

NEST, the Neural Simulation Tool

Ankur Sinha

November 26, 2017

The NEST (Neural Simulation Tool) is one of the many software tools that are used for modelling of brain mechanisms. Over 300 models have now been implemented in NEST since its initial release in 1994 as SYNOD. Since then, NEST has grown via contributions from developers from the world over under the guidance of the core team—the NEST Initiative. The most recent release, version 2.14.0, saw 700 contributions from as many as 30 developers. With a Python API that enables quick prototyping, an active community of users and maintainers, and an Open source development model that encourages collaboration while maintaining development standards, NEST is now considered one of the go-to simulators for modelling spiking neuronal networks.

NEST provides a high level API in the Python programming language that presents a simple, unified interface for researchers with different levels of technical expertise to build their models on. The API allows users access to the more than 50 neuron models and diverse set of synapse models that are built into the simulator. The API also includes efficient methods to inspect and modify the states of simulation entities. Detailed offline documentation and example scripts further supplement the API. In cases where these are insufficient, an active mailing list allows users to communicate with the NEST community. As an added feature, the simulator-independent PyNN framework also supports NEST.

One of the priorities of the development team is to take advantage of modern, parallelised hardware. In NEST, the same simulation code can be run on different hardware configurations without requiring the user to make any changes—from multicore PCs to clusters of connected computing cores. The distribution of computing tasks over the available hardware resources is handled internally by NEST. Indeed, recent studies have used NEST to simulate thousands of neurons and millions of synapses to study near full-scale models of local cortical microcircuits.