Pacific Symposium on Biocomputing 6:459-470 (2001)

Representation and simulation of biochemical processes using the π-calculus process algebra

Aviv Regev[1,2], William Silverman[2] and Ehud Shapiro[2]
[1] *Department of Cell Research and Immunology, Life Sciences Faculty*
Tel Aviv University, Tel Aviv 69978, Israel
[2] *Department of Computer Science and Applied Mathematics*
Weizmann Institute of Science, Rehovot, Israel
Tel: 972-8-9344506 Fax: 972-4-8348799
e-mail: {*aviv, udi, bill*}*@wisdom.weizmann.ac.il*

Abstract

Despite the rapidly accumulating body of knowledge about protein networks, there is currently no convenient way of sharing and manipulation of such information. We suggest that a formal computer language for describing the biomolecular processes underlying protein networks is essential for rapid advancement in this field. We propose to model biomolecular processes by using the π-Calculus, a process algebra, originally developed for describing computer processes. Our model for biochemical processes is mathematically well-defined, while remaining biologically faithful and transparent. It is amenable to computer simulation, analysis and formal verification. We have developed a computer simulation system, the PiFCP, for execution and analysis of π-calculus programs. The system allows us to trace, debug and monitor the behavior of biochemical networks under various manipulations. We present a π-calculus model for the RTK-MAPK signal transduction pathway, formally represent detailed molecular and biochemical information, and study it by various PiFCP simulations.

1 Introduction

Biochemical processes, carried out by networks of proteins, are responsible for most of the information processing inside the cell. The high complexity of these systems makes their proper understanding difficult. Even convenient storage of accumulated data in a way that would facilitate research is a challenging task.

Biochemical systems are usually analyzed either by simulating the continuous, mass-action differential equations, or by discrete, Monte-Carlo simulations. In recent years, various approaches from Computer Science have been adapted for the representation of pathways. These include Boolean networks [1], Petri nets [2], graph based approaches [3] and Object-oriented databases (e.g. EcoCyc [4,5]) and simulation environments (e.g. E-cell [6]). While each of these approaches captures some of the information regarding pathways and their components, none fully integrates dynamics, molecular and biochemical detail.

As an alternative, we suggest using the π-calculus, a formal language originally developed for specifying concurrent computational systems [7]. In such systems, multiple processes interact with each other by synchronized pair-wise communication on complementary communication channels, and modify each other by transmitting channels from one process to another. This feature, termed mobility, allows the network structure to change with interaction.

We show how the π-calculus can be used to model biochemical networks as mobile communication systems. We treat molecules and their individual domains as computational processes, where their complementary structural and chemical determinants correspond to communication channels. Chemical interaction and subsequent modification coincide with communication and channel transmission.

The π-calculus is suitable for modeling various molecular systems, including transcriptional circuits, metabolic pathways, and signal transduction (ST) networks. We illustrate the system using a model of the ST pathway leading from a receptor tyrosine kinase, through Ras and into the ERK1 MAPK cascade. Based on this formalism, we have developed a computer system, called PiFCP, for discrete simulation. We present data obtained from PiFCP simulations on the RTK-MAPK pathway, under normal and perturbed states.

2 A formal representation language for biochemical pathways

We focus our attention on signal transduction (ST) pathways. The well-studied RTK-MAPK pathway [8,9], is composed of 14 kinds of proteins. These bind and form complexes, modify certain residues on their counterparts (mostly by phosphorylation and dephosphorylation), change their conformation and activity, and translocate between different cellular compartments (cytosol, nucleus and membrane). A change in gene expression patterns is the end result computed by this network of interactions. An informal graphic representation is given in Figure 1. While visually appealing, such a representation lacks in coverage, formal semantics, and dynamics. As an alternative, we now incrementally describe how to model such biochemical processes using the π-calculus.

We view each biochemical pathway as a process, denoted by a capitalized name, P (Figure 2,(2.1)) e.g:

(1) RTK_MAPK_pathway

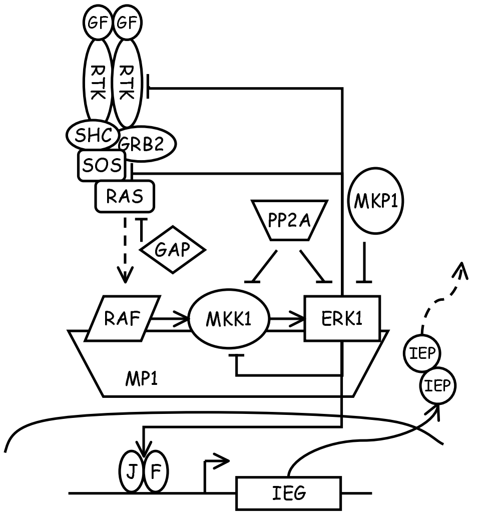

Figure 1: The RTK MAPK pathway: A protein ligand molecule (GF), with two identical domains, binds two receptor tyrosine kinase (RTK) molecules on their extracellular part. The bound receptors form a dimeric complex, and cross-phosphorylate and activate the protein tyrosine kinase in their intracellular part. The activated receptor can phosphorylate various targets, including its own tyrosines. The phosphorylated tyrosine is identified and bound by an adaptor molecule, SHC. A series of protein-protein binding events follows, leading to formation of a protein complex (SHC, GRB2, SOS, and Ras) at the receptor intracellular side. Within this complex, the SOS protein activates the Ras protein, which in turn recruits the serine/threonine protein kinase, Raf, to the membrane, where it is subsequently phosphorylated and activated. A cascade of phosphorylations/activations follows, from Raf to MEK1 to ERK1. This cascade culminates in activation of the threonine and tyrosine protein kinase, ERK1. Activated ERK1 translocates to the nucleus, where it phosphorylates and activates transcription factors, such as AP-1, leading to de novo gene expression.

A pathway is defined as a collection of concurrently operating molecules, seen as processes with potential behavior. Concurrency is denoted by the PAR operator, | (2.8).

(2) RTK_MAPK_pathway ::= Free_ligand | $\cdots$ | RTK | Ras | $\cdots$

A protein molecule is composed of several domains, each of which is modeled as a process as well.

(3) Free_ligand ::= Free_binding_domain | Free_binding_domain

Two molecules (or domains) interact with each other based on their structural and chemical complementarity[10]. Interaction is accomplished by the motifs and residues that constitute a domain. These are viewed as channels, or communication ports of the molecule. Two complementary motifs are denoted by a global name and co-name pair, x and $\overline{\text{x}}$ ((2.2),(2.3)) . An interaction event may occur only when such a complementary pair is shared by the interacting molecules.

$$\text{(4)}\quad \text{Free\_binding\_domain} ::= \overline{\text{ligand\_binding}} \cdots$$

$$\text{(5)}\quad \text{Free\_extracellular\_domain} ::= \text{ligand\_binding} \cdots$$

A process can also be defined in a parametric way where a parameter is a motif, e.g.

$$\text{(6)}\quad \text{Free\_intracellular\_domain}(\text{tyr}, \text{tyr}, \text{sh2}, \cdots)$$

Biochemical interaction events may occur in sequence, in parallel with other independent occurrences, or in a mutually exclusive, competitive fashion (e.g. binding of agonist or antagonist to a receptor). A sequence of interactions in which a molecule may participate is denoted by a prefix operator, . (2.9). Mutually exclusive interactions are summed together, by +, (2.10). For instance:

$$\begin{aligned}\text{(7)}\quad \text{Free\_extracellular\_domain} ::= \;& \text{ligand\_binding} \;.\; \text{rtk\_binding} \;.\; \cdots \\ & + \text{antagonist\_binding}\end{aligned}$$

Importantly, a pathway is not merely a bag of molecules and their domains. It is composed of defined compartments. First, the parallel domains of a single molecule are linked together by a single backbone. Then, distinct multi-molecular complexes form. Finally, molecules are separated into higher-order cellular compartments. In all three cases molecules which share a common compartment may interact with each other, while molecules excluded from the compartment may not [8]. We represent compartments by restricted communication scopes. Channel scope is restricted using the operator new (2.11):

$$\begin{aligned}\text{(8)}\quad \text{RTK} ::= (\textsf{new}\ \text{backbone})\ (&\text{Extracellular\_domain} \mid \\ &\text{Transmembranal\_domain} \mid \\ &\text{Intracellular\_domain})\end{aligned}$$

Biochemical interaction affects subsequent events in the pathway. For example, as a result of interaction between a protein tyrosine kinase and its substrate, the substrate tyrosine residue changes to a phosphorylated one, affecting its potential to interact with pTyr binding proteins.

This is modeled using *mobility*[7]. Mobility is achieved by allowing processes to send and receive global channel names which may be used for subsequent interaction ((2.6),(2.7),(2.23)). For instance, (9) and (10) communicate over the channel phosph_site. (9) sends the channel p_tyr to (10); (10) will now use p_tyr instead of tyr in its subsequent communications.

(9) Active_kinase ::= $\overline{\text{phosph\_site}}\langle\text{p\_tyr}\rangle$. ···

(10) Binding_domain ::= phosph_site(tyr) . tyr . ···

Mobility is also used to model compartment changes, such as complex formation. Private channels, representing scopes, are extruded and changed, thereby changing scopes dynamically. For example, consider the three-molecule complex that forms between a dimeric ligand and the extracellular domain of two RTK receptors:

(11) Free_ligand ::= (new backbone)
(Free_binding_domain | Free_binding_domain)

(12) Free_binding_domain ::= $\overline{\text{ligand\_binding}}\langle\text{backbone}\rangle$.
Bound_ligand

(13) Extracellular_domain ::= ligand_binding(cross_backbone) .
Bound_Extracellular_domain

When a Free_ligand interacts with two Extracellular_domain*s*, the same backbone channel is sent by the two Free_binding_domain*s* to the two Extracellular_domain*s*. These three molecules may now continue to exclusively communicate on the backbone channel, and are thus linked.

A molecule may often participate in one of several mutually exclusive interactions. This is denoted in the language by summation (2.10), representing competitive choice. Choice is resolved in a completely non-deterministic way: All enabled communications (where both input and output are available) are equi-potent. Once one is chosen the others are excluded[a] (2.23). For example, in a system with three molecular processes,

(14) Free_extracellular_domain ::= ligand_binding . rtk_binding . ···
+ antagonist_binding

(15) Ligand ::= $\overline{\text{ligand\_binding}}$. Bound_ligand

[a] A more biologically realistic model would assign different probabilities to different interactions, based on reaction rates. We augment the model to account for this in [11]

(16) $\text{Antagonist} ::= \overline{\text{antagonist\_binding}}\ .\ \text{Bound\_antagonist}$

either an interaction on ligand_binding, or an interaction on antagonist_binding, will occur. If ligand_binding is chosen (non-deterministically), then the antagonist_binding option is discarded.

Finally, the language allows us to follow a molecule's (process) fate as its structure (motifs) and compartment (scope) change with interaction. This is done by using recursive parametric definitions (2.20). For instance, an event may prefix (with the . operator) a reconstitution of a molecular process (e.g. an enzyme):

(17) $\text{Active\_kinase} ::= \overline{\text{tyr}}\langle \text{p} - \text{tyr}\rangle\ .\ \text{Active\_kinase}$

According to these general principles, detailed information on complex pathways, molecules and biochemical events, can be represented formally. We have developed such a full model for the RTK-MAPK pathway [12]. Since the π-calculus is a well-developed formalism, the resulting representations are well-defined, with clear semantics [13,7]. Congruence laws (Figure 2) ensure that a parallel system can be written in a sequential syntax. Several reaction rules (Figure 2), provide formal semantics to these representations, allowing the system to change with interaction, as described above.

3 Implementation

Once a detailed π-calculus model of a particular pathway is compiled, one would like to be able to run a simulation of it. To this end, we have developed a computer application, called PiFCP [12]. PiFCP is based on the Logix system [14], which implements Flat Concurrent Prolog (FCP [15]). Two unique features of FCP made it suitable for our purposes. First, the ability to pass logical variables in messages was used to implement the name-passing mechanism of the π-calculus. Second, its support of guarded atomic unification allowed synchronized interaction with both input and output guards. Note, that previous implementations of the π-calculus or of related formalisms [7] do not provide such full synchronous communication.

An appropriate surface syntax was devised for the full polyadic π-calculus syntax, in such a way that it is clearly insulated from general Logix procedures. Thus, a pure π-calculus representation is maintained in spite of the use of an FCP-based platform.

We built a compiler from PiFCP to FCP. PiFCP channels are represented by FCP message streams, on which messages are written and from which messages are read and consumed. Messages are transformed with identification tags that ensure synchronized release of input and output guards, as well as their appropriate consumption in cases of mutually exclusive choice. Each PiFCP process is transformed to an FCP procedure, and its channel set is identified. This allows full use of channels as in the original calculus: for global communication, as parameters in recursive definitions, newly restricted channels, and bound input channels, to be instantiated only following communication.

The PiFCP system is based on the classical π-calculus and it captures qualitative aspects of protein networks. As noted above, exact quantitative modeling requires a modification of the stochastic version of the calculus [16], and will be described elsewhere[11]. Note, that the π-calculus specifications may be associated with different semantics of interaction. Thus, the same pathway representation may be studied with the original non-deterministic semantics, described here, or with a stochastic quantitative semantics.

Several debugging tools are available for tracing step-wise execution of PiFCP programs. These include tree traces and step-by-step execution mode, with the ability to set specific break points. The level of detail in which a system is traced (process, channels, messages, senders, etc.) can be determined dynamically throughout a session. Thus, not only the net outcome of a computation can be studied, but also the specific scenario that has led to this outcome.

4 PiFCP simulation analysis of a π-calculus model for the RTK-MAPK pathway

We have constructed a detailed model of the RTK-MAPK pathway in the π-calculus. The model is composed of 15 molecular processes (including ATP and GTP), with 24 different domains and 15 sub-domains. Four compartments (extracellular, membrane, cytoplasm and nucleus) were defined. A major portion of current knowledge has been incorporated into this concise (250 lines) formal representation. The local nature of the π-calculus allowed us to build this large model incrementally. The complete specifications are available[12].

We performed simulation studies under two types of perturbations (Table 1): we either modified the quantities of one or more of the molecules, or we "mutated" molecules by "hacking" their code. Note, that such manipulation is similar to mutational analysis in laboratory experiments: domains and residues are deleted, inserted or modified. The effect of perturbation was observed both globally (e.g. amounts of active proteins), and specifically (step-by-step scenario of interaction). Global results are given in Table 1.

Most perturbations have yielded the expected effects. We have noticed

Perturbation Details	Signal
RAF increase	Increase
MP1 increase	Decrease
MP1 and MEK increase	Increase
ERK increase	Increase
MKP increase	Decrease
Monomer ligand (competitor) Ligand ::= (new backbone)(Free_binding_domain)	Slight decrease
Dominant-negative receptor Rtk ::= (new backbone) (Extracellular | Transmembranal)	Decrease
Membrane-localized Raf Membrane ::= (new mem_env) (Raf(mem_env) | ···)	Ligand independent
Constitutively active ERK Erk_Catalytic_Core ::= Erk_lip(glu, glu) | Active_Erk_Kinase | ···	Ligand independent

Table 1: PiFCP simulation results for the RTK-MAPK pathway: For each of the listed perturbations, the signal was evaluated based on the quantities of active Erk and de-novo gene expression. Each perturbation was typically administered in several doses. Note, that the slight decrease with monomer ligand, was only observed at the level of receptor activation.

that many perturbations can be buffered by the system, albeit not at extreme doses. This qualitative observation should be further verified in full stochastic simulations. Some results were initially surprising. In particular, note the inhibitory effect of large quantities of the adaptor protein MP1, which was reported to be a facilitator of Mek and Erk phosphorylation[17]. When this result was followed with step-by-step debugging sessions (not shown), MP1 was observed to sequester Mek, Erk, and Raf, becoming inhibitory at high quantities. Similar results were recently obtained by another simulation analysis of the MAPK cascade [18]. Thus, the ability to monitor the pathway in action at the local level of a single molecule, offers a unique opportunity to decipher global observations.

5 Discussion

Detailed biochemical and molecular data is available for an increasing number of biochemical systems. Storing and analyzing this data is an essential goal for pathway informatics. Since most of this data is available only as literature abstracts and articles, there is a need for a formal bioinformatic solution.

Formal representations or ontologies are essential for encoding this vast knowledge. Two types of formal representations have been distinguished [5]. *Declarative representations*, such as ontologies, break information down into atomic components and define relationships among those atomic components. They allow us to represent and store current knowledge in an exact non-ambiguous form, and then query and process it reliably. *Procedural implementations*, such as those embodied in most simulation programs, typically lack such well-structured representation of knowledge, and information is embedded in a convoluted fashion. Furthermore, most mathematical approaches underlying dynamic simulations treat molecules as atomic entities. Such simpler models are often insufficient.

Representing biochemical networks in the π-calculus offers the synthesis of both declarative and procedural representation. On the one hand, molecular detail is clearly defined in a well-structured, biologically faithful fashion. This is based on a strong correspondence between the syntax of the calculus and biochemical networks. Complex networks can be modeled incrementally due to the modular nature of both biochemical systems and the calculus. On the other hand, the formal semantics associated with the syntax allowed us to develop the PiFCP implementation for simulation purposes. As shown, PiFCP can be used to obtain both a step-by-step, lower level, understanding, as well global output, an important bioinformatics tool.

Our approach enjoys many of the benefits of other CS approaches which are currently being adapted for studying biochemical systems. Similar to Petri nets [2], it treats biochemical pathways as concurrent processes. Petri nets have been successful in treating many, mostly metabolic, biochemical systems. The advantage of the π-calculus is in the explicit, detailed description of each node or process in the system. In this it is closer to the hierarchy of an object-oriented systems, which although highly successful in storing pathway-related information, lack dynamics [5].

In order to become a full bioinformatics approach further development is required. The detailed level offered by the formalism can become deterring at times. Furthermore, some complex biochemical events require elaborate encoding. These problems may be alleviated by taking advantage of the modular nature of the calculus. Thus, different components of the same system can be specified in different levels of detail. This approach may be highly suitable for studying molecular modules [19]. Significant scaling may be achieved by basing the modeling on existing ontologies, which will allow integration with existing databases. Additional facilitation may be offered by modification of the original calculus based on biological motivations, by offering generic processes for routine events, or by integration of graphical representation approaches with the algebraic one. The graph replacement chemistry, a recently proposed formalism for DNA processing[3], may be particularly suitable for such integration.

The classical π-calculus and its PiFCP implementation offer only a semi-quantitative view of biochemical processes without explicitly associated time. All interactions that may happen have an equal chance of occuring. This is only a coarse approximation of biochemical systems, where different reactions have different rates. We have addressed this issue by developing a second system, the PsiFCP. This variant is based on a stochastic version of the calculus, where communication actions are equipped with rates, analogous to reaction rates [16,11], and a race condition governs the time evolution of the system. This implementation offers the full quantitative capabilities necessary for accurate simulation of biochemical processes.

6 Future Prospects

A specification-based approach also allows to formally analyze the behavioral properties of modeled systems, using methods and tools for formal verification [20,7]. Thus, desirable outcomes and properties of a biomolecular process, which is represented in the π-calculus, can be formally proven. Furthermore, the π-calculus theory allows us to formally compare two programs, in order to determine the degree of mutual similarity of their behavior, termed *bisimulation*. Different levels of similarity, of weakening strength, have been defined [7].

This opens up completely novel possibilities in the study of biochemical systems. Comparison of similar pathways is the first step in studying the evolution of entire processes [21], establishing a *homology of processes*. While in this paper we have represented pathways at the molecular level, biochemical processes may also be viewed at a higher, functional level. Formal comparison methods can also be used in order to prove that the molecular information indeed supports the behavior we expect and specify.

The use of formal and algorithmic approaches has greatly accelerated progress in the sequence and structure branches of biology. Adopting a common representation language for biomolecular processes may similarly accelerate progress in understanding their function and evolution.

1. S. A. Kauffman. *The Origins of Order: Self-Organization and Selection in Evolution.* Oxford University Press, 1993.
2. P.J.E. Goss and J. Peccoud. *Proceedings of the National Academy of Sciences USA*, 95:6750–6754, 1998.
3. J. S. McCaskill and U. Niemann. In A. Condon and G. Rozenberg, editors, *Proceeding of DNA6: 6th International Meeting on DNA based computers*, pages 89–99, 2000.

4. P. D. Karp, M. Riley, S. M. Paley, A. Pellegrini-Toole, and M. Krummenacker. *Nucleic Acids Research*, 25:43–50, 1997.
5. P.D. Karp. *Bioinformatics*, 16:269–285, 2000.
6. M. Tomita, K. Hashimoto, K. Takahashi, T. S. Shimizu, Y. Matsuzaki, F. Miyoshi, K. Saito, S. Tanida, K. Yugi, J. C. Venter, and C. A. Hutchison. *Bioinformatics*, 15:72–84, 1999.
7. R. Milner. *Communicating and Mobile Systems: The π-Calculus*. Cambridge University Press, 1999.
8. C-H. Heldin and M. Purton. *Signal Transduction*. Chapman and Hall, 1996. Modular Texts in Molecular and Cell Biology 1.
9. T. S. Lewis, P. S. Shapiro, and N. G. Ahn. *Advances in Cancer Research*, 74:49–139, 1998.
10. T. Pawson and P. Nash. *Genes and Development*, 14:1027–1047, 2000.
11. A. Regev, C. Priami, W. Silverman, and E. Shapiro. Stochastic process algebras for the modeling of biomolecular processes. Submitted for publication.
12. The PiFCP system, the RTK-MAPK π-calculus model, and other supplementary material is available upon request, and will be available at `http://www.wisdom.weizmann.ac.il/~aviv`.
13. R. Milner. In F. L. Bauer, W. Brauer, and H. Schwichtenberg, editors, *Logic and Algebra of Specification, Proceedings of International NATO Summer School (Marktoberdorf, Germany, 1991)*, volume 94, pages 428–440. Springer-Verlag, 1993.
14. W. Silverman, M. Hirsch, A. Houri, and E. Shapiro. In E. Shapiro, editor, *Concurrent Prolog (vol. II)*, pages 46–78. MIT Press, 1987.
15. E. Shapiro. In E. Shapiro, editor, *Concurrent Prolog (vol. I)*, pages 157–187. MIT Press, 1987.
16. C. Priami. *The Computer Journal*, 38:578–589, 1995.
17. H. J. Schaeffer, A. D. Catling, S. T. Eblen, L. S. Collier, A. Krauss, and M. J. Weber. *Science*, 281:1668–1671, 1998.
18. A. Levchenko, J. Bruck, and P.W. Sternberg. *Proceedings of the National Academy of Sciences USA*, 97:5818–5823, 2000.
19. L.H. Hartwell, J.J. Hopfield, S. Leibler, and A.W. Murray. *Nature*, 402(Suppl):C47–C52, 1999.
20. N. Francez. *Formal Verification*. Addison Wesley, 1992.
21. S. F. Gilbert, J. M. Opitz, and R. A. Raff. *Developmental Biology*, 173:357–372, 1996.

Processes and channels

$P,\ Q,\ \cdots$	process names (2.1)
$x,\ y,\ \cdots$	channel names (2.2)
$\overline{x},\ \overline{y},\ \cdots$	channel co-names (2.3)

Events

$\pi ::=$	x	communication on channel name x (2.4)
	$\overline{x}$	communication on channel co-name x (2.5)
	$x(y)$	receive y along x (2.6)
	$\overline{x}\langle y\rangle$	send y along x (2.7)

Process syntax

$P ::=$	$P_1 \mid P_2$	parallel processes (2.8)
	$\pi \,.\, P_1$	sequential prefixing by communication (2.9)
	$\pi_1 \,.\, P_1 \ +\ \pi_2 \,.\, P_2$	mutually exclusive communications (2.10)
	$(\mathsf{new}\ x)P$	new communication scope (2.11)
	0	inert process (2.12)

Structural congruence

$P \mid Q \equiv Q \mid P$	commutativity of PAR (2.13)
$(P \mid Q) \mid R \equiv P \mid (Q \mid R)$	associativity of PAR (2.14)
$P\ +\ Q \equiv Q\ +\ P$	commutativity of summation (2.15)
$(P\ +\ Q)\ +\ R \equiv P\ +\ (Q\ +\ R)$	associativity of summation (2.16)
$(\mathsf{new}\ x)0 \equiv 0$	scope of inert processes (2.17)
$(\mathsf{new}\ x)(\mathsf{new}\ y)P \equiv (\mathsf{new}\ y)(\mathsf{new}\ x)P$	multiple communication scopes (2.18)
$((\mathsf{new}\ x)P) \mid Q) \equiv (\mathsf{new}\ x)(P \mid Q)$ if $x \notin FN(Q)$	scope extrusion (2.19)
$A(\vec{y}) \equiv \{\vec{y}/\vec{x}\}Q_A$	recursive parametric definition (2.20)
$x(y).P = x(z).(\{z/y\}\,P)$ if $z \notin FN(P)$	renaming of input channel y (2.21)
$(\mathsf{new}\ y).P = (\mathsf{new}\ z).(\{z/y\}\,P)$ if $z \notin FN(P)$	renaming of restricted channel y (2.22)

Reaction rules

$(\cdots + \overline{x}\langle z\rangle.Q)\vert(\cdots + x(y).P) \rightarrow Q\vert P\,\{z/y\}$	communication (COMM)(2.23)
if $P \rightarrow P'$ then $P\vert Q \rightarrow P'\vert Q$	reaction under parallel composition (2.24)
if $P \rightarrow P'$ then $(\mathsf{new}\ x)P \rightarrow (\mathsf{new}\ x)P'$	reaction within restricted scope (2.25)
if $Q \equiv P, P \rightarrow P'$, and $P' \equiv Q'$ then $Q \rightarrow Q'$	reaction up to structural congruence (2.26)

Figure 2: The π-calculus: The calculus consists of three components: a simple syntax for writing formal descriptions; a set of congruence laws that determine when two syntactic expressions are equivalents; and an operational semantics, consisting of reduction rules, which delineate the potential changes in the system following a communication event. The use of the calculus for modeling of biochemical systems is explained in the text, highlighting the syntax, and the basic communication rule, COMM.