BMC Neuroscience

BioMed Central

Poster presentation

NeuroCAD – the modular simulation environment for effective biologically plausible neuromodeling

Ruben A Tikidji-Hamburyan*[1] and Sergey N Markin[1,2]

Address: [1]A.B. Kogan Research Institute for Neurocybernetics, Southern Federal University, Rostov – on – Don, Russia and [2]Drexel University College of Medicine, Philadelphia, PA, USA

Email: Ruben A Tikidji-Hamburyan* - rth@nisms.krinc.ru

* Corresponding author

from Seventeenth Annual Computational Neuroscience Meeting: CNS*2008
Portland, OR, USA. 19–24 July 2008

Published: 11 July 2008

BMC Neuroscience 2008, **9**(Suppl 1):P91 doi:10.1186/1471-2202-9-S1-P91

This abstract is available from: http://www.biomedcentral.com/1471-2202/9/S1/P91

Simulation of biologically plausible neural models from a single cell to networks usually requires substantial computational resources for numerical solution of differential equations. Various advanced methods to reduce the computational cost but keep an accuracy and effectiveness of solving differential equations are offered and implemented in several program environments. However efficacy of these methods is often reduced by non effective programming paradigms.

Here we describe the concept of NeuroCAD, the modular program environment especially designed for biologically plausible neural simulations. The main goal while implementing NeuroCAD was a combination of flexibility of script interpreter and effectiveness of highly optimized and monolithically compiled application in one simulation environment.

Applying the modern modular concepts and well known method of connections through shared common data allows removing all computational procedures from the NeuroCAD Engine to the independent dynamically linked libraries, which we call modules. Thus all model primitives such as ion channels, compartments or cells, synapses, external voltage or current clamps and so on should be implemented in the NeuroCAD environment as modules. All modules are stored in several data bases. Each data base corresponds to a level of its accessibility.

Before simulation, the environment component called NeuroCAD Node imports the C-code of required modules from corresponding data bases, locally compiles the code and prepares a work space for simulation. The next component, the NeuroCAD Engine, dynamically links all prepared modules and connects them through allocated common memory according to desired model. During simulation NeuroCAD does not perform any computation; indeed the NeuroCAD Engine only calls calculation procedures in each loaded module, and thereby maximizes effectiveness of computation.

Because of independence of modules, users may connect them without any limitation, which provides a possibility to flexibly build models. In the future, when the data base of modules will grow, the necessity to write modules in C-code will be minimized.

Mathematical problems which may be solved by the NeuroCAD environment correspond to those in the dynamical systems field. They could be time and event based models, with static or adaptive time steps. Because the history of dynamical variables is buffered by the NeuroCAD Engine there are simple ways to implement implicit and explicit methods to solve differential equations, such as iterative methods, predictor-corrector methods and so on. Thus NeuroCAD may be used in the numerical simulation of models from the level of active membrane models through multicompartmental models up to large neural networks with phenomenological single elements.