UNIVERSITY
OF TRENTO

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

38050 Povo – Trento (Italy), Via Sommarive 14
http://www.dit.unitn.it

ENHANCED OPERATIONAL SEMANTICS IN SYSTEMS BIOLOGY

Pierpaolo Degano and Corrado Priami

August 2003

Technical Report # DIT-03-046

Also: in proceedings of CMSB 2003

Enhanced Operational Semantics in Systems Biology. ⋆

Pierpaolo Degano[1] and Corrado Priami[2]

[1] Dipartimento di Informatica, Università di Pisa
Via Filippo Buonarroti, 2 – I-56127 Pisa, Italy – `degano@di.unipi.it`
[2] Dipartimento di Informatica e Telecomunicazioni, Università di Trento
Via Sommarive, 1438050 Povo (TN), Italy – `priami@dit.unitn.it`

We are faced with a great challenge: the cross-fertilization between the fields of formal methods for concurrency, in the computer science domain, and systems biology in the biological realm.

From the one side, re-using the theories developed in the last years for mobile and distributed systems may spread light in the systems biology field. Being based on sound and often deep mathematics, these theories may offer solid ways to describe biological systems and to safely reason upon them. Also, formal methods may provide biologists with software tools that can make them save time and efforts. On the other hand, biological systems often have a size of one or two order of magnitude bigger than that of computer systems. More importantly, their efficiency, flexibility and reliability is incredibly superior to that of any computing machinery. So, an effort of understanding biological mechanisms in terms of computer technology will bring over the computer science field new techniques to develop and analyse complex systems that will be more robust, reliable and efficient than the present ones. As an example, we certainly lack now linguistic primitives suitable to model biological systems. A programming language rich enough will help biologists in formalizing biological systems and in predicting their behaviour. At the very same time, such new primitives will help e.g. the computer scientists to design and program systems in the so-called disappearing computer scenario — a foreseen world where everyday life object is equipped with microchips constantly interacting each other.

Process calculi are maybe the most popular framework to study *global computing*, in which a great number of computing agents that

---

⋆ Work partially supported by EU project DEGAS (IST-2001-32072) funded within the FET proactive initiative on Global Computing.

cooperate to achieve common goals, possibly exchanging information through communications or interactions. These agents are geographically dispersed, may move from one site to another, without possibly stopping their ongoing calculations. Also, the knowledge of the running environment is limited and no centralized point of control is assumed. The long term objective of these calculi and their related formal theories is then to design applications, certified to have some clearly stated properties, that may involve thousands or millions of ubiquitous, cooperating entities. Typically, process calculi are built out from a very restricted number of primitives focused on the description of process interaction. They have extensively being investigated in the last three decades since the pioneering work by Hoare [6] and Milner [9]; a closely related sub-field is that of Petri nets, that studies concurrent sytems from the point of view of automata theory. A main result of the reasearch on process calculi are formal theories that deal with several aspects of this fascinating new computing paradigm. Among these aspects, particularly important are the *qualitative* ones, that mainly consider the behaviour of computing agents in terms of equivalences, as well as the *quantitative* aspects, that measure somehow systems behaviour, taking care of probability distributions, time constraints, and so on.

Systems biology [7] is a field that studies the system-level structure and behaviour of complex systems made up of molecular components. In turn, the nature and the behaviour of the components, when taken in isolation, is well-known from molecular biology. The overall goal of this discipline is to control the evolution of these complex systems and to design modifications to them, guaranteeing that some specific properties are satisfied.

As a matter of fact, the goals of systems biology are quite simlar to those of global computing application design. So, if we succeed in modelling biological systems as computer systems, e.g. as computing agents in some process calculus, we shall advance the state of the art in both disciplines.

The connection between the computer and biological world is very well described by the methaphor cells as computations [16]. Biological components (at various level of abstractions) are represented as processes and their interactions result to be communication between processes. Relying on calculi for mobility the effect of a communication can change the future interaction of processes as it happens on the biological side for interacting components.

The above methaphora has been reified exploiting a revised version of the stochastic π-calculus [13] implementd in the BioSPI system [17, 14]. Recently, a biological version of the ambient calculus as well has been considered to model biological systems [15].

Here, we propose to adopt enhanced operational semantics [5] as a descritpion tool to attack the complexity of biological systems within a uniform framework.

The enhanced operational semantics, EOS for short, is an operational way of specifying the behaviour of complex systems in terms of their components, reagrdless of their actual nature, computational or biological or whatelse. It enables its users to design, simulate and analyse systems keeping distinct the various aspects they may have, in particular the many facets related to qualitative and quantitative analysis. Being each aspect orthogonal to the others, it is then possible to combine some of them at a later stage, so taking advantage of a clear separation of concerns.

Recent developments in the EOS theory show that many different families of calculi can be simulated relying on a core π-calculus formalism. The main idea is that the annotations introduced on labels of transitions can be used to control the possible interaction of communicating systems [3]. A further result is that most of the transitions systems originated by calculi for mobility turns out to be a subset of the π-calculus [10] transition system. As a consequence, we can study properties of the selected formalism in the π-calculus by suitable assignment of parameters to transitions. This unifying framework allows us to select the most suitable formalism to design biological systems and relying on the implementation of a π-calculus kernel (e.g., the BioSPI system) to study properties and performing simulations.

As mentioned above, an important aspect of enhanced operational semantics is that it allows to describe various aspect of systems without changing the actual syntax (that remains the standard syntax of many process calculi). Thus, it is possible to define *causal* or *locational* relations between actions, so expressing, e.g., the need of some bio-chemical reactions to occur before a selected one or the need of the co-location of some reactants to enable an interaction.

In a similar, independent way, one can derive or assign *stochastic information* to the system activities, in terms of the probabilities or the statistical rates they have to occur. In this way, we describe system evolutions closer to reality, and we mechanically calculate the probability that a(n un)wanted chain of reactions has to show up. Exploiting a notion of *behavioural equivalence* between processes, one can then group apparently different biological systems, that however exhibit the same dynamics. This equivalence can take as a parameter the wanted (qualitative or quantitative) aspects of the systems under analysis: the user selects what has to be observed and a mechanical (and up to now inefficient) tool is able to detect similarities and dissimilarities. As another application of the equivalences mentioned above, one can study a property known in the sub-field of computer security as *non-interference* [2, 4]: a given system will not interfere with its surrounding environment, whichever it could be, or, in biological terms: does a selected component in a system interfere with others and produce unwanted effects while the whole system evolves?

We end this position paper by noting that although operational semantics is one of the most simple formal tools that can be used in the design of complex systems, it is quite far from the common practice of biologists.

An essential ingredient for the success of the challenge outlined at the beginning is the identification of a communication language between computer scientists and biologists. Such a language must have the following characteristics:

- it must hide as much technical details as possible from the user so that the biologists need not to have strong background in mathematics and formal methods to use it;
- it must be sufficiently structured (i) to avoid as much ambiguities as possible and (ii) to derive consistent formal methods to analyse and simulate systems;
- it must be possible to automatically translate specifications in this language into process calculi specification and to reflect back in the interface language the results of the formal analysis.

To satisfy all the above requirements we think that a graphical, semi-structured language could be a solution at the right level of abstraction. Indeed, biologists already have in use many graphical languages (e.g., [1, 8, 11]) and so computer scientists do. Additionally, software engineers make use of graphical standards as UML to model systems, and there are already definitions of extractors from UML specifcations to π-calculus descriptions [12] that show the feasibility of the approach and that allow reuse of existing tools.

References

1. The systems biology workbench software. 2000.
2. M. Abadi and A. D. Gordon. A Calculus for Cryptographic Protocols: The Spi Calculus. *Information and Computation*, 148(1):1–70, January 1999.
3. L. Brodo. *A tool for comparing calculi for mobility* PhD Thesis, Dipartimento di Informatica, University of Verona.
4. A. Durante, R. Focardi, and R. Gorrieri. A Compiler for Analysing Cryptographic Protocols Using Non-Interference. To appear on ACM TOSEM.
5. P. Degano and C. Priami. Enhanced operational semantics: A tool for describing and analysing concurrent systems. *ACM Computing Surveys*, 33,2:135–176, 2001.
6. C.A.R. Hoare *Communicating Sequential Processes.* Prentice-Hall, 1985.
7. H. Kitano. *Foundations of Systems Biology* MIT Press, 2002.
8. F. A. Kolpakov, E. A. Ananko, G. B. Kolesov, and N. A. Kolchanov. Genenet: a gene network database and its automated visualization. *Bioinformatics*, 14(8):529–537, 1998.
9. R. Milner. *Communication and Concurrency.* Prentice-Hall, London, 1989.
10. R. Milner, J. Parrow, and D. Walker. A calculus of mobile processes (I and II). *Information and Computation*, 100(1):1–77, 1992.
11. H. Ogata, S. Goto, K. Sato, W. Fujibuchi, H. Bono, and M. Kanehisa. Kegg: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, 27(1):29–34, 2000.
12. K. Pokozy and C. Priami. Formal analysis of UML specifications. DEGAS TR, www.omnys.it/degas
13. C. Priami. Stochastic π-calculus. *The Computer Journal*, 38(6):578–589, 1995.
14. C. Priami, A. Regev, W. Silverman, and E. Shapiro. Application of a stochastic name-passing calculus to representation and simulation of molecular processes. *Information Processing Letters*, 80(1), 2001.
15. A. Regev. *Computational systems biology: A calculus for biomolecular knowledge.* PhD Thesis, Department of computer Science and Applied mathematics, Weizmann Institute of Science.
16. A. Regev and E. Shapiro. Cells as computation. *Nature*, Vol. 419, p. 343, 2002.
17. A. Regev, W. Silverman, and E. Shapiro. Representing biomolecular processes with computer process algebra: π-calculus programs of signal transduction pathways. In *Proceedings of the Pacific Symposium of Biocomputing 2000*, 2000.