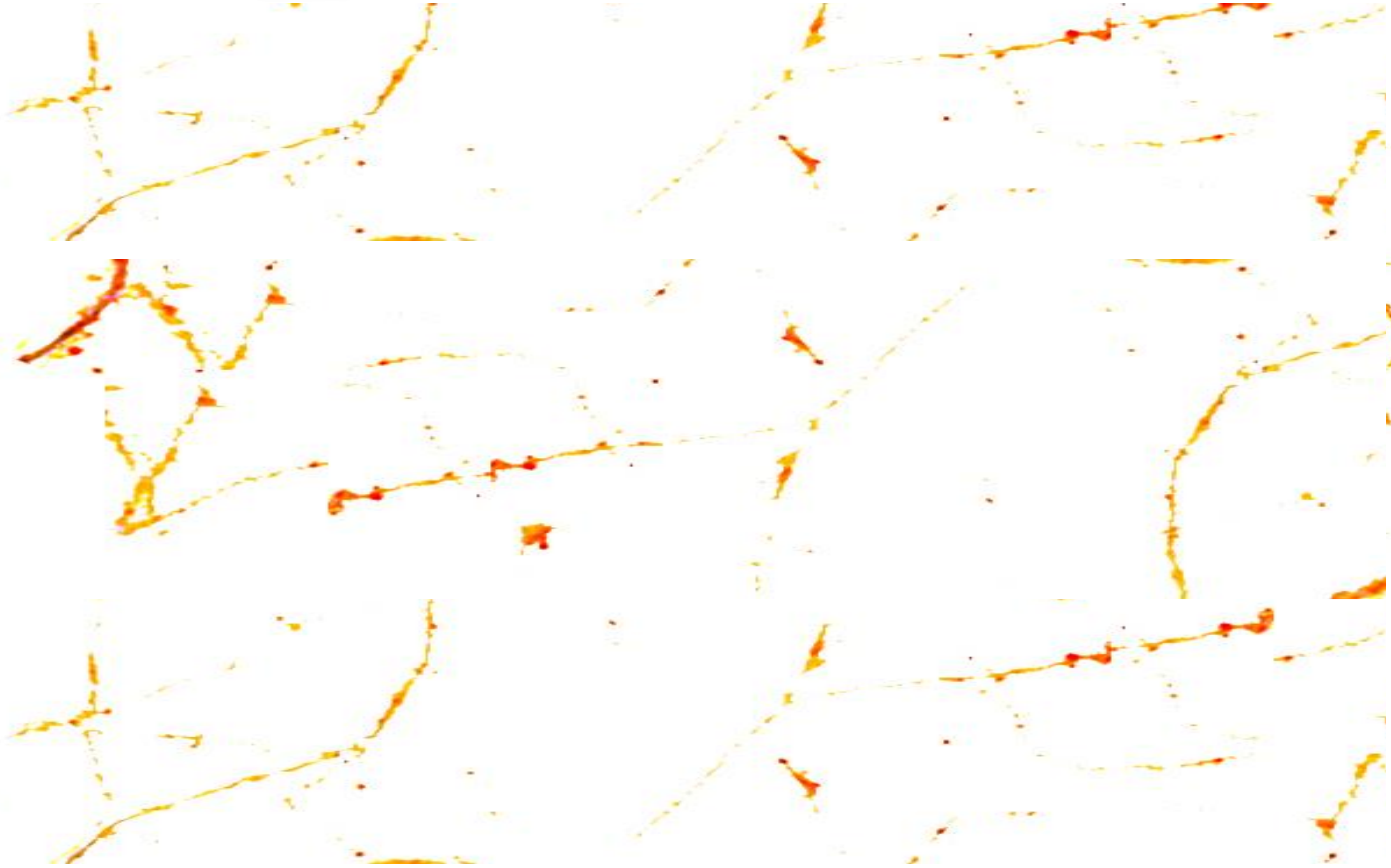
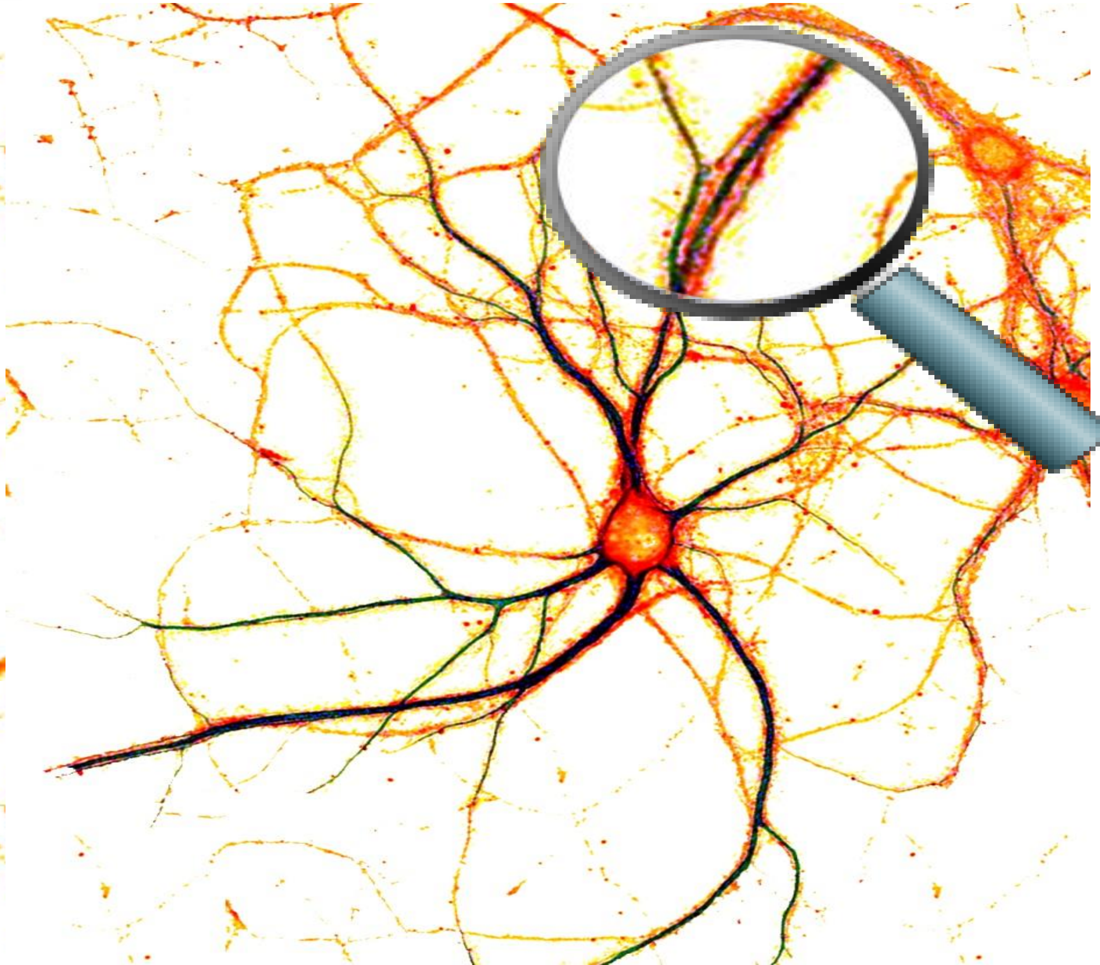
Dendritic computations

Spyridon Chavlis, Michalis Pagkalos, Institute of Molecular Biology and Biotechnology (IMBB), Foundation for Research and Technology-Hellas (FORTH)

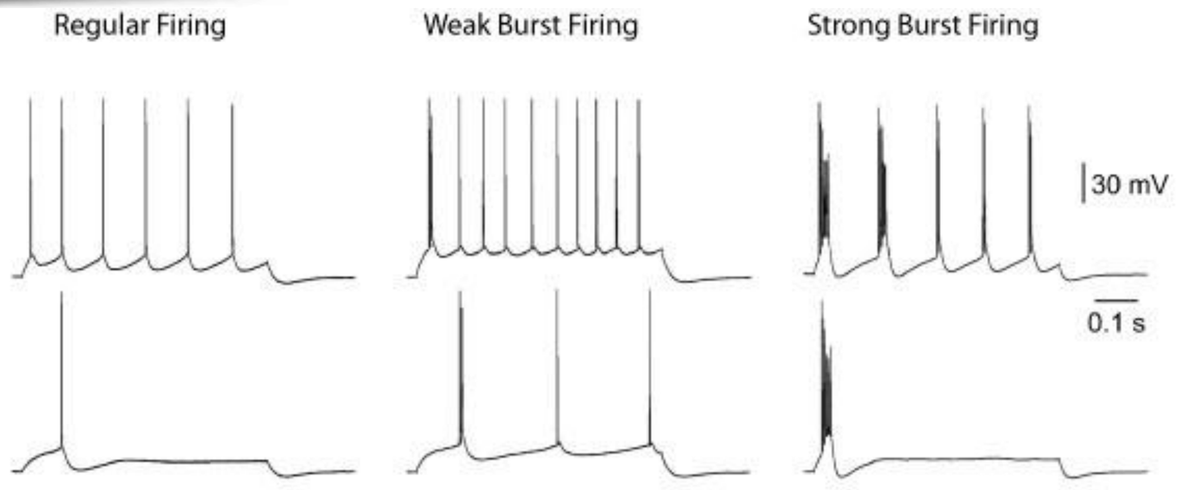




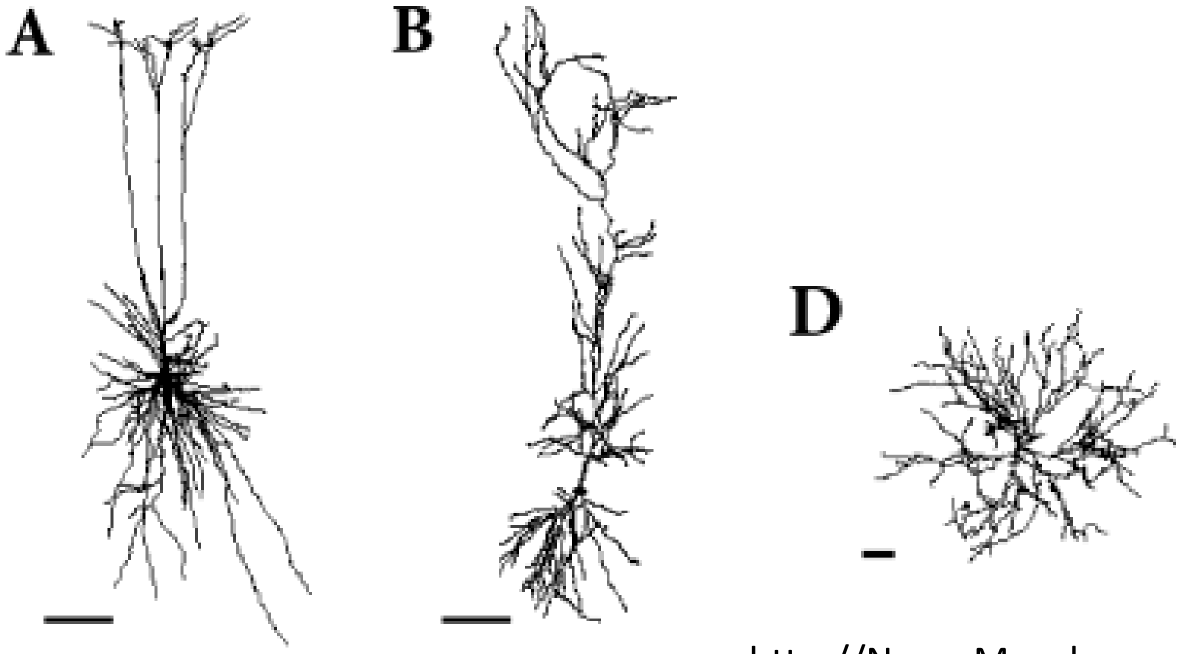
Neurons are (also) characterized by their dendritic tree

Neuron Types:

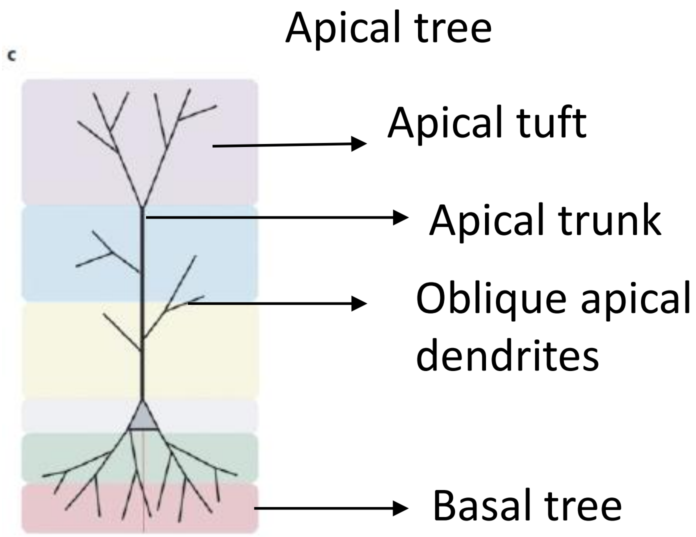
1. Classification based on Excitatory or Inhibitory neurotransmission.
2. Classification using gene expression
3. Classification using spiking activity
4. Classification by anatomical features



Williams, Stuart, J Physiol. 1999

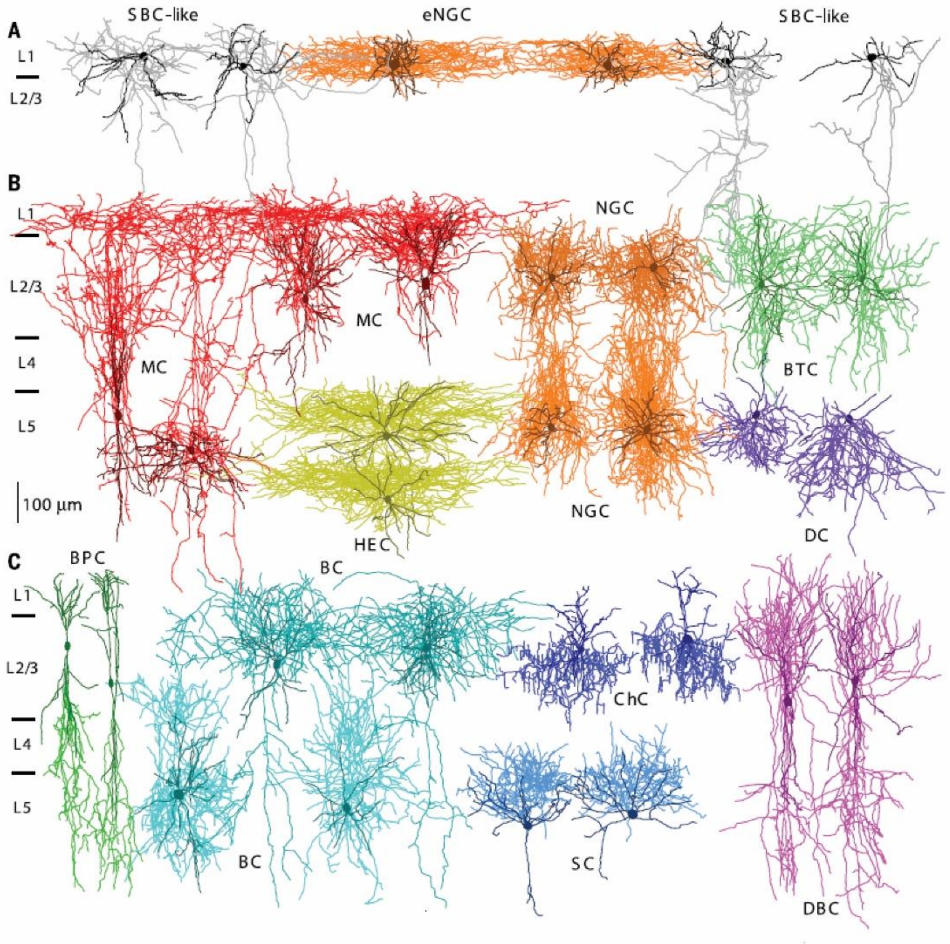


http://NeuroMorpho.org

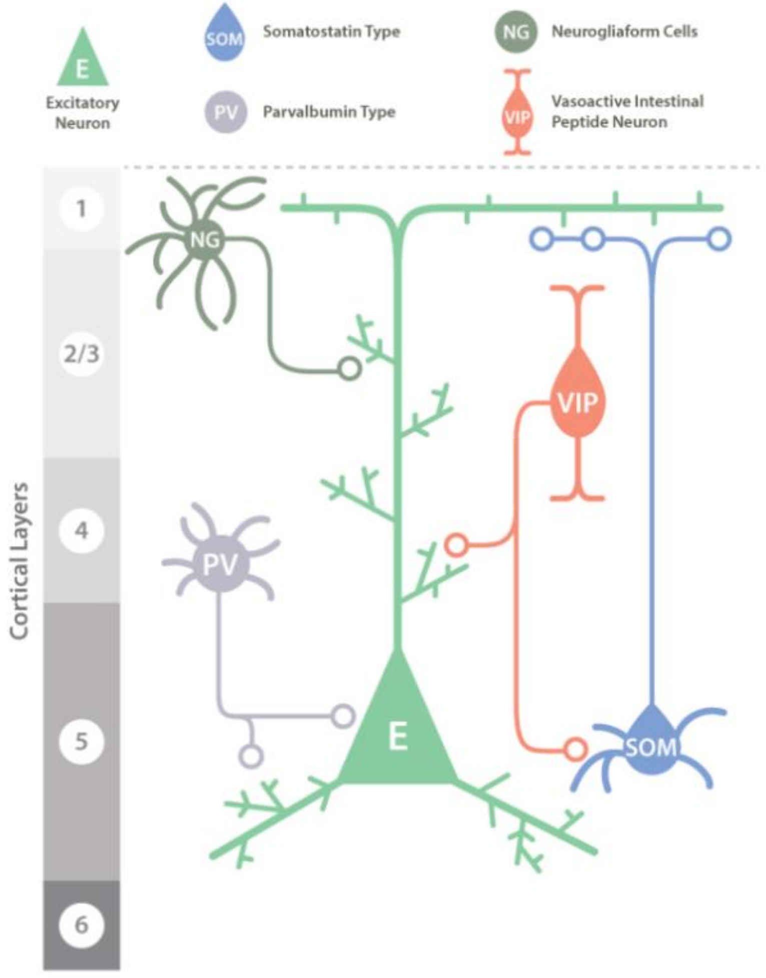


Spruston, Nature Neuro. Rev. 2008

Inhibitory neurons are (also) characterized by their dendritic tree



Jiang et al., Science 2015



Why model the details?

1. Suggest possible computational (functional) role for the modeled system (**predictions**).

2. Interpretation of **experimental** results - Gain insights into key biophysical parameters.

3. I**nspire** implementations in other fields.

Where can you find the details?

- Find Neuronal Morphologies in NeuroMorpho

http://neuromorpho.org/

- Find neuron's electrophysiological properties

http://neuroelectro.org/

- Allen Institute

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

- Blue Brain Portal

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

- Find models in ModelDB

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

- Choose your channels using **I**on**C**hannel**G**enealogy

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

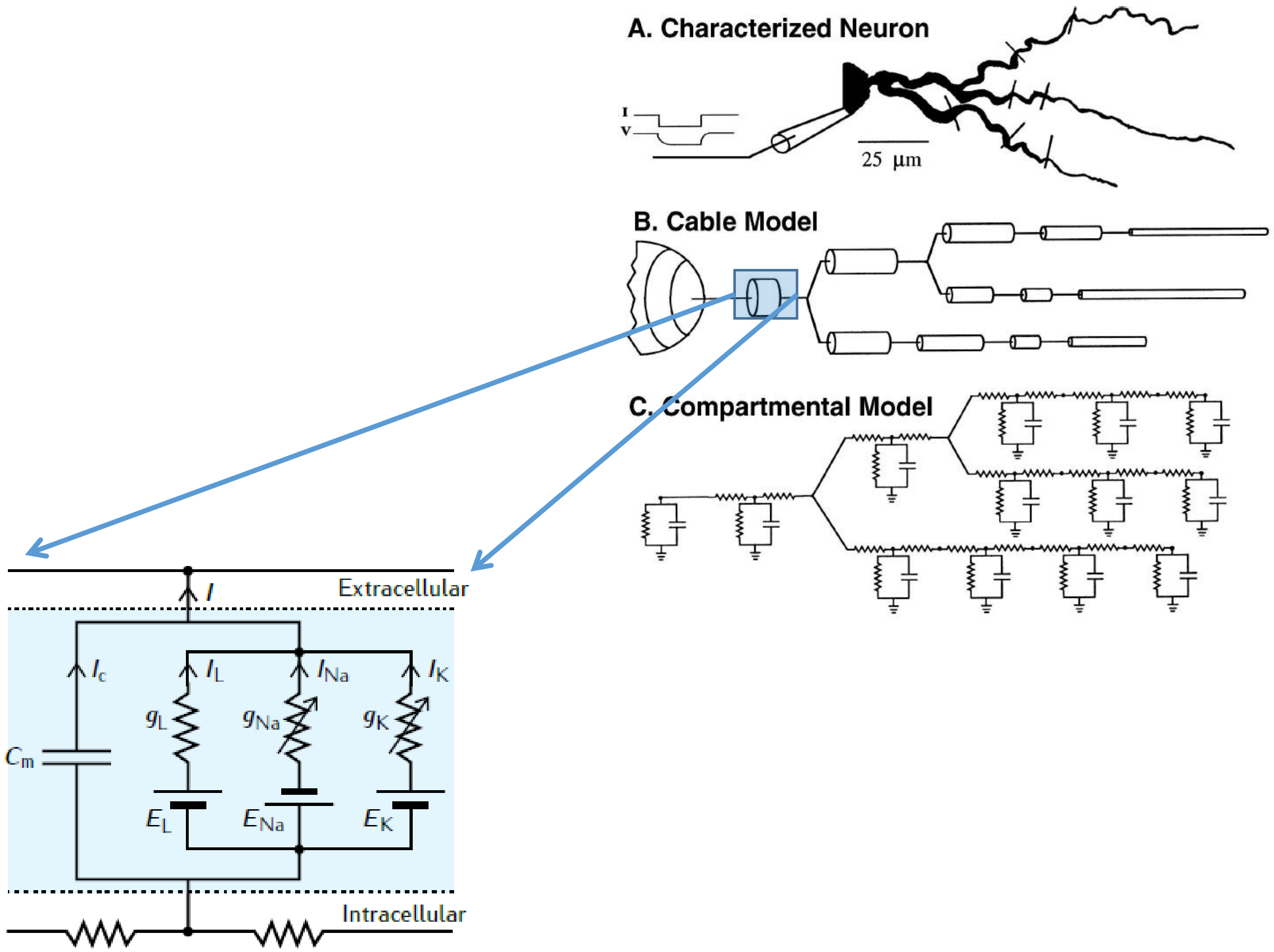
- Find Access to HPC (mostly for Parallel simulations) in…

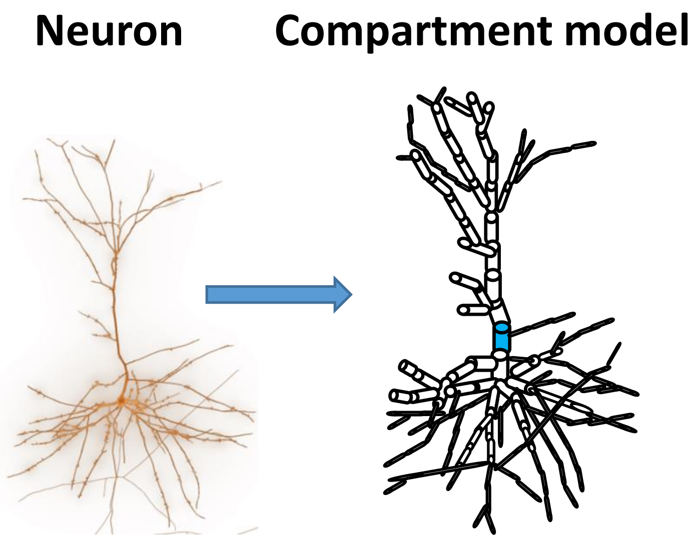
https://www.nsgportal.org/overview.html

Cable theory

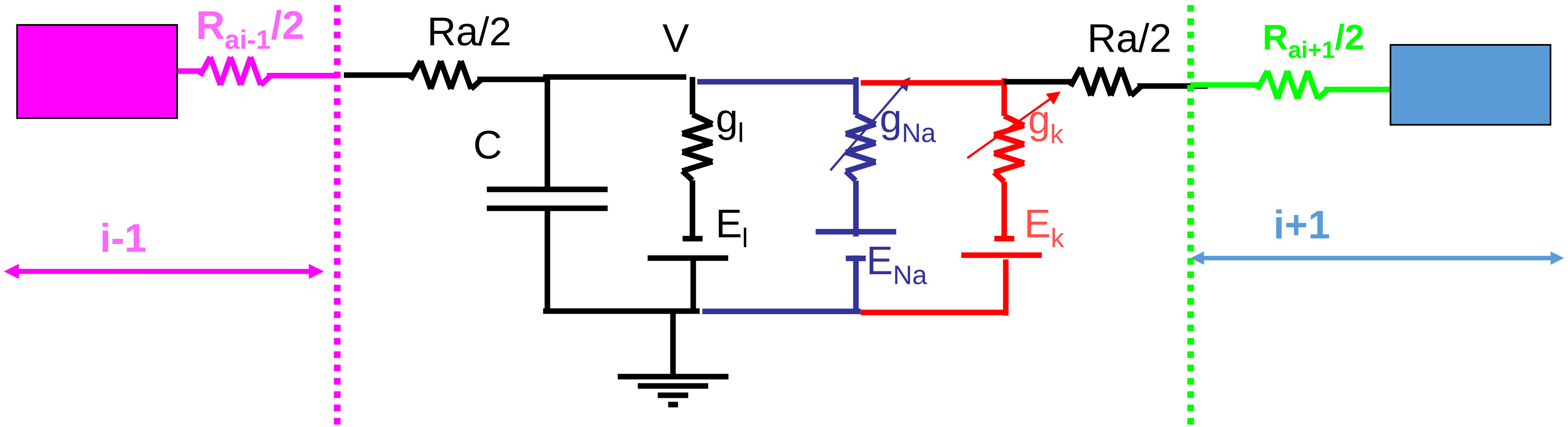
Cable theory is concerned with ***how inputs propagate*** *to the soma* or the axon initial segment, ***how these inputs interact*** with one another, and how the placement of an input on a dendritic tree affects its ***functional importance*** to the neuron.

Modeling dendrites: Compartmental Modeling



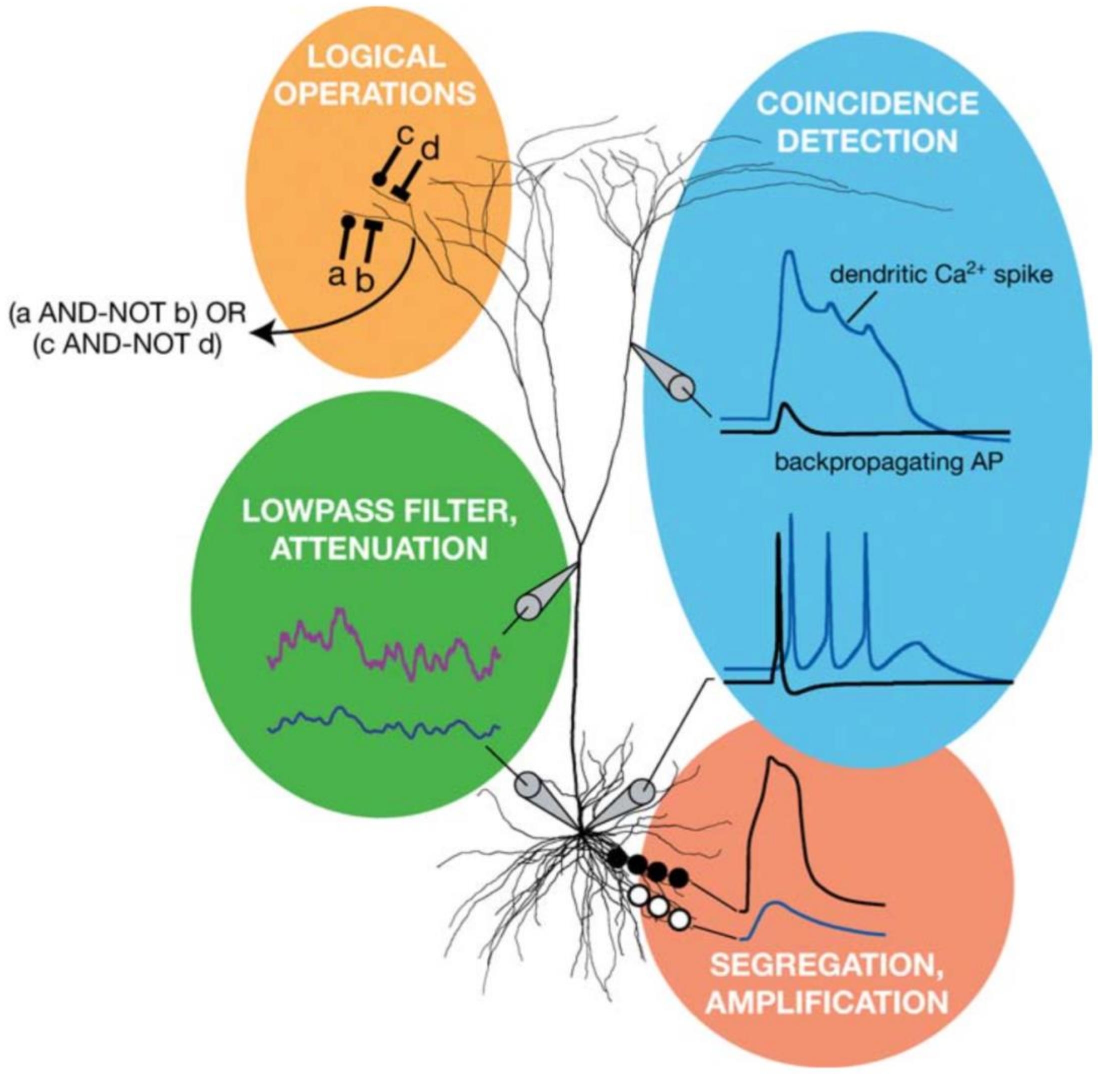


Compartmental model



$$\frac{CdV}{dt} = g_l(E_l - V) + g_{Na}(E_{Na} - V) + g_K(E_K - V) + 2 * \frac{V_{i+1} - V_i}{R_{ai+1} + R_a} + 2 * \frac{V_{i-1} - V_i}{R_{ai-1} + R_a} + I_{inj}$$

What can dendrites do?



London & Hausser 2005 Ann Rev Neuro

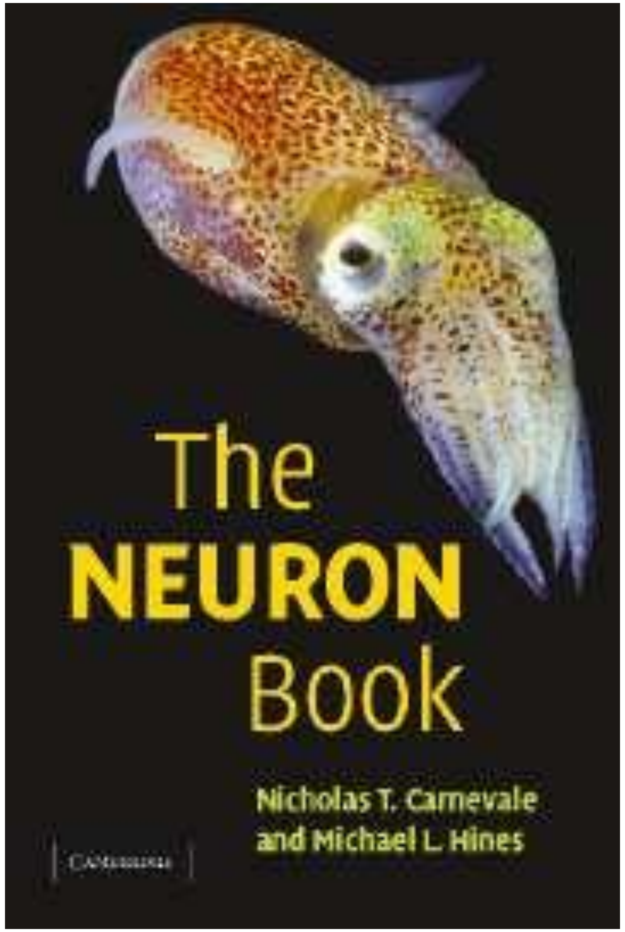
The NEURON simulation environment

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

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.

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
syn.tau=10 #ms
```

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])
```

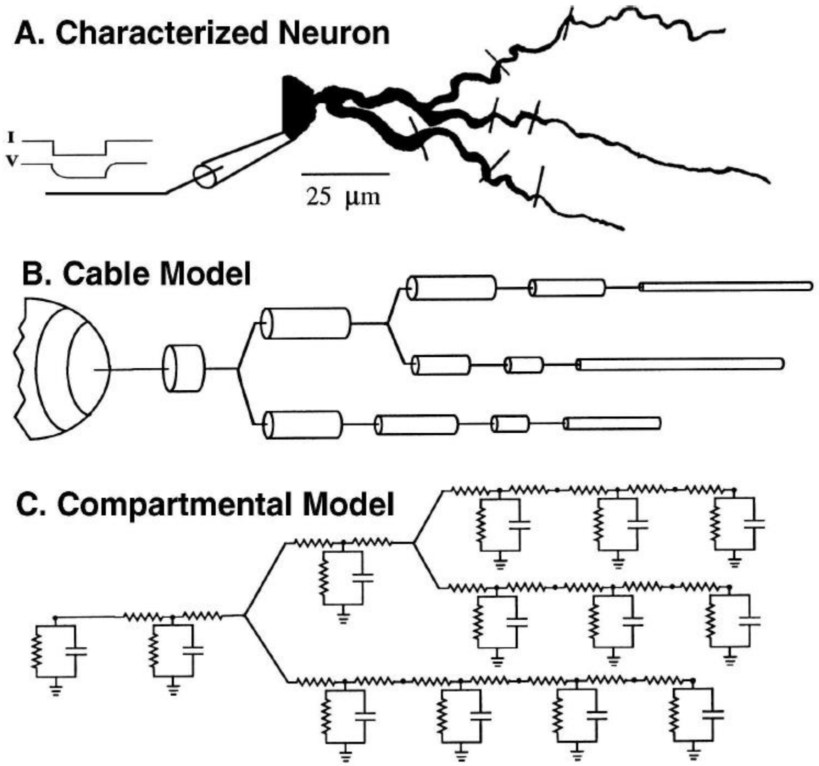
Cable theory

- Connecting compartments

```
child.connect(parent, [0 or 1])
```

e.g.

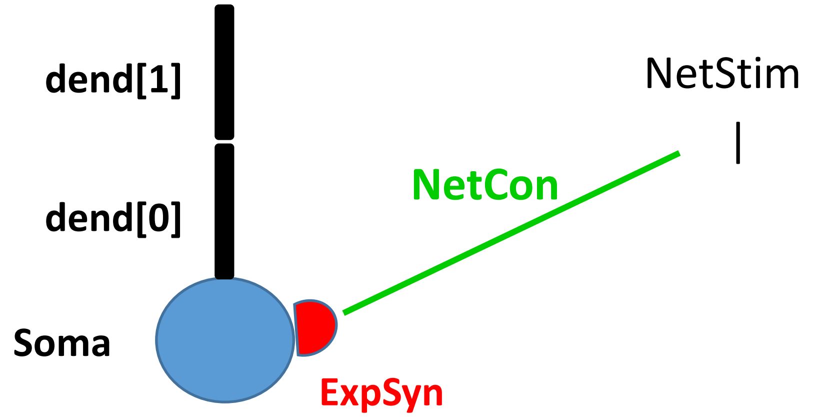
```
dend0.connect(soma(0),0)
```



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.

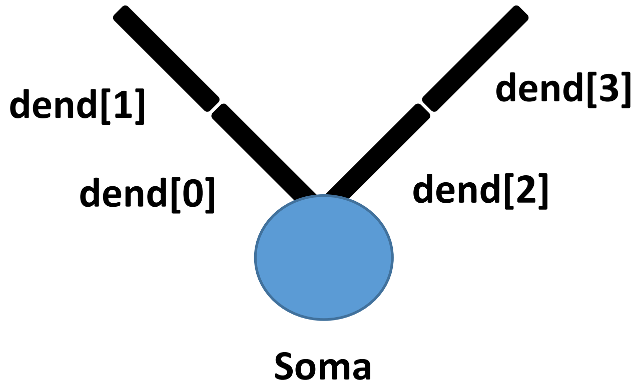
How inputs propagate / interact

Colab notebook – part 1



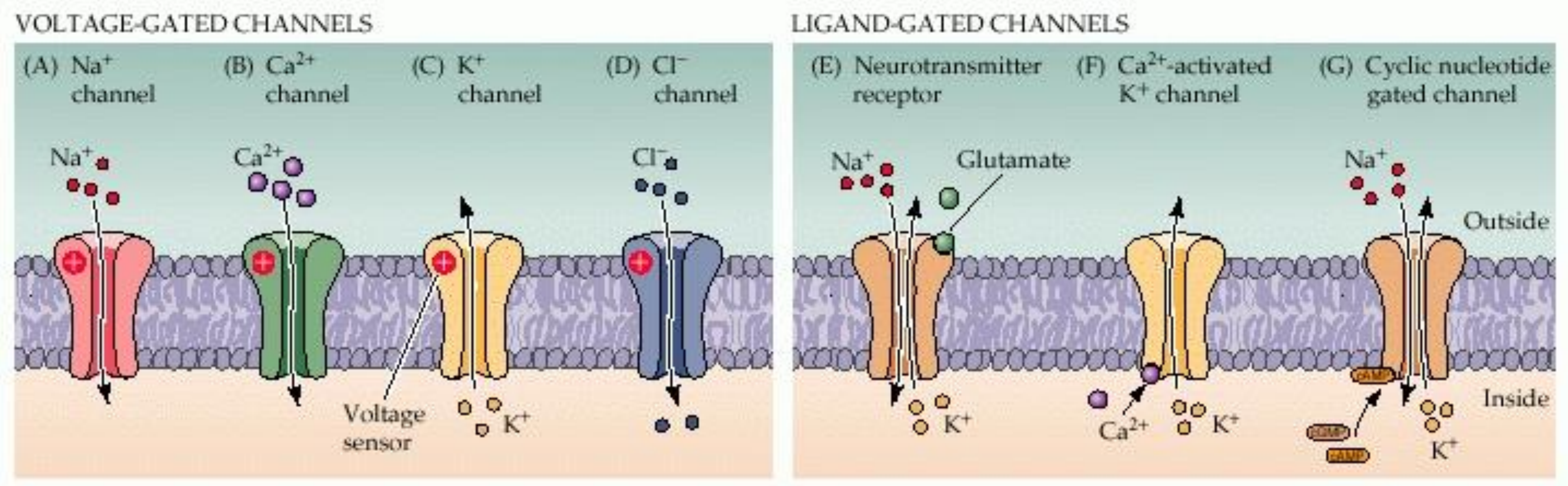
How inputs interact: Segregation

Colab notebook – part 2



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)

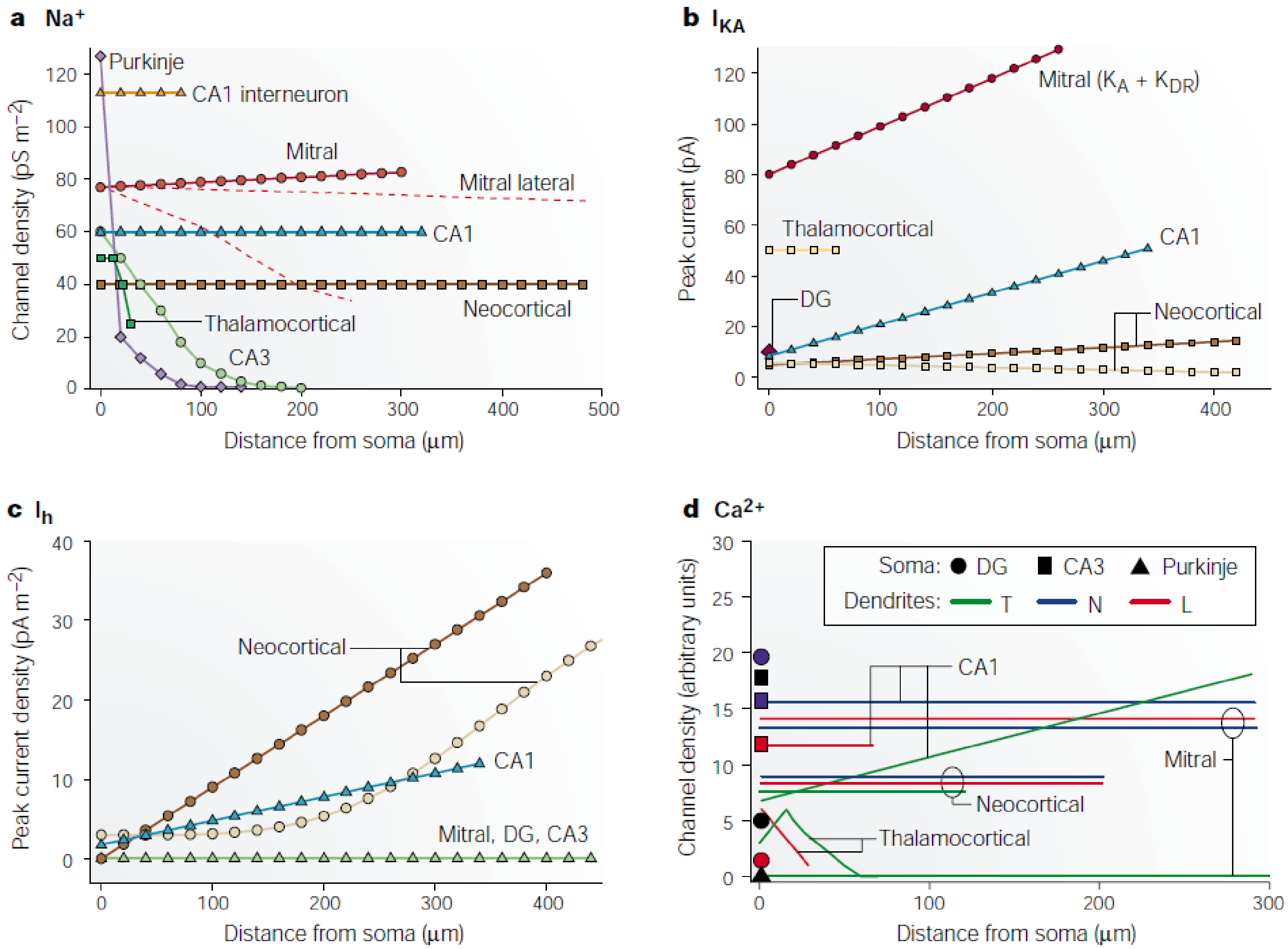


Neuroscience, 2nd edition

Are active mechanisms found in dendrites?

Voltage-gated ion channels are prevalent in neuronal dendrites

How do they contribute to active processing?



Migliore & Shepherd 2002

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

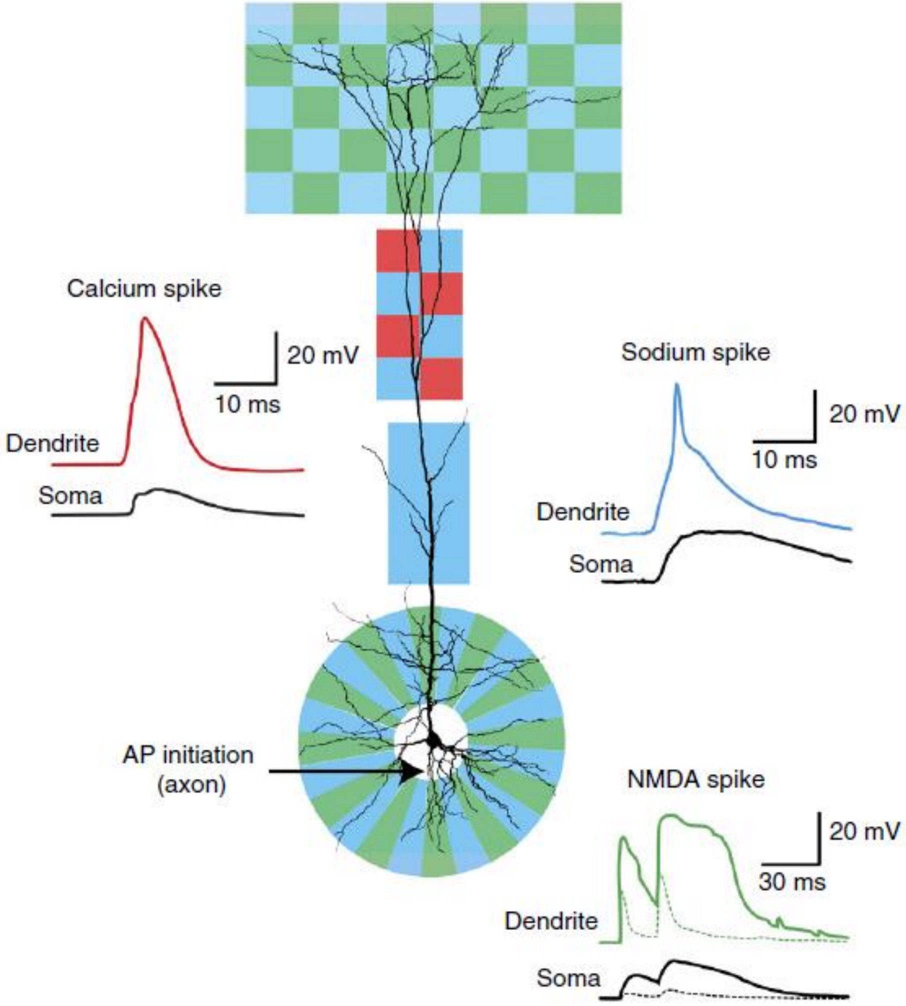
USEION / NONSPECIFIC_CURRENT statement: specifies which ionic species the model uses

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

Functional roles of active mechanisms

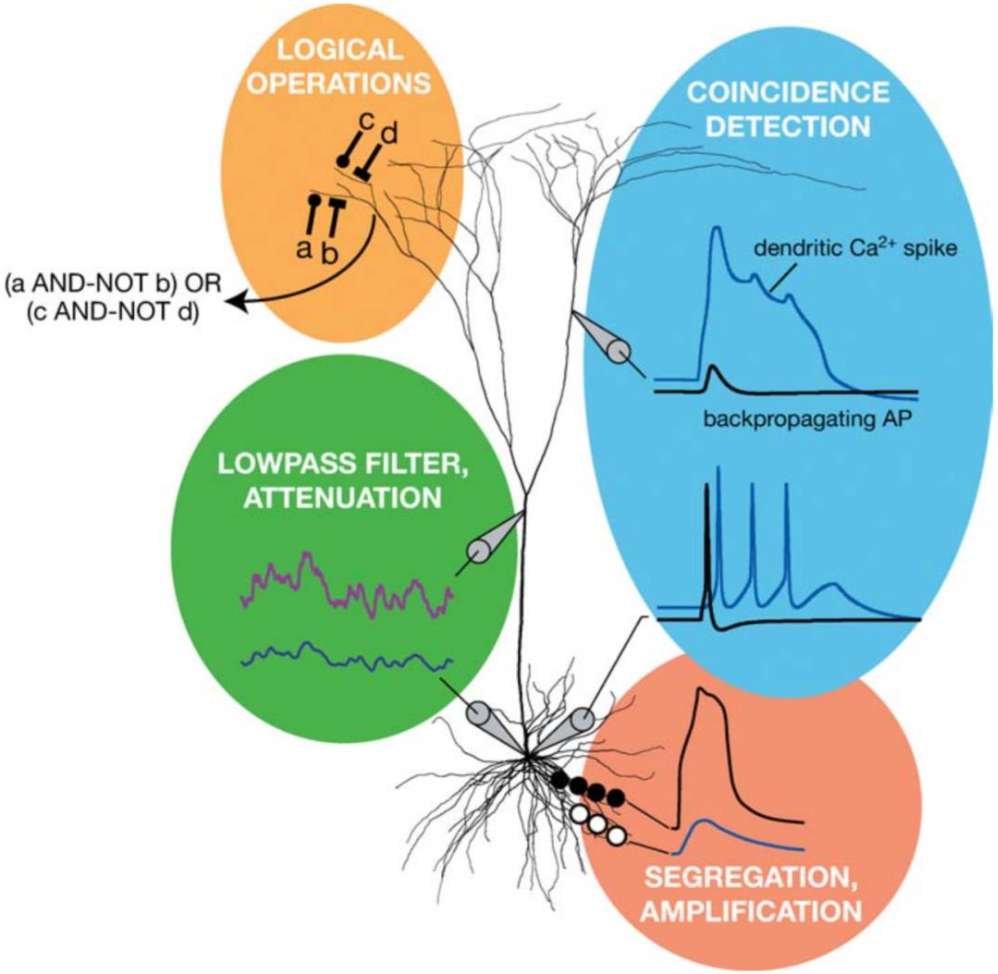
What active processes occur in dendrites?

What functional roles do they have?



Stuart, Spruston, Nature Reviews, 2015

Functional roles of active mechanisms

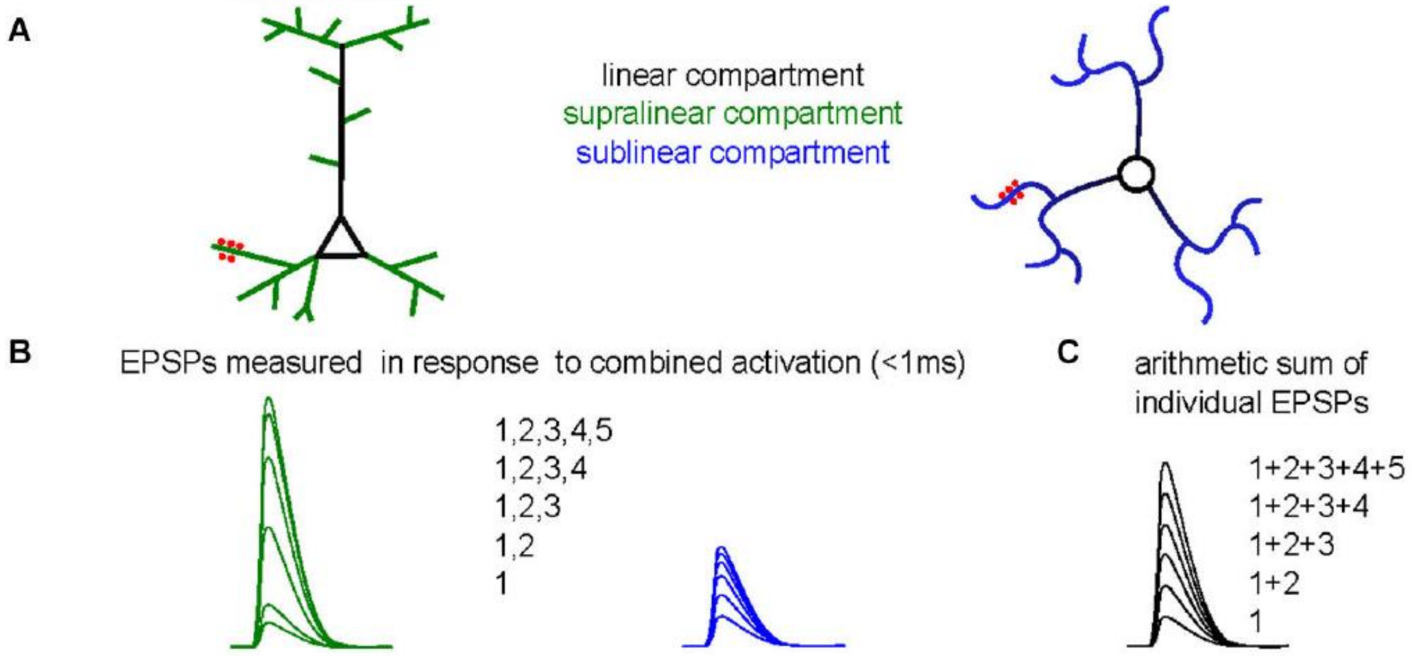


London & Hausser, Ann. Rev. Neuro., 2005

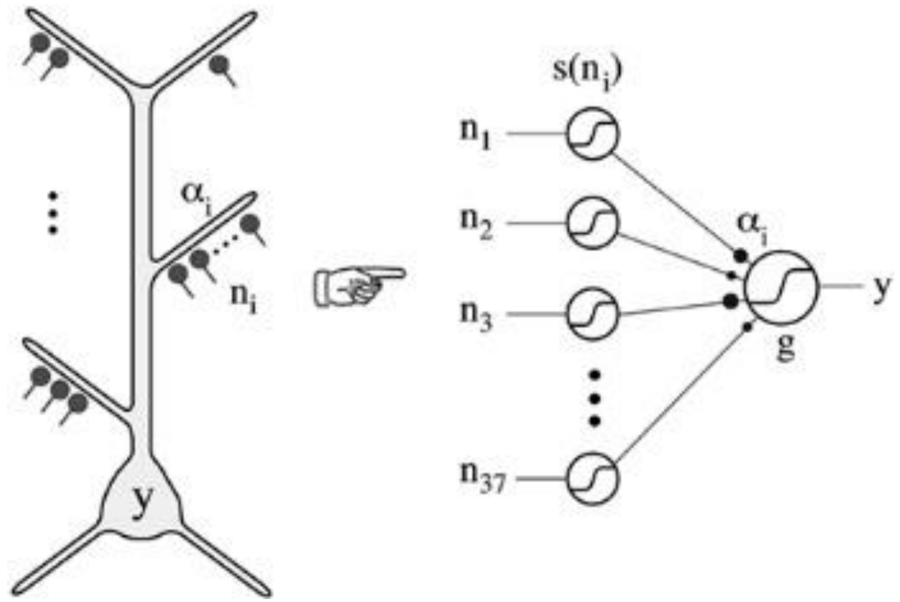
How inputs interact: Active Integration

Colab notebook – part 3

How inputs interact: Spatial Summation



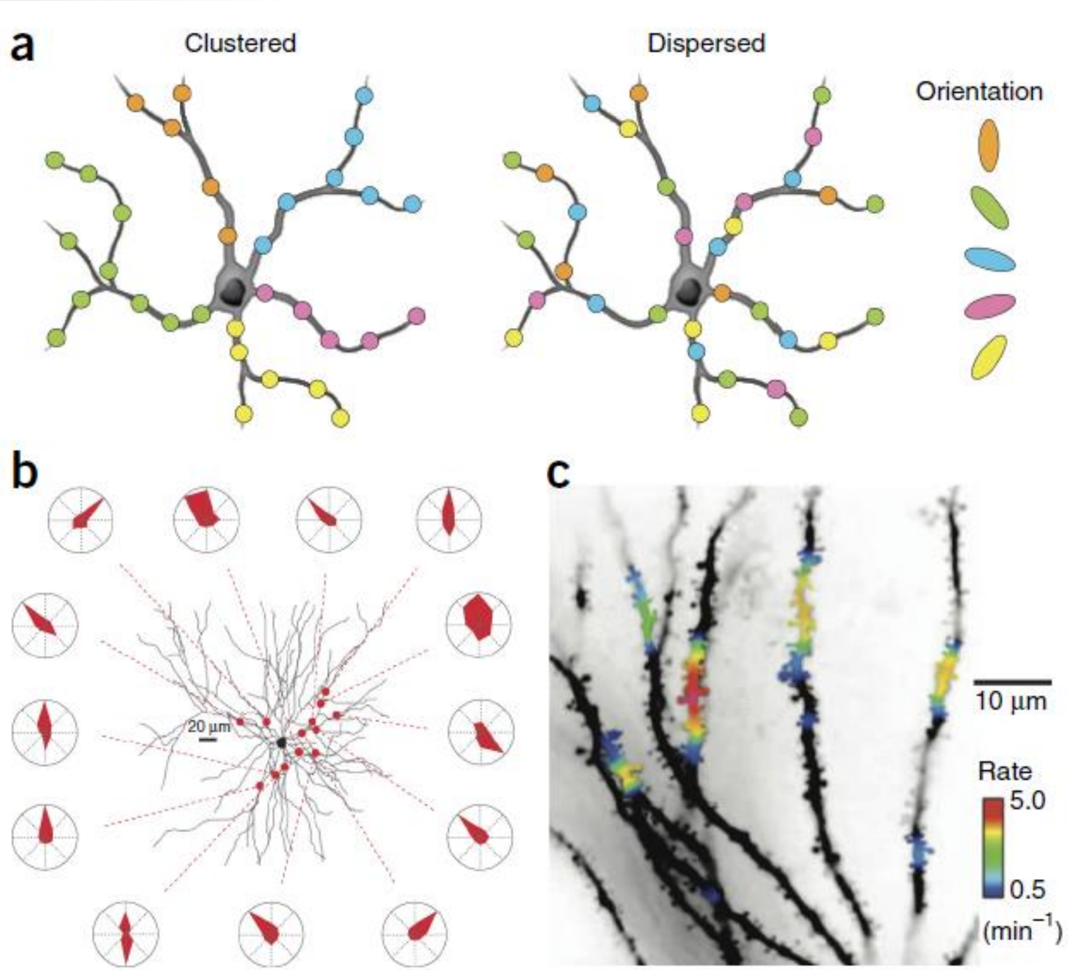
Tran-Van-Minh et al. Front. Cell. Neuroscience, 2015



Poirazi et al. 2003

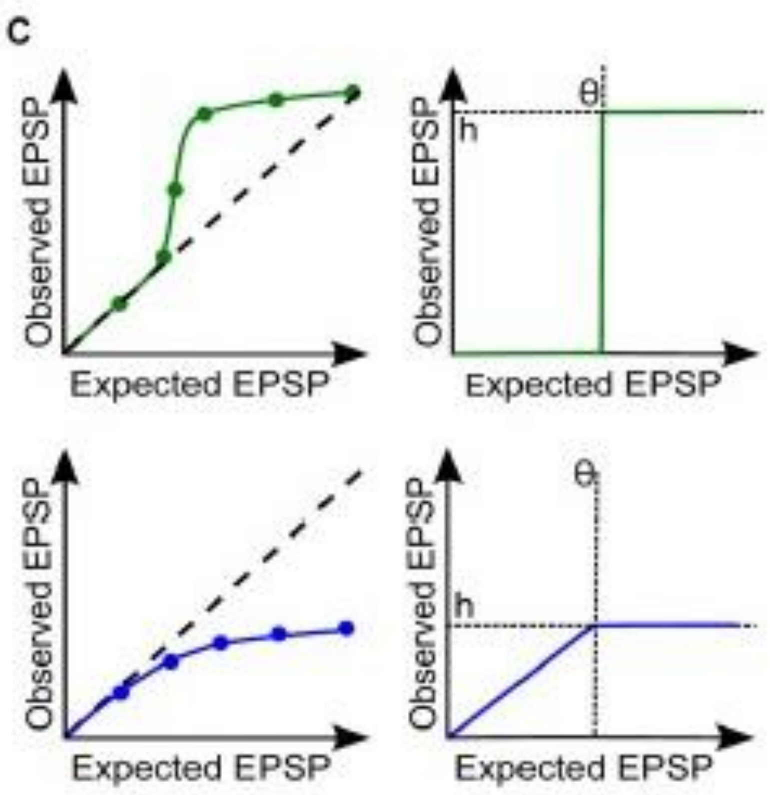
Which is the best strategy for maximum depolarization – scatter inputs to many dendrites or cluster them into one?

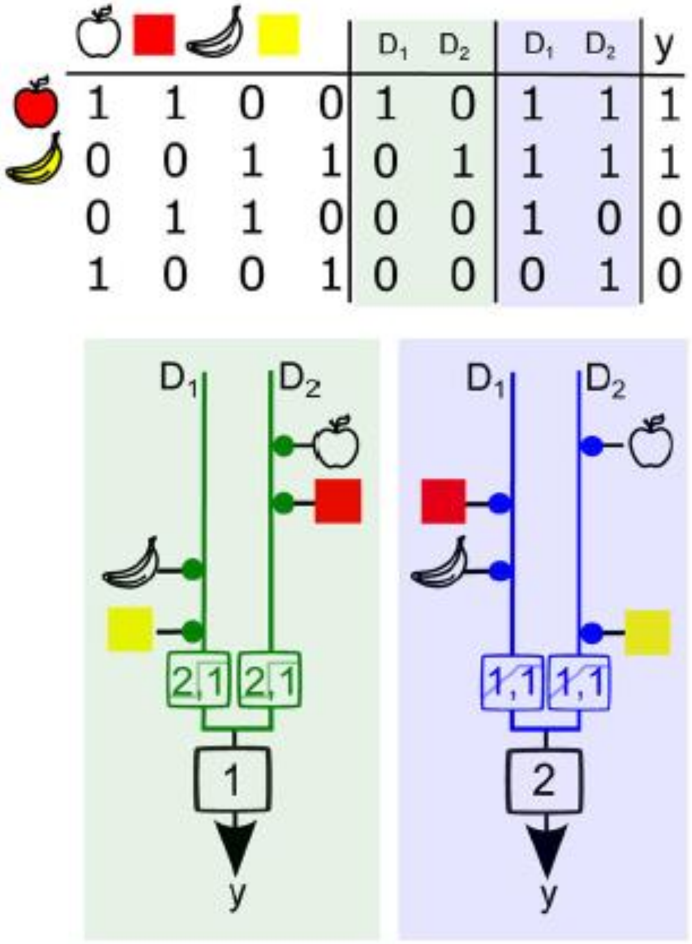
How inputs interact: Spatial Summation



Jia et al. Nature 2010 Takahashi et al. Science 2012

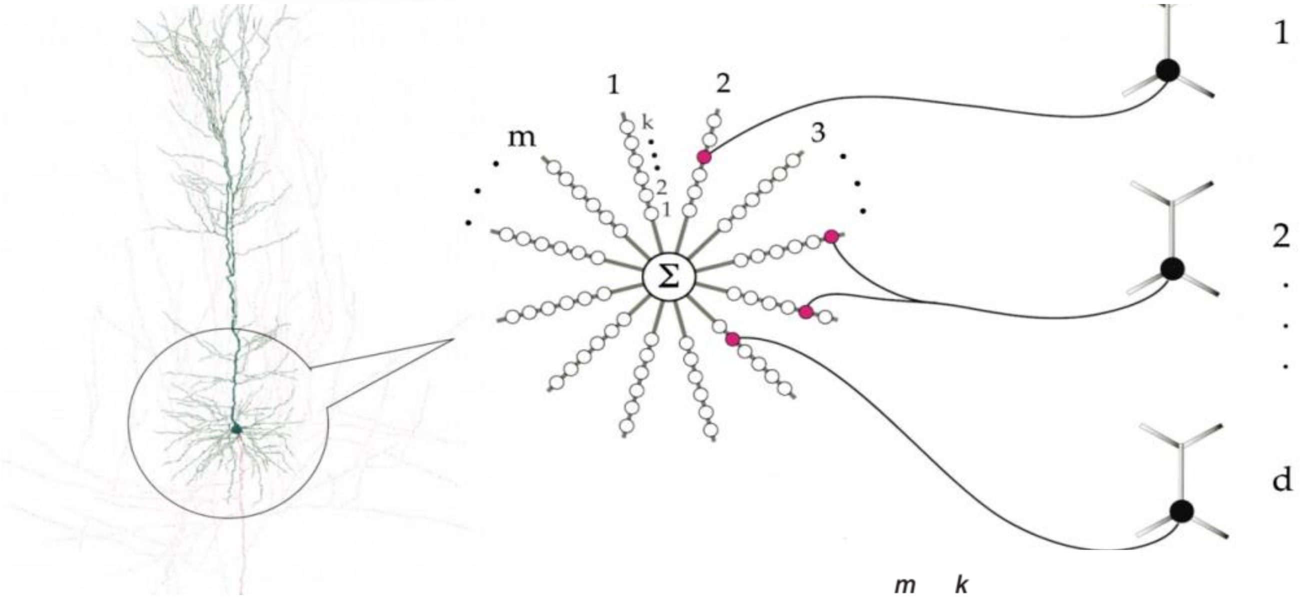
Spatial Summation & Feature Binding problem



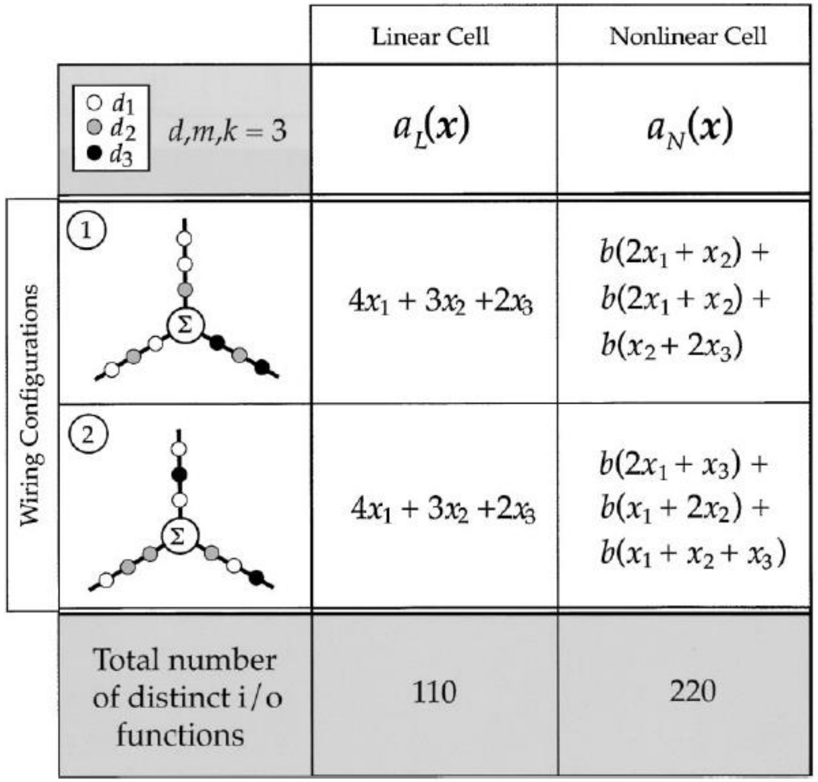


Tran-Van-Minh et al. Front. Cell. Neuroscience, 2015

Spatial Summation & Storage Capacity



A nonlinear cell can distinguish between wiring configurations



	Linear Cell	Nonlinear Cell
d_1, d_2, d_3; $d,m,k = 3$	$a_L(x)$	$a_N(x)$
Wiring Configurations ①	$4x_1 + 3x_2 + 2x_3$	$b(2x_1 + x_2) +$ $b(2x_1 + x_2) +$ $b(x_2 + 2x_3)$
Wiring Configurations ②	$4x_1 + 3x_2 + 2x_3$	$b(2x_1 + x_3) +$ $b(x_1 + 2x_2) +$ $b(x_1 + x_2 + x_3)$
Total number of distinct i/o functions	110	220

Cell's activation level:

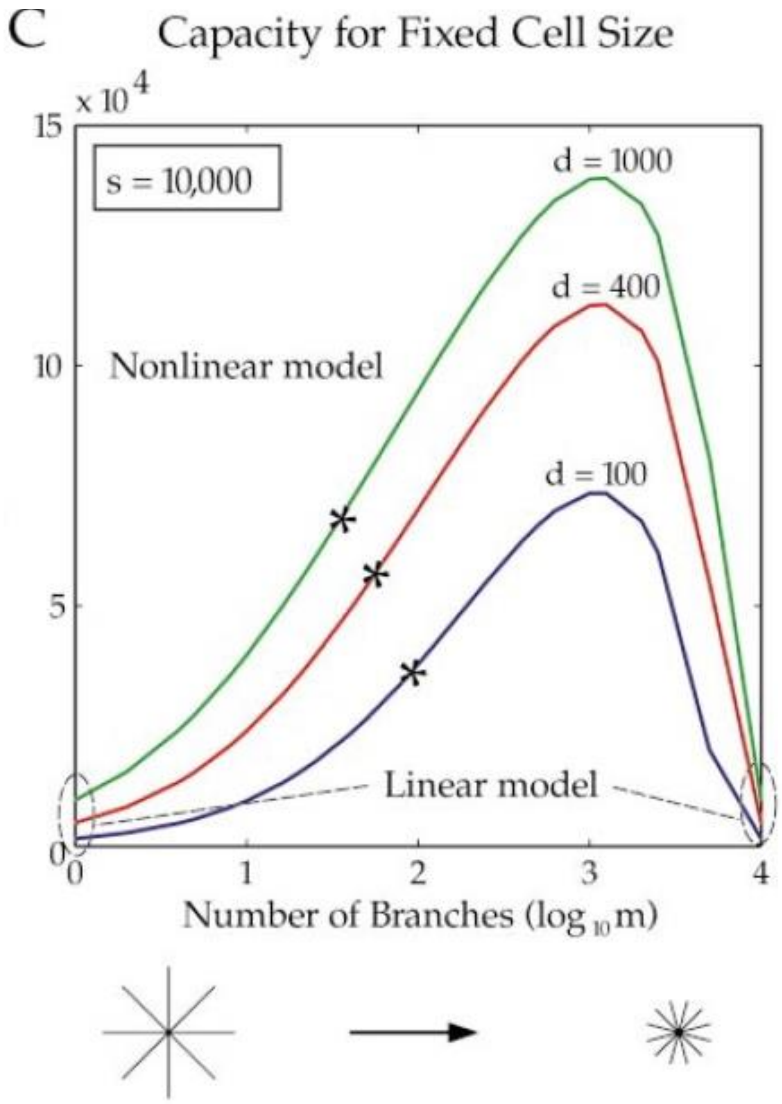
Linear
$$a_L(\mathbf{x}) = \sum_{j=1}^{m} \sum_{i=1}^{k} w_{ij} x_{ij}$$

Nonlinear
$$a_N(\mathbf{x}) = \sum_{j=1}^{m} b\left(\sum_{i=1}^{k} w_{ij} x_{ij}\right).$$

- m branches, where each branch contains k excitatory synaptic contacts.
- Each synapse is driven by one of d input lines.

Poirazi et al., Neuron 2001

Spatial Summation & Storage Capacity



$$B_L = 2\log_2 \binom{s+d-1}{s}$$

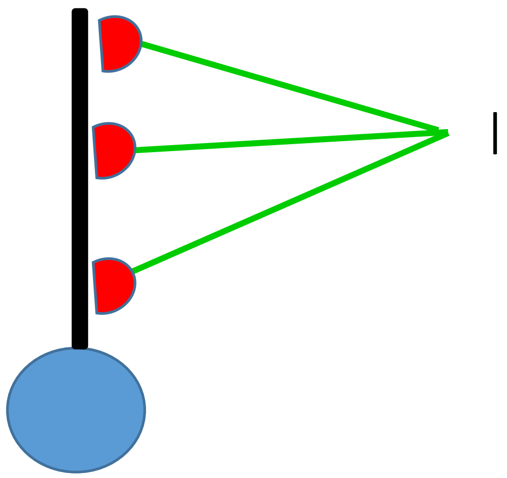
$$B_N = 2\log_2 \binom{\binom{k+d-1}{k}+m-1}{m}$$

$$s = m * k$$

Poirazi et al., Neuron 2001

How inputs interact: Temporal summation

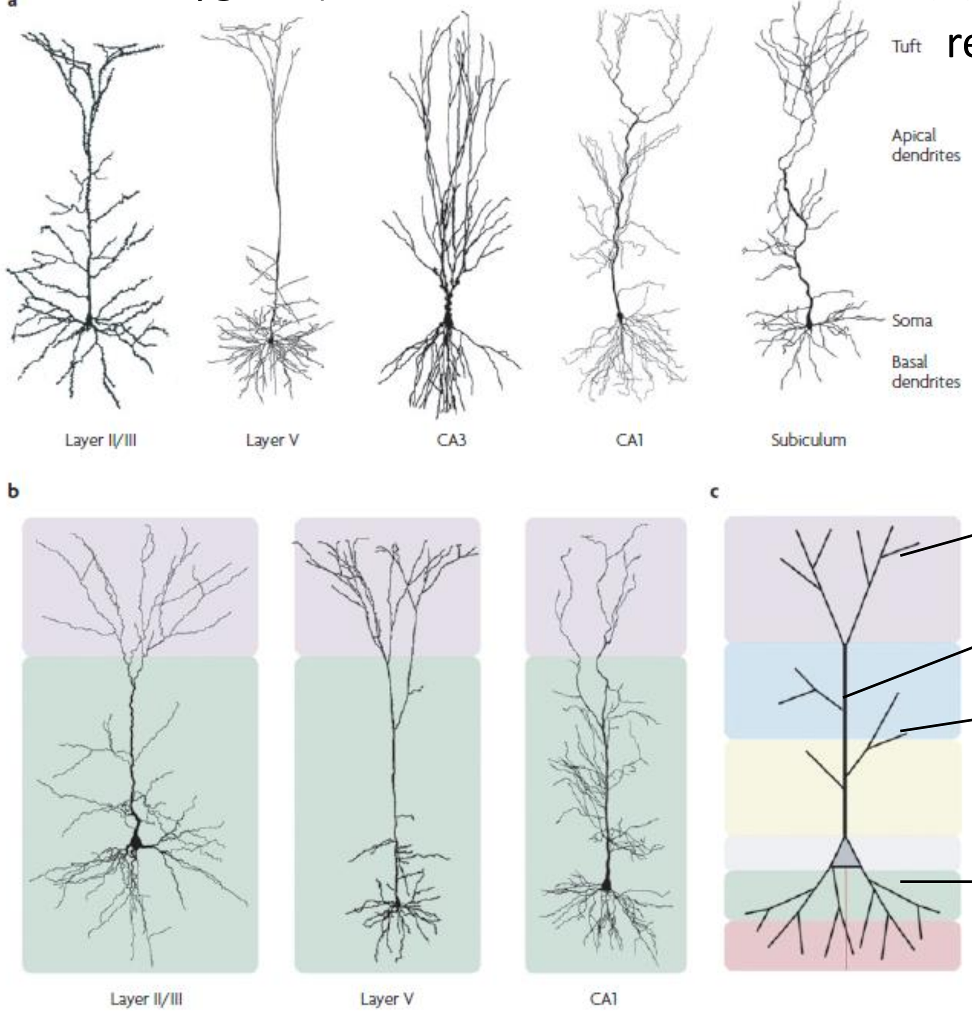
Colab notebook – part 4



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.

Apical tree

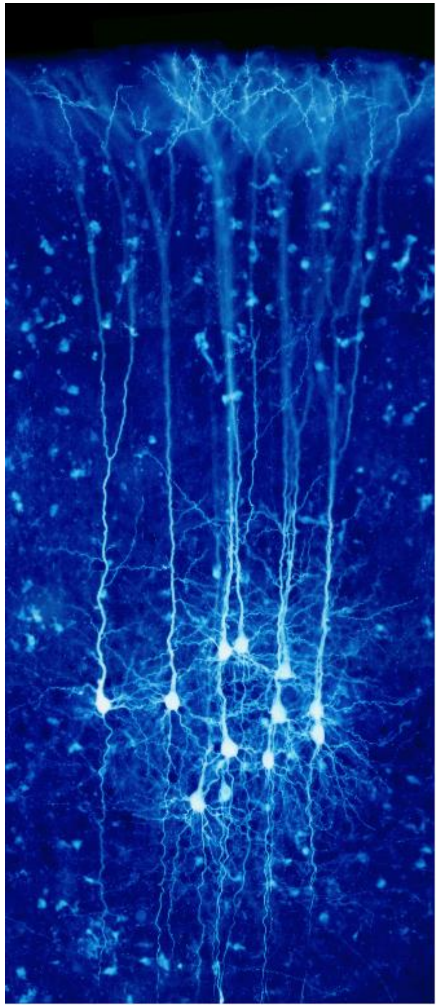
Apical tuft

Apical trunk

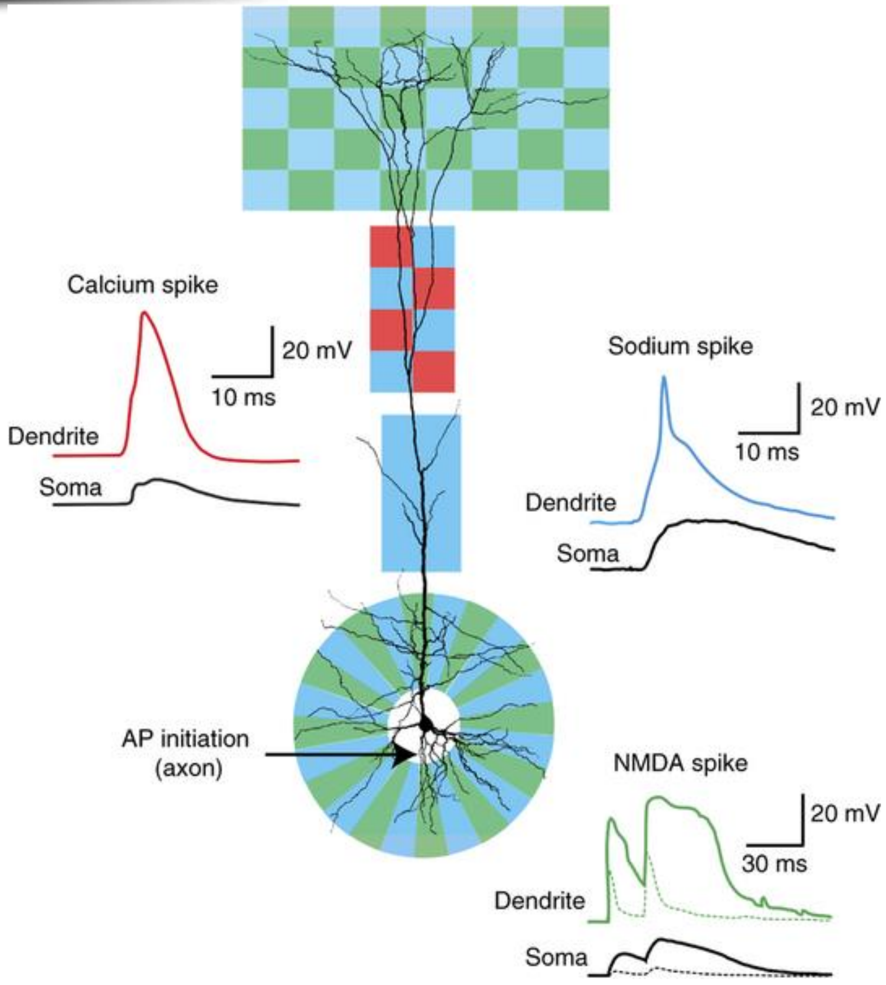
Oblique apical dendrites

Basal tree

Focusing on the cortex: The pyramidal neuron



Human brain project

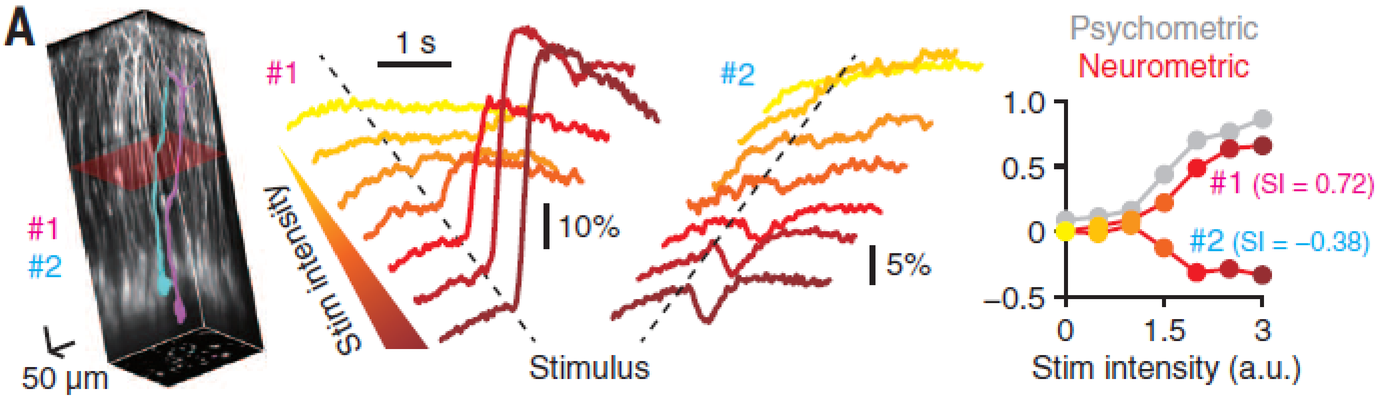


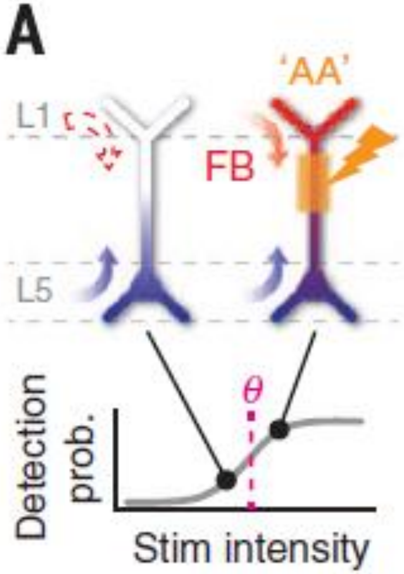
Stuart, Spruston, Nature Reviews, 2015

Functional importance: Coincidence detection

Colab notebook – part 5

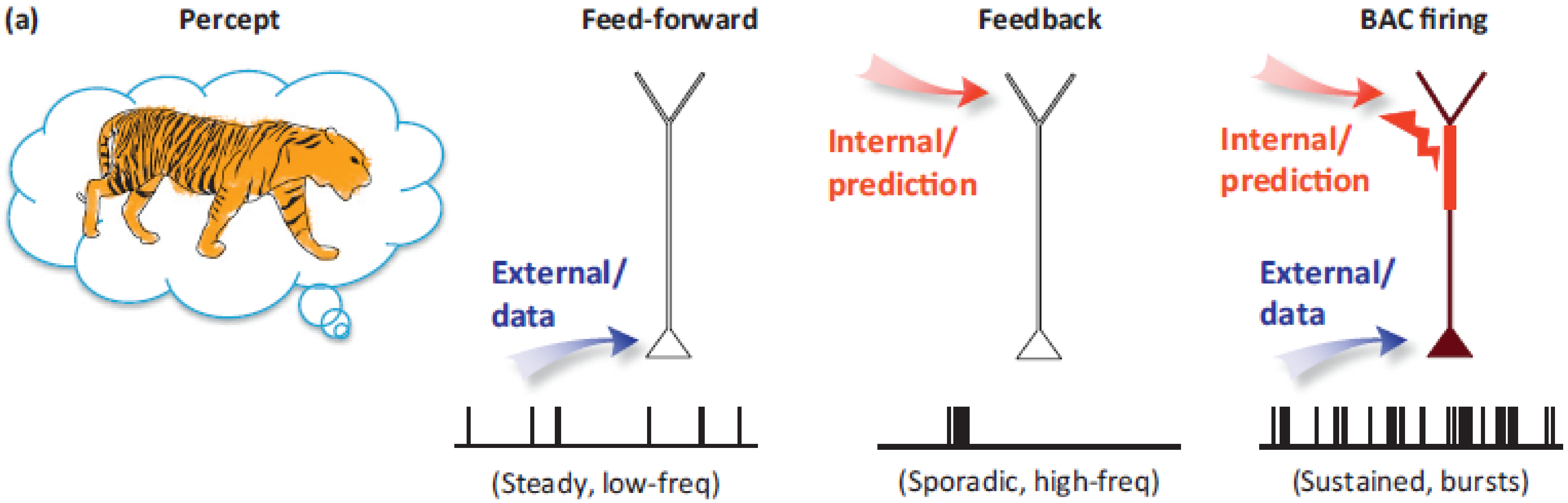
Functional importance: Perceptual threshold





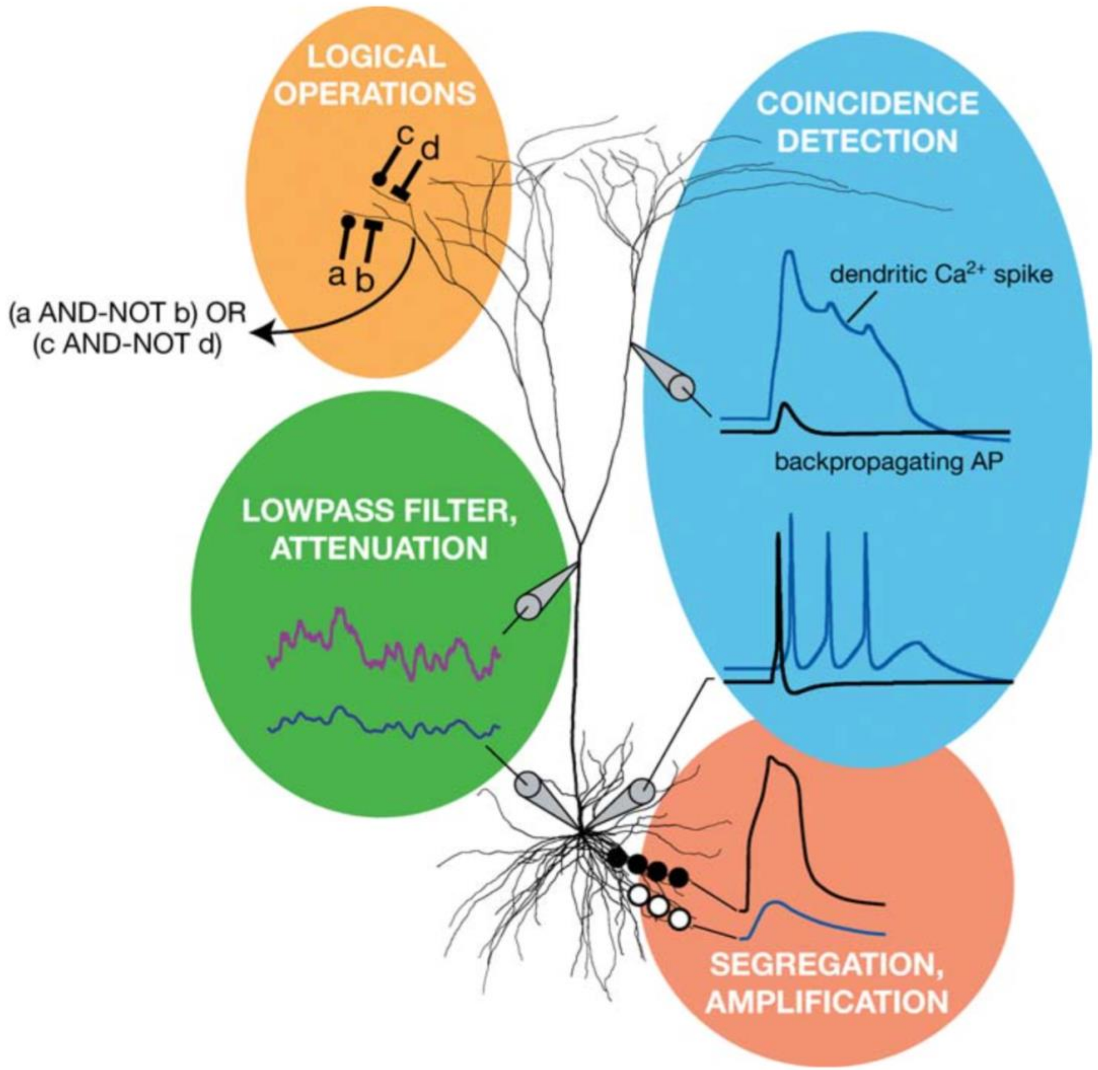
Takahashi et al. Science,2016

Functional importance: Coincidence detection



Larkum, TINS 2013

What can dendrites do?



London & Hausser 2005 Ann Rev Neuro

And why do I care how neurons do it?

Neuron
Perspective

CellPress

Engineering a Less Artificial Intelligence

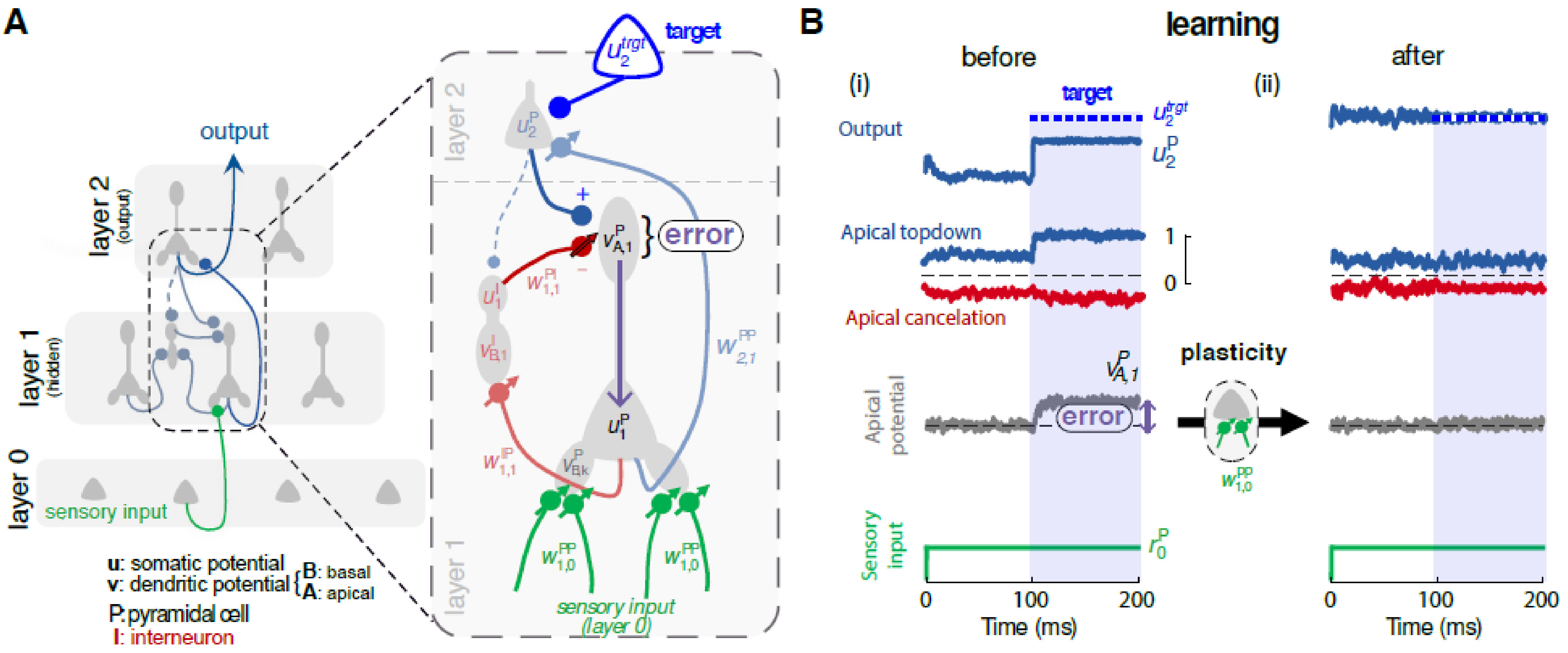
Fabian H. Sinz,[1,2,7,*] Xaq Pitkow,[6,7,8] Jacob Reimer,[6,7] Matthias Bethge,[2,3,4,5,7] and Andreas S. Tolias[6,7,8,*]
[1]Institute Bioinformatics and Medical Informatics (IBMI), University of Tübingen, Germany
[2]Bernstein Center for Computational Neuroscience, University of Tübingen, Germany
[3]Centre for Integrative Neuroscience, University of Tübingen, Germany
[4]Institute for Theoretical Physics, University of Tübingen, Germany
[5]Max Planck Institute for Biological Cybernetics, Tübingen, Germany
[6]Department of Neuroscience, Baylor College of Medicine, Houston, TX, USA
[7]Center for Neuroscience and Artificial Intelligence, BCM, Houston, TX, USA
[8]Department of Electrical and Computer Engineering, Rice University, Houston, TX, USA
*Correspondence: fabian.sinz@uni-tuebingen.de (F.H.S.), astolias@bcm.edu (A.S.T.)
https://doi.org/10.1016/j.neuron.2019.08.034

nature neuroscience
FOCUS | PERSPECTIVE
https://doi.org/10.1038/s41593-019-0520-2

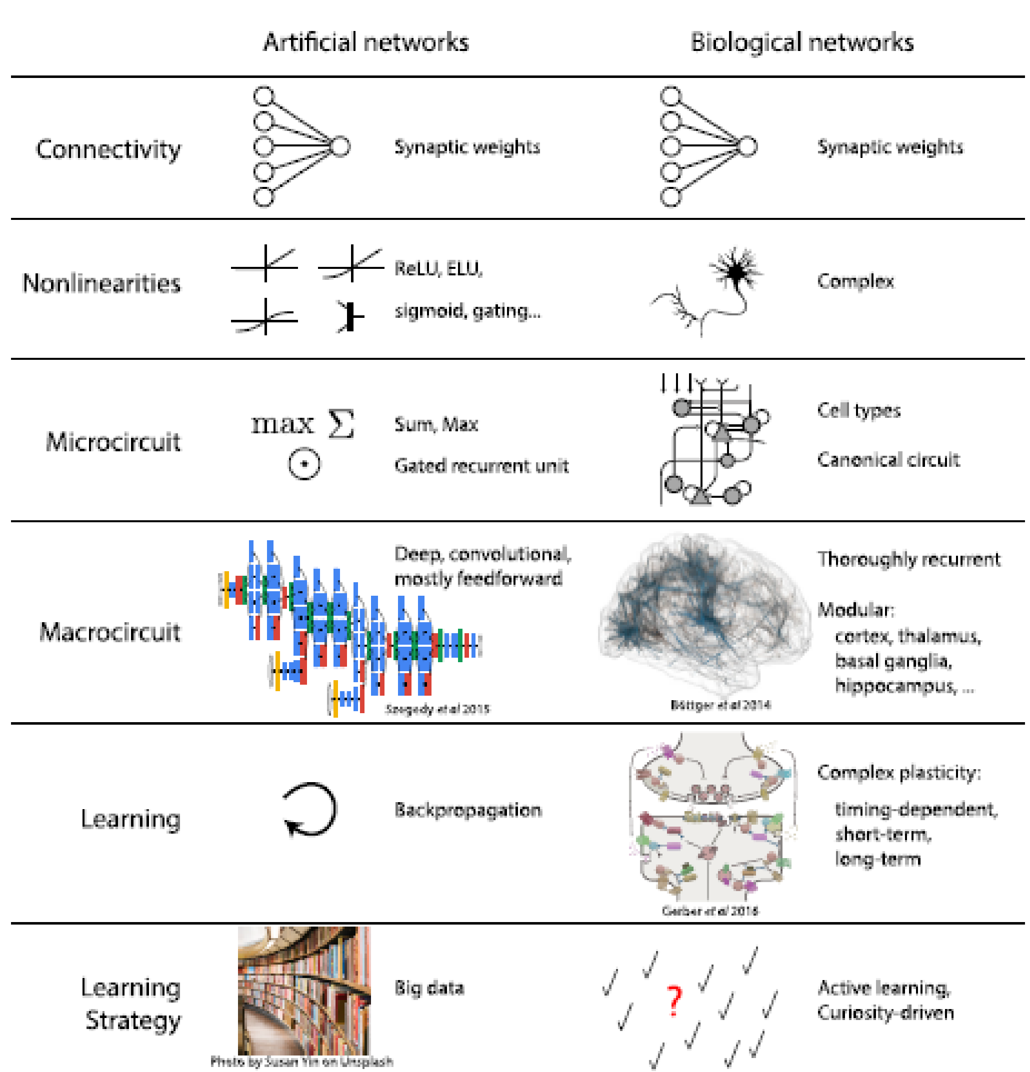
A deep learning framework for neuroscience

Blake A. Richards[1,2,3,4,42*], Timothy P. Lillicrap[5,6,42], Philippe Beaudoin[7], Yoshua Bengio[1,4,8], Rafal Bogacz[9], Amelia Christensen[10], Claudia Clopath[11], Rui Ponte Costa[12,13], Archy de Berker[7], Surya Ganguli[14,15], Colleen J. Gillon[16,17], Danijar Hafner[15,18,19], Adam Kepecs[20], Nikolaus Kriegeskorte[21,22], Peter Latham[23], Grace W. Lindsay[22,24], Kenneth D. Miller[22,24,25], Richard Naud[26,27], Christopher C. Pack[3], Panayiota Poirazi[28], Pieter Roelfsema[29], João Sacramento[30], Andrew Saxe[31], Benjamin Scellier[1,8], Anna C. Schapiro[32], Walter Senn[13], Greg Wayne[5], Daniel Yamins[33,34,35], Friedemann Zenke[36,37], Joel Zylberberg[4,38,39], Denis Therien[7,42] and Konrad P. Kording[4,40,41,42]

Dendritic cortical microcircuits approximate the backpropagation algorithm.



Sacramento et al. NIPS 2018



Sinz et al. Neuron 2019

Thank you!