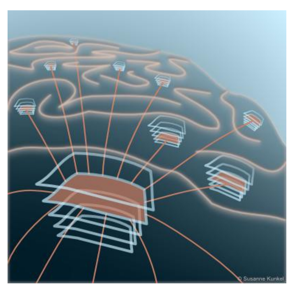
20 years of NEST: a mature brain simulator



Jochen Martin Eppler [1], Susanne Kunkel [2], Moritz Helias [1], Yury V. Zaytsev [2], Hans Ekkehard Plesser [3], Marc-Oliver Gewaltig [4], Abigail Morrison [1,2], Markus Diesmann [1]

[1] Institute of Neuroscience and Medicine (INM-6), Institute for Advanced Simulation (IAS-6), Jülich Research Centre and JARA, Jülich, Germany
[2] Simulation Lab Neuroscience, Jülich Supercomputing Centre, Jülich Research Centre, Jülich, Germany
[3] Department of Mathematical Sciences and Technology, Norwegian University of Life Sciences, Ås, Norway
[4] Blue Brain Project, École polytechnique fédérale de Lausanne EPFL, Lausanne, Switzerland

Contact: j.eppler@fz-juelich.de

Introduction

Efficient and reliable simulation tools are essential for progress in brain research. A wide range of simulators has been developed, each specialized on one or few spatial and temporal scales [1]. But the reliable and reproducible simulation of complex systems such as the brain is a very demanding challenge.

Thus, the Computational Neuroscience community concentrated on a few reliable and widely used simulation tools in recent years. This concentration was not least the result of a series of large-scale EU funded projects, such as FACETS, BrainScaleS and the recently announced Human Brain Project.

The Neural Simulation Tool NEST [2] saw its first incarnation in 1993. By tightly coupling software development with research in Computational Neuroscience, simulator technology evolved steadily and NEST is a powerful simulation tool for brain-scale simulations today.

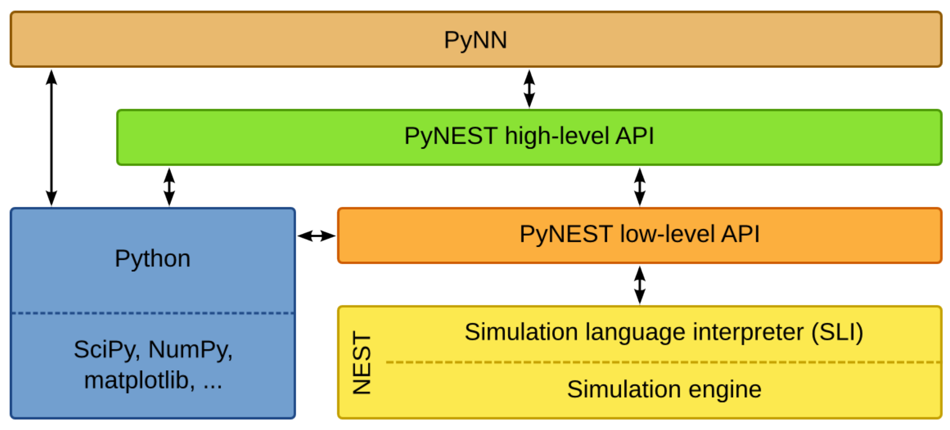
Python based user interfaces

PyNEST is the standard interface to NEST. It allows to specify
- complex stimuli and stimulation patterns
- the organization and connectivity of the network and
- data analysis

in a single programming language [3].

PyNN is a simulator-independent description language and an interface for Python that supports many simulators and controls them through simulator-specific backends [4].

Figure 1: The different software layers, which make up the Python UI subsystem of NEST. Users can access all of the layers from their simulation scripts.



Optimized for simulations on the brain-scale

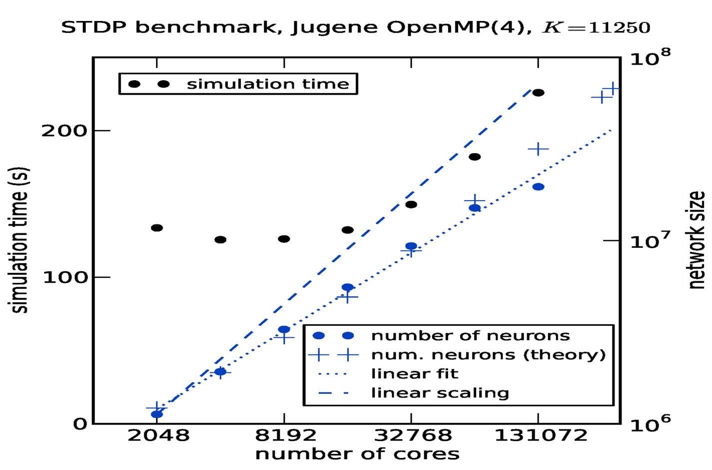


Figure 2: Scaling of NEST on the BlueGene/P JUGENE [7].

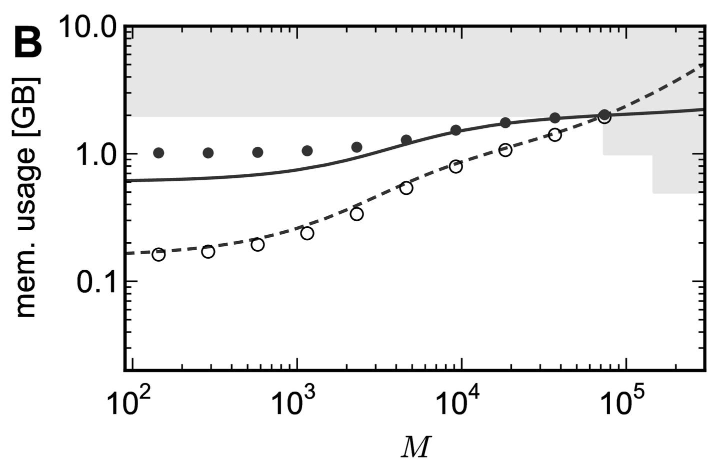


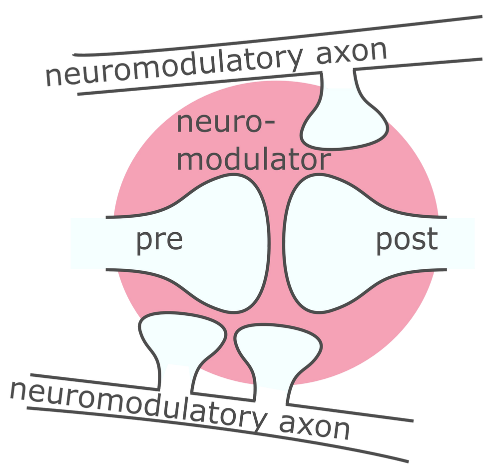
Figure 3: Memory usage of old and new NEST [6].

Efficient parallelization is the prerequisite for using modern multi-core desktop computers and HPC facilities such as clusters and supercomputers. To this end, NEST supports OpenMP and MPI to parallelize both the wiring and the simulation phase [5].

Optimized memory requirements allow to run bigger simulations on the same machine. The memory footprint of NEST was reduced significantly with the help of a memory model that allows to find and later optimize inefficient data structures by comparing their actual memory usage with the expected [6].

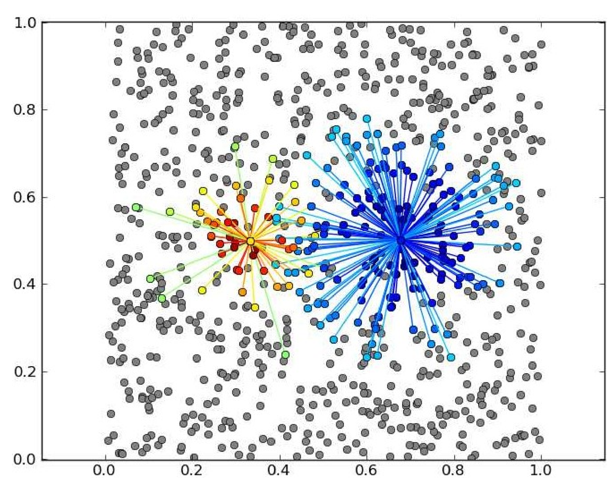
Tools for modern Neuroscience research

Synaptic plasticity in the form of short-term dynamics, spike-timing dependent plasticity (STDP) [8] and neuro-modulation via dopamine [9] allow to study learning and adaptation in neuronal networks.



Precise spike timing enables one to study synchronization in networks without side-effects caused by using a discrete time grid, on which the spikes are forced during the simulation [10].

The topology module and an **interface to the CSA** support the user with the construction of structured networks and spatial connectivity patterns [11, 12].



A MUSIC interface allows to couple NEST to other simulators at runtime and send and receive spikes as well as continuous values to bridge multiple scales, and to bind NEST to ones own program code [13].

Correctness and release stability

A battery of unit tests helps to make sure that all parts of NEST are working as expected both in serial and parallel scenarios. For neuron and synapse models, the tests compare simulated results to the analytical solutions [14].

Continuous integration (CI) techniques allow to build NEST with different compiler and library options upon every check-in and thus increase release stability and correctness [15].

Regular open source releases under the terms of the GPL give the users the chance to use and test the newest technology, while developers get feedback via bug reports and mails on the NEST mailing list. The releases are available on the homepage of the NEST Initiative at **http://www.nest-initiative.org**.

References

[1] Brette et al (2007) Simulation of networks of spiking neurons: A review of tools and strategies.
[2] Gewaltig & Diesmann (2007) NEST (NEural Simulation Tool). Scholarpedia.
[3] Eppler et al. (2008) PyNEST: a convenient interface to the NEST simulator.
[4] Davison et al. (2008) PyNN: a common interface for neuronal network simulators.
[5] Plesser et al. (2007) Efficient par. sim. of large-scale neuronal networks on clusters of multiprocessor computers.
[6] Kunkel (2011) Meeting the memory challenges of brain-scale network simulation.
[7] Helias et al. (2012) Supercomputers ready for use as discovery machines for neuroscience.
[8] Morrison et al. (2007) Spike-time dependent plasticity in balanced recurrent networks.
[9] Potjans et al. (2010) Enabling functional neural circuit sim. with distributed computing of neuromodulated plasticity.
[10] Hanuschkin et al. (2010) A general and efficient method for exact spike times in globally time-driven simulations.
[11] Plesser & Enger (2013) NEST Topology User Manual.
[12] Eppler et al. (2011) Evaluating the Connection-Set Algebra for the neural simulator NEST.
[13] Djurfeldt et al. (2010) Run-time interoperability between neuronal network sim. based on the MUSIC framework.
[14] Eppler et al. (2009) A testsuite for a neural simulation engine.
[15] Zaytsev (2013) Increasing quality and managing complexity in neuroinform. software dev. with cont. integration.

Acknowledgements: The development of NEST has been made possible by the support of many institutions, including Weizmann Institute, Ruhr-Universität Bochum, Universität Freiburg, Honda Research Institute Europe, Max-Planck-Institute for Fluid Dynamics, Norwegian University of Life Sciences, RIKEN Brain Science Institute, Bernstein Center for Computational Neuroscience Freiburg, Helmholtz Gesellschaft and FZ Jülich, EPFL and BlueBrainProject, EU grants FACETS and (FP6-15879) BrainScaleS (FP7-269921), Research Council of Norway grant eNeuro (178892/V30).