SPARK: A Framework for Multi-Scale Agent-Based Biomedical Modeling

Alexey Solovyev, University of Pittsburgh, USA
Maxim Mikheev, University of Pittsburgh, USA
Leming Zhou, University of Pittsburgh, USA
Joyeeta Dutta-Moscato, University of Pittsburgh, USA
Cordelia Ziraldo, University of Pittsburgh, USA
Gary An, University of Chicago and University of Pittsburgh, USA
Yoram Vodovotz, University of Pittsburgh, USA
Qi Mi, University of Pittsburgh, USA

ABSTRACT

Multi-scale modeling of complex biological systems remains a central challenge in the systems biology community. A method of dynamic knowledge representation known as agent-based modeling enables the study of higher level behavior emerging from discrete events performed by individual components. With the advancement of computer technology, agent-based modeling has emerged as an innovative technique to model the complexities of systems biology. In this work, the authors describe SPARK (Simple Platform for Agent-based Representation of Knowledge), a framework for agent-based modeling specifically designed for systems-level biomedical model development. SPARK is a stand-alone application written in Java. It provides a user-friendly interface, and a simple programming language for developing Agent-Based Models (ABMs). SPARK has the following features specialized for modeling biomedical systems: 1) continuous space that can simulate real physical space; 2) flexible agent size and shape that can represent the relative proportions of various cell types; 3) multiple spaces that can concurrently simulate and visualize multiple scales in biomedical models; 4) a convenient graphical user interface. Existing ABMs of diabetic foot ulcers and acute inflammation were implemented in SPARK. Models of identical complexity were run in both NetLogo and SPARK; the SPARK-based models ran two to three times faster.

Keywords: Agent-Based, Computer Simulation, Framework, Models, SPARK

DOI: 10.4018/jats.2010070102
BACKGROUND

Computational models in systems biology are intended to describe biological phenomena at various scales (Kitano, 2002). However, the ability to transcend multiple scales of biological phenomenon within a single, cohesive computational model remains a significant challenge to the biomedical modeling community. This capacity is particularly important for the goal of translating underlying mechanistic knowledge to the level of clinical relevance. Recently, translational systems biology has been introduced as a means of utilizing dynamic mathematical models and engineering principles to aid in the optimization of clinical practice (Vodovotz, 2008; An, 2008).

Traditional mathematical models in systems biology are built using statistics or differential equations. These models are best suited for circumstances in which the dimensions of the modeled biological problems are few. However, for most biological systems with a high degree of complexity, the models themselves quickly become intractable in terms of both analysis and computation. Agent-based modeling is an alternative technique with which to model complex biological systems. This type of modeling incorporates an object-oriented, rule-based, discrete event method of model construction (An, 2001; An, 2009; Bankes, 2002; Bonabeau, 2002; Grimm, 2005).

Earlier implementations of ABM-building software were geared towards developing models in the social sciences, such as Ascape (Inchiosa, 2002) and Repast (North, 2006), or towards general-purpose discrete-event simulations, such as MASON (Luke, 2003) and NetLogo (Wilensky, 1999). Among these, NetLogo is currently one of the most popular, particularly for nonformally-trained programmers, due to its user-friendly interface and the natural language-like syntax of its Logo-based programming language. These features greatly simplify the programming of ABMs for novice programmers. Many biomedical models have been developed successfully by using NetLogo (Mi, 2007; Li, 2008; An, 2004; Bailey, 2009).

However, despite its utility, we believe that the construction of biomedical ABMs would benefit from some capabilities currently not found in NetLogo and similar software. These features include the ability to vary agent size, to employ continuous model space, to organize code into modules that can map to biological processes, as well as offering the potential for parallelization in distributed computer architectures.

These criteria motivated the development of a new agent-based modeling platform — SPARK (Simple Platform for Agent-based Representation of Knowledge). This modeling platform incorporates a number of features currently offered by NetLogo, and offers several features designed to facilitate biological modeling. In SPARK, modelers can build models using a user-friendly language and graphical user interface. In addition, the software allows for agents of various sizes, sophisticated image effects, and facilitates multiscale modeling. We describe these features of SPARK in detail below.

IMPLEMENTATION

SPARK is implemented in the Java programming language. SPARK code can run on all machines with Java Standard Edition runtime environment version 1.5 or 1.6. The SPARK source code is freely available under the MIT license and can be retrieved from the SPARK repository at http://code.google.com/p/spark-abm/. The compiled SPARK packages, along with the tutorials, can be downloaded from the official SPARK website (www.pitt.edu/~cirm/spark).

There are several third-party libraries used in SPARK:

1) JFreeChart (http://www.jfree.org/jfreechart/)
2) Java OpenGL (JOGL, https://jogl.dev.java.net/)
3) Colt (http://acs.lbl.gov/~hoschek/colt/)
Two Informational Complexity Measures in Social Networks and Agent Communities
[www.igi-global.com/chapter/two-informational-complexity-measures-social/49355?camid=4v1a](www.igi-global.com/chapter/two-informational-complexity-measures-social/49355?camid=4v1a)

MAS-based Agent Societies by Means of Scout Movement
[www.igi-global.com/article/mas-based-agent-societies-means/72720?camid=4v1a](www.igi-global.com/article/mas-based-agent-societies-means/72720?camid=4v1a)