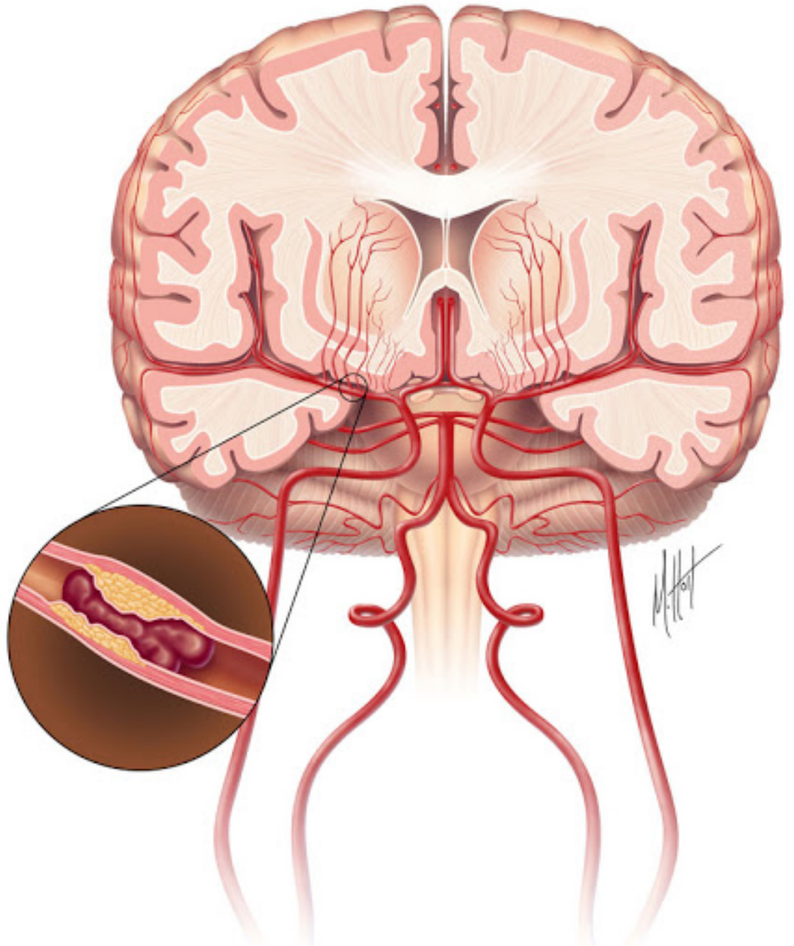
Modelling transcranial alternating current stimulation (tACS)

NetPyNE Course 2021

Monica Bell Vila, University of Toronto

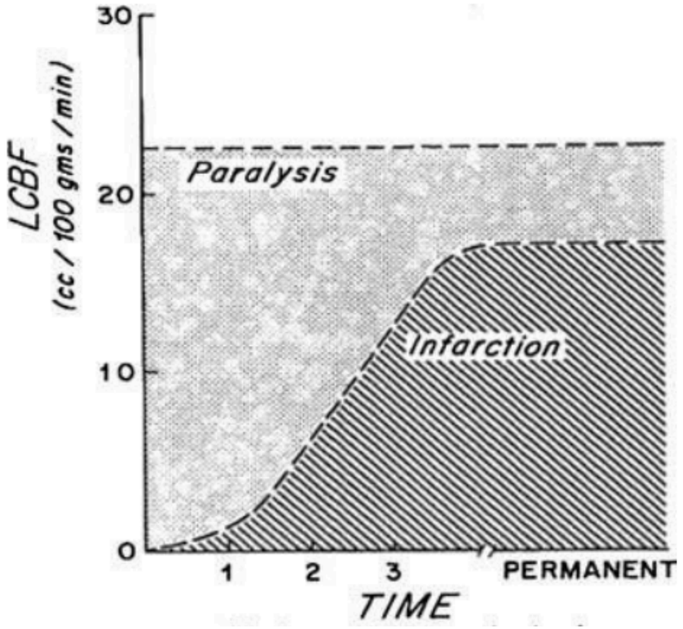
Introduction

- MSc Candidate at University of Toronto, Department of Medical Biophysics
- Investigating localized subcutaneous alternating current stimulation (sACS) to potentiate recovery from ischaemic stroke in preclinical rat model
- Thrombolysis: typically 3-4.5 hour timeframe
  Thrombectomy: typically within 6 hours [2]
- 60% of stroke patients do not entirely recover, greater than 40% have severe disability requiring ongoing support [5]

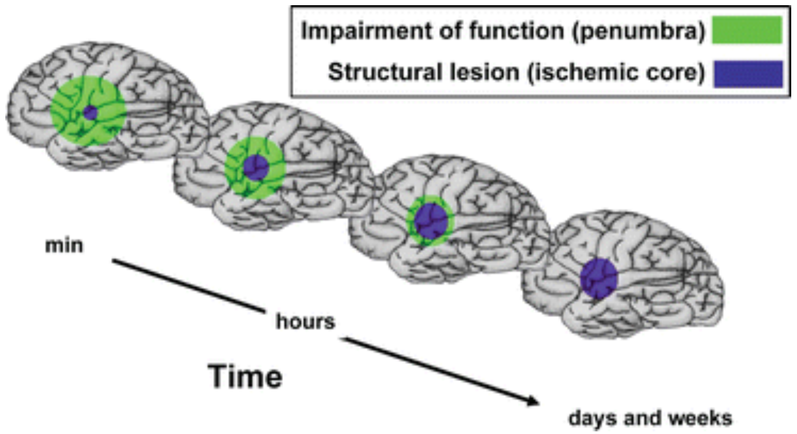


The Internet Stroke Center. www.strokecenter.org/patients/about-stroke/ischemic-stroke

Neuronal response following stroke



Donnan, G. A. (2007)



Mergenthaler, P., Dirnagl, U., & Kunz, A. (2016)

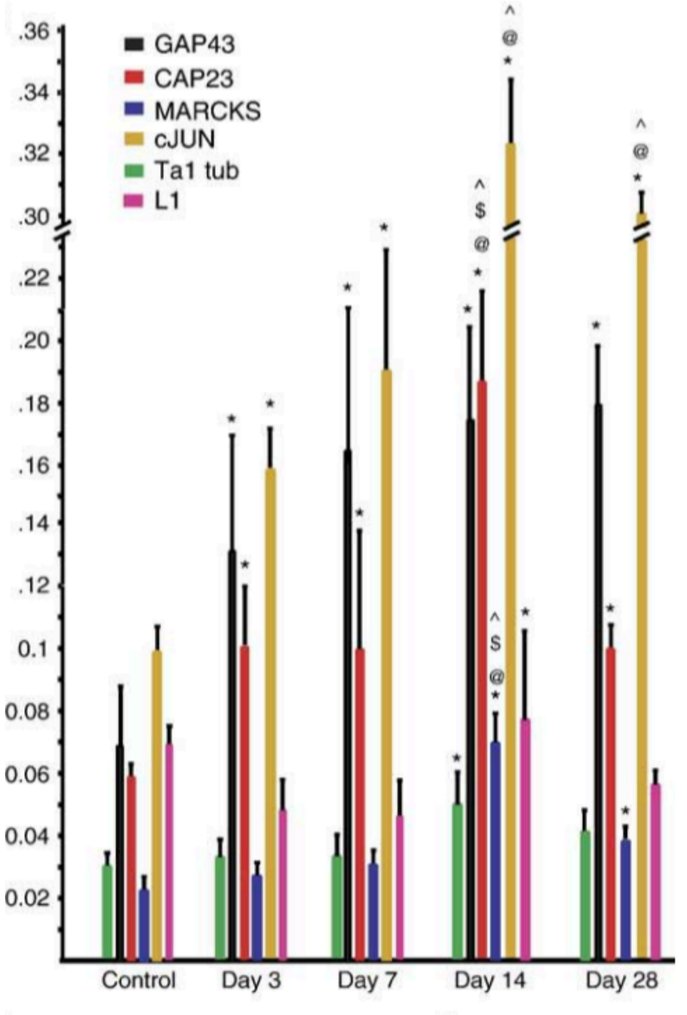
- Local Cerebral Blood Flow (LCBF) impaired because of occlusion
- Penumbra: region of impaired functioning, reversible
- Infarction: necrotic core, irreversible
- Sustained reduction of LCBF increases recruitment of penumbra into core [1,2]
- Penumbra region can be reversed with restored reperfusion

[1] Donnan, G. A. (2007). The ischemic penumbra pathophysiology, imaging and therapy, [2] Mergenthaler, P., Dirnagl, U., & Kunz, A. (2016). Ischemic Stroke: Basic Pathophysiology and Clinical Implication.

Neuronal response following stroke

- Expression of neuronal growth-promoting genes and axonal sprouting in perilesional cortex, correlated with functional recovery [1]
- Re-mapping of synaptic connections adjacent to infarct [2]
- Identifies potential growth promoting or regenerative zone

Neuroplasticity: the ability of the brain to change and form/ prune/re-shape synaptic connections

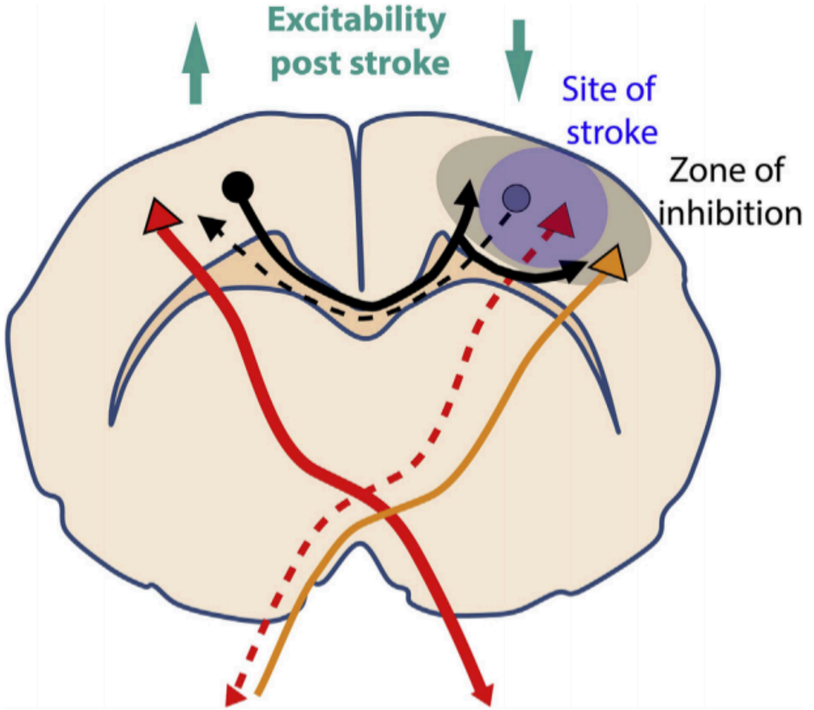


[1]Li, S., & Carmichael, T. *Neurobiology of disease*, (2006). , [2]Carmichael, T., et al. *Neurobiology of disease,* (2001).

Neuronal response following stroke

Interhemispheric imbalance model [1]:

- In healthy brain: each side of the brain inhibits the other equally
- After stroke, reduced interhemispheric inhibition in the stroke-affected area
- Increased interhemispheric inhibition in contralesional hemisphere
- Unopposed inhibition from the healthy to the damaged side prevents connectivity remapping
- Disruption to functional connectivity predicts poorer motor performance



[1] Boddington, L. & Reynolds, J. *Brain Stimulation,* (2017).

Targeted treatment

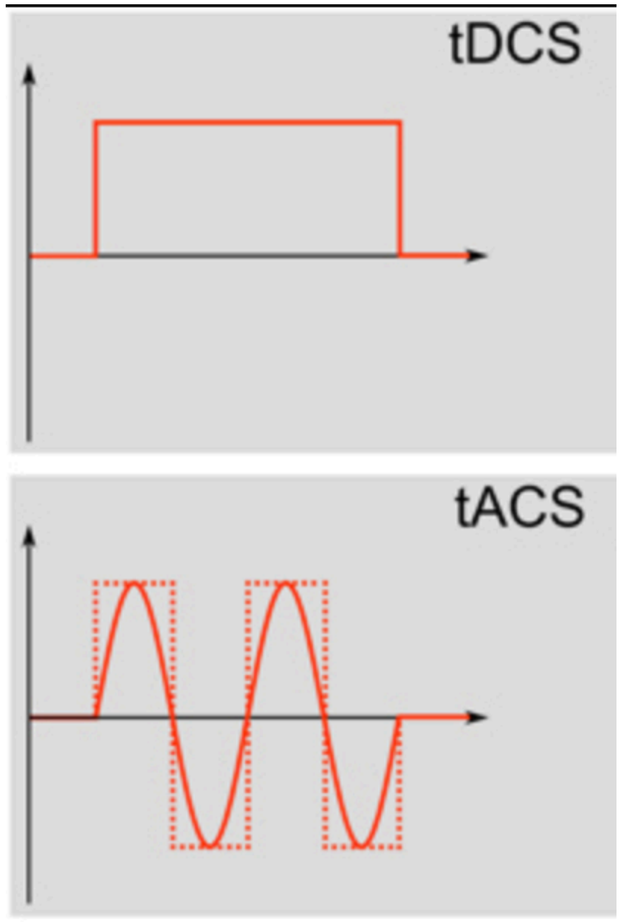
- **Neuromodulation:** Use of stimulus (chemical/electric/optical/sound) to modulate (amplifying/attenuating) neuronal activity

- Potentially improve stroke treatment by targeting stimulus to different regions [1]

- **Transcranial electric stimulation (tES):** delivery of low-intensity electrical currents to brain, potential to be localized via electrode placement

[1]Johnson et al, *IEEE Trans. Biomed. Eng.* (2013), [2] Rhim, Lee, & Lee *Pharmaceutical Research* (2013)

Subcutaneous alternating current stimulation (sACS)



[2] Herrmann,et al, *Frontiers in Human Neuroscience* (2013)

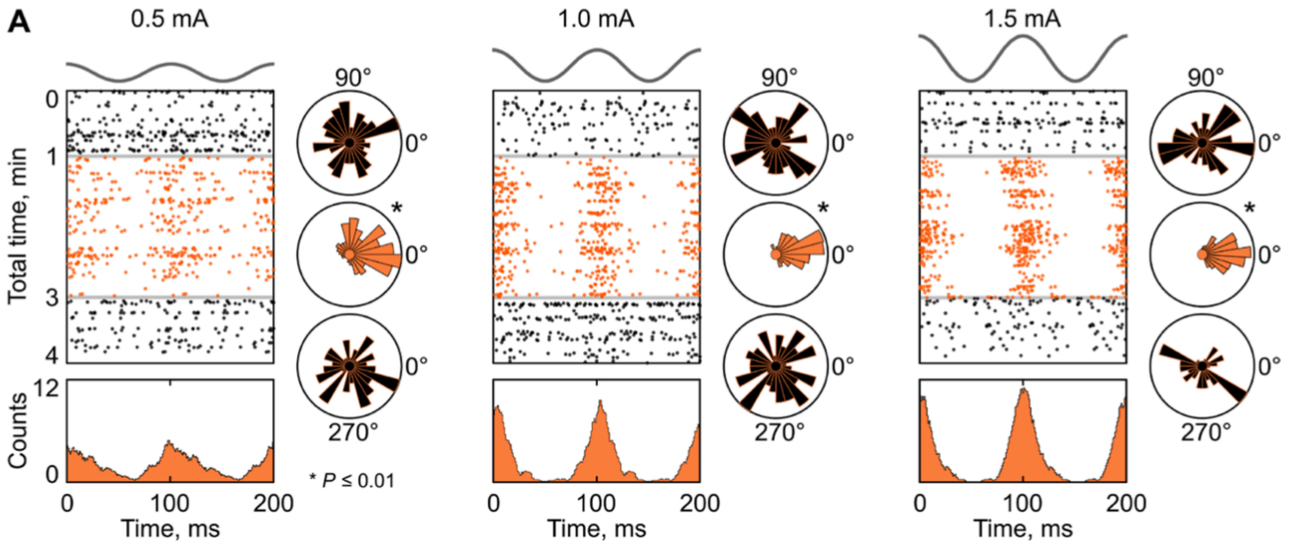
- Transcranial Direct Current Stimulation (tDCS): constant intensity for duration of stimulation
- Transcranial Alternating Current Stimulation (tACS): sinusoidal wave, oscillating in amplitude at given frequency
- tDCS modulates cortical excitability, tACS entrains cortical networks [1]

[1] Inukai et al, *Frontiers in Human Neuroscience* (2016)

ACS mechanisms

- ACS believed to specifically entrain intrinsic oscillations similar in frequency with applied stimulation [1]
- Non-linear dependence on intensity - 140Hz, 0.2mA resulted in cortical inhibition, 1mA resulted in excitation [2]
- 2 main responses to tACS: increased bursts and phase entrainment [3]
- Frequency dependent

…Not well understood

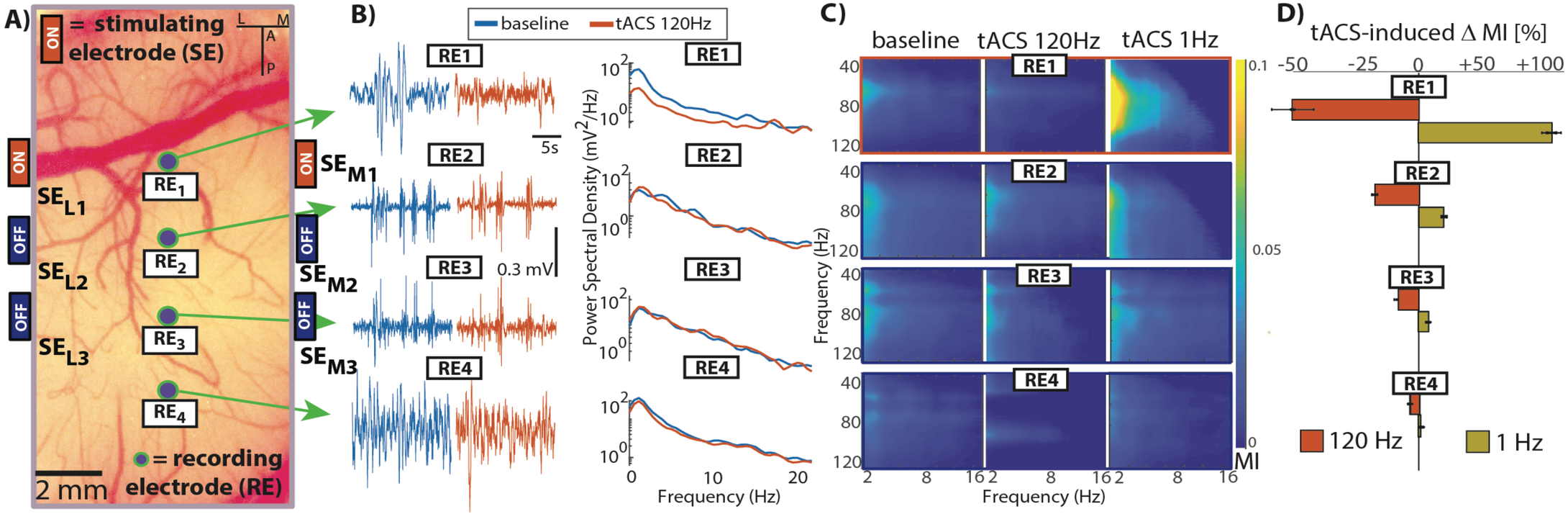


Johnson et al, *Science Advances* (2020)

[1] Helfrich, R., et al. *Current biology,* (2014). [2] Herrmann, C., et al. *Frontiers in human neuroscience, (*2013), [3] Johnson et al, *Science Advances* (2020)

Model of ACS

- Create a model of a network responding to differing ACS parameters
- Validate against experimental results
- Use to predict ACS parameters to deliver experimentally



Images courtesy of Dr. Paolo Bazzigaluppi

ACS in NetPyNE

- Currently no extracellular stimulation implementation in NetPyNE
- Option to insert extracellular potentials next to membranes in NEURON
- Goal: implement extracellular ACS stimulation to a network built using NetPyNE

Include extracellular mechanism

```
import matplotlib.pyplot as plt
import os
import numpy as np
from netpyne import specs, sim
import neuron
netParams = specs.NetParams()
```

```
## Cell parameters/rules
PYRcell = {'secs': {}}
PYRcell['secs']['soma'] = {'geom': {}, 'mechs': {}}
PYRcell['secs']['soma']['geom'] = {
    'diam': 18.8,
    'L': 18.8,
    'Ra': 123.0,
    }  # soma geometry
PYRcell['secs']['soma']['mechs']['hh'] = {
    'gnabar': 0.12,
    'gkbar': 0.036,
    'gl': 0.003,
    'el': -70}  # soma hh mechanism
PYRcell['secs']['soma']['mechs']['extracellular'] = {}
PYRcell['secs']['dend'] = {'geom': {}, 'topol': {}, 'mechs': {}}
PYRcell['secs']['dend']['geom'] = {'diam': 5.0, 'L': 150.0, 'Ra': 150.0, 'cm': 1}
PYRcell['secs']['dend']['topol'] = {'parentSec': 'soma', 'parentX': 1.0, 'childX': 0}
PYRcell['secs']['dend']['mechs']['pas'] = {'g': 0.0000357, 'e': -70}

netParams.cellParams['PYR'] = PYRcell
```

Specify population/synapse parameters

```
## Population parameters
netParams.popParams['E'] = {
    'cellType': 'PYR',
    'numCells': 20,
    'xRange' : [0, 250],
    'yRange' : [0, 250],
    'zRange' : [0, 250]}
netParams.popParams['I'] = {
    'cellType': 'PYR',
    'numCells': 5,
    'xRange' : [0, 250],
    'yRange' : [0, 250],
    'zRange' : [0, 250]
    }


## Synaptic mechanism parameters
netParams.synMechParams['exc'] = {
    'mod': 'Exp2Syn',
    'tau1': 0.2,
    'tau2': 1.8,
    'e': 0}  # excitatory synaptic mechanism

netParams.synMechParams['inh'] = {
    'mod': 'Exp2Syn',
    'tau1': 0.1,
    'tau2': 9.0,
    'e': -80}  # inhibitory synaptic mechanism
```

```
#Background activity parameters
netParams.stimSourceParams['bkg'] = {
    'type': 'NetStim',
    'rate': 2,
    'noise': 0.5}
netParams.stimTargetParams['bkg'] = {
    'source': 'bkg',
    'conds' : {'pop': ['E','I']},
    'weight': 0.015,
    'delay': 5 ,
    'synMech': 'exc'}
```

```
## Cell connectivity rules
netParams.connParams['E->all'] = {    #  S -> I label
    'preConds': {'pop': 'E'},       # conditions of presyn cells
    'postConds': {'pop': ['E','I']},      # conditions of postsyn cells
    'probability': 0.1,                 # probability of connection
    'weight': '0.005*post_ynorm',                 # synaptic weight
    'delay': 'dist_3D/propVelocity',  # transmission delay (ms), spatially dependent
    'synMech': 'exc'}                # synaptic mechanism
#'postConds': {'y': [100,1000]}

netParams.connParams['I->all'] = {    #  S -> I label
    'preConds': {'pop': 'I'},       # conditions of presyn cells
    'postConds': {'pop': ['E','I']},      # conditions of postsyn cells
    'probability': 0.1,                 # probability of connection
    'weight': '0.005*post_ynorm',                 # synaptic weight
    'delay': 'dist_3D/propVelocity',  # transmission delay (ms), spatially dependent
    'synMech': 'inh'}                # synaptic mechanism
```

Specify parameters for stimulation

```
# The parameters of the extracellular point current source
acs_params = {'position': [0., 0, 250.], #um
              'amp': 200.,  # uA,
              'stimstart' : 1000, # ms
              'stimend': 2000, #ms
              'frequency': 10, #Hz
              'sigma': 0.3 #decay constant
             }
```

Create network

```
# # Simulation options
simConfig = specs.SimConfig()       # object of class SimConfig to store simulation configurat

simConfig.duration = 4000          # Duration of the simulation, in ms
#simConfig.dt = 0.025                # Internal integration timestep to use
simConfig.verbose = False           # Show detailed messages
simConfig.recordTraces = {'V_soma':{'sec':'soma','loc':0.5,'var':'v'}}  # Dict with traces to
simConfig.recordStep = 0.1          # Step size in ms to save data (eg. V traces, LFP, etc)
simConfig.filename = 'netpynenetwork'  # Set file output name
simConfig.savePickle = False        # Save params, network and sim output to pickle file
simConfig.saveJson = False

simConfig.analysis['plotRaster'] = {}        # Plot a raster
simConfig.analysis['plotTraces'] = {'include': [0]}

sim.create(netParams = netParams)
```

Calculate stimulation intensity at cell

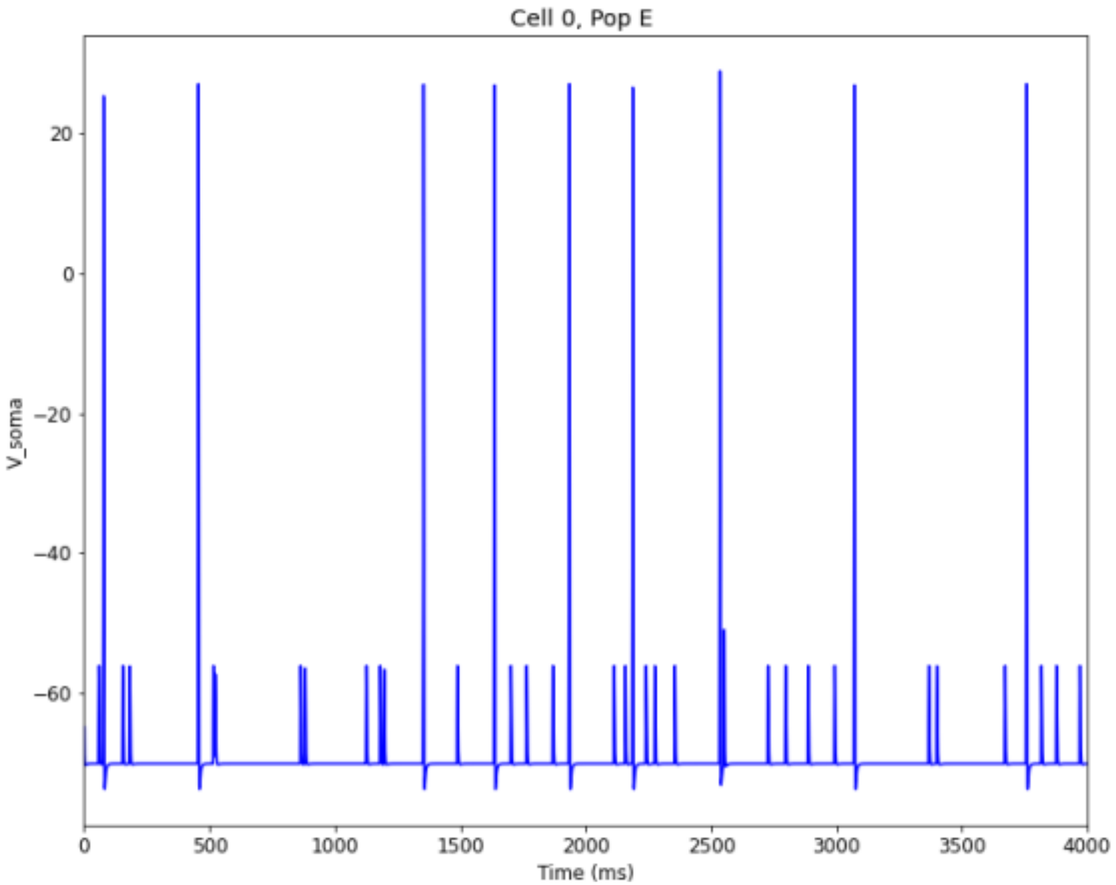
```python
def make_extracellular_stimuli(acs_params, cell):
    """ Function to calculate and apply external potential """
    x0, y0, z0 = acs_params['position']
    ext_field = np.vectorize(lambda x, y, z: 1 / (4 * np.pi *
                                                  acs_params['sigma'] * np.sqrt((x0 - x)**2 + (y0 - y)**2 + (z0 - z)**2)))

    stimstart = acs_params['stimstart']
    stimend = acs_params['stimend']
    stimdif = stimend-stimstart

    # MAKING THE EXTERNAL FIELD
    n_tsteps = int(stimdif/ simConfig.dt + 1)
    n_start = int(stimstart/simConfig.dt)
    n_end = int(stimend/simConfig.dt +1)
    t = np.arange(start=n_start, stop=n_end) * simConfig.dt
    pulse = acs_params['amp'] * 1000. * np.sin(2 * np.pi * acs_params['frequency'] * t / 1000)

    v_cell_ext = np.zeros((cell.secs['soma']['hObj'].nseg, n_tsteps))
    v_cell_ext[:, :] = ext_field(cell.getSomaPos()[0], cell.getSomaPos()[1], cell.getSomaPos()[2]).reshape(cell.secs['soma']['hObj'].nseg, 1) * pulse.reshape(1, n_tsteps)
    insert_v_ext(cell,v_cell_ext, t)
    return ext_field, pulse
```

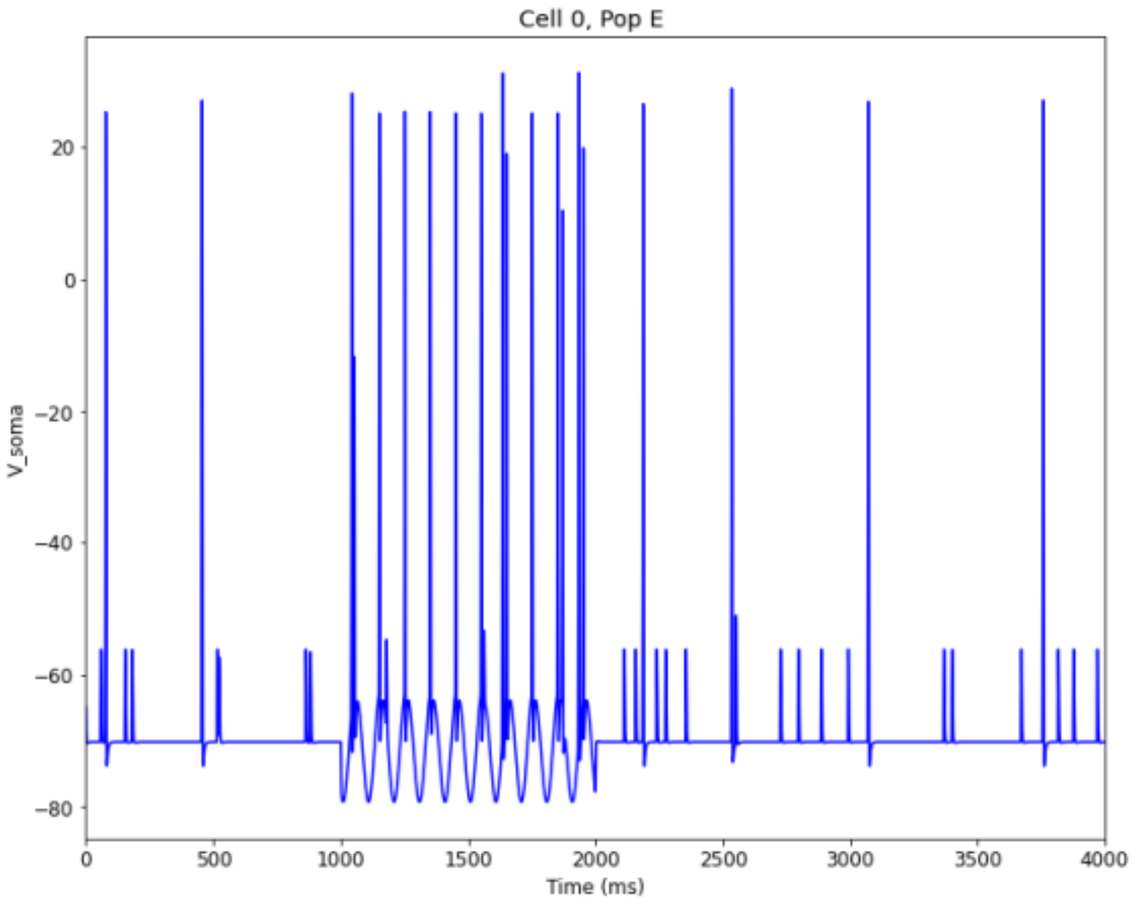
Insert stimulation to extracellular space

```
def insert_v_ext(cell, v_ext, t_ext):

    cell.t_ext = neuron.h.Vector(t_ext)
    cell.v_ext = []
    for v in v_ext:
        cell.v_ext.append(neuron.h.Vector(v))

    # play v_ext into e_extracellular reference
    i = 0
    cell.v_ext[i].play(cell.secs['soma']['hObj'](0.5)._ref_e_extracellular, cell.t_ext)
```

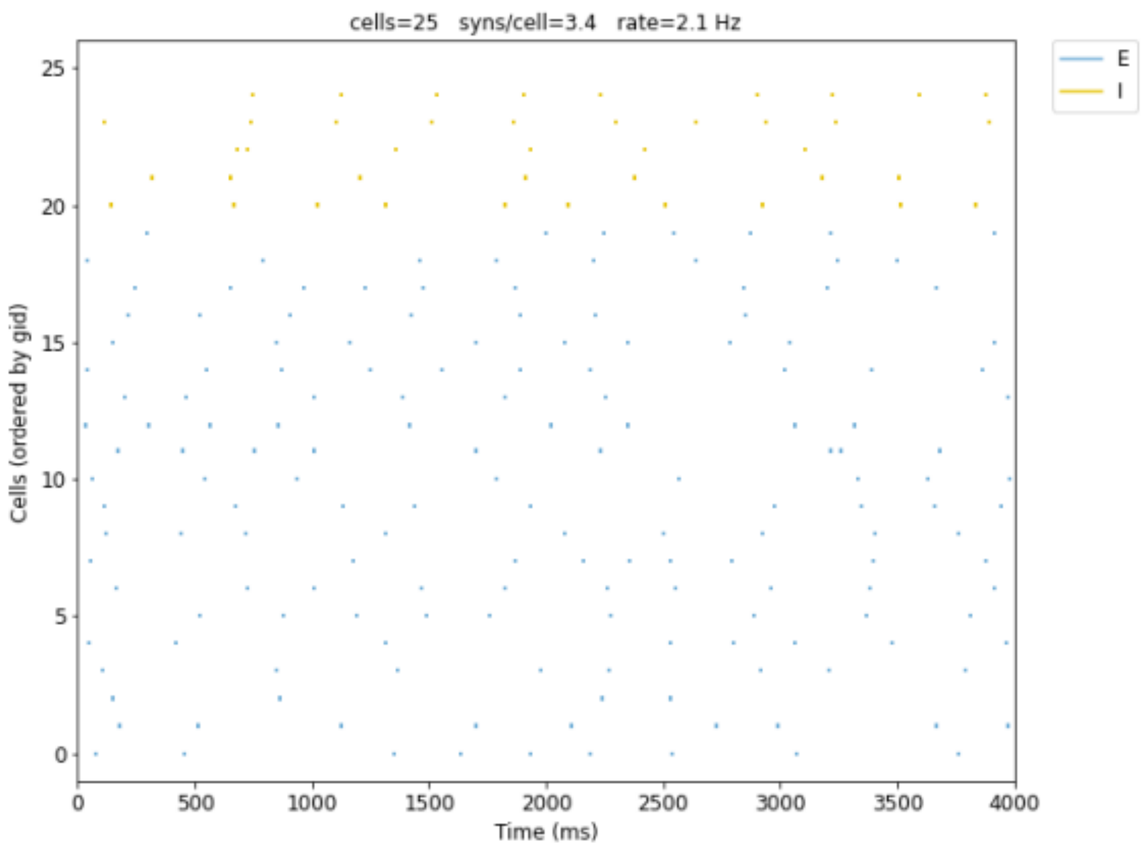
```
#Add extracellular stim
for c in range(len(sim.net.cells)):
    ext_field, pulse = make_extracellular_stimuli(acs_params, sim.net.cells[c])
```

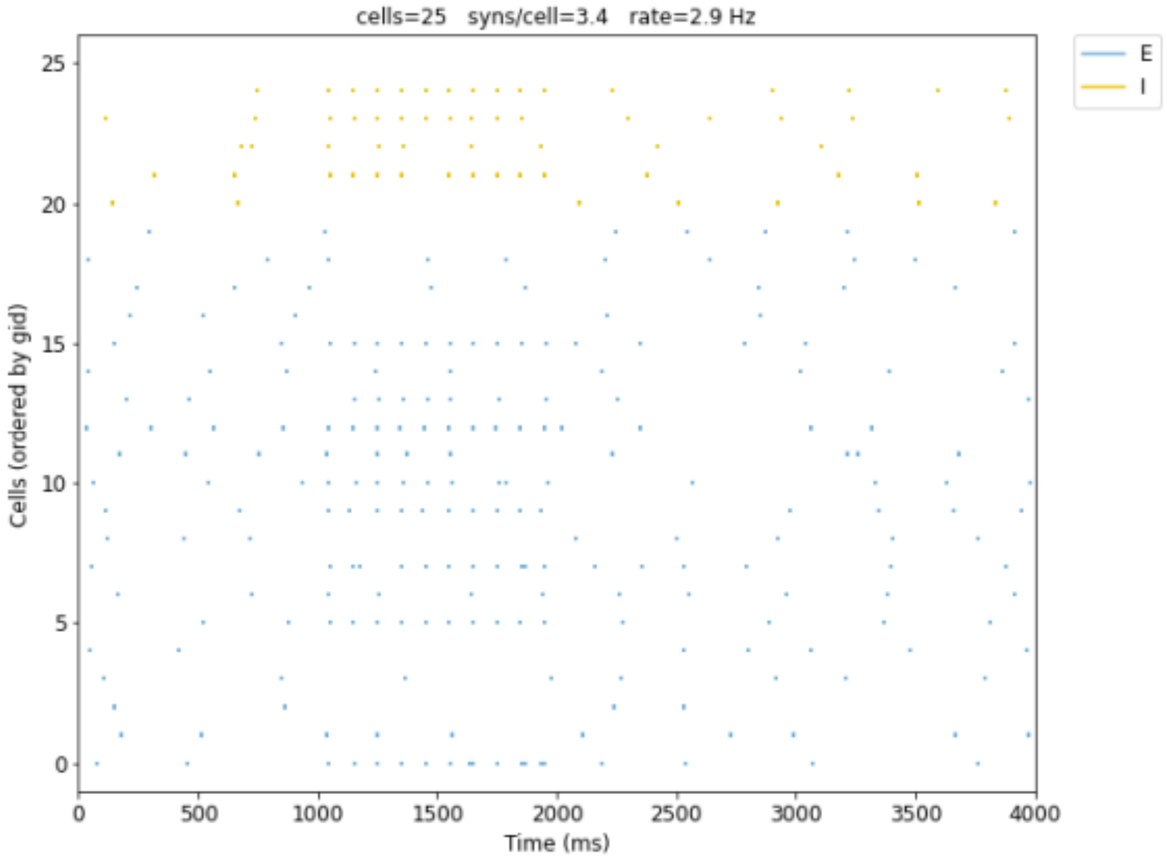
Results


Cell 0, Pop E
V_soma
Time (ms)
20
0
−20
−40
−60
0
500
1000
1500
2000
2500
3000
3500
4000

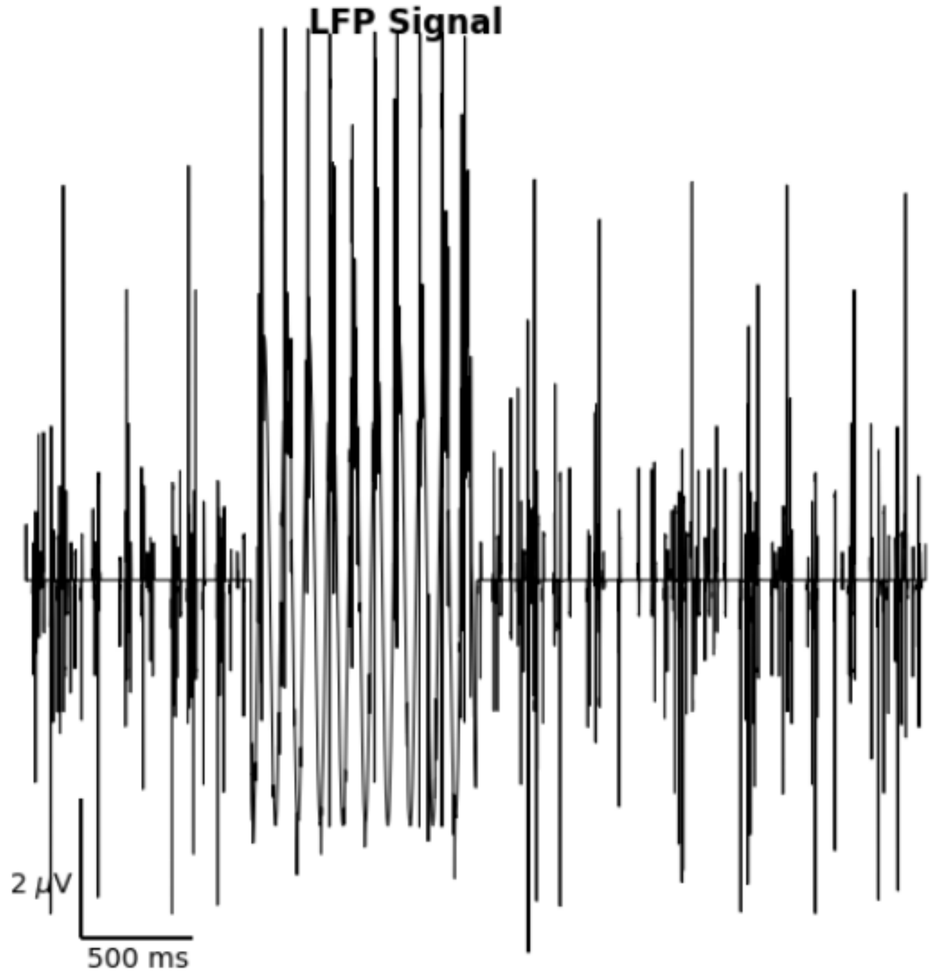

Cell 0, Pop E
V_soma
Time (ms)
20
0
−20
−40
−60
−80
0
500
1000
1500
2000
2500
3000
3500
4000

Results


cells=25 syns/cell=3.4 rate=2.1 Hz
Cells (ordered by gid)
Time (ms)
E
I
0
5
10
15
20
25
0
500
1000
1500
2000
2500
3000
3500
4000


cells=25 syns/cell=3.4 rate=2.9 Hz
Cells (ordered by gid)
Time (ms)
E
I
0
5
10
15
20
25
0
500
1000
1500
2000
2500
3000
3500
4000

Results



Next steps

- Build larger network model incorporating different layers
- Deliver extracellular stimulation to all components of the cell
- Increase complexity of cell morphologies