PyNN: a unified interface for neuronal network simulators

Andrew Davison[1], Eilif Muller[2], Mike Hull[3], Pierre Yger[1].

[1]Unité de Neuroscience, Information et Complexité, CNRS UPR 3293, Gif-sur-Yvette, France
[3]Blue Brain Project, EPFL, Lausanne, Switzerland
[3]University of Edinburgh, UK

Computational neuroscience has produced a diversity of simulator software for simulations of networks of spiking neurons. This diversity has both positive and negative consequences. The principal problem is that each simulator uses its own programming or configuration language, leading to considerable difficulty in porting models from one simulator to another or even in understanding someone else's code. This impedes communication between investigators and makes it harder to reproduce or build on other people's work. The advantages of having multiple simulators available are (i) each simulator has a different domain of excellence, allowing the most appropriate software to be chosen for a given problem; (ii) simulation results can be cross-checked between different simulators, giving greater confidence in their correctness.

The Python package PyNN provides a unified API across neural simulators (currently NEURON, NEST, PCSIM, Brian and a neuromorphic VLSI hardware implementation, with support for MOOSE and GENESIS 3 under development), making it possible to write a simulation script once and run it without modification on any supported simulator or hardware platform.Thus we keep the advantages of having multiple simulators (for cross-validation, etc) but remove the translation barrier.

Here we present recent developments within PyNN, notably improved support for higher-level network structures, support for the NineML and NeuroML languages for describing neuronal and synapse models, and support for the Connection Set Algebra for describing network connectivity.

PyNN is open-source software and is available from http://neuralensemble.org/PyNN.