A Static Analysis for Brane Calculi providing Global Occurrence Counting Information

C. Bodei[b], L. Brodo[c], R. Gori[b], F. Levi[b], A. Bernini[a], D. Hermith[d]

[a]*Dip. di Biotecnologie, Chimica e Farmacia, Università di Siena, Italy*
[b]*Dipartimento di Informatica, Università di Pisa, Italy*
[c]*Dip. di Scienze Politiche, Scienze della Com. e Ing. dell'Informazione, Università di Sassari, Italy*
[d]*Dip. di Ingegneria dell'informazione e Scienze Matematiche, Università di Siena, Italy*

Abstract

In this paper we propose a static analysis for Brane Calculi [1], based on *Abstract Interpretation* [2] techniques. Our analysis statically approximates the dynamic behaviour of Brane systems, by providing a description of the possible hierarchical structure of membranes and of the processes possibly associated to each membrane, together with global occurrence counting information. Our analysis can be computed in polynomial time. We apply it to investigate several biological systems in which occurrence counting information plays a crucial role. In particular, our case study concerns the formation of the haemoglobin polymer in presence of alterations and investigate the influence that such alterations have on the ability of the haemoglobin polymer to bind oxygen molecules.

Keywords: Brane Calculi, Static analysis, Abstract interpretation, Occurrence counting.

1. Introduction

Biological systems and networks interact in diverse and complex ways. There is a large amount of data concerning single components and functions. What is not trivial is inferring the overall emerging behaviour of the system, which is not just the sum of the individual parts.

Systems biology can contribute to breach the biological complexity, by tracing high level properties of systems back to the interactions among their components. In particular, it collects data and integrates them into mathematical and computational models. These models make *in silico* simulations of biological behaviour possible and suitable to support *in vitro* or *in vivo* experiments. Often, these frameworks come with the drawback of high computational cost, due to the expensive inspection of the models that capture dynamic behaviour. Static analysis techniques imply a lower computational cost, at the price of losing precision. In particular, they provide safe over-approximations of the dynamic behaviour: all the events that the analysis predicts *may* happen, while all the non predicted events will surely *never* happen

Contribution. We propose here a *static analysis* for Brane Calculi [1], based on *Abstract Interpretation* (AI) [2] techniques. Brane calculi have been introduced to more closely model the behaviour of membrane-enclosed compartments, in terms of membrane interactions, such as fusion

and phagocytosis. As in traditional process algebras, brane processes representing membranes interact by means of actions and co-actions that synchronise. These calculi are useful for modelling and reasoning about a large class of biological systems. In our analysis, the standard information on control flow analysis (as the one in [3]) is enriched with global occurrence counting information. Technically, the analysis "mimics" the evolution of the system, by using abstract values in place of concrete ones and by modelling the consequences of each possible interaction. In particular, the analysis computes an abstract state describing the possible structure of all the derivatives of the system under investigation. It provides information on the possible membrane hierarchies and on the processes that may be associated to each membrane, together with their multiplicity. The global occurrence counting information associates each membrane and each process with a bound on the number of their simultaneous occurrences, independently of their location. This association is performed in a non-relational way and therefore it may impact on the precision of the prediction, but at the same time it allows us to use a simpler schema of abstraction. The analysis can be efficiently computed in polynomial time, thanks to the properties of the abstract semantics.

Why counting is important. Our analysis can be applied to investigate systems in which the information on the possible number of membranes (given by the occurrence counting information) plays a crucial role in biological terms. There are many biological examples at all levels, e.g. at biochemical, genetic and cellular level. The following three examples witness how we can exploit the statically collected occurrence counting information to more accurately distinguish among different dynamic behaviours.

At the protein level, it is well known that proteins rarely act in isolation but that bind other biomolecules to exert their functions. These biomolecules are often other proteins, and a very large number of proteins self-associate to form oligomers (molecules composed of two or a few subunits) in order to gain several different structural and functional advantages. It has been estimated that 35% or more of the proteins in a cell are oligomeric [4]. In the case of enzymes or carriers, i.e. proteins that provide active sites for processing or trafficking of small molecules, oligomerisation brings multiplication of sites carried for single entity; for instance if a protein is hexameric (i.e. it consists of six subunits) in its native state, and each subunit provides one active site, the functional unit will then provide six active sites. Furthermore, cooperation between the active sites is established through subunit interfaces, allowing for complex regulation of the biological activity. One popular example is haemoglobin, the protein in charge for carrying oxygen in the blood. In the humans the most common form of haemoglobin is a tetramer (i.e. a macromolecular complex composed by four subunits), and its four binding sites cross regulate each other upon binding/unbinding of oxygen molecules. In general, by following regulatory input, oligomeric proteins add/subtract ligands to their count of bound molecules. As a consequence, "counting" how many binding sites are in place and how many are bound/unbound by ligands, or activated/deactivated by other factors (inhibitors, alterations, mutations, and so on) is of primary importance for understanding the fine mechanisms underlying biological function regulation. In Section 5, we propose a model that allows us to investigate the formation of the haemoglobin oligomer in presence of alterations and to study its interactions with oxygen molecules. Thanks to the occurrence counting information, our analysis will be able to model the influence that such alterations have on the ability of the haemoglobin complex to bind oxygen molecules.

Going from proteins to nucleic acids, occurrence counting is necessary in many molecular biology techniques, e.g. fluorescence in situ hybridisation (FISH), a test that "maps" the genetic material in a person's cells. This test can be used to visualise specific genes and is of great value when dealing with cells presenting extra copies of a gene, like the HER2 oncogene involved in breast cancer development. Such gene encodes for the HER2 protein, a receptor receiving signals that stimulate the growth of breast cancer cells. As for most genes, two copies are present in the human genome, but cancer cells often differentiate into mutants showing more copies of it: the more copies of the HER2 gene are present, the more HER2 receptors the cells have, and the faster the cancer develops. FISH testing is done on breast cancer tissue removed during biopsy and allows for occurrences of the HER2 gene to be counted and the proper targeted therapy to be administered. Therefore, also in this setting information on multiplicity can provide useful insights on the behaviour of biological systems.

Occurrence counting is a key information in biology not only for the study of a protein mechanism of action or for evaluating genetic mutations, but also for the study of physiological processes concerning cell-scale events like cell cycle and signalling. As a guiding example we describe, in Section 3, two substantially different scenarios of communication via *mobile vesicles* (revising the encoding proposed in [5]). In both cases a source membrane can communicate a molecule X, via a mobile vesicle, to a target membrane. In the first version, the target membrane allows entering just *one* vesicle at the time, while in the second one, the target membrane allows *more than one* vesicle to enter at the time. Both scenarios correspond to real biological behaviours, as detailed at the end of Section 3. While polynomial analyses such as the ones in [6, 7, 3, 8, 9] would predict the same hierarchical structure for both scenarios, our analysis is able to predict that the first scenario corresponds to a one-to-one interaction between a vesicle and the target membrane, while the second scenario models a one-to-many interaction among a vesicle and the target membrane. Therefore, thanks to the occurrence counting information, our analysis succeeds in capturing the relevant difference in the behaviour of the two systems.

As a consequence, exploiting the occurrence counting information at the exemplified levels of structural genomics, biochemical pathways and cell signalling levels will be of great value in offering new insights, by more faithfully modelling some quantitative aspects of biological systems.

Interplay between control flow and occurrence counting analysis. Finally, not only the occurrence counting information is useful from a modelling point of view, but also from an analysis one. It is indeed exploited in our analysis to refine the information on the approximation of the control flow and, in particular, on the membrane hierarchy representing all the possible evolution of the system. Occurrence counting information, for example, can be used to detect that a membrane cannot interact with another occurrence of the same membrane if the total number of occurrences of such membrane in the system is less than two. This is the case illustrated in Example 9 of Section 4. Note that an analysis without occurrence counting information would predict that such interaction can occur and therefore would update the abstract hierarchy with all consequences of such interaction, even though they do not correspond to any real evolution of the system. Hence, occurrence counting information allows us to gain precision also on the description of the topological structure component of the evolutions of the system under investigation.

Related work. Static analysis techniques have been applied to many biologically-oriented calculi (see, e.g. the survey in [10]). In particular, Control Flow Analysis (CFA) has been applied to Beta-binders [11] in [7], to Brane Calculi [3], and to BioAmbients [12] in [6, 8, 13, 14]; while Abstract Interpretation to BioAmbients [15, 16, 17], to Brane Calculi [9, 18] and to Reaction Systems [19, 20]. Many of these works are inspired by the application of static analysis techniques [21, 22, 23] to Mobile Ambients (MA) [24], from which, many bio-inspired calculi derive.

Part of the above mentioned static analyses [6, 7, 3, 8] abstract, with different precision, the behaviour of the investigated system by providing an approximate description of structure of all derivatives. As a consequence, these approaches can be applied to establish invariant properties showing that some events will not happen in any derivatives of the analysed system. These techniques have polynomial complexity but they provide a less precise description of the possible topological structure of derivatives given that they do not maintain any information on occurrence counting. The richer contextual CFA in [13], and the AI-based analyses in [9, 18] improve the prediction accuracy, but, still, they are not able to observe the multiplicity. In particular, the analyses in [9, 18] provide an abstract version of the causal semantics for Brane Calculi proposed by Busi [25]. As a consequence they are able to statically capture the possible causal dependencies among interactions, whose identification can be exploited to better understand the modelled biological phenomena (see also [26]).

Instead, in [15], the authors present a counting analysis for BioAmbients that is able to express that an ambient can reside in alternative locations. This analysis has exponential complexity and provides accurate information about the number of occurrences of ambients, by counting the local number inside any ambient rather than their global number.

There are several static analysis frameworks that include occurrence counting information applied to MA and to π-calculus [27]. The analyses for MA introduced in [22, 28] are rather expensive from a computational point of view. The authors propose in [22] an exponential analysis for counting the global number of occurrences of ambients. The approach based on CFA substantially differs from our analysis, which is computed by calculating an abstract semantics. At the expense of a higher complexity, the shape analysis in [28] uses context-dependent counts for inferring a more accurate description of the internal structure of an ambient, by taking care of the local multiplicity of ambients. In [29, 30, 31] the author proposes a framework based on AI, applied to the π-calculus that, differently from the previous proposals, is non-uniform. Their non standard semantics is a refined semantics that is able to distinguish among recursive instances of agents, a feature missing in our framework. In this approach, the occurrence number of instances of agents is approximated by using a relational abstraction. The analysis is quite precise and efficient: its complexity is polynomial. For instance, the analysis described in [30] has a worst time cost of n^4, where n is the number of processes in the initial configuration. This approach is adequate to capture mutual exclusion and other security properties of complex mobile systems, formalised in π-calculus. In [21] the author proposes an adaptation of these techniques to MA, with a focus on security properties such as non-interference or confinement. In [32], the author proposes polynomial and precise analyses for MA and BioAmbients, based on both global and local counting. Specifically, the global analysis is the same as the one proposed in [30] for the π-calculus. This approach can handle mass preservation like invariants, which are ubiquitous in biological systems. In particular, it is able to preserve precision when dealing with continuations of replicated prefixes.

All the above works present powerful relational analyses that require a non-standard semantics for the chosen calculi, and therefore a more involved technical treatment. In these non-uniform analyses, it is possible to distinguish among different recursive instances of agents. Rather than relying on *ad hoc* version of the calculus, our proposal is based on the standard syntax and semantics of Brane Calculi, and therefore we can resort to standard proof techniques. On the one hand, our analysis is less powerful than the analyses in [29, 30, 31], since it is not relational and cannot capture non-uniform properties. On the other hand, it is not trivial either, since it is simpler but it can obtain information on occurrence counting and exploit it to refine the information on control flow.

The static analyses proposed in [16, 17, 14] rely on a different approach since they compute an abstract transition system to approximate the system behaviour, by still exploiting occurrence counting information. These techniques obviously provide useful information to verify temporal properties but at the price of a high complexity.

Structure of the Paper. In Section 2, we recall the semantics of Brane calculi. In Section 3, we introduce our twofold running example, based on two hypothetical scenarios of communication via mobile vesicles in the style of [5]. In Section 4, we present the analysis and in Section 4.4 we apply it to our examples. In Section 5 illustrate our framework by means of a case study, i.e. the binding mechanism of the haemoglobin oligomer. Conclusions are drawn in Section 6. The auxiliary results and the proofs can be found in Appendix A.

This article is the full and deeply revised version of the paper in [33], endowed with several biological systems. More precisely, the new contributions of this paper with respect to [33] are:

- an important refinement of the analysis, in particular the introduction of new revised abstract rules that greatly enhance the precision of the occurrence counting information;
- the introduction of extended biological motivations that witness the relevance of occurrence counting information;
- the introduction of several new examples for illustrating our approach and its advantages;
- the presentation of a new case study to test our analysis in the biological setting;
- a rigorous formalisation of the relation between concrete and abstract semantics in the abstraction framework;
- the inclusion of extended definitions, results and proofs.

2. An Overview on Brane Calculus

The Brane Calculi [1] are a family of calculi based on a set of primitives inspired by biological membrane interactions. We focus here on the full version of the calculus with two sets of membrane primitives: Mate/Bud/Drip (MBD) and Phago/ Exo/Pino (PEP).

The Phago/Exo/Pino(PEP) actions represent the biological processes of endocytosis and exocytosis. The first indicates the process of incorporating external material into a cell, by engulfing

P, Q	::=	$\diamond \mid P \circ Q \mid !P \mid \sigma(\!\lvert P \rvert\!)^{\Gamma}$	systems $\mathtt{Sys}$
σ, τ	::=	$0 \mid \sigma \lvert \tau \mid !\sigma \mid a^{\lambda}.\sigma$	membrane processes $\mathtt{Proc}$
a, b	::=	$\mathtt{phago}_n \mid \overline{\mathtt{phago}}_n(\rho) \mid \mathtt{exo}_n \mid \overline{\mathtt{exo}}_n \mid \mathtt{pino}(\rho) \mid$	PEP actions $\mathtt{Act}$
		$\mathtt{mate}_n \mid \overline{\mathtt{mate}}_n \mid \mathtt{bud}_n \mid \overline{\mathtt{bud}}_n(\sigma) \mid \mathtt{drip}(\sigma)$	MBD actions $\mathtt{Act}$

Table 1: Syntax of (labelled) Brane Calculi.

it with the cell membrane, while the second one indicates the reverse process. Endocytosys is rendered by two more basic operations: phagocytosis (*phago*), which consists in engulfing just one external membrane, and pinocytosis (*pino*), which consists in engulfing zero external membranes. Exocytosis is instead denoted by (*exo*). The Mate/Bud/Drip (MBD) actions are instead inspired by membrane fusion and splitting. Because membrane fission is an uncontrollable process that can split a membrane at an arbitrary place, it is replaced by two simpler operations: budding (*bud*), which is splitting one internal membrane, and dripping (*drip*), which consists in splitting zero internal membranes. Membrane fusion, or merging, is called mating (*mate*).

It is worth mentioning that the MBD actions can be encoded with a sequence of PEP actions [1]. However, from the analysis point of view it is more convenient to use the MBD actions as primitives.

We introduce the syntax and the semantics for the calculus, considering a *labelled version* of the calculus. As usual in static analysis, labels are exploited to support the analysis (presented in Sect. 4) and *do not affect* the dynamic semantics of the calculus.

A membrane system consists of nested membranes, where each membrane has associated a membrane process. The syntax of the labelled calculus is described in Tab. 1, where n is taken from a countable set $\mathcal{N}$ of names, and where we write $P \in \mathtt{Sys}$ for *systems*, $\sigma \in \mathtt{Proc}$ for *membrane processes*, and $a \in \mathtt{Act}$ for *actions*. Each membrane is annotated with a *membrane label* $\Gamma \in \widehat{\mathtt{Lab}}_{\mathcal{M}}$ and each action is annotated with a *process label* $\lambda \in \mathtt{Lab}_{\mathcal{P}}$.

We therefore need two distinct sets of labels. We have the set of *process labels* $\mathtt{Lab}_{\mathcal{P}}$, ranged over by $\alpha, \beta, \gamma \ldots$. Moreover, given a set of basic membrane labels $\mathtt{Lab}_{\mathcal{M}}$, we have the associated set of *membrane labels* $\widehat{\mathtt{Lab}}_{\mathcal{M}}$, ranged over by $\Delta, \Gamma, \Psi \ldots$. The set $\widehat{\mathtt{Lab}}_{\mathcal{M}}$ is inductively defined as follows: (i) $\mathtt{Lab}_{\mathcal{M}} \subseteq \widehat{\mathtt{Lab}}_{\mathcal{M}}$; (ii) if $\Gamma, \Delta \in \widehat{\mathtt{Lab}}_{\mathcal{M}}$ and $\lambda, \mu \in \mathtt{Lab}_{\mathcal{P}}$, then $\mathtt{phago}(\Gamma, \Delta, \lambda, \mu)$, $\mathtt{mate}(\Gamma, \Delta, \lambda, \mu)$, $\mathtt{bud}(\Gamma, \Delta, \lambda, \mu)$, and $\mathtt{drip}(\Delta, \lambda)$ belong to $\widehat{\mathtt{Lab}}_{\mathcal{M}}$.

The system $\sigma(\!\lvert P \rvert\!)^{\Gamma}$ describes a *membrane*, decorated by label Γ[1], that contains the system P and that performs the *membrane process* σ, describing its interaction capabilities. The construct $a^{\lambda}.\sigma$ defines a sequential process that executes an action a, decorated by label λ, and then behaves as the process σ. We adopt standard syntactical abbreviations: a^{λ} stands for $a^{\lambda}.0$, $(\!\lvert P \rvert\!)^{\Gamma}$ stands for $0(\!\lvert P \rvert\!)^{\Gamma}$, and $\sigma(\!\lVert\rVert\!)^{\Gamma}$ is a shorthand for $\sigma(\!\lvert\diamond\rvert\!)^{\Gamma}$. We recall the precedence rules, according to which, $a.\tau \lvert \sigma$ stands for $(a.\tau)\lvert\sigma$, and $!\sigma\lvert\tau$ stands for $(!\sigma)\lvert\tau$.

The semantics of the calculus is given by the reduction rules in Table 3, modulo the structural congruence rules. The structural congruence on systems and processes is the least congruence satisfying the clauses in Table 2, whose definition is standard.

[1]For the sake of brevity, from now on, we will usually write membrane Γ, instead of membrane labelled by Γ.

$P \circ Q \equiv Q \circ P$	$\sigma \mid \tau \equiv \tau \mid \sigma$
$P \circ (Q \circ R) \equiv (P \circ Q) \circ R$	$\sigma \mid (\tau \mid \rho) \equiv (\sigma \mid \tau) \mid \rho$
$P \circ \diamond \equiv P$	$\sigma \mid 0 \equiv \sigma$
$!\diamond \equiv \diamond$	$!0 \equiv 0$
$!(P \circ Q) \equiv !P \circ !Q$	$!(\sigma \mid \tau) \equiv !\sigma \mid !\tau$
$!!P \equiv !P$	$!!\sigma \equiv !\sigma$
$!P \equiv P \circ !P$	$!\sigma \equiv \sigma \mid !\sigma$
$\sigma \equiv \tau \Rightarrow \sigma \mid \rho \equiv \tau \mid \rho$	$P \equiv Q \Rightarrow P \circ R \equiv Q \circ R$
$\sigma \equiv \tau \Rightarrow !\sigma \equiv !\tau$	$P \equiv Q \Rightarrow !P \equiv !Q$
$\sigma \equiv \tau \Rightarrow a^{\lambda}.\sigma \equiv a^{\lambda}.\tau$	$P \equiv Q \wedge \sigma \equiv \tau \Rightarrow \sigma(\!\lvert P \rvert\!)^{\Gamma} \equiv \tau(\!\lvert Q \rvert\!)^{\Gamma}$
$0(\!\lvert\rvert\!)^{\Gamma} \equiv \diamond$	

Table 2: Structural congruence for (labelled) Brane.

The (labelled) reduction rules given complete the definition of the semantics. The labelled transition relation is $\xrightarrow{l}$, where $P \xrightarrow{l} Q$ denotes that the system P evolves into the system Q performing a reaction described by the *transition label* $l \in \mathtt{Lab}_{\mathcal{T}}$. The set of transition labels $\mathtt{Lab}_{\mathcal{T}}$ (ranged over by $l_1, l_2 \ldots$) is defined as follows:

$$\begin{array}{rcl} \mathtt{Lab}_{\mathcal{T}} & = & \{\mathtt{pino}_1(\Delta,\lambda), \mathtt{drip}_1(\Delta,\lambda) \mid \Delta \in \widehat{\mathtt{Lab}}_{\mathcal{M}}, \lambda \in \mathtt{Lab}_{\mathcal{P}}\} \cup \\ & & \{\mathtt{mate}_1(\Gamma,\Delta,\lambda,\mu), \mathtt{bud}_1(\Gamma,\Delta,\lambda,\mu), \mathtt{exo}_1(\Gamma,\Delta,\lambda,\mu), \mathtt{phago}_1(\Gamma,\delta,\lambda,\mu) \mid \\ & & \Gamma,\Delta \in \widehat{\mathtt{Lab}}_{\mathcal{M}}, \lambda,\mu \in \mathtt{Lab}_{\mathcal{P}}\} \end{array}$$

Besides the standard reduction rule for congruence (Struct), and the contextual rules to propagate reductions across parallel composition (Par) and membrane nesting (Brane), there are the axioms specific of the membrane actions.

Rule (Phago) models the inclusion of an external membrane, labelled by Δ, inside a membrane, labelled by Γ. The two membranes Δ and Γ exercise the actions $\mathtt{phago}_n^{\lambda}$ and $\overline{\mathtt{phago}}_n^{\mu}(\rho)$, respectively. Once engulfed, the membrane Δ is enclosed inside a new membrane with label $\mathtt{phago}(\Delta,\Gamma,\lambda,\mu)$, which has associated the process ρ. The corresponding transition label is given by $\mathtt{phago}_1(\Delta,\Gamma,\lambda,\mu)$. Rule (Exo) models the expulsion of the membrane Δ, outside the external membrane Γ, triggered by the actions $\mathtt{exo}_n^{\lambda}$ and $\overline{\mathtt{exo}}_n^{\mu}$, respectively. The corresponding transition label is $\mathtt{exo}_1(\Delta,\Gamma,\lambda,\mu)$. In the rule (Pino), the membrane Δ, creates a new empty membrane, labelled by $\mathtt{pino}(\Delta,\lambda)$, inside itself. The action $\mathtt{pino}^{\lambda}(\rho)$ is equipped with a process ρ that will be associated to the new membrane. The corresponding transition label is $\mathtt{pino}_1(\Delta,\lambda)$. Rule (Mate) models the fusion of two membranes, labelled by Δ and Γ, which exercise the actions $\mathtt{mate}_n^{\lambda}$ and $\overline{\mathtt{mate}}_n^{\mu}$, respectively. The membrane introduced by the fusion takes the label $\mathtt{mate}(\Delta,\Gamma,\lambda,\mu)$ and has associated the parallel composition of the residual processes of the two membranes. The corresponding transition label is $\mathtt{mate}_1(\Delta,\Gamma,\lambda,\mu)$. Moreover, in the rule (Bud), the membrane Γ expels the child membrane Δ, performing the actions $\overline{\mathtt{bud}}_n^{\mu}(\rho)$ and $\mathtt{bud}_n^{\lambda}$, respectively. The membrane Δ is wrapped inside a new membrane with label $\mathtt{bud}(\Delta,\Gamma,\lambda,\mu)$ and has associated the process ρ. The corresponding transition label is $\mathtt{bud}_1(\Delta,\Gamma,\lambda,\mu)$. Finally, in the rule (Drip), the membrane Δ, performing the action $\mathtt{drip}^{\lambda}(\rho)$, creates a new empty membrane, labelled by $\mathtt{drip}(\Delta,\lambda)$, which has associated the process ρ. The corresponding transition label is $\mathtt{drip}_1(\Delta,\lambda)$.

$$\text{(Par)}\ \frac{P \xrightarrow{l} Q}{P \circ R \xrightarrow{l} Q \circ R} \qquad \text{(Brane)}\ \frac{P \xrightarrow{l} Q}{\sigma(\!|P|\!)^{\Gamma} \xrightarrow{l} \sigma(\!|Q|\!)^{\Gamma}} \qquad \text{(Struct)}\ \frac{P \equiv P' \ \wedge\ P' \xrightarrow{l} Q' \ \wedge\ Q' \equiv Q}{P \xrightarrow{l} Q}$$

(Phago)	$\mathtt{phago}^{\lambda}_{n}.\sigma\vert\sigma_0(\!|P|\!)^{\Delta} \circ \overline{\mathtt{phago}}^{\mu}_{n}(\rho).\tau\vert\tau_0(\!|Q|\!)^{\Gamma} \xrightarrow{\mathtt{phago}_1(\Delta,\Gamma,\lambda,\mu)} \tau\vert\tau_0(\!|\rho(\!|\sigma\vert\sigma_0(\!|P|\!)^{\Delta}|\!)^{\mathtt{phago}(\Delta,\Gamma,\lambda,\mu)} \circ Q|\!)^{\Gamma}$
(Exo)	$\overline{\mathtt{exo}}^{\mu}_{n}.\tau\vert\tau_0(\!|\mathtt{exo}^{\lambda}_{n}.\sigma\vert\sigma_0(\!|P|\!)^{\Delta} \circ Q|\!)^{\Gamma} \xrightarrow{\mathtt{exo}_1(\Delta,\Gamma,\lambda,\mu)} P \circ \sigma\vert\sigma_0\vert\tau\vert\tau_0(\!|Q|\!)^{\Gamma}$
(Pino)	$\mathtt{pino}^{\lambda}(\rho).\sigma\vert\tau(\!|P|\!)^{\Delta} \xrightarrow{\mathtt{pino}_1(\Delta,\lambda)} \sigma\vert\tau(\!|\rho(\!|\emptyset|\!)^{\mathtt{pino}(\Delta,\lambda)} \circ P|\!)^{\Delta}$
(Mate)	$\mathtt{mate}^{\lambda}_{n}.\sigma\vert\sigma_0(\!|P|\!)^{\Delta} \circ \overline{\mathtt{mate}}^{\mu}_{n}.\tau\vert\tau_0(\!|Q|\!)^{\Gamma} \xrightarrow{\mathtt{mate}_1(\Delta,\Gamma,\lambda,\mu)} \sigma\vert\sigma_0\vert\tau\vert\tau_0(\!|P \circ Q|\!)^{\mathtt{mate}(\Delta,\Gamma,\lambda,\mu)}$
(Bud)	$\overline{\mathtt{bud}}^{\mu}_{n}(\rho).\tau\vert\tau_0(\!|\mathtt{bud}^{\lambda}_{n}.\sigma\vert\sigma_0(\!|P|\!)^{\Delta} \circ Q|\!)^{\Gamma} \xrightarrow{\mathtt{bud}_1(\Delta,\Gamma,\lambda,\mu)} \rho(\!|\sigma\vert\sigma_0(\!|P|\!)^{\Delta}|\!)^{\mathtt{bud}(\Delta,\Gamma,\lambda,\mu)} \circ \tau\vert\tau_0(\!|Q|\!)^{\Gamma}$
(Drip)	$\mathtt{drip}^{\lambda}(\rho).\sigma\vert\tau(\!|P|\!)^{\Delta} \xrightarrow{\mathtt{drip}_1(\Delta,\lambda)} \rho(\!|\emptyset|\!)^{\mathtt{drip}(\Delta,\lambda)} \circ \sigma\vert\tau(\!|P|\!)^{\Delta}$

Table 3: Reduction rules for (labelled) Brane.

The *semantics* (called the *collecting semantics*) that our analysis will approximate in Section 4, is given in terms of finite paths. More precisely, the semantics of a system P is given the set of finite paths starting from P. We define finite paths as follows.

Definition 1 (Finite Paths)**.** *Let $P \in \mathtt{Sys}$ be a system. The* finite paths *of P are inductively defined as follows:*

1. *if $p = P \xrightarrow{l_1} P_1$ is obtained by applying the rules and axioms of Table 3 then p is a* path *of P;*
2. *if $P \xrightarrow{l_1} P_1 \ldots P_{n-1} \xrightarrow{l_n} P_n$ is a path of P and $P_n \xrightarrow{l_{n+1}} P_{n+1}$ is obtained by applying the rules and axioms of Table 3 then $P \xrightarrow{l_1} P_1 \ldots P_n \xrightarrow{l_{n+1}} P_{n+1}$ is a* path *of P.*

We use $\mathcal{T}(P)$ to denote the set of finite paths of P, and $\widehat{\mathcal{T}}$ to denote $\bigcup_{\{P \in \mathtt{Sys}\}} \mathcal{T}(P)$.

Definition 2 (The Collecting Semantics)**.** *We define a function $\mathfrak{I} : \mathtt{Sys} \to \widehat{\mathcal{T}}$ such that for $P \in \mathtt{Sys}$ we have $\mathfrak{I}(P) = \mathcal{T}(P)$.*

Moreover, we assume that the initial system P is *well labelled*, i.e. that the process labels that occur in P are all distinct. As it will be clear in Section 4, this is a condition necessary to obtain the correctness of our analysis. Intuitively, the well labelling condition guarantees that the processes with the same label correspond to different instances of the same process under the scope of a replication, in any derivative of P.

Remark 1. *It is worth briefly discussing the role of labels in our calculus. Process and membrane labels will be exploited in the analysis to maintain the information on the topological*

structure of systems. To decorate the new membranes introduced by reactions of the calculus, we adopt composite membrane labels that record the labels of the membrane and of the actions that interact. This labelling technique may introduce arbitrarily nested membrane labels (e.g. $\mathtt{bud}(\Delta, \mathtt{pino}(\Gamma, \delta), \lambda, \mu)$*) and even an infinite number of membrane labels.*

Furthermore, we decorate the reduction steps with transition labels giving information on the labels of the actions and on the membranes involved in the interaction. This information will be exploited in the analysis to establish a correspondence between reduction steps and abstract transitions.

3. Communication via Mobile Vesicles

To illustrate our analysis, we consider two different hypothetical scenarios of communication via transport vesicles.

The role of transport vesicles is strictly related to protein trafficking which is a central mechanism in cell biology by which proteins are transported to the appropriate destinations in the cell or outside of it. A well-known principle that governs protein trafficking is the transport of membrane and soluble proteins from one membrane-bounded compartment to another. This process is mediated by transport vesicles. These collect "cargo" proteins in buds arising from the membrane of one compartment and then deliver the same to the next compartment by fusing with the membrane of that compartment. Since vesicular transport is essential in the organisation of eukaryotic cells, understanding the mechanisms that control vesicle budding and fusion is an active research topic in cell biology.

More in detail, as shown in Figure 1, we model a vesicle containing (embedded in its membrane) a molecule that needs to be shuttled between two compartments, buds from a membrane *Source*. Then, it is engulfed by another compartment (the *Target* membrane) through phagocytosis (creating a coat membrane containing the vesicle) and, finally, the coat of the mobile vesicle is decomposed within the membrane *Target*, releasing the transported molecule in it.

Here, we introduce a more general situation, by presenting two substantially different scenarios of communication via mobile vesicles. In both cases *Source* can communicate the molecule X, via one of the mobile vesicle that can be created, to a membrane *Target*. In the first version, the membrane *Target* allows to enter just one vesicle at a time while in the second one the membrane *Target*$'$ allows any number of vesicles to enter at a time. Both the systems can be rendered along the lines of [5].

Example 1. *The encoding is described by the system Q in the upper part of Table 4, where the processes σ_Q, σ_{target}, τ_X, τ', τ'' and ρ stand for membranes processes (not specified as not relevant at this level of abstraction), and where we decorate actions and membranes with basic membrane labels in* $\mathtt{Lab}_{\mathcal{M}}$[2]*. Our system is $Q \stackrel{def}{=} \sigma_Q(\!|Source \circ Target|\!)^{skin}$, where Source communicates the molecule X, via one of the mobile vesicle that can be created, to the membrane Target that only accepts one vesicle at a time. The molecule X to be transmitted is initially enclosed inside a membrane labelled by* Γ*. Such membrane triggers the communication process, by exercising the*

[2]We also assume the system Q to be well labelled.

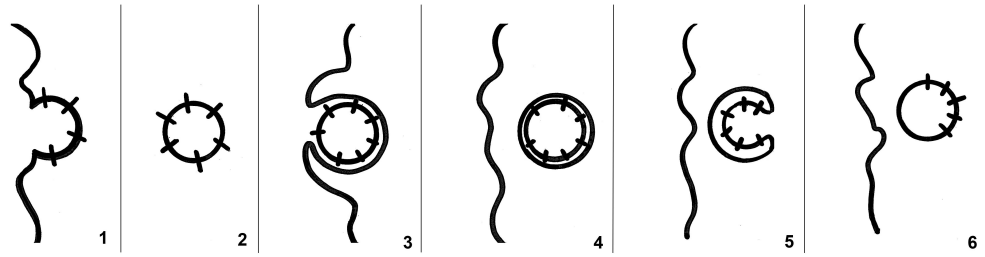

Figure 1: *A vesicle with the substance (rendered as |) embedded in its membrane, buds from the source membrane (1 and 2). The vesicle is then phagocytosed by the target membrane (3 and 4) and merges with it (5 and 6) so delivering the substance to the final destination.*

action bud_n^γ *and leading to the gemmation of the vesicle that will transport the molecule into the target.*

In the lower part of Table 4, we illustrate a possible dynamic evolution of the system, described by a finite path, in which two vesicles are created. The membrane Γ *buds a first time from the membrane labelled by source thus creating the vesicle modelled by the membrane* Π_1*, as illustrated in Figure 1 (1) and (2). Similarly, the membrane* Γ *buds a second time (still with a transition labelled* l_1*) and creates another vesicle with label* Π_1*. Since the bud prefixes are both replicated, any number of vesicles may be created.*

Then, one of the membranes Π_1 *is engulfed by the membrane labelled by target via a phago reaction (labelled* l_2*), as shown in Figure 1 (3) and (4). The new membrane is decorated with the label* Π_2*.*

Finally, the membrane Π_1 *interacts with the membrane* Γ *via an exo reaction (labelled* l_3*), thus releasing the molecule X inside membrane* Π_2*, as shown in Figure 1 (5) and (6).*

In this scenario only one vesicle can fuse with the membrane target, because the action $\overline{\mathsf{phago}}_n^\delta(\rho)$ *can be exercised only once.*

It should be clear that in this case only one occurrence of the molecule X can be inside the membrane *target*, despite the fact that more vesicles can be ready to be engulfed by the membrane target.

Example 2. *To model the second scenario, we just modify the membrane process associated with the membrane target, by replicating the action* $\overline{\mathsf{phago}}_n^\delta(\rho)$*. In this case the membrane target allows any number of molecules X to enter inside and fuse. The corresponding encoding is described by the system* Q' *in the upper part of Table 5, by using process and membrane labels as in Table 4. In the lower part of Table 5, we present a dynamic evolution in which two vesicles are created as in the previous scenario. In this case, differently from before, both the created vesicles can be engulfed and fused by the membrane target. Each membrane* Π_1 *is indeed engulfed via a phago reaction (labelled* l_2*), and interacts with the membrane* Γ *via an exo reaction (labelled* l_3*), thus releasing the molecule X inside membrane* Π_2*.*

It should be clear that, in this case, more occurrences of the molecule X may end up inside the membrane *target*.

$Q \stackrel{def}{=} \sigma_Q(\!|Source \circ Target|\!)^{skin}$ $\quad Target \stackrel{def}{=} \overline{\mathtt{phago}}^{\delta}_n(\rho)|\sigma_{target}(\!|\diamond|\!)^{target}$

$Vesicle \stackrel{def}{=} \mathtt{phago}^{\mu}_n.\overline{\mathtt{exo}}^{\beta}_n|\tau'$ $\quad Source \stackrel{def}{=} \sigma_{source}(\!|!(\sigma_X(\!|X|\!)^{\Gamma})|\!)^{source}$

$\sigma_X \stackrel{def}{=} \mathtt{bud}^{\gamma}_n.\mathtt{exo}^{\nu}_n|\tau_X$ $\quad \sigma_{source} \stackrel{def}{=} !\overline{\mathtt{bud}}^{\lambda}_n(Vesicle)|\tau''$

$Q = \sigma_Q(\!|Source \circ Target|\!)^{skin} \equiv$

$\sigma_Q(\!|!\overline{\mathtt{bud}}^{\lambda}_n(Vesicle)|\tau''(\!|!(\mathtt{bud}^{\gamma}_n.\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma})|\!)^{source} \circ Target|\!)^{skin} \equiv$

$\sigma_Q(\!|\underline{\overline{\mathtt{bud}}^{\lambda}_n(Vesicle)}|!\overline{\mathtt{bud}}^{\lambda}_n(Vesicle)|\tau''(\!|\underline{\mathtt{bud}^{\gamma}_n}.\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma} \circ !(\mathtt{bud}^{\gamma}_n.\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma})|\!)^{source} \circ Target|\!)^{skin} \xrightarrow{l_1}$

$\sigma_Q(\!|!\overline{\mathtt{bud}}^{\lambda}_n(Vesicle)|\tau''(\!|!(\mathtt{bud}^{\gamma}_n.\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma})|\!)^{source} \circ Vesicle(\!|\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ Target|\!)^{skin} \equiv$

$\sigma_Q(\!|Source \circ Vesicle(\!|\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ Target|\!)^{skin} \xrightarrow{l_1}$

$\sigma_Q(\!|Source \circ Vesicle(\!|\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ Vesicle(\!|\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ Target|\!)^{skin} \equiv$

$\sigma_Q(\!|Source \circ \mathtt{phago}^{\mu}_n.\overline{\mathtt{exo}}^{\beta}_n|\tau'(\!|\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ \underline{\mathtt{phago}^{\mu}_n}.\overline{\mathtt{exo}}^{\beta}_n|\tau'(\!|\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ$
$\underline{\overline{\mathtt{phago}}^{\delta}_n(\rho)}|\sigma_{target}(\!|\diamond|\!)^{target}|\!)^{skin} \xrightarrow{l_2}$

$\sigma_Q(\!|Source \circ Vesicle(\!|\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ \sigma_{target}(\!|\rho(\!|\underline{\overline{\mathtt{exo}}^{\beta}_n}|\tau'(\!|\underline{\mathtt{exo}^{\nu}_n}|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1}|\!)^{\Pi_2}|\!)^{target}|\!)^{skin} \xrightarrow{l_3}$

$\sigma_Q(\!|Source \circ Vesicle(\!|\mathtt{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ \sigma_{target}(\!|\rho(\!|X \circ \tau_X|\tau'(\!|\diamond|\!)^{\Pi_1}|\!)^{\Pi_2}|\!)^{target}|\!)^{skin}$

where

$l_1 = \mathtt{bud}_1(\Gamma, source, \gamma, \lambda)$ $\quad l_2 = \mathtt{phago}_1(\Pi_1, target, \mu, \delta)$ $\quad l_3 = \mathtt{exo}_1(\Gamma, \Pi_1, \nu, \beta)$

$\Pi_1 = \mathtt{bud}(\Gamma, source, \gamma, \lambda)$ $\quad \Pi_2 = \mathtt{phago}(\Pi_1, target, \mu, \delta)$

Table 4: First scenario: encoding (upper part) and a possible evolution (lower part), where the prefixes involved in each transition are underlined.

$Q' \stackrel{def}{=} \sigma_Q(\!|Source \circ Target'|\!)^{skin}$ $\quad Target' \stackrel{def}{=} (!\overline{\mathsf{phago}}^{\delta}_n(\rho))|\sigma_{target}(\!|\diamond|\!)^{target}$

$Vesicle \stackrel{def}{=} \mathsf{phago}^{\mu}_n.\overline{\mathsf{exo}}^{\beta}_n|\tau'$ $\quad Source \stackrel{def}{=} \sigma_{source}(\!|!(\sigma_X(\!|X|\!)^{\Gamma})|\!)^{source}$

$\sigma_X \stackrel{def}{=} \mathsf{bud}^{\gamma}_n.\mathsf{exo}^{\nu}_n|\tau_X$ $\quad \sigma_{source} \stackrel{def}{=} !\overline{\mathsf{bud}}^{\lambda}_n(Vesicle)|\tau''$

$Q' = \sigma_Q(\!|Source \circ Target'|\!)^{skin} \equiv$

$\sigma_Q(\!|\underline{\overline{\mathsf{bud}}^{\lambda}_n(Vesicle)}|!\overline{\mathsf{bud}}^{\lambda}_n(Vesicle)|\tau''(\!|(\underline{\mathsf{bud}^{\gamma}_n}.\mathsf{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma})\circ!(\mathsf{bud}^{\gamma}_n.\mathsf{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma})|\!)^{source} \circ Target'|\!)^{skin} \xrightarrow{l_1}$

$\sigma_Q(\!|!\overline{\mathsf{bud}}^{\lambda_1}_n(Vesicle)|\tau''(\!|!(\mathsf{bud}^{\gamma}_n.\mathsf{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma})|\!)^{source} \circ Vesicle(\!|\mathsf{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ Target'|\!)^{skin} \xrightarrow{l_1}$

$\sigma_Q(\!|Source \circ Vesicle(\!|\mathsf{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ Vesicle(\!|\mathsf{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ Target'|\!)^{skin} \equiv$

$\sigma_Q(\!|Source \circ \mathsf{phago}^{\mu}_n.\overline{\mathsf{exo}}^{\beta}_n|\tau'(\!|\mathsf{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ \underline{\mathsf{phago}^{\mu}_n}.\overline{\mathsf{exo}}^{\beta}_n|\tau'(\!|\mathsf{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ$
$\underline{\overline{\mathsf{phago}}^{\delta}_n(\rho)}|!\overline{\mathsf{phago}}^{\delta}_n(\rho)|\sigma_{target}(\!|\diamond|\!)^{target}|\!)^{skin} \xrightarrow{l_2}$

$\sigma_Q(\!|Source \circ \mathsf{phago}^{\mu}_n.\overline{\mathsf{exo}}^{\beta}_n|\tau'(\!|\mathsf{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ$
$!\overline{\mathsf{phago}}^{\delta}_n(\rho)|\sigma_{target}(\!|\rho(\!|\underline{\overline{\mathsf{exo}}^{\beta}_n|\tau'}(\!|\underline{\mathsf{exo}^{\nu}_n}|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1}|\!)^{\Pi_2}|\!)^{target}|\!)^{skin} \xrightarrow{l_3}$

$\sigma_Q(\!|Source \circ \underline{\mathsf{phago}^{\mu}_n}.\overline{\mathsf{exo}}^{\beta}_n|\tau'(\!|\mathsf{exo}^{\nu}_n|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1} \circ$
$\underline{\overline{\mathsf{phago}}^{\delta}_n(\rho)}|!\overline{\mathsf{phago}}^{\delta}_n(\rho)|\sigma_{target}(\!|\rho(\!|X \circ \tau'|\tau_X(\!|\diamond|\!)^{\Pi_1}|\!)^{\Pi_2}|\!)^{target}|\!)^{skin} \xrightarrow{l_2}$

$\sigma_Q(\!|Source \circ !\overline{\mathsf{phago}}^{\delta}_n(\rho)|\sigma_{target}(\!|\rho(\!|\underline{\overline{\mathsf{exo}}^{\beta}_n|\tau'}(\!|\underline{\mathsf{exo}^{\nu}_n}|\tau_X(\!|X|\!)^{\Gamma}|\!)^{\Pi_1}|\!)^{\Pi_2} \circ \rho(\!|X \circ \tau_X|\tau'(\!|\diamond|\!)^{\Pi_1}|\!)^{\Pi_2}|\!)^{target}|\!)^{skin} \xrightarrow{l_3}$

$\sigma_Q(\!|Source \circ !\overline{\mathsf{phago}}^{\delta}_n(\rho)|\sigma_{target}(\!|\rho(\!|X \circ \tau_X|\tau'(\!|\diamond|\!)^{\Pi_1}|\!)^{\Pi_2} \circ \rho(\!|X \circ \tau_X|\tau'(\!|\diamond|\!)^{\Pi_1}|\!)^{\Pi_2}|\!)^{target}|\!)^{skin}$

where

$l_1 = \mathsf{bud}_1(\Gamma, source, \gamma, \lambda)$ $\quad l_2 = \mathsf{phago}_1(\Pi_1, target, \mu, \delta)$ $\quad l_3 = \mathsf{exo}_1(\Gamma, \Pi_1, \nu, \beta)$

$\Pi_1 = \mathsf{bud}(\Gamma, source, \gamma, \lambda)$ $\quad \Pi_2 = \mathsf{phago}(\Pi_1, target, \mu, \delta)$

Table 5: Second scenario: encoding and a possible evolution where the prefixes involved in each transition are underlined.

We are interested in an analysis able to distinguish the behaviours of these different scenarios, both arising in biological settings. The first scenario is, e.g. the case for monomeric enzyme Rab-GTPases. Rab-GTPases are known as master regulators of intracellular membrane traffic [34]. They localise to different membrane compartments to control the specificity and directionality of membrane trafficking pathways, mostly related to vesicle transport. For instance, Rab-GTPases recruit effectors (other proteins) that promote vesicle formation, vesicle transport on microtubules, and vesicle fusion with target membranes. Hence, they allow one vesicle to enter into the target compartment/membrane. After the fusion, Rab-GTP hydrolyses the GTP to GDP and is released from the membrane. This can be understood as a one-to-one interaction between a vesicle and its corresponding target membrane.

The second scenario can be useful, for example, to model the case of exosomes. These are small membrane vesicles secreted by most cell types. Upon reaching their destinations, usually determined by the binding of specific ligands on their surfaces, exosomes can enter target cells either by being taken up by the target cell's endocytic pathway or by fusing to the target cell's membrane and releasing its contents directly into the cytoplasm. Normally, the target cell can receive more than one exosome at the time.

4. The Abstract Framework

Our analysis aims at computing an abstract description valid for of all the derivatives of the system under investigation. More specifically, we would like to derive information on the possible hierarchical structure of membranes, and on the processes that may be associated to each membrane, together with information about the possible number of occurrences of membrane and process labels. Following the *Abstract Interpretation* approach [2], the analysis result is computed by introducing an *abstract semantics* that abstractly mimics the execution of the system.

Although the final goal of analysis is computing information on derivatives of the system under investigation, it is convenient to approximate the paths of the system. This strategy allows us to more faithfully model the concrete behaviour and therefore to gain precision in our analysis, especially as far as occurrence counting information is concerned. This is the reason why the collecting semantics, is defined in terms of the concrete paths of the system (see Definition 2).

For the sake of clarity, the definition of the analysis is split into two parts.

- We first define the *abstract states* in order to approximate the systems. An abstract state reports information on the possible hierarchical structure of membranes, and on the processes that may be associated to each membrane, together with information about the possible number of occurrences of membrane and process labels. Moreover, we introduce abstract transitions that approximate the possible reactions between abstract states.

- Then we define the *configurations* in order to approximate the paths of a system. Configurations are obtained by enriching the abstract states with information on the set of reactions that have been already exercised to reach that abstract state. The set of reactions, together with occurrence counting information recorded by abstract states, is exploited to accurately determine the effect of the application of an abstract transition to the corresponding abstract state.

Finally, the abstract semantics is defined by the configuration which is computed by collecting all the configurations reachable from the initial configuration representing the system under investigation. We prove that our analysis is a *safe (over-)approximation* of the concrete semantics and that it can be effectively computed in polynomial time (see Theorems 5 and 6, respectively). The main result (see Theorem 5) relies on some auxiliary results showing that the abstract semantics safely approximates the paths of the system (see Theorems 2, 3 and 4), defined by the collecting semantics. As we have already pointed out, the correctness of the abstract semantics (and therefore of the analysis) requires the systems under investigation to be well labelled.

4.1. Abstract States and Abstract Transitions

We first introduce abstract membrane labels. Then, we present the abstract states and the corresponding abstract transitions. Moreover we show the properties and the related results.

Abstract Membrane Labels. To guarantee that the analysis can be computed in a finite number of steps, we need an *abstraction* of *membrane labels*. In the abstract setting, the basic membrane labels are defined as $\mathtt{Lab}^\circ_{\mathcal{M}} = \mathtt{Lab}_{\mathcal{M}} \cup \{@\}$, where the special symbol @ represents the outermost membrane. We can derive the corresponding set of *abstract membrane labels* as follows.

Definition 3. *The set of* abstract membrane labels $\widehat{\mathtt{Lab}}^\circ_{\mathcal{M}}$*, ranged over by* Γ°*,* Δ°*, ..., is defined as the least set such that: (i)* $\mathtt{Lab}^\circ_{\mathcal{M}} \subseteq \widehat{\mathtt{Lab}}^\circ_{\mathcal{M}}$*; and (ii) if* $\Gamma^\circ, \Delta^\circ \in \widehat{\mathtt{Lab}}^\circ_{\mathcal{M}}$ *and* $\lambda, \mu \in \mathtt{Lab}_{\mathcal{P}}$ *then* $(\Gamma^\circ, \Delta^\circ, \lambda, \mu) \in \widehat{\mathtt{Lab}}^\circ_{\mathcal{M}}$ *and* $(\Gamma^\circ, \lambda) \in \widehat{\mathtt{Lab}}^\circ_{\mathcal{M}}$.

Note that in the previously introduced abstraction of membrane labels, arbitrarily nested membrane labels can still arise (e.g. $(\Gamma^\circ, (\Delta^\circ, \Theta^\circ, \nu, \pi), \lambda, \mu)$). As a consequence, we introduce further approximations to guarantee that the abstract membrane labels generated in the analysis are finite. We first introduce an abstraction that depends on the chosen level of nesting depth d.

Definition 4. *Given* $d \in \mathbb{N}^+$ *the set of* abstract membrane labels parametric with respect to d *is defined as follows,*

$$\widehat{\mathtt{Lab}}^d_{\mathcal{M}} = \{\Delta^\circ | \Delta^\circ \in \widehat{\mathtt{Lab}}^\circ_{\mathcal{M}} \text{ and } depth(\Delta^\circ) \leq d\} \cup \quad \{(\top, \top, \lambda, \mu), (\top, \lambda) \mid \lambda, \mu \in \mathtt{Lab}_{\mathcal{P}}\}$$

where for $\Delta^\circ \in \widehat{\mathtt{Lab}}^\circ_{\mathcal{M}}$ *we have*

$$depth(\Delta^\circ) = \begin{cases} 1 & \text{if} \quad \Delta^\circ \in \mathtt{Lab}^\circ_{\mathcal{M}}, \\ 1 + max(depth(\Gamma^\circ), depth(\Psi^\circ)) & \text{if} \quad \Delta^\circ = (\Gamma^\circ, \Psi^\circ, \lambda, \mu), \\ 1 + depth(\Gamma^\circ) & \text{if} \quad \Delta^\circ = (\Gamma^\circ, \lambda) \end{cases}$$

Intuitively, all the abstract membrane labels with depth greater than d are approximated with the following new special membrane labels: $(\top, \top, \lambda, \mu)$ and $(\top, \lambda)$.

This is formalised by an *abstraction function* that maps a membrane label Δ into an abstract membrane label denoted by $\Delta^\bullet$ with respect to a given parameter d.

Definition 5. *Let* $d \in \mathbb{N}^+$ *and* $\Delta \in \widehat{\mathtt{Lab}}_{\mathcal{M}}$*. The* abstract version *of the membrane label* Δ *is denoted by* $\Delta^\bullet \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}$[3]*, and is inductively defined as follows:*

[3]For the sake of simplicity, we omit the explicit indication of the parameter d (assume fixed once for all).

1. $\Delta \in \mathtt{Lab}_{\mathcal{M}} \Rightarrow \Delta^{\bullet} = \Delta$;
2. $\Delta = \#(\Gamma, \Psi, \lambda, \mu)$ *with* $\# \in \{\mathtt{phago}, \mathtt{mate}, \mathtt{bud}\} \Rightarrow$

$$\Delta^{\bullet} = \begin{cases} (\Gamma^{\bullet}, \Psi^{\bullet}, \lambda, \mu) & \textit{if } depth((\Gamma^{\bullet}, \Psi^{\bullet}, \lambda, \mu)) \leq d \\ (\top, \top, \lambda, \mu) & \textit{otherwise} \end{cases}$$

3. $\Delta = \#(\Gamma, \lambda)$ *with* $\# \in \{\mathtt{pino}, \mathtt{drip}\} \Rightarrow \Delta^{\bullet} = \begin{cases} (\Gamma^{\bullet}, \lambda) & \textit{if } depth((\Gamma^{\bullet}, \lambda)) \leq d \\ (\top, \lambda) & \textit{otherwise} \end{cases}$

By summarising, Δ° denotes a generic abstract membrane label, while $\Delta^{\bullet}$ exactly denotes the abstract membrane label that is the abstract version of the membrane label Δ.

Abstract States. An *abstract state* carries information on the parent-child relation between membranes and a description of the processes possibly associated to each membrane. Furthermore, it records *occurrence counting*, i.e. information about the possible number of occurrences (or multiplicity) of membrane and process labels. The occurrence counting information approximates the *global* number of membrane and process labels that may appear in any system.

To describe the structure of systems, we adopt an *abstract representation*, formally given by a set of pairs, i.e. by a relation, that, for any abstract membrane label Δ°, records: (i) the abstract membrane labels that *may* be child membranes of Δ°; and (ii) the *sequential processes* that *may* be associated to membrane Δ°.

Definition 6 (Abstract Representation)**.** *An* abstract representation R° *is a relation* $R^{\circ} \subseteq \widehat{\mathtt{Lab}}^{d}_{\mathcal{M}} \times (\widehat{\mathtt{Lab}}^{d}_{\mathcal{M}} \cup S\,\mathtt{Proc})$*, where* $S\,\mathtt{Proc} = \{a^{\lambda}.\sigma \mid a^{\lambda}.\sigma \in \mathtt{Proc}\}$ *denotes the subset of sequential processes. We use* $\mathcal{R}^{\circ}$ *to denote the set of abstract representations.*

Given an abstract representation R°, $(\Delta^{\circ}, \Gamma^{\circ}) \in R^{\circ}$ says that the abstract membrane Γ° *may* be a child membrane of the membrane Δ°. As a consequence R° gives information on the possible membrane hierarchy. Similarly, $(\Delta^{\circ}, a^{\lambda}.\sigma) \in R^{\circ}$ says that the sequential process $a^{\lambda}.\sigma$ *may* be associated with the membrane Δ°.

To describe *occurrence counting* information, we choose a numerical domain able to distinguish between 0, 1 or more than 1 occurrences of an object. It is worth noting that this domain can be easily extended for distinguishing between 0, 1, 2, ..., k or more than k occurrences of an object, for any k.

For this reasons, we adopt the set $\mathtt{Mul} = \{1, \omega\}$ where each $x \in \mathtt{Mul}$ denotes a *multiplicity* with the expected interpretation: 1 indicates at most one occurrence, while ω indicates any number of occurrences. The set of multiplicities $\mathtt{Mul}$ comes equipped with the standard order $1 \leq \omega$ and with the addition operator $+^{\circ}$, presented in Table 6.

Definition 7 (Occurrence Counting)**.** *An* occurrence counting function *is a partial function* $O^{\circ} : \widehat{\mathtt{Lab}}^{d}_{\mathcal{M}} \cup \mathtt{Lab}_{\mathcal{P}} \rightarrow \mathtt{Mul}$*. We use* $\mathcal{O}^{\circ}$ *for the set of occurrence counting functions.*

By using a standard notation, an occurrence counting function O° can be alternatively represented by a set of pairs:

$$\{(\ell, x) \mid \ell \in dom(O^{\circ}) \wedge O^{\circ}(\ell) = x\}.$$

$+^\circ$	1	ω
1	ω	ω
ω	ω	ω

Table 6: The operator $+^\circ$ on multiplicities

Given an occurrence counting function O°, $(\Delta^\circ, 1) \in O^\circ$ says that the membrane label Δ° may appear no more than once in the abstract state.

We rely on some auxiliary operators on occurrence counting functions. First, we introduce the substitution operator $O^\circ[x/\ell]$ that, applied to an occurrence counting function $O^\circ \in \mathcal{O}^\circ$, returns the function where the multiplicity of $\ell \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}} \cup \mathtt{Lab}_{\mathcal{P}}$ is replaced by $x \in \mathtt{Mul}$.

Moreover, we define an operator $\cup^+$ that computes the addition of two functions using the operator $+^\circ$ (presented in Table 6). Given $O^\circ_1, O^\circ_2 \in \mathcal{O}^\circ$, the occurrence counting function $O^\circ_1 \cup^+ O^\circ_2$ is defined as follows, where $\ell \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}} \cup \mathtt{Lab}_{\mathcal{P}}$,

$$O^\circ_1 \cup^+ O^\circ_2(\ell) = \begin{cases} O^\circ_1(\ell) +^\circ O^\circ_2(\ell) & \text{if } \ell \in dom(O^\circ_1) \cap dom(O^\circ_2) \\ O^\circ_1(\ell) & \text{if } \ell \in dom(O^\circ_1), \ell \notin dom(O^\circ_2) \\ O^\circ_2(\ell) & \text{if } \ell \in dom(O^\circ_2), \ell \notin dom(O^\circ_1) \end{cases}$$

We now have all the ingredients to define our abstract states.

Definition 8 (Abstract State)**.** *An* abstract state *is a pair $S^\circ = (R^\circ, O^\circ)$, where $R^\circ \in \mathcal{R}^\circ$ is an abstract representation and $O^\circ \in \mathcal{O}^\circ$ is an occurrence counting function. We use $\mathcal{S}^\circ$ for the set of abstract states.*

Following the standard *Abstract Interpretation* style, our abstract states come equipped with an approximation order (denoted by $\sqsubseteq^\circ$) that allows us to compare approximations in terms of precision. Intuitively, $S^\circ_1 \sqsubseteq^\circ S^\circ_2$ says that the abstract state S°_1 is more precise than the abstract state S°_2 or, analogously that S°_2 *safely approximates* S°_1.

The definition of the approximation order on abstract states is defined component-wise, by using the corresponding orders on abstract representations and on occurrence counting functions.

Definition 9 (Approximation Orders)**.**

- *Given $O^\circ_1, O^\circ_2 \in \mathcal{O}^\circ$, we say that $O^\circ_1 \sqsubseteq_O O^\circ_2$ iff for each $\ell \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}} \cup \mathtt{Lab}_{\mathcal{P}}$ such that $\ell \in dom(O_1)$, we have $O_1(\ell) = x_1$ and $O_2(\ell) = x_2$ with $x_1 \leq x_2$.*
- *Given $S^\circ_1, S^\circ_2 \in \mathcal{S}^\circ$, we say that $S^\circ_1 \sqsubseteq^\circ S^\circ_2$ iff $S^\circ_1 = (R^\circ_1, O^\circ_1)$ and $S^\circ_2 = (R^\circ_2, O^\circ_2)$, $R^\circ_1 \subseteq R^\circ_2$ and $O^\circ_1 \sqsubseteq_O O^\circ_2$.*

Given the previous orders, the corresponding least upper bounds (l.u.b.) are defined as expected. The l.u.b. over occurrence counting functions and over abstract states are denoted by $\sqcup_O$ and $\sqcup^\circ$, respectively.

Abstract states are used to approximate systems. To formally relate systems and abstract states, we introduce a *translation function* t° that maps systems into abstract states. The function t° :

$\widehat{\mathtt{Lab}}^{d}_{\mathcal{M}} \times \mathtt{Sys} \to \mathcal{S}^{\circ}$, presented in Table 7, returns an abstract state, describing the system, with respect to an abstract membrane that represents the enclosing membrane. This function relies, in turn, on a corresponding translation function for processes $t^{\circ} : \widehat{\mathtt{Lab}}^{d}_{\mathcal{M}} \times \mathtt{Proc} \to \mathcal{S}^{\circ}$[4].

$$
\begin{array}{rcl}
t^{\circ}(\Delta^{\circ}, \diamond) & = & (\emptyset, \emptyset) \\
t^{\circ}(\Delta^{\circ}, P_1 \circ P_2) & = & (R^{\circ}_1 \cup R^{\circ}_2, O^{\circ}_1 \cup^{+} O^{\circ}_2) \text{ where } t^{\circ}(\Delta^{\circ}, P_i) = (R^{\circ}_i, O^{\circ}_i) \text{ with } i = 1, 2 \\
t^{\circ}(\Delta^{\circ}, !P) & = & (R^{\circ}, O^{\circ}[\omega/\ell]_{\ell \in dom(O^{\circ})})) \text{ where } t^{\circ}(\Delta^{\circ}, P) = (R^{\circ}, O^{\circ}) \\
t^{\circ}(\Delta^{\circ}, \sigma(\!|P|\!)^{\Gamma}) & = & \begin{cases} (R^{\circ}_1 \cup R^{\circ}_2 \cup \{(\Delta^{\circ}, \Gamma^{\bullet})\}, O^{\circ}_1 \cup^{+} O^{\circ}_2 \cup^{+} \{(\Gamma^{\bullet}, 1)\}) & \text{if } \sigma \not\equiv 0 \vee P \not\equiv \diamond \\ (\emptyset, \emptyset) & \text{otherwise} \end{cases} \\
 & & \text{where } t^{\circ}(\Gamma^{\bullet}, P) = (R^{\circ}_1, O^{\circ}_1) \text{ and } t^{\circ}(\Gamma^{\bullet}, \sigma) = (R^{\circ}_2, O^{\circ}_2) \\
t^{\circ}(\Delta^{\circ}, 0) & = & (\emptyset, \emptyset) \\
t^{\circ}(\Delta^{\circ}, \sigma_1 | \sigma_2) & = & (R^{\circ}_1 \cup R^{\circ}_2, O^{\circ}_1 \cup^{+} O^{\circ}_2) \text{ where } t^{\circ}(\Delta^{\circ}, \sigma_i) = (R^{\circ}_1, O^{\circ}_i) \text{ with } i = 1, 2 \\
t^{\circ}(\Delta^{\circ}, !\sigma) & = & (R^{\circ}, O^{\circ}[\omega/\ell]_{\ell \in dom(O^{\circ})}) \text{ where } t^{\circ}(\Delta^{\circ}, \sigma) = (R^{\circ}, O^{\circ}) \\
t^{\circ}(\Delta^{\circ}, a^{\lambda}.\sigma) & = & (\{(\Delta^{\circ}, a^{\lambda}.\sigma)\}, \{(\lambda, 1)\})
\end{array}
$$

Table 7: Translation function for systems and processes.

Based on this translation function, it is immediate deriving a corresponding *abstraction function* that, given a system, returns the abstract state that is its *best approximation*. Intuitively, the best approximation is the *most precise* (w.r.t. $\sqsubseteq^{\circ}$) abstract state that *safely represents* the information contained in the system.

Definition 10 (Abstraction function)**.** *We define* $\alpha_{\mathtt{Sys}} : \mathtt{Sys} \to \mathcal{S}^{\circ}$ *such that, given* $P \in \mathtt{Sys}$*,* $\alpha_{\mathtt{Sys}}(P) = (R^{\circ}, O^{\circ} \cup^{+} \{(@, 1)\})$*, where* $t^{\circ}(@, P) = (R^{\circ}, O^{\circ})$*.*

The best approximation of a system P is obtained by applying the translation function t° to P and to the abstract membrane label @ that represents the outermost membrane. The previously introduced notions can be used to capture the notion of safe approximation between abstract states and systems. Specifically, an abstract state S° *safely approximates* the system P if and only if $\alpha_{\mathtt{Sys}}(P) \sqsubseteq^{\circ} S^{\circ}$.

Moreover, the abstract state $\alpha_{\mathtt{Sys}}(P) \sqsubseteq^{\circ} S^{\circ}$ is the initial abstract state introduced for computing the abstract semantics (and thus the analysis) of the system P.

The next two lemmata describe a relevant property of the *translation function* and hence of the *abstraction function* with respect to the structural congruence. According to the first lemma, if two systems P and Q (two membrane processes σ and τ, resp.) are congruent, then their translations give rise to the same abstract state, up to congruence on the continuations of sequential processes.

[4]For the sake of simplicity, we use t° for both abstract systems and processes.

As a consequence, also the abstract states returned by the applications of abstraction function $\alpha_{\mathtt{Sys}}$ to P and Q coincide (Lemma 2). For the sake of simplicity, in the following we overload $=$ on abstract states to denote the syntactic equivalence of the states up to congruence on the continuations of sequential processes, e.g. $\{(\Delta^\circ, a^\lambda.\rho)\} = \{(\Delta^\circ, a^\lambda.\rho')\}$ if $\rho \equiv \rho'$.

Lemma 1 (Congruence 1)**.**

- *Let $\sigma, \tau \in \mathtt{Proc}$ be two membrane processes such that $\sigma \equiv \tau$. For any abstract membrane label $\Delta^\circ \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}$ we have that $t^\circ(\Delta^\circ, \sigma) = t^\circ(\Delta^\circ, \tau)$;*
- *Let $P, Q \in \mathtt{Sys}$ be two systems such that $P \equiv Q$. For any abstract membrane label $\Delta^\circ \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}$ we have that $t^\circ(\Delta^\circ, P) = t^\circ(\Delta^\circ, Q)$.*

Lemma 2 (Congruence 2)**.** *Let $P, Q \in \mathtt{Sys}$ be two systems. If $P \equiv Q$ then $\alpha_{\mathtt{Sys}}(P) = \alpha_{\mathtt{Sys}}(Q)$.*

Example 3. *Let us consider the system Q introduced in Ex. 1 (see Table 4), where we assume the level of maximal nesting depth $d = 3$. The initial abstract state, which is the best approximation of Q, is given by $\alpha_{\mathtt{Sys}}(Q) = S^\circ_0 = (R^\circ_0, O^\circ_0)$, where*

$$R^\circ_0 = \{(@, \mathit{skin}), (\mathit{skin}, \mathit{source}), (\mathit{skin}, \mathit{target}), (\mathit{skin}, \sigma_Q), (\mathit{source}, \Gamma), (\mathit{source}, \overline{\mathtt{bud}}^\lambda_n(\mathit{Vesicle})), (\mathit{target}, \overline{\mathtt{phago}}^\delta_n(\rho)), (\mathit{target}, \sigma_{\mathit{target}}), (\Gamma, X), (\Gamma, \mathtt{bud}^\gamma_n.\mathtt{exo}^\nu_n), (\Gamma, \tau_X)\}$$

$$O^\circ_0 = \{(@, 1), (\mathit{skin}, 1), (\mathit{source}, 1), (\mathit{target}, 1), (\Gamma, \omega), (X, \omega), (\lambda, \omega), (\gamma, \omega), (\delta, 1)\}$$

Alternatively, the component of R°_0 and O°_0 can be represented in tabular form to better emphasise the membrane hierarchy of the abstract state $\alpha_{\mathtt{Sys}}(Q)$, as in Table 8. For the sake clarity, in the rest of the paper we will use this representation.

membrane	*children*	*processes*
@	*skin*	
skin	*source*, *target*	σ_Q
source	Γ	$\overline{\mathtt{bud}}^\lambda_n(\mathit{Vesicle}), \tau''$
target		$\overline{\mathtt{phago}}^\delta_n(\rho), \sigma_{\mathit{target}}$
Γ	X	$\mathtt{bud}^\gamma_n.\mathtt{exo}^\nu_n, \tau_X$

membrane/process	multiplicity
@	1
skin, *source*, *target*	1
Γ, X	ω
λ, γ	ω
δ	1

Table 8: Abstract state $\alpha_{\mathtt{Sys}}(Q) = S^\circ_0 = (R^\circ_0, O^\circ_0)$.

More in detail, the abstract representation R°_0 and the occurrence counting function O°_0 are described by means of independent tables. The table on the left contains one row for each abstract membrane label Δ° in the domain of R°_0. For each Δ° the corresponding row reports in the second column, the set of abstract membrane labels that may *be children of Δ°, and in the third column the set of sequential processes that* may *be associated to membrane Δ°. Hence, the third line can*

be read as the membrane source may include the membrane Γ*, and it may have associated the process* $\overline{\mathtt{bud}}_n^{\lambda}(\mathit{Vesicle})$.

The table on the right reports the multiplicities for each abstract membrane and process label in the domain of O_0°*. For instance, the membrane labels skin, source and target have multiplicity* 1*, while the process labels* λ *and* γ *come with multiplicity* ω*. The corresponding prefixes occur indeed under the scope of a replication (see the rules in Table 7).*

The abstract state $\alpha_{\mathtt{Sys}}(Q') = S'^{\circ}_0 = (R'^{\circ}_0, O'^{\circ}_0)$ *computed for the system* Q' *(see Table 5) has the same abstract representation of* $\alpha_{\mathtt{Sys}}(Q)$*, i.e.* $R'^{\circ}_0 = R^{\circ}_0$ *(in particular the same membrane hierarchy), and also has the same occurrence counting information* O°_0 *except for the process label* δ *that has multiplicity* ω *(instead of* 1*).*

Finally, to formalise the relationship between the concrete and abstract domains, we introduce a pair of adjoint functions $\bar{\alpha}$ and $\bar{\gamma}$. The *abstraction function* computes the best approximation of a set of systems, by taking the l.u.b. of the best abstraction of each system contained in the set. The *concretisation function* $\bar{\gamma}$ records the set of systems safely approximated by an abstract state. The abstraction and concretisation functions form a *Galois connection* [2].

Definition 11. *We define the abstraction and concretisation functions* $\bar{\alpha} : \wp(\mathtt{Sys}) \to \mathcal{S}^\circ$ *and* $\bar{\gamma} : \mathcal{S}^\circ \to \wp(\mathtt{Sys})$ *functions as follows,*

1. *for* $X \in \wp(\mathtt{Sys})$*,* $\bar{\alpha}(X) = \bigsqcup^\circ_{P \in X} \alpha_{\mathtt{Sys}}(P)$*;*
2. *for* $S^\circ \in \mathcal{S}^\circ$*,* $\bar{\gamma}(S^\circ) = \bigcup_{\{P \mid \alpha_{\mathtt{Sys}}(P) \sqsubseteq^\circ S^\circ\}} P$*.*

Theorem 1 (Galois Connection)**.** *The pair of functions* $(\bar{\alpha}, \bar{\gamma})$ *in Definition 11 is a* Galois connection *between* $(\wp(\mathtt{Sys}), \subseteq)$ *and* $(\mathcal{S}^\circ, \sqsubseteq^\circ)$*.*

Abstract Transitions. The abstract semantics is given in terms of the abstract transition relation $\xrightarrow{l^\circ}_\circ$ among abstract states, where the *abstract transition label* $l^\circ \in \mathtt{Lab}^\circ_{\mathcal{T}}$ describes the reaction. The abstract transitions are obtained by introducing inference rules for abstract states that model the abstract counterpart of the membrane interactions possible in the concrete system: (Mate), (Bud) and (Drip) for MBD, and (Phago), (Exo) and (Pino) for PEP.

The set of *abstract transition labels* $\mathtt{Lab}^\circ_{\mathcal{T}}$[5] (ranged over by ${l_1}^\circ, {l_2}^\circ, \dots$) is defined as in the concrete case, by replacing membrane labels with abstract membrane labels. Thus, we have:

$$\begin{aligned}\mathtt{Lab}^\circ_{\mathcal{T}} \;=\; \{&\mathtt{phago}_1(\Gamma^\circ, \Delta^\circ, \lambda, \mu), \mathtt{exo}_1(\Gamma^\circ, \Delta^\circ, \lambda, \mu), \mathtt{pino}_1(\Delta^\circ, \lambda),\\ &\mathtt{mate}_1(\Gamma^\circ, \Delta^\circ, \lambda, \mu), \mathtt{bud}_1(\Gamma^\circ, \Delta^\circ, \lambda, \mu), \mathtt{drip}_1(\Delta^\circ, \lambda) \mid\\ &\Gamma^\circ, \Delta^\circ \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}, \lambda, \mu \in \mathtt{Lab}_{\mathcal{P}}\}\end{aligned}$$

Before presenting the abstract inference rules, we need to introduce some auxiliary operators. The first operator, given an abstract representation R° and an abstract membrane label Δ°, returns the set of abstract membrane labels that may be children of Δ°. The second operator,

[5]For simplicity, we omit the explicit indication of the parameter d when is clear from the context.

given an abstract representation R° and an abstract membrane label Δ°, returns the set of sequential processes that may be associated with the membrane Δ°. We define the two operators $children : \mathcal{R}^\circ \times \widehat{\mathtt{Lab}}^d_{\mathcal{M}} \to \wp(\widehat{\mathtt{Lab}}^d_{\mathcal{M}})$ and $processes : \mathcal{R}^\circ \times \widehat{\mathtt{Lab}}^d_{\mathcal{M}} \to \wp(S\,\mathtt{Proc})$ as follows,

$$children(R^\circ, \Delta^\circ) = \{\Gamma^\circ \mid (\Delta^\circ, \Gamma^\circ) \in R^\circ\}$$
$$processes(R^\circ, \Delta^\circ) = \{\sigma \mid (\Delta^\circ, \sigma) \in R^\circ\}$$

Finally, we introduce an operator $sub : \wp(S\,\mathtt{Proc}) \times \mathtt{Mul} \times S\,\mathtt{Proc} \to \wp(S\,\mathtt{Proc})$ that removes a sequential process $a^\lambda.\sigma$ from a set of sequential processes $C \subseteq S\,\mathtt{Proc}$, according to a multiplicity x as follows,

$$sub(C, x, a^\lambda.\sigma) = \begin{cases} C \setminus \{a^\lambda.\sigma\} & \text{if } x = 1 \\ C & \text{otherwise} \end{cases}$$

The abstract inference rules for the PEP rules are presented in Table 9, while the MBD rules are presented in Table 10. For the sake of simplicity, we comment here only on the abstract inference rules corresponding to the (Mate) and (Bud) interactions. The other abstract inference rules can be derived in similar way from their concrete versions.

The Rule (Mate°_c) models the fusion of two membranes (Δ° and Γ°) that *may* synchronise on actions $\mathtt{mate}^\lambda_n$ and $\overline{\mathtt{mate}}^\mu_n$. This requires that the following conditions hold:

- the abstract membranes Δ° and Γ° are given as possible siblings (having a common parent membrane Φ°), as stated by the premises $(\Phi^\circ, \Delta^\circ) \in R^\circ$ and $(\Phi^\circ, \Gamma^\circ) \in R^\circ$;
- there are two different membranes involved in the interaction, or at least two occurrences of the same membrane,
- the abstract representation R°, describing the processes associated to Δ° and Γ°, respectively, includes the actions mate and comate, as stated by the premises, since $(\Delta^\circ, \mathtt{mate}^\lambda_n.\sigma) \in R^\circ$, $(\Gamma^\circ, \overline{\mathtt{mate}}^\mu_n.\tau) \in R^\circ$; furthermore,
- the multiplicities $O^\circ(\lambda)$ and $O^\circ(\mu)$ related to the action labels are defined.

The resulting abstract state enriches the starting abstract state S° with information describing the effects of the possible fusion of the two membranes Δ° and Γ° both on the hierarchy and on the occurrence counting information of the resulting state. The membrane resulting from the possible fusion, is represented by the abstract membrane label Π°, obtained first using the transition label construct mate and then approximating the new label $l^\circ = \mathtt{mate}_1(\Delta^\circ, \Gamma^\circ, \lambda, \mu)$, according to its depth. Moreover, we add the following information on the resulting new abstract representation.

- The abstract membrane Π° is added as a possible child of the membrane Φ°, common parent of the two membranes Δ° and Γ°, as stated by the inclusion of (Φ°, Π°) in the new abstract representation.
- The membrane Π° inherits all the possible child membranes Θ of Δ° and Γ°, which thus become possible children of Π°, as stated by the inclusion of $(\Pi^\circ, \Theta^\circ)$ in the new abstract representation.

- The membrane Π° inherits all the membrane processes from Δ° and Γ°, apart from the process $\mathtt{mate}_n^\lambda.\sigma$ if λ has multiplicity equal to 1 and analogously for process $\overline{\mathtt{mate}}_n^\mu$. This models the fact that capabilities $\mathtt{mate}_n^\lambda$ and $\overline{\mathtt{mate}}_n^\mu$ may have been consumed. This is recorded by including the representation R_2° in the resulting abstract state.

- The new abstract representation also includes the sets R_3° and R_4°, obtained by the translation of the continuations $t^\circ(\Pi^\circ, \sigma)$ and $t^\circ(\Pi^\circ, \tau)$ of the mate and of the comate actions, respectively.

Finally, we have that the new occurrence counting function is obtained by adding the old occurring function to

- the function that records that the membrane Π° has multiplicity 1;

- the function O_3° obtained by the translation of the continuation of the mate action $\mathtt{mate}_n^\lambda$,

- the function O_4° obtained by the translation of the continuation of the comate action $\overline{\mathtt{mate}}_n^\mu$.

Similarly, rule (Bud$^\circ$) simulates the concrete (Bud) rule, by modelling the gemmation of a membrane Δ° from another membrane Γ° that may synchronise on actions $\overline{\mathtt{bud}}_n^\mu(\rho)$ and $\mathtt{bud}_n^\lambda$. This requires that: (i) the abstract membrane Δ° is given as a possible child of the membrane Γ° (i.e. $(\Gamma^\circ, \Delta^\circ) \in R^\circ$); (ii) there are at least two membranes involved in the interaction, (iii) according to the abstract representation R°, the actions cobud and bud may be associated to membranes Γ° and Δ°, respectively. Furthermore, the multiplicities of the process labels μ and λ associated to the actions must be defined.

The abstract transition label l° is derived, as in the concrete case, by combining the labels of the membranes and of the actions involved. The resulting abstract state is obtained by enriching the abstract state (R°, O°) with information reporting the effects of the possible movement of the membrane Δ° out from the membrane Γ°. This requires updating both the abstract representation and the occurrence counting function. Note that the membrane introduced by the bud reaction is described by the abstract membrane label Π°, obtained by approximating the membrane $(\Delta^\circ, \Gamma^\circ, \lambda, \mu)$ according to its depth.

The abstract representation is extended by introducing the abstract membrane Π° as a possible child of the membrane Φ° (in turn, parent of Γ°), and Δ° as a possible child of membrane Π°. Moreover, we introduce information on the membrane processes that may be associated to membranes Γ°, Δ° and Π°. In the case of membrane Π°, this requires adding R_2° obtained by applying the translation function to process ρ related to cobud. Similarly, in the case of the membranes Γ° and Δ° the related abstract representations R_3° and R_4° are obtained by applying the translation function to the continuations of the two coactions (σ and τ), respectively.

Finally, the occurrence counting function is updated by adding one occurrence of membrane Π° introduced by the bud reaction and the occurrence counting functions O_2°, O_3°, and O_4°, obtained by the translations of the process ρ and of the continuations of the coactions.

Note that these rules allow us to gain more precision with respect to the abstract rules in [33]. In particular, the constraint $(\Delta^\circ = \Gamma^\circ) \Rightarrow O^\circ(\Gamma^\circ) > 1$ prevents us to predict that a membrane interacts with itself when there is just one occurrence of that membrane. Furthermore, the continuations

of sequential processes do not automatically inherit the multiplicities of their prefixes. The right multiplicities are obtained by summing up the occurrence counting information obtained by each application of abstract transitions. As a consequence, the predictions are more accurate.

$$\text{(PHAGO}^\circ\text{)}\quad \frac{\begin{array}{c}(\Phi^\circ,\Delta^\circ)\in R^\circ,(\Phi^\circ,\Gamma^\circ)\in R^\circ,(\Delta^\circ=\Gamma^\circ)\Rightarrow O^\circ(\Gamma^\circ)>1\\ (\Delta^\circ,\mathtt{phago}_n^\lambda.\sigma)\in R^\circ,\ (\Gamma^\circ,\overline{\mathtt{phago}}_n^\mu(\rho).\tau)\in R^\circ\\ O^\circ(\lambda)=x,\ O^\circ(\mu)=y\end{array}}{\begin{array}{l}(R^\circ,O^\circ)\xrightarrow{l^\circ}_\circ (R^\circ\cup R_1^\circ\cup R_2^\circ\cup R_3^\circ\cup R_4^\circ, O^\circ\cup^+\{(\Pi^\circ,1)\}\cup^+ O_2^\circ\cup^+ O_3^\circ\cup^+ O_4^\circ)\\ \quad R_1^\circ=\{(\Gamma^\circ,\Pi^\circ),(\Pi^\circ,\Delta^\circ)\}\\ \quad t^\circ(\Pi^\circ,\rho)=(R_2^\circ,O_2^\circ), t^\circ(\Delta^\circ,\sigma)=(R_3^\circ,O_3^\circ), t^\circ(\Gamma^\circ,\tau)=(R_4^\circ,O_4^\circ)\\ \text{where}\quad l^\circ=\mathtt{phago}_1(\Delta^\circ,\Gamma^\circ,\lambda,\mu)\quad\text{and}\quad \Pi^\circ=\begin{cases}(\Delta^\circ,\Gamma^\circ,\lambda,\mu) & \text{if } (\Delta^\circ,\Gamma^\circ\lambda,\mu)\in\widehat{\mathtt{Lab}}^d_{\mathcal{M}},\\ (\top,\top,\lambda,\mu) & \text{otherwise}\end{cases}\end{array}}$$

$$\text{(Exo}^\circ\text{)}\quad \frac{\begin{array}{c}(\Phi^\circ,\Gamma^\circ)\in R^\circ,(\Gamma^\circ,\Delta^\circ)\in R^\circ,(\Delta^\circ=\Gamma^\circ)\Rightarrow O^\circ(\Gamma^\circ)>1\\ (\Gamma^\circ,\overline{\mathtt{exo}}_n^\mu.\tau)\in R^\circ,\ (\Delta^\circ,\mathtt{exo}_n^\lambda.\sigma)\in R^\circ,\\ O^\circ(\lambda)=x,\ O^\circ(\mu)=y\end{array}}{\begin{array}{l}(R^\circ,O^\circ)\xrightarrow{l^\circ}_\circ (R^\circ\cup R_1^\circ\cup R_2^\circ\cup R_3^\circ\cup R_4^\circ, O^\circ\cup^+ O_3^\circ\cup^+ O_4^\circ),\\ R_1^\circ=\{(\Phi^\circ,\Theta^\circ)\,|\,(\Delta^\circ,\Theta^\circ)\in R^\circ\},\ R_2^\circ=\{(\Gamma^\circ,\tau')\,|\,\tau'\in sub(C,x,\mathtt{exo}_n^\lambda.\sigma)\},\\ \quad C=\{\sigma'\mid(\Delta^\circ,\sigma')\in R^\circ\}\ t^\circ(\Gamma^\circ,\sigma)=(R_3^\circ,O_3^\circ), t^\circ(\Gamma^\circ,\tau)=(R_4^\circ,O_4^\circ)\\ \text{where}\quad l^\circ=\mathtt{exo}_1(\Delta^\circ,\Gamma^\circ,\lambda,\mu)\end{array}}$$

$$\text{(PINO}^\circ\text{)}\quad \frac{(\Delta^\circ,\mathtt{pino}^\lambda(\rho).\sigma)\in R^\circ,\ O^\circ(\lambda)=x}{\begin{array}{l}(R^\circ,O^\circ)\xrightarrow{l^\circ}_\circ (R^\circ\cup R_1^\circ\cup R_2^\circ\cup R_3^\circ, O^\circ\cup^+\{(\Pi^\circ,1)\}\cup^+ O_2^\circ\cup^+ O_3^\circ)\\ \quad R_1^\circ=\{(\Delta^\circ,\Pi^\circ)\},\ t^\circ(\Pi^\circ,\rho)=(R_2^\circ,O_2^\circ),\ t^\circ(\Delta^\circ,\sigma)=(R_3^\circ,O_3^\circ)\\ \text{where}\quad l^\circ=\mathtt{pino}_1(\Delta^\circ,\lambda)\quad\text{and}\quad \Pi^\circ=\begin{cases}(\Delta^\circ,\lambda)\text{ if }(\Delta^\circ,\lambda)\in\widehat{\mathtt{Lab}}^d_{\mathcal{M}},\\ (\top,\lambda),\ \text{otherwise}\end{cases}\end{array}}$$

Table 9: Abstract inference rules for (PHAGO), (Exo), and (PINO).

We prove that the abstract transitions (defined by the rules in Tables 9 and 10) *safely approximate* the concrete transitions (defined by the rules in Table 3). More specifically, we show that the abstract transitions that exit from an abstract state S°, which *safely approximates* a system P, *over-approximate* the transitions that exit from P. We recall that an abstract state S° *safely approximates* a system P, provided that $\alpha_{\mathtt{Sys}}(P)\sqsubseteq^\circ S^\circ$. Hence, if $\alpha_{\mathtt{Sys}}(P)\sqsubseteq^\circ S^\circ$ then for any transition of P there exists a corresponding abstract transition of S°.

Theorem 2. *Let $P_1\in\mathtt{Sys}$ be a system and let $S_1^\circ\in\mathcal{S}^\circ$ be an abstract state such that $\alpha_{\mathtt{Sys}}(P_1)\sqsubseteq^\circ S_1^\circ$. For any transition $P_1\xrightarrow{l}P_2$ there exists an abstract transition $S_1^\circ\xrightarrow{l^\bullet}_\circ S_2^\circ$ with $\alpha_{\mathtt{Sys}}(P_2)\sqsubseteq^\circ S_2^\circ$.*

4.2. The Abstract Semantics

As already mentioned, given a system P, our goal is to compute an *abstract state* that describes the possible topological structure of all the derivatives of P, together with occurrence counting

(Mate°)

$$\frac{(\Phi^\circ,\Delta^\circ)\in R^\circ,\ (\Phi^\circ,\Gamma^\circ)\in R^\circ,\ (\Delta^\circ=\Gamma^\circ)\Rightarrow O^\circ(\Gamma^\circ)>1 \quad (\Delta^\circ,\mathtt{mate}_n^\lambda.\sigma)\in R^\circ,\ (\Gamma^\circ,\overline{\mathtt{mate}}_n^\mu.\tau)\in R^\circ \quad O^\circ(\lambda)=x,\ O^\circ(\mu)=y}{(R^\circ,O^\circ)\xrightarrow{l^\circ}_\circ (R^\circ\cup R_1^\circ\cup R_2^\circ\cup R_3^\circ\cup R_4^\circ, O^\circ\cup^+\{(\Pi^\circ,1)\}\cup^+ O_3^\circ\cup^+ O_4^\circ)}$$

$$R_1^\circ=\{(\Phi^\circ,\Pi^\circ)\}\cup\{(\Pi^\circ,\Theta^\circ)\mid\Theta^\circ\in children(R^\circ,\Gamma^\circ)\cup children(R^\circ,\Delta^\circ)\},$$
$$R_2^\circ=\{(\Pi^\circ,\tau')\mid\tau'\in sub(C_1,x,\mathtt{mate}_n^\lambda.\sigma)\cup sub(C_2,y,\overline{\mathtt{mate}}_n^\mu.\tau)$$
$$C_1=processes(R^\circ,\Delta^\circ), C_2=processes(R^\circ,\Gamma^\circ),$$
$$t^\circ(\Pi^\circ,\sigma)=(R_3^\circ,O_3^\circ), t^\circ(\Pi^\circ,\tau)=(R_4^\circ,O_4^\circ)$$

where $l^\circ=\mathtt{mate}_1(\Delta^\circ,\Gamma^\circ,\lambda,\mu)$ and $\Pi^\circ=\begin{cases}(\Delta^\circ,\Gamma^\circ,\lambda,\mu)\text{ if }(\Delta^\circ,\Gamma^\circ\lambda,\mu)\in\widehat{\mathtt{Lab}}^d_{\mathcal{M}},\\(\top,\top,\lambda,\mu)\text{ otherwise}\end{cases}$

(Bud°)

$$\frac{(\Phi^\circ,\Gamma^\circ)\in R^\circ,(\Gamma^\circ,\Delta^\circ)\in R^\circ,(\Delta^\circ=\Gamma^\circ)\Rightarrow O^\circ(\Gamma^\circ)>1 \quad (\Gamma^\circ,\overline{\mathtt{bud}}_n^\mu(\rho).\tau)\in R^\circ,(\Delta^\circ,\mathtt{bud}_n^\lambda.\sigma)\in R^\circ \quad O^\circ(\lambda)=x,\ O^\circ(\mu)=y}{(R^\circ,O^\circ)\xrightarrow{l^\circ}_\circ (R^\circ\cup R_1^\circ\cup R_2^\circ\cup R_3^\circ\cup R_4^\circ, O^\circ\cup^+\{(\Pi^\circ,1)\}\cup^+ O_2^\circ\cup^+ O_3^\circ\cup^+ O_4^\circ)}$$

$$R_1^\circ=\{(\Phi^\circ,\Pi^\circ),(\Pi^\circ,\Delta^\circ)\},$$
$$t^\circ(\Pi^\circ,\rho)=(R_2^\circ,O_2^\circ), t^\circ(\Delta^\circ,\sigma)=(R_3^\circ,O_3^\circ), t^\circ(\Gamma^\circ,\tau)=(R_4^\circ,O_4^\circ)$$

where $l^\circ=\mathtt{bud}_1(\Delta^\circ,\Gamma^\circ,\lambda,\mu)$ and $\Pi^\circ=\begin{cases}(\Delta^\circ,\Gamma^\circ,\lambda,\mu) & \text{if }(\Delta^\circ,\Gamma^\circ\lambda,\mu)\in\widehat{\mathtt{Lab}}^d_{\mathcal{M}},\\(\top,\top,\lambda,\mu) & \text{otherwise}\end{cases}$

(Drip°)

$$\frac{(\Gamma^\circ,\Delta^\circ)\in R^\circ,\ (\Delta^\circ,\mathtt{drip}^\lambda(\rho).\sigma)\in R^\circ,\ O^\circ(\lambda)=x}{(R^\circ,O^\circ)\xrightarrow{l^\circ}_\circ (R^\circ\cup R_1^\circ\cup R_2^\circ\cup R_3^\circ, O^\circ\cup^+\{(\Pi^\circ,1)\}\cup^+ O_2^\circ\cup^+ O_3^\circ)}$$

$$R_1^\circ=\{(\Gamma^\circ,\Pi^\circ)\},\ t^\circ(\Pi^\circ,\rho)=(R_2^\circ,O_2^\circ), t^\circ(\Delta^\circ,\sigma)=(R_3^\circ,O_3^\circ)$$

where $l^\circ=\mathtt{drip}_1(\Delta^\circ,\lambda)$ and $\Pi^\circ=\begin{cases}(\Delta^\circ,\lambda)\text{ if }(\Delta^\circ,\lambda)\in\widehat{\mathtt{Lab}}^d_{\mathcal{M}},\\(\top,\lambda)\text{ otherwise}\end{cases}$

Table 10: Abstract inference rule for (Mate), (Bud) and (Drip).

information on membrane and process labels. A naive idea could be to compute this abstract state by collecting all the abstract states that can be reached from the initial one, $\alpha_{\mathtt{Sys}}(P)$, by applying the abstract inference rules presented in Tables 9 and 10. We recall that the initial abstract state is the best approximation of the system P to analyse. Following this approach would give a correct, but very coarse approximation especially of the counting information. The reason is that, in principle, any enabled reaction would be applied several times. As a consequence, infinite copies of the corresponding membranes and processes would be introduced, even though there are cases in which these copies do not correspond to the actual dynamic behaviour of the system.

For overcoming this problem, we need to add more information to our abstract description. This is obtained by approximating set of paths of the system rather than simply sets of reachable states. Indeed, from paths we can derive much more information.

To this aim, we introduce *configurations*, where abstract states are enriched with information

on the set of reactions that have been already exercised. The evolution of configurations is described by two meta-inference rules that model the effect of the application of a corresponding abstract transition to the abstract state. We exploit the information on the set of reactions that have been already exercised, together with the occurrence counting information reported in the abstract state for process labels to gain precision, in particular, in the approximation of the occurrence counting component.

Definition 12 (Configuration)**.** *A* configuration *is defined by $T^\circ \rhd S^\circ$, where $T^\circ \subseteq \mathtt{Lab}^\circ_{\mathcal{T}}$ is a set of abstract transition labels and $S^\circ \in \mathcal{S}^\circ$ is an abstract state. We use C° to denote the set of configurations.*

Intuitively, in a configuration $T^\circ \rhd S^\circ$, the component T° represents the labels of the reactions that have been already exercised to reach the abstract state S°. Hence, given a well labelled system P, we have that $\emptyset \rhd \alpha_{Sys}(P)$ is the *initial configuration* corresponding to the abstract state $\alpha_{Sys}(P)$.

To formally describe the information (and therefore the set of paths) represented by a configuration, we introduce an *abstraction framework* that relates paths and configurations.

First, we extend the approximation order on abstract states (given in the Definition 9) to configurations. The definition of the approximation order on configurations is defined component-wise, by relying on the corresponding orders on abstract states.

Definition 13 (Approximation Order on Configurations)**.** *Let $T^\circ_1 \rhd S^\circ_1, T^\circ_2 \rhd S^\circ_2 \in C^\circ$ be configurations. We say that $T^\circ_1 \rhd S^\circ_1 \sqsubseteq^{C^\circ} T^\circ_2 \rhd S^\circ_2$ iff $T^\circ_1 \subseteq T^\circ_2$ and $S^\circ_1 \sqsubseteq^\circ S^\circ_2$.*

Given the previous approximation order, we denote with $\sqcup^{C^\circ}$ the corresponding l.u.b. on configurations.

Moreover, we introduce an *abstraction function* α_{path} that associates a *configuration* to a path of a system P, which represents its *best approximation*. This allows us to express the notion of safe approximation between paths and configurations. Specifically, a configuration $T^\circ \rhd S^\circ \in C^\circ$ *safely approximates* a path p such that $p \in \mathcal{T}(P)$, provided that $\alpha_{path}(p) \sqsubseteq^{C^\circ} T^\circ \rhd S^\circ$.

Given a path p, we derive the corresponding *configuration* $T^\circ \rhd S^\circ$ where:

- T° represents the abstract transition labels of reactions that have been exercised in p;
- S° is an *abstract state* that does not only safely approximate all the systems appearing in the path, but it also provides occurrence counting information on the number of times the processes of the systems participate to reactions in the path p. In particular, for each process label, S° reports a multiplicity which approximates (i) *the number of occurrences of the same process that can appear simultaneously in any system of the path p* and (ii) *the number of occurrences of the process that are involved in the same reaction in the path p.*

The information described in items (i) and (ii) above reveals whether different occurrences of the processes involved in a reaction can be enabled more than once in the path, and therefore allows us to more faithfully model the concrete behaviour and to gain precision in our analysis.

The definition of abstraction function for paths relies on an auxiliary operator that counts the number of occurrences of a transition with label l in a given path.

Definition 14. *Let $P \in \mathtt{Sys}$ be a well labelled system. We define a function occ : $\mathcal{T}(P) \times \mathtt{Lab}_{\mathcal{T}} \to \mathtt{Mul}$ such that, given a path $p \in \mathcal{T}(P)$ and a transition label $l \in \mathtt{Lab}_{\mathcal{T}} e$:*

- *if $p = P \xrightarrow{l_1} P_1$ then $occ(p, l) = \begin{cases} 1 & \text{if } l = l_1 \\ 0 & \text{otherwise} \end{cases}$*
- *if $p = P \xrightarrow{l_1} P_1 \ldots P_{n-1} \xrightarrow{l_n} P_n$ then $occ(p, l) = occ(p_1, l) +^{\circ} x_n$*
  where $p_1 = P \xrightarrow{l_1} P_1 \ldots P_{n-2} \xrightarrow{l_{n-1}} P_{n-1}$ and $x_n = \begin{cases} 1 & \text{if } l = l_n \\ 0 & \text{otherwise} \end{cases}$

Intuitively, $occ(p, l)$ denotes the number of times (expressed by an abstract multiplicity) that a transition labelled l occurs in the path p: 0 if there is no transition, 1 if there is exactly one transition; and ω if the transitions are more than one.

Definition 15 (Abstraction of Paths)**.** *Let $P \in \mathtt{Sys}$ be a well labelled system. We define a function $\alpha_{path} : \mathcal{T}(P) \to C^{\circ}$ such that, given a path $p \in \mathcal{T}(P)$:*

- *if $p = P \xrightarrow{l_1} P_1$ then $\alpha_{path}(p) = \{l_1^{\bullet}\} \rhd \alpha_{\mathtt{Sys}}(P) \sqcup^{\circ} \alpha_{\mathtt{Sys}}(P_1)$;*
- *if $p = P \xrightarrow{l_1} P_1 \ldots P_{n-1} \xrightarrow{l_n} P_n$ where $p_1 = P \xrightarrow{l_1} P_1 \ldots P_{n-2} \xrightarrow{l_{n-1}} P_{n-1}$ and $\alpha_{path}(p_1) = \{l_1^{\bullet}, \ldots, l_{n-1}^{\bullet}\} \rhd S^{\circ}_{n-1}$ then*

$$\alpha_{path}(p) = \{l_1^{\bullet}, \ldots, l_n^{\bullet}\} \rhd S^{\circ}_{n-1} \sqcup^{\circ} \alpha_{Sys}(P_n) \sqcup^{\circ} (\emptyset, \{(\lambda^{\bullet}, occ(p, l_n)) \mid \lambda \in lab(l_n)\})$$

where $lab(l_n) \subseteq \mathtt{Lab}_{\mathcal{P}}$ stands for the set of process labels that occur in l_n.

The abstraction function is inductively defined. The configuration associated to a path contains the abstract version of all the transitions labels occurring in the path and a corresponding abstract state reporting, for each process label, an abstract multiplicity that approximates also the number of times a different occurrences of the such process is involved in the same transition of the path. To this aim, the abstract state approximates all the systems appearing in the path and is first obtained from the l.u.b (w.r.t. $\sqcup^{\circ}$) of the best abstraction of all the systems involved in the path (given by function $\alpha_{\mathtt{Sys}}$). Then the occurrence counting information related to process labels is updated (through a least upper bound operation) with a correct approximation of the number of times each process label was involved the same reaction of the path. This information is computed by using the *occ* operator. Since we assume the initial process to be well labelled, if a process label λ occurs more than once in a path (i.e. the process labelled λ is involved in a reaction appearing more than once), then it must be the case that more than one instances of such process is enabled, even though in each system of the path just one copy of the process labelled λ is present at the time.

To illustrate how the abstraction function works, we resort to the following example.

Example 4. *Let Q_1 be the following system*

$$Q_1 \quad = \quad (!\mathtt{drip}^{\lambda}(\rho).\mathtt{drip}^{\mu}(\sigma))(\!|\diamond|\!)^{\Delta}.$$

We consider a possible dynamic evolution of the system, described by the following path,

$$
\begin{array}{rll}
p_1 = & Q_1 \equiv (!\mathtt{drip}^{\lambda}(\rho).\mathtt{drip}^{\mu}(\sigma))|\mathtt{drip}^{\lambda}(\rho).\mathtt{drip}^{\mu}(\sigma)(\!|\diamond|\!)^{\Delta} & \xrightarrow{l_1} \\
& \rho(\!|\!|)^{\Pi_1} \circ \mathtt{drip}^{\mu}(\sigma)|(!\mathtt{drip}^{\lambda}(\rho).\mathtt{drip}^{\mu}(\sigma))(\!|\diamond|\!)^{\Delta} & \xrightarrow{l_2} \\
& \rho(\!|\!|)^{\Pi_1} \circ \sigma(\!|\!|)^{\Pi_2} \circ (!\mathtt{drip}^{\lambda}(\rho).\mathtt{drip}^{\mu}(\sigma))(\!|\diamond|\!)^{\Delta} \equiv & \\
& \rho(\!|\!|)^{\Pi_1} \circ \sigma(\!|\!|)^{\Pi_2} \circ (!\mathtt{drip}^{\lambda}(\rho).\mathtt{drip}^{\mu}(\sigma))|\mathtt{drip}^{\lambda}(\rho).\mathtt{drip}^{\mu}(\sigma)(\!|\diamond|\!)^{\Delta} & \xrightarrow{l_1} \\
& \rho(\!|\!|)^{\Pi_1} \circ \rho(\!|\!|)^{\Pi_1} \circ \sigma(\!|\!|)^{\Pi_2} \circ (!\mathtt{drip}^{\lambda}(\rho).\mathtt{drip}^{\mu}(\sigma))|\mathtt{drip}^{\mu}(\sigma)(\!|\diamond|\!)^{\Delta} &
\end{array}
$$

where

$$
\begin{array}{ll}
l_1 = \mathtt{drip}_1(\Delta, \lambda), & l_2 = \mathtt{drip}_1(\Delta, \mu), \\
\Pi_1 = \mathtt{drip}(\Delta, \lambda), & \Pi_2 = \mathtt{drip}(\Delta, \mu).
\end{array}
$$

The path describes the evolution in which the system (suitably unfolded by applying the structural congruence rule of replication of processes) performs the drip *reaction associated to the process label λ. This leads to the creation of a new empty membrane decorated by Π_1. Afterwards, the resulting system performs another* drip *reaction, this time associated to the label μ, that creates a further empty membrane with label Π_2. Again a* drip *reaction λ is possible (is at top-level, after a further unfolding) and, once fired, it produces an additional empty membrane Π_1.*

The configuration that models the best approximation of the previous path[6]*, is given by $\{l_1{}^{\bullet}, l_2{}^{\bullet}\}\triangleright (R_1^{\circ}, O_1^{\circ})$, described in Table 11, where $l_1{}^{\bullet} = l_1$, $l_2{}^{\bullet} = l_2$, $\Pi_1{}^{\bullet} = (\Delta, \lambda)$ and $\Pi_2{}^{\bullet} = (\Delta, \mu)$.*

membrane	*children*	*processes*
@	$\Delta, \Pi_1{}^{\bullet}, \Pi_2{}^{\bullet}$	
Δ		$\mathtt{drip}^{\lambda}(\rho).\mathtt{drip}^{\mu}(\sigma), \mathtt{drip}^{\mu}(\sigma)$
$\Pi_1{}^{\bullet}$		ρ
$\Pi_2{}^{\bullet}$		σ

membrane/process	multipl.
@	1
Δ	1
$\Pi_1{}^{\bullet}$	ω
$\Pi_2{}^{\bullet}$	1
λ	ω
μ	1

Table 11: Abstract state $S_1^{\circ} = (R_1^{\circ}, O_1^{\circ})$.

In this case the best approximation could be easily obtained by just collecting the abstraction of the process labels involved in the path p_1, together with the best approximation of the set of systems appearing in p_1. Here, indeed, the number of different occurrences of the same process that can be involved in the reactions of the path coincides with the maximal number of different occurrences of the same process that appears in parallel in the systems of the path. Hence, λ has multiplicity ω while μ has multiplicity 1*. Note that this is not always the case.*

Consider now the path p_1' obtained by adding to p_1 a further reaction corresponding to another drip *reaction with label μ, which creates an additional new empty membrane with label Π_2.*

$$p_1' = p_1 \xrightarrow{l_2} \rho(\!|\!|)^{\Pi_1} \circ \rho(\!|\!|)^{\Pi_1} \circ \sigma(\!|\!|)^{\Pi_2} \circ \sigma(\!|\!|)^{\Pi_2} \circ Q_1$$

[6]Also in this case we assume the level of maximal nesting depth as $d = 3$.

The best approximation of the path p'_1 is given by the configuration $\{l_1{}^\bullet, l_2{}^\bullet\} \rhd S^{\circ\prime}_1$, where $S^{\circ\prime}_1 = (R^\circ_1, O^\circ_1[\omega/\Pi_2{}^\bullet, \omega/\mu])$).

Note that the multiplicity ω assigned to membrane label $\Pi_2{}^\bullet$ derives from the abstraction of the last system of the path. By contrast, the multiplicity ω assigned to the process label μ derives from the information returned by the function occ applied to the corresponding transition label l_2. Actually, by considering the maximal number of processes decorated by μ, which may appear in parallel in any system of the path, we would obtain 1. Since $occ(p'_1, l_2) = \omega$, more than one reaction with label l_2 has been performed in the path and therefore the multiplicity of the corresponding process label μ is modified accordingly.

The example is intended to show the use of the *occ* operator in order to add to our description information also on the number of different occurrences of a process involved in the same reaction. Note that Definition 15 is necessary to guarantee the property of the Lemma 3, presented below.

We can also introduce an abstraction function for the set of paths of a given system, based on the one for paths.

Definition 16 (Abstraction of Sets of Paths)**.** *Let $P \in \mathtt{Sys}$ be a system. We define a function $\alpha_{coll} : \mathtt{Sys} \to C^\circ$ as follows,*

$$\alpha_{coll}(P) = \bigsqcup_{\{p \mid p \in \mathcal{T}(P)\}}^{C^\circ} \alpha_{path}(p).$$

Hence, given a well labelled system P, the configuration $\alpha_{coll}(P)$ records the most precise information that safely approximates the collecting semantics $\mathfrak{I}(P)$, given in Definition 2.

Example 5. *Consider again the system $Q_1 = (!\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma))(\!|\diamond|\!)^\Delta$ presented in Example 4, and consider a path different from those there shown.*

$$\begin{array}{lll}
p_2 = & Q_2 \equiv (!\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma))|\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma)(\!|\diamond|\!)^\Delta & \xrightarrow{l_1} \\
& \rho(\!|\!)^{\Pi_1} \circ \mathtt{drip}^\mu(\sigma)|(!\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma))(\!|\diamond|\!)^\Delta & \equiv \\
& \rho(\!|\!)^{\Pi_1} \circ \mathtt{drip}^\mu(\sigma)|(!\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma)|\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma))(\!|\diamond|\!)^\Delta & \xrightarrow{l_1} \\
& \rho(\!|\!)^{\Pi_1} \circ \rho(\!|\!)^{\Pi_1} \circ \mathtt{drip}^\mu(\sigma)|(!\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma))|\mathtt{drip}^\mu(\sigma)(\!|\diamond|\!)^\Delta & \xrightarrow{l_2} \\
& \rho(\!|\!)^{\Pi_1} \circ \rho(\!|\!)^{\Pi_1} \circ \sigma(\!|\!)^{\Pi_2} \circ (!\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma))|\mathtt{drip}^\mu(\sigma)(\!|\diamond|\!)^\Delta & \xrightarrow{l_2} \\
& \rho(\!|\!)^{\Pi_1} \circ \rho(\!|\!)^{\Pi_1} \circ \sigma(\!|\!)^{\Pi_2} \circ \sigma(\!|\!)^{\Pi_2} \circ (!\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma))(\!|\diamond|\!)^\Delta &
\end{array}$$

Although this path describes a different interleaving of drip actions with respect to the ones in the paths p_1 and p'_1, its best approximation coincides with the one of p'_1. The abstract path computed by our abstract semantics, which approximates both paths p_1 and p'_1, is shown in Example 8 of Section 4.4.

We can now introduce the technical machinery needed to define the evolution of configurations, by using the meta-inference rules in Table 12. Intuitively, given a configuration $T^\circ \rhd (R^\circ_1, O^\circ_1)$ and an abstract transition $(R^\circ_1, O^\circ_1) \xrightarrow{l^\circ}_\circ (R^\circ_2, O^\circ_2)$, we need to approximate the effect of the abstract reaction l°, when applied to the configuration $T^\circ \rhd (R^\circ_1, O^\circ_1)$. To this aim, we exploit the information

provided by the component T° which contains the abstract transition labels of the reactions which have been already exercised. Furthermore, we use the *multiplicity* of the abstract transition label l°, from the occurrence counting function O_1°, appearing in the abstract state (R_1°, O_1°).

Formally, the multiplicity of an abstract label in a given abstract state is computed by the function $mul : \mathcal{S}^\circ \times \mathtt{Lab}_{\mathcal{T}}^\circ \to \mathtt{Mul}$ defined as follows.

$$mul((R^\circ, O^\circ), l^\circ) = \begin{cases} \omega & \text{if } l^\circ \in \{\mathtt{pino}_1(\Delta^\circ, \lambda), \mathtt{drip}_1(\Delta^\circ, \lambda)\} \wedge O^\circ(\lambda) = \omega, \\ \omega & \text{if } l^\circ = a_1(\Gamma^\circ, \Theta^\circ, \lambda, \mu), a_1 \in \{\mathtt{phago}_1, \mathtt{exo}_1, \mathtt{mate}_1, \mathtt{bud}_1\} \wedge \\ & O^\circ(\lambda) = O^\circ(\mu) = \omega, \\ 1 & \text{otherwise} \end{cases}$$

Note that the multiplicity assigned to an abstract transition label in a given abstract state entirely depends on the multiplicity of the process labels associated to the actions that participate into the reaction in that state. In particular, if all the actions involved in the reaction have multiplicity ω, then so does the associated abstract transition label, otherwise it has multiplicity 1.

The following result shows the main property of the operator *mul* applied to the abstract state S°, appearing in a configuration $T^\circ \rhd S^\circ$, and to an abstract transition label $l^\bullet$. The abstract multiplicity computed by *mul* approximates the value of $occ(p, l)$, for each path p that is safely approximated by the configuration $T^\circ \rhd S^\circ$. This property holds provided that p is a path of a system P which is well labelled. The well labelling condition guarantees that processes with the same label correspond to different instances of the same process under the scope of a replication, in any derivative of P.

Lemma 3. *Let $P \in \mathtt{Sys}$ be a well labelled system. For any path $p \in \mathcal{T}(P)$ with $\alpha_{path}(p) \sqsubseteq^{C^\circ} T^\circ \rhd S^\circ$ and for any transition label $l \in \mathtt{Lab}_{\mathcal{T}}$, we have that*

$$occ(p, l) \leq mul(S^\circ, l^\bullet).$$

Hence there are two cases: (i) if $mul(S^\circ, l^\bullet) = \omega$ then in any path represented by the configuration $T^\circ \rhd S^\circ$ a transition related to $l^\bullet$ may have been realised *any number of times*; (ii) if $mul(S^\circ, l^\bullet) = 1$ in any path represented by the configuration $T^\circ \rhd S^\circ$, a transition related to $l^\bullet$ may have been realised *at most once*. This information can be exploited to predict how a transition labelled $l^\bullet$ has to be applied to the corresponding configuration.

We have now all the ingredients for explaining the two meta-inference inference rules in Table 12. The main difference between the two rules is that in the first rule the effect of the reaction $(R_1^\circ, O_1^\circ) \xrightarrow{l^\circ}_\circ (R_2^\circ, O_2^\circ)$ is propagated to all the components of the resulting configuration, while in the second one the occurrence counting component is not modified, the resulting configuration being (R_2°, O_1°).

- We can apply the rule (1) whenever there might exist a path p, represented by the configuration $T^\circ \rhd (R_1^\circ, O_1^\circ)$, that may execute the reaction corresponding to the transition label l. Thus, either the reaction associated to l° has never been applied before (condition $l^\circ \notin T^\circ$) or its multiplicity is ω (condition $mul((R_1^\circ, O_1^\circ), l^\circ) = \omega$). This implies that as far as we know, a transition related to l° might have been already performed *any number of times* in any path

$$\frac{(R_1^\circ, O_1^\circ) \xrightarrow{l^\circ}_\circ (R_2^\circ, O_2^\circ) \wedge (l^\circ \notin T^\circ \vee (l^\circ \in T^\circ \wedge mul((R_1^\circ, O_1^\circ), l^\circ) = \omega)}{T^\circ \rhd (R_1^\circ, O_1^\circ) \xrightarrow{l^\circ}_\rhd T^\circ \cup \{l^\circ\} \rhd (R_2^\circ, O_2^\circ)} \quad (1)$$

$$\frac{(R_1^\circ, O_1^\circ) \xrightarrow{l^\circ}_\circ (R_2^\circ, O_2^\circ) \wedge (l^\circ \in T^\circ \wedge mul((R_1^\circ, O_1^\circ), l^\circ) = 1)}{T^\circ \rhd (R_1^\circ, O_1^\circ) \xrightarrow{l^\circ}_\rhd T^\circ \rhd (R_2^\circ, O_1^\circ)} \quad (2)$$

Table 12: Meta-inference rules for configurations.

represented by $T^\circ \rhd (R_1^\circ, O_1^\circ)$. Due to the well labelling condition on systems under investigation, this reveals that the actions (or similarly the action) that participate(s) in the reaction associated to l° may appear under the scope of a replication. Hence, to be correct we have to report all the effects of the abstract transition on the resulting abstract configuration.

- We can apply the rule (2) whenever, in any path p represented by the configuration $T^\circ \rhd (R_1^\circ, O_1^\circ)$, a reaction corresponding to the transition label l might have been already performed *at most once*. Thus, in this case we have $l^\circ \in T^\circ$ and $mul((R_1^\circ, O_1^\circ), l^\circ) = 1$. Since, $mul((R_1^\circ, O_1^\circ), l^\circ) = 1$, by Lemma 3, the new reaction associated to l° can be performed by a path (always approximated by configuration $T^\circ \rhd (R_1^\circ, O_1^\circ)$) that has never performed such reaction. Therefore the effects on occurrence counting information of the new application of the reaction labelled l° do not need to be summed up. As a consequence, on the one hand the abstract representation R_1° has to be updated by taking into account the effect of the move. On the other hand, the occurrence counting information O_1° does not need to be modified, since the occurrence counting function O_1° already carries the correct multiplicities of the membrane and process labels involved in the reaction with respect to all the paths abstracted by $T^\circ \rhd (R_1^\circ, O_1^\circ)$.

Example 6. *To illustrate the application of meta-inference rule* (1) *in Table 12, consider the abstract state* $\alpha_{\mathsf{Sys}}(Q) = S_0^\circ = (R_0^\circ, O_0^\circ)$ *of Ex. 3 (see Table 8) that describes the best approximation of the system Q, presented in Ex. 1. Note that we can apply the abstract rule* (Bud°) *to* $\alpha_{\mathsf{Sys}}(Q)$*, because its premises are fulfilled:*

- $(skin, source), (source, \Gamma) \in R_0^\circ$*, with* $source \neq \Gamma$*, and*
- $(\Gamma, \mathsf{bud}_n^\gamma), (source, \overline{\mathsf{bud}}_n^\lambda(Vesicle)) \in R_0^\circ$*;*
- *furthermore,* $O_0^\circ(\lambda)$ *and* $O_0^\circ(\gamma)$ *are defined.*

As a consequence, we have

$$\alpha_{\mathsf{Sys}}(Q) \xrightarrow{l_1^\circ}_\circ S_1^\circ$$

where $l_1^\circ = \mathsf{bud}_1(\Gamma, source, \gamma, \lambda)$ *and the state* $S_1^\circ = (R_1^\circ, O_1^\circ)$ *is the one depicted in Table 13 (where the main differences, w.r.t. to the previous tables, are marked in blue in the pdf). Now, considering*

membrane	*children*	*processes*
@	*skin*	
skin	$source, target, \Pi_1^\circ$	σ_Q
source	Γ	$\overline{\mathtt{bud}}_n^\lambda(Vesicle), \tau''$
target		$\overline{\mathtt{phago}}_n^\delta(\rho), \sigma_{target}$
Γ	X	$\mathtt{bud}_n^\gamma.\mathtt{exo}_n^\nu$, $\mathtt{exo}_n^\nu, \tau_X$
$\Pi_1^\circ = (\Gamma, source, \gamma, \lambda)$	Γ	$\mathtt{phago}_n^\mu.\overline{\mathtt{exo}}_n^\beta, \tau'$

membrane/process	multipl.
@	1
skin, *source*, *target*	1
Γ, X	ω
λ, γ	ω
Π_1°	1
μ, δ, ν	1

Table 13: The abstract state $S_1^\circ = (R_1^\circ, O_1^\circ)$.

the configuration $\emptyset \rhd \alpha_{\mathtt{Sys}}(Q) = S_0^\circ$*, we can apply the meta-inference rule* (1) *in Table 12, since* $l_1^\circ \notin \emptyset$*, and obtain*

$$\emptyset \rhd \alpha_{\mathtt{Sys}}(Q) \xrightarrow{l_1^\circ}_\rhd \{l_1^\circ\} \rhd S_1^\circ$$

At this point, we could apply again the abstract rule (Bud°) *to the new state* S_1°*. It is worth noting that this is always possible because the new state is obtained from the previous one by adding new pairs in the hierarchy and summing up the occurrence counting information (i.e.* $S_0^\circ \sqsubseteq^\circ S_1^\circ$*). The premises of rule* (Bud°) *are still fulfilled in the new state* S_1°*: (i)* $(skin, source), (source, \Gamma) \in R_1^\circ$ *with* $source \neq skin$*, (ii)* $(\Gamma, \mathtt{bud}_n^\gamma), (source, \overline{\mathtt{bud}}_n^\lambda(Vesicle)) \in R_1^\circ$*. In this case, since* $O_1^\circ(\gamma) = \omega$ *and* $O_1^\circ(\lambda) = \omega$*, we have that* $mul((R_1^\circ, O_1^\circ), l_1^\circ) = \omega$*. Therefore, the meta-inference rule* (1) *in Table 12 allows us to apply again the reaction labelled* l_1° *and to accordingly update the occurrence counting information as follows,*

$$\emptyset \rhd \alpha_{\mathtt{Sys}}(Q) \xrightarrow{l_1^\circ}_\rhd \{l_1^\circ\} \rhd S_1^\circ \xrightarrow{l_1^\circ}_\rhd \{l_1^\circ\} \rhd S_2^\circ$$

where $S_2^\circ = (R_2^\circ, O_2^\circ)$ *is fully described in Table 14. In particular, in* R_2°*,* $\mathtt{exo}_n^\nu$ *is predicted as another process of* Γ *and* $O_2^\circ = O_1^\circ[\omega/\Pi_1^\circ, \omega/\mu, \omega/\nu]$*. Note that the main effect of the application of the rule* (Bud°) *to* S_1° *is the update of the occurrence counting information related to the membrane* Π_1° *and to the processes* μ *and* ν*.*

We can now present the definition of the *abstract semantics* of a system P, which is is computed by building the abstract paths of P on configurations. The abstract paths of a system P are formally defined as in the concrete case, by applying the meta-inference rules (1) and (2) in Table 12.

Definition 17 (Abstract Paths)**.** *Let* $P \in \mathtt{Sys}$ *be a system such that* $S^\circ = \alpha_{\mathtt{Sys}}(P)$*. The* abstract paths *of* P *are inductively defined as follows:*

1. *if* $p^\circ = \emptyset \rhd S^\circ \xrightarrow{l_1^\circ}_\rhd \{l_1^\circ\} \rhd S_1^\circ$ *is obtained by applying the meta-rules* (1) *and* (2) *in Table 12, then* p° *is an* abstract path *of* P*;*

2. *if* $\emptyset \rhd S^\circ \xrightarrow{l_1^\circ}_\rhd \{l_1^\circ\} \rhd S_1^\circ \ldots \{l_1^\circ, \ldots, l_{n-2}^\circ\} \rhd S_{n-2}^\circ \xrightarrow{l_{n-1}^\circ}_\rhd \{l_1^\circ, \ldots, l_{n-1}^\circ\} \rhd S_{n-1}^\circ$ *is an abstract path of* P *and* $\{l_1^\circ, \ldots, l_{n-1}^\circ\} \rhd S_{n-1}^\circ \xrightarrow{l_n^\circ}_\rhd \{l_1^\circ, \ldots, l_n^\circ\} \rhd S_n^\circ$ *is obtained by applying the meta-rules* (1) *and* (2) *in Table 12, then*

$$\emptyset \rhd S^\circ \xrightarrow{l_1^\circ}_\rhd \{l_1^\circ\} \rhd S_1^\circ \ldots \{l_1^\circ, \ldots, l_{n-1}^\circ\} \rhd S_{n-1}^\circ \xrightarrow{l_n^\circ}_\rhd \{l_1^\circ, \ldots, l_n^\circ\} \rhd S_n^\circ$$

is an abstract path *of* P.

We use $\mathcal{T}^\circ(P)$ *to denote the set of abstract paths of* P.

In the following, $\emptyset \rhd S^\circ \xrightarrow{\{l_1^\circ,\ldots,l_n^\circ\}^*}_\rhd \{l_1^\circ, ..., l_n^\circ\} \rhd S_n^\circ$ is used as a shorthand to denote an abstract path defined as follows

$$\emptyset \rhd S^\circ \xrightarrow{l_1^\circ}_\rhd \{l_1^\circ\} \rhd S_1^\circ \ldots \{l_1^\circ, \ldots, l_{n-1}^\circ\} \rhd S_{n-1}^\circ \xrightarrow{l_n^\circ}_\rhd \{l_1^\circ, \ldots, l_n^\circ\} \rhd S_n^\circ.$$

In the abstract case, the *semantics* of a system P is represented by a configuration that approximates the configurations occurring in the abstract paths of P. The abstract semantics of a system is derived by considering, for any abstract path, the configuration that is the l.u.b. (with respect to $\sqcup^{C^\circ}$) of the configurations occurring in the path.

Definition 18 (The Abstract Semantics)**.** *Let* $P \in \mathtt{Sys}$ *be a system. We define a function* $\mathfrak{I}^\circ : \mathtt{Sys} \to C^\circ$ *as*

$$\mathfrak{I}^\circ(P) = \sqcup^{C^\circ}_{\{p^\circ | p^\circ \in \mathcal{T}^\circ(p^\circ)\}} conf(p^\circ)$$

where for any abstract path $p^\circ \in \mathcal{T}^\circ(P)$ *we have* $conf(p^\circ) = \sqcup^{C^\circ}_{i\in\{0,\ldots,n\}} T_i^\circ \rhd S_i^\circ$, *assuming that* $S_0^\circ = \alpha_{\mathtt{Sys}}(P)$, $T_0^\circ = \emptyset$ *and* $p^\circ = T_0^\circ \rhd S_0^\circ \xrightarrow{l_1^\circ}_\rhd T_1^\circ \rhd S_1^\circ \ldots T_{n-1}^\circ \rhd S_{n-1}^\circ \xrightarrow{l_n^\circ}_\rhd T_n^\circ \rhd S_n^\circ$.

The next theorems state the main properties of the abstract semantics. Note that the system P is assumed to be well labelled in order to guarantee the soundness of the meta-rules presented in Table 12. The soundness of the meta-inference rules is based on the property formalised by Lemma 3.

The following result relates the abstract paths of a given system P with its concrete paths. More precisely, the theorem proves that for any path there exists a corresponding abstract path that approximates it.

Theorem 3. *Let* $P \in \mathtt{Sys}$ *be a well labelled system. For any path* $p \in \mathcal{T}(P)$ *there exists an abstract path* $p^\circ \in \mathcal{T}^\circ(P)$ *such that* $p^\circ = \emptyset \rhd \alpha_{Sys}(P) \xrightarrow{T^{\circ\,*}}_\rhd T^\circ \rhd S^{\circ\prime}$ *and* $S^\circ \sqsubseteq S^{\circ\prime}$, *where* $\alpha_{path}(p) = T^\circ \rhd S^\circ$.

Based on the previous result, we can now prove that the abstract semantics of a system *safely approximates* its concrete behaviour, described by the corresponding collecting semantics. The theorem establishes that the configuration computed by the abstract semantics of a system P safely approximates all the paths of P.

Theorem 4 (Safety of the Abstract Semantics)**.** *Given a well labelled system* $P \in \mathtt{Sys}$, *we have that*

$$\alpha_{coll}(\mathfrak{I}(P)) \sqsubseteq^{C^\circ} \mathfrak{I}^\circ(P).$$

4.3. The Analysis

The analysis of a system is derived from the abstract semantics, by considering only the abstract state component of the configuration.

Definition 19 (The Analysis)**.** *Given $P \in \mathtt{Sys}$, we define a function $\mathcal{A}^\circ : \mathtt{Sys} \to \mathcal{S}^\circ$ as $\mathcal{A}^\circ(P) = S^\circ$ where $\mathfrak{I}^\circ(P) = T^\circ \rhd S^\circ$.*

The next theorem (derived from Theorem 3) shows that the abstract state $S^\circ = (R^\circ, O^\circ)$ computed by the analysis of a system P *safely approximates* its concrete behaviour. This means that each derivative P' of the initial system P is approximated by the abstract state S° computed by the analysis of P. Hence, for each system P': (i) the abstract representation R° includes the information of the structure of P'; and analogously (ii) the occurrence counting function O° assigns to each process and abstract membrane label a multiplicity that over-approximates the number of occurrences of the corresponding labels in P'.

The analysis is obtained by approximating the set of all the systems that can be reached in finite paths.

Theorem 5 (Safety)**.** *Given a well labelled system $P \in \mathtt{Sys}$, we have that $\bar{\alpha}(X) \sqsubseteq^\circ \mathcal{A}^\circ(P)$ where $X = \bigcup_{\{p \mid p \in \mathcal{T}(P)\}} reach(p)$ and for any path $p \in \mathcal{T}(P)$, $reach(p) = \bigcup_{i \in \{0,\ldots,n\}} \{P_i\}$ for $p = P_0 \xrightarrow{l_1} P_1 \ldots P_{n-1} \xrightarrow{l_n} P_n$.*

Despite the fact that the analysis involves a computation over a power domain, which seems to admit exponentially long increasing paths, the analysis can be efficiently computed in polynomial time. Actually, we can effectively compute the analysis with a fixed-point computation, starting from the initial configuration $\emptyset \rhd \alpha_{\mathtt{Sys}}(P)$ and building a single abstract path reaching a final configuration. A *final configuration* is a configuration that cannot further evolve according to the meta-inference rules (1) and (2) in Table 12. This allows us to obtain a polynomial bound.

Definition 20 (Final Configuration)**.** *Let $T^\circ \rhd S^\circ \in C^\circ$ be a configuration. We say that $T^\circ \rhd S^\circ$ is a* final configuration *iff for any $T^\circ \rhd S^\circ \xrightarrow{l^\circ}_\rhd T^\circ_1 \rhd S^\circ_1$ for $T^\circ_1 \rhd S^\circ_1 \in C^\circ$ we have that $T^\circ_1 = T^\circ$ and $S^\circ_1 = S^\circ$.*

Definition 21. *Let $P \in \mathtt{Sys}$ be a well labelled system and an abstract path $p^\circ \in \mathcal{T}^\circ(P)$. Given a configuration $T^\circ \rhd S^\circ \in C^\circ$, we say that $T^\circ \rhd S^\circ$ is a* final configuration *of p° iff $T^\circ \rhd S^\circ$ is a final configuration and $p^\circ = T^\circ_0 \rhd S^\circ_0 \xrightarrow{l^\circ_1}_\rhd \{l^\circ_1\} \rhd S^\circ_1 \ldots \{l^\circ_1, \ldots, l^\circ_{n-1}\} \rhd S^\circ_{n-1} \xrightarrow{l^\circ_n}_\rhd T^\circ \rhd S^\circ$.*

Theorem 6. *Let $P \in \mathtt{Sys}$ be a well labelled system and $p^\circ_1, p^\circ_2 \in \mathcal{T}^\circ(P)$ be two abstract paths. Given two configurations $T^\circ_1 \rhd S^\circ_1, T^\circ_2 \rhd S^\circ_2 \in C$, if $T^\circ_1 \rhd S^\circ_1$ a final configuration of p°_1 and $T^\circ_2 \rhd S^\circ_2$ is a final configuration of p°_2, then $S^\circ_1 = S^\circ_2$.*

The previous property allows us to calculate the analysis of a system without computing all the configurations that can be reached from the initial one. Indeed, $\mathcal{A}^\circ(P)$ can be computed by just building a single abstract path

$$p^\circ = T^\circ_0 \rhd S^\circ_0, \quad T^\circ_1 \rhd S^\circ_1, \quad \ldots, \quad T^\circ_m \rhd S^\circ_m$$

where (i) $T^\circ_0 = \emptyset$, $S^\circ_0 = \alpha_{\mathtt{Sys}}(P)$; (ii) $T^\circ_m \rhd S^\circ_m$ is a final configuration; and, (iii) for each $i \in [1, m]$ the corresponding configuration is obtained by applying the meta-inference rules (1) and (2) in Table 12 to the previous configuration $T^\circ_{i-1} \rhd S^\circ_{i-1}$. Note that the abstract path p° is an ascending chain because, for each $i \in [0, m-1]$, we have that $T^\circ_i \rhd S^\circ_i \sqsubseteq^{C^\circ} T^\circ_{i+1} \rhd S^\circ_{i+1}$. More specifically, for each $i \in [0, m-1]$, either $T^\circ_i \subset T^\circ_{i+1}$ and $S^\circ_i \sqsubseteq^\circ S^\circ_{i+1}$ hold or $T^\circ_i \subseteq T^\circ_{i+1}$ and $S^\circ_i \sqsubset^\circ S^\circ_{i+1}$ hold. Hence, we have that the analysis of system P precisely coincides with the final state, i.e. $\mathcal{A}^\circ(P) = S^\circ_m$.

Corollary 1. *Let $P \in \mathtt{Sys}$ be a well labelled system. We have that $\mathcal{A}^\circ(P) = S^\circ$, where $T^\circ \rhd S^\circ \in C$ is a final configuration of an abstract path p° such that $p^\circ \in \mathcal{T}^\circ(P)$.*

The above result guarantees that the analysis can be computed in *polynomial time*, by observing that the number of abstract membranes and transition labels arising in the computation of the analysis is polynomial, when fixing the maximum depth d to a constant value, as detailed in the next paragraph.

Complexity of our approach. As a measure of complexity of our analysis, we consider the maximal number of different configurations that can be generated in a path starting from the initial configuration $\emptyset \rhd \alpha_{\mathtt{Sys}}(P)$ and reaching a final configuration, where P is a well labelled system. Let m be the number of process labels and n be the number of different membrane (labels) occurring in P, respectively.

First, we compute an upper bound (which depends on the chosen depth level d) to the maximal number of different abstract membrane labels that the analysis can introduce. Our abstract labels can be: (i) the quadruples belonging to $\widehat{\mathtt{Lab}}^\circ_{\mathcal{M}} \times \widehat{\mathtt{Lab}}^\circ_{\mathcal{M}} \times \mathtt{Lab}_{\mathcal{P}} \times \mathtt{Lab}_{\mathcal{P}}$, with depth no greater than d, (ii) the pairs belonging to $\widehat{\mathtt{Lab}}^\circ_{\mathcal{M}} \times \mathtt{Lab}_{\mathcal{P}}$, with depth no greater than d, (iii) the special quadruples belonging to $\{\top\} \times \{\top\} \times \mathtt{Lab}_{\mathcal{P}} \times \mathtt{Lab}_{\mathcal{P}}$, and, finally, (iv) the special pairs belonging to $\{\top\} \times \mathtt{Lab}_{\mathcal{P}}$. Hence, for $d = 1$, the maximal number of abstract membrane labels is $\mathtt{n}^\circ_1 = n^2m^2 + nm + m^2 + m \approx O(n^2m^2)$. For $d = 2$, the maximal number of abstract membrane labels is $\mathtt{n}^\circ_2 = (\mathtt{n}^\circ_1)^2 m_2 + \mathtt{n}^\circ_1 m + m^2 + m \approx O(n^4m^6)$. Similarly, for $d = 3$, we have $\mathtt{n}^\circ_3 = (\mathtt{n}^\circ_2)^2 m_2 + \mathtt{n}^\circ_2 m + m^2 + m \approx O(n^8m^{14})$. A further generalisation allows us to derive that the maximal number of abstract membrane labels, for a given d, is $\mathtt{n}^\circ_d = O(n^{n_l(d)}m^{m_l(d)})$, with $n_l(d) = 2^d$ and $m_l(d) = \sum_{i=1}^{i \le d} 2^i$.

Then, we compute an upper bound to the maximal number of different abstract transition labels that can be: (i) $\mathtt{mate}_1$, $\mathtt{bud}_1$, $\mathtt{exo}_1$ and $\mathtt{phago}_1$ that give quadruples and (ii) $\mathtt{pino}_1$ and $\mathtt{drip}_1$ that give pairs. Hence, they are $4((\mathtt{n}^\circ_d)^2 m_2) + 2(\mathtt{n}^\circ_d m) \approx O(n^{n_l(d+1)}m^{m_l(d+1)})$. Therefore, the number of the different abstract transitions that can be generated for a given d is $\approx \mathtt{n}^\circ_{d+1}$.

Given d, the maximal number of generable different configurations is obtained by considering that each step can, in the worst case, add to the current configuration $T^\circ \rhd (R^\circ, O^\circ)$ just one element among: (i) a new abstract transition label in T°; (ii) a new pair in the abstract representation R°; (iii) a pair in the occurrence counting information O°. By separately adding each component, we obtain $(\mathtt{n}^\circ_{d+1} + \mathtt{n}^\circ_d(\mathtt{n}^\circ_d + m) + 2\mathtt{n}^\circ_d + m) \approx (\mathtt{n}^\circ_{d+1} + (\mathtt{n}^\circ_d m))$. Recall that $\mathtt{n}^\circ_d = O(n^{n_l(d)}m^{m_l(d)})$, with $n_l(d) = 2^d$ and $m_l(d) = \sum_{i=1}^{i \le d} 2^i$. This ensures us that our analysis is *polynomial* in the number of process and membrane labels occurring in the analysed system P.

4.4. Our Analysis at work

To illustrate our analysis, we now apply it to the systems presented in Section 3. In particular, we better detail the analysis of the system Q (in Table 4 of Example 1), whose first steps have

been introduced in Examples 3 and 6. Furthermore, we present the analysis for the system Q_1 commented in the Examples 4 and 5. Finally, we apply our analsyis to a small system to show how the occurrence counting information allows us to gain precision also on the prediction. In all these examples, we assume depth $d = 3$.

Example 7. *The analysis of the system Q is computed starting from the initial configuration $\emptyset \rhd \alpha_{\mathsf{Sys}}(Q)$, where the abstract state $\alpha_{\mathsf{Sys}}(Q) = S^\circ_0 = (R^\circ_0, O^\circ_0)$ is described in Table 8 (commented in Ex. 3). As we have pointed out, the analysis can be computed with a fixed-point computation by building a single maximal path that starts from the initial configuration and ends into a final configuration, i.e. a configuration that cannot further evolve. For simplicity, we focus on an evolution that mimics the concrete one described in Ex. 6, i.e. the following abstract path.*

$$\emptyset \rhd \alpha_{\mathsf{Sys}}(Q) \xrightarrow{l^\circ_1}_{\rhd} \{l^\circ_1\} \rhd S^\circ_1 \xrightarrow{l^\circ_1}_{\rhd} \{l^\circ_1\} \rhd S^\circ_2$$

where the abstract state S°_2 is now fully listed in Table 14.

membrane	*children*	*processes*
@	*skin*	
skin	*source*, *target*, Π°_1	σ_Q
source	Γ	$\overline{\mathtt{bud}}^{\lambda}_n(\mathit{Vesicle}), \tau''$
target		$\overline{\mathtt{phago}}^{\delta}_n(\rho), \sigma_{\mathit{target}}$
Γ	X	$\mathtt{bud}^{\gamma}_n.\mathtt{exo}^{\nu}_n$, $\mathtt{exo}^{\nu}_n, \tau_X$
$\Pi^\circ_1 = (\Gamma, \mathit{source}, \gamma, \lambda)$	Γ	$\mathtt{phago}^{\mu}_n.\overline{\mathtt{exo}}^{\beta}_n, \tau'$

membrane/process	multipl.
@	1
skin, *source*, *target*	1
Γ, X	ω
$\lambda, \gamma, \mu, \nu$	ω
Π°_1	ω
δ	1

Table 14: The abstract state $S^\circ_2 = (R^\circ_2, O^\circ_2)$.

At this point, as in the concrete evolution of Section 3, S°_2 may evolve to the abstract state $S^\circ_3 = (R^\circ_3, O^\circ_3)$, described in Table 15, by applying the Rule (PHAGO°)*, as follows.*

$$S^\circ_2 \xrightarrow{l^\circ_2}_{\circ} S^\circ_3$$

where $l^\circ_2 = \mathtt{phago}_1(\Pi^\circ_1, \mathit{target}, \delta, \mu)$. The rule can be applied because its premises are fulfilled:

- $(\mathit{skin}, \Pi^\circ_1) \in R^\circ_2$, $(\mathit{skin}, \mathit{target}) \in R^\circ_2$ *with* $\Pi^\circ_1 \neq \mathit{target}$,
- $(\Pi^\circ_1, \mathtt{phago}^{\mu}_n.\overline{\mathtt{exo}}^{\beta}_n) \in R^\circ_2$, $(\mathit{target}, \overline{\mathtt{phago}}^{\delta}_n(\rho)) \in R^\circ_2$,
- *while $O^\circ_2(\mu)$ and $O^\circ_2(\delta)$ are defined.*

Since $l^\circ_2 \notin \{l^\circ_1\}$ we can only apply the meta-inference rule (1) *in Table 12 to the configuration $\{l^\circ_1\} \rhd S^\circ_2$, and obtain the following evolution.*

$$\{l^\circ_1\} \rhd S^\circ_2 \xrightarrow{l^\circ_2}_{\rhd} \{l^\circ_1, l^\circ_2\} \rhd S^\circ_3$$

membrane	*children*	*processes*
@	*skin*	
skin	$source, target, \Pi_1^\circ$	σ_Q
source	Γ	$\overline{\mathtt{bud}}_n^\lambda(Vesicle), \tau''$
target	Π_2°	$\overline{\mathtt{phago}}_n^\delta(\rho), \sigma_{target}$
Γ	X	$\mathtt{bud}_n^\gamma.\mathtt{exo}_n^\nu$, $\mathtt{exo}_n^\nu, \tau_X$
$\Pi_1^\circ = (\Gamma, source, \gamma, \lambda)$	Γ	$\mathtt{phago}_n^\mu.\overline{\mathtt{exo}}_n^\beta$, $\overline{\mathtt{exo}}_n^\beta, \tau'$
$\Pi_2^\circ = (\Pi_1^\circ, target, \delta, \mu)$	Π_1°	ρ

membrane/process	multipl.
@	1
skin, *source*, *target*	1
Γ, X	ω
$\lambda, \gamma, \mu, \nu$	ω
Π_1°	ω
δ, β	1
Π_2°	1

Table 15: The abstract state $S_3^\circ = (R_3^\circ, O_3^\circ)$.

At this point, we cannot apply any longer the meta-inference rule (1) *with transition label* l_2°, *because* $O_2^\circ(\delta) = 1$ *and then* $mul(S_2^\circ, l_2^\circ) = 1$.

Now, since $l_3^\circ \notin \{l_1^\circ, l_2^\circ\}$, S_3° *may evolve to* $S_4^\circ = (R_4^\circ, O_4^\circ)$, *depicted in Table 16, by applying the Rule* (EXO°), *and thus allowing the content X of membrane* Γ *to exit from membrane* Π_1°, *as follows.*

$$S_3^\circ \xrightarrow{l_3^\circ}_\circ S_4^\circ$$

where $l_3^\circ = \mathtt{exo}_1(\Gamma, \Pi_1^\circ, \nu, \beta)$. *Once again, by applying the meta-inference rule* (1) *to the configuration* $\{l_1^\circ, l_2^\circ\} \rhd S_3^\circ$, *we obtain the folllowing evolution.*

$$\{l_1^\circ, l_2^\circ\} \rhd S_3^\circ \xrightarrow{l_3^\circ}_\rhd \{l_1^\circ, l_2^\circ, l_3^\circ\} \rhd S_4^\circ$$

The effect of this transition is that the content X of membrane Γ *may now become a child of membrane* Π_1, *which, in turn, resides inside the membrane target. Moreover, the multiplicity of membrane* Π_1, *i.e. of the membrane that models the binding of the vesicle with the membrane target, is at most* 1, *as expected. We have just reached the fix-point in the computation of the abstract semantics and we can conclude that* $\mathcal{A}^\circ(Q) = S_4^\circ = (R_4^\circ, O_4^\circ)$. *Indeed* $O^\circ(\beta) = 1$ *and then* $mul(S_4^\circ, l_3^\circ) = 1$, *we cannot apply any longer the meta-inference rule* (1) *with transition label* l_3°. *Furthermore, we can only apply meta-inference rule* (2), *whose application does not update the topological abstract description of the system.*

For clarity, the membrane hierarchy described by abstract representation R_4° *is shown in the tree in Figure 2, where the nodes represent the abstract membrane labels and the edges represent the parent-child relation. According to the analysis, the only way for X to be inside the membrane target is when it is enclosed by membrane* Π_2 *(in blue in the pdf). Moreover, membrane* Π_2 *may enclose the molecule X directly or through the inclusion of* Π_1. *The occurrence counting information expressed by* O° *guarantees that the number of occurrences of membranes* Π_2 *inside target is at most* 1, *thus reflecting the fact that in this scenario, membrane target allows just one vesicle to enter at a time.*

membrane	*children*	*processes*
@	*skin*	
skin	$source, target, \Pi_1^\circ$	σ_Q
source	Γ	$\overline{\mathtt{bud}}_n^\lambda(Vesicle), \tau''$
target	Π_2°	$\overline{\mathtt{phago}}_n^\delta(\rho), \sigma_{target}$
Γ	X	$\mathtt{bud}_n^\gamma.\mathtt{exo}_n^\nu$, $\mathtt{exo}_n^\nu, \tau_X$
$\Pi_1^\circ = (\Gamma, source, \gamma, \lambda)$	Γ	$\mathtt{phago}_n^\mu.\overline{\mathtt{exo}}_n^\beta$, $\overline{\mathtt{exo}}_n^\beta, \tau'$
$\Pi_2^\circ = (\Pi_1^\circ, target, \delta, \mu)$	Π_1°, X	ρ

membrane/process	multipl.
@	1
skin, *source*, *target*	1
Γ, X	ω
$\lambda, \gamma, \mu, \nu$	ω
Π_1°	ω
δ, β	1
Π_2°	1

Table 16: The abstract state $S_4^\circ = (R_4^\circ, O_4^\circ)$.

As a technical remark, note the importance of the meta-inference rules (1) *and* (2) *in Table 12 for the precision of our analysis. Without them, we would have repeatedly updated the occurrence counting information, while applying the abstract inference rules. As a consequence, we would have obtained that* $O^\circ(\Pi_2^\circ) = \omega$*, as in the final configuration of* Q' *(see Table 17 in the next example), thus losing the information necessary to capture the main difference in the behaviour between systems* Q *and* Q'*.*

In a similar way, the analysis of the system Q' *described in Table 5 of Ex. 2 is given by the abstract state* $S_4'^\circ = (R_4'^\circ, O_4'^\circ)$ *illustrated in Table 17, which is obtained starting from* $\alpha_{\mathsf{Sys}}(Q') = S_0'^\circ = (R_0'^\circ, O_0'^\circ)$*, where* $R_0'^\circ = R_0^\circ$ *and* $O_0'^\circ = O_0{}^\circ[\omega/\delta]$*.*

Let us now discuss the results of our analysis for the system Q'*. Note that the abstract representation* $R_4'^\circ$ *describes the same membrane hierarchy of system* Q *(as illustrated in Figure 2), while the obtained occurrence counting information* $O_4'^\circ$ *is different, in particular, the multiplicity of membrane* Π_2 *in this case amounts to* ω*. As a consequence, the analysis reveals that target may enclose* more than one *occurrence of* Π_2 *(and therefore of X) at the same time, while in Q, the membrane target may enclose* at most one *occurrence of* Π_2*, since the multiplicity is* 1

We can then conclude that our analysis, thanks to the occurrence counting information, allows us to observe that the two systems Q *and* Q' *exhibit a different dynamical behaviour.*

Note that, in general, information on the possible presence/absence of a component in a membrane could also be exploited when developing a biological model, to detect errors in the model specification.

It is worth noting that the information on occurrence counting is necessary for distinguishing the two quite different behaviours, which have been previously described. Indeed, analyses able to approximate the hierarchy of membranes and processes only, such as [6, 7, 3, 8, 9], would predict the same hierarchical structure for the two biological systems.

Example 8. *We now consider the system* Q_1 *presented in Example 4*

$$Q_1 \quad = \quad (\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma))(\!|\diamond|\!)^\Delta.$$

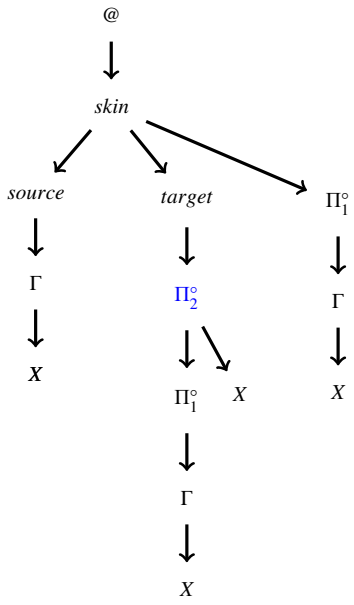

Figure 2: The membrane hierarchy trees described by both R°_4 and ${R'_4}^\circ$.

The analysis of the system Q_1 is computed starting from the initial configuration $\emptyset \rhd \alpha_{\mathtt{Sys}}(Q_1)$, where $\alpha_{\mathtt{Sys}}(Q_1) = S^\circ_{0,1} = (R^\circ_{0,1}, O^\circ_{0,1})$ is the abstract state described in Table 18.

In the configuration $\emptyset \rhd S^\circ_{0,1}$ we can only apply the meta-inference rule (1) to the abstract transition of $S^\circ_{0,1}$, which corresponds to the drip *reaction associated to λ. Therefore, we obtain the following evolution step*

$$\emptyset \rhd S^\circ_{0,1} \xrightarrow{l^\bullet_1}_\rhd \{l^\bullet_1\} \rhd S^\circ_{1,1}$$

where $S^\circ_{1,1}$ is the abstract state, described in Table 19, with $l^\bullet_1 = \mathtt{drip}_1(\Delta, \lambda)$ and $\Pi^\bullet_1 = (\Delta, \lambda)$.

The resulting configuration can now perform the abstract transition relative to the drip *reaction associated to μ, which simulates the concrete evolution of system Q_1 described in Ex. 4. By applying again the meta-inference rule (1) we then derive*

$$\{l^\bullet_1\} \rhd S^\circ_{1,1} \xrightarrow{l^\bullet_2}_\rhd \{l^\bullet_1, l^\bullet_2\} \rhd S^\circ_{2,1}$$

where $l^\bullet_2 = \mathtt{drip}_1(\Delta, \mu)$, $\Pi^\bullet_2 = (\Delta, \mu)$ and $S^\circ_{2,1}$ is the abstract state shown in Table 20.

It is worth noting that in the abstract state $S^\circ_{2,1}$ the process label λ has multiplicity ω because in the initial system Q_1 the corresponding process appears under a replication. By contrast, the process label μ has multiplicity 1 and consequently $mul(l^\bullet_2, S^\circ_{2,1}) = 1$. Since $l^\bullet_2 \in \{l^\bullet_1, l^\bullet_2\}$ and $mul(l^\bullet_2, S^\circ_{2,1}) = 1$, in the configuration $\{l^\bullet_1, l^\bullet_2\} \rhd S^\circ_{2,1}$ we cannot apply again the meta-inference rule (1) for performing another drip *reaction associated to μ. We can only apply the meta-inference rule (2), which does not modify the configuration as effect of the* drip *reaction associated to μ.*

However the configuration $\{l^\bullet_1, l^\bullet_2\} \rhd S^\circ_{2,1}$ can evolve into another configuration by exercising again a drip *reaction associated to λ. Since $mul(l^\bullet_1, S^\circ_{2,1}) = \omega$ we can apply again the meta-inference rule (1) and we derive*

membrane	*children*	*processes*
@	*skin*	
skin	$source, target, \Pi_1^\circ$	σ_Q
source	Γ	$\overline{\mathtt{bud}}_n^\lambda(Vesicle), \tau''$
target	Π_2°	$\overline{\mathtt{phago}}_n^\delta(\rho), \sigma_{target}$
Γ	X	$\mathtt{bud}_n^\gamma.\mathtt{exo}_n^\nu$, $\mathtt{exo}_n^\nu, \tau_X$
$\Pi_1^\circ = (\Gamma, source, \gamma, \lambda)$	Γ	$\mathtt{phago}_n^\mu.\overline{\mathtt{exo}}_n^\beta$, $\overline{\mathtt{exo}}_n^\beta, \tau'$
$\Pi_2^\circ = (\Pi_1^\circ, target, \delta, \mu)$	Π_1°, X	ρ

membrane/process	multipl.
@	1
skin, *source*, *target*	1
Γ, X	ω
$\lambda, \gamma, \delta, \nu$	ω
Π_1°	ω
μ, β	ω
Π_2°	ω

Table 17: The abstract state $S'^\circ_4 = (R'^\circ_4, O'^\circ_4)$.

membrane	*children*	*processes*
@	Δ	
Δ		$\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma)$

membrane/process	multipl.
@	1
Δ	1
λ	ω

Table 18: The abstract state $\alpha_{\mathsf{Sys}}(Q_1) = S^\circ_{0,1} = (R^\circ_{0,1}, O^\circ_{0,1})$.

$$\{l_1^\bullet, l_2^\bullet\} \rhd S^\circ_{2,1} \xrightarrow{l_1^\bullet}_\rhd \{l_1^\bullet, l_2^\bullet\} \rhd S^\circ_{3,1}$$

where $S^\circ_{3,1} = (R^\circ_{2,1}, O^\circ_{3,1})$ *and* $O^\circ_{3,1}$ *is the occurrence counting function shown in Table 21.*

Once executed the drip *reaction associated to* λ*, the multiplicity of the process label* μ *has been changed and therefore* $mul(l_2^\bullet, S^\circ_{3,1}) = \omega$*. As a consequence, we can now apply the meta-inference rule (1) to perform another* drip *reaction associated to* μ*, by deriving*

$$\{l_1^\bullet, l_2^\bullet\} \rhd S^\circ_{3,1} \xrightarrow{l_2^\bullet}_\rhd \{l_1^\bullet, l_2^\bullet\} \rhd S^\circ_{4,1}$$

where $S^\circ_{4,1} = (R^\circ_{2,1}, O^\circ_{4,1})$ *and* $O^\circ_{4,1}$ *is the occurrence counting function shown in Table 22.*

membrane	*children*	*processes*
@	$\Delta, \Pi_1{}^\bullet$	
Δ		$\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma), \mathtt{drip}^\mu(\sigma)$
$\Pi_1{}^\bullet$	ρ	

membrane/process	multipl.
@	1
Δ	1
$\Pi_1{}^\bullet$	1
λ	ω
μ	1

Table 19: The abstract state $S^\circ_{1,1} = (R^\circ_{1,1}, O^\circ_{1,1})$.

membrane	*children*	*processes*
@	$\Delta, \Pi_1^\bullet, \Pi_2^\bullet$	
Δ		$\mathtt{drip}^\lambda(\rho).\mathtt{drip}^\mu(\sigma), \mathtt{drip}^\mu(\sigma)$
$\Pi_1^\bullet$	ρ	
$\Pi_2^\bullet$	σ	

membrane/process	multipl.
@	1
Δ	1
$\Pi_1^\bullet$	1
$\Pi_2^\bullet$	1
λ	ω
μ	1

Table 20: The sbstract state $S^\circ_{2,1} = (R^\circ_{2,1}, O^\circ_{2,1})$.

membrane/process	multipl.
@	1
Δ	1
$\Pi_1^\bullet$	ω
$\Pi_2^\bullet$	1
λ	ω
μ	ω

Table 21: The occurrence counting function $O^\circ_{3,1}$.

It is worth noting that the configuration $\{l_1^\bullet, l_2^\bullet\} \rhd S^\circ_{4,1}$ with $S^\circ_{4,1} = (R^\circ_{2,1}, O^\circ_{4,1})$ is the final configuration that coincides with the best approximation of the paths p_1' and p_2 of system Q_1, illustrated in Examples 4 and 5, respectively.

membrane/process	multipl.
@	1
Δ	1
$\Pi_1^\bullet$	ω
$\Pi_2^\bullet$	ω
λ	ω
μ	ω

Table 22: The occurrence counting function O°_4.

Example 9. *To illustrate how the occurrence counting information allows us to gain precision also on the prediction of the possible evolution of the membrane hierarchy, with respect to analyses without occurrence counting, consider the following toy example. Suppose to have a very simple system Sys, described below, composed by a replicated membrane and another single membrane willing to fuse and then proceed with a phagocytosis.*

$$Sys = !\overline{\mathtt{mate}}_n^\lambda.\overline{\mathtt{phago}}_n^\mu(\rho)(\!|Q|\!)^{el_1} \circ \mathtt{mate}_n^\nu.\mathtt{phago}_n^\beta(\!|P|\!)^{el_2}$$

Phagocytosis cannot be performed because after the fusion the two corresponding co-actions reside in the same membrane, resulting from the fusion, and a single membrane cannot enter inside itself. Moreover no further membranes able to synchronise with the new membrane can be created. Even though the first membrane could fuse again with another membrane, by offering a comate action, because the replication operator precedes its prefixes. Nevertheless, after the fusion there are no other membranes able to provide the corresponding mate. As a consequence, no further membranes can be created.

The initial abstract state $\alpha_{\mathtt{Sys}}(S\,ys) = S\,ys^\circ_0 = (R^\circ_0, O^\circ_0)$ *is described in the upper part of Table 23 and we have the following evolution*

$$\emptyset \rhd S\,ys^\circ_0 \overset{l^\circ}{\rightarrow}_\rhd \{l^\circ\} \rhd S\,ys^\circ_1$$

where $l^\circ = \mathtt{mate}_1(el_2, el_1, \lambda, \nu)$ *and* $S\,ys^\circ_1$ *is described in the lower part of Table 23.*

membrane	*children*	*processes*
@	el_1, el_2	
el_1	Q	$\overline{\mathtt{mate}}^{\lambda}_n.\overline{\mathtt{phago}}^{\mu}_n(\rho)$
el_2	P	$\mathtt{mate}^{\nu}_n.\mathtt{phago}^{\beta}_n$

membrane/process	multipl.
@	1
el_1, el_2	1
ν	1
λ	ω

membrane	*children*	*processes*
@	el_1, el_2, Π°_1	
el_1	Q	$\overline{\mathtt{mate}}^{\lambda}_n.\overline{\mathtt{phago}}^{\mu}_n(\rho)$
el_2	P	$\mathtt{mate}^{\nu}_n.\mathtt{phago}^{\beta}_n$
$\Pi^\circ_1 = (el_1, el_2, \lambda, \nu)$	P, Q	$\overline{\mathtt{phago}}^{\mu}_n(\rho), \mathtt{phago}^{\beta}_n$

membrane/process	multipl.
@	1
el_1, el_2	1
ν, β, μ	1
λ	ω
Π°_1	1

Table 23: The abstract states $S\,ys^\circ_0 = (R^\circ_0, O^\circ_0)$ (above) and $S\,ys^\circ_1 = (R^\circ_1, O^\circ_1)$ (below).

Now, $(\mathtt{phago}^{\beta}_n, \Pi^\circ_1) \in R^\circ_1$ *and* $(\overline{\mathtt{phago}}^{\mu}_n(\rho), \Pi^\circ_1) \in R^\circ_1$*, but since* $O^\circ_1(\Pi^\circ_1) = 1$*, we cannot further apply the abstract rule (Phago*$^\circ$*). As a consequence, the system cannot further evolve: we have reached the fix-point.*

Hence, our analysis faithfully predicts, as expected, that (i) only a new membrane Π°_1 *can be generated by the fusion; (ii) no further membranes can be created in this system* $(O^\circ_1(\Pi^\circ_1) = 1)$*; and finally that (iii) the action* $\mathtt{phago}_n$ *and the corresponding co-action* $\overline{\mathtt{phago}}_n(\rho)$ *will never be executed.*

An analysis without the corrective effect of the occurrence counting information would have predicted an imprecise result: (i) more than one membrane can be generated by the mate fusion; (ii) the action $\mathtt{phago}_n$ *and the corresponding* $\overline{\mathtt{phago}}_n(\rho)$ *can then be executed; and consequently (iii) a new membrane* $\Pi^\circ_2 = (\Pi^\circ_1, \Pi^\circ_1, \beta, \mu)$ *resulting from such interaction should be created and added to the membrane hierarchy inside the membrane* Π°_1*.*

5. Case study: haemoglobin system

Among proteins that self-associate to form oligomers, haemoglobin is definitely one of the most significant representative. In mammalian, haemoglobin is the oxygen carrier of the blood, and has a tetrameric structure. Each subunit carries a macrocyclic molecule complexating a ferrous iron, the so-called haem group. One oxygen molecule (O2) can bind to each haem, thus the protein can bind up to four oxygen molecules, going from the so-called deoxy-haemoglobin to the oxy-haemoglobin form (see Figure 3). The four subunits are arranged into two structurally similar pairs, placed in a symmetric fashion to form the tetramer: one pair is of type α-globin and the other is of type β-globin. Many autosomal recessive blood disorders are associated with mutation of either *HBA* or *HBB* gene, e.g. sickle-cell disease origins from mutation of the *HBB* gene, expressing abnormal, non-functional β-globin, i.e. not able to bind oxygen. In the ideal case the overall structure is not affected by mutations and negative-cooperating effects are negligible, therefore haemoglobin binds to four oxygen molecules. Instead, a tetramer with functional α subunits and non-functional β subunits has an overall capability of binding oxygen limited to two molecules. Hence, in such case, being able to count the number of functional sites is strictly correlated to genetic mutation leading to abnormal proteins originating pathologies.

We are interested in the formation of the oxy-haemoglobin polymer starting from the deoxy-haemoglobin polymer that was formed by the binding of functional or non-functional haems. Note that mutations of gene *HBA* imply that both the α subunits are non-functional, while mutations of gene *HBB* imply that both the β subunits are non-functional. Since there exist disorders related to mutations of both genes *HBA* and *HBB*, we only distinguish between haems able to bind to oxygen molecules (healthy haems) and haems that are not able to bind to oxygen molecules (defective haems). As a consequence, a haemoglobin tetramer may bind four, two or even zero oxygen

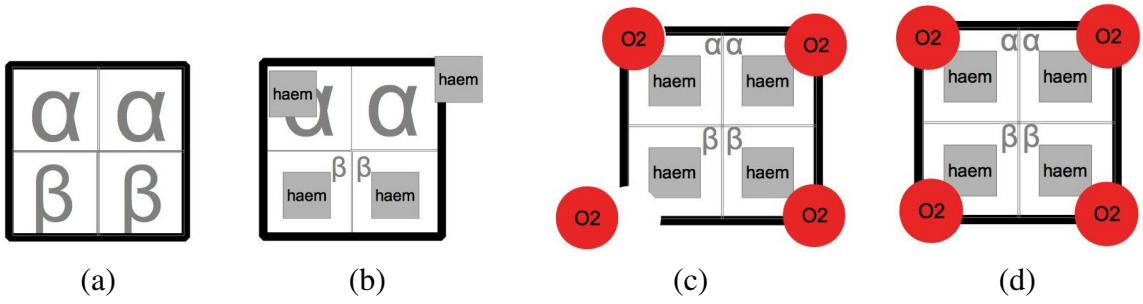

Figure 3: *The formation of the deoxy-haemoglobin: (a) and (b); and the formation of the 4-oxy-haemoglobin: (c) and (d).*

molecules, since the α subunits are both either functional or non-functional and the same holds for the β subunits.

For the sake of simplicity, in our brane model the haemoglobin is composed by just two haems, one representing the two α subunits and the other one representing the two β subunits. Consequently it can bind zero, one or two oxygen molecules. Note that it is straightforward to extend such model to the case of haemoglobin composed by four haems. The main system is as follows:

$$S \stackrel{def}{=} Co \circ !H_{aem} \circ !H'_{aem} \circ !O2,$$

$$
\begin{array}{lcl}
Co & \stackrel{def}{=} & \overline{\mathtt{mate}}_{\alpha}^{\lambda_1}.\overline{\mathtt{mate}}_{\beta}^{\lambda_2}.(\overline{\mathtt{exo}}_1^{\mu_1}|\overline{\mathtt{exo}}_2^{\mu_2})(\!|\overline{\mathtt{mate}}_1^{\lambda_7}(\!|)^{\Gamma_{C_1}} \circ \overline{\mathtt{mate}}_2^{\lambda_8}(\!|)^{\Gamma_{C_2}}|\!)^{\Gamma_C} \\
H'_{aem} & \stackrel{def}{=} & \mathtt{mate}_{\alpha}^{\lambda_5}(\!|)^{\Gamma_{\alpha'}} \circ \mathtt{mate}_{\beta}^{\lambda_6}(\!|)^{\Gamma_{\beta'}} \\
H_{aem} & \stackrel{def}{=} & \mathtt{mate}_{\alpha}^{\lambda_3}(\!|\mathtt{mate}_1^{\lambda_9}.\mathtt{exo}_1^{\mu_3}.\overline{\mathtt{phago}}_O^{\delta_1}(\rho_1)(\!|)^{\Gamma_{H_{11}}}|\!)^{\Gamma_\alpha} \circ \mathtt{mate}_{\beta}^{\lambda_4}(\!|\mathtt{mate}_2^{\lambda_{10}}.\mathtt{exo}_2^{\mu_4}.\overline{\mathtt{phago}}_O^{\delta_2}(\rho_2)(\!|)^{\Gamma_{H_{22}}}|\!)^{\Gamma_\beta} \\
O2 & \stackrel{def}{=} & \mathtt{phago}_O^{\delta_3}(\!|)^{\Gamma_O}
\end{array}
$$

Table 24: The haemoglobin model.

where the process Co works as a coordinator, by letting the haem for subunit α to bind the aim for subunit β.

A *healthy haem* is modelled by the subsystem H_{aem}, while a *defective haem* is modelled by H'_{aem}. Since both haems can be either healthy of defective, both subsystems occur under the scope of a replication. Finally, the oxygen molecules are modelled by the subsystem $O2$. Also in this case, the subsystem $O2$ occurs under the scope of a replication, since we assume to have an unbounded number of oxygen molecules. Table 24 provides the full specification of the system.

The coordinator process Co allows two mate operations to be executed with molecules H_{aem} or H'_{aem}, non-deterministically. We distinguish the different subunits α and β in the formation of haemoglobin polymer. Note that the process Co requires that both the haems for subunits α and subunits β are bound together before letting the haemoglobin bind oxygen molecules. In more detail, the healthy heam, after executing a mate operation with the coordinator, offers the suitable operations to let the haemoglobin polymer to exhibit the phago, which is the operation that allows the binding with the oxygen. Instead, the defective haem, after the fusion with the coordinator, does not offer any other operation. More technically, once the $mate_\alpha$ and $mate_\beta$ operations have been executed, the haemoglobin polymer is formed. At this point, the haemoglobin offers some (zero, one or two) *phago* operations, depending on the type of bound haems: healthy or defective. Finally, a molecule of oxygen can bind to each healthy haem.

We apply our analysis to investigate how the different kinds of haemoglobin oligomers that can be formed (binding healthy or defective haems) influence the subsequent bindings with the oxygen molecules.

By applying our approach, we obtain the final abstract state $S^\circ = (R^\circ, O^\circ)$, where the component R° is depicted in Table 25, while the component O° can be found in Table 26. To help the intuition, we use labels starting with Σ° for membranes that model the possible haemoglobin oligomers (the ones obtained by the binding with defective or healthy haems) and labels starting with Ω° for membranes that model the binding of the haemoglobin complex with the oxygen molecules. Figure 4 graphically depicts the hierarchical structure of the system, limited to the membranes that are interesting for our aim, mainly the hierarchies of the ones starting with Σ (in blue in the pdf) and Ω.

The abstract state S° describes the possible different ways to form the haemoglobin polymer. These different situations can be distinguished by inspecting the labels of the haemoglobin membranes (the ones starting with Σ), which carry information on the previous bindings with healthy or defective haems. For instance, the label $\Sigma_{1'2'} = \mathtt{mate}(\mathtt{mate}(\Gamma_C, \Gamma_{\alpha'}, \lambda_1, \lambda_5), \Gamma_{\beta'}, \lambda_2, \lambda_6)$ corresponds

to the membrane that represents the haemoglobin polymer obtained by the binding of two defective haems. Indeed, from its label we can deduce that it is obtained by the binding of membrane Γ_C (of the coordinator Co) to the defective haem for the α subunit $\Gamma_{\alpha'}$, and then by the following binding to the defective haem for the β subunit, $\Gamma_{\beta'}$. Note that this means that both subunits α and β are, in this case, non-functional.

The analysis is able to predict how these different kinds of haemoglobin oligomer, obtained by all the non deterministic possible evolutions of our system, bind to the oxygen molecules. In more detail, our system predicts the formation of four different kinds of deoxy-haemoglobin. From our analysis we can observe the behaviours of each kind of deoxy-haemoglobin oligomer. In more detail:

- Membrane $\Sigma_{1'2'}$, which describes the haemoglobin polymer obtained by the binding of *two defective haems*, does not contain any membrane whose label starts with Ω°. We recall that such membranes model the binding of the haemoglobin with the oxygen molecule. Therefore, we can conclude that *such complex will never bind any oxygen molecule.*

- Membrane $\Sigma_{12'}$, which describes the haemoglobin polymer obtained by the binding of *one healthy haem* (for subunit α) and *one defective haem* (for subunit β), contains one occurrence of membrane $\Omega^\circ_{O2'}$ (since multiplicity of $\Omega^\circ_{O2'}$ is equal to 1). Since membrane $\Omega^\circ_{O2'}$ models the binding with the oxygen molecule, we can conclude that *this complex can bind at most one oxygen molecule*.

- The same reasoning applies to membrane $\Sigma_{1'2}$, which describes the haemoglobin polymer obtained by the binding of *one defective haem* (for subunit α) and *one healthy haem* (for subunit β).

- Finally, membrane Σ_{12}, which describes the haemoglobin polymer obtained by the binding of *two healthy haems*, contains two membranes representing the binding with the oxygen molecule: membrane Ω°_{O2} and membrane Ω°_{1O}. Therefore, we can conclude that in this case *the haemoglobin polymer may bind with more oxygen molecules.* More precisely, since our abstraction on membrane labels distinguishes between labels Ω°_{O2} and Ω°_{1O}, we can conclude that, in this case, *the haemoglobin complex may bind with at most two membranes* (since the multiplicity of both Ω°_{O2} and Ω°_{1O} described in O° is 1).

Therefore our analysis correctly models the negative influence that, in the deoxy-haemoglobin polymer, defective haems have on the capability of such complex to bind to oxygen molecules. Note that while an analysis without occurrence counting could predict that the haemoglobin oligomer with two defective haems can never bind any oxygen molecules, in all the other (three) cases it has to predict that the haemoglobin oligomer could bind an unbounded number of oxygen molecules.

6. Conclusions

We presented an *Abstract Interpretation*-analysis technique to approximate the behaviour of biological systems described in Brane Calculi [1]. The analysis consists in two components. The first component, which over-approximates the possible membrane hierarchy in a control flow style,

membrane	*children*	*processes*
@	$\Gamma_C, \Gamma_\alpha, \Gamma_\beta, \Gamma_{\alpha'}, \Gamma_{\beta'}, \Gamma_O,$ $\Pi^\circ_1, \Pi^\circ_{1'}, \Sigma^\circ_{12}, \Sigma^\circ_{1'2'}, \Sigma^\circ_{12'}, \Sigma^\circ_{1'2}$	
Γ_C	$\Gamma_{C_1}, \Gamma_{C_2}$	$\lambda_1.\lambda_2.(\mu_1 \vert \mu_2)$
Γ_{C_1}		λ_7
Γ_{C_2}		λ_8
Γ_α	$\Gamma_{H_{11}}$	λ_3
$\Gamma_{H_{11}}$		$\lambda_9.\mu_3.\delta_1$
Γ_β	$\Gamma_{H_{22}}$	λ_4
$\Gamma_{H_{22}}$		$\lambda_{10}.\mu_4.\delta_2$
$\Gamma_{\alpha'}$		λ_5
$\Gamma_{\beta'}$		λ_6
Γ_O		δ_3
$\Pi^\circ_1 = \texttt{mate}(\Gamma_C, \Gamma_\alpha, \lambda_1, \lambda_3)$	$\Gamma_{C_1}, \Gamma_{C_2}, \Gamma_{H_{11}}, \Pi^\circ_{I1}$	$\lambda_2.(\mu_1 \vert \mu_2)$
$\Pi^\circ_{I1} = \texttt{mate}(\Gamma_{C_1}, \Gamma_{H_{11}}, \lambda_7, \lambda_9)$		$\mu_3.\delta_1$
$\Sigma^\circ_{12'} = \texttt{mate}(\Pi^\circ_1, \Gamma_{\beta'}, \lambda_2, \lambda_6)$	$\Gamma_{C_1}, \Gamma_{C_2}, \Gamma_{H_{11}},$ $\Pi^\circ_{I1}, \Omega^\circ_{O2'}$	μ_1, μ_2, δ_1
$\Sigma^\circ_{12} = \texttt{mate}(\Pi^\circ_1, \Gamma^\circ_\beta, \lambda_2, \lambda_4)$	$\Gamma_{C_1}, \Gamma_{C_2}, \Gamma_{H_{11}}, \Pi^\circ_{I1},$ $\Gamma_{H_{22}}, \Pi^\circ_{I2}, \Omega^\circ_{O2}, \Omega^\circ_{1O}$	$\mu_1, \mu_2, \delta_1, \delta_2$
$\Pi^\circ_{I2} = \texttt{mate}(\Gamma_{C_2}, \Gamma_{H_{22}}, \lambda_8, \lambda_{10})$		$\mu_4.\delta_2$
$\Pi^\circ_{1'} = \texttt{mate}(\Gamma_C, \Gamma_{\alpha'}, \lambda_1, \lambda_5)$	$\Gamma_{C_1}, \Gamma_{C_2}$	$\lambda_2.(\mu_1 \vert \mu_2)$
$\Sigma^\circ_{1'2} = \texttt{mate}(\Pi^\circ_{1'}, \Gamma_\beta, \lambda_2, \lambda_4)$	$\Gamma_{C_1}, \Gamma_{C_2}, \Gamma_{H_{22}},$ $\Pi^\circ_{I2}, \Omega^\circ_{1'O}$	μ_1, μ_2, δ_2
$\Sigma^\circ_{1'2'} = \texttt{mate}(\Pi^\circ_{1'}, \Gamma_{\beta'}, \lambda_2, \lambda_6)$	$\Gamma_{C_1}, \Gamma_{C_2}$	μ_1, μ_2
$\Omega^\circ_{O2'} = \texttt{phago}(\Gamma_O, \Sigma^\circ_{12'}, \delta_3, \delta_1)$	Γ_O	ρ_1
$\Omega^\circ_{O2} = \texttt{phago}(\Gamma_O, \Sigma^\circ_{12}, \delta_3, \delta_1)$	Γ_O	ρ_1
$\Omega^\circ_{1O} = \texttt{phago}(\Gamma_O, \Sigma^\circ_{12}, \delta_3, \delta_2)$	Γ_O	ρ_2
$\Omega^\circ_{1'O} = \texttt{phago}(\Gamma_O, \Sigma^\circ_{1'2}, \delta_3, \delta_2)$	Γ_O	ρ_2

Table 25: The abstract state $S^\circ = (R^\circ, O^\circ)$.

membrane	multiplicity
@	1
$\Gamma_C, \Gamma_{C_1}, \Gamma_{C_2}$	1
$\Gamma_\alpha, \Gamma_{H_{11}}, \Gamma_\beta, \Gamma_{H_{22}}$	ω
$\Gamma_{\alpha'}, \Gamma_{\beta'}, \Gamma_O$	ω
$\Pi^\circ_1, \Pi^\circ_{1'}$	1
$\Sigma^\circ_{12}, \Sigma^\circ_{12'}, \Sigma^\circ_{1'2}, \Sigma^\circ_{1'2'}$	1
$\Pi^\circ_{I1}, \Pi^\circ_{I2}$	1
$\Omega^\circ_{O2'}, \Omega^\circ_{O2}, \Omega^\circ_{1O}, \Omega^\circ_{1'O}$	1

process	multiplicity
$\lambda_1, \lambda_2, \lambda_7, \lambda_8$	1
$\lambda_3, \lambda_4, \lambda_9, \lambda_{10}, \lambda_5, \lambda_6$	ω
μ_1, μ_2	1
$\mu_3, \mu_4, \delta_1, \delta_2$	1
δ_3	ω

Table 26: The multiplicity of the abstract state $S^\circ = (R^\circ, O^\circ)$.

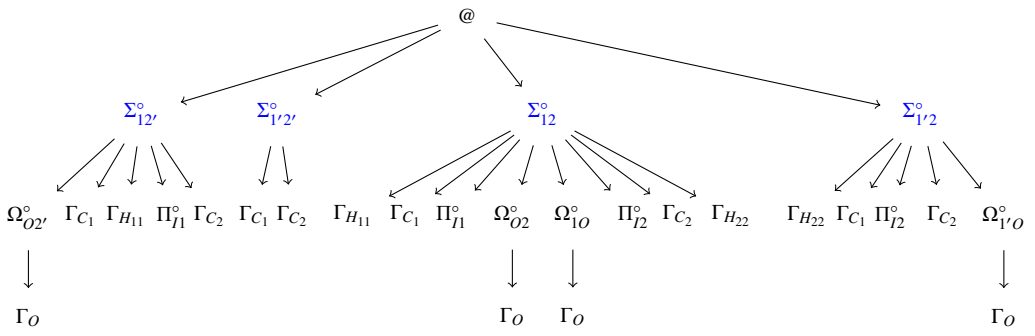

Figure 4: The membrane hierarchy tree described by R°, limited to the interesting membranes for our results.

is obtained by adapting static analysis techniques used for process algebras handling biological compartments (see e.g. [6, 7, 3, 9, 18]). The analysis is enhanced by a second and less standard component, providing global occurrence counting information, which is used to predict whether some component may occur at most once in any system reachable from the initial one. The two components influence and strengthen each other. In particular, the prediction on the possible membrane hierarchy is refined due to global occurrence counting information, thus allowing us to increase the precision with respect to the previous static approaches for Brane Calculi [3, 9, 18]. Example 9 of Section 4.4 is a clear witness of this feature.

It is worthwhile noting that the analyses that provide occurrence counting information proposed for BioAmbients [15, 16, 17, 14], the sibling bio-inspired calculus, cannot be straightforwardly adapted to Brane Calculi. We need indeed a careful labelling technique for membranes, because of bitonality, i.e. the fact that brane interactions possibly introduce new membranes, in order not to mix what is inside a membrane with what is outside (entities can be shuttled inside or outside, only if wrapped by another membrane).

To validate the applicability of our analysis in the biological setting, we applied it to several examples. The first example illustrates two different systems of communication via mobile vesicles that exhibit different dynamical behaviours with respect to number of occurrences of a substance X that may end up inside a target: an unbounded number in the second case, just one in the first case. Despite its simplicity, without the occurrence counting, the analysis failed to detect differences between the two cases, thus not predicting that, in the first case, at most one occurrence of molecule X can end up in the target, while, in the second case, an unbounded number can end up in the target.

Moreover, in Section 5 we present a small case study that investigates the formation of the deoxy-haemoglobin oligomer and the subsequent formation of the oxy-haemoglobin complex. Our analysis with occurrence counting information is able to show the negative influence that defective haems in the deoxy-haemoglobin oligomer have on the capability of such complex to bind oxygen molecules. Note that an analysis without occurrence counting information could not capture that different kinds of deoxy-haemoglobin oligomer may bind different number of oxygen molecules. Recall that, as we have already pointed out, our numerical domain could be easily extended in order to distinguish $0, 1, 2.., k$ or more than k occurrences, for any k.

As future work, we would like to further improve the accuracy of our approach. In particular, we would like to better analyse the systems that contain different instances of the same membrane or of the same process, and to better handle replication. One possible direction consists in refining our analysis by providing local occurrence counting information in the style of [15, 17]. In this way, we could apply our approach to more complex biological case studies, such as the one modelled in [35], for investigating the relationships occurring among events.

References

[1] L. Cardelli, Brane calculi, in: Proc. of Computational Methods in Systems Biology (CMSB'04), Vol. 3082 of Lecture Notes in Computer Science, Springer, 2005, pp. 257–280.

[2] P. Cousot, R. Cousot, Static determination of dynamic properties of generalized type unions, in: Language Design for Reliable Software, 1977, pp. 77–94.

[3] C. Bodei, L. Brodo, Brane calculi systems: A static preview of their possible behaviour, CoRR abs/1108.3429.

[4] D. S. Goodsell, A. J. Olson, Structural symmetry and protein function, Annual Review of Biophysics and Biomolecular Structure 29 (2000) 105–153.
[5] A. Vitale, G. Mauri, Communication via mobile vesicles in brane calculi, Electronic Notes in Theoretical Computer Science 171 (2) (2007) 187–196.
[6] F. Nielson, H. R. Nielson, C. Priami, D. S. da Rosa, Control flow analysis for bioambients, Electronic Notes in Theoretical Computer Science 180 (3) (2007) 65–79.
[7] C. Bodei, Control flow analysis for beta-binders with and without static compartments, Theor. Comput. Sci. 410 (33-34) (2009) 3110–3127.
[8] H. R. Nielson, F. Nielson, H. Pilegaard, Spatial analysis of bioambients, in: Proc. of Static Analysis International Symposium (SAS'04), Vol. 3148 of Lecture Notes in Computer Science, Springer, 2004, pp. 69–83.
[9] C. Bodei, R. Gori, F. Levi, An analysis for causal properties of membrane interactions, Electr. Notes Theor. Comput. Sci. 299 (2013) 15–31.
[10] M. Guerriero, D. Prandi, C. Priami, P. Quaglia, Process calculi abstractions for biology, in: Algorithmic Bioprocesses Natural Computing Series, Springer, 2009, pp. 463–486.
[11] C. Priami, P. Quaglia, Beta binders for biological interactions, in: Proc. of Computational Methods in Systems Biology, Vol. 3082 of LNBI, Springer, 2005, pp. 20 – 33.
[12] A. Regev, E. Panina, W. Silverman, L. Cardelli, E. Shapiro, Bioambients: An abstraction for biological compartements, Theor. Comput. Sci. 325 (1) (2004) 141–167.
[13] H. Pilegaard, H. R. Nielson, F. Nielson, Context dependent analysis of bioambients, in: Simulation and Verification of Dynamic Systems, Vol. 06161 of Dagstuhl Seminar Proceedings, 2006.
[14] H. Pilegaard, H. R. Nielson, F. Nielson, Pathway analysis for bioambients, J. Log. Algebr. Program. 77 (1-2) (2008) 92–130.
[15] R. Gori, F. Levi, A new occurrence counting analysis for bioambients, in: Proc. of Programming Languages and Systems (APLAS'05), Vol. 3780 of Lecture Notes in Computer Science, Springer, 2005, pp. 381–400.
[16] R. Gori, F. Levi, An analysis for proving temporal properties of biological systems, in: Proc. of Programming Languages and Systems (APLAS'06), Vol. 4279 of Lecture Notes in Computer Science, Springer, 2005, pp. 234–252.
[17] R. Gori, F. Levi, Abstract interpretation based verification of temporal properties for bioambients, Inf. Comput. 208 (8) (2010) 869–921.
[18] C. Bodei, R. Gori, F. Levi, Causal static analysis for brane calculi, Theor. Comput. Sci. 587 (2015) 73–103.
[19] R. Barbuti, R. Gori, F. Levi, P. Milazzo, Investigating dynamic causalities in reaction systems, Theor. Comput. Sci. 623 (2016) 114–145.
[20] R. Barbuti, R. Gori, F. Levi, P. Milazzo, Specialized predictor for reaction systems with context properties, Fundam. Inform. 147 (2-3) (2016) 173–191.
[21] J. Feret, Abstract interpretation-based static analysis of mobile ambients, in: P. Cousot (Ed.), Proc. of Static Analysis International Symposium (SAS'01), Vol. 2126 of Lecture Notes in Computer Science, Springer, 2001, pp. 412–430.
[22] F. Nielson, R. Hansen, H. R. Nielson, Abstract interpretation of mobile ambients, Sci. Comput. Program. 47 (2-3) (2003) 145–175.
[23] F. Levi, S. Maffeis, On abstract interpretation of mobile ambients, Inf. Comput. 188 (2) (2004) 179–240.
[24] L. Cardelli, A. Gordon, Mobile ambients, Theor. Comp. Sci. 240 (1) (2000) 177–213.
[25] N. Busi, Towards a causal semantics for brane calculi, in: What is it About Government that Americans Dislike, University Press, 2007, pp. 1945–1965.
[26] C. Bodei, A. Bracciali, D. Chiarugi, On deducing causality in metabolic networks, BMC Bioinformatics 9 (4).
[27] R. Milner, Communicating and mobile systems: the π-calculus, Cambridge University Press, Cambridge, 1999.
[28] H. R. Nielson, F. Nielson, Shape analysis for mobile ambients, in: Proc. of the 27th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'00), ACM Press, 2000, pp. 142–154.
[29] J. Feret, Confidentiality analysis of mobile systems, in: J. Palsberg (Ed.), Proc. of Static Analysis International Symposium (SAS'00), Vol. 1824 of Lecture Notes in Computer Science, Springer, 2000, pp. 135–154.
[30] J. Feret, Occurrence counting analysis for the pi-calculus, Electr. Notes Theor. Comput. Sci. 39 (2) (2001) 1–18.
[31] J. Feret, Abstract interpretation of mobile systems, J. Log. Algebr. Program. 63 (1) (2005) 59–130.

[32] J. Feret, Analysis of mobile systems by abstract interpretation, Ph.D. thesis, Ècole Polytechnique, Paris, France (2005).
[33] C. Bodei, L. Brodo, R. Gori, F. Levi, D. Hermith, F. Levi, A global occurrence counting analysis for brane calculi., in: Proc. of the Logic-Based Program Synthesis and Transformation (LOPSTR'15), Vol. 9527 of Lecture Notes in Compuer Science, 2015, pp. 179–200.
[34] A. Wandinger-Ness, M. Zerial, Rab proteins and the compartmentalization of the endosomal system, Cold Spring Harb Perspect. Biol. 6 (11).
[35] N. Busi, C. Zandron, Modeling and analysis of biological processes by mem(brane) calculi and systems, in: Proc. of the Winter Simulation Conference WSC 2006, 2006, pp. 1646–1655.

Appendix A. The formal proofs

In this appendix, we restate the lemmata and theorems presented earlier in the paper and give the proofs of their correctness. For the sake of brevity, we prove our results on the MBD fragment of Brane calculi. The proofs for the PEP fragment are similar.

Appendix A.1. Properties of the abstract states

The proofs of the main theorems rely on some auxiliary properties. The next lemma describes a relevant property of the *translation function*. If two systems P and Q (two membrane processes σ and τ, resp.) are congruent, then their translations give rise to the same abstract state, up to congruence on the continuations of sequential processes. As a consequence, we derive the related property for the *abstraction function* (Lemma 2). Recall that we overload = on abstract states to denote the syntactic equivalence of the states up to congruence on the continuations of sequential processes.

Lemma 1 (Congruence 1)**.**

- *Let* $\sigma, \tau \in \mathtt{Proc}$ *be two membrane processes such that* $\sigma \equiv \tau$*. For any abstract membrane label* $\Delta^\circ \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}$ *we have that* $t^\circ(\Delta^\circ, \sigma) = t^\circ(\Delta^\circ, \tau)$*;*
- *Let* $P, Q \in \mathtt{Sys}$ *be two systems such that* $P \equiv Q$*. For any abstract membrane label* $\Delta^\circ \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}$ *we have that* $t^\circ(\Delta^\circ, P) = t^\circ(\Delta^\circ, Q)$*.*

Proof. It suffices to inspect the rules for $\equiv$ in Table 2.

- The proof is done by induction on the depth of the derivation of $\sigma \equiv \tau$.
  - **case** $\sigma'|\tau' \equiv \tau'|\sigma'$ with $\sigma = \sigma'|\tau'$ and $\tau = \tau'|\sigma'$. By applying the translation function to σ, we derive $t^\circ(\Delta^\circ, \sigma'|\tau') = (R_1^\circ \cup R_2^\circ, O_1^\circ \cup^+ O_2^\circ)$, while by applying t° to τ, we derive $t^\circ(\Delta^\circ, \tau'|\sigma') = (R_2^\circ \cup R_1^\circ, O_2^\circ \cup^+ O_1^\circ)$, where $t^\circ(\Delta^\circ, \sigma') = (R_1^\circ, O_1^\circ)$ and $t^\circ(\Delta^\circ, \tau') = (R_2^\circ, O_2^\circ)$. Hence, the thesis derives from the commutativity property of the operators $\cup$ and $\cup^+$.
  - **case** $\sigma'|(\tau'|\tau'') \equiv (\tau'|\sigma')|\tau''$ with $\sigma = \sigma'|(\tau'|\tau'')$ and $\tau = (\tau'|\sigma')|\tau''$. Similarly to the above case, the thesis derives from the associative property of $\cup$ and $\cup^+$.

- **case** $\sigma'|0 \equiv \sigma'$ with $\sigma = \sigma'|0$ and $\tau = \sigma'$. The thesis follows because we have $t^\circ(\Delta^\circ, \sigma) = t^\circ(\Delta^\circ, \sigma'|0) = (R^\circ \cup \emptyset, O^\circ \cup^+ \emptyset) = (R^\circ, O^\circ) = t^\circ(\Delta^\circ, \tau')$, where $t^\circ(\Delta^\circ, \sigma') = (R^\circ, O^\circ)$ and $t^\circ(\Delta^\circ, 0) = (\emptyset, \emptyset)$.
- **case** $!0 \equiv 0$ with $\sigma = !0$ and $\tau = 0$. In this case, we have $t^\circ(\Delta^\circ, \sigma) = t^\circ(\Delta^\circ, !0) = (\emptyset, \emptyset[\omega/\ell]_{\ell \in dom(O^\circ)}) = (\emptyset, \emptyset)$, since by definition, $\emptyset[\omega/\ell]_{\ell \in dom(O^\circ)} = \emptyset$. The thesis follows because $t^\circ(\Delta^\circ, 0) = (\emptyset, \emptyset)$.
- **case** $!(\sigma'|\tau') \equiv !\sigma'|!\tau'$ with $\sigma = !(\sigma'|\tau')$ and $\tau = !\sigma'|!\tau'$. In this case, we have $t^\circ(\Delta^\circ, \sigma) = t^\circ(\Delta^\circ, !(\sigma'|\tau')) = (R_1^\circ \cup R_2^\circ, (O_1^\circ \cup^+ O_2^\circ)[\omega/\ell]_{\ell \in dom(O_1^\circ \cup^+ O_2^\circ)})$ where $t^\circ(\Delta^\circ, \sigma') = (R_1^\circ, O_1^\circ)$ and $t^\circ(\Delta^\circ, \tau') = (R_2^\circ, O_2^\circ)$.
  Instead, $t^\circ(\Delta^\circ, \tau) = t^\circ(\Delta^\circ, !\sigma'|!\tau') = (R_1^\circ \cup R_2^\circ, O_1^{\circ\prime} \cup^+ O_2^{\circ\prime})$, where $O_i^{\circ\prime} = O_i^\circ[\omega/\ell]_{\ell \in dom(O_i^\circ)}$, for $i \in \{1, 2\}$. The thesis follows, because, by definition of $\cup^+$, we have that $(O_1^\circ \cup^+ O_2^\circ)[\omega/\ell]_{\ell \in dom(O_1^\circ \cup^+ O_2^\circ)} = O_1^\circ[\omega/\ell]_{\ell \in dom(O_1^\circ)} \cup^+ O_2^\circ[\omega/\ell]_{\ell \in dom(O_2^\circ)}$.
- **case** $!!\sigma' \equiv !\sigma'$ with $\sigma = !!\sigma'$ and $\tau = !\sigma'$. In this case, we have $t^\circ(\Delta^\circ, \sigma) = t^\circ(\Delta^\circ, !!\sigma') = (R^\circ, O_1^\circ)$, where $O_1^\circ = (O^\circ[\omega/\ell]_{\ell \in dom(O^\circ)})[\omega/\ell]_{\ell \in dom(O^\circ)}$ for $t^\circ(\Delta^\circ, \sigma') = (R^\circ, O^\circ)$.
  Instead, $t^\circ(\Delta^\circ, \tau) = t^\circ(\Delta^\circ, !\sigma') = (R^\circ, O^\circ[\omega/\ell]_{\ell \in dom(O^\circ)})$. The thesis follows because $(O^\circ[\omega/\ell]_{\ell \in dom(O^\circ)})[\omega/\ell]_{\ell \in dom(O^\circ)} = O^\circ[\omega/\ell]_{\ell \in dom(O^\circ)}$.
- **case** $!\sigma' \equiv \sigma'|!\sigma'$ with $\sigma = !\sigma'$ and $\tau = \sigma'|!\sigma'$. In this case, we have $t^\circ(\Delta^\circ, \sigma) = t^\circ(\Delta^\circ, !\sigma') = (R^\circ, O^\circ[\omega/\ell]_{\ell \in dom(O^\circ)})$ where $t^\circ(\Delta^\circ, \sigma') = (R^\circ, O^\circ)$.
  Instead, $t^\circ(\Delta^\circ, \tau) = t^\circ(\Delta^\circ, \sigma'|!\sigma') = (R^\circ \cup R^\circ, O^\circ[\omega/\ell]_{\ell \in dom(O^\circ)} \cup^+ O^\circ)$. We observe that $R^\circ \cup R^\circ = R^\circ$ and $O^\circ[\omega/\ell]_{\ell \in dom(O^\circ)} \cup^+ O^\circ = O^\circ[\omega/\ell]_{\ell \in dom(O^\circ)}$ by definition of $\cup^+$. Indeed, for any $\ell \in dom(O^\circ)$, we have that $O^\circ[\omega/\ell]_{\ell \in dom(O^\circ)}(\ell) = \omega$. Hence, the thesis follows.
- **case** $\rho|\tau' \equiv \rho'|\tau'$, where $\rho \equiv \rho'$, with $\sigma = \rho|\tau'$ and $\tau = \rho'|\tau'$. In this case, we have $t^\circ(\Delta^\circ, \sigma) = t^\circ(\Delta^\circ, \rho|\tau') = (R_1^\circ \cup R_2^\circ, O_1^\circ \cup^+ O_2^\circ)$, where $t^\circ(\Delta^\circ, \rho) = (R_1^\circ, O_1^\circ)$ and $t^\circ(\Delta^\circ, \tau') = (R_2^\circ, O_2^\circ)$.
  Instead, $t^\circ(\Delta^\circ, \tau) = t^\circ(\Delta^\circ, \rho'|\tau') = (R_1^{\circ\prime} \cup R_2^\circ, O_1^{\circ\prime} \cup^+ O_2^\circ)$, where $t^\circ(\Delta^\circ, \rho') = (R_1^{\circ\prime}, O_1^{\circ\prime})$.
  Since $\rho \equiv \rho'$, by induction hypothesis, we derive that $t^\circ(\Delta^\circ, \rho) = (R_1^\circ, O_1^\circ) = (R_1^{\circ\prime}, O_1^{\circ\prime}) = t^\circ(\Delta^\circ, \rho')$. Hence, $(R_1^\circ \cup R_2^\circ, O_1^\circ \cup^+ O_2^\circ) = (R_1^{\circ\prime} \cup R_2^\circ, O_1^{\circ\prime} \cup^+ O_2^\circ)$ and the thesis follows.
- **case** $!\rho \equiv !\rho'$ where $\rho \equiv \rho'$with $\sigma = !\rho$ and $\tau = !\rho'$. In this case, we have $t^\circ(\Delta^\circ, \sigma) = t^\circ(\Delta^\circ, !\rho) = (R^\circ, O^\circ[\omega/\ell]_{\ell \in dom(O^\circ)})$, where $t^\circ(\Delta^\circ, \rho) = (R^\circ, O^\circ)$.
  Instead, $t^\circ(\Delta^\circ, \tau) = t^\circ(\Delta^\circ, !\rho') = (R^{\circ\prime}, O^{\circ\prime}[\omega/\ell]_{\ell \in dom(O^\circ)})$, where $t^\circ(\Delta^\circ, \rho') = (R^{\circ\prime}, O^{\circ\prime})$.
  Since $\rho \equiv \rho'$, the thesis follows, by induction hypothesis, as in the previous case.
- **case** $a^\lambda.\rho \equiv a^\lambda.\rho'$, where $\rho \equiv \rho'$ with $\sigma = a^\lambda.\rho$ and $\tau = a^\lambda.\rho'$. In this case, we have $t^\circ(\Delta^\circ, \sigma) = t^\circ(\Delta^\circ, a^\lambda.\rho) = (\{(\Delta^\circ, a^\lambda.\rho)\}, \{(\lambda, 1)\})$.
  Instead, $t^\circ(\Delta^\circ, \tau) = t^\circ(\Delta^\circ, a^\lambda.\rho') = (\{(\Delta^\circ, a^\lambda.\rho')\}, \{(\lambda, 1)\})$. Since we reason up to congruence on the continuations of sequential processes, we can conclude that $t^\circ(\Delta^\circ, \sigma) = t^\circ(\Delta^\circ, \tau)$.

- The proof is done by induction on the depth of the derivation of $P \equiv Q$. Most of the cases are similar to the corresponding ones in the proof for membrane processes. For the sake of brevity, we only focus on the most relevant cases.

– **case** $\sigma(\!|P'|\!)^\Gamma \equiv \tau(\!|Q'|\!)^\Gamma$, where $P' \equiv Q'$, $\sigma \equiv \tau$, and with $P = \sigma(\!|P'|\!)^\Gamma$ and $Q = \tau(\!|Q'|\!)^\Gamma$.
In this case, by applying the translation function to P, we have

$$t^\circ(\Delta^\circ, \sigma(\!|P'|\!)^\Gamma) \quad = \quad \begin{cases} (R_1^\circ \cup R_2^\circ \cup \{(\Delta^\circ, \Gamma^\bullet)\}, O_1^\circ \cup^+ O_2^\circ \cup^+ \{(\Gamma^\bullet, 1)\}) & \text{if } \sigma \not\equiv 0 \vee P' \not\equiv \diamond \\ (\emptyset, \emptyset) & \text{otherwise} \end{cases}$$

where $t^\circ(\Gamma^\bullet, P') = (R_1^\circ, O_1^\circ)$ and $t^\circ(\Gamma^\bullet, \sigma) = (R_2^\circ, O_2^\circ)$.
Instead, by applying the translation function to Q, we have

$$t^\circ(\Delta^\circ, \tau(\!|Q'|\!)^\Gamma) \quad = \quad \begin{cases} (R_1^{\circ\prime} \cup R_2^{\circ\prime} \cup \{(\Delta^\circ, \Gamma^\bullet)\}, O_1^{\circ\prime} \cup^+ O_2^{\circ\prime} \cup^+ \{(\Gamma^\bullet, 1)\}) & \text{if } \tau \not\equiv 0 \vee Q' \not\equiv \diamond \\ (\emptyset, \emptyset) & \text{otherwise} \end{cases}$$

where $t^\circ(\Gamma^\bullet, Q') = (R_1^{\circ\prime}, O_1^{\circ\prime})$ and $t^\circ(\Gamma^\bullet, \tau) = (R_2^{\circ\prime}, O_2^{\circ\prime})$.
Since $\sigma \equiv \tau$, we derive (by applying the first part of the lemma) that $t^\circ(\Gamma^\bullet, \sigma) = (R_2^\circ, O_2^\circ) = (R_2^{\circ\prime}, O_2^{\circ\prime}) = t^\circ(\Gamma^\bullet, \tau)$. Moreover, since $P' \equiv Q'$, by applying the induction hypothesis, we also have that $t^\circ(\Gamma^\bullet, P') = (R_1^\circ, O_1^\circ) = (R_1^{\circ\prime}, O_1^{\circ\prime}) = t^\circ(\Gamma^\bullet, Q')$. Hence, the thesis follows.

– **case** $0(\!|\!|)^\Gamma \equiv \diamond$ with $P = 0(\!|\!|)^\Gamma$ and $Q = \diamond$. In this case, we have $t^\circ(\Gamma^\bullet, \sigma) = t^\circ(\Delta^\circ, 0(\!|\!|)^\Gamma) = (\emptyset, \emptyset)$. Since $t^\circ(\Gamma^\bullet, \tau) = t^\circ(\Delta^\circ, \diamond) = (\emptyset, \emptyset)$, the thesis follows.

□

From the previous result we derive a related property for the *abstraction function*.

Lemma 2 (Congruence 2)**.** *Let $P, Q \in \mathtt{Sys}$ be two systems. If $P \equiv Q$ then $\alpha_{\mathtt{Sys}}(P) = \alpha_{\mathtt{Sys}}(Q)$.*

Proof. By Definition 10 of the abstraction function, we have $\alpha_{\mathtt{Sys}}(P) = (R_1^\circ, O_1^\circ \cup^+ \{(@, 1)\})$ with $t^\circ(@, P) = (R_1^\circ, O_1^\circ)$ and $\alpha_{\mathtt{Sys}}(Q) = (R_2^\circ, O_2^\circ \cup^+ \{(@, 1)\})$ with $t^\circ(@, Q) = (R_2^\circ, O_2^\circ)$. Since $P \equiv Q$, by Lemma 1, we have that $t^\circ(@, P) = t^\circ(@, Q)$ and, therefore, that $(R_2^\circ, O_2^\circ) = (R_1^\circ, O_1^\circ)$.

□

Theorem 1 (Galois Connection)**.** *The pair of functions $(\bar\alpha, \bar\gamma)$ in Definition 11 is a* Galois connection *between $(\wp(\mathtt{Sys}), \subseteq)$ and $(\mathcal{S}^\circ, \sqsubseteq^\circ)$.*

Proof. The functions $\bar\alpha : \wp(\mathtt{Sys}) \to \mathcal{S}^\circ$ and $\bar\gamma : \mathcal{S}^\circ \to \wp(\mathtt{Sys})$ are obviously monotone.

- for each $X \in \wp(\mathtt{Sys})$, we have to prove that $\bar\gamma(\bar\alpha(X)) \supseteq X$. Since $\bar\alpha(X) = \bigsqcup^\circ_{P \in X} \alpha_{\mathtt{Sys}}(P)$, we have that $\bar\gamma(\bar\alpha(X)) = \{P \mid \alpha_{\mathtt{Sys}}(P) \sqsubseteq^\circ \bigsqcup^\circ_{P \in X} \alpha_{\mathtt{Sys}}(P)\}$. We recall that $\bigsqcup^\circ$ stands for the least upper bound on the $\mathcal{S}^\circ$ domain. Hence, for each $P \in X$, we have that $\alpha_{\mathtt{Sys}}(P) \sqsubseteq^\circ \bigsqcup^\circ_{P \in X} \alpha_{sys}(P)$. We can then conclude that $X \subseteq \bar\gamma(\bar\alpha(X))$.

- for each $S^\circ \in \mathcal{S}^\circ$, we have to prove that $\bar\alpha(\bar\gamma(S^\circ)) \sqsubseteq^\circ S^\circ$. We have $\bar\gamma(S^\circ) = \{P \mid \alpha_{\mathtt{Sys}}(P) \sqsubseteq^\circ S^\circ\}$, and, therefore, $\bar\alpha(\bar\gamma(S^\circ)) = \bigsqcup^\circ_{\{P \mid \alpha_{\mathtt{Sys}}(P) \sqsubseteq^\circ S^\circ\}} \alpha_{\mathtt{Sys}}(P)$.
Now it can be easily seen that $\bigsqcup^\circ_{\{P \mid \alpha_{\mathtt{Sys}}(P) \sqsubseteq^\circ S^\circ\}} \alpha_{\mathtt{Sys}}(P) \sqsubseteq^\circ S^\circ$, because, by definition, $\bigsqcup^\circ$ is the least upper bound on $\mathcal{S}^\circ$.

□

Appendix A.2. Safety of the analysis

We prove the safety of the abstract semantics with respect to the collecting semantics. First, we relate concrete transitions (defined by the rules in Table 3) with the abstract ones (defined by the rules in Tables 9 and 10). More specifically, we show that the abstract transitions that exit from an abstract state S°, which *safely approximates* a system P, *over-approximate* the transitions that exit from P. We recall that an abstract state S° *safely approximates* a system P, provided that $\alpha_{\mathtt{Sys}}(P) \sqsubseteq^\circ S^\circ$. Hence, if $\alpha_{\mathtt{Sys}}(P) \sqsubseteq^\circ S^\circ$ then for any transition of P there exists a corresponding abstract transition of S°. This property (formalised by Theorem 2) requires the following auxiliary properties.

Proposition Appendix A.1. *Let $S_1^\circ \in \mathcal{S}^\circ$ be an abstract state such that $S_1^\circ = (R_1^\circ, O_1^\circ)$. For any abstract transition $S_1^\circ \xrightarrow{l^\circ}_\circ S_2^\circ$ where $S_2^\circ = (R_2^\circ, O_2^\circ)$ we have that $R_1^\circ \subseteq R_2^\circ$ and $O_2^\circ = O_1^\circ \cup^+ O^{\circ\prime}$.*

It should be clear that the previous result guarantees, by definition of the approximation order $\sqsubseteq^\circ$ (given in Definition 9), that for any $S_1^\circ \xrightarrow{l^\circ}_\circ S_2^\circ$, we have that $S_1^\circ \sqsubseteq^\circ S_2^\circ$.

Lemma Appendix A.1.

- *Let $\sigma \in \mathtt{Proc}$ be a membrane process and $\Delta^\circ \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}$ be a membrane label s.t. $t^\circ(\Delta^\circ, \sigma) = (R^\circ, O^\circ)$. For any $\Gamma^\circ \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}$, we have that $t^\circ(\Gamma^\circ, \sigma) = (R_1^\circ, O^\circ)$, where*

$$R_1^\circ = (R^\circ \backslash \{(\Delta^\circ, \tau) \mid \tau \in \mathit{processes}(R^\circ, \Delta^\circ)\}) \cup \{(\Gamma^\circ, \tau) \mid \tau \in \mathit{processes}(R^\circ, \Delta^\circ)\}.$$

- *Let $P \in \mathtt{Sys}$ be a system and $\Delta^\circ \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}$ be a membrane label, let $t^\circ(\Delta^\circ, P) = (R^\circ, O^\circ)$. For any $\Gamma^\circ \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}$, we have that $t^\circ(\Gamma^\circ, P) = (R_1^\circ, O^\circ)$ where*

$$\begin{array}{rcl} R_1^\circ & = & (R^\circ \backslash (\{(\Delta^\circ, \tau) \mid \tau \in \mathit{processes}(R^\circ, \Delta^\circ)\} \cup \{(\Delta^\circ, \Theta^\circ) \mid \Theta^\circ \in \mathit{children}(R^\circ, \Delta^\circ)\})) \cup \\ & & \{(\Gamma^\circ, \tau) \mid \tau \in \mathit{processes}(R^\circ, \Delta^\circ)\} \cup \{(\Gamma^\circ, \Theta^\circ) \mid \Theta^\circ \in \mathit{children}(R^\circ, \Delta^\circ)\} \end{array}$$

Proof. Both results trivially hold, by simply observing that the rules of the translation function in Table 7 do not depend on the particular name of the enclosing ambient. Therefore, we can substitute Δ with another membrane Γ, by only replacing Δ with Γ in every pair (Δ°, τ) in $\mathit{processes}(R^\circ, \Delta^\circ)$ and in $\mathit{children}(R^\circ, \Delta^\circ)$. Note that the occurrence counting information O° is related to labels of P and does not involve Δ.

□

Lemma Appendix A.2. *Let $P_1 \in \mathtt{Sys}$ be a system and $\Delta^\circ \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}$ be an abstract membrane label such that $t^\circ(\Delta^\circ, P_1) = (R_1^{\circ\prime}, O_1^{\circ\prime})$. Moreover, let $S_1^\circ \in \mathcal{S}^\circ$ be an abstract state such that $S_1^\circ = (R_1^\circ, O_1^\circ)$ and $t^\circ(\Delta^\circ, P_1) \sqsubseteq^\circ S_1^\circ$. For any transition $P_1 \xrightarrow{l} P_2$, there exists an abstract transition $S_1^\circ \xrightarrow{l^\bullet}_\circ S_2^\circ$ with $S_2^\circ = (R_2^\circ, O_2^\circ)$ such that*

- *$t^\circ(\Delta^\circ, P_2) \sqsubseteq^\circ S_2^\circ$ with $t^\circ(\Delta^\circ, P_2) = (R_2^{\circ\prime}, O_2^{\circ\prime})$;*
- *$O_2^\circ = O_1^\circ \cup^+ O^{\circ\prime}$, $O_2^{\circ\prime} = O_{2,1}^\circ \cup^+ O_{2,2}^\circ$, $O_{2,1}^\circ \sqsubseteq_O O_1^{\circ\prime}$, and $O_{2,2}^\circ \sqsubseteq_O O^{\circ\prime}$.*

Proof. The proof is done by induction on the depth of derivation of $P_1 \xrightarrow{l} P_2$. We proceed by considering the rules in Table 3, which could have been applied to derive the transition $P_1 \xrightarrow{l} P_2$.

(Mate) In this case $P_1 \xrightarrow{l} P_2$ has been obtained by applying the rule (Mate), where $P_1 = \mathtt{mate}_n^{\lambda}.\sigma|\sigma_0(\!|P|\!)^{\Phi} \circ \overline{\mathtt{mate}}_n^{\mu}.\tau|\tau_0(\!|Q|\!)^{\Gamma}$, $P_2 = \sigma|\sigma_0|\tau|\tau_0(\!|P \circ Q|\!)^{\mathtt{mate}(\Phi,\Gamma,\lambda,\mu)}$ and $l = \mathtt{mate}_1(\Phi, \Gamma, \lambda, \mu)$.

We prove that the transition step can be simulated in the abstract setting, by starting from the abstract state $S_1^\circ = (R_1^\circ, O_1^\circ)$, by applying the corresponding abstract inference rule (Mate°). Hence, we use the hypothesis that $t^\circ(\Delta^\circ, P_1) \sqsubseteq^\circ S_1^\circ$. By applying the translation function to P_1 we have $t^\circ(\Delta^\circ, P_1) = (R_1^{\circ\prime}, O_1^{\circ\prime})$, where

$$
\begin{array}{rl}
R_1^{\circ\prime} = & R_{1,1}^\circ \cup R_{1,2}^\circ \cup R_{1,3}^\circ \cup R_{1,4}^\circ \cup \\
& \{(\Delta^\circ, \Phi^\bullet), (\Delta^\circ, \Gamma^\bullet)\} \cup \{(\Phi^\bullet, \mathtt{mate}_n^{\lambda}.\sigma) \cup \{(\Gamma^\bullet, \overline{\mathtt{mate}}_n^{\mu}.\tau)\} \\
O_1^{\circ\prime} = & O_{1,1}^\circ \cup^+ O_{1,2}^\circ \cup^+ O_{1,3}^\circ \cup^+ O_{1,4}^\circ \cup^+ \\
& \{(\Phi^\bullet, 1)\} \cup^+ \{(\Gamma^\bullet, 1)\} \cup^+ \{(\lambda, 1)\} \cup^+ \{(\mu, 1)\}
\end{array}
$$

where

$$
\begin{array}{ll}
t^\circ(\Phi^\bullet, P) = (R_{1,1}^\circ, O_{1,1}^\circ) & t^\circ(\Phi^\bullet, \sigma_0) = (R_{1,2}^\circ, O_{1,2}^\circ) \\
t^\circ(\Gamma^\bullet, Q) = (R_{1,3}^\circ, O_{1,3}^\circ) & t^\circ(\Gamma^\bullet, \tau_0) = (R_{1,4}^\circ, O_{1,4}^\circ)
\end{array}
$$

Since $t^\circ(\Delta^\circ, P_1) = (R_1^{\circ\prime}, O_1^{\circ\prime})$ and $t^\circ(\Delta^\circ, P_1) \sqsubseteq^\circ (R_1^\circ, O_1^\circ)$ in the abstract state S_1° the conditions for the application of the abstract inference rule (Mate°) are fulfilled. In fact, both $(\Delta^\circ, \Phi^\bullet), (\Delta^\circ, \Gamma^\bullet)$ are in R_1° and $(\Phi^\bullet, \mathtt{mate}_n^{\lambda}.\sigma), (\Gamma^\bullet, \overline{\mathtt{mate}}_n^{\mu}.\tau) \in R_1^\circ$. Moreover, $O_1^{\circ\prime} \sqsubseteq_O O_1^\circ$ implies that $O_1^\circ(\lambda) = x$ and $O_1^\circ(\mu) = y$ for some values x and y.

Therefore, by applying rule (Mate°), we obtain an abstract transition $S_1^\circ \xrightarrow{l^\circ}_\circ S_2^\circ$ such that $l^\circ = \mathtt{mate}_1(\Phi^\bullet, \Gamma^\bullet, \lambda, \mu) = l^\bullet$ and $S_2^\circ = (R_2^\circ, O_2^\circ)$, where

$$R_2^\circ = R_1^\circ \cup \{(\Delta^\circ, \Pi^\circ)\} \cup R_{3,1}^\circ \cup R_{3,2}^\circ \cup R^{\circ\prime}$$

$$O_2^\circ = O_1^\circ \cup^+ \{(\Pi^\circ, 1)\} \cup^+ O_{3,1}^\circ \cup^+ O_{3,2}^\circ$$

where $t^\circ(\Pi^\circ, \sigma) = (R_{3,1}^\circ, O_{3,1}^\circ)$, $t^\circ(\Pi^\circ, \tau) = (R_{3,2}^\circ, O_{3,2}^\circ)$ and

$$\Pi^\circ = \begin{cases} (\Phi^\bullet, \Gamma^\bullet, \lambda, \mu) \text{ if } (\Phi^\bullet, \Gamma^\bullet, \lambda, \mu) \in \widehat{\mathtt{Lab}}_{\mathcal{M}}^d, \\ (\top, \top, \lambda, \mu) \text{ otherwise} \end{cases}$$

and

$$
\begin{array}{rl}
R^{\circ\prime} = & \{(\Pi^\circ, \Theta^\circ) \mid \Theta^\circ \in \mathit{children}(R_1^\circ, \Phi^\bullet) \cup \mathit{children}(R_1^\circ, \Gamma^\bullet)\} \\
& \cup \{(\Pi^\circ, \tau') \mid \tau' \in \mathit{sub}(C_1, x, \mathtt{mate}_n^{\lambda}.\sigma) \cup \mathit{sub}(C_2, y, \overline{\mathtt{mate}}_n^{\mu}.\tau)\}
\end{array}
$$

and $C_1 = \mathit{processes}(R_1^\circ, \Phi^\bullet)$ and $C_2 = \mathit{processes}(R_1^\circ, \Gamma^\bullet)$.

We are left to prove that $t^\circ(\Delta^\circ, P_2) \sqsubseteq^\circ S_2^\circ$,where $P_2 = \sigma|\sigma_0|\tau|\tau_0(\!|P \circ Q|\!)^{\Pi}$ with $\Pi = \mathtt{mate}(\Phi, \Gamma, \lambda, \mu)$. In this case, by applying the translation function to P_2, we have

$$t^\circ(\Delta^\circ, P_2) = t^\circ(\Delta^\circ, \sigma|\sigma_0|\tau|\tau_0(\!|P \circ Q|\!)^\Pi) = (R_2^{\circ\prime}, O_2^{\circ\prime})$$

where

$$R_2^{\circ\prime} = \{(\Delta^\circ, \Pi^\circ)\} \cup R_{3,1}^\circ \cup R_{3,2}^\circ \cup R_{1,1}^{\circ\,\prime} \cup R_{1,2}^{\circ\,\prime} \cup R_{1,3}^{\circ\,\prime} \cup R_{1,4}^{\circ\,\prime}$$

$$O_2^{\circ\prime} = \{(\Pi^\circ, 1)\} \cup^+ O_{3,1}^\circ \cup^+ O_{3,2}^\circ \cup^+ O_{1,1}^{\circ\,\prime} \cup^+ O_{1,2}^{\circ\,\prime} \cup^+ O_{1,3}^{\circ\,\prime} \cup^+ O_{1,4}^{\circ\,\prime}$$

where

$$t^\circ(\Pi^\circ, P) = (R_{1,1}^{\circ\,\prime}, O_{1,1}^{\circ\,\prime}) \quad t^\circ(\Pi^\circ, \sigma_0) = (R_{1,2}^{\circ\,\prime}, O_{1,2}^{\circ\,\prime})$$
$$t^\circ(\Pi^\circ, Q) = (R_{1,3}^{\circ\,\prime}, O_{1,3}^{\circ\,\prime}) \quad t^\circ(\Pi^\circ, \tau_0) = (R_{1,4}^{\circ\,\prime}, O_{1,4}^{\circ\,\prime})$$

To prove that $(R_2^{\circ\prime}, O_2^{\circ\prime}) \sqsubseteq^\circ (R_2^\circ, O_2^\circ)$, we prove that $R_{1,1}^{\circ\,\prime} \cup R_{1,2}^{\circ\,\prime} \cup R_{1,3}^{\circ\,\prime} \cup R_{1,4}^{\circ\,\prime} \sqsubseteq R^{\circ\prime}$ and that $O_{1,1}^{\circ\,\prime} \cup^+ O_{1,2}^{\circ\,\prime} \cup^+ O_{1,3}^{\circ\,\prime} \cup^+ O_{1,4}^{\circ\,\prime} \sqsubseteq_O O_1^\circ$.

Recall that $t^\circ(\Phi^\bullet, P) = (R_{1,1}^\circ, O_{1,1}^\circ)$, $t^\circ(\Phi^\bullet, \sigma_0) = (R_{1,2}^\circ, O_{1,2}^\circ)$, $t^\circ(\Gamma^\bullet, Q) = (R_{1,3}^\circ, O_{1,3}^\circ)$ and $t^\circ(\Gamma^\bullet, \tau_0) = (R_{1,4}^\circ, O_{1,4}^\circ)$ and $R_{1,1}^\circ \cup R_{1,2}^\circ \cup R_{1,3}^\circ \cup R_{1,4}^\circ \subseteq R_1^{\circ\prime}$ and $O_{1,1}^\circ \cup^+ O_{1,2}^\circ \cup^+ O_{1,3}^\circ \cup^+ O_{1,4}^\circ \sqsubseteq_O O_1^{\circ\prime}$. Since, by hypothesis, $(R_1^{\circ\prime}, O_1^{\circ\prime}) \sqsubseteq^\circ (R_1^\circ, O_1^\circ)$, we also have that $R_{1,1}^\circ \cup R_{1,2}^\circ \cup R_{1,3}^\circ \cup R_{1,4}^\circ \subseteq R_1^\circ$ and $O_{1,1}^\circ \cup^+ O_{1,2}^\circ \cup^+ O_{1,3}^\circ \cup^+ O_{1,4}^\circ \sqsubseteq_O O_1^\circ$.

By Lemma Appendix A.1, we can replace $\Phi^\bullet$ and $\Gamma^\bullet$ with the membrane Π° inside $t^\circ(\Phi^\bullet, P)$, $t^\circ(\Phi^\bullet, \sigma_0)$, $t^\circ(\Gamma^\bullet, Q)$, $t^\circ(\Gamma^\bullet, \tau_0)$. Therefore, we obtain the following pairs in the first component

$$\{(\Pi^\circ, \Theta^\circ) \mid \Theta^\circ \in children(R_{1,1}^\circ, \Phi^\bullet) \cup children(R_{1,3}^\circ, \Gamma^\bullet)\}$$
$$\cup \{(\Pi^\circ, \tau') \mid \tau' \in sub(C_1', x, \mathtt{mate}_n^\lambda.\sigma) \cup sub(C_2', y, \overline{\mathtt{mate}}_n^\mu.\tau)\}$$

with $C_1' = processes(R_{1,2}^\circ, \Phi^\bullet)$ and $C_2' = processes(R_{1,4}^\circ, \Gamma^\bullet)$ and the following result for the second component: $O_{1,1}^{\circ\,\prime} \cup^+ O_{1,2}^{\circ\,\prime} \cup^+ O_{1,3}^{\circ\,\prime} \cup^+ O_{1,4}^{\circ\,\prime} = O_{1,1}^\circ \cup^+ O_{1,2}^\circ \cup^+ O_{1,3}^\circ \cup^+ O_{1,4}^\circ$. Since $R_{1,1}^\circ \cup R_{1,2}^\circ \cup R_{1,3}^\circ \cup R_{1,4}^\circ \subseteq R_1^\circ$, we can conclude that $R_{1,1}^{\circ\,\prime} \cup R_{1,2}^{\circ\,\prime} \cup R_{1,3}^{\circ\,\prime} \cup R_{1,4}^{\circ\,\prime} \sqsubseteq R^{\circ\prime}$ and also that $O_{1,1}^{\circ\,\prime} \cup^+ O_{1,2}^{\circ\,\prime} \cup^+ O_{1,3}^{\circ\,\prime} \cup^+ O_{1,4}^{\circ\,\prime} \sqsubseteq_O O_1^\circ$.

Finally, O_2° can be rewritten as $O_1^\circ \cup^+ O^{\circ\prime}$, where $O^{\circ\prime} = \{(\Pi^\circ, 1)\} \cup^+ O_{3,1}^\circ \cup^+ O_{3,2}^\circ$. Furthermore, $O_2^{\circ\prime} = O_{2,1}^\circ \cup^+ O_{2,2}^\circ$, where $O_{2,1}^\circ = O_{1,1}^{\circ\,\prime} \cup^+ O_{1,2}^{\circ\,\prime} \cup^+ O_{1,3}^{\circ\,\prime} \cup^+ O_{1,4}^{\circ\,\prime}$ and thus $O_{2,1}^\circ \sqsubseteq_O O_1^\circ$; while $O_{2,2}^\circ = \{(\Pi^\circ, 1)\} \cup^+ O_{3,1}^\circ \cup^+ O_{3,2}^\circ$ that coincides with $O^{\circ\prime}$.

(Bud) The proof is similar to the cases of the inference rules (Mate) and (Drip).

(Drip) In this case $P_1 \xrightarrow{l} P_2$ has been obtained by applying the rule (Drip), where $P_1 = \mathtt{drip}^\lambda(\rho).\sigma|\tau(\!|P|\!)^\Gamma$, $P_2 = \rho(\!|\,|\!)^{\mathtt{drip}(\Gamma,\lambda)} \circ \sigma|\tau(\!|P|\!)^\Gamma$ and $l = \mathtt{drip}_1(\Gamma, \lambda)$.

We prove that the transition step can be simulated in the abstract setting, by starting from the abstract state $S_1^\circ = (R_1^\circ, O_1^\circ)$, by applying the corresponding abstract inference rule (Drip°). Hence, we use the fact that $t^\circ(\Delta^\circ, P_1) \sqsubseteq^\circ S_1^\circ$. By applying the translation function to P_1, we have

$$t^\circ(\Delta^\circ, P_1) = t^\circ(\Delta^\circ, \mathtt{drip}^\lambda(\rho).\sigma|\tau(\!|P|\!)^\Gamma) = (R_1^{\circ\prime}, O_1^{\circ\prime})$$

$$R_1^{\circ\prime} = R_{1,1}^\circ \cup R_{1,2}^\circ \cup \{(\Delta^\circ, \Gamma^\bullet)\} \cup \{(\Gamma^\bullet, \mathtt{drip}^\lambda(\rho).\sigma)\}$$

$$O_1^{\circ\prime} = O_{1,1}^\circ \cup^+ O_{1,2}^\circ \cup^+ \{(\Gamma^\bullet, 1)\} \cup^+ \{(\lambda, 1)\}$$

where $t^\circ(\Gamma^\bullet, P) = (R^\circ_{1,1}, O^\circ_{1,1})$ and $t^\circ(\Gamma^\bullet, \tau) = (R^\circ_{1,2}, O^\circ_{1,2})$.

Since $t^\circ(\Delta^\circ, P_1) = (R_1^{\circ\prime}, O_1^{\circ\prime})$ and $t^\circ(\Delta^\circ, P_1) \sqsubseteq^\circ (R^\circ_1, O^\circ_1)$ in the abstract state S°_1 the conditions for the application of the abstract inference rule (Drip°) are fulfilled. In fact, we have that $(\Delta^\circ, \Gamma^\bullet), (\Gamma^\bullet, \mathtt{drip}^\lambda(\rho).\sigma) \in R_1^{\circ\prime}$ and $R_1^{\circ\prime} \subseteq R^\circ_1$ implies that $(\Delta^\circ, \Gamma^\bullet), (\Gamma^\bullet, \mathtt{drip}^\lambda(\rho).\sigma) \in R^\circ_1$. Moreover, $O_1^{\circ\prime} \sqsubseteq_O O^\circ_1$ implies that $O^\circ_1(\lambda) = x$ for some multiplicity x.

Therefore, by applying rule (Drip°), we obtain an abstract transition $S^\circ_1 \xrightarrow{l^\circ}_\circ S^\circ_2$ such that $l^\circ = \mathtt{drip}_1(\Gamma^\bullet, \lambda) = l^\bullet$ and $S^\circ_2 = (R^\circ_2, O^\circ_2)$ where

$$R^\circ_2 = R^\circ_1 \cup \{(\Delta^\circ, \Pi^\circ)\} \cup R^\circ_{3,1} \cup R^\circ_{3,2}$$

$$O^\circ_2 = O^\circ_1 \cup^+ \{(\Pi^\circ, 1)\} \cup^+ O^\circ_{3,1} \cup^+ O^\circ_{3,2}$$

where $t^\circ(\Pi^\circ, \rho) = (R^\circ_{3,1}, O^\circ_{3,1})$, $t^\circ(\Gamma^\bullet, \sigma) = (R^\circ_{3,2}, O^\circ_{3,2})$ and

$$\Pi^\circ = \begin{cases} (\Gamma^\bullet, \lambda) \text{ if } (\Gamma^\bullet, \lambda) \in \widehat{\mathtt{Lab}}^d_{\mathcal{M}}, \\ (\top, \lambda) \text{ otherwise} \end{cases}$$

We are left to prove that $t^\circ(\Delta^\circ, P_2) \sqsubseteq^\circ S^\circ_2$, where $P_2 = \rho(\!|\!|^\Pi \circ \sigma|\tau(\!|P|\!|^\Gamma$ with $\Pi = \mathtt{drip}(\Gamma, \lambda)$. In this case, by applying the translation function to P_2, we have

$$t^\circ(\Delta^\circ, P_2) = t^\circ(\Delta^\circ, \rho(\!|\!|^\Pi \circ \sigma|\tau(\!|P|\!|^\Gamma) = (R_2^{\circ\prime}, O_2^{\circ\prime}) =$$

$$((\{(\Delta^\circ, \Pi^\circ) \cup R^\circ_{3,1}) \cup (\{(\Delta^\circ, \Gamma^\bullet)\}) \cup R^\circ_{1,1} \cup R^\circ_{1,2} \cup R^\circ_{3,2}),$$

$$(\{(\Pi^\circ, 1)\}) \cup^+ O^\circ_{3,1}) \cup^+ (\{(\Gamma^\bullet, 1)\}) \cup^+ O^\circ_{1,1} \cup^+ O^\circ_{1,2} \cup^+ O^\circ_{3,2}))).$$

Hence, by definition of the approximation order $\sqsubseteq^\circ$, it remains to prove that:

$$((R^\circ_{1,1} \cup R^\circ_{1,2} \cup \{(\Delta^\circ, \Gamma^\bullet)\}) \cup (R^\circ_{3,1} \cup R^\circ_{3,2} \cup \{(\Delta^\circ, \Pi^\circ)\}) \subseteq R^\circ_2$$

$$(O^\circ_{1,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\}) \cup^+ (O^\circ_{3,1} \cup^+ O^\circ_{3,2} \cup^+ \{(\Pi^\circ, 1)\})) \sqsubseteq_O O^\circ_2$$

Concerning the abstract representation, we recall that $(R^\circ_{1,1} \cup R^\circ_{1,2} \cup \{(\Delta^\circ, \Gamma^\bullet)\}) \subseteq R^\circ_1$. Moreover, by definition $R^\circ_2 = R^\circ_1 \cup \{(\Delta^\circ, \Pi^\circ)\} \cup R^\circ_{3,1} \cup R^\circ_{3,2}$; therefore we have that $R_2^{\circ\prime} \subseteq R^\circ_2$.

Concerning the occurrence counting function, we apply a similar argument. We recall that $(O^\circ_{1,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\}) \sqsubseteq_O O^\circ_1$. Moreover, by definition $O^\circ_2 = O^\circ_1 \cup^+ \{(\Pi^\circ, 1)\} \cup^+ O^\circ_{3,1} \cup^+ O^\circ_{3,2}$. As a consequence, we derive that

$$(O^\circ_{1,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\}) \cup^+ (O^\circ_{3,1} \cup^+ O^\circ_{3,2} \cup^+ \{(\Pi^\circ, 1)\})) \sqsubseteq_O O^\circ_2.$$

Finally, we show that the occurrence counting function $O_2^{\circ\prime}$ satisfies the claim of the theorem. We recall that $O^\circ_2 = O^\circ_1 \cup^+ \{(\Pi^\circ, 1)\} \cup^+ O^\circ_{3,1} \cup^+ O^\circ_{3,2}$. We conclude because $O_2^{\circ\prime} = (O^\circ_{1,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\}) \cup^+ (O^\circ_{3,1} \cup^+ O^\circ_{3,2} \cup^+ \{(\Pi^\circ, 1)\}))$, where $(O^\circ_{1,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\}) \sqsubseteq_O O_1{}'$.

(Par) In this case $P_1 \xrightarrow{l} P_2$ has been obtained by applying the inference rule (Par) with $P_1 = Q \circ P'$, $P_2 = Q' \circ P'$ and that there exists a transition $Q \xrightarrow{l} Q'$.

We prove that the transition step can be simulated in the abstract setting, starting from the abstract state $S_1^\circ = (R_1^\circ, O_1^\circ)$. Hence, we use the fact that $t^\circ(\Delta^\circ, P_1) \sqsubseteq^\circ S_1^\circ$. By applying the translation function to P_1, we derive

$$t^\circ(\Delta^\circ, P_1) = t^\circ(\Delta^\circ, Q \circ P') = (R_{1,1}^\circ \cup R_{1,2}^\circ, O_{1,1}^\circ \cup^+ O_{1,2}^\circ)$$

where $t^\circ(\Delta^\circ, Q) = (R_{1,1}^\circ, O_{1,1}^\circ)$ and $t^\circ(\Delta^\circ, P') = (R_{1,2}^\circ, O_{1,2}^\circ)$.

We observe that, by definition of $\sqsubseteq^\circ$ (see Definition 9), $t^\circ(\Delta^\circ, P_1) \sqsubseteq^\circ S_1^\circ$ implies that $R_{1,1}^\circ \cup R_{1,2}^\circ \subseteq R_1^\circ$ and $O_{1,1}^\circ \cup^+ O_{1,2}^\circ \sqsubseteq_O O_1^\circ$.

In particular, we have that $R_{1,1}^\circ \subseteq R_1^\circ$ and $O_{1,1}^\circ \sqsubseteq_O O_1^\circ$ and thus also $t^\circ(\Delta^\circ, Q) \sqsubseteq^\circ S_1^\circ$. Therefore, by applying the induction hypothesis, there exists an abstract transition from the abstract state S_1° that approximates the transition $Q \xrightarrow{l} Q'$. Let $S_1^\circ \xrightarrow{l^\bullet}_\circ S_2^\circ$ be the abstract transition such that $t^\circ(\Delta^\circ, Q') \sqsubseteq^\circ S_2^\circ$, where $S_2^\circ = (R_2^\circ, O_2^\circ)$ and $t^\circ(\Delta^\circ, Q') = (R_{1,1}^\circ{}', O_{1,1}^\circ{}')$. Moreover we have that $O_2^\circ = O_1^\circ \cup^+ O^{\circ\prime}$ such that $O_{1,1}^\circ{}' = O_{2,1}^\circ \cup^+ O_{2,2}^\circ$, $O_{2,1}^\circ \sqsubseteq_O O_{1,1}^\circ$ and $O_{2,2}^\circ \sqsubseteq_O O^{\circ\prime}$. Note that $t^\circ(\Delta^\circ, Q') \sqsubseteq^\circ S_2^\circ$ implies that $R_{1,1}^\circ{}' \subseteq R_2^\circ$.

We are left to prove that $t^\circ(\Delta^\circ, P_2) \sqsubseteq^\circ S_2^\circ$, where $P_2 = Q' \circ P'$. In this case, by applying the translation function to P_2, we have

$$t^\circ(\Delta^\circ, P_2) = t^\circ(\Delta^\circ, Q' \circ P') = (R_{1,1}^\circ{}' \cup R_{1,2}^\circ, O_{1,1}^\circ{}' \cup^+ O_{1,2}^\circ).$$

Hence, it remains to show that $R_{1,1}^\circ{}' \cup R_{1,2}^\circ \subseteq R_2^\circ$ and $O_{1,1}^\circ{}' \cup^+ O_{1,2}^\circ \sqsubseteq_O O_2^\circ$. We observe that, by Proposition Appendix A.1, $S_1^\circ \sqsubseteq^\circ S_2^\circ$ meaning that $R_1^\circ \subseteq R_2^\circ$ and $O_1^\circ \sqsubseteq_O O_2^\circ$.

For the abstract representations, we have $R_{1,1}^\circ \cup R_{1,2}^\circ \subseteq R_1^\circ \subseteq R_2^\circ$ and therefore $R_{1,2}^\circ \subseteq R_2^\circ$. Furthermore, from $R_{1,1}^\circ{}' \subseteq R_2^\circ$ and $R_{1,2}^\circ \subseteq R_2^\circ$, we derive that $R_{1,1}^\circ{}' \cup R_{1,2}^\circ \subseteq R_2^\circ$.

In the case of occurrence counting functions, we have that $O_{1,1}^\circ{}' = O_{2,1}^\circ \cup^+ O_{2,2}^\circ$ where $O_{2,1}^\circ \sqsubseteq_O O_{1,1}^\circ$ and $O_{2,2}^\circ \sqsubseteq_O O^{\circ\prime}$. From $O_{1,1}^\circ \cup^+ O_{1,2}^\circ \sqsubseteq_O O_1^\circ$ and $O_{2,1}^\circ \sqsubseteq_O O_{1,1}^\circ$, we obtain that $O_{2,1}^\circ \cup^+ O_{1,2}^\circ \sqsubseteq_O O_1^\circ$. Moreover we have that $O_2^\circ = O_1^\circ \cup^+ O^{\circ\prime}$ and $O_{2,2}^\circ \sqsubseteq_O O_2^{\circ\prime}$. As a consequence, we have $O_1^\circ \cup^+ O_{2,2}^\circ \sqsubseteq_O O_2^\circ$. Thus, since $O_{2,1}^\circ \cup^+ O_{1,2}^\circ \sqsubseteq_O O_1^\circ$, we have that $(O_{2,1}^\circ \cup^+ O_{1,2}^\circ) \cup^+ O_{2,2}^\circ \sqsubseteq_O O_2^\circ$. Hence, we have $O_{1,1}^\circ{}' \cup^+ O_{1,2}^\circ \sqsubseteq_O O_2^\circ$.

Finally, we show that the occurrence counting function $O_{1,1}^\circ{}' \cup^+ O_{1,2}^\circ$ (that is $(O_{2,1}^\circ \cup^+ O_{2,2}^\circ) \cup^+ O_{1,2}^\circ$) satisfies the claim of the theorem. We recall that $O_2^\circ = O_1^\circ \cup^+ O^{\circ\prime}$ such that $O_{2,2}^\circ \sqsubseteq_O O^{\circ\prime}$. Moreover, we have $O_{2,1}^\circ \sqsubseteq_O O_{1,1}^\circ$, and therefore we have also $O_{2,1}^\circ \cup^+ O_{1,2}^\circ \sqsubseteq_O O_{1,1}^\circ \cup^+ O_{1,2}^\circ$.

(Brane) In this case $P_1 \xrightarrow{l} P_2$ has been obtained by applying the inference rule (Brane) with $P_1 = \sigma(\!|P|\!)^\Gamma$, $P_2 = \sigma(\!|Q|\!)^\Gamma$ and that there exists a transition $P \xrightarrow{l} Q$.

We prove that the transition step can be simulated in the abstract setting, by starting from the abstract state $S_1^\circ = (R_1^\circ, O_1^\circ)$. Hence, we use the fact that $t^\circ(\Delta^\circ, P_1) \sqsubseteq^\circ S_1^\circ$. By applying

the translation function to P_1 we derive

$$t^\circ(\Delta^\circ, \sigma(\!|P|\!)^\Gamma) = \begin{cases} (R^\circ_{1,1} \cup R^\circ_{1,2} \cup \{(\Delta^\circ, \Gamma^\bullet)\}, O^\circ_{1,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\}) & \text{if } \sigma \not\equiv 0 \vee P \not\equiv \diamond \\ (\emptyset, \emptyset) & \text{otherwise} \end{cases}$$

where $t^\circ(\Gamma^\bullet, P) = (R^\circ_{1,1}, O^\circ_{1,1})$ and $t^\circ(\Gamma^\bullet, \sigma) = (R^\circ_{1,2}, O^\circ_{1,2})$.

We observe that $t^\circ(\Delta^\circ, P_1) \sqsubseteq^\circ S^\circ_1$ implies (by Definition 9) that $R^\circ_{1,1} \cup R^\circ_{1,2} \cup \{(\Delta^\circ, \Gamma^\bullet)\} \subseteq R^\circ_1$ and $O^\circ_{1,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\} \sqsubseteq_O O^\circ_1$.

In particular, we have $R^\circ_{1,1} \subseteq R^\circ_1$ and $O^\circ_{1,1} \sqsubseteq_O O^\circ_1$ and thus $t^\circ(\Gamma^\bullet, P) \sqsubseteq^\circ S^\circ_1$. Therefore, by applying the induction hypothesis, there exists an abstract transition from the abstract state S°_1 that approximates $P \xrightarrow{l} Q$. Let $S^\circ_1 \xrightarrow{l^\bullet}_\circ S^\circ_2$ be the abstract transition such that $t^\circ(\Gamma^\bullet, Q) \sqsubseteq^\circ S^\circ_2$ where $S^\circ_2 = (R^\circ_2, O^\circ_2)$ and $t^\circ(\Delta^\circ, Q) = ({R^\circ_{1,1}}', {O^\circ_{1,1}}')$. Moreover we have that $O^\circ_2 = O^\circ_1 \cup^+ O^{\circ\prime}$ such that ${O^\circ_{1,1}}' = O^\circ_{2,1} \cup^+ O^\circ_{2,2}$, $O^\circ_{2,1} \sqsubseteq_O O^\circ_{1,1}$ and $O^\circ_{2,2} \sqsubseteq_O O^{\circ\prime}$. Note that $t^\circ(\Gamma^\bullet, Q) \sqsubseteq^\circ S^\circ_2$ implies that ${R^\circ_{1,1}}' \subseteq R^\circ_2$ and ${O^\circ_{1,1}}' \sqsubseteq_O O^\circ_2$.

We are left to prove that $t^\circ(\Delta^\circ, P_2) \sqsubseteq^\circ S^\circ_2$ where $P_2 = \sigma(\!|Q|\!)^\Gamma$. In this case, by applying the translation function to P_2, we obtain

$$t^\circ(\Delta^\circ, \sigma(\!|Q|\!)^\Gamma) = \begin{cases} ({R^\circ_{1,1}}' \cup R^\circ_{1,2} \cup \{(\Delta^\circ, \Gamma^\bullet)\}, {O^\circ_{1,1}}' \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\}) & \text{if } \sigma \not\equiv 0 \vee Q \not\equiv \diamond \\ (\emptyset, \emptyset) & \text{otherwise} \end{cases}$$

Hence, it remains to show that ${R^\circ_{1,1}}' \cup R^\circ_{1,2} \cup \{(\Delta^\circ, \Gamma^\bullet)\} \subseteq R^\circ_2$ and ${O^\circ_{1,1}}' \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\}) \sqsubseteq_O O^\circ_2$. We observe that, by Proposition Appendix A.1, $S^\circ_1 \sqsubseteq^\circ S^\circ_2$ meaning that $R^\circ_1 \subseteq R^\circ_2$ and $O^\circ_1 \sqsubseteq_O O^\circ_2$.

In the case of abstract representations, we have $R^\circ_{1,1} \cup R^\circ_{1,2} \cup \{(\Delta^\circ, \Gamma^\bullet)\} \subseteq R^\circ_1 \subseteq R^\circ_2$ and therefore $R^\circ_{1,2} \cup \{(\Delta^\circ, \Gamma^\bullet)\} \subseteq R^\circ_2$. Furthermore, we have also ${R^\circ_{1,1}}' \subseteq R^\circ_2$ and thus also ${R^\circ_{1,1}}' \cup R^\circ_{1,2} \cup \{(\Delta^\circ, \Gamma^\bullet)\} \subseteq R^\circ_2$.

In the case of occurrence counting functions, we have that ${O^\circ_{1,1}}' = O^\circ_{2,1} \cup^+ O^\circ_{2,2}$ where $O^\circ_{2,1} \sqsubseteq_O O^\circ_{1,1}$. From $O^\circ_{1,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\} \sqsubseteq_O O^\circ_1$ and $O^\circ_{2,1} \sqsubseteq_O O^\circ_{1,1}$, we obtain that $O^\circ_{2,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\} \sqsubseteq_O O^\circ_1$. Moreover we have that $O^\circ_2 = O^\circ_1 \cup^+ O^{\circ\prime}$ such that $O^\circ_{2,2} \sqsubseteq_O O^{\circ\prime}$. As a consequence, we derive that $O^\circ_{2,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\} \cup^+ O^\circ_{2,2} \sqsubseteq_O O^\circ_2$.

Finally, we show that the occurrence counting function ${O^\circ_{1,1}}' \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\})$ (that is $O^\circ_{2,1} \cup^+ O^\circ_{2,2} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\}$) satisfies the claim of the theorem. We recall that $O^\circ_2 = O^\circ_1 \cup^+ O^{\circ\prime}$ such that $O^\circ_{2,2} \sqsubseteq_O O^{\circ\prime}$. Moreover, we have $O^\circ_{2,1} \sqsubseteq_O O^\circ_{1,1}$ and therefore we conclude that $O^\circ_{2,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\} \sqsubseteq_O O^\circ_{1,1} \cup^+ O^\circ_{1,2} \cup^+ \{(\Gamma^\bullet, 1)\})$

(Struct) In this case $P_1 \xrightarrow{l} P_2$ has been obtained by applying the inference rule (Struct) with Q_1 and Q_2 such that $P_1 \equiv Q_1$, $P_2 \equiv Q_2$ and a transition $Q_1 \xrightarrow{l} Q_2$.

We prove that the transition step can be simulated in the abstract setting, starting from the abstract state $S^\circ_1 = (R^\circ_1, O^\circ_1)$. Hence, we use the fact that $t^\circ(\Delta^\circ, P_1) \sqsubseteq^\circ S^\circ_1$ where $t^\circ(\Delta^\circ, P_1) =$

$(R_1^{\circ\prime}, O_1^{\circ\prime})$. Since $P_1 \equiv Q_1$ then, by Lemma 1, we have that $t^\circ(\Delta^\circ, P_1) = t^\circ(\Delta^\circ, Q_1) = (R_1^{\circ\prime}, O_1^{\circ\prime})$. As a consequence, we have also that $t^\circ(\Delta^\circ, Q_1) \sqsubseteq^\circ S_1^\circ$.

Therefore, by applying the induction hypothesis, there exists an abstract transition from the abstract state S_1° that approximates $Q_1 \xrightarrow{l} Q_2$. Let $S_1^\circ \xrightarrow{l^\bullet}_\circ S_2^\circ$ be the abstract transition such that $t^\circ(\Delta^\circ, Q_2) \sqsubseteq^\circ S_2^\circ$ where $S_2^\circ = (R_2^\circ, O_2^\circ)$. Furthermore, we have that $t^\circ(\Delta^\circ, Q_2) = (R_2^{\circ\prime}, O_2^{\circ\prime})$ where $O_2^\circ = O_1^\circ \cup^+ O^{\circ\prime}$ and $O_2^{\circ\prime} = O_{2,1}^\circ \cup^+ O_{2,2}^\circ$, such that $O_{2,1}^\circ \sqsubseteq_O O_1^{\circ\prime}$ and $O_{2,2}^\circ \sqsubseteq_O O^{\circ\prime}$.

Since $P_2 \equiv Q_2$ then, by applying again Lemma 1, we have that $t^\circ(\Delta^\circ, Q_2) = t^\circ(\Delta^\circ, P_2) = (R_2^{\circ\prime}, O_2^{\circ\prime})$. Therefore, we have also that $t^\circ(\Delta^\circ, P_2) \sqsubseteq^\circ S_2^\circ$. Finally, we observe that the occurrence counting function satisfies the claim of the theorem, because $O_2^{\circ\prime} = O_{2,1}^\circ \cup^+ O_{2,2}^\circ$ and $O_2^\circ = O_1^\circ \cup^+ O^{\circ\prime}$ such that $O_{2,1}^\circ \sqsubseteq_O O_1^{\circ\prime}$ and $O_{2,2}^\circ \sqsubseteq_O O^{\circ\prime}$.

□

Theorem 2. *Let $P_1 \in \mathtt{Sys}$ be a system and let $S_1^\circ \in \mathcal{S}^\circ$ be an abstract state such that $\alpha_{\mathtt{Sys}}(P_1) \sqsubseteq^\circ S_1^\circ$. For any transition $P_1 \xrightarrow{l} P_2$ there exists an abstract transition $S_1^\circ \xrightarrow{l^\bullet}_\circ S_2^\circ$ with $\alpha_{\mathtt{Sys}}(P_2) \sqsubseteq^\circ S_2^\circ$.*

Proof. By Definition 10 of the abstraction function, we have $\alpha_{\mathtt{Sys}}(P_1) = (R_1^\circ, O_1^\circ \cup^+ \{(@, 1)\})$ with $t^\circ(@, P_1) = (R_1^\circ, O_1^\circ)$. Since $\alpha_{\mathtt{Sys}}(P_1) \sqsubseteq^\circ S_1^\circ$ then, by applying Lemma Appendix A.2, there exists an abstract transition $S_1^\circ \xrightarrow{l^\bullet}_\circ S_2^\circ$ such that $t^\circ(@, P_2) \sqsubseteq^\circ S_2^\circ$ with $t^\circ(@, P_2) = (R_2^\circ, O_2^\circ)$. Hence, since $\alpha_{\mathtt{Sys}}(P_2) = (R_2^\circ, O_2^\circ \cup^+ \{(@, 1)\})$ we have also $\alpha_{\mathtt{Sys}}(P_2) \sqsubseteq^\circ S_2^\circ$. □

We have to prove now the main property (formalised by Theorem 4 below). It proves the safety of the collecting semantics w.r.t. the abstract semantics. We show that the configuration computed by the abstract semantics of a system *P safely approximates* all the paths of *P*. The proof is based on some auxiliary properties.

The following lemmata state some relevant properties of the operator *mul* on abstract states and, therefore, of configurations obtained by using meta-inference rules (1) and (2) in Table 12.

Lemma Appendix A.3. *Let $S_1^\circ, S_2^\circ \in \mathcal{S}^\circ$ be two abstract states such that $S_1^\circ \sqsubseteq^\circ S_2^\circ$. For any abstract transition label $l^\circ \in \mathtt{Lab}_\mathcal{T}^\circ$, we have that $mul(S_1^\circ, l^\circ) \leq mul(S_2^\circ, l^\circ)$.*

Proof. The proof directly follows from the definition of *mul*. □

The following property relates the operator *mul* to the number of occurrences given by the operator *occ* (presented in Definition 14).

Lemma 3. *Let $P \in \mathtt{Sys}$ be a well labelled system. For any path $p \in \mathcal{T}(P)$ with $\alpha_{path}(p) \sqsubseteq^{C^\circ} T^\circ \rhd S^\circ$ and for any transition label $l \in \mathtt{Lab}_\mathcal{T}$, we have that*

$$occ(p, l) \leq mul(S^\circ, l^\bullet).$$

Proof. We will prove the claim for $T^\circ \rhd S^\circ = \alpha_{path}(p)$. Since the operator *mul* is monotone w.r.t. the order on configurations, the case $T^\circ \rhd S^\circ \sqsupseteq^C \alpha_{path}(p)$ follows immediately.

The proof is by induction on the length n of the path p.

- $n = 1$. In this case, we have that $p = P \xrightarrow{l} P_1$ and $occ(p, l) = 1$. By definition of α_{path}, we have that $\alpha_{path}(p) = \{l^\bullet\} \rhd \alpha_{\mathtt{Sys}}(P) \sqcup^\circ \alpha_{\mathtt{Sys}}(P_1)$. Since process P makes a transition labelled l, this implies that the processes involved in l, without loss of generality let us call them λ and μ, are enabled in P. Therefore, by definition of $\alpha_{\mathtt{Sys}}$, λ and μ have a multiplicity equal or greater than 1 in $\alpha_{\mathtt{Sys}}(P)$. As a consequence, by definition of *mul*, $mul(\alpha_{\mathtt{Sys}}(P), l^\bullet) \geq 1 = occ(p, l)$. Moreover, by Lemma Appendix A.3, we have that $mul(\alpha_{\mathtt{Sys}}(P) \sqcup^\circ \alpha_{\mathtt{Sys}}(P_1), l^\bullet) \geq mul(\alpha_{\mathtt{Sys}}(P), l^\bullet) \geq 1 = occ(p, l)$.

- $n \geq 1$. In the inductive case we have that $p = P \xrightarrow{l_1} P_1 \dots P_{n-1} \xrightarrow{l_n} P_n$. By definition of α_{path} we have $\alpha_{path}(p) = \{l_1^\bullet, \dots, l_n^\bullet\} \rhd S^\circ_{n-1} \sqcup^\circ \alpha_{Sys}(P_n) \sqcup^\circ (\emptyset, \{(\lambda^\bullet, occ(p, l_n)) \mid \lambda \in lab(l_n)\})$ where $\alpha_{path}(p_1) = \{l_1^\bullet, \dots, l_{n-1}^\bullet\} \rhd S^\circ_{n-1}$ for $p_1 = P \xrightarrow{l_1} P_1 \dots P_{n-2} \xrightarrow{l_{n-1}} P_{n-1}$.

  By applying the induction hypothesis, we have that for any $l \in \mathtt{Lab}_{\mathcal{T}}$, $occ(p_1, l) \leq mul(S^\circ_{n-1}, l^\bullet)$.

  Note that for any $l \in \mathtt{Lab}_{\mathcal{T}}$ such that $l_n \neq l$, $occ(p, l) = occ(p_1, l)$. Therefore, in this case $occ(p_1, l) = occ(p, l) \leq mul(S^\circ_{n-1}, l^\bullet)$. Moreover, by definition of *mul*, we have that $mul(\ (\emptyset, \{(\lambda^\bullet, occ(p, l_n)) \mid \lambda \in lab(l_n)\}), l_n^\bullet) = occ(p, l_n)$. Hence, by Lemma Appendix A.3, we conclude that for any $l \in \mathtt{Lab}_{\mathcal{T}}$, $mul(S^\circ_{n-1} \sqcup^\circ \alpha_{Sys}(P_n) \sqcup^\circ (\emptyset, \{(\lambda^\bullet, occ(p, l_n)) \mid \lambda \in lab(l_n)\}, l^\bullet) = occ(p, l)$.

□

Lemma Appendix A.4. *Let $P \in \mathtt{Sys}$ be a system. For any abstract path $p^\circ \in \mathcal{T}^\circ(P)$ we have that $T^\circ_i \rhd S^\circ_i \sqsubseteq^{C^\circ} T^\circ_{i+1} \rhd S^\circ_{i+1}$, for any $i \in \{0, \dots, n-1\}$, where $S^\circ_0 = \alpha_{\mathtt{Sys}}(P)$, $T^\circ_0 = \emptyset$ and $p^\circ = T^\circ_0 \rhd S^\circ_0 \xrightarrow{l^\circ_1}_\rhd T^\circ_1 \rhd S^\circ_1 \dots T^\circ_{n-1} \rhd S^\circ_{n-1} \xrightarrow{l^\circ_n}_\rhd T^\circ_n \rhd S^\circ_n$.*

Proof. The claim follows from Proposition Appendix A.1 and by definition of meta-inference rules (1) and (2) in Table 12.

□

The following result establishes the soundness of the abstract paths with respect to the paths of a system. More precisely, the theorem shows that for any path there exists a corresponding abstract path, that approximates it.

Theorem 3. *Let $P \in \mathtt{Sys}$ be a well labelled system. For any path $p \in \mathcal{T}(P)$ there exists an abstract path $p^\circ \in \mathcal{T}^\circ(P)$ such that $p^\circ = \emptyset \rhd \alpha_{Sys}(P) \xrightarrow{T^{\circ\,*}}_\rhd T^\circ \rhd S^{\circ\prime}$ and $S^\circ \sqsubseteq S^{\circ\prime}$, where $\alpha_{path}(p) = T^\circ \rhd S^\circ$.*

Proof. The proof is by induction on the length n of the concrete path p.

- $n = 1$. In this case, we have that $p = P \xrightarrow{l} P_1$. By definition of α_{path}, we have that $\alpha_{path}(p) = \{l^\bullet\} \rhd \alpha_{\mathtt{Sys}}(P) \sqcup^\circ \alpha_{\mathtt{Sys}}(P_1)$. Moreover, from $P \xrightarrow{l} P_1$ we derive, by Theorem 2, that there exists a corresponding abstract transition $\alpha_{\mathtt{Sys}}(P) \xrightarrow{l^\bullet}_\circ S^\circ_1$ with $\alpha_{\mathtt{Sys}}(P_1) \sqsubseteq^\circ S^\circ_1$. Note that, in this case, we have $l^\bullet \notin \emptyset$ and therefore the meta-inference rule (1) in Table 12 can be applied to the initial configuration $\emptyset \rhd \alpha_{\mathtt{Sys}}(P)$. As a consequence, we obtain $\emptyset \rhd \alpha_{\mathtt{Sys}}(P) \xrightarrow{l^\bullet}_\rhd \{l^\bullet\} \rhd S^\circ_1$.

By Lemma Appendix A.4, we can conclude that $\alpha_{\mathsf{Sys}}(P) \sqsubseteq^{\circ} S_1^{\circ}$. Hence, $\alpha_{\mathsf{Sys}}(P) \sqsubseteq^{\circ} S_1^{\circ}$ together with $\alpha_{\mathsf{Sys}}(P_1) \sqsubseteq^{\circ} S_1^{\circ}$ implies that $\alpha_{\mathsf{Sys}}(P) \sqcup^{\circ} \alpha_{\mathsf{Sys}}(P_1) \sqsubseteq^{\circ} S_1^{\circ}$.

- $n \geq 1$. In the inductive case, we have that $p = P \xrightarrow{l_1} P_1 \ldots P_{n-1} \xrightarrow{l_n} P_n$. By definition of α_{path}, we have $\alpha_{path}(p) = \{l_1^{\bullet}, \ldots, l_n^{\bullet}\} \rhd S_{n-1}^{\circ} \sqcup^{\circ} \alpha_{Sys}(P_n) \sqcup^{\circ} (\emptyset, \{(\lambda^{\bullet}, occ(p, l_n)) \mid \lambda \in lab(l_n)\})$ where $\alpha_{path}(p_1) = \{l_1^{\bullet}, \ldots, l_{n-1}^{\bullet}\} \rhd S_{n-1}^{\circ}$ for $p_1 = P \xrightarrow{l_1} P_1 \ldots P_{n-2} \xrightarrow{l_{n-1}} P_{n-1}$.
  By applying the induction hypothesis, there exists an abstract path $p_1{}^{\circ} \in \mathcal{T}^{\circ}(P)$ such that $p_1{}^{\circ} = \emptyset \rhd \alpha_{Sys}(P) \xrightarrow{\{l_1^{\bullet}, \ldots, l_{n-1}^{\bullet}\}^*}_{\rhd} \{l_1^{\bullet}, \ldots, l_{n-1}^{\bullet}\} \rhd S_{n-1}^{\circ}{}'$ and $S_{n-1}^{\circ} \sqsubseteq S_{n-1}^{\circ}{}'$.
  Note that, by definition of α_{path}, we have that $\alpha_{Sys}(P_{n-1}) \sqsubseteq^{\circ} S_{n-1}^{\circ} \sqsubseteq S_{n-1}^{\circ}{}'$.

  Therefore, given $P_{n-1} \xrightarrow{l_n} P_n$ by Theorem 2, there exists a corresponding abstract transition $S_{n-1}^{\circ}{}' \xrightarrow{l_n^{\bullet}}_{\circ} S_n^{\circ\prime}$ such that $\alpha_{\mathsf{Sys}}(P_n) \sqsubseteq^{\circ} S_n^{\circ\prime}$.

  To compute the meta-transition step corresponding to $S_{n-1}^{\circ}{}' \xrightarrow{l_n^{\bullet}}_{\circ} S_n^{\circ\prime}$ for the configuration $\{l_1^{\bullet}, \ldots, l_{n-1}^{\bullet}\} \rhd S_{n-1}^{\circ}{}'$, we need to apply the meta-inference rules in Table 12. We recall that the choice of the meta-inference rules (1) and (2) depends on the multiplicity of the abstract transition label $l_n^{\bullet}$ (given by $mul(S_{n-1}^{\circ}{}', l_n^{\bullet})$) and on the set of abstract transition labels $\{l_1^{\bullet}, \ldots, l_{n-1}^{\bullet}\}$.

  Given the abstract transition $S_{n-1}^{\circ}{}' \xrightarrow{l_n^{\bullet}}_{\circ} S_n^{\circ\prime}$ and the configuration $\{l_1^{\bullet}, \ldots, l_{n-1}^{\bullet}\} \rhd S_{n-1}^{\circ}{}'$, we have the following cases:

  - $mul(S_{n-1}^{\circ}{}', l_n^{\bullet}) = \omega$ or, analogously, $mul(S_{n-1}^{\circ}{}', l_n^{\bullet}) = 1$ and $l_n^{\bullet} \notin \{l_1^{\bullet}, \ldots, l_{n-1}^{\bullet}\}$.
    In this case, the premises of the meta-inference rule (1) in Table 12 are satisfied, considering the abstract transition $S_{n-1}^{\circ}{}' \xrightarrow{l_n^{\bullet}}_{\circ} S_n^{\circ\prime}$. Hence we obtain

    $$\{l_1^{\bullet}, \ldots, l_{n-1}^{\bullet}\} \rhd S_{n-1}^{\circ}{}' \xrightarrow{l_n^{\bullet}}_{\rhd} \{l_1^{\bullet}, \ldots, l_n^{\bullet}\} \rhd S_n^{\circ\prime}.$$

    We are left to prove that $S_{n-1}^{\circ} \sqcup^{\circ} \alpha_{Sys}(P_n) \sqcup^{\circ} (\emptyset, \{(\lambda^{\bullet}, occ(p, l_n)) \mid \lambda \in lab(l_n)\}) \sqsubseteq^{\circ} S_n^{\circ\prime}$
    We have that $S_{n-1}^{\circ} \sqsubseteq^{\circ} S_n^{\circ\prime}$ and $\alpha_{\mathsf{Sys}}(P_n) \sqsubseteq^{\circ} S_n^{\circ\prime}$. As a consequence, we are left to show that $(\emptyset, \{(\lambda^{\bullet}, occ(p, l_n)) \mid \lambda \in lab(l_n)\}) \sqsubseteq^{\circ} S_n^{\circ\prime}$ where and $S_n^{\circ\prime} = (R_n^{\circ\prime}, O_n^{\circ\prime})$. We have the following two cases:

    * $mul(S_{n-1}^{\circ}{}', l_n^{\bullet}) = \omega$. In this case, since $S_{n-1}^{\circ}{}' \sqsubseteq^{\circ} S_n^{\circ\prime}$, by Lemma Appendix A.3, we have that $O_n^{\circ\prime}(\lambda) = \omega \geq occ(p, l_n)$ for $\lambda \in lab(l_n)$.
    * $mul(S_{n-1}^{\circ}{}', l_n^{\bullet}) = 1$ and $l_n^{\bullet} \notin \{l_1^{\bullet}, \ldots, l_{n-1}^{\bullet}\}$. This implies that $l_n \notin \{l_1, ..., l_{n-1}\}$ and therefore, $occ(p_1, l_n) = 0$ and $occ(p, l_n) = 1$. Now $S_{n-1}^{\circ} \sqsubseteq^{\circ} S_n^{\circ\prime}$ and $\alpha_{\mathsf{Sys}}(P_n) \sqsubseteq^{\circ} S_n^{\circ\prime}$ implies that the multiplicities of the action and the corresponding co-action involved in the move are greater than 0. Since $S_{n-1}^{\circ} \sqsubseteq^{\circ} S_n^{\circ\prime}$, we can conclude that $O_n^{\circ\prime}(\lambda^{\bullet}) \geq 1 = occ(p, l_n)$ for any $\lambda \in lab(l_n)$. Therefore, we have that t $O_n^{\circ\prime}(\lambda) = \omega \geq occ(p, l_n)$ for $\lambda \in lab(l_n)$.

– $mul(S^{\circ}_{n-1}{}', l^{\bullet}_n) = 1$ and $l^{\bullet}_n \in \{l^{\bullet}_1, \ldots, l^{\bullet}_{n-1}\}$.
In this case we have that also $l_n \in \{l_1, ..., l_{n-1}\}$. Note that we we cannot apply the meta-inference rules labelled $l^{\bullet}_n$ to configuration $\{l^{\bullet}_1, \ldots, l^{\bullet}_{n-1}\} \rhd S^{\circ}_{n-1}{}'$. Therefore, we prove that there exists another abstract state S°_k such that

$$\{l^{\bullet}_1, \ldots, l^{\bullet}_{n-1}\} \rhd S^{\circ}_{n-1}{}' \xrightarrow{\{l^{\bullet}_1,...,l^{\bullet}_{n-1}\}}{}^{*}_{\rhd} \{l^{\bullet}_1, \ldots, l^{\bullet}_{n-1}\} \rhd S^{\circ}_k \tag{A.1}$$

but this time $mul(S^{\circ}_k, l^{\bullet}_n) = \omega$.
To this aim, we assume, without loss of generality, that a move labelled l_n involves an action (let us assume with label λ) and a corresponding co-action (let us assume with label μ). Since $mul(\ S^{\circ}_{n-1}{}', l^{\bullet}) = 1$ this implies that at least one between λ and μ has multiplicity equal to 1 according to the information in $S^{\circ}_{n-1}{}'$. For the sake of simplicity, we assume that λ has multiplicity 1 in $S^{\circ}_{n-1}{}'$, while μ has multiplicity ω in $S^{\circ}_{n-1}{}'$. If also μ has multiplicity 1, we can apply the same reasoning also to μ.
Note that the transition labelled l_n can be performed more than once in the evolution of the well labelled process P, therefore it has to appear under a scope of a replication. It is worth noting that process λ cannot be directly under a replication, otherwise the translation function t° would have been assigned ω to λ (see Table 7). Hence, we can conclude that there exists a (sequential) process σ under the scope of a replication in P of the form $\sigma = \ldots.\sigma^{\lambda_n}_n.a^{\lambda}_1. \ldots$.
Moreover, since process λ was consumed by the first application of transition l_n but such process (labelled λ) was available again in P_{n-1}, we have that there must exist a move that enables the process $a^{\lambda}_1. \ldots$. Let us call $\tilde{l}$ such move involving a process labelled λ_n. Note that $\tilde{l} \neq l$ and $\tilde{l} \in \{l_1, ..., l_n\}$. It is worth noting that $\tilde{l}$ has to be performed at least twice in the concrete path $p_1 = P \xrightarrow{\{l_1,...,l_{n-1}\}}{}^{*} P_{n-1}$, because action labelled λ is consumed once by transition labelled l_n, but it is available in P_{n-1}. Therefore, we have that $occ(p_1, \tilde{l}) = \omega$. By Lemma 3, $mul(S^{\circ}_{n-1}, \tilde{l}^{\bullet}) \geq occ(p_1, \tilde{l}) = \omega$. Since $S^{\circ}_{n-1}{}' \sqsupseteq^{\circ} S^{\circ}_{n-1}$, $\{l^{\bullet}_1, ..., l^{\bullet}_{n-1}\} \rhd S^{\circ}_{n-1}{}' \xrightarrow{\tilde{l}^{\bullet}}_{\rhd} \{l^{\bullet}_1, ..., l^{\bullet}_{n-1}\} \rhd S^{\circ}_k$, for some S°_k, by applying the meta-inference rule (1) in Table 12. Note that $\alpha_{\mathsf{Sys}}(P_{n-1}) \sqsubseteq^{\circ} S^{\circ}_{n-1}{}' \sqsubseteq^{\circ} S^{\circ}_k$. Moreover, since we applied an abstract transition rule labelled $\tilde{l}^{\bullet}$ for the second time, and, each application of such rule implies that a new occurrence of a process labelled λ is introduced, we have that the multiplicity of λ, according to the information in S°_k, is now equal to ω. Moreover, we had assumed that μ has multiplicity equal to ω according to the information in $S^{\circ}_{n-1}{}'$, therefore we have that μ has multiplicity equal to ω according to the information in S°_k, since $S^{\circ}_{n-1}{}' \sqsubseteq^{\circ} S^{\circ}_k$ by Lemma Appendix A.4. Hence, we can conclude that $mul(S^{\circ}_k, l^{\bullet}_n) = \omega$.
Note that we have proved what it was claimed in A.1.
As we have already pointed out, $\alpha_{Sys}(P_{n-1}) \sqsubseteq^{\circ} S^{\circ}_{n-1}{}'$ and therefore also $\alpha_{Sys}(P_{n-1}) \sqsubseteq^{\circ} S^{\circ}_k$. By Theorem 2, there exists a corresponding abstract transition $S^{\circ}_k \xrightarrow{l^{\bullet}_n}_{\circ} S^{\circ\prime}_n$ where $\alpha_{\mathsf{Sys}}(P_n) \sqsubseteq^{\circ} S^{\circ\prime}_n$. This time we could apply the meta-inference rule (1) of Table 12 to prove that $\{l^{\bullet}_1, ..., l^{\bullet}_{n-1}\} \rhd S^{\circ}_k \xrightarrow{l^{\bullet}_n}_{\rhd} \{l^{\bullet}_1, ..., l^{\bullet}_{n-1}, l^{\bullet}_n\} \rhd S^{\circ\prime}_n$. Hence, we obtain that $\{l^{\bullet}_1, ..., l^{\bullet}_{n-1}\} \rhd$

${S^\circ_{n-1}}' \xrightarrow{\{l^\bullet_1,...,l^\bullet_{n-1},l^\bullet_n\}^*}_\triangleright \{l^\bullet_1,...,l^\bullet_{n-1},l^\bullet_n\} \rhd S^{\circ\prime}_n$. By applying the induction hypothesis that assures us the existence of the abstract path $p_1{}^\circ = \emptyset \rhd \alpha_{Sys}(P) \xrightarrow{\{l^\bullet_1,...,l^\bullet_{n-1}\}^*}_\triangleright \{{l_1}^\bullet,\ldots,l^\bullet_{n-1}\} \rhd {S^\circ_{n-1}}'$, we can conclude that the existence of an abstract path $\emptyset \rhd \alpha_{Sys}(P) \xrightarrow{\{l^\bullet_1,...,l^\bullet_{n-1}\}^*}_\triangleright \{{l_1}^\bullet,\ldots,l^\bullet_{n-1}\} \rhd {S^\circ_{n-1}}' \xrightarrow{\{l^\bullet_1,...,l^\bullet_{n-1},l^\bullet_n\}^*}_\triangleright \{l^\bullet_1,...,l^\bullet_{n-1},l^\bullet_n\} \rhd S^{\circ\prime}_n \in \mathcal{T}^\circ(P)$.
We are left to prove that $S^\circ_{n-1} \sqcup^\circ \alpha_{Sys}(P_n) \sqcup^\circ (\emptyset, \{(\lambda^\bullet, occ(p,l_n)) \mid \lambda \in lab(l_n)\}) \sqsubseteq^\circ S^{\circ\prime}_n$
We have that $S^\circ_{n-1} \sqsubseteq^\circ {S^\circ_{n-1}}' \sqsubseteq^\circ S^{\circ\prime}_n$ and $\alpha_{\mathtt{Sys}}(P_n) \sqsubseteq^\circ S^{\circ\prime}_n$. Therefore it remains to show that $(\emptyset, \{(\lambda^\bullet, occ(p,l_n)) \mid \lambda \in lab(l_n)\}) \sqsubseteq^\circ S^{\circ\prime}_n$ where $S^{\circ\prime}_n = (R^{\circ\prime}_n, O^{\circ\prime}_n)$. Since $mul(S^\circ_k, l^\bullet_n) = \omega$ and $S^\circ_k \sqsubseteq^\circ S^{\circ\prime}_n$, by Lemma Appendix A.3, $mul(S^{\circ\prime}_n, l^\bullet_n) \geq mul(S^\circ_k, l^\bullet_n) = \omega$.

□

Based on the previous results, we prove that the abstract semantics of a system *safely approximates* its concrete behaviour, described by the corresponding collecting semantics. As a consequence we derive also the *safety* of the analysis.

Theorem 4 (Safety of the Abstract Semantics)**.** *Given a well labelled system $P \in \mathtt{Sys}$, we have that*

$$\alpha_{coll}(\mathfrak{I}(P)) \sqsubseteq^{C^\circ} \mathfrak{I}^\circ(P).$$

Proof. The collecting semantics $\mathfrak{I}(P)$ returns the set of paths of P, formally $\mathfrak{I}(P) = \mathcal{T}(P)$. Furthermore, by definition of the abstraction function α_{coll}, we have that

$$\alpha_{coll}(P) = \bigsqcup^{C^\circ}_{\{p \mid p \in \mathcal{T}(P)\}} \alpha_{path}(p).$$

The abstract semantics $\mathfrak{I}^\circ(P)$ returns a configuration defined as follows,

$$\mathfrak{I}^\circ(P) = \sqcup^{C^\circ}_{\{p^\circ \mid p^\circ \in \mathcal{T}^\circ(p^\circ)\}} conf(p^\circ)$$

where for any abstract path $p^\circ \in \mathcal{T}^\circ(P)$, $conf(p^\circ) = \sqcup^{C^\circ}_{i\in\{0,\ldots,n\}} T^\circ_i \rhd S^\circ_i$ assuming that $S^\circ_0 = \alpha_{\mathtt{Sys}}(P)$, $T^\circ_0 = \emptyset$ and $p^\circ = T^\circ_0 \rhd S^\circ_0 \xrightarrow{l^\circ_1}_\triangleright T^\circ_1 \rhd S^\circ_1 \ldots T^\circ_{n-1} \rhd S^\circ_{n-1} \xrightarrow{l^\circ_n}_\triangleright T^\circ_n \rhd S^\circ_n$.

Now, by applying Theorem 3, we know that for any path $p \in \mathfrak{I}(P)$ there exists a corresponding abstract path $p^\circ \in \mathcal{T}^\circ(P)$ such that $p^\circ = \emptyset \rhd \alpha_{Sys}(P) \xrightarrow{T^\circ}{}^*_\triangleright T^\circ \rhd S^\circ$ and $S^{\circ\prime} \sqsubseteq S^\circ$ where $\alpha_{path}(p) = T^\circ \rhd S^{\circ\prime}$.

Note that, by definition of $\sqsubseteq^{C^\circ}$, we have $T^\circ \rhd S^{\circ\prime} \sqsubseteq^{C^\circ} T^\circ \rhd S^\circ$. Moreover, by Lemma Appendix A.4, we derive that $conf(p^\circ) = T^\circ \rhd S^\circ$. Hence we can conclude that for any path $p \in \mathfrak{I}(P)$ there exists a corresponding abstract path $p^\circ \in \mathcal{T}^\circ(P)$ such that $\alpha_{path}(p) \sqsubseteq^{C^\circ} conf(p^\circ)$. By definition of $\sqcup^{C^\circ}$, we also obtain that $\alpha_{path}(p) \sqsubseteq^{C^\circ} \mathfrak{I}^\circ(P)$. □

Theorem 5 (Safety)**.** *Given a well labelled system $P \in \mathtt{Sys}$, we have that $\bar{\alpha}(X) \sqsubseteq^\circ \mathcal{A}^\circ(P)$ where $X = \bigcup_{\{p \mid p \in \mathcal{T}(P)\}} reach(p)$ and for any path $p \in \mathcal{T}(P)$, $reach(p) = \bigcup_{i\in\{0,\ldots,n\}}\{P_i\}$ for $p = P_0 \xrightarrow{l_1} P_1 \ldots P_{n-1} \xrightarrow{l_n} P_n$.*

Proof. By definition, we have that $\mathcal{A}^\circ(P) = S^\circ$ where $\mathfrak{I}^\circ(P) = T^\circ \rhd S^\circ$. Moreover, we have that $X = \bigcup_{\{p|p\in\mathcal{T}(P)\}} reach(p)$. We observe that, by definition of $\bar{\alpha}$, we have $\bar{\alpha}(X) = \bigsqcup^\circ_{P'\in X}\alpha_{\mathtt{Sys}}(P')$. Therefore, we have

$$\bar{\alpha}(X) = \bar{\alpha}(\bigcup_{\{p|p\in\mathcal{T}(P)\}} reach(p)) = \bigsqcup^\circ_{\{P'|P'\in reach(p),p\in\mathcal{T}(\mathcal{P})\}}\alpha_{\mathtt{Sys}}(P').$$

Now observe that, $\bigsqcup^\circ_{\{P'|P'\in reach(p),p\in\mathcal{T}(\mathcal{P})\}}\alpha_{\mathtt{Sys}}(P') = \bigsqcup^\circ_{\{p|p\in\mathcal{T}(P)\}}\left(\bigsqcup^\circ_{\{P'|P'\in\in reach(p)\}}\alpha_{\mathtt{Sys}}(P')\right)$. By definition of α_{path}, for any path $p \in \mathcal{T}(\mathcal{P})$ such that $\alpha_{path}(p) = T^\circ_p \rhd S^\circ_p$ then $\alpha_{\mathtt{Sys}}(P')\sqsubseteq^\circ S^\circ_p$, for each $P' \in reach(p)$. As consequence, we have also that $\bigsqcup^\circ_{\{P'|P'\in\in reach(p)\}}\alpha_{\mathtt{Sys}}(P')\sqsubseteq^\circ S^\circ_p$ where $\alpha_{path}(p) = T^\circ_p \rhd S^\circ_p$.
Hence, $\bigsqcup^\circ_{\{p|p\in\mathcal{T}(\mathcal{P})\}}\left(\bigsqcup^\circ_{\{P'|P'\in\in reach(p)\}}\alpha_{\mathtt{Sys}}(P')\right)\sqsubseteq^\circ\bigsqcup^\circ_{\{p|p\in\mathcal{T}(\mathcal{P})\}}S^\circ_p$. Now the result is a direct consequence of Theorem 4.

□

Appendix A.3. Properties of the analysis

The next theorems show that the analysis can be computed by calculating a single abstract path leading to a final configuration.

Lemma Appendix A.5. *Let $S^\circ_1, S^\circ_2 \in \mathcal{S}^\circ$ be abstract states such that $S^\circ_1 = (R^\circ_1, O^\circ_1)$ and $S^\circ_2 = (R^\circ_2, O^\circ_2)$. If $S^\circ_1 \xrightarrow{l^\circ}_\circ S^\circ_3$ and $S^\circ_2 \xrightarrow{l^\circ}_\circ S^\circ_4$, then it must be the case that $S^\circ_3 = (R^\circ_3, O^\circ_1 \cup^+ O^\circ)$ and $S^\circ_4 = (R^\circ_4, O^\circ_2 \cup^+ O^\circ)$.*

Proof. The proof is by cases on the abstract rules. This result is based on the observation that the occurrence counting information introduced by an abstract rule is obtained, by adding the translation of the continuations and (possibly) of the process ρ and one occurrence of the abstract membrane label related to abstract label of the transition. Note that this occurrence counting information does not depend on the occurrence counting information present in the starting state. □

Proposition Appendix A.2. *Let $S^\circ_1 \in \mathcal{S}^\circ$ be an abstract state such that $S^\circ_1 \xrightarrow{l^\circ}_\circ S^\circ_2$. If $S^\circ_1 \sqsubseteq^\circ S^\circ_3$ then there exists a transition $S^\circ_3 \xrightarrow{l^\circ}_\circ S^\circ_4$.*

Proof. Since $S^\circ_1 \sqsubseteq^\circ S^\circ_3$ by definition means that if $S^\circ_i = (R^\circ_i, O^\circ_i)$ (for $i = 1, 3$) then $R^\circ_1 \subseteq R^\circ_3$ and $O^\circ_1 \subseteq_O O^\circ_3$, we have that if the premises of the abstract rule are fulfilled for S°_1 *a fortiori* they are fulfilled for S°_3. □

Theorem 6. *Let $P \in \mathtt{Sys}$ be a well labelled system and $p^\circ_1, p^\circ_2 \in \mathcal{T}^\circ(P)$ be two abstract paths. Given two configurations $T^\circ_1 \rhd S^\circ_1, T^\circ_2 \rhd S^\circ_2 \in C$, if $T^\circ_1 \rhd S^\circ_1$ a final configuration of p°_1 and $T^\circ_2 \rhd S^\circ_2$ is a final configuration of p°_2, then $S^\circ_1 = S^\circ_2$.*

Proof. Let us consider the two abstract paths $p^\circ_1, p^\circ_2 \in \mathcal{T}^\circ(P)$ such that

$$p_1^\circ = \emptyset \rhd S_0^\circ \xrightarrow{l_{1,1}^\circ}_\rhd T_{1,1}^\circ \rhd S_{1,1}^\circ \xrightarrow{l_{1,2}^\circ}_\rhd T_{1,2}^\circ \rhd S_{1,2}^\circ \dots \xrightarrow{l_{1,n-1}^\circ}_\rhd T_{1,n}^\circ \rhd S_{1,n}^\circ,$$

$$p_2^\circ = \emptyset \rhd S_0^\circ \xrightarrow{l_{2,1}^\circ}_\rhd T_{2,1}^\circ \rhd S_{2,1}^\circ \xrightarrow{l_{2,2}^\circ}_\rhd T_{2,2}^\circ \rhd S_{2,2}^\circ \dots \xrightarrow{l_{2,k-1}^\circ}_\rhd T_{2,k}^\circ \rhd S_{2,k}^\circ.$$

where $T_{1,n}^\circ = T_1^\circ$ and $S_{1,n}^\circ = S_1^\circ$, and analogously $T_{2,k}^\circ = T_2^\circ$ and $S_{2,k}^\circ = S_2^\circ$. Moreover, let us assume that $T_{1,n}^\circ \rhd S_{1,n}^\circ$ is a final configuration of p_1° and $T_{2,k}^\circ \rhd S_{2,k}^\circ$ is a final configuration of p_2°. Moreover, let us assume that $S_{1,i}^\circ = (R_{1,i}^\circ, O_{1,i}^\circ)$ for any $i \in \{0, \dots, n\}$ and $S_{2,j}^\circ = (R_{2,j}^\circ, O_{2,j}^\circ)$ for any $j \in \{0, \dots, k\}$.

We first prove that for any $i \in \{0, \dots, n\}$ we have that $S_{1,i}^\circ \sqsubseteq^\circ S_{2,k}^\circ$. The proof is done by induction on i.

case $i = 0$**.** We have to prove that $S_0^\circ \sqsubseteq^\circ S_{2,k}^\circ$. In this case by applying Lemma Appendix A.4 we have

$$S_0^\circ \sqsubseteq^\circ S_{2,1}^\circ \sqsubseteq^\circ S_{2,2}^\circ \sqsubseteq^\circ \dots \sqsubseteq^\circ S_{2,k}^\circ.$$

inductive case $i > 0$**.** In this case we prove that $S_{1,i+1}^\circ \sqsubseteq^\circ S_{2,k}^\circ$ holds assuming that $S_{1,i}^\circ \sqsubseteq^\circ S_{2,k}^\circ$ holds. We observe that the move $T_{1,i}^\circ \rhd S_{1,i}^\circ \xrightarrow{l_{1,i}^\circ}_\rhd T_{1,i+1}^\circ \rhd S_{1,i+1}^\circ$ has been obtained by applying one of the meta-inference in Table 12. Therefore the proof proceeds by cases on the applied meta-inference rule.

- (1). In this case the move $T_{1,i}^\circ \rhd S_{1,i}^\circ \xrightarrow{l_{1,i}^\circ}_\rhd T_{1,i+1}^\circ \rhd S_{1,i+1}^\circ$ has been derived from the corresponding abstract transition $S_{1,i}^\circ \xrightarrow{l_{1,i}^\circ}_\circ S_{1,i+1}^\circ$ and $T_{1,i+1}^\circ = T_{1,i}^\circ \cup \{l_{1,i}^\circ\}$. Moreover, we have that either $mul(S_{1,i}^\circ, l_{1,i}^\circ) = \omega$ or $mul(S_{1,i}^\circ, l_{1,i}^\circ) = 1$ and $l_{1,i}^\circ \notin T_{1,i}^\circ$.
  Note that, by induction hypothesis, we have that $S_{1,i}^\circ \sqsubseteq^\circ S_{2,k}^\circ$, and therefore, by Proposition Appendix A.2, there exists a corresponding abstract transition $S_{2,k}^\circ \xrightarrow{l_{1,i}^\circ}_\circ S_{2,k+1}^\circ$ such that $S_{1,i+1}^\circ \sqsubseteq^\circ S_{2,k+1}^\circ$. Two cases are possible.
  - Suppose that $mul(S_{1,i}^\circ, l_{1,i}^\circ) = \omega$. Since $S_{1,i}^\circ \sqsubseteq^\circ S_{2,k}^\circ$ and $mul(S_{1,i}^\circ, l_{1,i}^\circ) = \omega$, we have also $mul(S_{2,k}^\circ, l_{1,i}^\circ) = \omega$.
    Therefore, in the abstract state $S_{2,k}^\circ$ the conditions for the application of the meta-inference rule (1) in Table 12 are fulfilled. Hence, we derive $T_{2,k}^\circ \rhd S_{2,k}^\circ \xrightarrow{l_{1,i}^\circ}_\rhd T_{2,k+1}^\circ \rhd S_{2,k+1}^\circ$ where $T_{2,k+1}^\circ = T_{2,k}^\circ \cup \{l_{1,i}^\circ\}$ from the corresponding abstract transition $S_{2,k}^\circ \xrightarrow{l_{1,i}^\circ}_\circ S_{2,k+1}^\circ$.
    Now we recall that $T_{2,k}^\circ \rhd S_{2,k}^\circ$ is a final configuration and thus $S_{2,k}^\circ = S_{2,k+1}^\circ$. Therefore $S_{1,i+1}^\circ \sqsubseteq^\circ S_{2,k+1}^\circ$ implies also that $S_{1,i+1}^\circ \sqsubseteq^\circ S_{2,k}^\circ$.
  - Suppose that $mul(S_{1,i}^\circ, l_{1,i}^\circ) = 1$ and $l_{1,i}^\circ \notin T_{1,i}^\circ$. As in the previous case, we have that $S_{1,i}^\circ \sqsubseteq^\circ S_{2,k}^\circ$ and $mul(S_{1,i}^\circ, l_{1,i}^\circ) \leq mul(S_{2,k}^\circ, l_{1,i}^\circ)$.
    We also note that $T_{2,k}^\circ \rhd S_{2,k}^\circ$ is a final configuration and therefore it must be the case that $l_{1,i}^\circ \in T_{2,k}^\circ$.
    Two cases are possible.

* $mul(S^\circ_{2,k}, l^\circ_{1,i}) = \omega$. In this case in the abstract state $S^\circ_{2,k}$ the conditions for the application of the meta-inference rule (1) in Table 12. Thus, the proof is similar to the previous case.

* $mul(S^\circ_{2,k}, l^\circ_{1,i}) = 1$. Since $S^\circ_{2,k} \xrightarrow{l^\circ_{1,i}}_\circ S^\circ_{2,k+1}$ and $l^\circ_{1,i} \in T^\circ_{2,k}$ in this case the conditions for the application of the meta-inference rule (2) in Table 12 are fulfilled. Hence, we derive a corresponding abstract move

$$T^\circ_{2,k} \rhd (R^\circ_{2,k}, O^\circ_{2,k}) \xrightarrow{l^\circ_{1,i}}_\rhd T^\circ_{2,k} \rhd (R^\circ_{2,k+1}, O^\circ_{2,k})$$

where $S^\circ_{2,k} = (R^\circ_{2,k}, O^\circ_{2,k})$ and $S^\circ_{2,k+1} = (R^\circ_{2,k+1}, O^\circ_{2,k+1})$.
We now note that $T^\circ_{2,k} \rhd S^\circ_{2,k}$ is a final configuration and therefore $S^\circ_{2,k} = (R^\circ_{2,k}, O^\circ_{2,k}) = (R^\circ_{2,k+1}, O^\circ_{2,k})$. As a consequence, we derive that $R^\circ_{2,k} = R^\circ_{2,k+1}$.
We also recall that $S^\circ_{1,i} \sqsubseteq^\circ S^\circ_{2,k}$ and $S^\circ_{1,i+1} \sqsubseteq^\circ S^\circ_{2,k+1}$ where $S^\circ_{1,i} = (R^\circ_{1,i}, O^\circ_{1,i})$ and $S^\circ_{1,i+1} = (R^\circ_{1,i+1}, O^\circ_{1,i+1})$.
We aim at proving that $S^\circ_{1,i+1} \sqsubseteq^\circ S^\circ_{2,k}$ For the abstract representation we have $R^\circ_{1,i+1} \subseteq R^\circ_{2,k+1}$ and $R^\circ_{2,k} = R^\circ_{2,k+1}$. Therefore, we have also $R^\circ_{1,i+1} \subseteq R^\circ_{2,k}$.
Hence, we are left to prove that $O^\circ_{1,i+1} \sqsubseteq_O O^\circ_{2,k}$. We now use the fact that $l^\circ_{1,i} \in T^\circ_{2,k}$. This implies that a similar move labelled $l^\circ_{1,i}$ has been realised before. Thus, there exists $j \in \{1, \ldots, k-1\}$ such that $l^\circ_{1,i} = l^\circ_{2,j}$ in the abstract path:

$$\emptyset \rhd S^\circ_0 \xrightarrow{l^\circ_{2,1}}_\rhd T^\circ_{2,1} \rhd S^\circ_{2,1} \xrightarrow{l^\circ_{2,2}}_\rhd T^\circ_{2,2} \rhd S^\circ_{2,2} \ldots \xrightarrow{l^\circ_{2,k-1}}_\rhd T^\circ_{2,k} \rhd S^\circ_{2,k}$$

such that $l^\circ_{2,j} \notin T^\circ_{2,j}$. Since it must be the case that the move $T^\circ_{2,j} \rhd S^\circ_{2,j} \xrightarrow{l^\circ_{2,j}}_\rhd T^\circ_{2,j+1} \rhd S^\circ_{2,j+1}$ by using a meta inference rule (1). Thus in this case we have $S^\circ_{2,j} = (R^\circ_{2,j}, O^\circ_{2,j})$, $S^\circ_{2,j+1} = (R^\circ_{2,j+1}, O^\circ_{2,j+1})$ and $O^\circ_{2,j+1} = O^\circ_{2,j} \cup^+ O'^\circ$.
Since $O^\circ_{2,j+1} \sqsubseteq_O O^\circ_{2,k}$ we have also that $O'^\circ \sqsubseteq_O O^\circ_{2,k}$. Then we can always write $O^\circ_{2,k}$ as follows $O^\circ_{2,k} = \overline{O^\circ_{2,k}} \cup^+ O'^\circ$ where $\overline{O^\circ_{2,k}}$ is obtained by summing up all the subsequent occurrence counting information derived by the moves that lead to $O^\circ_{2,k}$, by starting from step $j+1$.
Note that, by Lemma Appendix A.5, we also have that $O^\circ_{1,i+1} = O^\circ_{1,i} \cup^+ O'^\circ$. In order to prove that $O^\circ_{1,i+1} \sqsubseteq_O O^\circ_{2,k}$, we prove by contradiction that $O^\circ_{1,i} \sqsubseteq_O \overline{O^\circ_{2,k}}$, by knowing that, by inductive hypothesis, $O^\circ_{1,i} \sqsubseteq_O O^\circ_{2,k} = \overline{O^\circ_{2,k}} \cup^+ O'^\circ$.
Assume, by contradiction, that there exists a (process or membrane) label that is in the domain of $O^\circ_{1,i}$ but that it is not correctly approximated by the occurrence counting information in $\overline{O^\circ_{2,k}}$. Remember that $\overline{O^\circ_{2,k}}$ collects the sum of all occurrence counting pieces of information introduced by all moves in $T^\circ_{2,k}$ except for effects of the move labelled $l^\circ_{1,i}$ (component O'°) which we are sure was summed up only once.
We have the following cases for such abstract process/membrane label that we will call λ°.

· λ° has occurrence 1 in $O^\circ_{1,i}$ but is not present in the domain of $\overline{O^\circ_{2,k}}$. Since $O^\circ_{1,i}$ did not perform any move labelled $l^\circ_{1,i}$ before (this is the first time), this means that there exists another previous move whose effects were to introduce also one occurrence of label λ° in some step $O^\circ_{1,h}$, with $h \leq i$. The label of this move belongs to $T^\circ_{1,n}$. Since $T^\circ_{1,n} \subseteq T^\circ_{2,k}$ and such move was not labelled $l^\circ_{1,i}$, we can conclude that $\overline{O^\circ}_{2,k}$ contains the effects of such move that were summed up to the occurrence counting information related to the other moves. Therefore, λ° has occurrence at least 1 in $\overline{O^\circ_{2,k}}$. This leads to a contradiction.

· λ° has occurrence ω in $O^\circ_{1,i}$ but has occurrence 0 or 1 in $\overline{O^\circ_{2,k}}$. Now we have two further cases.

• There may exist two moves with different abstract labels that introduced two occurrence of the label λ° in $O^\circ_{1,i}$. In this case, since $T^\circ_{1,n} \subseteq T^\circ_{2,k}$ and such move were surely not labelled $l^\circ_{1,i}$, we can conclude that $\overline{O^\circ_{2,k}}$ contains the effects of such moves that were summed up to the occurrence counting information related to the other moves. Therefore, λ° has occurrence ω in $\overline{O^\circ_{2,k}}$. This leads to a contradiction.

• It was the same move (repeated twice) that introduced two occurrences of the label λ° in $O^\circ_{1,i}$. In this case, by definition, the multiplicity of this move would be ω, otherwise, the second move could not update the occurrence counting information. Note that the multiplicity of a transition label depends on the multiplicity of the process label involved in the move. Therefore, we can conclude that the multiplicity of process labels involved in the move, let us call them ν° and μ°, was ω. We now want to prove that the multiplicity of labels ν° and μ° in $\overline{O^\circ_{2,k}}$ were ω. Remember that $O^\circ_{1,i} \sqsubseteq_O O^\circ_{2,k} = \overline{O^\circ_{2,k}} \cup^+ O'^\circ$, hence we are sure that multiplicity of labels ν° and μ° is ω in $O^\circ_{2,k}$. Therefore also this derivation performed the move (repeated twice) that introduced two occurrence of label λ° in $O^\circ_{2,k}$. Since this move is different from $l^\circ_{1,i}$, we can be sure that the two occurrences of the label λ° are in the $\overline{O^\circ_{2,k}}$ component. This leads to a contradiction.

By summarising, we proved that $O^\circ_{1,i} \sqsubseteq_O \overline{O^\circ_{2,k}}$. Together with $O^\circ_{1,i+1} = O^\circ_{1,i} \cup^+ O'^\circ$ and $O^\circ_{2,k} = \overline{O^\circ_{2,k}} \cup^+ O'^\circ$. This allows us to conclude that $O^\circ_{1,i+1} \sqsubseteq_O O^\circ_{2,k}$. Hence, we conclude that $R^\circ_{1,i+1} \subseteq R^\circ_{2,k}$ and $O^\circ_{1,i+1} \sqsubseteq_O O^\circ_{2,k}$ and therefore $S^\circ_{1,j+1} \sqsubseteq^\circ S^\circ_{2,k}$.

- (2). In this case we have $mul(S^\circ_{1,i}, l^\circ_{1,i}) = 1$ and $l^\circ_{1,i} \in T^\circ_{1,i}$. Moreover the move $T^\circ_{1,i} \rhd S^\circ_{1,i} \xrightarrow{l^\circ_{1,i}}_\rhd T^\circ_{1,i+1} \rhd S^\circ_{1,i+1}$ has been derived from the corresponding abstract transition $S^\circ_{1,i} \xrightarrow{l^\circ_{1,i}}_\circ \overline{S^\circ}_{1,i+1}$ where $\overline{S^\circ}_{1,i+1} = (R^\circ_{1,i+1}, \overline{O^\circ}_{1,i+1})$, $S^\circ{}_{1,i+1} = (R^\circ_{1,i+1}, O^\circ_{1,i})$ and $T^\circ_{1,i+1} = T^\circ_{1,i}$. Note that, by induction hypothesis, we have that $S^\circ_{1,i} \sqsubseteq^\circ S^\circ_{2,k}$, and therefore, by Proposition Appendix A.2, there exists a corresponding abstract transition $S^\circ_{2,k} \xrightarrow{l^\circ_{1,i}}_\circ S^\circ_{2,k+1}$ such that $\overline{S^\circ}_{1,i+1} \sqsubseteq^\circ S^\circ_{2,k+1}$. Moreover, $\overline{S^\circ}_{1,i+1} = (R^\circ_{1,i+1}, \overline{O^\circ}_{1,i+1})$ implies by definition of

$\sqsubseteq^\circ$ that $R^\circ_{1,i+1} \subseteq R^\circ_{2,k+1}$ and $\overline{O^\circ}_{1,i+1} \sqsubseteq_O O^\circ_{2,k+1}$.
As in the previous case, we have that $S^\circ_{1,i} \sqsubseteq^\circ S^\circ_{2,k}$ and $mul(S^\circ_{1,i}, l^\circ_{1,i}) \leq mul(S^\circ_{2,k}, l^\circ_{1,i})$. Furthermore, since $T^\circ_{2,k} \rhd S^\circ_{2,k}$ is a final configuration, it must be the case that $l^\circ_{1,i} \in T^\circ_{2,k}$. Two cases are possible.

- Suppose that $mul(S^\circ_{2,k}, l^\circ_{1,i}) = 1$. Since $S^\circ_{2,k} \xrightarrow{l^\circ_{1,i}}_\circ S^\circ_{2,k+1}$ and $l^\circ_{1,i} \in T^\circ_{2,k}$ the conditions for the application of the meta-inference rule (2) are fulfilled. Hence, we derive a move $T^\circ_{2,k} \rhd S^\circ_{2,k} \xrightarrow{l^\circ_{1,i}}_\rhd T^\circ_{2,k} \rhd \overline{S^\circ}_{2,k+1}$ where $\overline{S^\circ}_{2,k+1} = (R^\circ_{2,k+1}, O^\circ_{2,k})$.
  We now use the fact that $T^\circ_{2,k} \rhd S^\circ_{2,k}$ is a final configuration and thus $\overline{S^\circ}_{2,k+1} = S^\circ_{2,k}$. Therefore, we have that $R^\circ_{2,k+1} = R^\circ_{2,k}$.
  Furthermore, we have that $S^\circ{}_{1,i+1} = (R^\circ_{1,i+1}, O^\circ_{1,i})$ where $R^\circ_{1,i+1} \subseteq R^\circ_{2,k+1}$ and $R^\circ_{2,k+1} = R^\circ_{2,k}$. Moreover, we have also that $S^\circ_{1,i} \sqsubseteq^\circ S^\circ_{2,k}$ and thus $O^\circ_{1,i} \sqsubseteq_O O^\circ_{2,k}$. Hence we conclude that $S^\circ_{1,i+1} \sqsubseteq^\circ S^\circ_{2,k}$.
- Suppose that $mul(S^\circ_{2,k}, l^\circ_{1,i}) = \omega$. Since $S^\circ_{2,k} \xrightarrow{l^\circ_{1,i}}_\circ S^\circ_{2,k+1}$, in this case the conditions for the application of the meta-inference rule (1) are fulfilled. Hence, we derive a corresponding move $T^\circ_{2,k} \rhd S^\circ_{2,k} \xrightarrow{l^\circ_{1,i}}_\rhd T^\circ_{2,k} \rhd S^\circ{}_{2,k+1}$.
  As in the previous case, we use the fact that $T^\circ_{2,k} \rhd S^\circ_{2,k}$ is a final configuration and thus $S^\circ{}_{2,k+1} = S^\circ_{2,k}$. Therefore, we have that $R^\circ_{2,k+1} = R^\circ_{2,k}$ and $O^\circ_{2,k+1} = O^\circ_{2,k}$. Moreover, we have that $R^\circ_{1,i+1} \subseteq R^\circ_{2,k+1}$, where $R^\circ_{2,k+1} = R^\circ_{2,k}$. Furthermore, we have $S^\circ_{1,i} \sqsubseteq^\circ S^\circ_{2,k}$ and thus $O^\circ_{1,i} \sqsubseteq_O O^\circ_{2,k}$. Therefore, we conclude that $S^\circ_{1,i+1} \sqsubseteq^\circ S^\circ_{2,k}$ where $S^\circ{}_{1,i+1} = (R^\circ_{1,i+1}, O^\circ_{1,i})$.

It should be clear that, by applying similar arguments, we can also prove that $S^\circ_{2,j} \sqsubseteq^\circ S^\circ_{1,n}$ for any $j \in \{1, \ldots, k\}$. Hence, we conclude that $S^\circ_{1,n} = S^\circ_{2,k}$.

□

Corollary 1. *Let $P \in \mathtt{Sys}$ be a well labelled system. We have that $\mathcal{A}^\circ(P) = S^\circ$, where $T^\circ \rhd S^\circ \in C$ is a final configuration of an abstract path p° such that $p^\circ \in \mathcal{T}^\circ(P)$.*

Proof. We recall that, by Definition 19, we have that $\mathcal{A}^\circ(P) = S^\circ$ where $\mathfrak{I}^\circ(P) = T^\circ \rhd S^\circ$. Moreover, by Definition 18, $\mathfrak{I}^\circ(P) = \sqcup^{C^\circ}_{\{p^\circ | p^\circ \in \mathcal{T}^\circ(p^\circ)\}} conf(p^\circ) = T^\circ \rhd S^\circ$.

Let us consider an abstract path $p^\circ \in \mathcal{T}^\circ(p^\circ)$ such that

$$p^\circ = T^\circ_0 \rhd S^\circ_0 \xrightarrow{l^\circ_1}_\rhd T^\circ_1 \rhd S^\circ_1 \ldots T^\circ_{n-1} \rhd S^\circ_{n-1} \xrightarrow{l^\circ_n}_\rhd T^\circ_n \rhd S^\circ_n,$$

where $S^\circ_0 = \alpha_{\mathtt{Sys}}(P)$, $T^\circ_0 = \emptyset$ and such that $T^\circ_n \rhd S^\circ_n$ is a final configuration. By applying Lemma Appendix A.4, we derive that

$$T^\circ_0 \rhd S^\circ_0 \sqsubseteq^{C^\circ} T^\circ_1 \rhd S^\circ_1 \sqsubseteq^{C^\circ} T^\circ_2 \rhd S^\circ_2 \sqsubseteq^{C^\circ} \ldots \sqsubseteq^{C^\circ} T^\circ_n \rhd S^\circ_n.$$

By definition of $\sqsubseteq^{C^\circ}$ this implies that

$$S^\circ_0 \sqsubseteq^\circ S^\circ_1 \sqsubseteq^\circ S^\circ_2 \sqsubseteq^\circ \ldots \sqsubseteq^\circ S^\circ_n.$$

As a consequence we have that $\mathcal{A}^\circ(P) = \sqcup^\circ\{S_n^\circ\}$ such that $T_n^\circ \rhd S_n^\circ$ is a final configuration of an abstract path p° such that $p^\circ \in \mathcal{T}^\circ(P)$. However, by applying Theorem 6, any abstract path p° with $p^\circ \in \mathcal{T}^\circ(P)$ has the same abstract state S_n° in the final configuration $T_n^\circ \rhd S_n^\circ$. Therefore we can conclude that $\mathcal{A}^\circ(P) = S_n^\circ$ for some abstract path p° with $p^\circ \in \mathcal{T}^\circ(P)$ such that $T_n^\circ \rhd S_n^\circ$ is a final configuration. □