Systems biology

Cyto-Sim: a formal language model and stochastic simulator of membrane-enclosed biochemical processes

Sean Sedwards¹ and Tommaso Mazza¹,²,*

¹Microsoft Research – University of Trento Centre for Computational & Systems Biology and
²Department of Experimental & Clinical Medicine, Magna Graecia University of Catanzaro, Italy

Received on June 8, 2007; revised on July 25, 2007; accepted on August 9, 2007

ABSTRACT

Motivation: Compartments and membranes are the basis of cell topology and more than 30% of the human genome codes for membrane proteins. While it is possible to represent compartments and membrane proteins in a nominal way with many mathematical formalisms used in systems biology, few, if any, explicitly model the topology of the membranes themselves. Discrete stochastic simulation potentially offers the most accurate representation of cell dynamics. Since the details of every molecular interaction in a pathway are often not known, the relationship between chemical species in not necessarily best described at the lowest level, i.e. by mass action. Simulation is a form of computer-aided analysis, relying on human interpretation to derive meaning. To improve efficiency and gain meaning in an automatic way, it is necessary to have a formalism based on a model which has decidable properties.

Results: We present Cyto-Sim, a stochastic simulator of membrane-enclosed hierarchies of biochemical processes, where the membranes comprise an inner, outer and integral layer. The underlying model is based on formal language theory and has been shown to have decidable properties (Cavaliere and Sedwards, 2006), allowing formal analysis in addition to simulation. The simulator provides variable levels of abstraction via arbitrary chemical kinetics which link to ordinary differential equations. In addition to its compact native syntax, Cyto-Sim currently supports models described as Petri nets, can import all versions of SBML and can export SBML and MATLAB® m-files.

Availability: Cyto-Sim is available free, either as an applet or a stand-alone Java program via the web page (http://www.cosbi.eu/Rpty_Soft_CytoSim.php). Other versions can be made available upon request.

Contact: t.mazza@unicz.it

1 INTRODUCTION

The function of membranes in cells is fundamental to their activity, separating them from other cells to permit differentiation of function and separating organelles within cells for similar purposes. Membrane proteins regulate the communication between the membrane-enclosed compartments and play a statistically important role in cell activity: more than 30% of the human genome codes for membrane proteins. Since membrane proteins control the entry of substances to cells, it is no surprise that in 2000 almost 50% of the drugs prescribed in the USA targeted one class of membrane proteins alone (GPCRs).

In computational terms, compartments correspond to scope, that is, regions where calculations can be performed in a local context. In computer programs, scope is the basis of functions, which are a means to perform the same calculation on different data without creating new code. Biology has a finite repertoire of molecules to perform its calculations and uses compartments to increase computational power in the same way, i.e. by having a multiplicity of membrane-enclosed cells containing membrane-enclosed organelles.

Our aim is to create predictive or otherwise useful models of biological systems, with our current focus on inter- and intracellular pathways. Recognizing the important roles of membrane proteins and biological scope, we require an intuitive model that allows us to simultaneously represent different (types of) cells which communicate via ligands and receptors. In addition to the obvious biological analogues, membranes can also be used to model notional compartments in order to represent, for example, diffusion or localized behaviour. Further, compartments can be used like assay plate wells, enabling several experiments to be run simultaneously and efficiently.

2 APPROACH

Our simulator model is based on that of P systems (http://psystems.disco.unimib.it/) (a.k.a Membrane Systems), a computational paradigm based on compartments, which draws its inspiration from biological cells: multisets of objects are enclosed in a nested hierarchy of compartments and acted upon by local rewriting rules. This translates to: populations of molecules in a nested hierarchy of cellular compartments acted upon by local chemical reactions. We have extended the basic Membrane Systems model to include peripheral and integral membrane proteins (Cavaliere and Sedwards, 2006) and incorporate an efficient Monte Carlo algorithm to simulate the
time evolution of the system. The model includes standard, ‘chemical’ rewriting rules and rules which allow the passage of objects between compartments, mediated by membrane objects (membrane proteins). We have shown in Cavaliere and Sedwards, (2006) that useful properties (e.g., reachability of configurations) of the general model are decidable, hence it is possible to export specific models and investigate them using model checking techniques [e.g. via SBML to PRISM (http://www.prismmodelchecker.org/)] or analyze them using the techniques of symbolic dynamics (Lind and Marcus, 2005).

Our default level of abstraction is chemistry: objects interact governed by stoichiometric rules at a rate defined by mass action kinetics. A variable level of abstraction is facilitated by the use of arbitrary kinetic laws: objects are consumed and produced at rates defined by arbitrary functions of reactants. This facility is a necessity to represent Michaelis–Menten enzyme kinetics, for example. In general, complex behaviour involving many reactions can be reduced to a single rule and an appropriate kinetic function. This allows us to model the diffusion of substances through membranes without the computational burden of spatial simulation.

The simulator native language aims to be intuitive and uncluttered (Cavaliere and Sedwards, 2006). Objects, rules and compartments are defined and then (a subset of) these are composed to create the final system to simulate. In this way it is not necessary to explicitly define rules and objects for each compartment, as is necessary, e.g. in SBML. This compositional approach leads to a compact description of the model.

Petri nets are a popular graphical representation of logical flows, which have been successfully applied to biology [e.g., (Nutsch et al., 2005)]. Hence, in addition to the native rule syntax, Cyto-Sim supports rule definitions in the form of Petri net incidence matrices. By default, the transitions adopt mass action kinetics, i.e., the rate of the transition is proportional to a constant multiplied by the product of the numbers of tokens in the incoming places. Other Petri net dynamics are also possible by explicitly defining appropriate kinetic laws, in the same manner as the native syntax.

3 METHODS

The software is written in J#, which is part of the .NET framework. This choice allows easy porting to both Java and C# and hence maximizes cross-platform compatibility. There are currently two versions available on the internet (http://www.cosbi.eu/Rpty_Soft_CytoSim.php/); an applet and a stand-alone version, both implemented in Java and having graphical user interfaces. Other versions can be made available upon request to the authors.

The core simulation engine is an efficient implementation of a Markov chain Monte Carlo algorithm, using ideas from (Gillespie, 1977). Performance is optimized for mass action chemical kinetics, however the algorithm also supports chemical reactions with arbitrary kinetic laws based on functions of the reactants. This is achieved in an efficient way, using compiler technology and a virtual machine, in such a way as not to adversely affect the performance of the default kinetics.

4 DISCUSSION

Much of the modelling in systems biology is done in the framework of deterministic differential equations, which are usually solved by numerical methods. Such solutions might more accurately be described as deterministic simulations. Recognizing that molecular interactions are discrete stochastic events and that this stochasticity has a significant effect on the behaviour of models which have neutral or unstable manifolds (Gillespie, 1977), discrete stochastic simulations are now often used to give a more accurate representation of such systems. The principal qualitative difference between the two approaches is that, for a given set of simulation parameters, a deterministic simulation will have a unique, average, solution, whereas a single stochastic simulation is just one possible trajectory through the solution space. By using Monte Carlo techniques, the stochastic trajectory is guaranteed to be statistically consistent with the probability distribution describing the system, however this does not guarantee that a particular behaviour will be observed in any given simulation.

Hence, a deterministic simulation gives an average characterization of the system but does not show the variance, whereas a stochastic simulation is only guaranteed to display all the behaviour in the limit of simulations. Given that even some robust behaviour may not be observed in a deterministic simulation, e.g. behaviour that relies on a stochastic divergence at a bifurcation point, performing many stochastic simulations is usually preferable and is a necessity if meaningful inference is to be drawn. The efficiency of the Cyto-Sim algorithm, which can outperform the deterministic simulation of some models (particularly those with many reactions and low molecular concentrations, see Table 1) makes such statistical analysis feasible. Models can also be exported from Cyto-Sim as sets of differential equations in the form of MATLAB® m-files. Having performed initial stochastic simulations and discrete analysis, it is thus possible to perform further algebraic or numerical analysis in the continuous domain (e.g., flux analysis, sensitivity analysis, bifurcation analysis, etc.).

5 CONCLUSION

We have developed a theoretical model and implemented a tool to provide stochastic simulation of biological systems in compartments. Working at mixed levels of abstraction,
we can simulate micro- and macroscopic biochemical processes using arbitrary kinetic laws. Our use of hierarchical compartments has wide application in biological modelling, while our three layered membrane is specifically aimed at molecular cell biology. The explicit modelling of compartments and membranes facilitates the construction of models by composing instances of the variously defined parts, making them compact and intuitive. With a common root in Markov theory, the simulator accepts models defined with combinations of stoichiometric-like rules, Petri net matrices and kinetic functions. Analysis can thus be performed on the models in the frameworks of formal language and automata theory (e.g. symbolic dynamics), (probabilistic) model checking, Petri nets, Markov theory and differential equations, as appropriate.

REFERENCES

P Systems web page, http://pstyles.disco.unimib.it/
PRISM web page, http://www.prismodelchecker.org/