Formal Executable Models of Cell Signaling Primitives

Carolyn Talcott
SRI International
Menlo Park, CA 94025
Email: clt@csl.sri.com

I. Introduction

Mammalian cells monitor their internal states and the outside environment through biochemical signaling pathways—collections of interacting molecules that detect and transduce diverse signals and transmit information to the nucleus and other subcellular locations (e.g., see [1], [2], [3]). Although signaling pathways are complex, fundamental concepts are emerging from contemporary research. Most signaling pathways involve the hierarchical assembly in *space and time* of multi-protein complexes or modules that coordinate and regulate the flow of information according to logical rules [4], [5]. Compartmentalization, moving from one location to another (translocation), and signaling cascades (sequences of signal propagation) are important features. Moreover, these pathways are embedded in networks having stimulatory, inhibitory, cooperative, and other connections to ensure that a signal will be interpreted appropriately in a particular cell or tissue [6], [7]. For example, EGFR (epidermal growth factor receptor) and other receptor tyrosine kinases, a ubiquitous class of cell surface receptors, can influence fundamental cell fates such as proliferation, growth, survival, and differentiation [8], [9]. Another example is cross-talk between receptors receptors that detect mitogenic signals (such as EGFR) and those that detect adhesion of a normal cell to its substrate (integrin receptors) which is important for cell survival under certain conditions [8].

Thinking of proteins and protein complexes as computational or processing agents we can examine some of the features: each agent encapsulates both processor and code (hardware-software co-design); composite agents (little machines) form dynamically as needed and disband when no longer required; and agents are mobile and move from one location to another (translocation) as needed. Post-translational modifications such as phosporylation or ubitquitination play an important role in determining behavior. Such modifications might activate or inhibit a given functionality, or tag an agent for degradation (garbage collection).

Agents work together by forming "coalitions" in a number of ways, including binding of a ligand to a receptor, homo- and hetero-dimerization, and formation of larger complexes facilitated by scaffold and adaptor proteins. Effects of these complex formations include recruiting an agent to a new location, sequestering an agent (hiding it), and enabling specific functionality (for example by exposing active sites). Examples of functionality includes kinase/phospatase activity, scaffold, adaptor, and switch.

Our objective is two fold: (1) to study the basic signaling mechansims and their interactions as fundamental computational primitives; and (2) to develop a framework to help biologists develop analytical and predictive models of cellular signaling. For this purpose we propose using executable formal models. In section II we review the main ideas underlying the use of executable formal models, and give a brief summary of related work in the area of modeling biological processes. In section III we illustrate modeling of different protein behaviors in Pathway Logic in a model of Toll-like receptor signaling. Section IV gives some concluding thoughts.

II. Executable formal models of signaling processes

Pathway Logic [10], [11], [12], [13] is an approach to the modeling and analysis of molecular and cellular processes based on rewriting logic [14]. A Pathway Logic knowledge base includes data types representing cellular components such as proteins, small molecules, or complexes; compartments/locations; and post-translational modifications. Rewrite rules describe the behavior of proteins and other components depending on modification state and biological context. Each rule represents a step in a biological process such as metabolism or intra/inter- cellular signaling. A collection of such facts forms a formal knowledge base. A model is then a specification of an initial state (cell components and locations) interpreted in the context of a knowledge base. Such models are executable and can be understood as specifying possibly ways a sytem can evolve. Logical inference and analysis techniques are used for simulation to study possible ways a system could evolve, to assemble pathways as answers to queries, and to reason about dynamic assembly of complexes, cascading transmission of signals, feedback-loops, cross talk between subsystems, and larger pathways. Logical and computational reflection can be used to transform and further analyze models.

Given an executable model such as that described above, the path graph of a given initial state is a graph whose nodes are the reachable states and whose edges are the rules connecting them. Paths through the graph then correspond to possible ways a system can evolve. An execution strategy picks out a particular path among those possible. In such a model,

there are many kinds of computation that can be carried out, including: static analysis, forward simulation, forward search, backward search, explicit state model checking, and meta analysis.

Static analysis allows one to examine the structure of the model and to understand how the elements are related and organized (the sort structure). It can be used to infer flow of control and dependencies. Static analysis also provides a means to check for inconsistencies or ill-formed declarations and to look for missing information.

Forward simulation runs the model from a given initial state using a specified strategy either for a fixed number of steps, or until no more rewrites apply. This is extremely fast, and very useful for initial exploration.

Forward search is a breadth-first search of all paths through the transition graph for a given initial state. It will find ALL possible outcomes from a given initial state. Search can also be constrained to find a possibly limited number of states satisfying a given property.

Backward search runs the model backwards. For models satisfying certain constraints, backwards search can answer the question: "From what initial states can we get to this state?". For example it can be used to find all possible precursors to a particular checkpoint.

Explicit state model checking expands the collection of properties that can be investigated. Search concerns only properties of individual states. Model checking also considers properties of paths. For example, we can ask: "If we reach a state that satisfies P then do we always later reach a state satisfying Q?"

Meta analysis allows us to reason about the models themselves. Essential features of models can be abstracted to form families of related models, allowing us to work with uncertainty about reactions. Starting with a base set of known reactions, different instantiations of sets of reactions can be explored. For example, we can search for models where a given path property is true in a given initial state. In addition, rules themselves can be abstracted into families of rules, each family corresponding, for example, to a particular type of reaction, such as activation, inhibition, or translocation. It also allows the knowledge base to be queried as data base, for example finding all rules that involve a given protein (in any or a specified state or location). Finally, using mappings of logics a model can be mapped to another formalism to take advantage of additional tools.

Related Work.

Models of biological systems have been developed using a variety of computational formalisms and logics originally intended for modeling and analysis of computer systems. Much of the effort has been devoted to developing techniques to represent relevant biological concepts and to simulate their behavior, with some work also on analyzing models. Examples include Petri Nets [15], [16], [17]; variants of the Pi-calculus [18], [19]; stochastic logics and associated model checkers [20], [21]; membrane calculi [22], [23], [24]; Statecharts [25]; Life Sequence Charts [26]; Rewriting Logic [11], [12]; and Computation Tree Logic [27].

III. Pathway Logic Modeling of TLR Signaling Processes

Pathway Logic (`http://www.csl.sri.com/users/clt/PLweb/`) models of biological processes are developed using the Maude system (`http://maude.cs.uiuc.edu/`) a formal language and tool set based on rewriting logic. The Rewriting logic [14] formalism is based on two simple ideas: states of a system are represented as elements of an algebraic data type; and the behavior of a system is given by local transitions between states described by *rewrite rules*. A rewrite rule has the form $t \Rightarrow t'$ *if* c where t and t' are patterns (terms possibly containing place holder variables) and c is a condition (a boolean term). Such a rule applies to a system in state s if t can be matched to a part of s by supplying the right values for the place holders, and if the condition c holds when supplied with those values. In this case the rule can be applied by replacing the part of s matching t by t' using the matching values for the place holders in t'. The process of application of rewrite rules generates computations (also thought of as deductions). In the case of biological processes these computations correspond to pathways.

A. Pathway Logic Basics

Pathway Logic models are structured in four layers: (1) sorts and operations, (2) components, (3) rules, and (4) queries. The *sorts and operations* layer declares the main sorts and subsort relations, the logical analog to ontology. The sorts of entities include Chemical, Protein, Complex, and Location (cellular compartments), and Cell. These are all subsorts of the sort, Soup, that represents 'liquid' mixtures, as multisets (unordered collections) of entities. The sort Modification is used to represent post-translational protein modifications. They can be abstract, just specifying being activated, bound, or phosphorylated, or more specific, for example, phosphorylation at a particular site. Modifications are applied using the operator `[ - ]`. For example the term `[TLR4 - act]` represents the activation of the the Toll-like receptor TLR4.

A cell state is represented by a term of the form

```
[cellType | locs]
```

where `cellType` specifies the type of cell, for example Macrophage, and `locs` represents the contents of a cell organized by cellular location. Each location is represented by a term of the form `{ locName | components }` where `locName` identifies the location, for example `CLm` for cell membrane, and `components` stands for the mixture of proteins and other compounds in that location.

The *components* layer specifies particular entities (proteins, chemicals) and introduces additional sorts for grouping proteins in families. The *rules* layer contains rewrite rules specifying individual steps of a process. These correspond to reactions in traditional metabolic and interaction databases. The *queries* layer specifies initial states and properties of

interest. Initial states are insilico Petri dishes containing a cell and ligands of interest.

As explained above, pathways are not predefined. Instead they are assembled by applying the rules starting from an initial state, searching for a state meeting given conditions. The Pathway Logic Assistant (PLA) provides an interactive visual representation of PL models. Using PLA a biologist can: ask for a list of dishes available for study; display the network of signaling reactions for a given dish; formulate and submit queries to find pathways, for example activating one protein without activating a second protein; visualize gene expression data in the context of a network (by coloring the coded proteins according to expression level); or compute and display the downstream subnet of one or more proteins. Given an initial dish, the PLA selects the relevant rules from the rule set and represents the resulting reaction network as a Petri net. This provides a natural graphical representation that is similar to the hand drawn pictures used by biologists, as well as very efficient algorithms for answering queries. PLA, sample models, tutorial material, papers and presentations are available from the Pathway Logic web site, `http://pl.csl.sri.com/`.

B. Toll-like receptor signaling in Pathway Logic

To illustrate how different types of protein behaviors are modeled using rewrite rules we consider the signaling network induced by LPS activation of the Toll-like receptor-4 (TLR4), leading to activation of transcription factors Irf3 and Nfkb1 which in turn bind to an Ifnbgene (Interferon-beta gene) promoter region thus enabling transcription. For this example, the initial state contains the ligands LPS, Lpy, and Ly96 and a cell with Cd14 attached to the cell exterior and all modeled components in other relevant locations (including membrane, cytoplasm, nucleus, mitochondria, endoplasmic reticulum, and clatherin coated pits). Figure 1 shows a screen shot of this signaling network represented as a Petri net. Ovals are places, representing proteins, complexes and other components in different states and locations. Rectangles represent reactions. Darker colored ovals are components present initially. The network can be executed by firing transitions for which all components connected by incoming arrows are present. When a transition fires, incoming components with solid arrows are removed from the state and components connected by outgoing arrows are added to the state. The following we give informal descriptions of some of the signaling reactions in this pathway along with the Maude and/or Petri net representation.

Circulating LBP recognizes LPS in the plasma and brings it to CD14. This aids the loading of LPS onto the LPS receptor complex, which is composed of dimerized TLR4 receptors and two molecules of the extracellular adapter Ly96. This process, is represented by the following rewrite rule.

```
rl[612.TLR4.onby.Lps]:
  (Lps : Lbp) Ly96
  [CellType:CellType | ct
   {CLo | clo  Cd14 } {CLm | clm  TLR4 }]
   =>
  [CellType:CellType | ct
   {CLo | clo  Cd14 (Ly96 : (Lps : Lbp))}
   {CLm | clm [TLR4 - act] }] .
```

The text above the arrow describes the state of the system necessary for the reaction to occur, and the text below the arrow describes the local change. The first line is the rule label, used to relate visual representations with the underlying formal model. The next line shows the ligands outside the cell. `{CLo | clo Cd14}` says that Cd14 is attached to the outside of the cell membrane (tag `CLo`), and `{CLm | cm TLR4}` says that the cell membrane (tag `CLm` must contain TLR4 along with possibly other unspecified components, indicated by the place holder `cm`. After the rule is applied, simulating a reaction step, the ligands are now also bound to the outside of the cell,

```
{CLo | clo  Cd14 (Ly96 : (Lps : Lbp))},
```

and TLR is activated, `{CLm | clm [TLR4 - act]}`. The graphical representation of this rule is shown in the upper right corner of the TLR4 pathway screen shot show in Figure 1.

Triggering of TLR4 causes the adaptor protein Tirap to be recruited to the receptor complex which then recruits Myd88, followed by Irak4. This is represented by rules labeled 208, 857, and 728 respectively in Figure 1.

Activated TLR4 also recruits the Tollip-Irak1 complex (rule 189) and Ticam2 (rule 859) to the membrane. Ticam2 recruits Ticam1 (rule 860) which recruits Tbk1 and causes it to be activated (rule 863).

Activated Tbk1 phosphorylates the transcription factor Irf3 causing it to translocate to the nucleus where it binds to the Ifnbgene (Interferon-beta gene) promoter (rule 605).

Meanwhile, back at the receptor complex, the components are assembled in a way which allows Irak1 and Irak4 to phosphorylate each other (rule 773). This changes their affinity for the receptor and allows Irak1 and Traf6 to detach from the receptor complex and interact with the preformed complex made of Tab2, Tab1, and Tak1 (rule 665) phosphorylating members of this complex (rule 805 off screen). The new complex translocates to the cytoplasm where Tak1 is subsequently activated by a series of reactions which involve ubiquitation of the phosphorylated proteins ultimately leading to activation of Nfkb1 in the nucleus.

Viewing the signaling pathway as a network of reactions allows us to see patterns of interaction. For example we see that activated TLR4 initiates several parallel threads of activity. Three of these are integrated by the reaction (rule 773) that phosphorylates Irak1 and Irak4. The thread activating Irf3 (rule 605) proceeds independently but shares use of Ticam1 recruited to the membrane. The threads through rules 773/805 and 605 proceed independently to produce active transcription factors for the Ifnb gene.

IV. Conclusion

We have discussed key features of intra-cellular signaling processes as computational primitives and shown how they can be modeled, executed and analyzed using Pathway Logic.

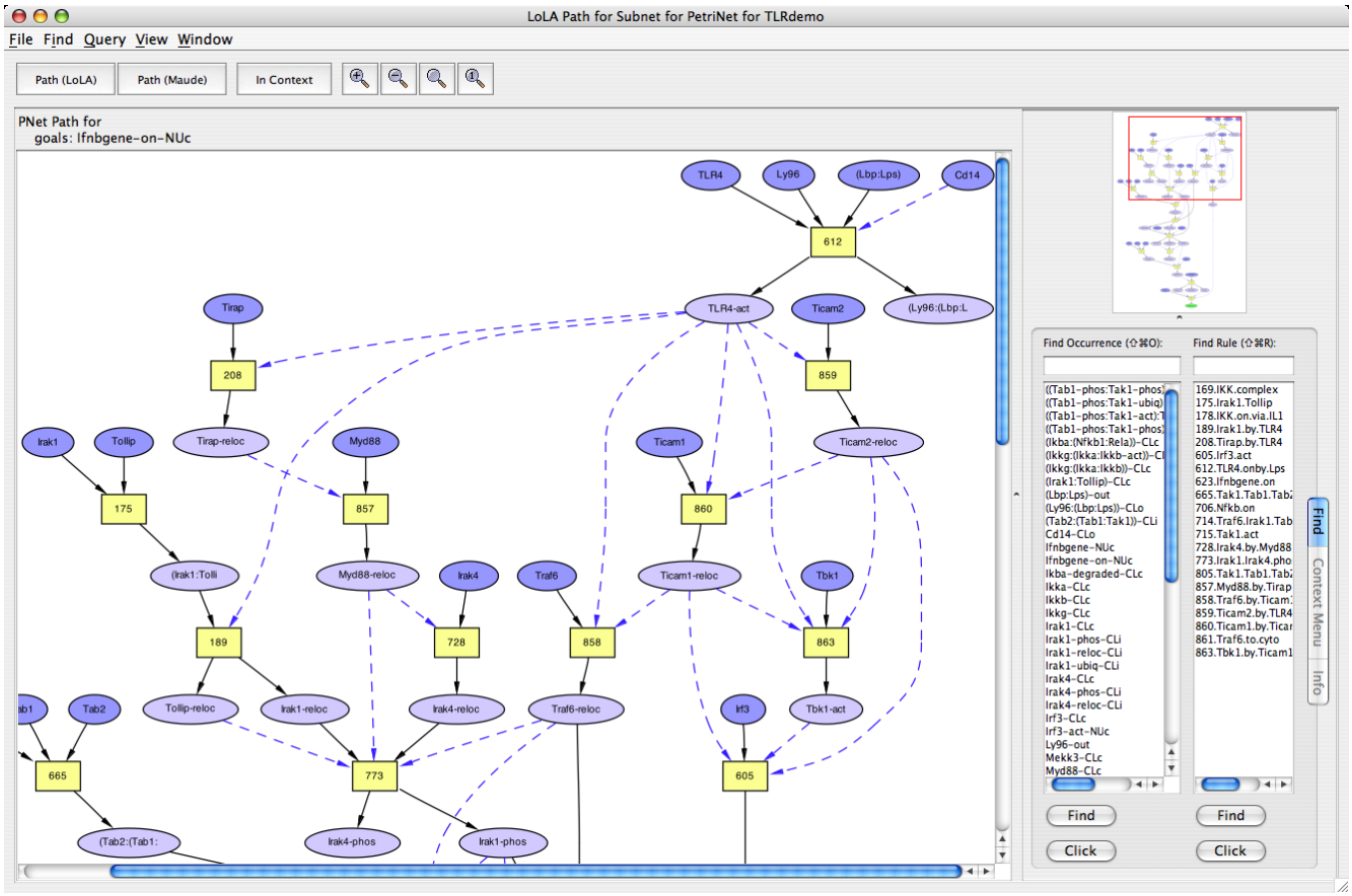

Fig. 1. TLR4 pathway

This is intended to lay the ground for deeper formal studies of the relations between the building blocks and composition mechanisms of biological processes and foundations of computing.

Currently Pathway Logic has a knowledge base of over 1000 rules and 600 basic components and is used to analyze signaling in different cell types. Questions of interest include effects of perturbations (knockouts, knockins) and finding upstream and downstream effects of given components.

Acknowledgment

The author would like to thank the Pathway Logic team for lively discussions and lots of hard work developing models and analysis techniques. This work was partially funded by NIH/NIGMS grant GM68146 and NSF grant IIS-0513857.

References

[1] J. M. Kyriakis and J. Avruch, "Mammalian mitogen-activated protein kinase signal transduction pathways activated by stress and inflammation," *Physiol. Rev.*, vol. 81, pp. 807–869, 2001.

[2] G. P. et al., "Mitogen-activated protein (MAP) kinase pathways: regulation and physiological functions," *Endocr. Rev.*, pp. 153–183, 2001.

[3] J. E. D. et al., "Cross signaling, cell specificity, and physiology," *Am. J. Physiol. Cell. Physiol.*, vol. 283, pp. C2–C28, 2002.

[4] J. Jordan, E. Landau, and R. Iyengar, "Signaling networks: The origins of cellular multitasking," *Cell*, vol. 103, pp. 193–200, 2000.

[5] K. W. Kohn, "Molecular interaction map of the mammalian cell cycle control and DNA repair systems," *Mol Biol Cell*, vol. 10, pp. 2703–2734, 1999.

[6] D. Fambrough, K. McClure, A. Kazlauskas, and E. S. Lander, "Diverse signaling pathways activated by growth factor receptors induce broadly overlapping, rather than independent, sets of genes," *Cell*, vol. 97, pp. 727–741, 1999.

[7] T. Pawson and T. M. Saxton, "Signaling networks–do all roads lead to the same genes?" *Cell*, vol. 97, pp. 675–678, 1999.

[8] A. Gschwind, E. Zwick, N. Prenzel, M. Leserer, and A. Ullrich, "Cell communication networks: epidermal growth factor receptor transactivation as the paradigm for interreceptor signal transmission," *Oncogene*, vol. 20, pp. 1594–1600, 2001.

[9] Y. Yarden and M. X. Sliwkowski, "Untangling the erbb signalling network," *Nature Review Molecular Cell Biology*, vol. 2, pp. 127–137, 2001.

[10] S. Eker, M. Knapp, K. Laderoute, P. Lincoln, J. Meseguer, and K. Sonmez, "Pathway Logic: Symbolic analysis of biological signaling," in *Proceedings of the Pacific Symposium on Biocomputing*, January 2002, pp. 400–412. [Online]. Available: http://www.csl.sri.com/papers/lincoln-pathway-logic-psb-2002/

[11] S. Eker, M. Knapp, K. Laderoute, P. Lincoln, and C. Talcott, "Pathway Logic: Executable models of biological networks," in *Fourth International Workshop on Rewriting Logic and Its Applications (WRLA'2002), Pisa, Italy, September 19 — 21, 2002*, ser. Electronic Notes in Theoretical Computer Science, vol. 71. Elsevier, 2002, http://www.elsevier.nl/locate/entcs/volume71.html.

[12] C. Talcott, S. Eker, M. Knapp, P. Lincoln, and K. Laderoute, "Pathway logic modeling of protein functional domains in signal transduction," in *Proceedings of the Pacific Symposium on Biocomputing*, January 2004.

[13] C. Talcott and D. L. Dill, "The pathway logic assistant," in *Third*

International Workshop on Computational Methods in Systems Biology, G. Plotkin, Ed., 2005, pp. 228–239.

[14] J. Meseguer, "Conditional Rewriting Logic as a unified model of concurrency," *Theoretical Computer Science*, vol. 96, no. 1, pp. 73–155, 1992.

[15] R. Hofestädt, "A Petri net application to model metabolic processes," *Syst. Anal. Mod. Simul.*, vol. 16, pp. 113–122, 1994.

[16] V. N. Reddy, M. N. Liebmann, and M. L. Mavrovouniotis, "Qualitative analysis of biochemical reaction systems," *Comput. Biol. Med.*, vol. 26, pp. 9–24, 1996.

[17] I. Zevedei-Oancea and S. Schuster, "Topological analysis of metabolic networks based on Petri net theory," *In Silico Biology*, vol. 3, no. 0029, 2003. [Online]. Available: \url{http://www.bioinfo.de/isb/abstracts/03/0029.html}

[18] A. Regev, W. Silverman, and E. Shapiro, "Representation and simulation of biochemical processes using the pi-calculus process algebra," in *Pacific Symposium on Biocomputing*, R. B. Altman, A. K. Dunker, L. Hunter, and T. E. Klein, Eds., vol. 6. World Scientific Press, 2001, pp. 459–470.

[19] C. Priami, A. Regev, E. Shapiro, and W. Silverman, "Application of a stochastic name-passing calculus to representation and simulation of molecular processes," *Information Processing Letters*, 2001, in press.

[20] M. Calder, V. Vyshemirsky, D. Gilbert, and R. Orton, "Analysis of signalling pathways using the PRISM model checker," in *Proceedings of the Third International Conference on Computational Methods in System Biology*, G. Plotkin, Ed., 2005.

[21] A. Hinton, M. Kwiatkowska, G. Norman, and D. Parker, "PRISM: A tool for automatic verification of probabilistic systems," in *12th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'06)*, ser. Lecture Notes in Computer Science, H. Hermanns and J. Palsberg, Eds., vol. 3920. Springer, 2006, pp. 441–444.

[22] A. Regev, E. Panina, W. Silverman, L. Cardelli, and E. Shaprio, "Bioambients: An abstraction for biological compartments," 2003, to appear TCS.

[23] F. Nielson, H. R. Nielson, C. Priami, and D. Rosa, "Control flow analysis for bioambients," in *BioConcur*, 2003.

[24] L.Cardelli, "Brane calculi interactions of biological membranes," in *Computational Methods in Systems Biology*. Springer, 2004.

[25] S. Efroni, D. Harel, and I. Cohen, "Towards rigorous comprehension of biological complexity: Modeling, execution and visualization of thymic t-cell maturation," *Genome Research*, 2003, Special issue on Systems Biology, in press.

[26] N. Kam, D. Harel, H. Kugler, R. Marelly, A. Pnueli, J. Hubbard, and M. Stern, "Formal modeling of C.elegans development: A scenario-based approach," in *First International Workshop on Computational Methods in Systems Biology*, ser. Lecture Notes in Computer Science, vol. 2602. Springer, 2003, pp. 4–20.

[27] N. Chabrier-Rivier, M. Chiaverini, V. Danos, F. Fages, and V. Schächter, "Modeling and querying biomolecular interaction networks," 2004, to appear in TCS. [Online]. Available: \url{http://pauillac.inria.fr/%7Efages/Papers/CCDFS04tcs.pdf}