Technical Report CoSBi 09/2006

Beta-binders with Static Compartments

Maria Luisa Guerriero
Universitá di Trento
guerrier@dit.unitn.it

Corrado Priami
CoSBi and Universitá di Trento
priami@cosbi.eu

Alessandro Romanel
CoSBi and Universitá di Trento
romanel@cosbi.eu

Abstract

We investigate the modeling of biological systems with static compartments through Beta-binders, a recently developed process calculus. Biological entities are represented as bio-processes and the calculus is extended with the notion of compartment. Entities can either be internal to compartments or reside on compartment borders. Movement in and out of compartments is requested by internal objects and mediated by border objects. The extended calculus is equipped with the notion of locality, and various kinds of relations between actions are defined. Moreover, we compare our proposal with similar formalisms and we show how to use the proposed calculus for modeling and analyzing the cAMP-signaling pathway in OSNs.

1 Introduction

Compartments are present in all biological systems: a cell is a compartment, which in turn contains other compartments (the most important of which is the nucleus). Compartments are fundamental for the evolution of biological systems, because they provide a means for isolating their content from the external environment, still allowing some exchange of information, mainly through membrane proteins.

Several languages have been proposed to model biological compartments (*Brane calculi* [1], *BioAmbients* [2], *Beta-binders* [3, 4]). All of them have some differences in the considered notion of compartment and in the kinds of operations allowed (see Sect. 6 for a discussion). We focus here on Beta-binders, a process calculus with a two level syntax. The main objects are called bio-processes that are boxes with typed interfaces and whose behavior is driven by simplified π-calculus [5] like processes that they enclose.

In Beta-binders the nesting of boxes is not allowed, but typed interfaces ensures that a virtual form of nesting can be represented. Modeling complex hierarchies, however, is quite a difficult task that is simplified by our current proposal. We rely on a general interpretation of bio-processes as structured communicating objects and we propose an extension with the notion of *static compartments*, that permits an intuitive representation of hierarchical structures, still forbidding the explicit nesting of boxes.

The structure of compartments and boxes allows us to consider *spatial* relations between events, e.g. the location where a protein-protein interaction occurs. Therefore, we enrich the calculus with some locality relations, both on compartments and on boxes. We adopt the transition system-based technique used in [6, 7] to define a locality relation for the π-calculus. Ex-

amples of use of locality relations in modeling bio-chemical systems are in [8].

In the next section we briefly describe a slightly variant of Beta-binders. In Sect. 3 we introduce the notion of compartments and in Sect. 4 we present the labeled semantics of Beta-binders; in Sect. 5 some locality relations are defined. In Sect. 6 we compare our proposal with some existing works, while the application of our proposal to a model of the cAMP-signaling pathway in OSNs is shown in Sect. 7. Finally some concluding remarks are presented.

2 Beta-binders

Beta-binders [3, 4] is a bio-inspired process calculus developed for better adhere to the structure and dynamics of biological systems. By introducing the concept of *affinity*, the calculus relaxes the *key-lock* model of interaction, commonly assumed in classical process calculi, and hence it permits to model more correctly domains and interactions between enzymes and small molecules based on their types and affinities. In Beta-binders, *pi-processes* are encapsulated into *boxes* with interaction capabilities, called also *bio-processes*. Like the π-calculus, Beta-binders is based on the notion of *naming*. Thus, we assume the existence of a countably infinite set $\mathcal{N}$ of names (ranged over by lower-case letters).

The processes wrapped into boxes, also called *pi-processes*, are given by the following context free grammar:

$$P \ ::= \ \mathsf{nil} \ \big| \ \pi.\, P \ \big| \ P|P \ \big| \ !\,\pi.P$$
$$\pi \ ::= \ x(y) \ \big| \ \overline{x}\langle y\rangle \ \big| \ \mathsf{expose}(x,\, \Gamma) \ \big| \ \mathsf{hide}(x) \ \big| \ \mathsf{unhide}(x) \ .$$

The π-calculus syntax is enriched by the last three prefixes for π to manipulate the interaction *sites* of the boxes. The object y of the input prefix $x(y)$ as well as the x of the $\mathsf{expose}(x,\, \Gamma)$ prefix act as binding occurrences. Hence we can define free fn and bound bn names as usual. Bio-processes are defined as pi-processes prefixed by specialized binders that represent interaction capabilities. An *elementary beta binder* has the form $\beta(x : \Gamma)$ (active) or $\beta^{\mathrm{h}}(x : \Gamma)$ (hidden) where the name x is the subject of the beta binder and Γ represents the type of x. With $\widehat{\beta}$ we denote either β or β^{h}. A *well-formed* beta binder (ranged over by $\mathbf{B}$, $\mathbf{B}_1$, $\mathbf{B}'$, $\cdots$) is a non-empty string of elementary beta binders whose subjects and types are all distinct. The function $\mathsf{sub}(\mathbf{B})$ returns the set of all the beta binder subjects in $\mathbf{B}$. Moreover, $\mathbf{B}^*$ denotes either a well-formed beta binder or the empty string. The function $\alpha(\Delta, \Gamma) \to \mathbb{R}$ returns the affinity of the types Δ and Γ. Types are

any algebraic structure for which it exists a decidable equality procedure. Hereafter, we also assume that substitution is not defined over the elements in a type. Note also that we do not have restriction in pi-processes and ! is guarded. This choice is done to adhere to the implementation of the language developed so far [9], but the general case works perfectly with the following development in this paper.

Bio-processes (ranged over by B, B_1, B', $\cdots$) are generated by the following context free grammar:

$$B \ ::= \ \mathsf{Nil} \ \big| \ \mathbf{B}\big[\, P \,\big] \ \big| \ B \parallel B \ .$$

The system is a parallel composition of bio-processes that can be either the deadlock bio-process Nil or the elementary bio-process $\mathbf{B}\big[\, P \,\big]$. The semantics of bio-processes is given in [3] in terms of a *reduction relation* ($\longrightarrow$), which uses a *structural congruence relation* ($\equiv$). We postpone the formal definitions of these relations to the next sections. For their standard definitions, see [3].

3 Compartments

The way in which a biological system is modeled here with Beta-binders is a composition of boxes, where each box represents a biological entity. Although the nesting of boxes is forbidden, the typing for sites and the operational semantics ensures that a virtual form of nesting can be represented [4]. This model might be too abstract, but it has been chosen to keep the formalism as simple as possible.

Consider for example the system

$$S \ = B \ \big\| \ B' \ \big\| \ B''$$

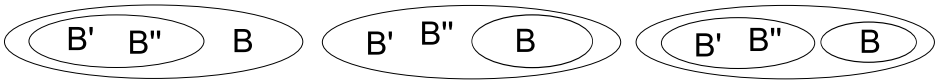
where:

- $B = \beta(s : \Delta_2)\big[\ s(v)\,.\,R\ \big]$
- $B' = \beta(x : \Delta_0)\big[\ \overline{x}\langle z\rangle\ .\ \mathsf{hide}(x)\,.\,P\ \big]$
- $B'' = \beta(y : \Delta_1)\big[\ y(v)\,.\,Q\ \big]$

and where $\alpha(\Delta_0, \Delta_1) > Th \wedge \alpha(\Delta_j, \Delta_2) < Th$ with $j \in \{0, 1\}$.[1] The affinity between the types exposed by the boxes could give us an a idea of how the

[1]The value Th represents a context dependent threshold over which two types are considered compatible.

boxes are grouped in compartments. In fact, we could imagine that the first and the second boxes are in the same compartment and that the third box is in another one. However, this kind of virtual nesting is ambiguous and for each defined system several different hierarchical structures can be deduced. In fact all the following three compartmentalization would be valid:



Moreover, consider the movement of objects across compartments. Since types encode compartments, moving an object from a compartment to another one means changing the types of the sites properly, using sequences of hide, unhide and expose operations. As the complexity of the model grows, the number of necessary actions makes this approach not practical and difficult to manage.

For these reasons we decided to introduce a finer and more explicit notion of compartments. Since we do not want to diverge from the original language, we decided to maintain the representation of systems as parallel composition of bio-processes, enriching statically them with labels acting as unique names which specify their location.

3.1 The Abstraction

Our goal is to provide a simple framework for modeling systems with static compartments and movements of components across compartments. A component is a structured object that can interact with other components through an affinity interaction model. Moreover, the movement of components between compartments is mediated by other components lying on compartment borders. From a biological point of view this can be seen as a system where molecules and complexes can change compartment through interaction with transmembrane proteins. However, since compartmentalization and movement of components across compartments play a critical role in computational systems, our approach can be applied in different contexts and at different levels of abstraction.

3.2 Static Compartment Hierarchy

Consider the system represented in Fig. 1(a). There is an outside compartment (S) that represents the whole system. The system contains three subcompartments (A, B, C). Moreover, the compartment B contains another compartment (D). The rectangles and the triangles are the components of

the system. In particular, rectangles represent components internal to compartments, while triangles represent components that reside on compartment borders. We call *i-components* the internal components and *b-components* the border ones. We introduce the distinction between i- and b-components to be closer to real biological systems, in which objects residing on membranes and objects residing inside compartments have different and specific functions.

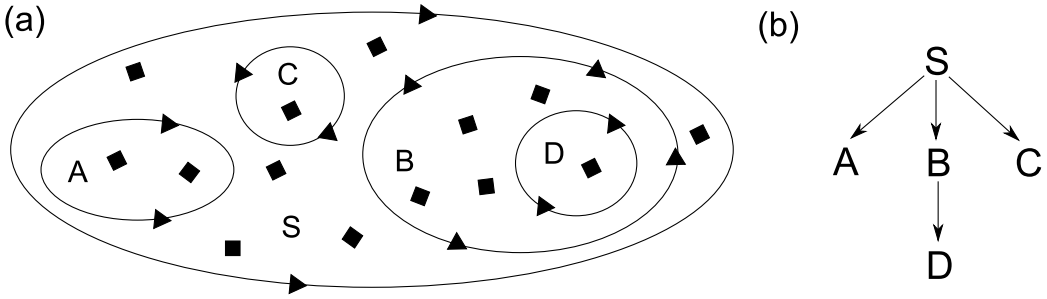


Figure 1: (a) System with static compartments; (b) The tree representation of the hierarchical structure of the compartments.

The static hierarchical structure of the system can be seen as a tree (Fig. 1(b)) where the nodes represent the compartments, and the numbers on the edges represent the numbering of the children.

Instead of using specific labels (e.g. S, A, ...) we can identify each compartment of the system with a sequences of natural numbers representing the position of the compartment inside the tree structure of the system, similarly to the Dewey's indexes. Thus, the compartments of the system in Fig. 1 can be identified with the following sequences:

$$S \to 0 \qquad A \to 0,0 \qquad B \to 0,1 \qquad C \to 0,2 \qquad D \to 0,1,0 \ .$$

Since each component of the system is represented by a bio-process and resides in a particular compartment, we modify the syntax of Beta-binders by labeling bio-processes with the identifier of the compartment in which the bio-process resides. Moreover, to distinguish between components lying inside a compartment and components lying on compartment borders, we add to each bio-process a special marker representing the component type. Formally, the definition of bio-processes is modified as follows:

$$B \ ::= \ \mathsf{Nil} \mid \mathbf{B}\big[\, P \,\big]_s^{\kappa} \mid B \parallel B \qquad\qquad \kappa \ ::= \ n \mid \kappa, n \ .$$

where $n \in \mathbb{N}$ specifies the position in the static structure and $s \in \{\mathrm{i}, \mathrm{b}\}$ denotes whether the component is an internal or a border one. As an example, a component lying on the border of the compartment D is represented with a bio-process $B = \mathbf{B}\big[\, P \,\big]_{\mathrm{b}}^{0,1,0}$.

3.3 Movements across Compartments

An i-component can move across a compartment border only through the interaction with a b-component residing on that border. This assumption mimics the role of transmembrane proteins in biological compartments. Since the calculus is based on a binary synchronous communication model, we still use affinity to mediate movement. Therefore, we modify the syntax of Beta-binders by adding the following new complementary actions prefixes:

$$\pi \ ::= \cdots \ \big| \ \mathsf{move}(x) \ \big| \ \mathsf{in}(x) \ \big| \ \mathsf{out}(x)$$

where $x \in \mathcal{N}$. The move action synchronizes with in or out actions, thus giving to i-components the ability to move across compartment borders, and b-components the ability to control the flow direction. As an example, consider the system in Fig. 2(a), described in Beta-binders by the bio-process:

$$S \ = \ (B_1 = \mathbf{B}_1\big[\, P_1 \,\big]^0_{\mathrm{i}}) \parallel (B_2 = \mathbf{B}_2\big[\, P_2 \,\big]^{0,0}_{\mathrm{b}}) \ .$$

Intuitively, B_1 can move into the sub-compartment interacting through a complementary $\mathsf{move}(x)/\mathsf{in}(y)$ action with B_2, where x and y are subjects of binders with affine types. The new configuration of the system (Fig. 2(b)) after the movement of B_1, is described in Beta-binders by the bio-process:

$$S' \ = \ (B'_1 = \mathbf{B}_1\big[\, P'_1 \,\big]^{0,0}_{\mathrm{i}}) \parallel (B'_2 = \mathbf{B}_2\big[\, P'_2 \,\big]^{0,0}_{\mathrm{b}}) \ .$$

The detailed semantics is presented in the next section.

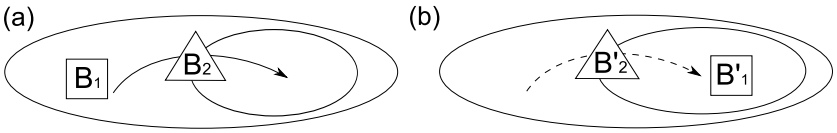


Figure 2: Example of movement across compartment.

4 Labeled Semantics

To introduce locality relations we first enrich the language and its semantics with labels that allow us to uniquely identify the location of bio-processes and compartments.

We define $\vartheta \in \{\|_0, \|_1\}^*$ and we use it to label bio-processes. We statically replace each bio-process $\mathbf{B}\big[\, P \,\big]^\kappa_s$ with a labeled process $\vartheta\mathbf{B}\big[\, P \,\big]^\kappa_s$ (where ϑ provides a linear encoding of the syntactical location of the sub-tree of $\mathbf{B}\big[\, P \,\big]^\kappa_s$

in the syntax tree of the whole system). We chose this approach to take advantage of the syntax of the calculus and ease the implementation of the naming structure. We could have used any unique name generator just to distinguish the locations of bio-processes.

For instance, the bio-process $\beta^{\mathrm{h}}(y:\Sigma)\big[\, P_0 \big| P_1 \,\big]_{s_0}^{\kappa_0} \parallel \beta(z:\Sigma)\,\big[\, Q_0 \big| Q_1 \,\big]_{s_1}^{\kappa_1}$ is mapped to $\|_0\,\beta^{\mathrm{h}}(y:\Sigma)\big[\, P_0 \big| P_1 \,\big]_{s_0}^{\kappa_0} \parallel \|_1\,\beta(z:\Sigma)\,\big[\, Q_0 \big| Q_1 \,\big]_{s_1}^{\kappa_1}$.

Each transition is labeled by a pair $\phi = \langle \theta ; \kappa \rangle$, where κ is defined as in Sect. 3.2 and θ is defined by the following BNF-like grammar:

$$\begin{array}{rcl} \theta & ::= & \vartheta\mu \;\big|\; \vartheta\rho \;\big|\; \vartheta\langle x(w), \overline{x}\langle z\rangle\rangle \;\big|\; \vartheta\langle \|_j \vartheta_0\, {}'x(w)\backslash, \|_{1-j}\vartheta_1\, {}'\overline{y}\langle z\rangle\backslash\rangle \;\big| \\ & & \vartheta\,\langle \|_0\vartheta_0 \mathsf{join}\, P_0, \|_1\vartheta_1 \mathsf{join}\, P_1\rangle \;\big|\; \vartheta\langle \|_j\vartheta_0\psi, \|_{1-j}\vartheta_1\, \mathsf{move}(x)\rangle \end{array}$$

where $\mu ::= a \big| c \big| d$ (with $a ::= \mathsf{expose}(x,\,\Gamma) \,\big|\, \mathsf{hide}(x) \,\big|\, \mathsf{unhide}(x)$, $c ::= x(w) \,\big|\, \overline{x}\langle y\rangle$, and $d ::= {}'x(w)\backslash \,\big|\, {}'\overline{x}\langle y\rangle\backslash$), $\rho ::= \mathsf{split}\langle P_0, P_1\rangle \,\big|\, \mathsf{join}\, P$ and $\psi ::= \mathsf{in}(x) \,\big|\, \mathsf{out}(x)$. The first pair of labels is used to denote intra-communications (communications within one bio-process), while the second one is used to denote inter-communications (communications between different bio-processes); the third and the fourth ones are used to denote join and movement operations, respectively. Note that the definition of d allows us to distinguish between the input/output actions used in intra-communications ($x(w)$ / $\overline{x}\langle y\rangle$) and the ones used in inter-communications (${}'x(w)\backslash$ / ${}'\overline{x}\langle y\rangle\backslash$). According to the definition of binders, y is a bound name in $x(y)$, in ${}'x(y)\backslash$ and in $\mathsf{expose}(y,\,\Gamma)$.

We introduce two new sets of labels, with metavariable γ and δ respectively, that will be useful in the following:

$$\gamma ::= a \;\big|\; \langle c_0, c_1\rangle \qquad\qquad \delta ::= d \;\big|\; \psi \big|\; \mathsf{move}(x) \quad .$$

Definition 1 *The* structural congruence *over pi-processes, denoted by* $\equiv$*, is the smallest relation which satisfies the laws in Table 1(a). The* structural congruence *over bio-processes is identified by two relations, denoted respectively by* $\equiv$ *and* $\equiv_{\mathrm{c}}$*, that are the smallest ones which satisfy the laws in Table 1(b).*

Definition 2 *The* reduction relation $\longrightarrow$ *is the smallest relation over bio-processes defined by the axioms and rules in Table 2.*

As in [3], f_{join} and f_{split} functions are user defined λ-calculus functions which describe the aggregation and disaggregation of boxes and depend on the structure of bio-processes.

Table 1: Laws for structural congruence.

(a) Pi-processes	(b) Boxes and bio-processes
	$\mathbf{B}\big[\, P_1 \,\big] \equiv \mathbf{B}\big[\, P_2 \,\big]$ provided $P_1 \equiv P_2$
$P_1 \equiv P_2$, provided P_1 α-converse of P_2	$\mathbf{B}^* \hat{\beta}(x:\Gamma)\big[\, P \,\big] \equiv \mathbf{B}^* \hat{\beta}(y:\Gamma)\big[\, P\{y/x\} \,\big]$ provided y fresh in the system
$P_1 \big| (P_2 \big| P_3) \equiv (P_1 \big| P_2) \big| P_3$	$\mathbf{B_1}\mathbf{B_2}\big[\, P \,\big] \equiv \mathbf{B_2}\mathbf{B_1}\big[\, P \,\big]$
	$B_1 \parallel B_2 \equiv_{\mathrm{c}} B_2 \parallel B_1$
$P_1 \big| P_2 \equiv P_2 \big| P_1$	$B_1 \parallel (B_2 \parallel B_3) \equiv_{\mathrm{c}} (B_1 \parallel B_2) \parallel B_3$
	$B \parallel \mathsf{Nil} \equiv_{\mathrm{c}} B$
$P \big| \, \mathsf{nil} \equiv P$	$\vartheta \mathbf{B}_1 \big[\, P_1 \,\big]_s^\kappa \equiv_{\mathrm{c}} \vartheta \mathbf{B}_2 \big[\, P_2 \,\big]_s^\kappa$ provided $\mathbf{B}_1\big[\, P_1 \,\big] \equiv \mathbf{B}_2\big[\, P_2 \,\big]$

5 Locality Relations

To define some locality relations on Beta-binders transitions we first need two auxiliary functions for each transition ϕ: $\mathsf{act}(\phi)$ specifies the action executed, and $\mathsf{comp}(\phi)$ specifies the compartment in which the action is executed.

$$\mathsf{act}(\langle \theta, \kappa \rangle) = \theta \qquad\qquad \mathsf{comp}(\langle \theta, \kappa \rangle) = \kappa \ .$$

We first define some relations concerning compartments (i.e. the compartments in which the actions triggering the transitions occur). Finally, we consider the level of bio-processes (i.e. the bio-processes involved in the transitions).

Based on the definition of localities described in [10], compartments are *static* localities: they do not change dynamically during execution, and hence they represent the sites in which events occur. Therefore, the relations introduced in the next section refer to the relative positions of the considered transitions. ϑ labels of bio-processes are, instead, *dynamical* localities: in fact they are built incrementally when actions are performed (actually, only split operations modify the labels by adding sublabels to the labels of the created bio-processes), and hence they represent, for each action, the ones that locally precede it.

Table 2: Axioms and rules for the reduction relation.

(intra)
$$\frac{P \equiv x(w) \,.\, P_0 \,\Big|\, \overline{x}\langle z\rangle \,.\, P_1 \,\Big|\, P_2}{\vartheta \mathbf{B}\Big[\ P\ \Big]_s^{\kappa} \xrightarrow{\langle \vartheta\langle x(w),\overline{x}\langle z\rangle\rangle;\kappa\rangle} \vartheta \mathbf{B}\Big[\ P_0\{z/w\} \,\Big|\, P_1 \,\Big|\, P_2\ \Big]_s^{\kappa}}$$

(inter)
$$\frac{P \equiv x(w) \,.\, P_1 \,\Big|\, P_2 \qquad Q \equiv \overline{y}\langle z\rangle \,.\, Q_1 \,\Big|\, Q_2}{X \xrightarrow{\langle \vartheta\langle \|_j \vartheta_0\, {}'x(w)\backprime, \|_{1-j} \vartheta_1\, {}'\overline{y}\langle z\rangle\backprime\rangle;\kappa_l\rangle} Y}$$, where

$$X = \vartheta\|_j\vartheta_0\, \beta(x:\Gamma)\, \mathbf{B}_0^*\Big[\ P\ \Big]_{s_0}^{\kappa_0} \,\Big\|\, \vartheta\|_{1-j}\vartheta_1\, \beta(y:\Sigma)\, \mathbf{B}_1^*\Big[\ Q\ \Big]_{s_1}^{\kappa_1},$$
$$Y = \vartheta\|_j\vartheta_0\, \beta(x:\Gamma)\, \mathbf{B}_0^*\Big[\ P_1\{z/w\} \,\Big|\, P_2\ \Big]_{s_0}^{\kappa_0} \,\Big\|\, \vartheta\|_{1-j}\vartheta_1\, \beta(y:\Sigma)\, \mathbf{B}_1^*\Big[\ Q_1 \,\Big|\, Q_2\ \Big]_{s_1}^{\kappa_1}$$
provided $\alpha(\Gamma,\Sigma) \geq Th$ and $(\kappa_l = \kappa_{1-l} \vee (\kappa_l = \kappa_{1-l}, n \wedge s_l = \mathrm{b} \wedge s_{1-l} = \mathrm{i}))$

(expose)
$$\frac{P \equiv \mathsf{expose}(x,\,\Gamma) \,.\, P_1 \,\Big|\, P_2}{\vartheta \mathbf{B}\Big[\ P\ \Big]_s^{\kappa} \xrightarrow{\langle \vartheta\, \mathsf{expose}(x,\,\Gamma);\kappa\rangle} \vartheta\, \mathbf{B}\, \beta(y:\Gamma)\Big[\ P_1\{y/x\} \,\Big|\, P_2\ \Big]_s^{\kappa}}$$, y fresh in the system

(hide)
$$\frac{P \equiv \mathsf{hide}(x) \,.\, P_1 \,\Big|\, P_2}{\vartheta \mathbf{B}^*\, \beta(x:\Gamma)\Big[\ P\ \Big]_s^{\kappa} \xrightarrow{\langle \vartheta\, \mathsf{hide}(x);\kappa\rangle} \vartheta \mathbf{B}^*\, \beta^{\mathrm{h}}(x:\Gamma)\Big[\ P_1 \,\Big|\, P_2\ \Big]_s^{\kappa}}$$

(unhide)
$$\frac{P \equiv \mathsf{unhide}(x) \,.\, P_1 \,\Big|\, P_2}{\vartheta \mathbf{B}^*\, \beta^{\mathrm{h}}(x:\Gamma)\Big[\ P\ \Big]_s^{\kappa} \xrightarrow{\langle \vartheta\, \mathsf{unhide}(x);\kappa\rangle} \vartheta \mathbf{B}^*\, \beta(x:\Gamma)\Big[\ P_1 \,\Big|\, P_2\ \Big]_s^{\kappa}}$$

(join)
$$\vartheta\vartheta_0\mathbf{B}_0\Big[\ P_0\ \Big]_s^{\kappa} \,\Big\|\, \vartheta\vartheta_1\mathbf{B}_1\Big[\ P_1\ \Big]_s^{\kappa} \xrightarrow{\langle\theta;\kappa\rangle} \vartheta\vartheta_0\mathbf{B}\Big[\ P_0\sigma_0 \,\Big|\, P_1\sigma_1\ \Big]_s^{\kappa} \,\Big\|\, \vartheta\vartheta_1\, \mathsf{Nil}$$
where $\theta = \vartheta\,\langle \vartheta_0 \mathsf{join}\, P_0, \vartheta_1 \mathsf{join}\, P_1\rangle$, $\vartheta_0 = \|_0\vartheta_0'$, $\vartheta_1 = \|_1\vartheta_1'$
provided $f_{\mathsf{join}}(\mathbf{B}_0, \mathbf{B}_1, P_0, P_1) = (\mathbf{B}, \sigma_0, \sigma_1)$

(split)
$$\vartheta\mathbf{B}\Big[\ P_0 \,\Big|\, P_1\ \Big]_s^{\kappa} \xrightarrow{\langle \vartheta\, \mathsf{split}\langle P_0, P_1\rangle;\kappa\rangle} \vartheta\|_0\mathbf{B}_0\Big[\ P_0\sigma_0\ \Big]_s^{\kappa} \,\Big\|\, \vartheta\|_1\mathbf{B}_1\Big[\ P_1\sigma_1\ \Big]_s^{\kappa}$$
provided $f_{\mathsf{split}}(\mathbf{B}, P_0, P_1) = (\mathbf{B}_0, \mathbf{B}_1, \sigma_0, \sigma_1)$

(in)
$$\frac{P \equiv \mathsf{in}(x) \,.\, P_1 \,\Big|\, P_2 \qquad Q \equiv \mathsf{move}(x) \,.\, Q_1 \,\Big|\, Q_2}{X \xrightarrow{\langle \vartheta\langle \|_j \vartheta_0\, \mathsf{in}(x), \|_{1-j} \vartheta_1\, \mathsf{move}(x)\rangle;\kappa,n\rangle} Y}$$, provided $\alpha(\Gamma,\Sigma) \geq Th$ and where

$$X = \vartheta\|_j\vartheta_0\, \beta(x:\Gamma)\, \mathbf{B}_0^*\Big[\ P\ \Big]_{\mathrm{b}}^{\kappa,n} \,\Big\|\, \vartheta\|_{1-j}\vartheta_1\, \beta(y:\Sigma)\, \mathbf{B}_1^*\Big[\ Q\ \Big]_{\mathrm{i}}^{\kappa},$$
$$Y = \vartheta\|_j\vartheta_0\, \beta(x:\Gamma)\, \mathbf{B}_0^*\Big[\ P_1 \,\Big|\, P_2\ \Big]_{\mathrm{b}}^{\kappa,n} \,\Big\|\, \vartheta\|_{1-j}\vartheta_1\, \beta(y:\Sigma)\, \mathbf{B}_1^*\Big[\ Q_1 \,\Big|\, Q_2\ \Big]_{\mathrm{i}}^{\kappa,n}$$

(out)
$$\frac{P \equiv \mathsf{out}(x) \,.\, P_1 \,\Big|\, P_2 \qquad Q \equiv \mathsf{move}(x) \,.\, Q_1 \,\Big|\, Q_2}{X \xrightarrow{\langle \vartheta\langle \|_j \vartheta_0\, \mathsf{out}(x), \|_{1-j} \vartheta_1\, \mathsf{move}(x)\rangle;\kappa,n\rangle} Y}$$, provided $\alpha(\Gamma,\Sigma) \geq Th$ and where

$$X = \vartheta\|_j\vartheta_0\, \beta(x:\Gamma)\, \mathbf{B}_0^*\Big[\ P\ \Big]_{\mathrm{b}}^{\kappa,n} \,\Big\|\, \vartheta\|_{1-j}\vartheta_1\, \beta(y:\Sigma)\, \mathbf{B}_1^*\Big[\ Q\ \Big]_{\mathrm{i}}^{\kappa,n},$$
$$Y = \vartheta\|_j\vartheta_0\, \beta(x:\Gamma)\, \mathbf{B}_0^*\Big[\ P_1 \,\Big|\, P_2\ \Big]_{\mathrm{b}}^{\kappa,n} \,\Big\|\, \vartheta\|_{1-j}\vartheta_1\, \beta(y:\Sigma)\, \mathbf{B}_1^*\Big[\ Q_1 \,\Big|\, Q_2\ \Big]_{\mathrm{i}}^{\kappa}$$

(redex)
$$\frac{B \xrightarrow{\phi} B'}{B \,\Big\|\, B'' \xrightarrow{\phi} B' \,\Big\|\, B''}$$

(struct)
$$\frac{B_1 \equiv_{\mathrm{c}} B_1' \qquad B_1' \xrightarrow{\phi} B_2}{B_1 \xrightarrow{\phi} B_2}$$

5.1 Compartments Locality Relations

In this section a bunch of significant locality relations between transitions is defined, assuming a computation $B_0 \xrightarrow{\phi_0} B_1 \xrightarrow{\phi_1} \cdots \xrightarrow{\phi_n} B_{n+1}$, according to the relative position of the compartments in which the transitions occur. We end this section by discussing how our relations can be useful in biological systems.

Definition 3 (Same-compartment relation) *We say that ϕ_n has a same-compartment dependency on ϕ_h (denoted with $\phi_h \asymp \phi_n$) if $h < n$ and $\mathsf{comp}(\phi_h) = \mathsf{comp}(\phi_n)$.*

With this relation we underline the fact that the two actions ϕ_h and ϕ_n occur in the same compartment.

Definition 4 (Son-father relation) *We say that ϕ_n has a son-father dependency on ϕ_h (denoted with $\phi_h \curlywedge \phi_n$) if $h < n$ and $(\mathsf{comp}(\phi_n), m) = \mathsf{comp}(\phi_h)$ ($m \in \mathbb{N}$).*

This means that the compartment in which the action ϕ_h occurs is the son of the compartment in which the action ϕ_n occurs.

Definition 5 (Father-son relation) *We say that ϕ_n has a father-son dependency on ϕ_h (denoted with $\phi_h \curlyvee \phi_n$) if $h < n$ and $(\mathsf{comp}(\phi_h), m) = \mathsf{comp}(\phi_n)$ ($m \in \mathbb{N}$).*

In this case, the compartment in which the action ϕ_h occurs is the father of the compartment in which the action ϕ_n occurs.

Son-father and *father-son* relations can be easily generalized to *sub-compartment* and *super-compartment* relations respectively, by considering their transitive closures.

Definition 6 (Sub-compartment and super-compartment relations) *Let $\overline{\curlywedge} \triangleq (\curlywedge)^*$ be the transitive closure of $\curlywedge$. We say that ϕ_n has a sub-compartment dependency on ϕ_h if $\phi_h \,\overline{\curlywedge}\, \phi_n$.*

Let $\underline{\curlyvee} \triangleq (\curlyvee)^*$ be the transitive closure of $\curlywedge$. We say that ϕ_n has a super-compartment dependency on ϕ_h if $\phi_h \,\underline{\curlyvee}\, \phi_n$.

The relation $\overline{\curlywedge}$ means that the compartment in which the action ϕ_h occurs is a sub-compartment of the compartment in which the action ϕ_n occurs, and $\underline{\curlyvee}$ means that the compartment in which the action ϕ_n occurs is a sub-compartment of the compartment in which the action ϕ_h occurs.

Note that $\phi_h \mathbin{\overline{\wedge}} \phi_n \Rightarrow h < n$ and $(\mathsf{comp}(\phi_n), \kappa) = \mathsf{comp}(\phi_h)$, and that $\phi_h \mathbin{\underline{\vee}} \phi_n \Rightarrow h < n$ and $(\mathsf{comp}(\phi_h), \kappa) = \mathsf{comp}(\phi_n)$.

In the area of dynamical modeling of biological systems, locality relations can be useful for analyzing the spatial distribution of entities. For example, when observing transitions originated by an interesting event, we could be interested in investigating what happened in the same compartment before. The relation $\asymp$ can be used for that. Similarly, we could use the other relations to study what happened in super- or sub-compartments.

5.2 Inter-boxes Locality Relation

In this section we define the inter-boxes locality relation between pairs of transitions in a computation: an inter-boxes locality relation exists between an activity A and an activity B if A and B are executed by pi-processes in the same bio-process. Our labels can be used as unique names for the transitions as they are linearizations encoding the position of the prefixes and processes originating the transitions in the syntax tree.

Definition 7 (Direct inter-boxes locality relation) *Given a computation $B_0 \xrightarrow{\phi_0} B_1 \xrightarrow{\phi_1} \cdots \xrightarrow{\phi_n} B_{n+1}$, we say that ϕ_n has a direct inter-boxes locality dependency on ϕ_h if $h < n$ and $\mathsf{act}(\phi_h) \lessdot \mathsf{act}(\phi_n)$ can be derived by repeated applications of the following rules, where $j \in \{0, 1\}$.*

1. $\|_j\theta \lessdot \|_j\theta'$ *if* $\theta \lessdot \theta'$
2. $\gamma \lessdot \gamma'$
3. $\langle \|_j\vartheta_0\delta_0, \|_{1-j}\vartheta_1\delta_1\rangle \lessdot \langle \|_l\vartheta'_0\delta'_0, \|_{1-l}\vartheta'_1\delta'_1\rangle$
   if $((\|_j\vartheta_0 = \|_l\vartheta'_0 \wedge \|_{1-j}\vartheta_1 = \|_{1-l}\vartheta'_1) \vee (\|_j\vartheta_0 = \|_{1-l}\vartheta'_1 \wedge \|_{1-j}\vartheta_1 = \|_l\vartheta'_0))$.

The rules listed above are applied recursively to a pair of actions θ_h, θ_n in order to verify if there is an inter-boxes locality dependency between them. Since inter-boxes locality only concerns the bio-processes (and not their internal structure), the recursive step is implemented by removing the common prefixes of θ_h and θ_n through rule 1, as long as they are relative to the labels of bio-processes ($\|_0$ and $\|_1$). Then, at the end of the recursive steps (i.e. if θ_h and θ_n refer to the same bio-process), either rule 2 or rule 3 could be applied. Rule 2 states that actions on beta binders (i.e. expose, hide and unhide) and communications have an inter-boxes locality dependence on other actions on beta binders and other communications executed by pi-processes in the same bio-process. Rule 3 states that inter-communications

and transport operations have an inter-boxes locality dependence on other inter-communications and other transport operations if both operations are between the same bio-processes (i.e. each partner of one operation is executed by the same bio-process of one of the partners of the other operation).

Note that our mechanism is not affected by the associativity and commutativity of $\|$, because the ϑ labels are attached statically to processes and updated in the operational semantics by the rules that affect the structure of the system.

The definition of the inter-boxes locality relation between two transitions of a computation is obtained by taking into account the transitive closure of the direct inter-boxes locality relation.

Definition 8 (Inter-boxes locality relation) *Let* $< \triangleq (\lessdot)^*$ *be the transitive closure of* $\lessdot$*. Then, given a computation* $B_0 \xrightarrow{\phi_0} B_1 \xrightarrow{\phi_1} \cdots \xrightarrow{\phi_n} B_{n+1}$*, we say that* ϕ_n *has an inter-boxes locality dependency on* ϕ_h *if* $\mathsf{act}(\phi_h) < \mathsf{act}(\phi_n)$*.*

From a biological point of view, when observing an interesting action performed by a bio-process, we could be interested in investigating other actions performed before by the same bio-process. The relation $<$ can be used for that. Inter-boxes relation, together with compartments locality relations, can be useful for analyzing the spatial distribution of the actions executed by a bio-process.

6 Related Works

Several languages have been proposed to model biological compartments; the most common ones are *Brane calculi* [1] and *BioAmbients calculus* [2].

Differently from these calculi, the main aim of our work is to represent static compartments and movements of objects across them; hence, we just give an informal comparison on the usability of the languages with respect to different biological domains.

6.1 Brane Calculi

The main feature of *Brane Calculi* is that membranes are considered active elements and hence the whole computation happens *on* membranes: they can move, merge, split, enter in and exit from other membranes. A system is represented as a set of nested membranes, and a membrane as a set of actions; actions carry out the mentioned membrane transformations. The

main events that can be directly modeled are phagocytosis (the engulfment of a membrane by another one) and exocytosis (the expulsion of a membrane by another one). Moreover, operations such as mitosis (the splitting of one membrane in two membranes) and mating (the merging of two membranes) can also be described. On-membrane and cross-membrane communications can also be modeled.

Being Brane Calculi primarily concerned on membrane interaction, it permits to easily model membrane operations; on the other hand, it does not take the internal structure of membrane-bound compartments into account, therefore it is not easy to describe events such as protein activation, phosphorylation, etc. Beta-binders, instead, is primarily focused on interaction between internal processes, hence compartments are used to describe the relative positions of the interacting bio-processes and to forbid interactions between processes which are in different compartments; hence, compartments (i.e. compartmental membranes) are static containers (it is not possible to create, destroy, or merge them) and bio-processes (i.e. proteins) can move across their borders. Therefore, operations involving membrane fusion, such as phagocytosis, exocytosis, mitosis and mating, cannot be modeled in Beta-binders by operations on compartments. For example, if compartments represent cells, it is not possible to merge compartments to model cell mating; however, we point out that it is sufficient to change the level of abstraction, i.e. to represent cells with bio-processes and use f_{join} and f_{split} functions to model such operations. Events that are not directly related to cellular membranes (e.g. phosphorylation) can easily be modeled by standard Beta-binders communications and operations on bio-processes interfaces.

Finally, in Brane Calculi everything is interpreted as a membrane, which means that membrane-bound cellular compartments (e.g. cells and organelles) and molecular compartments (e.g. proteins) are modeled in the same way: this seems a bit odd from a conceptual point of view. The proposed Beta-binders extension, instead, provides a double layer of compartmentalization (bio-processes and compartments), which permits a clear distinction between the two compartments types: when modeling cellular processes, cellular compartments are represented by compartments, while proteins are represented by bio-processes.

6.2 BioAmbients

BioAmbients calculus is an extension of the work described in [11], enriched with a concept of compartments similar to the one of *Ambient calculus* [12]. A system is represented as a set of nested ambients, and an ambient is a bounded compartment containing processes whose actions specify the evo-

lution of the system. Ambients can enter in and exit from other ambients (phagocytosis and exocytosis) and they can merge together (mating). π-calculus-style communications can occur within an ambient, between sibling ambients, and between father-child ambients.

Similarly to membranes in Brane calculi, ambients are used to represent both membrane-bound cellular compartments and molecular compartments (proteins and protein complexes). Moreover, BioAmbients does not provide an explicit way to model membrane proteins (they are implicitly considered by the primitives through which an ambient can allow another one to enter, exit or merge with). Hence, it is not easy to model complex interactions between membrane proteins and internal proteins: the movement of an ambient in or out from another one is obtained by complementary actions executed by the two ambients. For example, there is no way to describe the expulsion of a molecule (an ambient "m") from a cell (an ambient "c" containing "m"), mediated by a membrane protein (an ambient "p" lying inside "c"). This, instead, can be easily done in Beta-binders. Finally, in BioAmbients it is not possible to move processes which are not lying in some ambient; hence, in order to describe the movement of small molecules across cellular membranes, they need to be enclosed within an ambient.

As previously stated, operations involving membrane fusion cannot be modeled in Beta-binders by operations on compartments (though they can be modeled by operations on bio-processes). In BioAmbients, instead, it is easy to model them (except mitosis, whose description in BioAmbients is not straightforward).

Entities in the same compartment can interact in BioAmbients though a *local* communication on a channel. In Beta-binders interaction is also done through inter-communications if the interfaces of the two entities are compatible.

Finally, it is not easy to model in BioAmbients events such as protein activations (in particular multi-step chains of proteins activations), which, instead, can be easily done in the proposed Beta-binders extension (as shown in the example described in the following section).

7 The cAMP-Signaling Pathway in OSNs

In this section we present how to use the extended Beta-binders formalism for modeling a biological system that includes membranes and membrane proteins. In particular, we model the cAMP-signaling pathway in olfactory sensory neurons (OSNs). The pathway describes how G protein-coupled receptors indirectly modulate the activity of ion channels via the action of

second messengers (Fig. 3).

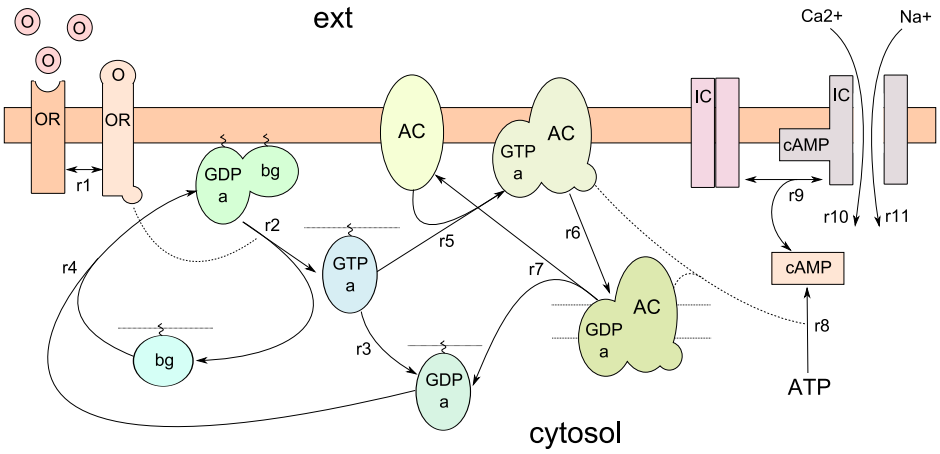


Figure 3: The cAMP-signaling pathway in OSNs.

An odorant ligand O can bind with an odorant receptor OR through the reversible reaction $r1$, activating it. The active OR stimulates ($r2$) the G-protein GDP$\alpha\beta\gamma$ (denoted with GDPabg), causing the dissociation of the trimer in two active subunits GTPa and bg. At this point, GTPa can either hydrolyze ($r3$), returning GDPa, or activate the adenylyl cyclase (AC), his target protein ($r5$). If the reaction $r3$ takes place, the subunit GDPa reassociates with the subunit bg ($r4$). If, instead, the reaction $r5$ takes place, the activation of AC produces, through the synthesis of ATP ($r8$), an increase in the concentration of the second messenger cAMP. A cAMP molecule can open, through a reversible binding ($r9$), the ion-channel IC, allowing Na^+ and Ca^{2+} molecules to enter. However, the hydrolysis of GTP to GDP causes GTPa to dissociate from AC and reassociate with bg. For a more detailed description of the pathway we refer the reader to [13].

Table 3 shows the specification of the Beta-binders model of the presented pathway.[2] Moreover, $\alpha(\Delta_k, \Delta_j) > Th$ iff $(k = j) \vee (k = 4 \wedge j \in \{5, 6\})$. The parallel composition of all the defined bio-processes

$$S = B_{\mathrm{O}} \parallel B_{\mathrm{OR}} \parallel B_{\mathrm{G}} \parallel B_{\mathrm{AC}} \parallel B_{\mathrm{ATP}} \parallel B_{\mathrm{IC}} \parallel B_{\mathrm{Na^+}} \parallel B_{\mathrm{Ca^{2+}}}$$

represents the initial configuration of the system, denoted with S. All the communications enabled by the bio-processes represent the reactions $r1, \cdots, r11$ shown in Fig. 3, and all the intermediate configurations that the system S

[2] With $\|_k^n$ we indicate the sequence $\underbrace{\|_k \cdots \|_k}_{n}$.

can reach through the execution of communications represent all the possible configurations of the biological system.

Table 3: Specification of the model.

$\mathrm{O} = \overline{x}\langle z\rangle \,.\, x(z) \,.\mathrm{O}$
$\mathrm{OR} = x(z) \,.\, \mathsf{unhide}(y) \,.\, \mathsf{hide}(y) \,.\, \overline{x}\langle z\rangle \,.\mathrm{OR}$
$\mathrm{A} = \overline{y}\langle z\rangle \,.\mathrm{A}$
$\mathrm{GDP} = x(z) \,.\mathrm{GTP}$
$\mathrm{GTP} = y(z) \,.\mathrm{GDP}$
$\mathrm{P}_\alpha = \overline{y}\langle z\rangle \,.\mathrm{P}_\alpha$

$\mathrm{AC} = y(z) \,.\mathrm{AC}$
$\mathrm{cAMP} = \overline{x}\langle z\rangle \,.\, x(z) \,.\mathrm{cAMP}$
$\mathrm{IC} = x(z) \,.\, \mathsf{unhide}(y) \,.\, \mathsf{hide}(y) \,.\, \overline{x}\langle z\rangle \,.\mathrm{IC}$
$\mathrm{M} = \mathsf{in}(y) \,.\mathrm{M}$
$\mathrm{Na}^+ = \mathsf{move}(x) \,.\mathrm{Na}^+$
$\mathrm{Ca}^{2+} = \mathsf{move}(x) \,.\mathrm{Ca}^{2+}$

$B_{\mathrm{O}} = \|_0 \, \beta(x : \Delta_0)[\mathrm{O}]^0_{\mathrm{i}}$
$B_{\mathrm{OR}} = \|_1 \|_0 \, \beta(x : \Delta_0) \, \beta^{\mathrm{h}}(y : \Delta_1)[\mathrm{OR}|\mathrm{A}]^{0,0}_{\mathrm{b}}$
$B_{\mathrm{G}} = \|^2_1 \|_0 \, \beta(x : \Delta_1)[\mathrm{GDP}|\mathrm{P}_\alpha|\mathrm{P}_{\beta\gamma}]^{0,0}_{\mathrm{b}}$
$B_{\mathrm{AC}} = \|^3_1 \|_0 \, \beta^{\mathrm{h}}(y : \Delta_2)[\mathrm{AC}]^{0,0}_{\mathrm{b}}$

$B_{\mathrm{ATP}} = \|^4_1 \|_0 \, \beta(x : \Delta_3) \, \beta(y : \Delta_2)[\overline{y}\langle z\rangle \,.\mathrm{cAMP}]^{0,0}_{\mathrm{i}}$
$B_{\mathrm{IC}} = \|^5_1 \|_0 \, \beta(x : \Delta_3) \, \beta^{\mathrm{h}}(y : \Delta_4)[\mathrm{IC}|\mathrm{M}]^{0,0}_{\mathrm{b}}$
$B_{\mathrm{Na}^+} = \|^6_1 \|_0 \, \beta(x : \Delta_5)[\mathrm{Na}^+]^0_{\mathrm{i}}$
$B_{\mathrm{Ca}^{2+}} = \|^7_1 \, \beta(x : \Delta_6)[\mathrm{Ca}^{2+}]^0_{\mathrm{i}}$

$f_{\mathsf{split}_{\mathrm{G}}}(B, P_0, P_1) \;=$
$\mathrm{if}(B[P_0|P_1] \equiv \beta(x : \Delta_1)[(\mathrm{GTP}|\mathrm{P}_\alpha)|\mathrm{P}_{\beta\gamma}])$
$\quad \mathrm{then}(\beta^{\mathrm{h}}(x : \Delta_1), \beta^{\mathrm{h}}(x : \Delta_1), id, id)$
$\quad \mathrm{else} \perp$

$f_{\mathsf{split}_{\mathrm{AC}}}(B, P_0, P_1) \;=$
$\mathrm{if}(B[P_0|P_1] \equiv \beta^{\mathrm{h}}(x : \Delta_1) \, \beta(y : \Delta_2)[(\mathrm{GDP}|\mathrm{P}_\alpha)|\mathrm{AC}]$
$\quad \mathrm{then}(\beta^{\mathrm{h}}(x : \Delta_1), \beta^{\mathrm{h}}(y : \Delta_2), id, id)$
$\quad \mathrm{else} \perp$

$f_{\mathsf{join}_{\mathrm{G}}}(B_0, B_1, P_0, P_1) \;=$
$\mathrm{if}(B_0[P_0] \equiv \beta^{\mathrm{h}}(x : \Delta_1)[\mathrm{GDP}|\mathrm{P}_\alpha] \wedge$
$\quad B_1[P_1] \equiv \beta^{\mathrm{h}}(x : \Delta_1)[\mathrm{P}_{\beta\gamma}])$
$\quad \mathrm{then}(\beta(x : \Delta_1), id, id)$
$\quad \mathrm{else} \perp$

$f_{\mathsf{join}_{\mathrm{AC}}}(B_0, B_1, P_0, P_1) \;=$
$\mathrm{if}(B_0[P_0] \equiv \beta^{\mathrm{h}}(x : \Delta_1)[\mathrm{GTP}|\mathrm{P}_\alpha] \wedge$
$\quad B_1[P_1] \equiv \beta^{\mathrm{h}}(y : \Delta_2)[\mathrm{AC}])$
$\quad \mathrm{then}(\beta^{\mathrm{h}}(x : \Delta_1) \, \beta(y : \Delta_2), id, id)$
$\quad \mathrm{else} \perp$

Now consider one of the possible computations, in which, starting from the initial configuration S, the ion-channel IC is activated, causing the entrance of a Ca^{2+} molecule:

$\phi_1 : \quad \langle\langle \|_1 \|_0 \, 'x(z)\backslash, \|_0 \, '\overline{x}\langle z\rangle\backslash\rangle; 0\rangle$
$\phi_2 : \quad \langle \|_1 \|_0 \, \mathsf{unhide}(y); 0, 0\rangle$
$\phi_3 : \quad \langle \|_1 \langle \|_0 \, '\overline{y}\langle z\rangle\backslash, \|_1 \|_0 \, 'x(z)\backslash\rangle; 0, 0\rangle$
$\phi_4 : \quad \langle \|^2_1 \|_0 \, \mathsf{split}\langle \mathrm{GTP} \mid \mathrm{P}_\alpha, \mathrm{P}_{\beta\gamma}\rangle; 0, 0\rangle$
$\phi_5 : \quad \langle \|^2_1 \, \langle \|^2_0 \mathsf{join} \; \mathrm{GTP} \mid \mathrm{P}_\alpha, \|_1 \|_0 \mathsf{join} \; \mathrm{AC}\rangle; 0, 0\rangle$
$\phi_6 : \quad \langle \|^2_1 \langle \|^2_0 \, 'y(z)\backslash, \|_1 \|_0 \, '\overline{y}\langle z\rangle\backslash\rangle; 0, 0\rangle$
$\phi_7 : \quad \langle \|^4_1 \langle \|_0 \, '\overline{x}\langle z\rangle\backslash, \|_1 \|_0 \, 'x(z)\backslash\rangle; 0, 0\rangle$
$\phi_8 : \quad \langle \|^5_1 \|_0 \, \mathsf{unhide}(y); 0, 0\rangle$
$\phi_9 : \quad \langle \|^5_1 \langle \|_0 \, \mathsf{in}(y), \|^2_1 \, \mathsf{move}(x)\rangle; 0, 0\rangle \;.$

By analyzing the computation with the locality relations previously defined, we can observe, for example, that ϕ_4 has a *father-son* dependency on ϕ_1 (denoted with $\phi_1 \curlyvee \phi_4$), while ϕ_6 has a *same-compartment* dependency on ϕ_9 (denoted with $\phi_6 \asymp \phi_9$).

From a biological point of view, these relations state that a spatial relation $\curlyvee$ and a spatial relation $\asymp$ exist, respectively, between the dissociation of the G-protein GTP$\alpha\beta\gamma$ and the activation of the receptor OR, and between the entrance of a Ca^{2+} molecule and the activation of the target protein AC.

8 Conclusions and Further Work

We extend Beta-binders with compartments and localities. Since the nesting of boxes is forbidden, modeling hierarchies of compartments in the standard version of the calculus is not primitive. We overcome this limitation by introducing the concept of static compartments. Compartments allow the representation of operations such as the movement of objects across compartment borders and the communication between internal and border objects.

Finally, some locality definitions have been introduced (both on compartments and on bio-processes/boxes) which can be useful when studying the spatial distributions of objects (and events) in complex systems.

Many further extensions are possible. One is to differentiate the various types of border objects: transmembrane proteins and those lying on the internal/external side of the membrane. The definition of the calculus should be slightly modified in order to take those differences into account. In addition to this, most proteins cannot freely move on the membrane, hence interactions between proteins on the same membrane are not always possible: the position of the proteins is important. This could be considered by introducing additional constraints, based on the position of bio-processes, which permits interactions solely between "near" (according to some definition of distance) bio-processes.

A simulator for the extended calculus is under implementation. This will allow us to test our framework on large scale biological problems.

References

[1] Cardelli, L.: Brane Calculi - Interactions of Biological Membranes. In: Proceedings of Workshop on Computational Methods in Systems Biology (CMSB'04). Volume 3082 of LNCS. (2005) 257–278

[2] Regev, A., Panina, E.M., Silverman, W., Cardelli, L., Shapiro, E.Y.: BioAmbients: an Abstraction for Biological Compartments. Theoretical Computer Science **325**(1) (2004) 141–167

[3] Priami, C., Quaglia, P.: Beta binders for biological interactions. In: Proceedings of Workshop on Computational Methods in Systems Biology (CMSB'04). Volume 3082 of LNCS. (2005) 20–33

[4] Priami, C., Quaglia, P.: Operational patterns in Beta-binders. Transactions on Computational Systems Biology **1** (2005) 50–65

[5] Milner, R.: Communicating and Mobile Systems: the π-Calculus Cambridge University Press, 1999.

[6] Degano, P., Priami, C.: Non-interleaving semantics for mobile processes. Theoretical Computer Science **216**(1–2) (1999) 237–270

[7] Degano, P., Gadducci, F., Priami, C.: Causality and Replication in Concurrent Processes. In: Proc. of Perspectives of System Informatics. Volume 2890 of LNCS. (2003) 307–318

[8] Curti, M., Degano, P., Priami, C., Baldari, C.T.: Modelling biochemical pathways through enhanced π-calculus. Theoretical Computer Science **325**(1) (2004) 111–140

[9] Romanel, A., Dematté, L., Priami, C.: The BetaSIM System. Technical Report TR-03-2007, The Microsoft Research - University of Trento Centre for Computational and Systems Biology, 2007.

[10] Bergstra, J.A., Ponse, A., Smolka, S.A., eds.: Handbook of Process Algebra. Elsevier (2001)

[11] Regev, A., Silverman, W., Shapiro, E.: Representation and simulation of biochemical processes using the π-calculus process algebra. In: Proceedings of Pacific Symposium on Biocomputing (PSB'01). Volume 6. (2001) 459–470

[12] Cardelli, L., Gordon, A.D.: Mobile Ambients. In: Proceedings of Conference on Foundations of Software Science and Computation Structures (FoSSaCS'98). Volume 1378 of LNCS. (1998) 140–155

[13] Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walter, P.: Molecular biology of the cell. Garland Science (2002)