

Tutorial:

# Modeling Networks with NEST

Hans Ekkehard Plessner

Dept of Data Science, Norwegian University of Life Sciences

Institute of Advanced Simulation (IAS-6), Jülich Research Centre, Germany

LASCON X — 9 Jan 2026



Co-funded by  
the European Union

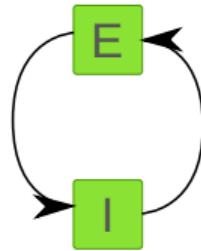
# Acknowledgments

This presentation is based on previous work by many people:

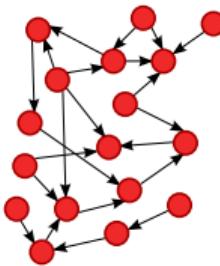
- Hannah Bos
- David Dahmen
- Moritz Deger
- Jochen Martin Eppler
- Espen Hagen
- Abigail Morrison
- Jannis Schuecker
- Johanna Senk
- Tom Tetzlaff
- Sacha van Albada
- Charl Linssen

# When to use NEST?

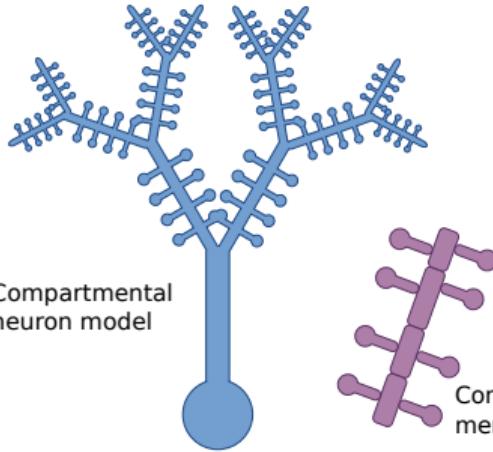
Population model



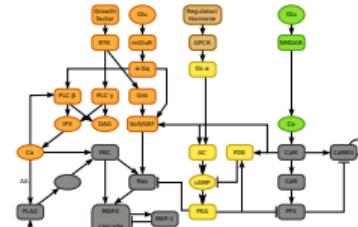
Point neuron network model



Compartmental neuron model



Reaction-diffusion model



Co-funded by  
the European Union

Possibility to simulate large networks

Complexity of single elements



## The State of Affairs in 2008 ...

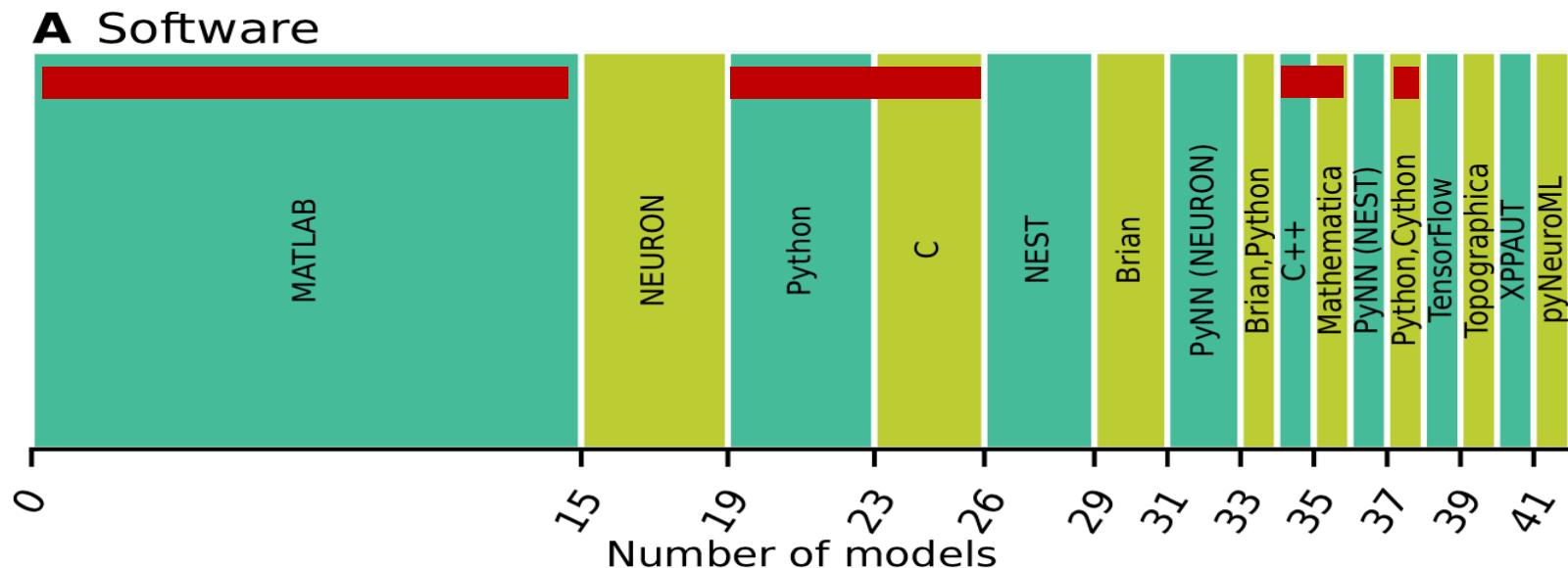
- De Schutter, E. (2008).  
Why are computational neuroscience and systems biology so separate?  
PLoS Computational Biology, 4(5), 78.
- «... many established computational neuroscience software is shackled by legacy code (such as GENESIS [25], at <http://www.genesis-sim.org/>, and NEURON [26], at <http://www.neuron.yale.edu/neuron/>). Even worse, large communities, especially in neural network modeling, still use “homegrown” software that is simply not available electronically.» (de Schutter, 2008)

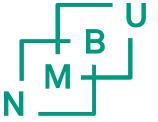
# ... and in 2022



- J. Senk, B. Kriener, M. Djurfeldt, N. Voges, H.-J. Jiang, L. Schüttler, G. Gramelsberger, M. Diesmann, H. E. Plesser, and S. J. v Albada. Connectivity concepts in neuronal network modeling. PLOS Comput Biol, 18:1-49, 2022.
- All models from ModelDB or OpenSourceBrain
- Many models only published in author's Github repository

- See also  
Aimone, J. B., et al (2023). Editorial: Neuroscience, computing, performance, and benchmarks: Why it matters to neuroscience how fast we can compute. Frontiers in Neuroinformatics, 17, 1157418.





# What is the problem with Matlab/Python/C codes?

- Compare

```
z = 0  
if x < y  
    z = x  
else  
    z = y
```

- with

```
z = min(x, y)
```

- Using higher-level concept results in more expressive code

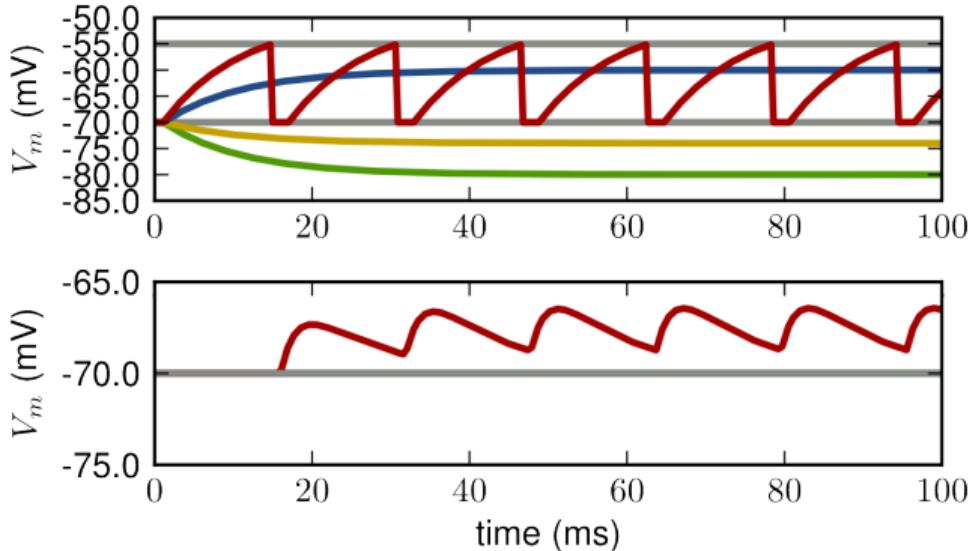
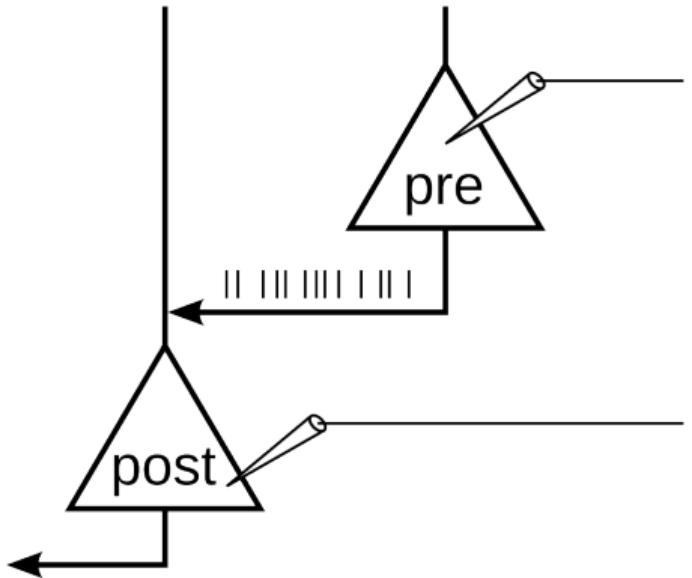
- Compare

```
for i = 1 to len(srcs)  
    for j = 1 to len(tgts)  
        if rnd() < p  
            Connect(srcs[i], tgts[j])
```

- with

```
Connect(srcs, tgts,  
rule='Bernoulli',  
p=0.1)
```

# NEST simulation mimics a neuroscientific experiment



# NEST = NEural Simulation Tool

The logo for NEST, consisting of the word "nest" in a white sans-serif font inside an orange rounded rectangle.

- Simulator for spiking neural networks with a focus on the dynamics, size and structure of neural systems, rather than on the exact morphology of individual neurons
- C++ core, hybrid parallelization (OpenMP+MPI), Python frontend PyNEST
- Same code from laptops to supercomputers → simulation of large-scale models
- Development is driven by scientific needs with a focus on accuracy and flexibility as well as quality assurance

<https://www.nest-simulator.org>

<https://github.com/nest/nest-simulator>

<https://nest-simulator.readthedocs.io>

<https://ebrains.eu/data-tools-services/tools/nest>

# Built-in neuron models

- Integrate-and-fire models (`iaf_*`)
  - Current-based (`iaf_psc_*`)
  - Conductance-based (`iaf_cond_*`)
  - Different post-synaptic shapes (`*_delta`, `*_alpha`, `*_exp`)
  - Width precise spike timing (not on grid, `*_ps`)
- Adaptive exponential integrate-and-fire models (`aeif_*`)
- Allen Institute generalised leaky integrate-and fire family (`gif_*`)
- Hodgkin-Huxley models (`hh_*`)
- Hill-Tononi model ([Hill & Tononi, 2005](#)) (`ht_*`)
- Astrocytes (`astrocyte_lr_1994`, since NEST 3.6)
- Support for eprop (back-propagation through time, since NEST 3.7)
- Compartmental neurons (`cm_*`)
- ... and many more!

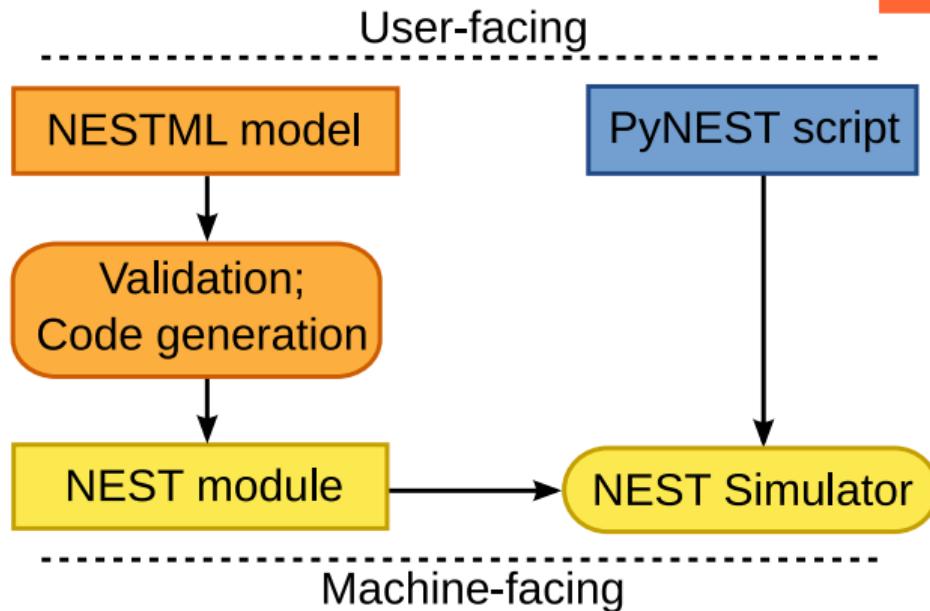
# Built-in synapse models

- Static synapse
- Short term depression and facilitation ([Tsodyks, Uziel, Markram, 2000](#))
- Spike-timing dependent plasticity (STDP)
  - All-to-all and several nearest-neighbour variants
- Triplet STDP ([Pfister & Gerstner, 2006](#))
- Voltage-based STDP ([Clopath et al., 2010](#))
- Dopamine-modulated STDP ([Potjans et al., 2010](#))
- Gap junctions (electrical synapses)
  - ... and many more!

# Custom models with NESTML

A domain-specific language for neuron and synapse models

nest::ml



<https://github.com/nest/nestml>

<https://nestml.readthedocs.io>

# Stimulation devices

## Spike generators

- `spike_train_injector` spikes at prescribed points in time
- `poisson_generator` spikes according to a Poisson distribution
- `sinusiodal_poisson, gamma_generator` spikes according to a sinusiodally modulated Poisson or Gamma distribution

## Current generators

- `dc_generator` provides a constant current
- `ac_generator` provides a sine-shaped current
- `step_current_generator` provides a step-wise constant current
- `noise_generator` provides a random noise current

## Recording devices

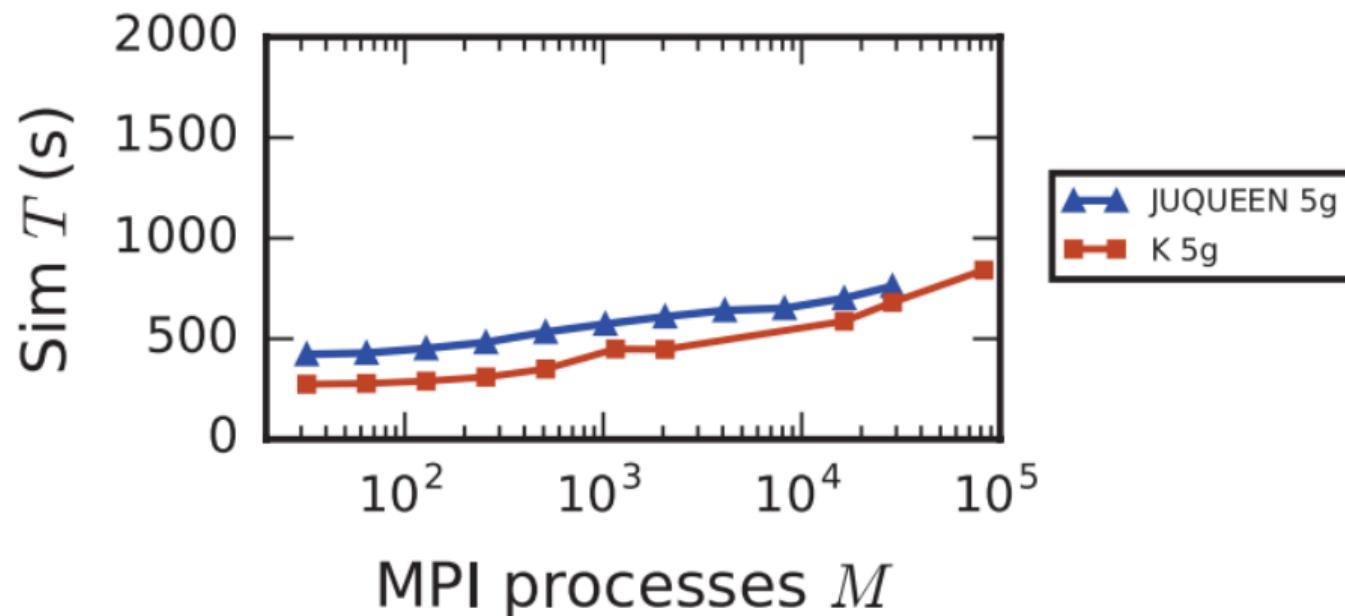
- `spike_recorder` records incoming spikes
- `multimeter` records analog quantities (potentials, conductances, ...)
- `voltmeter` records the membrane potential
- `correlation_detector` records pairwise cross-correlations between the spiking activity of neurons
- `weight_recorder` records the weight of connections

# Parameterization

Parameters can be used to set node status, to create positions in space, and to define connection probabilities, weights, and delays.

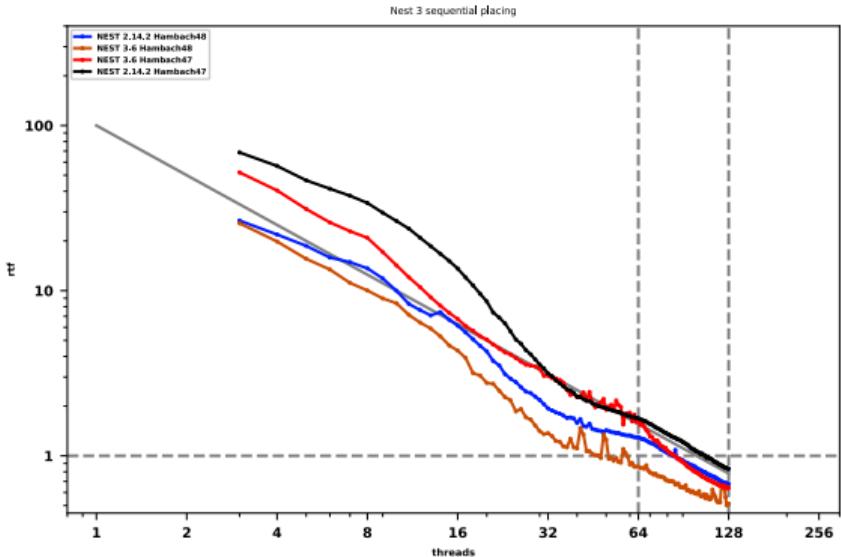
- Random parameters
- Spatial parameters
- Spatially distributed parameters
- Mathematical functions
- Clipping, redrawing, and conditional parameters
- Combination of parameters

## Performance: Weak scaling



Jordan, Ippen, Helias, Kitayama, Sato, Igarashi, Diesmann & Kunkel (2018) Front. Neuroinform.

# Performance: Strong scaling



Potjans-Diesmann Microcircuit ( $\sim 80.000$  neurons) on  $2 \times$  AMD Epyc Rome 64 cores  
(Sebastian Gillesen)

# NEST GPU

- Previously developed as NeuronGPU, recently joined the NEST Initiative
- Ongoing work:  
alignment to the community-centered workflow of NEST:
  - Documentation
  - CMake installation
  - PyNEST interface
  - NESTML
  - Continuous integration
  - Tests
  - Take NEST GPU to JUPITER



<https://github.com/nest/nest-gpu>

Golosio, Tiddia, De Luca, Pastorelli, Simula & Paolucci (2021) *Front. Comput. Neurosci.*

Tiddia, Golosio, Albers, Senk, Simula, Pronold, Fanti, Pastorelli, Paolucci & van Albada (2022)  
*Front. Neuroinform.*

Golosio, Villamar, Tiddia, Pastorelli, Stapmanns, Fanti, Paolucci, Morrisson & Senk (in preparation)

# Large community with over 30 years of experience

SYNOD: An Environment for Neural Systems Simulations

*Language Interface and Tutorial*

Markus Diesmann<sup>1</sup>, Marc-Oliver Gewaltig<sup>2</sup>, and Ad Aertsen<sup>1</sup>



r9 | gewaltig | 1996-06-23 11:26:44 +0200 (Sun, 23 Jun 1996) | 2 lines

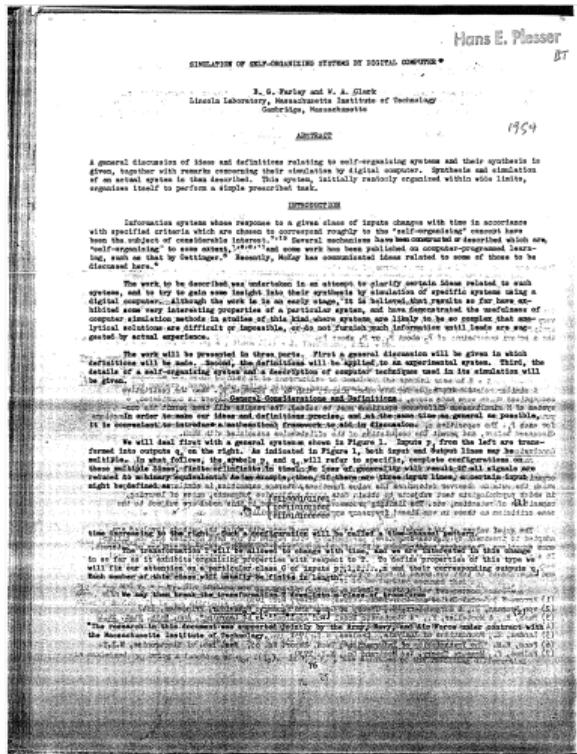
Initial revision

- Online documentation
- User Mailing List
- Open Development on Github
- Bi-weekly developer VC
- Annual virtual conference



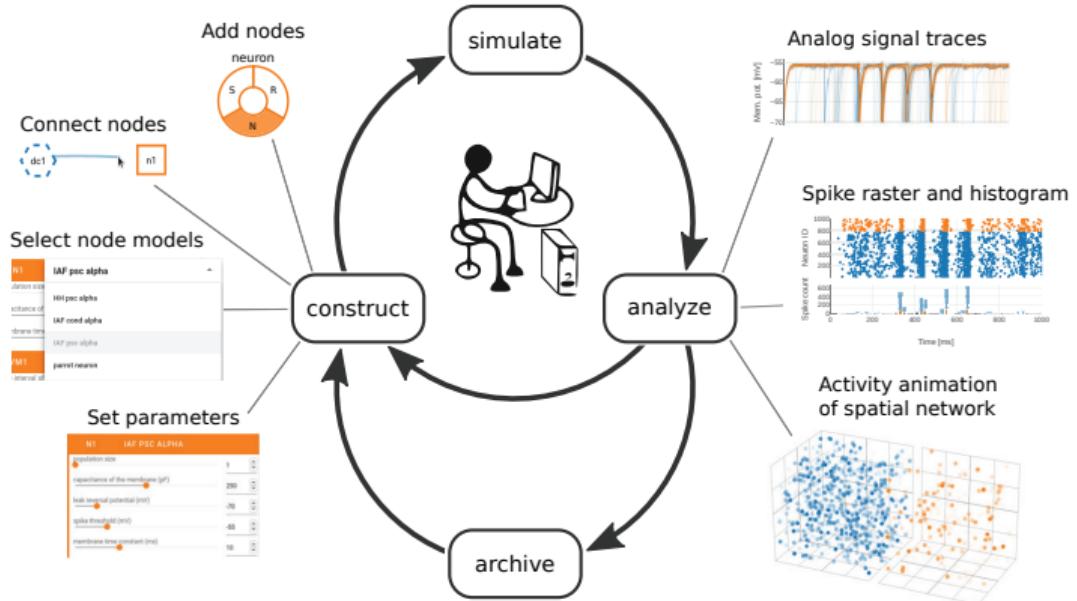
# Ideas for NEST-related projects

- Gallery of NEST Examples
- Papers based on NEST Simulations
- Reproduce Farley & Clark (1954)
- Reproduce Davis *et al* (2021) *Spontaneous traveling waves naturally emerge ...*
- Explore models with NMDA dynamics using NEST's super-fast approximate implementation
- Explore astrocyte dynamics using NEST astrocyte support (talk to Marja-Leena Linne about the biology!)
- Explore spike-based learning using eligibility propagation support in NEST (more on Tuesday)



# NEST Desktop—a GUI for NEST

- Web-based graphical user interface as educational application
- Interactive construction and exploration of neural networks



Spreizer, Senk, Rotter, Diesmann & Weyers (2021) eNeuro

# Working with NEST Desktop

**Documentation** <https://nest-desktop.readthedocs.io>

**On EBRAINS** No installation required

- Register on <https://ebrains.eu/sign-up> (use **institutional** email)
- Use NEST Desktop at <https://ebrains.eu/service/nest-desktop>

**On your computer** Using the VM or the Conda/Mamba setup

- In Terminal, run

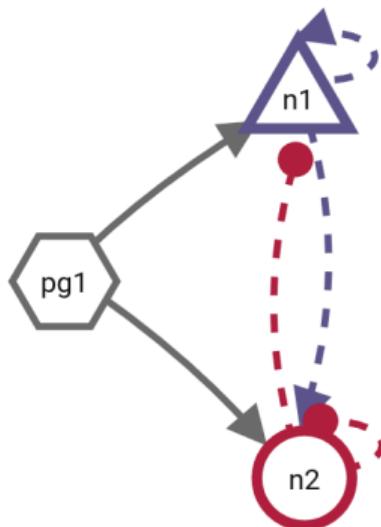
```
export NEST_SERVER_DISABLE_AUTH=1
nest-server start -d
nest-desktop start
```
- Open the URL shown after nest-desktop start in your browser
- To shut down, first use Ctrl-C to stop nest-desktop, then

```
nest-server stop
```
- We explore examples provided with NEST Desktop

# A balanced random network in NEST Desktop

Recurrently connected networks of excitatory and inhibitory neuron populations with Poissonian input

See also `LasconX_NESTDesktop_Brunel.json`.



Try it out without installation in the browser:  
<https://nest-desktop.apps.hbp.eu>

- `iaf_psc_delta`
- $N_E = 800, N_I = 200$
- $C_E = 80, C_I = 20, D = 1.5\text{ms}$
- $g = 5, J_E = J = 1, J_I = -gJ = -5$
- $\nu_{\text{ext}} = 1\,000\text{s}^{-1}$
- $C_m = 1\text{pF}, \tau_m = 20\text{ms}$
- $E_L = V_{\text{reset}} = 0\text{mV}, V_{th} = 20\text{mV}$
- $t_{\text{ref}} = 2\text{ms}$

Brunel (2000) *J Comput Neurosci*



Human Brain Project



EBRAINS

# Thank you

[www.humanbrainproject.eu](http://www.humanbrainproject.eu)

[www.ebrains.eu](http://www.ebrains.eu)



Co-funded by  
the European Union

