Managing Complexity in Multi-Algorithm, Multi-Scale Biological Simulations: An Integrated Software Engineering and Neuroinformatics Approach.

Giovanni Idili[1], Matteo Cantarelli[2], Marius Buibas[3], Tim Busbice[4],Jay Coggan[5], Christian Grove[6], Sergey Khayrulin[7], Andrey Palyanov[7], Stephen D. Larson[8]

1 Department of Computing, Cork Institute of Technology, Cork, Ireland, 2 Department of Electric and Electronic Engineering, Università di Cagliari, Cagliari, Italy 3 Department of Bioengineering, University of California, San Diego, La Jolla, CA, 4 InterIntelligence Research, Los Angeles, CA, 5 Salk Institute, La Jolla, CA, 6 WormBase, California Institute of Technology, Pasadena CA, 7 A.P. Ershov Institute of Informatics Systems SB RAS, Lab. of Complex Systems Simulation, Novosibirsk, Russia, 8 Whole Brain Project, University of California, San Diego, La Jolla, CA.

Computational biology is asserting itself as an important approach to understanding complex biological systems. In this context, there are several domain-specific simulation engines which only perform calculations within a limited scope such as conductance-based membrane modeling (Brette et al. 2007), biochemical reactions (Blackwell 2005), and biomechanics (Fang-Yen et al., 2010). At present, we are witnessing the first attempts to simulate entire organisms and their surrounding environments, encompassing molecular, cellular and biomechanical aspects of the simulation (Kitano et al., 1998; Boyle 2009). This comes with a burden of complexity that emerges from the need to handle interactions of multi-scale and multi-algorithm simulations in computation-intensive scenarios.

In order to be able to effectively manage the complexity that comes with integrating and maintaining coarse-grained architectures, tools, digital information artifacts and code-bases, it is important for computational biology to fully embrace software engineering methodologies and best practices and follow the lead of the simulation based research in the physical sciences. Taking cues from pioneering projects in computational neuroscience that are addressing this problem (MOOSE, http://j.mp/gSZZNF, Clones; http://j.mp/gzC5CP), we describe our approach to the integration of close-to-the-metal massively parallel simulations with high-level abstractions through the use of design patterns, including emerging paradigms for GPU-based parallel programming. The approach presented here is currently being explored as part of the OpenWorm project (full scale *C. elegans* simulation; http://openworm.googlecode.com), which is presented as a case study.

References

Blackwell, K. T. (2005). A new era in computational neuroscience. *Neuroinformatics*, *3*(2), 163–166. Springer. doi: 10.1385/NI.

Boyle, J. H. (2009). C . elegans locomotion : an integrated approach. Retrieved from http://j.mp/fqKPEw.

Brette, R., Rudolph, M., Carnevale, T., Hines, M., Beeman, D., Bower, J. M., et al. (2007). Simulation of networks of spiking neurons: a review of tools and strategies. *Journal of computational neuroscience*, *23*(3), 349-98. doi: 10.1007/s10827-007-0038-6.

Fang-Yen, C., Wyart, M., Xie, J., Kawai, R., Kodger, T., Chen, S., et al. (2010). Biomechanical analysis of gait adaptation in the nematode Caenorhabditis elegans. *Proceedings of the National Academy of Sciences of the United States of America*, 2-7. doi: 10.1073/pnas.1003016107.

Hines, M. L., & Carnevale, N. T. (1997). The NEURON simulation environment. *Neural computation*, *9*(6), 1179-209. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/9248061.

Kitano, H., Hamahashi, S., & Luke, S. (1998). The perfect C. elegans project: an initial report. *Artificial life*, *4*(2), 141-56. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/9847421.