Ivan Raikov

Curriculum Vitae

Education

2008-present **Ph.D. candidate, Biomedical Sciences** , *University of Antwerp, Belgium*.
- (Advisor: Erik De Schutter)

2004-2006 **M.S. in Computer Science**, *Georgia Institute of Technology, USA*.

1998-2004 **B.S. in Computer Science**, *Georgia Institute of Technology, USA*.

Career

2007-present **Research technician**, *Okinawa Institute of Science and Technology (OIST)*.

2001-2007 **Research assistant**, *Georgia Tech School of Electrical and Computer Engineering*.

1999-2001 **Cooperative education student**, *Vertisoft Corp*.

Relevant specialized knowledge and training

Programming languages

- Well-versed in C, Standard ML, Scheme, Matlab/Octave programming languages.
- Working knowledge in C++, Python, Ocaml, Yacc, Lex, and the small languages native to Unix.
- Familiarity with Java, Haskell, Lisp, Pascal, Fortran and most open-source scripting languages.

Algorithms and data structures

- Extensive experience with common numerical methods for solving systems of ordinary differential equations and linear algebra methods.
- Familiarity with algorithms and data structures for efficient processing of heterogeneous data, such as search trees, spatial data structures, graphs.

Software engineering

- Experience with building parallel code based on MPI.
- Systems-level programming experience with Linux and RTAI.

List of Publications

Journal articles

1. The Layer-Oriented Approach to Declarative Languages for Biological Modeling. I. Raikov and E. De Schutter, PLoS Computational Biology, 2012.
2. Custom-made multiphoton microscope for long-term imaging of neuronal cultures to explore structural and functional plasticity, K. Rambani, M. Booth, E. Brown, I. Raikov, S. Potter, Multiphoton Microscopy in the Biomedical Sciences V. Proceedings of the SPIE, 5700:102-108 (2005).
3. MRCI: A flexible real-time dynamic clamp system for electrophysiology experiments I. Raikov, A. J. Preyer and R. J. Butera, Journal of Neuroscience Methods. 132:109-123 (2004).

Onna 7168, Onna-son – Okinawa 904-0411, JAPAN
✉ *ivan.g.raikov@gmail.com*

Conference articles

Numerical Model Compiler: A Design for Generating Real-Time Numerical Simulation Code I. Raikov and R. J. Butera, Eighth Real-Time Linux Workshop. Lanzhou University - SISE, 2006.

Editorials

The Promise and Shortcomings of XML as an Interchange Format for Computational Models of Biology. I. Raikov and E. De Schutter, Neuroinformatics 10, 2012.

Conferences and Workshops

1. Twenty-second Annual Computational Neuroscience Meeting, July 2013 (Paris, France). Poster: Challenges of declarative modeling of conductance-based neurons in diverse simulation environments.
2. Twenty-first Annual Computational Neuroscience Meeting, July 2012 (Atlanta, Georgia). Poster: Exploring the functional implications of brain architecture and connectivity: a declarative language framework.
3. Meeting of the INCF Task Force for Multiscale Modeling, November 2011. (Paris, France). Presentation: NineML Native Interpreter Architecture.
4. 2nd COMBINE Meeting, September 2011 (Heidelberg, Germany). Presentation: An introduction to NineML.
5. Twentieth Annual Computational Neuroscience Meeting, July 2011 (Stockholm, Sweden). Poster: NineML: A Description Language for Spiking Neuronal Network Modeling.
6. Twentieth Annual Computational Neuroscience Meeting, July 2011 (Stockholm, Sweden). Workshop: Emerging standards for network modeling in neuroscience. Presentation: The NineML Native Interpreter: A Workflow for Network Model Construction.
7. Meeting of INCF Task Force for Multiscale Modeling, May 30 âĂŞ June 1, 2011. (Stockholm, Sweden) Presentation: NineML Abstraction Layer Architecture and Object Model.
8. Meeting of the INCF Task Force for Multiscale Modeling, January 2011. (Stockholm, Sweden). Presentation title: NineML Abstraction Layer Architecture and Object Model.
9. Neuroinformatics 2010 (Kobe, Japan). Poster presentation. Poster: NineML: A Description Language for Spriking Neuronal Network Modeling: The Abstraction Layer.
10. Nineteenth Annual Computational Neuroscience Meeting, July 2010 (San Antonio, USA). Poster: NineML: A Description Language for Spriking Neuronal Network Modeling: The Abstraction Layer.
11. Nineteenth Annual Computational Neuroscience Meeting, July 2010 (San Antonio, USA). Workshop on Methods in Neuroinformatics. Presentation: NineML: The Abstraction Layer.
12. 7th FENS Forum of European Neuroscience, July 2010 (Amsterdam, Netherlands). Poster: NineML: A Description Language for Spriking Neuronal Network Modeling: The Abstraction Layer.
13. Meeting of the INCF Task Force for Multiscale Modeling, June 2010. (Stockholm, Sweden). Presentation: Developments in the NineML Abstraction Layer.
14. Meeting of the INCF Task Force for Multiscale Modeling, February 2010. (Stockholm, Sweden). Presentation: NineML Abstraction Layer: A Proposal.
15. Society for Neuroscience 2009 (Chicago, USA). Poster: Design of hybrid neuroscience modeling languages.
16. Meeting of the INCF Task Force for Multiscale Modeling, November 2009. (Antwerp, Belgium). Presentation: Entities and Abstraction Layer in the Neural Network Model Description Language.
17. Computational Neuroscience Meeting, 2009 (Berlin, Germany). Poster: The Layer Oriented Approach to Modeling Language Design.
18. INCF Task Force meeting on Large Scale Modeling of the Nervous System, March 2009. (Tokyo, Japan). Presentation: How to design a science modeling language?
19. Computational Neuroscience Meeting, 2008 (Portland, USA). Workshop on interoperability of neuroscience software. Presentation title: Neuroscience modeling languages: practice and theory.

Neuroscience Software Projects

NineML, 2009-present

NineML (`http://software.incf.org/nineml`) is a declarative computer language for describing large-scale networks of integrate-and-fire neurons. The project was initiated by the International Neuroinformatics Coordinating Facility. Contributions to the development of NineML include the design of

the notation for describing hybrid dynamics and a graph library. Implementation work includes a NineML interpreter capable of generating executable code for GNU Octave, Scheme, Standard ML programming languages.

NEMO, 2008-present

NEMO (`http://wiki.call-cc.org/nemo`) is an implementation of a layer oriented language for describing computational models of neuronal ionic currents. The input language is equation oriented and supports arbitrary rate laws and reaction kinetics. NEMO is capable of generating code for NEURON, NEST and MATLAB/Octave.

Neurolucida conversion utility, 2011-present

The Neurolucida conversion utility (`http://wiki.call-cc.org/neurolucida`) is a program to read Neurolucida XML files and convert them to SWC format and other representations.

Neuromorpho data mining utility, 2009-present

The Neuromorpho data mining utility (`http://wiki.call-cc.org/neuromorpho`) is a program to query the Neuromorpho database and extract models according to given field values.

Numerical Model Compiler (NMC), 2005-2007

NMC (`https://sourceforge.net/projects/numc`) aims to become a framework for generation of real-time numerical simulations of dynamical systems from differential and difference equations. It consists of a general-purpose compiler core, and a frontend framework that can be used to implement different equation-oriented domain-specific languages.

Real-Time Experiment Interface (RTXI), 2004-2007

RTXI (`http://www.rtxi.org`) is a collaborative open-source software development project aimed at producing a real-time Linux based software system for hard real-time data acquisition and control applications in biological research. One of the goals of RTXI is to merge three existing open-source real-time experiment control systems: RTLab (Cornell University), RTLDC (Boston University), and MRCI.

Model Reference Current Injection (MRCI), 2001-2007

MRCI (`http://www.neuro.gatech.edu/mrci` is a real-time platform that implements the dynamic clamp protocol. The basic idea behind this technique is to simulate ionic currents in real-time, which can be used to:

- artificially insert ion channels into a neuron
- connect in vitro neurons with simulated synapses
- connect simulated neurons to in vitro neurons

Other Software Projects

Chicken Scheme, 2005-present

Chicken Scheme (`http://www.chicken-scheme.org`) is a compiler and interpreter for the Scheme programming language that compiles Scheme code to standard C. It is based on innovative ideas from the computer science research community, and offers a vast number of libraries. My contributions to this project have included over 80 libraries related to algorithms and methods for scientific computing, as well as managing several recent releases of the core Chicken distribution.

Professional activities

- Manuscript reviewer, Neuroinformatics, 2012-2013

References

- Dr. Robert J Butera, Associate Professor Bioengineering and Computer Engineering. Phone: +1-404-894-2935 Email: robert.butera@ece.gatech.edu
- Dr. David Christini, Associate Professor of Physiology and Biophysics. Phone: 212-746-6280 Email: dchristi@med.cornell.edu

Onna 7168, Onna-son – Okinawa 904-0411, JAPAN
✉ *ivan.g.raikov@gmail.com*

- Dr. Erik De Schutter, Professor. Email available upon request

Onna 7168, Onna-son – Okinawa 904-0411, JAPAN
✉ *ivan.g.raikov@gmail.com*