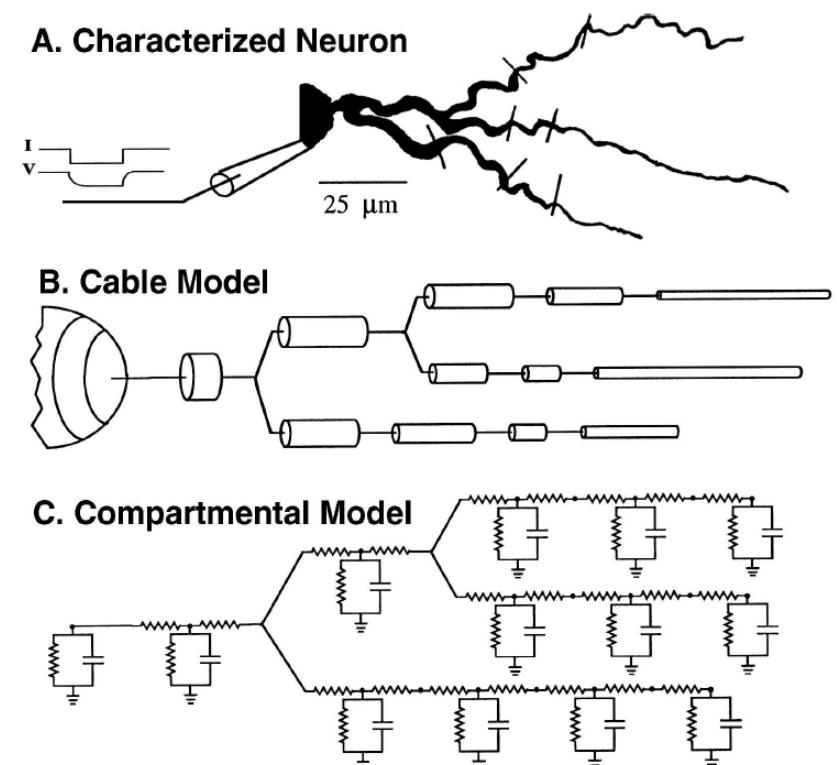


Cable theory

- Cable theory is concerned with ***how inputs propagate*** to the soma or the axon initial segment, ***how these inputs interact*** with one another, and how the placement of an input on a dendritic tree affects its ***functional importance*** to the neuron.



Where can you find the details?

- Find Neuronal Morphologies in NeuroMorpho

<http://neuromorpho.org/>

- Find models in ModelDB

<https://senselab.med.yale.edu/modeldb/>

- Find neuron's electrophysiological properties

<http://neuroelectro.org/>

- Allen Institute

<http://portal.brain-map.org/>

- Choose your channels using
IonChannelGenealogy

<https://icg.neurotheory.ox.ac.uk/>

- Blue Brain Portal

<https://portal.bluebrain.epfl.ch/>

The NEURON simulation environment

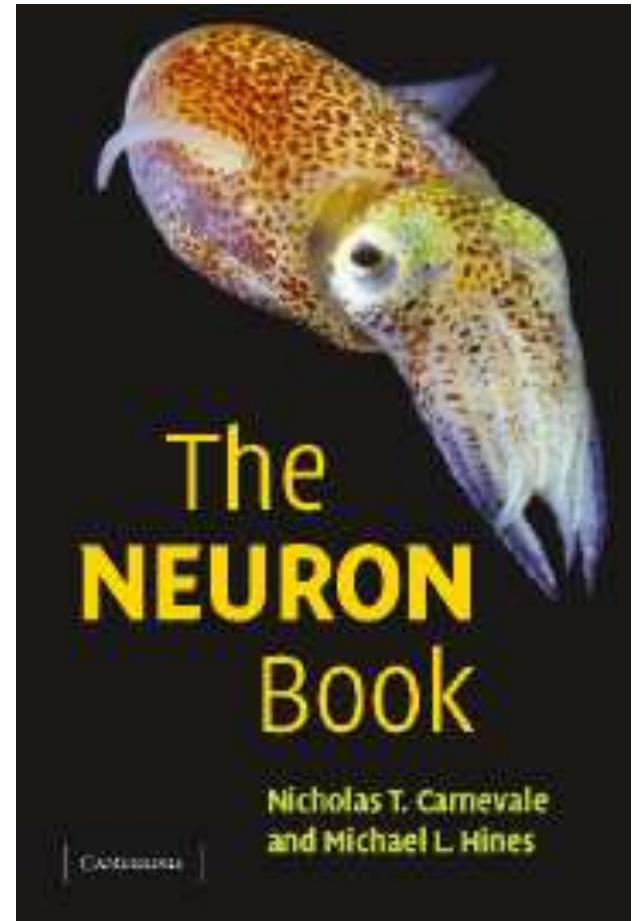
NEURON is a simulation environment for models of individual neurons and networks of neurons that are closely linked to ***experimental data***.

Documentation:

https://www.neuron.yale.edu/neuron/static/new_doc/index.html

- *Neuron with Python*

from neuron import h, gui



The main features of NEURON

- Graphical User Interface (GUI)
- Two programming languages:
 1. NMODL (MOdel Description Language->MODL)
Ion channel mechanisms (Hodgkin-Huxley-like kinetics, calcium-dependent kinetics, synaptic properties and others).
 2. All other operations in NEURON are performed using scripts written in HOC (High Order Calculator).

The main features of NEURON

- Defining the **anatomical** and **biophysical** properties of models of neurons and neuronal networks,
- Controlling simulations etc.,

e.g.

```
soma=h.Section(name='soma')  
soma.diam=10      #μm  
soma.L=3.18#μm
```

Distributed (or Density) Mechanisms in hoc

- Properties that are distributed over the cell surface (e.g. membrane capacitance, active and passive ionic conductances) or throughout the cytoplasm (e.g. buffers).

e.g.

```
soma=h.Section(name='soma')
```

```
soma.diam=10      #μm
```

```
soma.L=3.18 #μm
```

```
soma.cm=1      #membrane capacitance, μF/cm2
```

```
soma.Ra=100    #axial resistance, ohm cm
```

```
soma.insert('hh') #HH channels
```

Point Processes in hoc

Synapse or electrode for passing current (current clamp or voltage clamp) is represented by a **point** source of current which is associated with a localized conductance.

- Syntax:

`varname = h.Classname(section_name(x))`

`varname.attribute = value`

Point Processes in hoc

Example 1

Current clamp

```
ic = h.IClamp(soma(0.5))  
ic.del = 100 #ms  
ic.dur = 200 #ms  
ic.amp = 0.1 #nA
```

Example 2

Synapse

```
syn = h.ExpSyn(soma(0.5))  
syn.e = 0 #mV
```

Example 3

NetStim

```
ns = h.NetStim(0.5)  
ns.start = 100 #ms  
ns.number = 2  
ns.interval = 10 #ms
```

Example 4

NetCon

```
nc = h.NetCon(source,target, [threshold, delay, weight])
```

Example 5

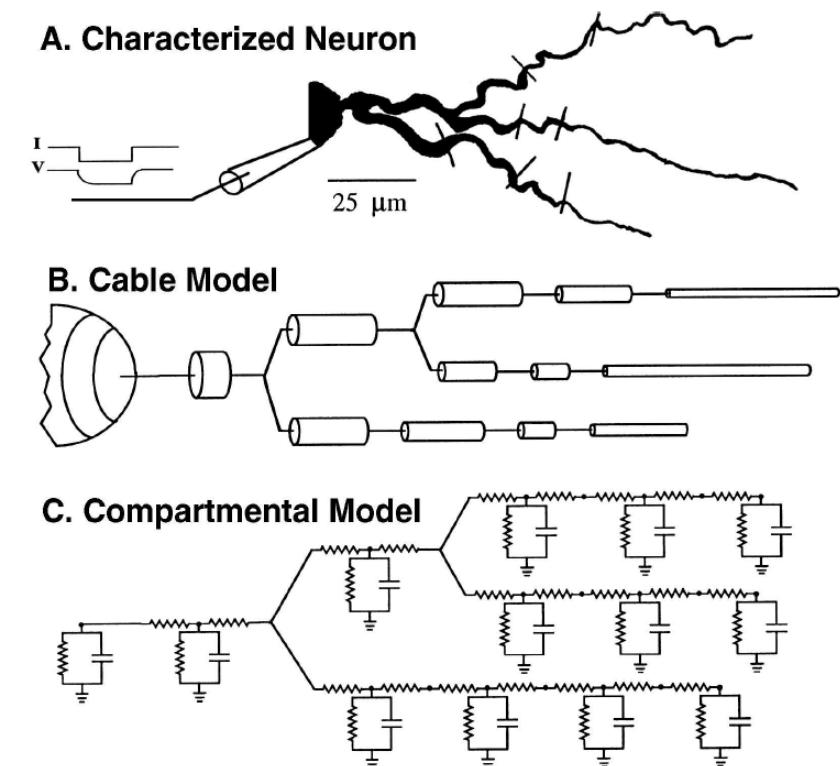
SEClamp

```
se= h.SEClamp(section(x))  
se.dur = 200 #ms  
se.amp = -20 #mV
```

Cable theory

- Connecting compartments
`child.connect(parent, [0 or 1])`
e.g.
`dend0.connect(soma(0),0)`

Cable theory is concerned with *how inputs propagate to the soma or the axon initial segment, how these inputs interact* with one another, and how the placement of an input on a dendritic tree affects its *functional importance* to the neuron.



How inputs propagate / interact

Jupyter notebook: Exc_1a

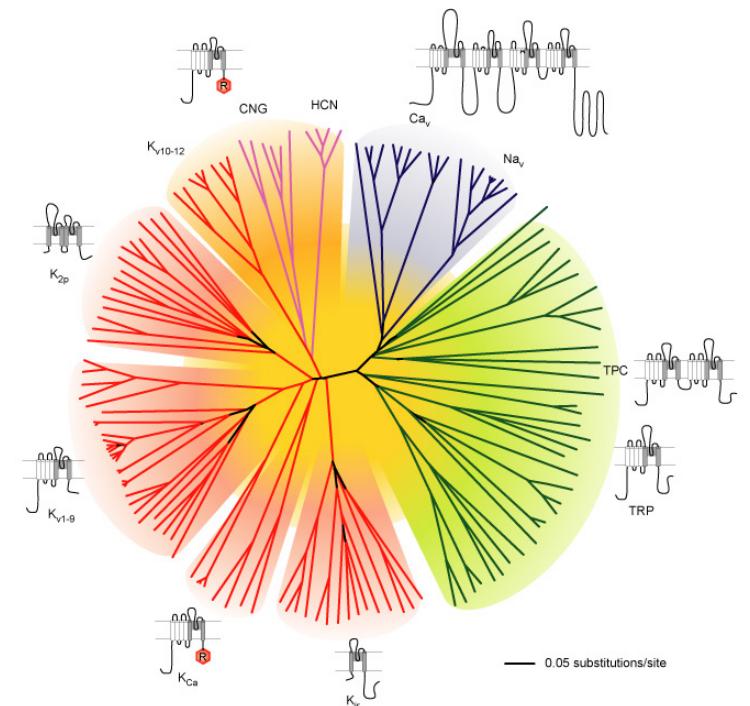
How inputs interact: Segregation

Jupyter notebook: Ex1b

Active mechanisms

- Passive ion channels -- no change in permeability due to external factors
- Active ion channels -- permeability is dependent upon factors such as:
 - Membrane potential (e.g., voltage-dependent)
 - Ionic concentrations (e.g., calcium-dependent)
 - Ligands (e.g., neurotransmitters)
- Which ions matter for neurons?

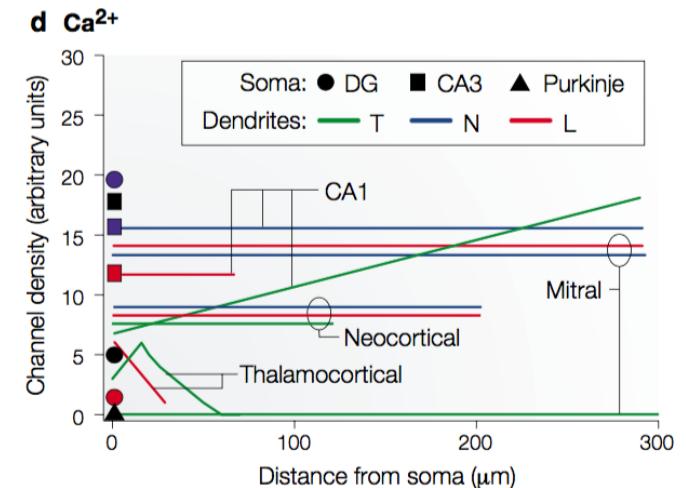
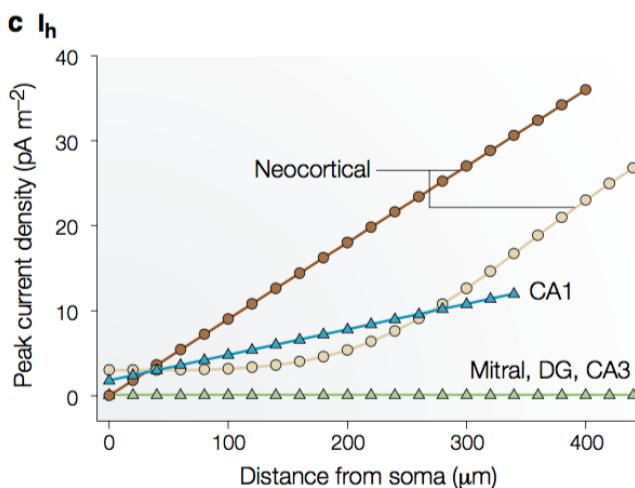
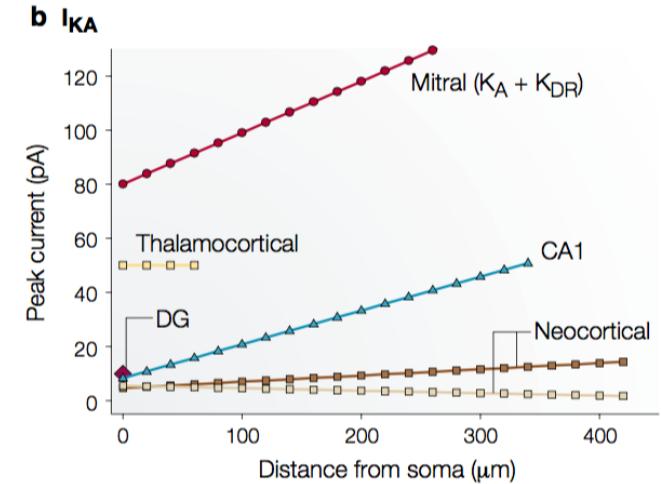
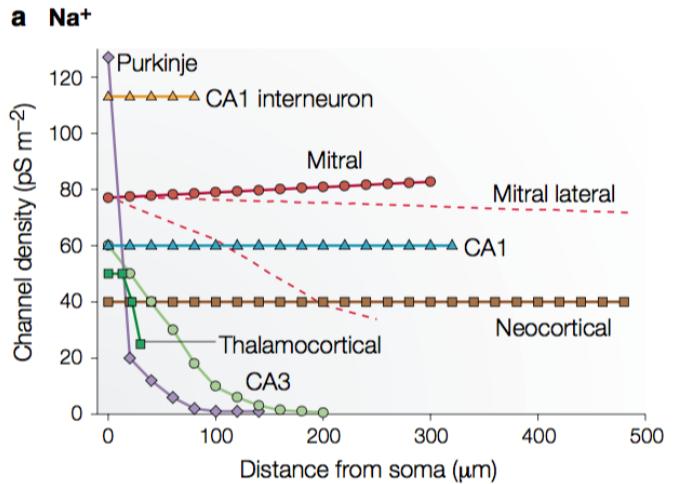
| | Squid Giant Axon | | | Mammalian Neuron | | |
|------------------|------------------|----------------|----------------|------------------|----------------|----------------|
| | Conc. in (mM) | Conc. out (mM) | E_{rev} (mV) | Conc. in (mM) | Conc. out (mM) | E_{rev} (mV) |
| K ⁺ | 400 | 20 | -75 | 140 | 5 | -87 |
| Na ⁺ | 50 | 440 | 55 | 5 - 15 | 145 | 60 - 90 |
| Cl ⁻ | 52 | 560 | -60 | 4 - 30 | 110 | -90 - -60 |
| Ca ²⁺ | 0.0001 | 10 | 145 | 0.0001 | 1 - 2 | 120 - 130 |



Are active mechanisms found in dendrites?

Voltage-gated ion channels are prevalent in neuronal dendrites

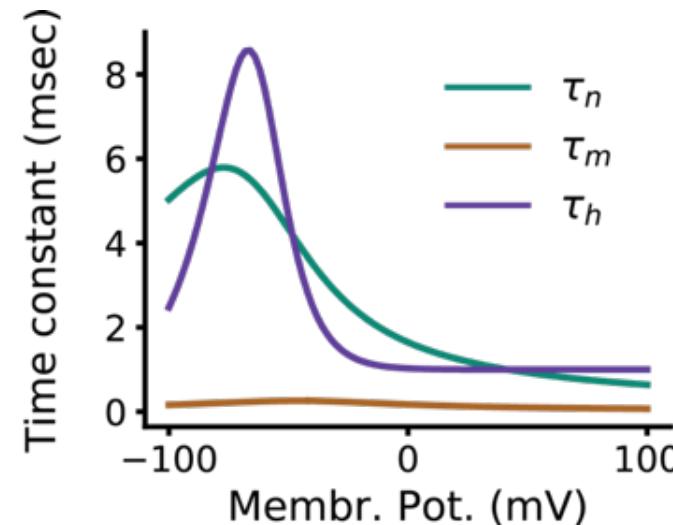
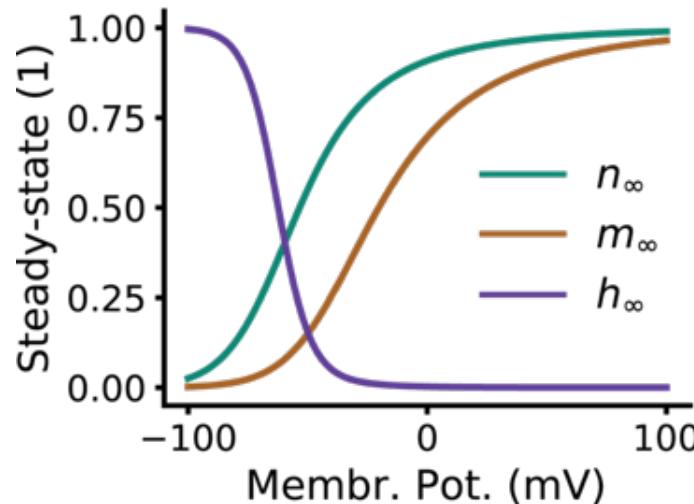
How do they contribute to active processing?



Modelling active conductances

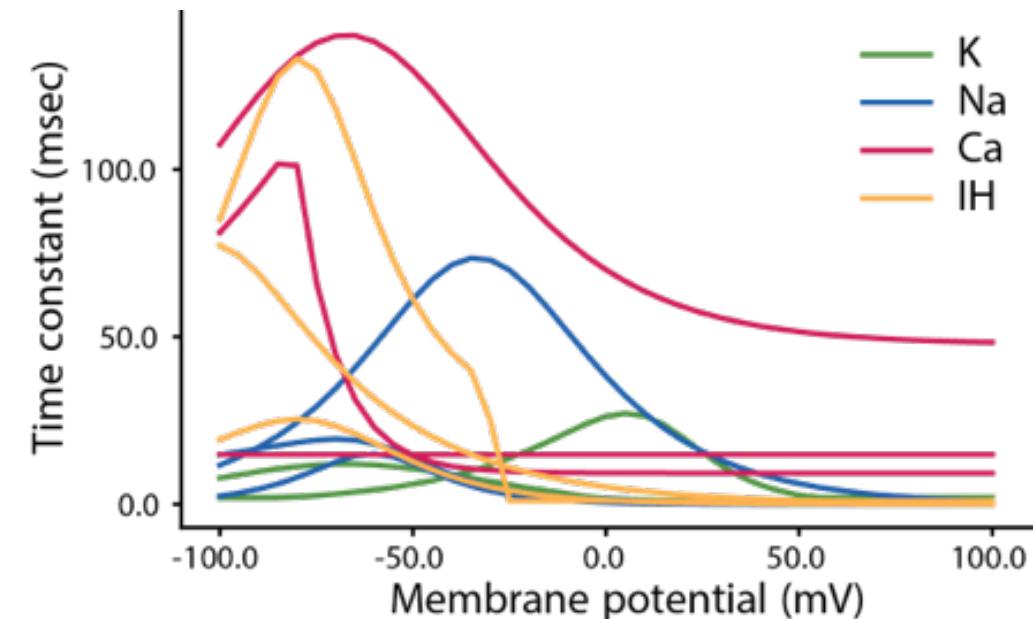
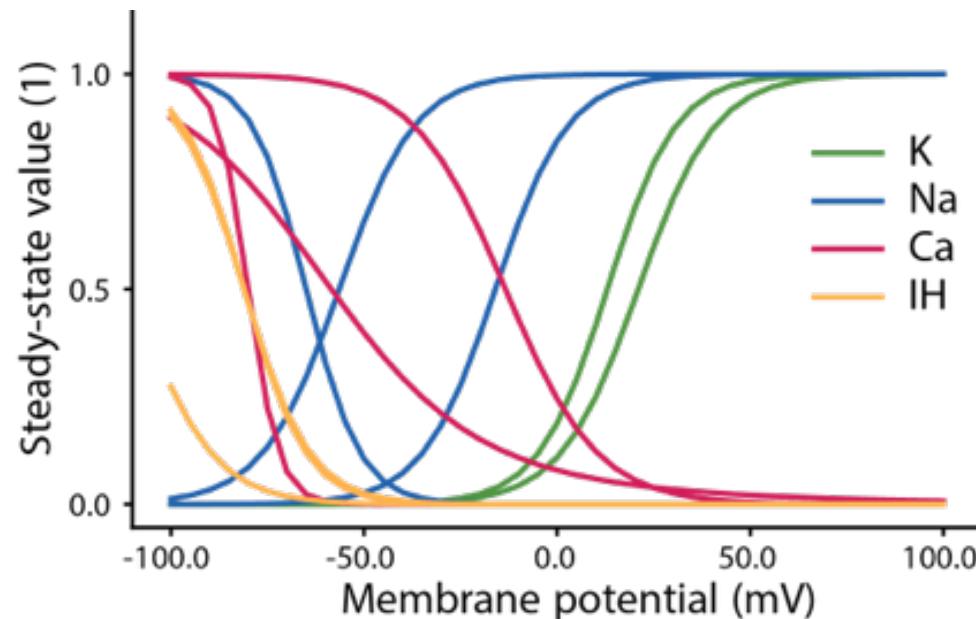
$$C \frac{dV}{dt} = -g_L(V - E_L) - g_{Na} \textcolor{brown}{m}^3 \textcolor{blue}{h}(V - E_{Na}) - g_K \textcolor{teal}{n}^4(V - E_K) + I_{ex}$$

Each gating variable: $\tau_x(V) \frac{dx(t)}{dt} = x_\infty(V) - x(t)$



Modelling active conductances

- What range of voltages does the channel activate (or inactivate)?
- What are the timescales of activation (and inactivation)?
- What is the reversal potential of the ionic species?



Modelling active conductances in NEURON

- Custom ion channel models are made in separate files with the extension “.mod”
- They need to be compiled before the code is run (nrnivmodl / mknrndll)
- They must be inserted to each compartment separately (with separate parameters)

The NEURON block:

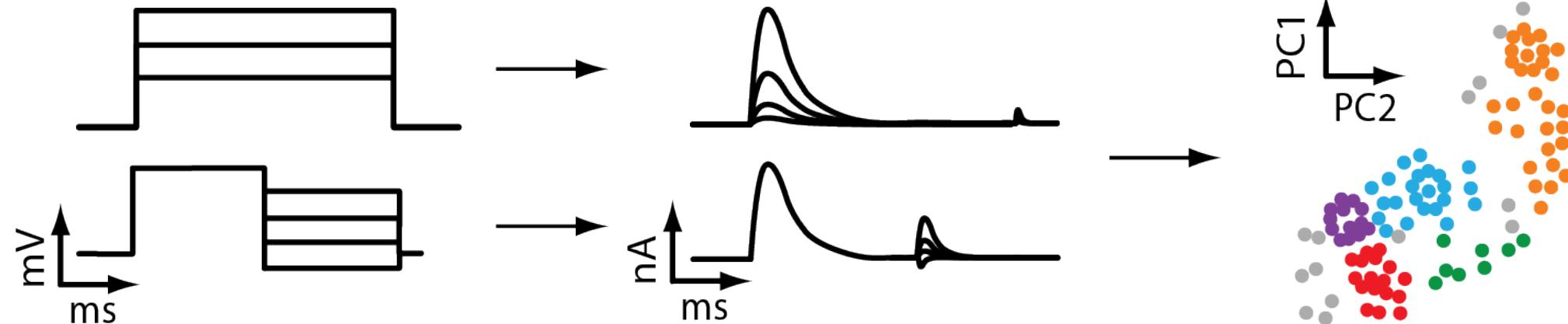
SUFFIX: the name of the mechanism

USEION / NONSPECIFIC_CURRENT statement: specifies which ionic species the model uses

RANGE variables: can be accessed from hoc / python code (e.g., gbar)

ICGenealogy: a resource for biophysical neuron modelling

- Contains nearly 4000 published voltage-gated ion channel models coded in NEURON
- Metadata information: subtype, neuron type, brain area, ...
- Quantitative analysis and comparison of models

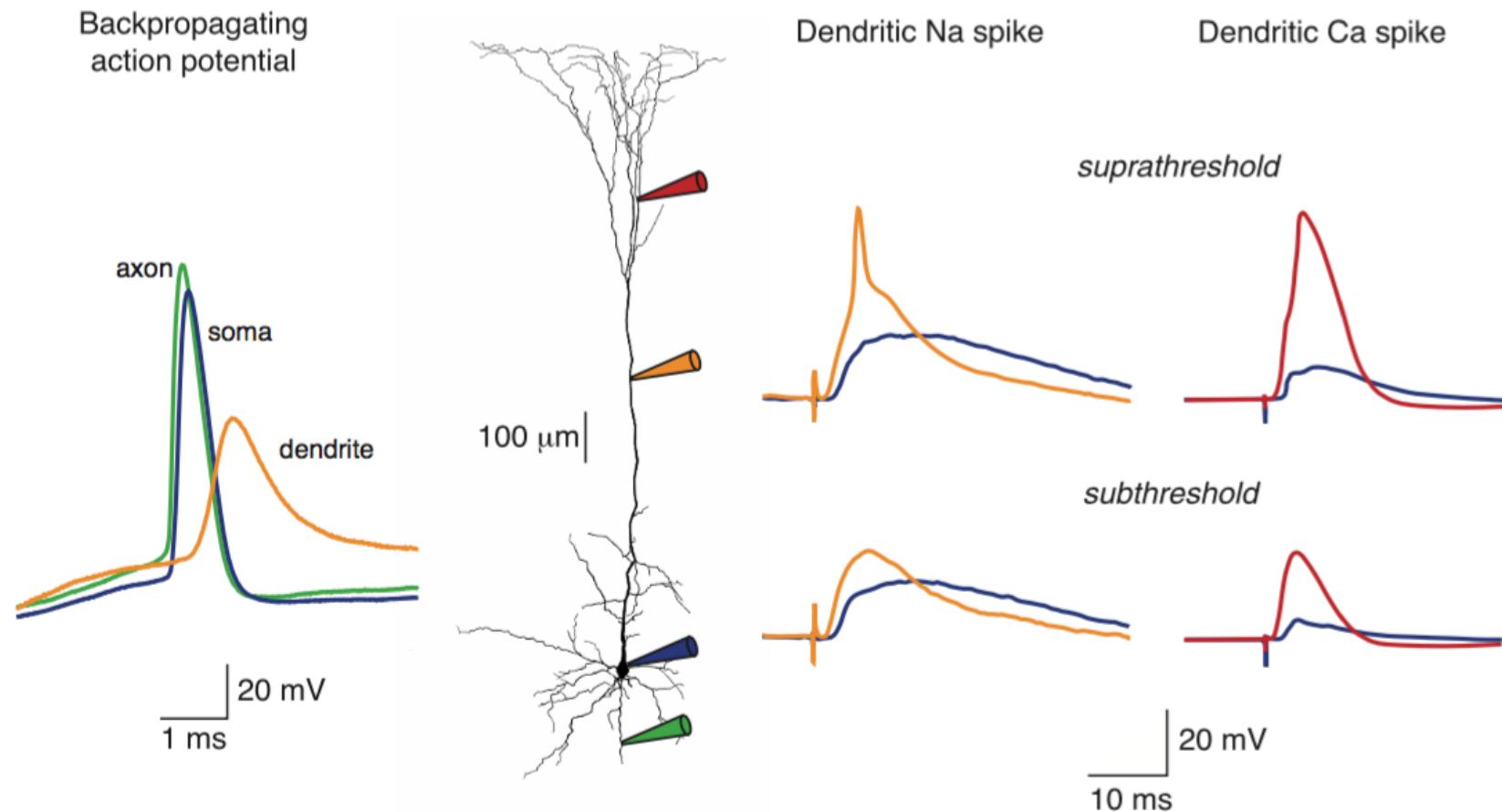


<https://icg.neurotheory.ox.ac.uk/>

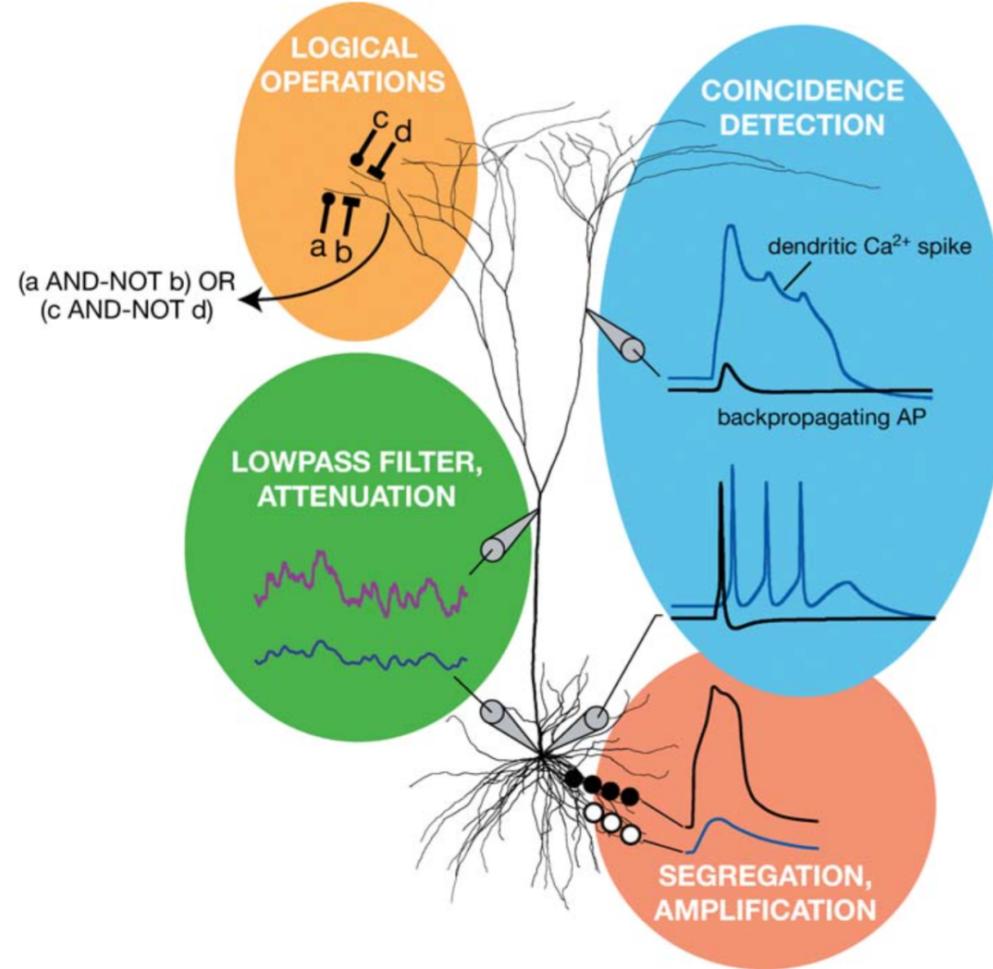
Functional roles of active mechanisms

What active processes occur in dendrites?

What functional roles do they have?



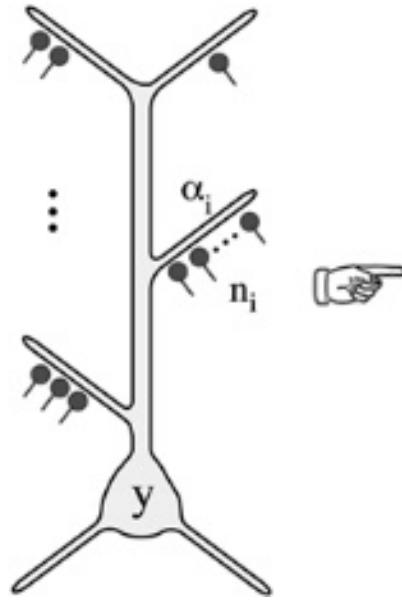
Functional roles of active mechanisms



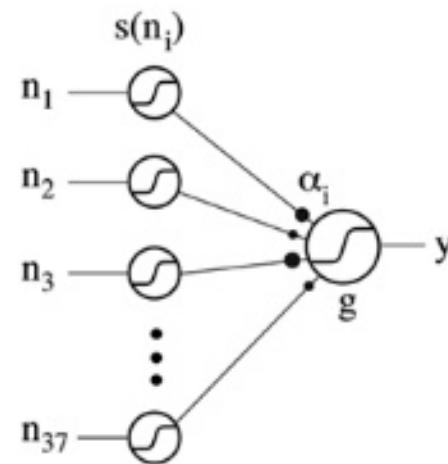
How inputs interact: Active Integration

Jupyter notebook: Ex1c

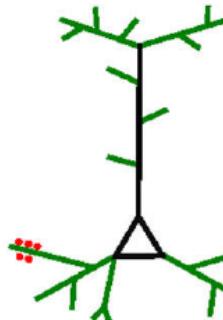
How inputs interact: Spatial Summation



Poirazi et al. 2003



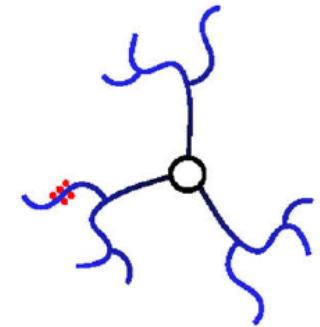
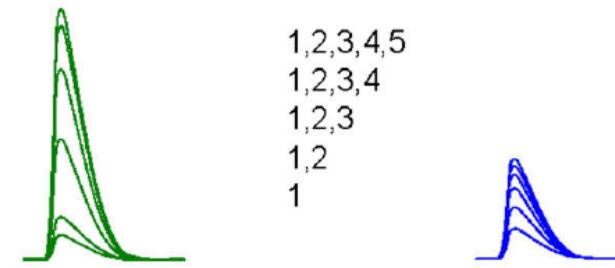
A



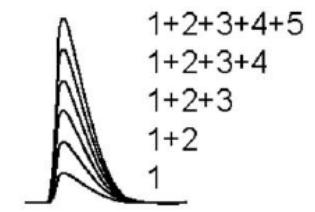
linear compartment
supralinear compartment
sublinear compartment

B

EPSPs measured in response to combined activation (<1ms)



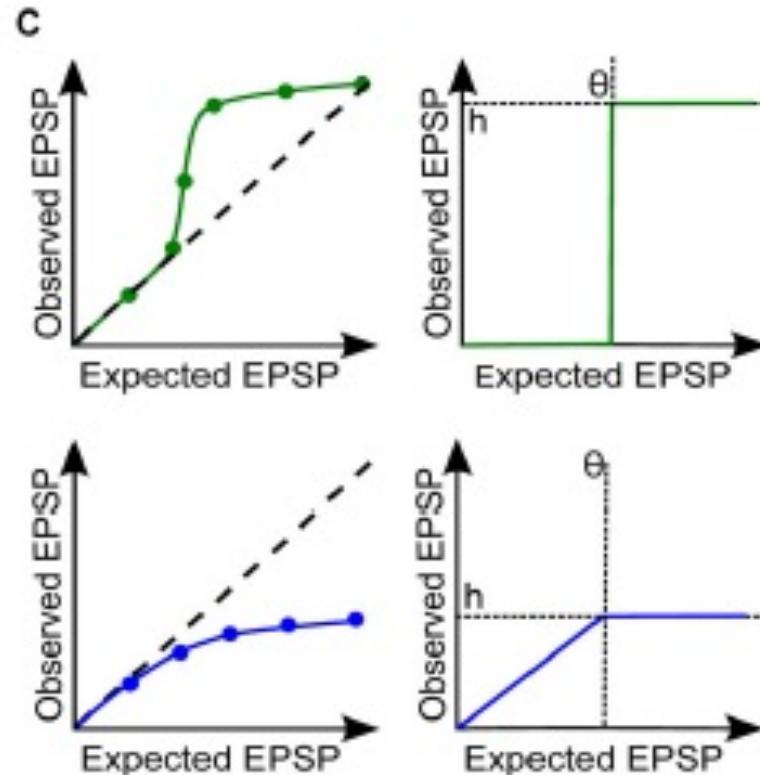
C arithmetic sum of individual EPSPs



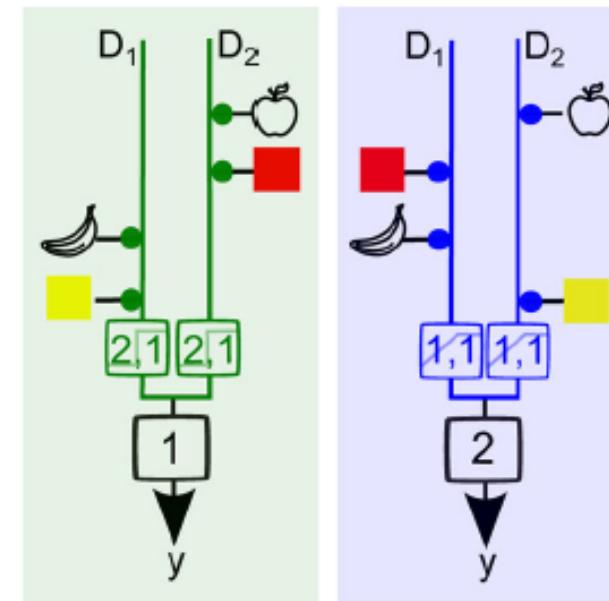
Tran-Van-Minh et al. Front. Cell. Neuroscience, 2015

Which is the best strategy for maximum depolarization – scatter inputs to many dendrites or cluster them into one?

Spatial Summation & Feature Binding problem

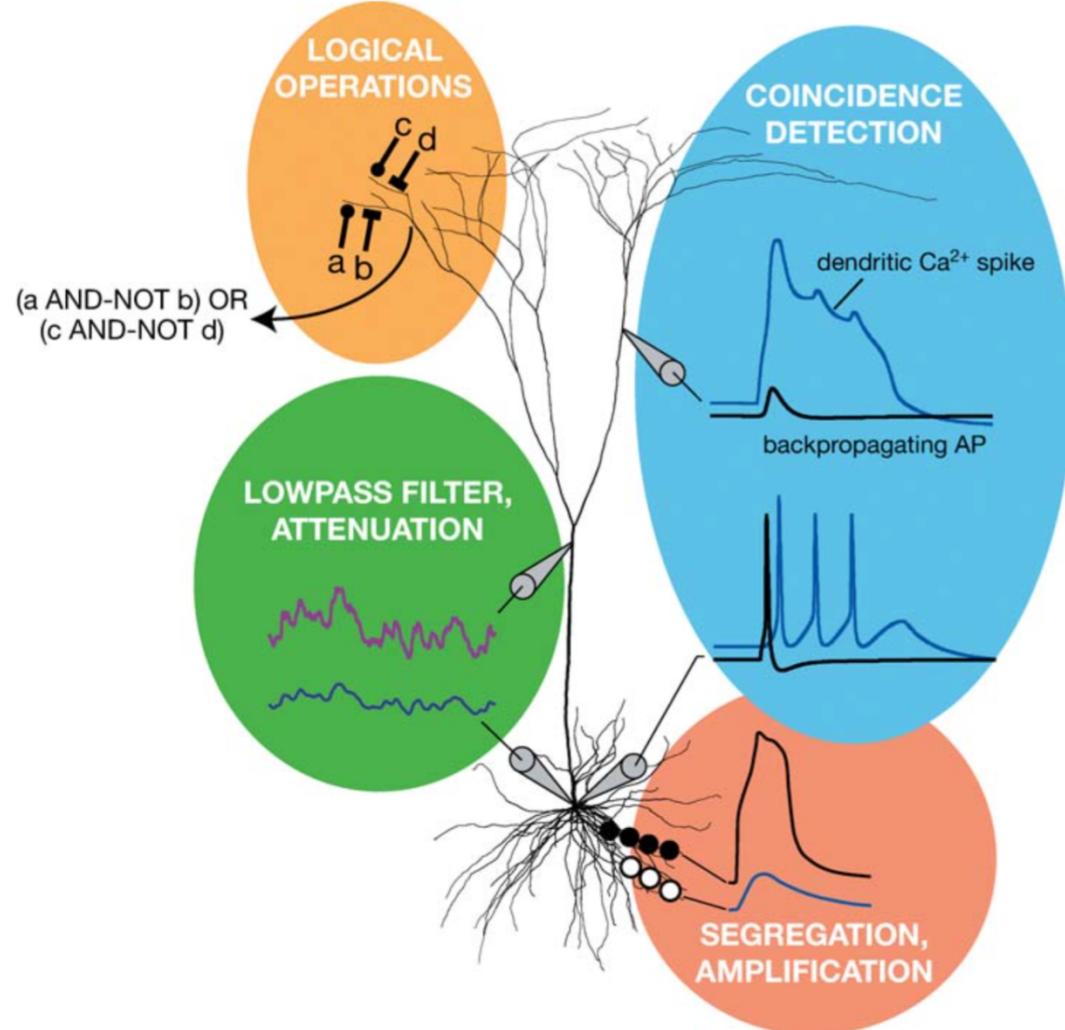


| | apple | red square | banana | yellow square | D ₁ | D ₂ | D ₁ | D ₂ | y |
|--------|-------|------------|--------|---------------|----------------|----------------|----------------|----------------|---|
| apple | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| banana | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |



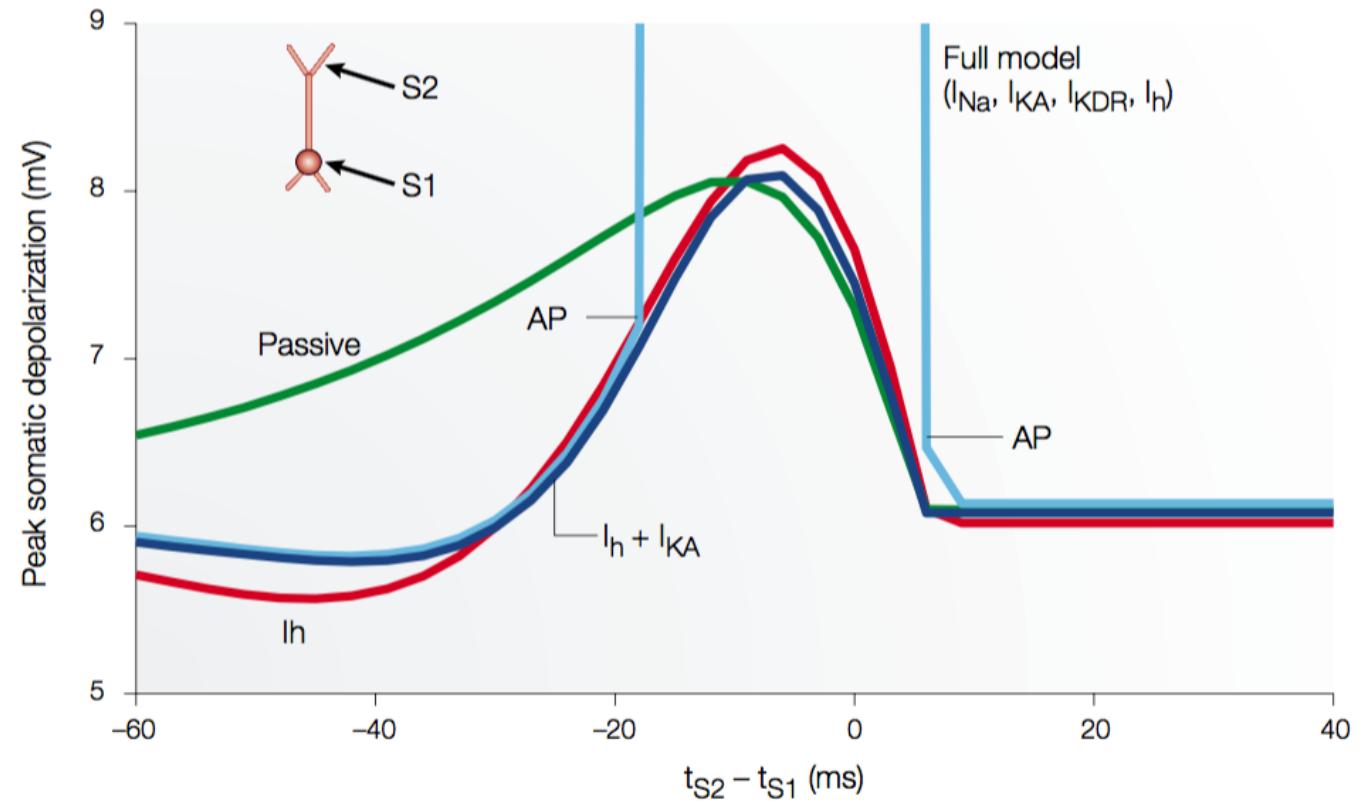
Tran-Van-Minh et al. Front. Cell. Neuroscience, 2015

Active Mechanisms in the Dendrites



Active Mechanisms in the Dendrites

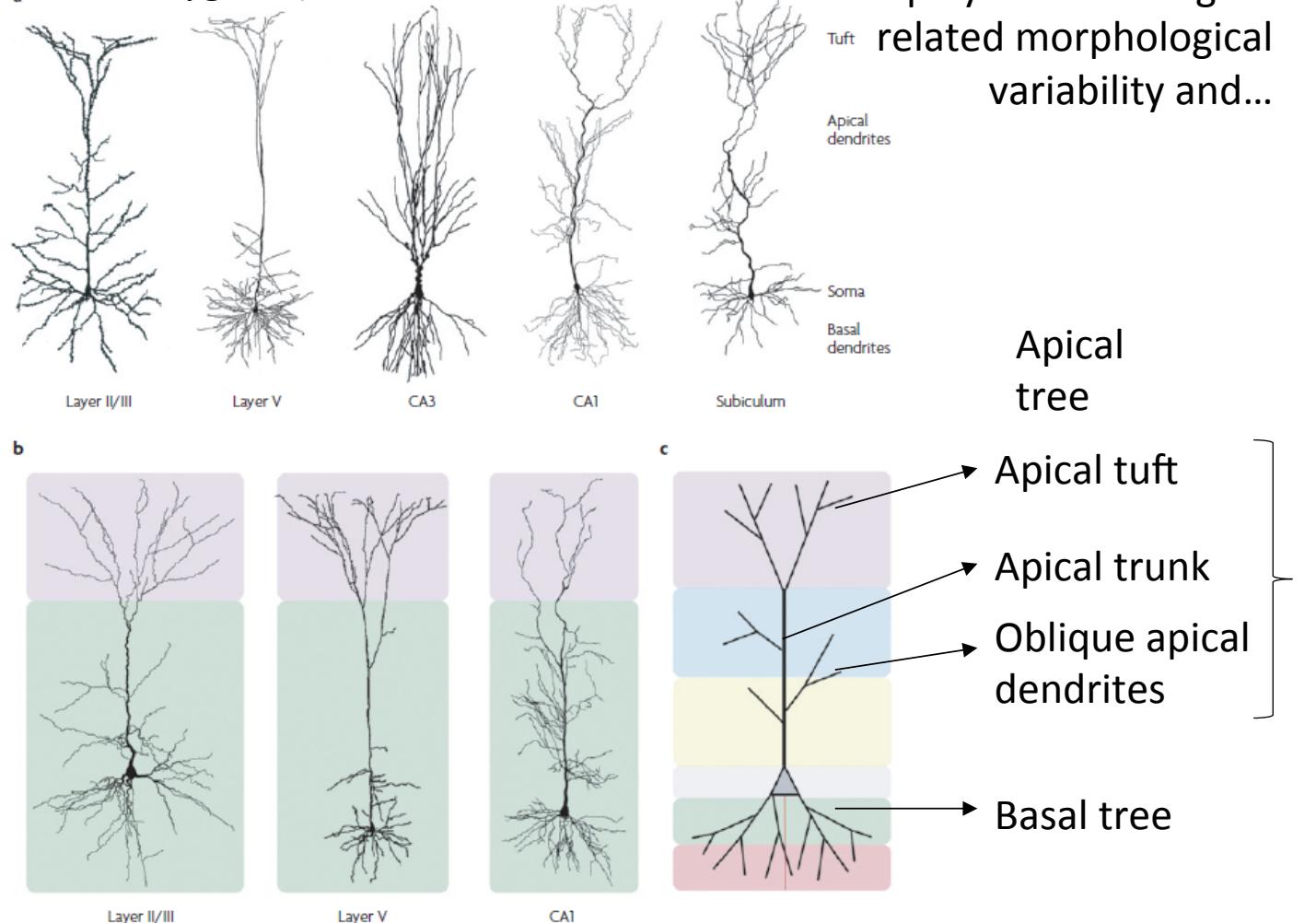
Jupyter notebook:
Exc_2a



Focusing on the cortex: The pyramidal neuron

(also hippocampus and amygdala)

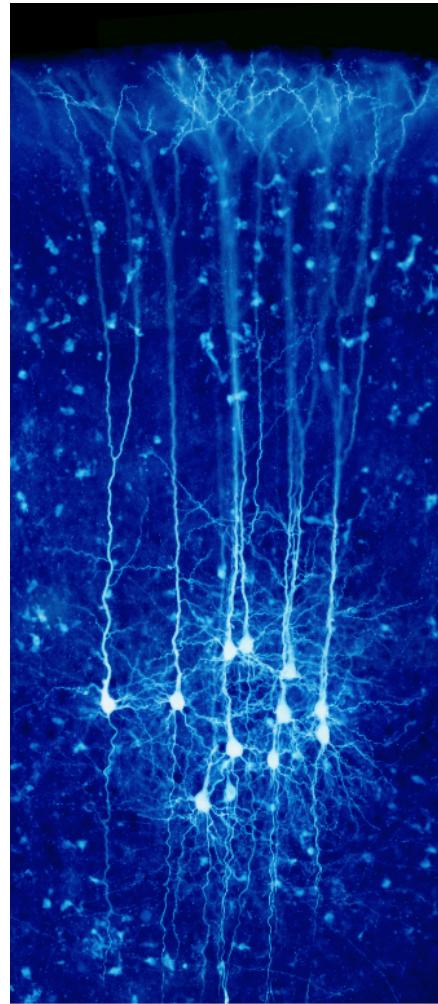
Spruston,
Nature Reviews,
2008



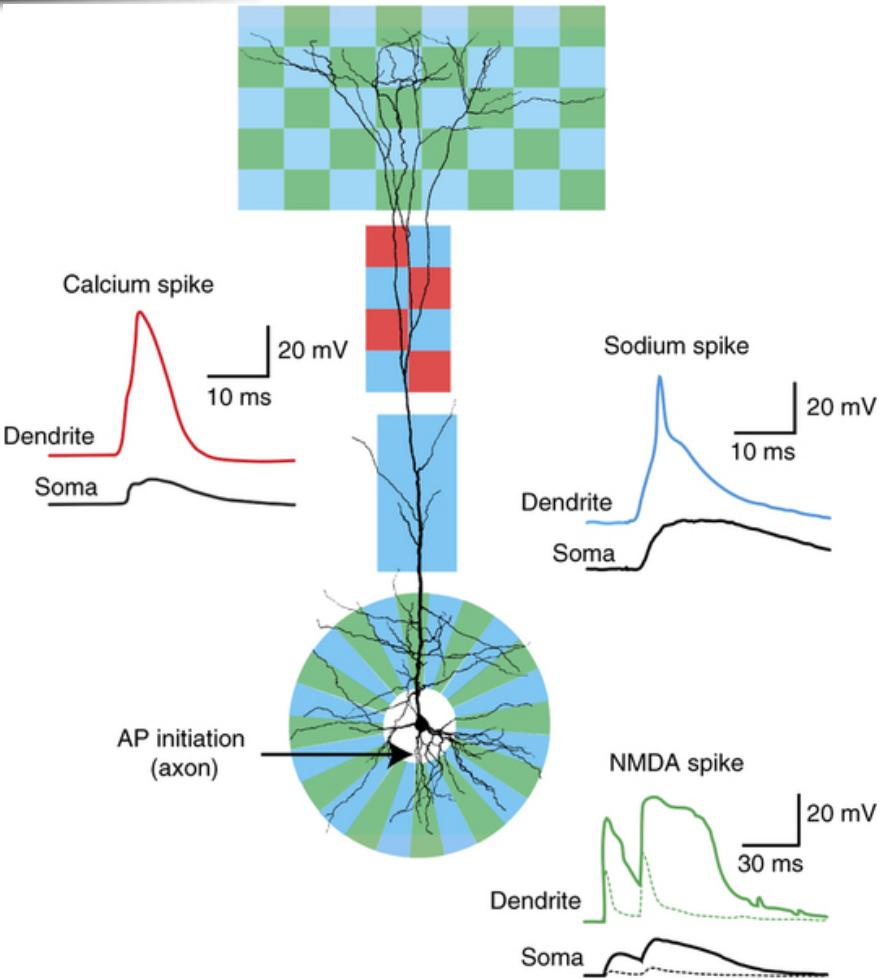
...display area and region
related morphological
variability and...

...dendritic
domains
that receive
unique
synaptic
inputs.

Focusing on the cortex: The pyramidal neuron

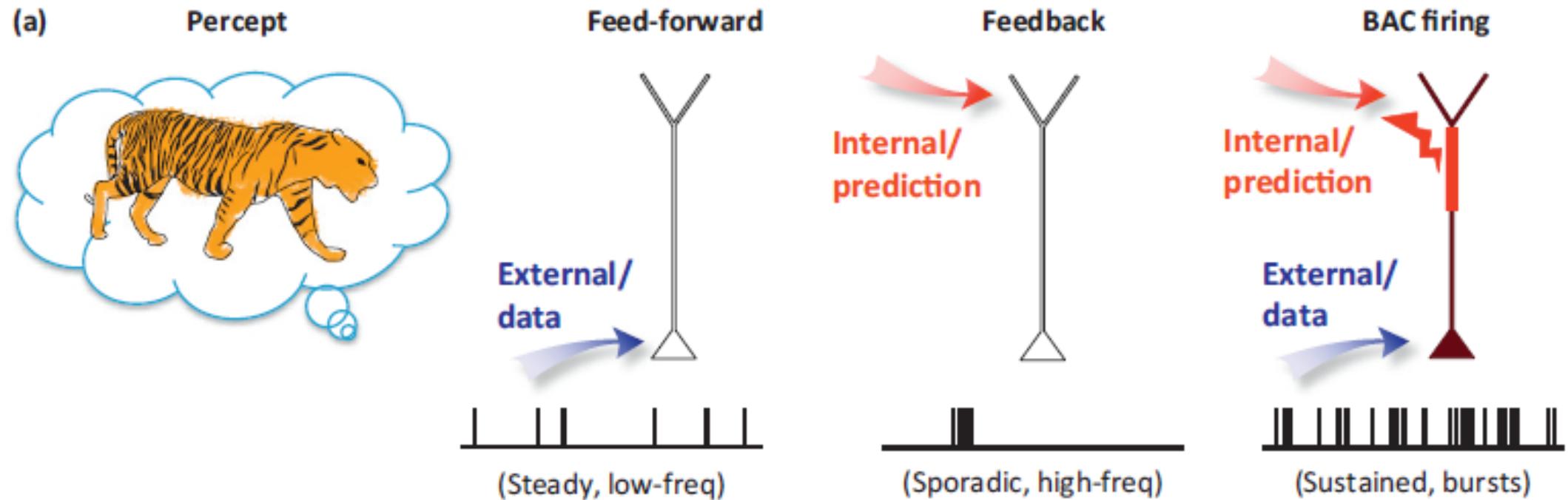


Human brain project



Stuart, Spruston, Nature Reviews, 2015

Functional importance: Coincidence detection



Larkum, TINS 2013

Functional importance: Coincidence detection

Coincidence Detection in Pyramidal Neurons Is Tuned by Their Dendritic Branching Pattern

Andreas T. Schaefer, Matthew E. Larkum, Bert Sakmann and Arnd Roth

J Neurophysiol 89:3143-3154, 2003. First published 26 February 2003;
doi: 10.1152/jn.00046.2003

Functional importance: Coincidence detection

Jupyter notebook: Ex3 – BAC model

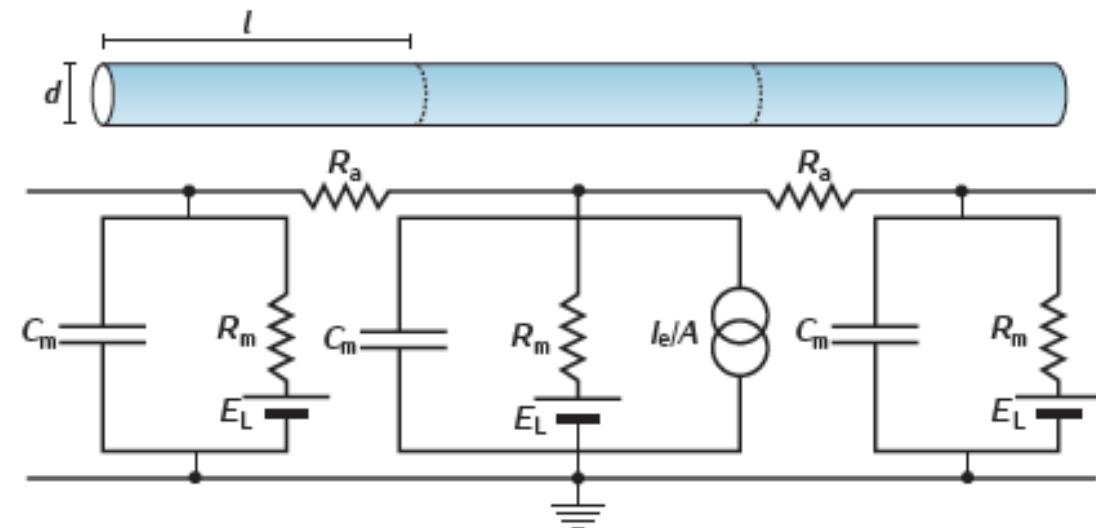
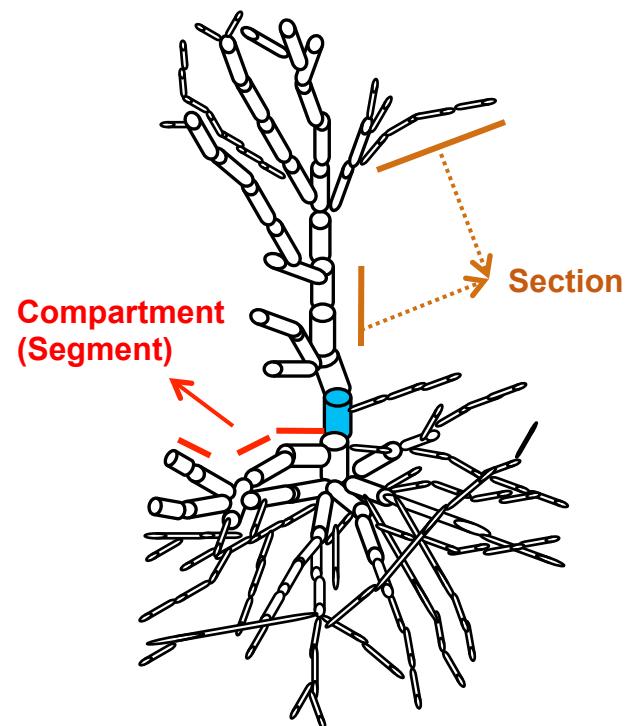
More active dendrites: gating plasticity

Jupyter notebook: Exc_2b

Thank you!

One issue to consider... The spatial grid

- Computing the values of spatiotemporally continuous variables over a set of discrete points in space (a "grid" of "nodes") for a finite number of instants in time.
- Although the time scale of biophysical processes may suggest a natural Dt , it is usually not clear how fine the spatial grid should be.



One issue to consider... The spatial grid

- The d_lamda rule
 $\bar{\lambda} = \sqrt{2dR\lambda_m / R\lambda_a}$

$$\lambda \downarrow f \approx 1/2 \sqrt{d / (\pi f R \downarrow a C \downarrow m)}$$

Example:

diameter=0.5μm, $R_m = 50.000 \Omega \text{ cm}^2$, $R_i = 100 \Omega \text{ cm}$, and $C_m = 1 \mu\text{f/cm}^2$

$$\lambda_{dc} = 1500 \mu\text{m}$$

$$\lambda_f = 200 \mu\text{m}$$

Maximum distance between two grid points=

$$d_{\text{lambda}} * \lambda_{AC}, d_{\text{lambda}} = 0.1$$

