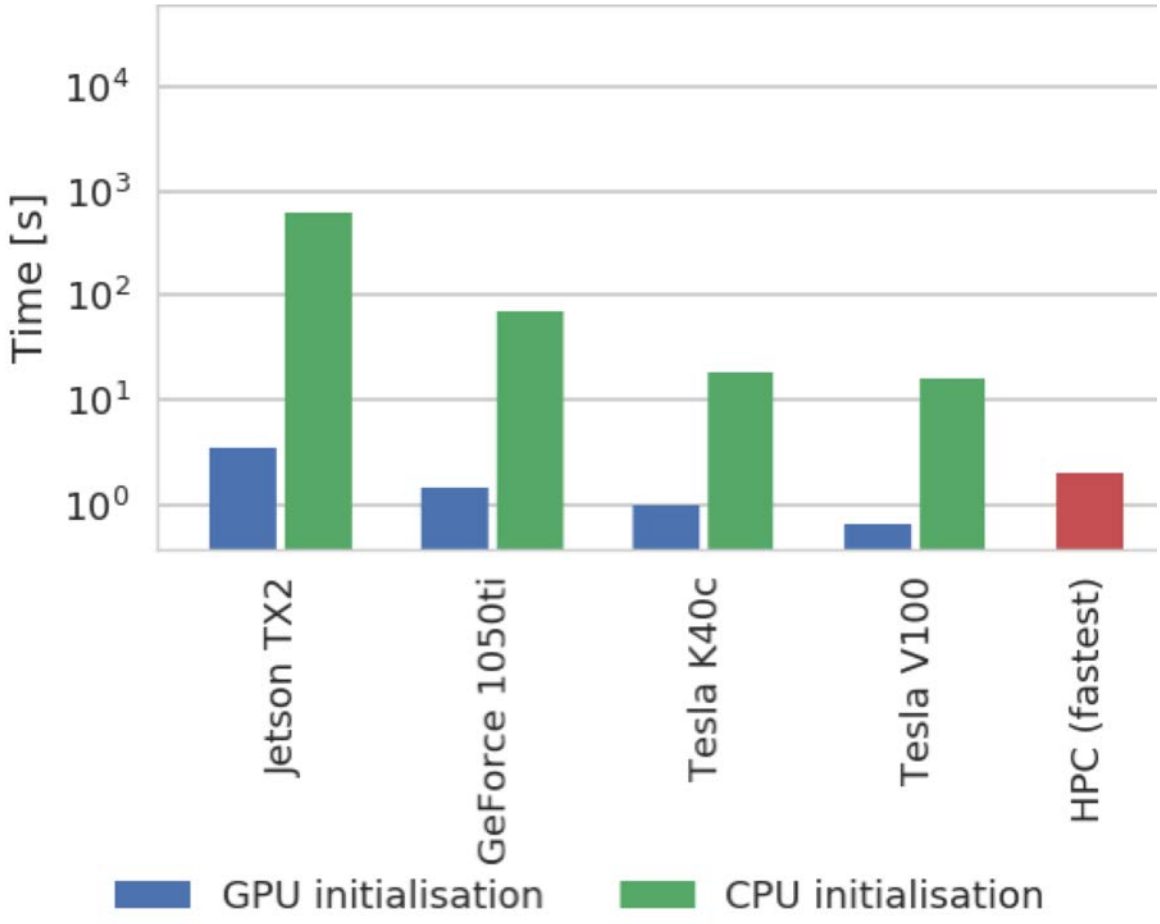




~ 80,000 neurons
300M static synapses

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Network setup time



Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX

Next time …

- A bit about insect olfaction and a odour object recognition
- Some introduction to our work in the Brains on Board project regarding insect navigation and autonomous robots

Prof. Thomas Nowotny (@drtnowotny)
CCNR and Sussex Neuroscience, School of Engineering and Informatics
UNIVERSITY OF SUSSEX