

Control Flow Analysis for Brane Calculi

Chiara Bodei¹, Andrea Bracciali¹ and Davide Chiarugi²

¹ *Dipartimento di Informatica, Università di Pisa,
Via Pontecorvo, 3 - 56127 Pisa - Italia
{chiara,braccia}@di.unipi.it*

² *Dipartimento di Scienze Matematiche e Informatiche, Università di Siena,
Pian dei Mantellini, 4 - 53100 Siena - Italia
chiarugi3@unisi.it*

Abstract

We introduce a Control Flow Analysis for Brane Calculi. This verification technique allows properties regarding the behaviour of biological systems to be checked. This is an approximate technique that focusses on the static specification of a system, rather than on its dynamics, striving for effectiveness. Examples illustrate the approach.

Keywords: Brane calculi, control flow analysis, systems biology

1 Introduction

Systems biology gives computer scientists the opportunity of providing models and formalisms for describing and analysing complex biological systems. In this regard, well established and founded theories and techniques from formal methods offer a fertile ground.

The underlying idea is that a biological system can be abstractly modelled as a concurrent system [18]. Several approaches – developed to predict the dynamic behaviour of the modelled systems – have introduced the idea of performing *in silico* experiments to establish which *in vitro* experiments are more promising.

The behaviour of a system is usually given in terms of its transition system, whose size can be huge, making its exploration computationally hard. Resorting to static techniques offers the possibility of drastically reducing the computational costs, particular high when modelling complex biological systems. The specification of the system is statically analysed in order to obtain information on the dynamic behaviour and to check the related dynamic properties, without actually exploring the whole state space of the associated transition system. The price to be paid is a loss in precision, because usually these techniques can only approximate the actual

behaviour. Static analysis can be exploited for a sort of preliminary screening of the *in silico* experiments by efficiently testing different hypothesis and rapidly identifying which *in vitro* experiments worth to be performed. We first introduce a Control Flow Analysis for the version of Brane Calculi [6,7], called MBD and based on the operations for membrane fusion and splitting. Afterwards, we extend the framework, in order to include also the PEP actions that represent the operations modeling endocytosis and exocytosis. The analysis offers predictions on the contents of membranes: which membranes can be contained in the analysed one and which actions can affect it. This information offers a basis for studying dynamic properties, by suitably handling the over-approximation the static analysis introduces. Having a safe over-approximation of the *exact* behaviour of a system means that all the valid behaviour are captured. More precisely, all those events that the prediction does not include will *never* happen, while when included, the events *can* happen, i.e. they are only possible.

We apply our analysis to two simple examples: (1) a model of the infection due to membrane enveloped viruses, recalling in particular the Semliki Forest Virus [6] and (2) a model of communication via mobile vesicles [19].

The paper follows the tradition initiated by [12,13] and continued with [15,16,3] of applying static techniques and, in particular, Control Flow Analysis to process calculi used for modelling biological phenomena. Our choice of the Brane calculi depends on the fact they have resulted to be particularly useful for modelling and reasoning about a large class of biological systems, such as the one of the eukaryotic cells that, differently from the prokaryotes, possess a set of internal membranes. Other applications of static analysis techniques have been proposed, like the Abstract Interpretation for BioAmbient in [10]. In a different context, the behaviour of processes is safely approximated and the properties of a fragment of Computation Tree Logic preserved. This makes it possible to address temporal properties and therefore some kinds of causality.

Among the first formalisms used to investigate biological membranes there are the P Systems [14], introduced by Păun, which formalize distributed parallel computations biologically-inspired: a biological system is seen as a complex hierarchical structure of nested membranes inspired by the structure of living cells.

Finally, besides Brane, there are other calculi of interest for our approach, that have been specifically defined for modelling biological structures such as compartments and membranes, e.g. κ -calculus [9], Beta Binders [17] and the Calculus of Looping Sequences [2].

The rest of the paper is organised as follows. In Section 2 we present the MBD version of Brane Calculi. We introduce the Control Flow Analysis in Section 3. In Section 4 the analysis is extended in order to also treat the PEP actions. In Section 5 the new analysis is applied to a model of infective cycle of the Semliki Forest Virus and to a model of communication via a mobile vesicle. Section 6 presents some concluding remarks. Proofs of theorems and lemmata presented throughout the paper are collected in Appendix A.

2 The Calculus

Syntax

The basic structure of Brane Calculi [6] consists of two commutative monoids plus a replication operator, where \circ is used for system composition, with unit \diamond , and $|$ is used for membrane composition, with unit 0 . Replication $!$ is used to represent the composition of an unbounded number of instances. We focus here on the version of the calculus without communication primitives and molecular complexes, first introducing the MBD actions and, only afterwards, the PEP ones. A system consists of nested membranes and membranes consist of collections of actions. The MBD actions are inspired by membrane fusion and splitting. The first is modelled by the *mating* operation, while the second can be rendered both by *budding*, that consists in splitting off exactly one internal membrane, and *dripping*, that consists in splitting off one empty membrane.

$$\begin{aligned}
 P, Q &::= \diamond \mid P \circ Q \mid !P \mid \sigma \langle P \rangle^\mu && \text{systems } \mathcal{S} \\
 \sigma, \tau &::= 0 \mid \sigma | \tau \mid !\sigma \mid a.\sigma && \text{membrane processes } \mathcal{B} \\
 a, b &::= \text{mate}_n \mid \text{mate}_n^\perp \mid \text{bud}_n \mid \text{bud}_n^\perp(\rho) \mid \text{drip}(\rho) && \text{MBD actions } \mathcal{Act}_{MBD}
 \end{aligned}$$

Here, n is taken by a countable set Λ of names. Membranes are nameless entities, but to facilitate our analysis, we annotate them as in $\sigma \langle P \rangle^\mu$, in order to distinguish different syntactic occurrences of membranes. We refer to $\mu \in \mathbf{M}$ as the identity of the membrane $\sigma \langle P \rangle^\mu$, where \mathbf{M} is the finite set of membrane identities. Still to simplify our analysis, we assume that each considered system is contained in an ideal outermost membrane, identified by a distinguished element $* \in \mathbf{M}$.

The term $\sigma \langle P \rangle^\mu$ denotes a system whose behaviour depends on the actions in σ and that includes the system P : σ represents a membrane and P its contents. The term $a.\sigma$ denotes a process that executes a and then behaves as σ . Actions for mating (mate_n) and budding (bud_n) have the corresponding co-actions (mate_n^\perp , bud_n^\perp resp.) to synchronise with. The actions $\text{bud}_n^\perp(\rho)$ and $\text{drip}(\rho)$ are equipped with a process ρ to associate with the membrane created by the performed action.

Semantics

The standard *structural congruence* \equiv on systems and membranes is the least congruence satisfying the following clauses:

- $(\mathcal{S}/\equiv, \circ, \diamond)$ is a commutative monoid;
- $! \diamond \equiv \diamond$, $!(P \circ Q) \equiv !P \circ !Q$, $!!P \equiv !P$, $!P \equiv P \circ !P$;
- $(\mathcal{B}/\equiv, |, 0)$ is a commutative monoid;
- $!0 \equiv 0$, $!(\sigma | \tau) \equiv !\sigma | !\tau$, $!!\sigma \equiv !\sigma$, $!\sigma \equiv \sigma | !\sigma$;
- $0 \langle \diamond \rangle^{\mu_\diamond} \equiv \diamond$;
- $P \equiv Q \Rightarrow P \circ R \equiv Q \circ R$, $P \equiv Q \Rightarrow !P \equiv !Q$, $P \equiv Q \wedge \sigma \equiv \tau \Rightarrow \sigma \langle P \rangle^\mu \equiv \tau \langle Q \rangle^\mu$;
- $\sigma \equiv \tau \Rightarrow \sigma | \rho \equiv \tau | \rho$, $\sigma \equiv \tau \Rightarrow !\sigma \equiv !\tau$, $\sigma \equiv \tau \Rightarrow a.\sigma \equiv a.\tau$.

(Par)	$(Brane)$	$(Struct)$
$\frac{P \rightarrow Q}{P \circ R \rightarrow Q \circ R}$	$\frac{P \rightarrow Q}{\sigma\langle P \rangle^\mu \rightarrow \sigma\langle Q \rangle^\mu}$	$\frac{P \equiv P' \wedge P' \rightarrow Q' \wedge Q' \equiv Q}{P \rightarrow Q}$
$(Mate) \text{ mate}_n.\sigma \sigma_0\langle P \rangle^{\mu_P} \circ \text{mate}_n^\perp.\tau \tau_0\langle Q \rangle^{\mu_Q} \rightarrow \sigma \sigma_0 \tau \tau_0\langle P \circ Q \rangle^{\mu_{PQ}}$ where $\mu_{PQ} = \mathbf{MI}_{\mathbf{mate}}(\text{mate}_n, \mu_P, \mu_Q, \mu)$ and μ identifies the closest membrane surrounding μ_P and μ_Q		
$(Bud) \text{ bud}_n^\perp(\rho).\tau \tau_0\langle \text{bud}_n.\sigma \sigma_0\langle P \rangle^{\mu_P} \circ Q \rangle^{\mu_Q} \rightarrow \rho\langle \sigma \sigma_0\langle P \rangle^{\mu_P} \rangle^{\mu_R} \circ \tau \tau_0\langle Q \rangle^{\mu_Q}$ where $\mu_R = \mathbf{MI}_{\mathbf{bud}}(\text{bud}_n, \mu_P, \mu_Q, \mu)$ and μ identifies the closest membrane surrounding μ_Q		
$(Drip) \text{ drip}(\rho).\sigma \sigma_0\langle P \rangle^{\mu_P} \rightarrow \rho\langle \rangle^{\mu_R} \circ \sigma \sigma_0\langle P \rangle^{\mu_P}$ where $\mu_R = \mathbf{MI}_{\mathbf{drip}}(\text{drip}(\rho), \mu_P, \mu)$ and μ identifies the closest membrane surrounding μ_P		

Table 1
Reduction Semantics for Brane Calculus.

The interleaving semantics is based on the above notion of structural congruence and is composed by the basic reaction rules, valid for all brane calculi (upper part of Table 1) and by the reaction axioms for the MBD version (lower part of Table 1). Observe that, given a system, the set of its membrane identities is finite. Indeed, the structural congruence rule $!P \equiv !P|P$ applied to $!\sigma\langle P \rangle^\mu$ gives $!\sigma\langle P \rangle^\mu \equiv !\sigma\langle P \rangle^\mu|\sigma\langle P \rangle^\mu$, i.e. no new identity is ever introduced in recursive calls. However, we need a distinguished membrane identity each time a certain action is performed, in order to identify the membrane obtained in the target. More precisely, we need a membrane identity for the membrane resulting from the firing of an action *mate*, possible whenever a *mate_n* (*mate_n[⊥]*, resp.) resides on a membrane identified by μ_P (μ_Q , resp.), with μ_P and μ_Q included in a membrane identified by μ . Also we need a single membrane identity for each other possible pair of action/coaction occurring in siblings membranes. The number of such membrane identities is finite, due to the above structural congruence rule. Similarly, we need a distinguished membrane identity for each action *bud* or *drip*, i.e. either when a *bud_n* (*bud_n[⊥]*, resp.) resides on a membrane identified by μ_P (μ_Q , resp.), with the membrane μ_P included in the membrane μ_Q , in turn included in a membrane μ , or when a *drip* resides on a membrane identified by μ_P , included in the membrane μ . To obtain the needed membrane identities, one can exploit functions that are injective and surjective on their images. These auxiliary functions return fresh membrane identities, depending on the actions and on their syntactic contexts. More precisely:

$$\mathbf{MI}_{\text{mate}} : \mathcal{Act}_{MBD} \times \mathbf{M} \times \mathbf{M} \times \mathbf{M} \rightarrow \mathbf{M}$$

$$\mathbf{MI}_{\text{bud}} : \mathcal{Act}_{MBD} \times \mathbf{M} \times \mathbf{M} \times \mathbf{M} \rightarrow \mathbf{M}$$

$$\mathbf{MI}_{\text{drip}} : \mathcal{Act}_{MBD} \times \mathbf{M} \times \mathbf{M} \rightarrow \mathbf{M}$$

We dispense from the actual definition of these functions. We just recall that, given a system, the number of needed membrane identities is finite, as finite are the possible combinations of actions and contexts. Therefore, we choose \mathbf{M} such that, given an action and the identities of the membranes on which the action (and the corresponding co-action, if any) reside, \mathbf{M} includes the membrane identity needed to identify the membrane obtained by firing that action.

3 Control Flow Analysis

We develop a Control Flow Analysis for analysing Brane calculus systems, borrowing some ideas from [13]. The aim of the analysis is over-approximating all the possible behaviour of a Brane system. In particular, our analysis keeps track of the possible contents of membranes, thus taking care of the possible modifications of the containment hierarchy, due to the dynamics. An approximation of the contents of a membrane or *estimate* \mathcal{I} is defined as

$$\mathcal{I} : \mathbf{M} \rightarrow \wp(\mathbf{M} \cup \mathcal{Act}),$$

where $\wp(S)$ stands for the power-set of the set S and \mathcal{Act} is the set of Brane actions. Here, $\mu' \in \mathcal{I}(\mu)$ means that the membrane identified by μ may contain the one identified by μ' ; $a \in \mathcal{I}(\mu)$ means that the action a may reside on and affect the membrane identified by μ . To validate the correctness of a proposed estimate \mathcal{I} , we state a set of clauses operating upon judgements like $\mathcal{I} \models^\mu P$. This judgement expresses that when P is enclosed within a membrane identified by $\mu \in \mathbf{M}$, then \mathcal{I} correctly captures the behaviour of P , i.e. the estimate is valid also for all the states Q passed through a computation of P .

Following [13], the analysis is specified in two phases. First, we check that \mathcal{I} describes the initial process. This is done in Table 2, where the clauses amount to a structural traversal of process syntax. These clauses rely on the following auxiliary function that collects all the actions in a membrane process σ .

Definition 3.1 $A : \mathcal{B} \rightarrow \mathcal{Act}$

- $A(0) = \emptyset$;
- $A(\sigma|\tau) = A(\sigma) \cup A(\tau)$;
- $A(!\sigma) = A(\sigma)$;
- $A(a.\sigma) = \{a\} \cup A(\sigma)$;

Note that the actions collected by A , e.g., in $\sigma = \sigma_0.\sigma_1$ are equal to the ones in $\sigma' = \sigma_0|\sigma_1$, witnessing the fact that the analysis is insensitive to the context and introduces imprecision and approximation.

The clause for membrane system $\sigma\langle P\rangle^{\mu'}$ checks that whenever a membrane μ' is introduced inside a membrane μ , the relative hierarchy position must be reflected in \mathcal{I} , i.e. $\mu' \in \mathcal{I}(\mu)$. Furthermore, the actions in σ that affect the membrane μ' and that are collected in $A(\sigma)$, are recorded in $\mathcal{I}(\mu')$. Finally, when inspecting the content P , the fact that the enclosing membrane is μ' is recorded, as reflected by the judgement $\mathcal{I} \models^{\mu'} P$. The rule for \diamond and 0 do not restrict the analysis result, while the rules for parallel composition \circ , and replication $!$ ensure that the analysis also holds for the immediate sub-systems, by ensuring their traversal. In particular, the analysis of $!P$ is equal to the one of P . This is another source of imprecision.

Secondly, we check that \mathcal{I} also takes into account the dynamics of the process under consideration; in particular, the dynamics of the containment hierarchy of membranes. This is expressed by the closure conditions in the lower part of Table 2 that mimic the semantics, by modelling, without exceeding the precision boundaries of the analysis, the semantic preconditions and the consequences of the possible actions. More precisely, the precondition checks, in terms of \mathcal{I} , for the possible presence of the redexes necessary for an action to be performed. The conclusion imposes the additional requirements on \mathcal{I} , necessary to give a valid prediction of the analysed action. Consider e.g., the clause for $(Mate)$ (the other clauses are similar). We have to make sure that the precondition requirements are satisfied, i.e. that:

- there exists an occurrence of mate action: $mate_n \in \mathcal{I}(\mu_P)$;
- there exists an occurrence of the corresponding co-mate action: $mate_n^\perp \in \mathcal{I}(\mu_Q)$;
- the corresponding membranes must be siblings: $\mu_P, \mu_Q \in \mathcal{I}(\mu)$

If the precondition requirements are satisfied, then the conclusions of the clause express the consequences of performing the transition $mate$. In this case, we have that \mathcal{I} must reflect that

- there may exist a membrane μ_{PQ} inside μ , with the same father of the membranes μ_P and μ_Q : $\mu_{PQ} \in \mathcal{I}(\mu)$; and that
- the contents of μ_P and of μ_Q may also be inside μ_{PQ} , and therefore μ_{PQ} contains every membrane that is inside μ_P or μ_Q , while each action affecting μ_P or μ_Q , affects also μ_{PQ} : $\mathcal{I}(\mu_P) \cup \mathcal{I}(\mu_Q) \subseteq \mathcal{I}(\mu_{PQ})$.

This corresponds to the application of the semantic rule $(Mate)$ that would result in the fusion of the two membranes.

Example 3.2 We illustrate our analysis on a simple example, taken from [5], of which we report one of the possible steps of computation: the $(Mate)$ one.

$$(mate_n | bud_m^\perp(\rho_1)) \langle bud_m \rangle^{\mu_{P_0}} \circ bud_o \langle \rangle^{\mu_{P_1}} \rangle^{\mu_P} \circ (mate_n^\perp | bud_o^\perp(\rho_2)) \langle \rangle^{\mu_Q} \xrightarrow{Mate} (bud_m^\perp(\rho_1) | bud_o^\perp(\rho_2)) \langle bud_m \rangle^{\mu_{P_0}} \circ bud_o \langle \rangle^{\mu_{P_1}} \circ \diamond \rangle^{\mu_{PQ}}$$

The main entries of the analysis are reported in Table 3, where $*$ identifies the ideal outermost membrane in which the system top-level membranes are. It is easy to check that \mathcal{I} is a valid estimate, by following the two stages explained above. The transition $mate_n$ is predicted as possible in \mathcal{I} , as its precondition requirements are

$\mathcal{I} \models^\mu \diamond$	iff <i>true</i>
$\mathcal{I} \models^\mu P \circ Q$	iff $\mathcal{I} \models^\mu P \wedge \mathcal{I} \models^\mu Q$
$\mathcal{I} \models^\mu !P$	iff $\mathcal{I} \models^\mu P$
$\mathcal{I} \models^\mu \sigma\langle P \rangle^{\mu'}$	iff $\mu' \in \mathcal{I}(\mu) \wedge A(\sigma) \subseteq \mathcal{I}(\mu') \wedge \mathcal{I} \models^{\mu'} P$
<p>(<i>Mate</i>) $mate_n \in \mathcal{I}(\mu_P) \wedge mate_n^\perp \in \mathcal{I}(\mu_Q) \wedge \mu_P, \mu_Q \in \mathcal{I}(\mu)$ $\Rightarrow \mu_{PQ} \in \mathcal{I}(\mu) \wedge (\mathcal{I}(\mu_P) \cup \mathcal{I}(\mu_Q)) \subseteq \mathcal{I}(\mu_{PQ})$ where $\mu_{PQ} = \mathbf{MI}_{\text{mate}}(mate_n, \mu_P, \mu_Q, \mu)$</p> <p>(<i>Bud</i>) $bud_n \in \mathcal{I}(\mu_P) \wedge bud_n^\perp(\rho) \in \mathcal{I}(\mu_Q) \wedge \mu_P \in \mathcal{I}(\mu_Q) \wedge \mu_Q \in \mathcal{I}(\mu)$ $\Rightarrow A(\rho) \in \mathcal{I}(\mu_R) \wedge \mu_R \in \mathcal{I}(\mu) \wedge \mu_P \in \mathcal{I}(\mu_R)$ where $\mu_R = \mathbf{MI}_{\text{bud}}(bud_n, \mu_P, \mu_Q, \mu)$</p> <p>(<i>Drip</i>) $drip \in \mathcal{I}(\mu_P) \wedge \mu_P \in \mathcal{I}(\mu)$ $\Rightarrow A(\rho) \in \mathcal{I}(\mu_R) \wedge \mu_R \in \mathcal{I}(\mu)$ where $\mu_R = \mathbf{MI}_{\text{drip}}(drip(\rho), \mu_P, \mu)$</p>	

Table 2
Analysis for Brane Processes

$\mu_{P_0}, \mu_{P_1} \in \mathcal{I}(\mu_P), \mu_P, \mu_Q \in \mathcal{I}(\ast)$	$\mu_{P_0}, \mu_{P_1} \in \mathcal{I}(\mu_{PQ}), \mu_{PQ} \in \mathcal{I}(\ast)$
$mate_n \in \mathcal{I}(\mu_P)$	$mate_n^\perp \in \mathcal{I}(\mu_Q)$
$bud_m \in \mathcal{I}(\mu_{P_0}), bud_m^\perp(\rho_1) \in \mathcal{I}(\mu_P)$	$bud_m^\perp(\rho_1) \in \mathcal{I}(\mu_{PQ})$
$bud_o \in \mathcal{I}(\mu_{P_1}), bud_o^\perp(\rho_2) \in \mathcal{I}(\mu_Q)$	$bud_o^\perp(\rho_2) \in \mathcal{I}(\mu_{PQ})$
$\mu_{P_0} \in \mathcal{I}(\mu_{R1}), \mu_{P_0} \in \mathcal{I}(\mu'_{R1})$	$\mu_{P_1} \in \mathcal{I}(\mu_{R2})$
where $\mu_{PQ} = \mathbf{MI}_{\text{mate}}(mate_n, \mu_P, \mu_Q, \ast)$ $\mu_{R2} = \mathbf{MI}_{\text{bud}_o}(bud_o, \mu_{P_1}, \mu_{PQ}, \ast)$	
$\mu_{R1} = \mathbf{MI}_{\text{bud}_m}(bud_m, \mu_{P_0}, \mu_P, \ast)$	$\mu'_{R1} = \mathbf{MI}_{\text{bud}_m}(bud_m, \mu_{P_0}, \mu_{PQ}, \ast)$

Table 3
Some Entries of the Example Analysis

satisfied: $mate_n \in \mathcal{I}(\mu_P)$ and $mate_n^\perp \in \mathcal{I}(\mu_Q)$, and the two membranes are siblings. We can observe that the corresponding conclusion requirements are satisfied as well, because both $\mathcal{I}(\mu_P)$ and $\mathcal{I}(\mu_Q)$ are included in $\mathcal{I}(\mu_{PQ})$, where μ_{PQ} identifies the new membrane created by the fusion. Also the transition on bud_m is initially possible and this result is actually predicted by the analysis. Instead, we can observe that the transition on bud_o cannot be performed in the initial system, because the actions do not reside on two membranes where one is the father of the other. This

is reflected by the analysis; indeed we have that the precondition requirements are not satisfied: $bud_o \in \mathcal{I}(\mu_{P_1})$, $bud_o^\perp \in \mathcal{I}(\mu_Q)$, but $\mu_{P_1} \notin \mathcal{I}(\mu_Q)$. Nonetheless, the transition on bud_o can be performed, after the mate transition (reported above), as correctly predicted by the analysis, since $bud_o \in \mathcal{I}(\mu_{P_1})$, with $\mu_{P_1} \in \mathcal{I}(\mu_{PQ})$ and $bud_o^\perp \in \mathcal{I}(\mu_{PQ})$. Note that the action bud_m can be performed in the context in which $bud_m \in \mathcal{I}(\mu_{P_0})$, $bud_m^\perp \in \mathcal{I}(\mu_P)$, with $\mu_{P_0} \in \mathcal{I}(\mu_P)$, but also in the one where $bud_m \in \mathcal{I}(\mu_{P_0})$, $bud_m^\perp \in \mathcal{I}(\mu_{PQ})$, with $\mu_{P_0} \in \mathcal{I}(\mu_{PQ})$. The obtained membranes are therefore differently identified: μ_{R_1} in the first case and μ'_{R_1} in the second.

Semantic Correctness

Our analysis is semantically correct with respect to the given semantics, i.e. a valid estimate enjoys the following subject reduction property with respect to the semantics.

Theorem 3.3 (*Subject Reduction*)

If $P \rightarrow Q$ and $\mathcal{I} \models^\mu P$ then also $\mathcal{I} \models^\mu Q$.

This result depends on the fact that analysis is invariant under the structural congruence, as stated below.

Lemma 3.4 (*Invariance of Structural Congruence*) If $P \equiv Q$ and $\mathcal{I} \models^\mu P$ then also $\mathcal{I} \models^\mu Q$.

Existence of the Analysis

We have previously seen a procedure for verifying whether or not a proposed estimate \mathcal{I} is valid. We now show that for any given P there always is a least choice of \mathcal{I} that is acceptable according to the rules in Table 2, i.e. such that $\mathcal{I} \models^\mu P$.

Definition 3.5 The set of proposed solutions can be partially ordered by setting $\mathcal{I} \sqsubseteq \mathcal{I}'$ iff $\forall \mu : \mathcal{I}(\mu) \sqsubseteq \mathcal{I}'(\mu)$.

This suffices for making the set of proposed solutions into a complete lattice; using standard notation we write $\mathcal{I} \sqcup \mathcal{I}'$ for the binary least upper bound (defined point-wise), $\sqcap \mathbf{E}$ for the greatest lower bound of a set \mathbf{E} of proposed estimates (also defined pointwise), and \perp for the least element.

Definition 3.6 A set \mathbf{E} of proposed estimates is a *Moore family* if and only if it contains $\sqcap \mathbf{F}$ for all $\mathbf{F} \subseteq \mathbf{E}$ (in particular $\mathbf{F} = \emptyset$ and $\mathbf{F} = \mathbf{E}$).

When \mathbf{E} is a Moore family it contains a greatest element ($\sqcap \emptyset$) as well as a least element ($\sqcap \mathbf{E}$). The following theorem then guarantees that there always is an estimate satisfying the specification in Table 2.

Theorem 3.7 (*Moore Family*)

For any system P , the set $\{\mathcal{I} \mid \mathcal{I} \models^\mu P\}$ is a Moore family.

Currently, our analysis is not implemented, but it can, along the lines of the Control Flow Analysis for BioAmbients [13]. This means that it is possible, given a process, to compute its least estimate.

4 Extension to the PEP Actions

The presented analysis can be extended (see below) in order to deal with the Phago/Exo/Pino (PEP) version of Brane Calculus. These further operations are used to describe endocytosis and exocytosis processes. The first indicates the process of incorporating external material into a cell, by engulfing it with the cell membrane, while the second one indicates the reverse process. Endocytosis is rendered by two more basic operations: *phagocytosis* (denoted by *phago*), that consists in engulfing just one external membrane, and *pinocytosis* (denoted by *pino*), consists in engulfing zero external membranes; exocytosis is instead denoted by *exo*. The detailed extensions to the syntax follow, while the semantics ones are in Table 4.

$$a ::= phago_n \mid phago_n^\perp(\rho) \mid exo_n \mid exo_n^\perp \mid pino(\rho) \text{ PEP actions } \mathcal{Act}_{PEP}$$

$$\mathbf{MI}_{phago} : \mathcal{Act}_{PEP} \times \mathbf{M} \times \mathbf{M} \times \mathbf{M} \rightarrow \mathbf{M}$$

$$\mathbf{MI}_{pino} : \mathcal{Act}_{PEP} \times \mathbf{M} \times \mathbf{M} \rightarrow \mathbf{M}$$

(Phago)	$phago_n.\sigma \sigma_0\langle P\rangle^{\mu_P} \circ phago_n^\perp(\rho).\tau \tau_0\langle Q\rangle^{\mu_Q} \rightarrow \tau \tau_0\langle \rho \sigma \sigma_0.\langle P\rangle^{\mu_P}\rangle^{\mu_R} \circ Q\rangle^{\mu_Q}$
	where $\mu_R = \mathbf{MI}_{phago}(phago_n, \mu_P, \mu_Q, \mu)$
	and μ identifies the closest membrane surrounding μ_P and μ_Q
(Exo)	$exo_n^\perp.\tau \tau_0\langle exo_n.\sigma \sigma_0\langle P\rangle^{\mu_P} \circ Q\rangle^{\mu_Q} \rightarrow P \circ \sigma \sigma_0 \tau \tau_0\langle Q\rangle^{\mu_Q}$
(Pino)	$pino(\rho).\sigma \sigma_0\langle P\rangle^{\mu_P} \rightarrow \sigma \sigma_0\langle \rho\rangle^{\mu_R} \circ P\rangle^{\mu_P}$
	where $\mu_R = \mathbf{MI}_{pino}(pino(\rho), \mu_P, \mu)$
	and μ identifies the closest membrane surrounding μ_P

Table 4
Reduction Rules for PEP Actions.

The Control Flow Analysis can be extended by adding the closure conditions in Table 5. Our extended analysis is still semantically correct with respect to the given semantics and the estimates still form a Moore family, therefore guaranteeing the existence of a least estimate satisfying the clauses in Tables 2 and 5. The extended results are handled and proved in the Appendix.

5 The Analysis at Work

In Brane calculi the dynamics of biological membranes is abstracted by means of interactions that lead to modifications of the hierarchy of compartments. Control Flow Analysis gives information on the possible variations of the containment hierarchy. Both examples presented below show why this information is important, as it offers the basis for qualified predictions about the possible dynamic behaviour.

<i>(Phago)</i>	$phago_n \in \mathcal{I}(\mu_P) \wedge phago_n^\perp(\rho) \in \mathcal{I}(\mu_Q) \wedge \mu_P, \mu_Q \in \mathcal{I}(\mu)$ $\Rightarrow A(\rho) \in \mathcal{I}(\mu_R) \wedge \mu_R \in \mathcal{I}(\mu_Q) \wedge \mu_P \in \mathcal{I}(\mu_R)$ where $\mu_R = \mathbf{MI}_{phago}(phago_n, \mu_P, \mu_Q, \mu)$
<i>(Exo)</i>	$exo_n \in \mathcal{I}(\mu_P) \wedge exo_n^\perp \in \mathcal{I}(\mu_Q) \wedge \mu_P \in \mathcal{I}(\mu_Q) \wedge \mu_Q \in \mathcal{I}(\mu)$ $\Rightarrow A(\sigma), A(\sigma_0) \in \mathcal{I}(\mu_Q) \wedge \mathcal{I}(\mu_P) \subseteq \mathcal{I}(\mu)$
<i>(Pino)</i>	$pino(\rho) \in \mathcal{I}(\mu_P) \wedge \mu_P \in \mathcal{I}(\mu)$ $\Rightarrow A(\rho) \in \mathcal{I}(\mu_R) \wedge \mu_R \in \mathcal{I}(\mu_P)$ where $\mu_R = \mathbf{MI}_{pino}(pino(\rho), \mu_P, \mu)$

Table 5
Closure Rules for PEP Actions

5.1 Viral Infection

We first refer to the abstract specification introduced in [6] of the viral infection cycle of the Semliki Forest Virus, shown in Figure 1. This is one of the so-called “enveloped viruses”. These viruses possess a lipidic bilayer envelope that surrounds the nucleocapsid, the part of the virus composed by the genome and its protective coat formed by identical protein subunits (capsomers). The viral outer envelope is a modified form of one of the membranes of the host cell, either the cell membrane, or an internal membrane, e.g. the nuclear membrane or endoplasmic reticulum. Noticeably, most enveloped viruses depend on their envelope for their infectivity.

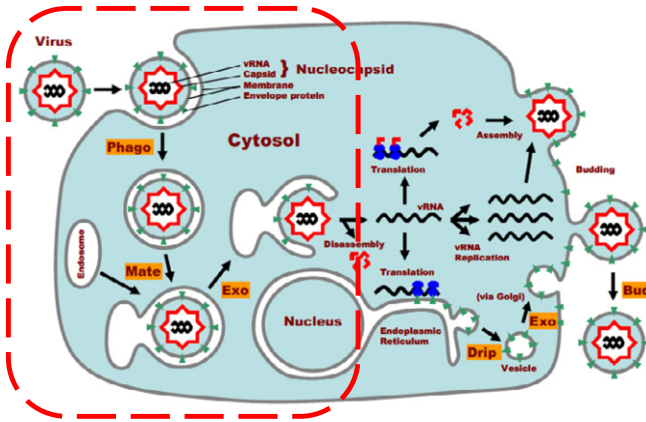


Fig. 1. Viral Infection (highlighted part) and Reproduction. [Adapted from [6] and [1]]

In our example, we consider the first stage of the infective cycle of the Semliki Forest Virus (highlighted part in Figure 1), whose steps can be coded as in Table 6.

$virus \stackrel{def}{=} phago.exo\langle nucap \rangle^{\mu_{virus}}$	$membrane \stackrel{def}{=} !phago^{\perp}(mate)!exo^{\perp}$
$nucap \stackrel{def}{=} !bud X\langle vRNA \rangle^{\mu_{nucap}}$	$cytosol \stackrel{def}{=} endosome \circ Z$
$cell \stackrel{def}{=} membrane\langle cytosol \rangle^{\mu_{memb}}$	$endosome \stackrel{def}{=} !mate^{\perp}!exo^{\perp}\rangle^{\mu_{endo}}$

Table 6
Viral Infection System

$$\begin{aligned}
& virus \circ cell \\
& \equiv (phago.exo)\langle nucap \rangle^{\mu_{virus}} \circ (!phago^{\perp}(mate)!exo^{\perp})\langle cytosol \rangle^{\mu_{memb}} \xrightarrow{phago} \\
& (!phago^{\perp}(mate)!exo^{\perp})\langle mate(exo\langle nucap \rangle^{\mu_{virus}})^{\mu_{ph}} \circ (!mate^{\perp}!exo^{\perp})\rangle^{\mu_{endo} \circ Z} \rangle^{\mu_{memb}} \\
& \xrightarrow{mate} (!phago^{\perp}(mate)!exo^{\perp})\langle (!mate^{\perp}!exo^{\perp})(exo\langle nucap \rangle^{\mu_{virus}})^{\mu_{ph-endo}} \circ Z \rangle^{\mu_{memb}} \xrightarrow{exo} \\
& (!phago^{\perp}(mate)!exo^{\perp})\langle (!mate^{\perp}!exo^{\perp})\rangle^{\mu_{ph-endo} \circ nucap \circ Z} \rangle^{\mu_{memb}} \equiv \\
& membrane\langle nucap \circ cytosol \rangle^{\mu_{memb}}
\end{aligned}$$

Table 7
Viral Infection Evolution

$$\begin{aligned}
& \mu_{nucap} \in \mathcal{I}(\mu_{virus}), \mu_{endo} \in \mathcal{I}(\mu_{memb}), \mu_{virus}, \mu_{memb} \in \mathcal{I}(\ast) \\
& phago, exo \in \mathcal{I}(\mu_{virus}), phago^{\perp}(mate), exo^{\perp} \in \mathcal{I}(\mu_{memb}) \\
& mate \in \mathcal{I}(\mu_{ph}), \mu_{ph} \in \mathcal{I}(\mu_{memb}), \mu_{virus} \in \mathcal{I}(\mu_{ph}), \mathcal{I}(\mu_{ph}) \cup \mathcal{I}(\mu_{endo}) \subseteq \mathcal{I}(\mu_{ph-endo}) \\
& mate^{\perp}, exo^{\perp} \in \mathcal{I}(\mu_{endo}) \\
& \mu_{ma} \in \mathcal{I}(\mu_{memb}), \mu_{virus} \in \mathcal{I}(\mu_{ph-endo}), mate^{\perp}, exo^{\perp} \in \mathcal{I}(\mu_{ph-endo}) \\
& \mathcal{I}(\mu_{virus}) \subseteq \mathcal{I}(\mu_{memb}) \text{ and in particular } \mu_{nucap} \in \mathcal{I}(\mu_{memb})
\end{aligned}$$

Table 8
Viral Infection Analysis Results

We focus only on the first stage, because it lends itself to show how the Control Flow Analysis predictions could reflect the dynamic modifications of the membranes hierarchy. In addition, striving for simplicity, it does not require further extensions to the bunch of Brane Calculi primitives used here.

Usually, the Semliki Forest Virus is brought into the cell by phagocytosis, thus wrapped by an additional membrane layer. An endosome compartment is merged with the wrapped-up virus. At this point, the virus uses one of its special (viral encoded) membrane protein to trigger the exocytosis process that leads the naked nucleocapsid into the cytosol, ready to continue the infective cycle. By summarising, if the *cell* gets close to the *virus*, then it evolves into an infected cell:

$$virus \circ cell \rightarrow^* membrane\langle nucap \circ cytosol \rangle^{\mu_{memb}}$$

The complete evolution of the viral infection is reported in Table 7, while the main analysis entries are in Table 8. The analysis results allow us to predict the effects of the infection. Indeed, the inclusion $\mu_{nucap} \in \mathcal{I}(\mu_{memb})$ reflects the fact that, at the end of the shown computation, *nucap* is inside *membrane*, together with *cytosol*.

5.2 Communication via a Mobile Vesicle

In eucaryotic cells a large variety of proteins is targetted to its final destination via mobile transport vesicles, i.e. small membrane-enclosed sacs separated from the cytosol by a lipidic bilayer. Proteins can be contained in the vesicles (e.g. secretory proteins) or embedded in their membrane, e.g. transmembrane proteins.

$G \stackrel{def}{=} \omega_G \langle Source \circ Target \rangle^{\mu_G}$	$Target \stackrel{def}{=} !phago_{n2}^\perp(exo^\perp) \omega_{Target} \langle \diamond \rangle^{\mu_T}$
$vesicle \stackrel{def}{=} phago_{n2}.exo$	$Source \stackrel{def}{=} !bud_{n1}^\perp(vesicle) \omega_{Source} \langle bud_{n1} \omega_X \langle \diamond \rangle^{\mu_X} \rangle^{\mu_S}$

Table 9
Communication via a Mobile Vesicle: Encoding

$$\begin{aligned}
 G &= \omega_G \langle \underbrace{!bud_{n1}^\perp(phago_{n2}.exo) | \omega_{Source} \langle bud_{n1} | \omega_X \langle \diamond \rangle^{\mu_X} \rangle^{\mu_S}}_{Source} \circ Target \rangle^{\mu_G} \xrightarrow{bud_{n1}} \\
 &\omega_G \langle \underbrace{!bud_{n1}^\perp(vesicle) | \omega_{Source} \langle \diamond \rangle^{\mu_S}}_{Source^*} \circ vesicle \langle \omega_X \langle \diamond \rangle^{\mu_X} \rangle^{\mu_V} \circ Target \rangle^{\mu_G} \equiv \\
 &\omega_G \langle Source^* \circ phago_{n2}.exo \langle \omega_X \langle \diamond \rangle^{\mu_X} \rangle^{\mu_V} \circ !phago_{n2}^\perp(exo^\perp) | \omega_{Target} \langle \diamond \rangle^{\mu_T} \rangle^{\mu_G} \xrightarrow{phago_{n2}} \\
 &\omega_G \langle Source^* \circ !phago_{n2}^\perp(exo^\perp) | \omega_{Target} \langle exo^\perp \langle exo \langle \omega_X \langle \diamond \rangle^{\mu_X} \rangle^{\mu_V} \rangle^{\mu_E} \rangle^{\mu_T} \rangle^{\mu_G} \xrightarrow{exo} \\
 &\omega_G \langle Source^* \circ !phago_{n2}^\perp(exo^\perp) | \omega_{Target} \langle \omega_X \langle \diamond \rangle^{\mu_X} \circ 0 \langle \diamond \rangle^{\mu_E} \rangle^{\mu_T} \rangle^{\mu_G} \equiv \\
 &\omega_G \langle !bud_{n1}^\perp(vesicle) | \omega_{Source} \langle \diamond \rangle^{\mu_S} \circ !phago_{n2}^\perp(exo^\perp) | \omega_{Target} \langle \omega_X \langle \diamond \rangle^{\mu_X} \rangle^{\mu_T} \rangle^{\mu_G}
 \end{aligned}$$

Table 10
Communication via a Mobile Vesicle: Evolution

Through vesicular trafficking, proteins follow routes involving intracellular locations (e.g. endoplasmic reticulum, Golgi apparatus or lysosomes) as well as the plasma membrane, in the case of endo- and exocytosis.

In [19], the PEP Brane calculus, plus the *bud* action, had been used to model a hypothetical scenario of communication via a mobile vesicle. As shown in Figure 2, a vesicle containing (embedded in its membrane) a substance that needs to be shuttled between two compartments, buds from a membrane *Source*. Then, it is engulfed by another compartment through phagocytosis and, finally, the mobile vesicle merges with the membrane *Target* releasing the transported substance in it. The encoding of this scenario is in Table 9. There, ω_G , ω_X , ω_{Source} , and ω_{Target} stand for skin membranes processes (not specified as not relevant at this level of abstraction).

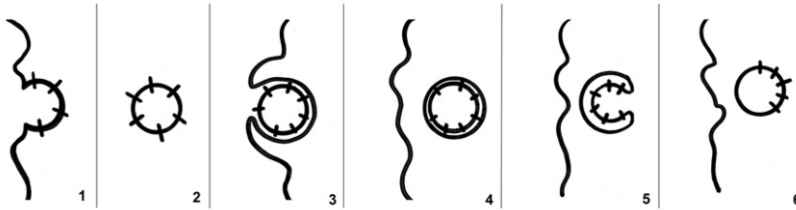


Fig. 2. A vesicle with the substance (rendered as |) embedded in its membrane, buds from the source membrane (1 and 2). The vesicle is then phagocytated by the target membrane (3 and 4) and merges with it (5 and 6) so delivering the substance to the final destination.

The dynamic evolution is reported in Table 10. The action bud_{n1} triggers the communication process, by leading to the gemmation of a vesicle containing the substance $\omega_X \langle \diamond \rangle^{\mu_X}$ coming from the membrane *Source*. The vesicle fuses with membrane *Target*, by means of actions *phago_{n2}* and *exo*.

The main entries of the Control Flow Analysis are reported in Table 11. The inclusion $\mu_X \in \mathcal{I}(\mu_T)$ reflects that the substance to be transmitted can eventually finish inside the target membrane.

$$\begin{aligned}
&\mu_G \in \mathcal{I}(*), \omega_G \in \mathcal{I}(\mu_G); \\
&bud_{n1}^\perp(vesicle), \omega_{Source} \in \mathcal{I}(\mu_S), \mu_S \in \mathcal{I}(\mu_G) \\
&bud_{n1}, \omega_X \in \mathcal{I}(\mu_X), \mu_X \in \mathcal{I}(\mu_S) \\
&phago_{n2}^\perp(exo^\perp), \omega_{Target} \in \mathcal{I}(\mu_T), \mu_T \in \mathcal{I}(\mu_G) \\
&phago_{n2}, exo \in \mathcal{I}(\mu_V), \mu_V \in \mathcal{I}(\mu_G), \mu_X \in \mathcal{I}(\mu_V) \\
&exo^\perp \in \mathcal{I}(\mu_E), \mu_E \in \mathcal{I}(\mu_T), \mu_V \in \mathcal{I}(\mu_E) \\
&\mathcal{I}(\mu_V) \subseteq \mathcal{I}(\mu_T) \text{ and, in particular, } \mu_X \in \mathcal{I}(\mu_T)
\end{aligned}$$

Table 11
Communication via a Mobile Vesicle: Analysis Results

6 Conclusion

We have introduced a Control Flow Analysis for the static approximation of the dynamic behaviour of processes, expressed in the MBD + PEP version of Brane calculi.

Like the ones in [12,13,3], our analysis is context-insensitive and also flow-insensitive and these features affect the analysis precision. Future work regards the improvement of our analysis precision, along the lines of [15,16]. Preliminary results make us confident that useful information on the behaviour of the analysed systems can be obtained and used in order to establish biological properties of the systems under consideration. In particular, the properties that can be expressed as reachability properties of the model. Furthermore, we can obtain information on the role played by the various elements composing the investigated system. Therefore, it becomes easier to evaluate the behaviour of the whole system when a single element is added or removed, following an approach similar to [4].

Acknowledgement

We are grateful to Pierpaolo Degano for his helpful discussions and to the referees for their valuable suggestions.

References

- [1] B. Alberts, D. Bray, J. Lewis, M. Raff, K. Roberts, and J.D. Watson. “Molecular Biology of the Cell”. Third Edition, Garland.
- [2] R. Barbuti, G. Caravagna, A. Maggiolo-Schettini, P. Milazzo, and G. Pardini. *The calculus of looping sequences*. In *Proc. of SFM’08*, Lecture Notes in Computer Science **5016** (2008), 387–423.
- [3] C. Bodei. *A Control Flow Analysis for Beta-binders with and without Static Compartments*. To appear in Theoretical Computer Science (2008), Elsevier.
- [4] C. Bodei, A. Bracciali, and D. Chiarugi. *On Deducing Causality in Metabolic Networks*. In BMC Bioinformatics, **9**(4) (2008).

- [5] N. Busi. *Towards a Causal Semantics for Brane Calculi*. In *What is it About Government that Americans Dislike*, 1945–1965, University Press, 2007.
- [6] L. Cardelli. *Brane calculi - interactions of biological membranes*. In *Proc. of Computational Methods in Systems Biology (CMSB'04)* Lecture Notes in Computer Science **3082** (2005), 257–280, Springer.
- [7] L. Cardelli and G. Paun. *An universality result for a (mem)brane calculus based on mate/drip operations*. *Int. J. Found. Comput. Sci.* **17** (1) (2006), 49–68, World Scientific.
- [8] V. Danos and J. Krivine. *Transactions in RCCS*. In *Proc. of Conference on Concurrency Theory (CONCUR'05)*, Lecture Notes in Computer Science **3653** (2005), 398–412, Springer.
- [9] V. Danos and Cosimo. Laneve. *Graphs for core molecular biology*. In *Proc. of Computational Methods in Systems Biology (CMSB'03)*, Lecture Notes in Computer Science **2602** (2003), 34–46, Springer.
- [10] R. Gori, F. Levi. *An Analysis for Proving Temporal Properties of Biological Systems*. In *Proc. of APLAS*, Lecture Notes in Computer Science **4279** (2006), 234–252.
- [11] R. Milner. “Communicating and mobile systems: the π -calculus”. Cambridge University Press, 1999.
- [12] F. Nielson, H. Riis-Nielson, D. Schuch-Da-Rosa, and C. Priami. *Static analysis for systems biology*. In *Proc. of Workshop on Systematics - dynamic biological systems informatics*, 1–6. Computer Science Press, 2004.
- [13] F. Nielson, H. Riis Nielson, C. Priami, and D. Schuch da Rosa. *Control Flow Analysis for BioAmbients*. *Electronic Notes in Theoretical Computer Science* **180(3)** (2007), 65–79, Elsevier.
- [14] G. Paun. *Computing with membranes (P systems): A variant*. *Int. J. Found. Comput. Sci.*, **11(1)** (2000), 167–181.
- [15] H. Pilegaard, F. Nielson, H. R. Nielson. *Static analysis of a Model of the LDL degradation pathway*. In *Proc. of Workshop on Computational Methods in Systems Biology (CMSB'05)*, 2005.
- [16] H. Pilegaard, F. Nielson, H. Riis Nielson. *Context Dependent Analysis of BioAmbients*. In *Proc. of Emerging Aspects of Abstract Interpretation '06*, 2006.
- [17] C. Priami and P. Quaglia. *Beta binders for biological interactions*. In *Proceedings of Computational Methods in Systems Biology (CMSB'04)*, Lecture Notes in Computer Science **3082** (2005), 20–33, Springer.
- [18] A. Regev and E. Shapiro. *Cellular Abstractions: Cells as Computation*, *Nature*, **419** (2002).
- [19] A. Vitale and G. Mauri. *Communication via Mobile Vesicles in Brane Calculi*. *Electronic Notes in Theoretical Computer Science* **171** (2) (2007), 187–196, Elsevier.

A Proofs

This appendix restates the lemmata and theorems presented earlier in the paper and gives the proofs of their correctness.

To establish the semantic correctness, the following auxiliary results are needed.

Fact A.1 *If $\mathcal{I} \models^{\mu_1} P$ and $\mathcal{I}(\mu_1) \subseteq \mathcal{I}(\mu_2)$, then $\mathcal{I} \models^{\mu_2} P$.*

Proof. By structural induction on P . This is straightforward, because the membrane identity is only used in recursive calls, to establish a fact like $\mu \in \mathcal{I}(\mu_i)$ or $a \in \mathcal{I}(\mu_i)$. We show just one case.

Case $P = \sigma\langle P \rangle^{\mu'}$. We have that $\mathcal{I} \models^{\mu_1} P$ is equivalent to $\mu' \in \mathcal{I}(\mu_1) \wedge A(\sigma) \in \mathcal{I}(\mu')$ and $\mathcal{I} \models^{\mu'} P$. Now, $\mu' \in \mathcal{I}(\mu_1)$ and $\mathcal{I}(\mu_1) \subseteq \mathcal{I}(\mu_2)$ imply $\mu' \in \mathcal{I}(\mu_2)$, and by induction hypothesis, we have that $\mathcal{I} \models^{\mu_2} P$. \square

Fact A.2 *If $\sigma \equiv \tau$ then $A(\sigma) = A(\tau)$.*

Proof. The proof amounts to a straightforward inspection of each of the clauses defining the structural congruence clauses relative to membranes. We show only two cases, the others are similar.

Case $\sigma_0|\sigma_1 \equiv \sigma_1|\sigma_0$. We have that $A(\sigma_0|\sigma_1) = A(\sigma_0) \cup A(\sigma_1) = A(\sigma_1|\sigma_0)$.

Case $\sigma \equiv \tau \Rightarrow \sigma|\rho \equiv \tau|\rho$. We have that $A(\sigma|\rho) = A(\sigma) \cup A(\rho)$. Now, since $\sigma \equiv \tau$, we have that $A(\sigma) = A(\tau)$ and therefore $A(\sigma|\rho) = A(\tau) \cup A(\rho)$, from which the required $A(\tau|\rho)$. \square

Lemma A.3 (Invariance of Structural Congruence) *If $P \equiv Q$ and $\mathcal{I} \models^\mu P$ then also $\mathcal{I} \models^\mu Q$.*

Proof. The proof amounts to a straightforward inspection of each of the clauses defining the structural congruence clauses.

Case $P_0 \circ P_1 \equiv P_1 \circ P_0$. We have that $\mathcal{I} \models^\mu P_0 \circ P_1$ is equivalent to $\mathcal{I} \models^\mu P_0 \wedge \mathcal{I} \models^\mu P_1$, that is equivalent to $\mathcal{I} \models^\mu P_1 \wedge \mathcal{I} \models^\mu P_0$ and therefore to $\mathcal{I} \models^\mu P_1 \circ P_0$.

Case $P_0 \circ (P_1 \circ P_2) \equiv (P_0 \circ P_1) \circ P_2$. We have that $\mathcal{I} \models^\mu P_0 \circ (P_1 \circ P_2)$ is equivalent to $\mathcal{I} \models^\mu P_0 \wedge \mathcal{I} \models^\mu P_1 \circ P_2$, that is equivalent to $\mathcal{I} \models^\mu P_0 \wedge \mathcal{I} \models^\mu P_1 \wedge \mathcal{I} \models^\mu P_2$ and, in turn, to $\mathcal{I} \models^\mu P_0 \circ P_1 \wedge \mathcal{I} \models^\mu P_2$, and therefore to $\mathcal{I} \models^\mu (P_0 \circ P_1) \circ P_2$.

Case $P_0 \circ \diamond \equiv P_0$. We have that $\mathcal{I} \models^\mu P_0 \circ \diamond$ is equivalent to $\mathcal{I} \models^\mu P_0 \wedge \mathcal{I} \models^\mu \diamond$, that is equivalent to $\mathcal{I} \models^\mu P_0 \wedge \text{true}$, and therefore to $\mathcal{I} \models^\mu P_0$.

Case $!\diamond \equiv \diamond$. We have that $\mathcal{I} \models^\mu !\diamond$ is equivalent to $\mathcal{I} \models^\mu \diamond$.

Case $!(P_0 \circ P_1) \equiv !P_0 \circ !P_1$. We have that $\mathcal{I} \models^\mu !(P_0 \circ P_1)$ is equivalent to $\mathcal{I} \models^\mu (P_0 \circ P_1)$, that is equivalent to $\mathcal{I} \models^\mu P_0 \wedge \mathcal{I} \models^\mu P_1$, that is equivalent to $\mathcal{I} \models^\mu !P_0 \wedge \mathcal{I} \models^\mu !P_1$ and therefore to $\mathcal{I} \models^\mu !P_0 \circ !P_1$.

Case $!!P \equiv !P$. We have that $\mathcal{I} \models^\mu !!P$ is equivalent to $\mathcal{I} \models^\mu !P$.

Case $!P \equiv P \circ !P$. We have that $\mathcal{I} \models^\mu !P$ is equivalent to $\mathcal{I} \models^\mu P$, that is equivalent to $\mathcal{I} \models^\mu P \wedge \mathcal{I} \models^\mu !P$, and therefore to $\mathcal{I} \models^\mu P \circ !P$.

Case $0\langle\diamond\rangle^{\mu'} \equiv \diamond$. We have that $\mathcal{I} \models^\mu 0\langle\diamond\rangle^{\mu'}$ is equivalent to $\mu' \in \mathcal{I}(\mu) \wedge A(0) = \emptyset \subseteq \mathcal{I}(\mu') \wedge \mathcal{I} \models^{\mu'} \diamond$, that is equivalent to $\mathcal{I} \models^{\mu'} \diamond \wedge \text{true}$, that implies true and, in turn, $\mathcal{I} \models^\mu \diamond$.

Case $P \equiv Q \Rightarrow P \circ R \equiv Q \circ R$. We have that $\mathcal{I} \models^\mu P \circ R$ is equivalent to $\mathcal{I} \models^\mu P \wedge \mathcal{I} \models^\mu R$, and from the hypothesis $\mathcal{I} \models^\mu P$, we have that $\mathcal{I} \models^\mu Q$. Therefore from $\mathcal{I} \models^\mu Q \wedge \mathcal{I} \models^\mu R$, we obtain the required $\mathcal{I} \models^\mu Q \circ R$.

Case $P \equiv Q \Rightarrow !P \equiv !Q$: similar.

Case $P \equiv Q \wedge \sigma \equiv \tau \Rightarrow \sigma\langle P\rangle^{\mu'} \equiv \tau\langle Q\rangle^{\mu'}$. We have that $\mathcal{I} \models^\mu \sigma\langle P\rangle^{\mu'}$ is equivalent to $\mu' \in \mathcal{I}(\mu) \wedge A(\sigma) \in \mathcal{I}(\mu') \wedge \mathcal{I} \models^{\mu'} P$. By Fact A.2, $A(\tau) \in \mathcal{I}(\mu')$, and by induction hypothesis, we have that $\mu' \in \mathcal{I}(\mu) \wedge A(\tau) \in \mathcal{I}(\mu') \wedge \mathcal{I} \models^{\mu'} Q$ and therefore $\mathcal{I} \models^\mu \tau\langle Q\rangle^{\mu'}$. \square

Theorem A.4 (Subject Reduction)

If $P \rightarrow Q$ and $\mathcal{I} \models^\mu P$ then also $\mathcal{I} \models^\mu Q$.

Proof. The proof is by induction on $P \rightarrow Q$. The proofs for the rules (*Par*) and (*Brane*) are straightforward, using the induction hypothesis and the clauses in Table 2. The proof for the (*Struct*) uses instead the induction hypothesis and Lemma A.3. The proofs for the basic actions in the lower part of Table 1 and in Table 4 are straightforward, using the clauses in Tables 2 and 5.

Case (Par). Let P be $P_0 \circ P_1$ and Q be $P'_0 \circ P_1$. We have to prove that $\mathcal{I} \models^\mu Q$. Now $\mathcal{I} \models^\mu P$ is equivalent to $\mathcal{I} \models^\mu P_0 \wedge \mathcal{I} \models^\mu P_1$. By induction hypothesis, we have that $\mathcal{I} \models^\mu P'_0$, and from $\mathcal{I} \models^\mu P'_0 \wedge \mathcal{I} \models^\mu P_1$ we obtain the required $\mathcal{I} \models^\mu Q$.

Case (Brane). Let P be $\sigma\langle P_0 \rangle^{\mu'}$ and Q be $\sigma\langle P'_0 \rangle^{\mu'}$. We have to prove that $\mathcal{I} \models^\mu \sigma\langle P'_0 \rangle^{\mu'}$. Now $\mathcal{I} \models^\mu P$ is equivalent to $\mu' \in \mathcal{I}(\mu) \wedge A(\sigma) \in \mathcal{I}(\mu') \wedge \mathcal{I} \models^{\mu'} P_0$. By induction hypothesis, we have that $\mathcal{I} \models^\mu P'_0$, and from $\mu' \in \mathcal{I}(\mu) \wedge A(\sigma) \in \mathcal{I}(\mu') \wedge \mathcal{I} \models^{\mu'} P'_0$ we obtain the required $\mathcal{I} \models^\mu Q$.

Case (Struct). Let $P \equiv P_0$, with $P_0 \rightarrow P_1$ such that $P_1 \equiv Q$. By Lemma A.3, we have that $\mathcal{I} \models^\mu P_0$, by induction hypothesis $\mathcal{I} \models^\mu P_1$ and, again by Lemma A.3, $\mathcal{I} \models^\mu Q$.

Case (Mate). Let P be $\text{mate}_n.\sigma|\sigma_0\langle P_0 \rangle^{\mu_0} \circ \text{mate}_n^\perp.\tau|\tau_0\langle P_1 \rangle^{\mu_1}$ and Q be $\sigma|\sigma_0|\tau|\tau_0\langle P_0 \circ P_1 \rangle^{\mu_{01}}$. We have that $\mathcal{I} \models^\mu P$ is equivalent to $\text{mate}_n \in \mathcal{I}(\mu_0) \wedge \text{mate}_n^\perp \in \mathcal{I}(\mu_1) \wedge \mu_0, \mu_1 \in \mathcal{I}(\mu) \text{ (1) and } A(\sigma) \in \mathcal{I}(\mu_0) \wedge A(\sigma_0) \in \mathcal{I}(\mu_0) \wedge \mathcal{I} \models^{\mu_0} P_0 \wedge A(\tau) \in \mathcal{I}(\mu_1) \wedge A(\tau_0) \in \mathcal{I}(\mu_1) \wedge \mathcal{I} \models^{\mu_1} P_1$. In particular, because of the closure conditions, from (1), we have that $\mu_{01} \in \mathcal{I}(\mu) \wedge \mathcal{I}(\mu_0) \cup \mathcal{I}(\mu_1) \subseteq \mathcal{I}(\mu_{01})$. Since $\mathcal{I}(\mu_i) \subseteq \mathcal{I}(\mu_{01})$ for $i = 0, 1$, then, by Fact A.1, we have that $A(\sigma) \in \mathcal{I}(\mu_{01}) \wedge A(\sigma_0) \in \mathcal{I}(\mu_{01}) \wedge \mathcal{I} \models^{\mu_{01}} P_0$ and $A(\tau) \in \mathcal{I}(\mu_{01}) \wedge A(\tau_0) \in \mathcal{I}(\mu_{01}) \wedge \mathcal{I} \models^{\mu_{01}} P_1$, and hence the required $\mathcal{I} \models^\mu Q$.

Case (Bud). Let P be $\text{bud}_n^\perp(\rho).\tau|\tau_0\langle \text{bud}_n.\sigma|\sigma_0\langle P_0 \rangle^{\mu_0} \circ P_1 \rangle^{\mu_1}$ and Q be $\rho\langle \sigma|\sigma_0\langle P_0 \rangle^{\mu_0} \rangle^{\mu_R} \circ \tau|\tau_0\langle P_1 \rangle^{\mu_1}$. We have that $\mathcal{I} \models^\mu P$ is equivalent to $\text{bud}_n \in \mathcal{I}(\mu_0) \wedge \text{bud}_n^\perp(\rho) \in \mathcal{I}(\mu_1) \wedge \mu_0 \in \mathcal{I}(\mu_1) \wedge \mu_1 \in \mathcal{I}(\mu) \text{ (1) and } A(\sigma) \in \mathcal{I}(\mu_0) \wedge A(\sigma_0) \in \mathcal{I}(\mu_0) \wedge \mathcal{I} \models^{\mu_0} P_0 \wedge A(\tau) \in \mathcal{I}(\mu_1) \wedge A(\tau_0) \in \mathcal{I}(\mu_1) \wedge \mathcal{I} \models^{\mu_1} P_1$. In particular, because of the closure conditions, from (1), we have that $A(\rho) \in \mathcal{I}(\mu_R) \wedge \mu_R \in \mathcal{I}(\mu)$ and $\mu_0 \in \mathcal{I}(\mu_R)$, and therefore the required $\mathcal{I} \models^\mu Q$.

Case (Drip). Let P be $\text{drip}(\rho).\sigma|\sigma_0\langle P_0 \rangle^{\mu_0}$ and Q be $\rho\langle \rangle^{\mu_R} \circ \sigma|\sigma_0\langle P_0 \rangle^{\mu_0}$. We have that $\mathcal{I} \models^\mu P$ is equivalent to $\text{drip}(\rho) \in \mathcal{I}(\mu_0) \wedge \mu_0 \in \mathcal{I}(\mu) \text{ (1) and } A(\sigma) \in \mathcal{I}(\mu_0) \wedge A(\sigma_0) \in \mathcal{I}(\mu_0) \wedge \mathcal{I} \models^{\mu_0} P_0$. In particular, because of the closure conditions, from (1), we have that $A(\rho) \in \mathcal{I}(\mu_R) \wedge \mu_R \in \mathcal{I}(\mu)$, and therefore the required $\mathcal{I} \models^\mu Q$.

Case (Phago). Let P be $\text{phago}_n.\sigma|\sigma_0\langle P_0 \rangle^{\mu_0} \circ \text{phago}_n^\perp(\rho).\tau|\tau_0\langle P_1 \rangle^{\mu_1}$ and Q be $\tau|\tau_0\langle \rho\langle \sigma|\sigma_0\langle P_0 \rangle^{\mu_0} \rangle^{\mu_R} \circ P_1 \rangle^{\mu_1}$. We have that $\mathcal{I} \models^\mu P$ is equivalent to $\text{phago}_n \in \mathcal{I}(\mu_0) \wedge \text{phago}_n^\perp(\rho) \in \mathcal{I}(\mu_1) \wedge \mu_0, \mu_1 \in \mathcal{I}(\mu) \text{ (1) and } A(\sigma) \in \mathcal{I}(\mu_0) \wedge A(\sigma_0) \in \mathcal{I}(\mu_0) \wedge \mathcal{I} \models^{\mu_0} P_0 \wedge A(\tau) \in \mathcal{I}(\mu_1) \wedge A(\tau_0) \in \mathcal{I}(\mu_1) \wedge \mathcal{I} \models^{\mu_1} P_1$. In particular, because of the closure conditions, from (1), we have that $A(\rho) \in \mathcal{I}(\mu_R) \wedge \mu_R \in \mathcal{I}(\mu_1) \wedge \mu_0 \in \mathcal{I}(\mu_R)$, and hence the required $\mathcal{I} \models^\mu Q$.

Case (Exo). Let P be $\text{exo}_n^\perp.\tau|\tau_0\langle \text{exo}_n.\sigma|\sigma_0\langle P_0 \rangle^{\mu_0} \circ P_1 \rangle^{\mu_1}$ and let Q be $P_0 \circ \sigma|\sigma_0|\tau|\tau_0\langle P_1 \rangle^{\mu_1}$. We have that $\mathcal{I} \models^\mu P$ is equivalent to $\text{exo}_n \in \mathcal{I}(\mu_0) \wedge \text{exo}_n^\perp \in \mathcal{I}(\mu_1) \wedge \mu_0 \in \mathcal{I}(\mu_1) \wedge \mu_1 \in \mathcal{I}(\mu) \text{ (1) and } A(\sigma) \in \mathcal{I}(\mu_0) \wedge A(\sigma_0) \in \mathcal{I}(\mu_0) \wedge \mathcal{I} \models^{\mu_0} P_0 \wedge A(\tau) \in \mathcal{I}(\mu_1) \wedge A(\tau_0) \in \mathcal{I}(\mu_1) \wedge \mathcal{I} \models^{\mu_1} P_1$. In particular, because of the closure conditions, from (1), we have that $A(\sigma), A(\sigma_0) \in \mathcal{I}(\mu_1) \wedge \mathcal{I}(\mu_0) \subseteq \mathcal{I}(\mu)$. By Fact A.1, we have that $\mathcal{I} \models^\mu P_0$ and therefore the required $\mathcal{I} \models^\mu Q$.

Case (Pino). Let P be $\text{pino}(\rho).\sigma|\sigma_0\langle P_0 \rangle^{\mu_0}$ and Q be $\sigma|\sigma_0\langle \rho\langle \rangle^{\mu_R} \circ P_0 \rangle^{\mu_0}$. We have that $\mathcal{I} \models^\mu P$ is equivalent to $\text{pino}(\rho) \in \mathcal{I}(\mu_0) \wedge \mu_0 \in \mathcal{I}(\mu) \text{ (1) and } A(\sigma) \in \mathcal{I}(\mu_0) \wedge A(\sigma_0) \in \mathcal{I}(\mu_0) \wedge \mathcal{I} \models^{\mu_0} P_0$. In particular, because of the closure condi-

tions, from (1), we have that $A(\rho) \in \mathcal{I}(\mu_R) \wedge \mu_R \in \mathcal{I}(\mu_0)$, and therefore the required $\mathcal{I} \models^\mu Q$. □

Theorem A.5 (Moore Family)

For any system P , the set $\{\mathcal{I} \mid \mathcal{I} \models^\mu P\}$ is a Moore family.

Proof. We proceed by structural induction on P . Let \mathbf{E} be a set of proposed estimates and let F and \mathcal{I}_j such that $\mathbf{F} \subseteq \mathbf{E} = \{\mathcal{I}_j \mid j \in F\}$. Next, define $\mathcal{I}' = \sqcap \mathbf{F}$. We have to check that $\mathcal{I}' \models^{\mu'} P$. We just consider one case. The others are similar.

Case $(\sigma\langle P_0 \rangle^{\mu'})$. Since $\forall j \in F : \mathcal{I}_j \models^\mu \sigma\langle P_0 \rangle^\mu$, then

$$\forall j \in F : \mu' \in \mathcal{I}_j(\mu) \wedge A(\sigma) \in \mathcal{I}_j(\mu') \wedge \mathcal{I}_j \models^{\mu'} P$$

Using the induction hypothesis and the fact that \mathcal{I}' is obtained in a pointwise way, we then obtain that

$$\mu' \in \mathcal{I}'(\mu) \wedge A(\sigma) \in \mathcal{I}'(\mu') \wedge \mathcal{I}' \models^{\mu'} P$$

thus establishing the required $\mathcal{I}' \models^\mu \sigma\langle P_0 \rangle^{\mu'}$. □