

Hiding Resources that Can Fail*

Anna Philippou¹, Oleg Sokolsky², Insup Lee²,
Rance Cleaveland³, and Scott A. Smolka³

¹University of Cyprus. annap@cs.ucy.ac.cy

²University of Pennsylvania, USA. {sokolsky,lee}@cis.upenn.edu

³SUNY at Stony Brook, USA. {rance, sas}@cs.sunysb.edu

Abstract

In earlier work, we presented a process algebra, PACSR, that uses resource failures to capture probabilistic behavior in reactive systems. In this paper, we explore the effects of resource failures in the situation where resources may be hidden from the environment. For this purpose, we introduce a subset of PACSR, called “PACSR-lite,” that allows us to isolate the issues surrounding resource hiding, and we provide a sound and complete axiomatization of strong bisimulation for this fragment.

1 Introduction

The real-time process algebra ACSR [6] features a notion of *resource-dependent* actions. A process needs to have access to a set of resources specified in an action, before it can proceed with the action. Recently, in the context of the process algebra PACSR [9], we extended the ACSR framework with the possibility of resource failures, which happen with a given probability.

Previous work on extending process algebra with probability information, such as [4, 12, 1, 3, 5, 11] typically associates probabilities with process terms. An ad-

vantage of associating probabilities with resources, rather than with process terms, is that the specification of a process does not involve probabilities directly. Failure probabilities of individual resources are defined separately and are used only during analysis. This makes specifications simpler and ensures a more systematic way of applying probabilistic information. In addition, this approach allows one to explore the impact of changing probabilities of failures on the overall behavior, without changing the specification.

In this paper, we explore the effects of resource failures in the situation where resources may be hidden from the observer (i.e., private to a process). Specifically, we present PACSR-lite, a fragment of PACSR that allows us to isolate the issues surrounding resource hiding, and present a sound and complete axiomatization of strong bisimulation equivalence for this fragment. An axiomatization of full PACSR is currently being undertaken and will be published in a future paper.

2 The Syntax of PACSR-lite

PACSR-lite is a subset of the probabilistic process algebra PACSR [9]. The actions of PACSR-lite specify access to a (possibly empty) set of resources that the process requires to perform the action. Moreover, each resource has an associated failure

*Research supported in part by grants AFOSR F49620-95-1-0508, ARO DAAH04-95-1-0092, NSF CCR-9988409, NSF CCR-9619910, and ONR N00014-97-1-0505 (MURI).

probability. Resources can be *hidden* in that their identity is not visible to the environment, but their failures can be observed.

Resources and actions. We assume that a system contains a finite set of serially reusable resources drawn from the infinite set Res . We write \overline{Res} for the set that contains, for each $r \in Res$, an element \overline{r} , representing the *failed* resource r , and R for $Res \cup \overline{Res}$. An action is drawn from the domain $P(R)$ with the restriction that each resource is represented at most once. For example, the singleton action $\{r\}$ denotes the use of resource r . This action cannot happen if r has failed. On the other hand, action $\{\overline{r}\}$ takes place given that resource r has failed. A notation for failed resources is useful for specifying recovery from failures. Action \emptyset represents idling, since no resource is consumed. We let Act denote the domain of actions and A, B range over Act .

Each resource has an associated probability specifying the rate at which the resource may fail. For all $r \in Res$ we denote by $p(r) \in [0, 1]$ the probability of resource r being up, while $p(\overline{r}) = 1 - p(r)$ denotes the probability of r failing. We assume an infinite number of resources for each probability failure in $[0, 1]$. For example, consider the action $\{cpu\}$, where resource cpu has probability of failure $1/3$, i.e. $p(cpu) = 2/3$. Then, $\{cpu\}$ may occur with probability $2/3$ and fails with probability $1/3$.

Processes. The set $Proc$ of PACSR-lite processes, ranged over by P and Q , is given by:

$$P ::= \text{NIL} \mid A : P \mid P + P \mid P \setminus I$$

The process NIL represents the inactive process. $A : P$ executes a resource-consuming action and proceeds to process P . The process $P + Q$ represents a non-deterministic choice between the two summands. $P \setminus I$ hides the identity of resources in I so that they are not visible on the interface with the environment. The formal semantics of processes is given in the next sec-

tion. The full process algebra, PACSR, additionally contains the usual constructs for recursion, parallel composition, restriction, etc.

The operator $P \setminus I$ binds all free occurrences of the resources in I in P . This binder gives rise to the sets of *free* and *bound* resources of a process P , denoted by $\text{fr}(P)$ and $\text{br}(P)$ respectively. We write $\text{res}(P)$ for the set of all resources of P .

Let $Z = \{r_1, \dots, r_n\} \subseteq R$. Then $p(Z) = \prod_{1 \leq i \leq n} p(r_i)$; $\mathcal{W}(Z) = \{Z' \subseteq Z \cup \overline{Z} \mid r \in Z' \text{ iff } \overline{r} \notin Z'\}$; and $\text{res}(Z) = \{r \in Res \mid r \in Z \text{ or } \overline{r} \in Z\}$.

Thus $\mathcal{W}(Z)$ denotes the set of all possible worlds involving the set of resources Z , that is, the set of all combinations of the resources in Z being up or down. For example, $\mathcal{W}(\{r_1, \overline{r_2}\}) = \{\{\overline{r_1}, \overline{r_2}\}, \{\overline{r_1}, r_2\}, \{r_1, \overline{r_2}\}, \{r_1, r_2\}\}$. Note that $p(\emptyset) = 1$ and $\mathcal{W}(\emptyset) = \{\emptyset\}$. Finally, the function $\text{imr}(P)$, defined inductively below, associates each process with the set of resources on which its behavior immediately depends:

$$\begin{aligned} \text{imr}(\text{NIL}) &= \emptyset \\ \text{imr}(P_1 + P_2) &= \text{imr}(P_1) \cup \text{imr}(P_2) \\ \text{imr}(A : P) &= \text{res}(A) \\ \text{imr}(P \setminus I) &= \text{imr}(P) \end{aligned}$$

3 Operational Semantics

A *configuration* is a pair of the form $(P, W) \in Proc \times 2^R$, representing a process P in world W . A world captures the state (up or down) of resources relevant to P . We write S for the set of configurations. The semantics of PACSR-lite is given in terms of a labeled transition system whose states are configurations and whose transitions are either probabilistic (labeled by a probability) or nondeterministic (labeled by an action). The idea is that, for a process P , computation begins in the *initial configuration* (P, \emptyset) . A probabilistic transition is then performed to determine the status of resources which are immediately relevant for execution (as specified by $\text{imr}(P)$) but for which there is

no knowledge in the configuration's world. The status of a resource does not change until the next action-labeled transition occurs; moreover, actions erase all previous knowledge of the configuration's world (see law (Act)). Nondeterministic transitions are possible from configurations that contain all necessary knowledge regarding the state of resources.

With this view of computation in mind, we partition \mathbf{S} as follows:

$$S_n = \{(P, W) \in \mathbf{S} \mid \text{imr}(P) - \text{res}(W) = \emptyset\},$$

the set of nondeterministic configurations, and

$$S_p = \{(P, W) \in \mathbf{S} \mid \text{imr}(P) - \text{res}(W) \neq \emptyset\},$$

the set of probabilistic configurations.

The operational semantics of PACSR-lite processes is given as a combination of two labeled transition systems: $\dashv\vdash \subset S_p \times [0, 1] \times S_n$ is the probabilistic transition relation and $\longrightarrow \subset S_n \times \text{Act} \times \mathbf{S}$ is the nondeterministic transition relation. We write elements of $\dashv\vdash$ as $(P, W) \dashv\vdash^p (P', W')$ and elements of \longrightarrow as $(P, W) \xrightarrow{\alpha} (P', W')$.

The probabilistic transition relation is given by the rule (PROB) in Table 1. Note that configuration (P, W) evolves into $(P, W \cup Z_2)$ which is, by definition, a nondeterministic configuration. Further, it can be shown that for all $s \in S_p$, $\sum \{\!| p \mid (s, p, s') \in \dashv\vdash \!\} = 1$, where $\{\!|$ and $\!\}$ are multiset brackets and the summation over the empty multiset is 1.

The nondeterministic transition relation is given by rules (Act), (Sum), and (Hide) of Table 1. The symmetric version of rule (Sum) has been omitted. Note that in rule (Act), the occurrence of an action A re-initializes the world to \emptyset . It can be shown that the semantics of PACSR-lite processes define alternating transition systems, that is, transition systems where nondeterministic and probabilistic states alternate [4].

For example, consider process $\{r_1, \overline{r_2}\} : P$, which, in a world where resource r_1 is up and r_2 is down, may evolve to P . Let $\text{p}(r_1) = \text{p}(r_2) = 0.5$. Then, by

(PROB), $(\{r_1, \overline{r_2}\} : P, \emptyset) \xrightarrow{0.25} (\{r_1, \overline{r_2}\} : P, W)$, for each $W \in \mathcal{W}(\{r_1, r_2\})$, and, by (Act), $(\{r_1, \overline{r_2}\} : P, \{r_1, \overline{r_2}\}) \xrightarrow{\{r_1, \overline{r_2}\}} (P, \emptyset)$, whereas $(\{r_1, \overline{r_2}\} : P, \{r_1, r_2\})$, $(\{r_1, \overline{r_2}\} : P, \{\overline{r_1}, r_2\})$, and $(\{r_1, \overline{r_2}\} : P, \{\overline{r_1}, \overline{r_2}\})$ have no transitions.

4 Strong Bisimulation

We introduce the notion of (strong) bisimulation [8] for PACSR-lite processes. It captures formally the notion that equivalent systems exhibit the same behavior, including probabilistic behavior, at their interfaces with the environment. Our definition of probabilistic strong bisimulation is closely related to those studied by [4, 10].

Definition 4.1 For $s \in \mathbf{S}$ and $\mathcal{M} \subseteq \mathbf{S}$, we define $\mu(s, \mathcal{M}) = \sum_{s' \in \mathcal{M}} \{\!| p \mid (s, p, s') \in \dashv\vdash \!\}$ \square

Thus, $\mu(s, \mathcal{M})$ denotes the probability that s may perform a probabilistic transition to a configuration in \mathcal{M} .

Definition 4.2 An equivalence relation $\mathcal{R} \subseteq \mathbf{S} \times \mathbf{S}$ is a *strong bisimulation* if, whenever PRQ (1) for all $\alpha \in \text{Act}$, if $P, Q \in S_n$ and $P \xrightarrow{\alpha} P'$ then $Q \xrightarrow{\alpha} Q'$ and $P'\mathcal{R}Q'$; (2) for all $\mathcal{M} \in \mathbf{S}/\mathcal{R}$, if $P, Q \in S_p$, $\mu(P, \mathcal{M}) = \mu(Q, \mathcal{M})$.

Two configurations P and Q are *strong bisimulation equivalent*, written $P \sim Q$, if there exists a strong bisimulation \mathcal{R} such that PRQ . \square

Consequently, two configurations are related by a strong bisimulation \mathcal{R} , if they can reach all equivalence classes of the relation with the same probability and they can simulate each other's behavior.

It is easy to show that there is a largest strong bisimulation which we denote as \sim . We consider two PACSR-lite processes bisimilar when their initial configurations are bisimilar.

$$\begin{array}{l}
(\text{PROB}) \quad \frac{(P, W) \in S_p, Z_1 = \text{imr}(P) - \text{res}(W), Z_2 \in \mathcal{W}(Z_1)}{(P, W) \xrightarrow{\mathfrak{p}(Z_2)} (P, W \cup Z_2)} \\
(\text{Act}) \quad (A : P, W) \xrightarrow{A} (P, \emptyset), \text{ if } A \subseteq W \quad (\text{Sum}) \quad \frac{(P_1, W) \xrightarrow{\alpha} (P, W')}{(P_1 + P_2, W) \xrightarrow{\alpha} (P, W')} \\
(\text{Hide}) \quad \frac{(P, W) \xrightarrow{A} (P', W'), A' = A - I}{(P \setminus I, W) \xrightarrow{A'} (P' \setminus I, W')}
\end{array}$$

Table 1: The probabilistic and nondeterministic transition relations

5 The Laws

Tables 2 and 3 contain our axiomatization of strong bisimulation for PACSR-lite, which we refer to as \mathcal{A} . We shall subsequently show that \mathcal{A} is a sound and complete axiomatization of strong bisimulation. In the sequel, we will use the equality symbol “=” when two processes can be shown to be equivalent using \mathcal{A} .

Law Hide(2) describes how the hiding operator distributes over summation. In order to push a summation outside a hiding operator, we must ensure that no pair of summands share any bound resources, otherwise a resource that was shared in the two summands of the left-hand side process will become two different resources on the right-hand side, which causes a problem with the probabilistic behavior of the process, as will be shown in the example below. Law Down states that a process which is only willing to engage in an action involving a failed resource is in fact a failed process. Law Rename establishes the equivalence of processes that only differ by a change of bound resources that have the same probability of failure.

The laws of Table 3 are central for the completeness of the strong bisimulation characterization. Law Extend, allows us to rewrite a summation of prefixes by enriching each summand with information about the state of a new hidden resource, thus replacing each process $A : P$ with the summation $(A \cup \{r\}) : P + (A \cup \{\bar{r}\}) : P$, assuming $r \notin \text{res}(A)$. Law Standard, provides a stan-

dard form for a summation of processes by identifying probabilistic branches that have the same observable behavior, although possibly using different sets of hidden resources, and grouping these into a single branch by using a new set of hidden resources. This set of new resources are used to create a number of mutually-exclusive worlds, each of which will be used to represent different behaviors of the left-hand side process. The probabilities of each of the required resources can be obtained by solving the set of equations $\mathfrak{p}(C_i) = \sum_{j=1}^{J_i} \mathfrak{p}(B_j)$, for all i . It can be easily shown that a unique solution exists to this set of equations with each of the solutions in $[0, 1]$, as required.

We illustrate the intuition behind laws Standard with two examples. First, let $I = 1$ and $J_1 = K_1 = 2$. Then, assuming all resources are hidden and omitting the index i , the left-hand side process of both laws Standard is $P = B_1:P_1 + B_1:P_2 + B_2:P_1 + B_2:P_2$. Figure 1,a) gives the transitions for (P, \emptyset) . The law Standard(1) allows us to merge the probabilistic branches that lead to the same processes, arriving at a bisimilar process $Q = C:P_1 + C:P_2$, as illustrated in Figure 1,b). In this case, $C = \{\rho\}$ with a matching probability. For a more detailed example, consider the process $P = (\{r_1, r_2\} : P_1 + \{r_1, r_2\} : P_2 + \{r_1, \bar{r}_2\} : P_3 + \{\bar{r}_1, r_2\} : P_3) \setminus \{r_1, r_2\}$. If both resources r_1, r_2 are available, P can silently evolve into either P_1 or P_2 . If either one of the resources is available, P can evolve into P_3 . Otherwise, P is deadlocked. We need to group together the cases where P

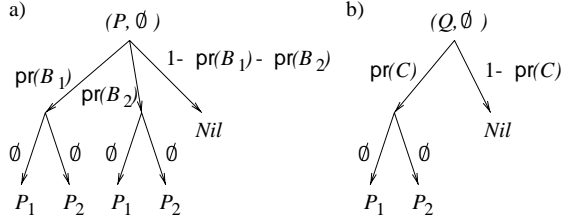


Figure 1: Law Standard(1)

evolves into P_3 into a single term, introducing new hidden resources in order to match the probability of arriving at P_3 . Applying the law Standard 1, we obtain the process $Q = (\{\rho_1, \rho_2\} : P_1 + \{\rho_1, \rho_2\} : P_2 + \{\overline{\rho}_1, \rho_2\} : P_3) \setminus \{r_1, r_2, \rho_1, \rho_2\}$, where the failure probabilities are assigned to ρ_1 and ρ_2 is such a way that $p(\rho_1) \cdot p(\rho_2) = p(r_1) \cdot p(r_2)$ and $p(\overline{\rho}_1) \cdot p(\rho_2) = p(r_1) \cdot p(\overline{r}_2) + p(\overline{r}_1) \cdot p(r_2)$.

6 Soundness

In this section we establish the soundness of the laws, that is, we prove that the equations respect strong bisimulation. To do this, we rely on the result of [2], which provides a sound axiomatization for a non-probabilistic process algebra ACSR. We note that every PACSR-lite term is also an ACSR term, and that all Choice and Hide laws in Table 2 hold for ACSR as well. We refer to these laws as \mathcal{A}' . Their soundness with respect to probabilistic strong bisimulation can be derived as a consequence of this fact. In the sequel we will use $P =^l Q$ to denote that P and Q can be shown to be equivalent by using laws \mathcal{A}' , \sim^l to refer to strong nonprobabilistic bisimulation, and \longrightarrow^l to refer to the transition relation of ACSR, as defined in [2]. We introduce the notion of compatibility between PACSR-lite processes defined as follows.

Definition 6.1 An equivalence relation $\mathcal{R} \subseteq \sim^l$ is a *compatibility relation* if, whenever PRQ , 1) $\text{imr}(P) = \text{imr}(Q)$, and 2) for all $\alpha \in \text{Act}$, if $P \xrightarrow{\alpha} P'$ then $Q \xrightarrow{\alpha} Q'$ and $P'\mathcal{R}Q'$. Two states P and Q are *compati-*

ble, if there exists a compatibility relation \mathcal{R} such that PRQ . \square

Thus two processes are compatible if they are nonprobabilistically strongly bisimilar, have the same immediate resources, and preserve these properties under reduction. A useful fact that we will be using is that if $P =^l Q$ then P and Q are compatible to each other. This can be easily proved by induction on the size of the $=^l$ -proof.

The following theorem achieves that, if two PACSR processes can be shown to be equivalent by using laws in \mathcal{A}' , then they are bisimilar.

Theorem 6.2 If $P =^l Q$ then $P \sim Q$.

PROOF: Let

$$\mathcal{R} = \{((P, \emptyset), (Q, \emptyset)) \mid P, Q \text{ are compatible}\} \cup \{((P, W), (Q, W)) \mid P, Q \text{ are compatible, } (P, W), (Q, W) \in S_n\}$$

The proof involves showing that $\mathcal{R} \subseteq \sim$. This follows easily given the compatibility of the processes in the two types of configurations. Then, since $P =^l Q$ implies that P and Q are compatible, we may conclude that $P \sim Q$ as required. \square

It remains to establish the soundness of laws Rename, Down, and the laws of Table 3.

Lemma 6.3 If P and Q are related by one of the laws Rename, Down, Extend, Standard(1), or Standard(2), then $P \sim Q$.

PROOF: The proof of this result follows easily from the definitions of strong bisimulation. We consider the two most interesting laws:

Extend Let $Q \equiv (\sum_{i \in I} A_i : P_i) \setminus V$ and $R \equiv (\sum_{j \in I, r \notin A_j} (A_j \cup \{r\}) : P_j + (A_j \cup \{\overline{r}\}) : P_j) + \sum_{j \in I, r \in A_j} A_j : P_j \setminus V, r \in V$. Clearly, $\text{imr}(Q) = \text{imr}(R)$, thus $(Q, \emptyset) \xrightarrow{p} (Q, W)$ iff $(R, \emptyset) \xrightarrow{p} (R, W)$. Furthermore, for each world W , Q has exactly the following transitions: (1) if $r \notin A_i$ and $A_i \cup \{r\} \subseteq W$,

then $(Q, W) \xrightarrow{A_i - V} (P_i, W)$, (2) if $r \notin A_i$ and $A_i \cup \{\bar{r}\} \subseteq W$, then $(Q, W) \xrightarrow{A_i - V} (P_i, W)$, and (3) if $r \in A_i$ and $A_i \subseteq W$, then $(Q, W) \xrightarrow{A_i - V} (P_i, W)$. (R, W) has exactly the same transitions. Hence the result follows.

Standard(1) Here we prove the soundness of a reduced version of this law where the processes have no free resources and $K_i = 1$, for all i . This allows us to concentrate on the essence of the law, that is the effect or renaming bound resources. The full result follows easily given that the processes on each side of the equation have the same behavior under each valuation of the bound resources. Let $Q \equiv (\sum_{i=1}^I \sum_{j=1}^{J_i} B_{ij} : P_i) \setminus V$ and $R \equiv (\sum_{i=1}^I C_i : P_i) \setminus V \cup \{\rho_1, \dots, \rho_I\}$, satisfying the conditions of the lemma. We observe that both processes can evolve into exactly one of the equivalence classes $\mathcal{M}_i = \{\emptyset : P_i\}_{\sim}$, $1 \leq i \leq I$ and the equivalence class $\{\text{NIL}\}_{\sim}$. Then, $\mu(Q, \mathcal{M}_i) = \sum_{j=1}^{J_i} p(B_{ij}) = p(C_i) = \mu(Q, \mathcal{M}_i)$. Therefore, $Q \sim R$. \square

7 Completeness of the axiomatization

In this section we will prove that the laws given in Tables 2 and 3 are complete for PACSR-lite. The completeness proof is carried out in the standard way: First, we develop a kind of standard set of equations and show that it is satisfied by any PACSR-lite process. We then show that two bisimilar processes can be shown to satisfy a common set of standard equations and, finally, we appeal to the result that such sets of equations have a unique solution up to bisimulation.

7.1 Standard Set of Equations

In this section we prove that any $P \in \text{Pr}$ provably satisfies a particular set of equations.

Let \tilde{X} be a set of variables and \tilde{H} be terms. We say that a process P provably satisfies a set of equations $S : \tilde{X} = \tilde{H}$ if there is a set of terms $\tilde{P} = \{P_1, P_2, \dots, P_n\}$ such that $\tilde{P} = \tilde{H}[\tilde{P}/\tilde{X}]$ and $P = P_1$.

A set of equations S is said to be *standard* if $X_1 = X_2 \setminus V$ and for all $i \geq 2$

$$X_i = \sum_{j \in J_i} \sum_{k \in K_i} A_{jk} \cup B_j : X_{jk},$$

where

1. $\bigcup_{j \in J_i, k \in K_i} A_{jk} \cap V = \emptyset$, $\bigcup_{j \in J} B_j \subset V$, $\text{res}(B_{j_l}) = \text{res}(B_{j_m})$ for all $j_l, j_m \in J_i$ and $B_{j_l} \neq B_{j_m}$ for all $j_l \neq j_m$,
2. $V \cap \text{res}(X_2) = V$.

The first point to note in this definition is the treatment of the hide operator. This construct cannot be eliminated from standard sets of equations: the probabilistic information that accompanies a hidden resource is necessary for defining the semantics of a process and it can not be encoded by any other means. (Note that in ACSR this is possible and standard sets of equations can be given as unrestricted summations.) However, the Hide-laws allow us, after possibly renaming some resources, to push the hiding operators *outwards* in a given process. Thus in a standard set of equations, the top equation consists of a variable restricted by a set of resources and each of the remaining variables contains the summation of a set of prefixed variables. The first condition stipulates that A_{jk} are the visible resources of the process and B_j the hidden resources, where the hidden resources that appear in each summand are the same but the world each combination describes is distinct from all others. Clause 2 stipulates that set V contains only resources that actually occur in the process.

Theorem 7.1 Every PACSR-lite process R provably satisfies a standard set of equations.

Proof: By induction on the structure of R . We present the most interesting case: $(R = P + Q)$. By induction hypothesis, P provably satisfies $S : \tilde{X} = \tilde{H}$ and Q provably satisfies $T : \tilde{Y} = \tilde{G}$. This implies that there exists terms \tilde{P} and \tilde{Q} such that $P = P_2 \parallel V$ and $Q = Q_2 \parallel U$ and $P + Q$ has the form

$$\begin{aligned} & (\sum_{j \in J_2} \sum_{k \in K_2} A_{jk} \cup B_j : P_{jk}) \parallel V \\ & + (\sum_{l \in L_2} \sum_{m \in M_2} C_{lm} \cup D_l : Q_{lm}) \parallel U. \end{aligned}$$

Using Rename, we can rewrite the second summand so that all its bound resources are fresh and different from the resources of the first summand, and the probabilities of the bound resources are preserved. Then using Hide(2) we can pull the hide operation to the outer level of the term, and $P + Q$:

$$\begin{aligned} & = (\sum_{j \in J_2} \sum_{k \in K_2} A_{jk} \cup B_j : P_{jk}) \parallel V \\ & + (\sum_{l \in L_2} \sum_{m \in M_2} C_{lm} \cup D'_l : Q'_{lm}) \parallel U' \\ & = (\sum_{j \in J_2} \sum_{k \in K_2} A_{jk} \cup B_j : P_{jk} \\ & + \sum_{l \in L_2} \sum_{m \in M_2} C_{lm} \cup D'_l : Q'_{lm}) \parallel V \cup U', \end{aligned}$$

where, if $\tilde{y} = \text{res}(P) \cap \text{res}(Q)$, $U' = U[\tilde{x}/\tilde{y}]$, $Q'_{lm} = Q_{lm}[\tilde{x}/\tilde{y}]$, $D'_l = D_l[\tilde{x}/\tilde{y}]$, s.t. for all i , $\text{p}(x_i) = \text{p}(y_i)$, $\tilde{x} \cap (\text{res}(P) \cup \text{res}(Q)) = \emptyset$.

To transform the above process to standard form and in particular to satisfy condition (1), we will need to apply laws Extend and Standard. First, we close the summands of the process with information about all immediate hidden resources of the process by applying law Extend once for every $r \in \bigcup_{j \in J_2} B_j \cup \bigcup_{l \in L_2} D'_l$ to obtain:

$$\begin{aligned} & (\sum_{n \in N} \sum_{j \in J_2} \sum_{k \in K_2} A_{jk} \cup B_j \cup E_n : P_{jk} + \\ & \sum_{n \in N'} \sum_{l \in L_2} \sum_{m \in M_2} C_{lm} \cup D'_l \cup F_n : Q'_{lm}) \parallel V \cup U' \end{aligned}$$

where the $\bigcup_{n \in N} \tilde{E}_n$ are the possible combinations of the immediate bound resources of process Q , $\bigcup_{l \in L_2} D'_l$, and similarly, the $\bigcup_{n \in N'} F_n$ are the possible combinations of the immediate bound resources of process P , $\bigcup_{j \in J_2} B_j$. Now it remains to rearrange the last two summands in the style of the left-hand side of law Standard, by grouping together all processes that can take

place under the same evaluation of the hidden resources, and then isolating all worlds that exhibit the same behavior. So, using Choice(3), Choice(4) and finally Standard we obtain:

$$\begin{aligned} P + Q & = (\sum_{i \in I} \sum_{j \in J_i} \sum_{k \in K_i} A'_{ik} \cup B'_{ij} : P'_{ik}) \parallel V \cup U' \\ & = (\sum_{i \in I} \sum_{k \in K_i} A'_{ik} \cup \tilde{\rho}_i : P'_{ik}) \parallel V \cup U' \cup \bigcup_i \tilde{\rho}_i \end{aligned}$$

where $\text{res}(\tilde{\rho}_{i_m}) = \text{res}(\tilde{\rho}_{i_n})$ and $\tilde{\rho}_{i_m} \neq \tilde{\rho}_{i_n}$ for all $i_m \neq i_n$. By the induction hypothesis each P'_{ik} provably satisfies a standard set of equations $S^{ik} : \tilde{X}^{ik} = \tilde{H}^{ik}$ with distinguished variable X_1^{ik} . Then $P + Q$, satisfies the standard set of equations: $\{X = X_1 \parallel V \cup U' \cup \bigcup_i \tilde{\rho}_i, X_1 = (\sum_{i \in I} \sum_{k \in K_i} A'_{ik} \cup \tilde{\rho}_i : X_1^{ik})\} \bigcup_{i,k} S^{ik}$.

7.2 Common Set of Prioritized Standard Equations

Theorem 7.2 *Let P and Q provably satisfy two standard sets of equations S and T . If P and Q are bisimilar, then there exists a third standard set of equations S' satisfied by both P and Q .*

Proof: We will restrict our attention to processes with standard sets of equations containing no visible resources, and $K_i = 1$. This allows us to focus on the central aspects of the proof that involve the renaming of bound resources.

Suppose that \tilde{X} and \tilde{Y} , are disjoint sets of variables, and that the given sets of equations are $S : \tilde{X} = \tilde{H}$, $T : \tilde{Y} = \tilde{G}$. Further, let \tilde{P} and \tilde{Q} be such that $\tilde{P} = \tilde{H}[\tilde{P}/\tilde{X}]$, $\tilde{Q} = \tilde{G}[\tilde{Q}/\tilde{Y}]$, with $P = P_1$, $Q = Q_1$, so that $P_1 = P_2 \parallel U$, $Q_1 = Q_2 \parallel V$, and $P_i = \sum_{j \in J_i} B_j : P_j$, $Q_i = \sum_{l \in L_i} D_l : Q_l$. Let us consider the relation \mathcal{R} such that $(u, v) \in \mathcal{R}$ iff $H_u \sim G_v$. Clearly, $(1, 1) \in \mathcal{R}$. Let $(u, v) \in \mathcal{R}$ and consider P_u and Q_v . Suppose that there exists $W \in \mathcal{W}(\bigcup_j B_j)$ such that for all j , $B_j \not\subseteq W$. (The other case follows similarly with the exception that law Standard(2) is used instead of Standard(1).) We

may construct a partition $\Lambda = \{\tilde{j}_1, \dots, \tilde{j}_n\}$ of J_u , such that if $j, j' \in \tilde{j}_\lambda$, $P_j \sim P_{j'}$, and vice versa. Similarly, let $\Lambda' = \{\tilde{l}_1, \dots, \tilde{l}_{n'}\}$ be the equivalent partition of Q_v . Since P_u and Q_v must have equal transitions, the following statement is true:

$$\begin{aligned} n &= n', \text{ and for each } \tilde{j}_\lambda \in \Lambda, \\ &\text{there exists } \tilde{l}_{\lambda'} \in \Lambda' \text{ such that for} \\ &\text{any } j \in \tilde{j}_\lambda, l \in \tilde{l}_{\lambda'}, (j, l) \in \mathcal{R}, \\ &\sum_{j \in \tilde{j}_\lambda} \mathfrak{p}(B_j) = \sum_{l \in \tilde{l}_{\lambda'}} \mathfrak{p}(D_l). \end{aligned}$$

Thus P_u and Q_v can be rewritten as follows $P_u = \sum_{\lambda=1}^n \sum_{j \in \tilde{j}_\lambda} B_j : P_j, Q_v = \sum_{\lambda=1}^n \sum_{l \in \tilde{l}_\lambda} D_l : Q_l$, where we assume that the summations are ordered so that for all $1 \leq \lambda \leq n$, classes $\tilde{j}_\lambda, \tilde{l}_\lambda$ are matching, in the sense of the above statement.

However, despite the bisimilarity of the two processes and the fact that they have the same cumulative probability of reaching each equivalence class of \sim , they have possibly different branching structures. Our intention is to show, that P_u and Q_v can be rewritten into equal processes which have identical branching structures. To do this we will employ a set of new hidden resources, and rewrite the two processes in such a way that each probabilistic transition of the initial processes with probability p is replaced by a set of probabilistic transitions with cumulative probability p . While doing this we want to ensure that both resulting processes have exactly the same probabilistic transitions to each equivalence class of \sim . We achieve this as follows:

For every λ , let γ_λ be the greatest common divisor of the probabilities $\mathfrak{p}(B_j)$, $\mathfrak{p}(D_l)$, for all $j \in \tilde{j}_\lambda, l \in \tilde{l}_\lambda$. Further, let $\Delta^\lambda = \frac{\sum_{j \in \tilde{j}_\lambda} \mathfrak{p}(B_j)}{\gamma_\lambda}$ and $\Delta = \sum_\lambda \Delta^\lambda$. By the definition of γ_λ , Δ^λ and thus Δ are integers. Let $\tilde{\rho}_{uv} = \rho_1, \dots, \rho_\Delta$ be new resources and $\tilde{\rho}_1, \dots, \tilde{\rho}_\Delta$, mutually exclusive worlds involving these resources such that the first Δ^1 worlds have probability γ_1 , the next Δ^2 worlds probability γ_2 and so on. Finally, if $j \in \tilde{j}_\lambda$, let $\epsilon_j = \frac{\mathfrak{p}(B_j)}{\gamma_\lambda}$,

and let $\{\Delta_1^\lambda, \dots, \Delta_{j_u}^\lambda\}$, be a partition of $\{1, \dots, \Delta^\lambda\}$, such that $\Delta_i^\lambda = \{\sum_{r < i} \epsilon_r + 1, \sum_{r < i} \epsilon_r + 2, \dots, \sum_{r \leq i} \epsilon_r\}$.

Similarly, if $l \in \tilde{l}_\lambda$, let $\epsilon'_j = \frac{\mathfrak{p}(D_j)}{\gamma_\lambda}$, and let $\{E_1^\lambda, \dots, E_{j_u}^\lambda\}$, be a partition of $\{1, \dots, \Delta^\lambda\}$, such that $E_i^\lambda = \{\sum_{r < i} \epsilon'_r + 1, \sum_{r < i} \epsilon'_r + 2, \dots, \sum_{r \leq i} \epsilon'_r\}$. By Standard(1), we have that the two processes satisfy the following equations: $P_u \parallel U = (\sum_{\lambda=1}^n \sum_{j \in \tilde{j}_\lambda} \sum_{\nu \in \Delta_j^\lambda} \tilde{\rho}_\nu : P_j) \parallel V \cup \tilde{\rho}_{uv}$, $Q_v \parallel V = (\sum_{\lambda=1}^n \sum_{l \in \tilde{l}_\lambda} \sum_{\nu \in E_l^\lambda} \tilde{\rho}_\nu : Q_l) \parallel V \cup \tilde{\rho}_{uv}$.

Let \tilde{X}' and \tilde{Y}' , be disjoint sets of variables, and consider the sets of equations $S' : \tilde{X}' = \tilde{H}', T' : \tilde{Y}' = \tilde{G}'$, where $X'_1 = X'_2 \parallel \bigcup_{u,v} \tilde{\rho}_{uv}$, $Y'_1 = Y'_2 \parallel \bigcup_{u,v} \tilde{\rho}_{uv}$, and $X'_u = (\sum_{\lambda=1}^n \sum_{j \in \tilde{j}_\lambda} \sum_{\nu \in \Delta_j^\lambda} \tilde{\rho}_\nu : X'_j)$, $Y'_v = (\sum_{\lambda=1}^n \sum_{l \in \tilde{l}_\lambda} \sum_{\nu \in E_l^\lambda} \tilde{\rho}_\nu : Y'_l)$. It can be shown that S' and T' are satisfied by P and Q respectively.

Let us now consider the set of equations $\tilde{Z} = \tilde{F}$, defined for all $(u, v) \in \mathcal{R}$ by $Z_{0,0} = Z_{1,1} \parallel \bigcup_{u,v} \tilde{\rho}_{uv}$, where

$$Z_{u,v} = \sum_{\lambda=1}^n \sum_{\nu=1}^{\Delta} \sum_{(j,l) \in K_{uv\nu}} \tilde{\rho}_\nu : Z_{j,l}$$

with $K_{uv\nu} = \{(j, l)\}$ s.t. $\tilde{\rho}_\nu : P_j$ is a summand of P_u , $\tilde{\rho}_\nu : Q_l$ is a summand of Q_v , $(j, l) \in \mathcal{R}$. Again, it is easy to prove that this is a set of standard equations.

Now take the set of processes $R_{j,l} = P_j$, for all l . We may see that the terms $F_{i,j}[\tilde{R}/\tilde{Z}]$ contain the same summands as $H_i[\tilde{P}/\tilde{X}']$ with some possible duplications. In particular, $F_{1,1}[\tilde{R}/\tilde{Z}] = P_1 = P$. Hence P satisfies this new set of equations. A similar reasoning can be applied to show that Q satisfies the same set of equations. \square

7.3 Unique Solution

We now have to prove that if two processes satisfy the same set of standard equations, they are bisimilar. Such is the objective of

the following theorem. Its proof follows exactly the proof given by Milner[7].

Theorem 7.3 *A set of standard equations has a unique solution up to a bisimulation.*

Since, by Theorem 7.2, P' and Q' satisfy a common set of standard equations, and by Theorem 7.3 $P' = Q'$, we have the final result:

Theorem 7.4 *For any two FS processes P and Q , if $P \sim Q$ then $P = Q$.*

8 Conclusions

We have presented a sound and complete axiomatization of strong bisimulation for the fragment PACSR-lite of the resource-oriented process algebra PACSR. The key technical hurdle was to axiomatically characterize the effects of resource hiding within a probabilistic setting. We are working to extend the axiomatization to the full PACSR.

References

- [1] J. Baeten, J. Bergstra, and S. Smolka. Axiomatizing probabilistic processes: ACP with generative probabilities. *Information and Computation*, 121(2):234–255, Sept. 1995.
- [2] P. Brémont-Grégoire, J. Choi, and I. Lee. A complete axiomatization of finite-states ACSR processes. *Information and Computation*, 138(2):124–159, Nov 1997.
- [3] A. Giacalone, C. Jou, and S. Smolka. Algebraic reasoning for probabilistic concurrent systems. In *Proceedings of Working Conference on Programming Concepts and Methods*, Sea of Gallilee, Israel, Apr. 1990. IFIP TC 2, North-Holland.
- [4] H. Hansson. *Time and Probability in Formal Design of Distributed Systems*. PhD thesis, Department of Computer Systems, Uppsala University, 1991. DoCS 91/27.
- [5] J.-P. Katoen, R. Langerak, and D. Latella. Modeling systems by probabilistic process algebra: An event structures approach. In *Proceedings of FORTE '92 – Fifth International Conference on Formal Description Techniques*, pages 255–270, Oct. 1993.
- [6] I. Lee, P. Brémont-Grégoire, and R. Gerber. A process algebraic approach to the specification and analysis of resource-bound real-time systems. *Proceedings of the IEEE*, pages 158–171, Jan 1994.
- [7] R. Milner. A complete axiomatization for observational congruence of finite-state behaviors. *Information and Computation*, 81:227–247, 1989.
- [8] D. Park. Concurrency and automata on infinite sequences. In *Proceedings of 5th GI Conference*, volume 104, of *Lecture Notes in Computer Science*, pages 167–183, 1981.
- [9] A. Philippou, O. Sokolsky, R. Cleaveland, I. Lee, and S. Smolka. Probabilistic resource failure in a real-time process algebra. In *Proceedings of CONCUR '98*, Sept. 1998.
- [10] R. Segala and N. Lynch. Probabilistic simulations for probabilistic processes. In B. Jonsson and J. Parrow, editors, *Proceedings CONCUR 94*, Uppsala, Sweden, volume 836 of *Lecture Notes in Computer Science*, pages 481–496. Springer-Verlag, 1994.
- [11] K. Seidel. *Probabilistic CSP*. PhD thesis, Oxford University, 1992.
- [12] C. Tofts. Processes with probabilities, priorities and time. *Formal Aspects of Computing*, 4:536–564, 1994.

Choice(1)	$P + \text{NIL} = P$
Choice(2)	$P + P = P$
Choice(3)	$P + Q = Q + P$
Choice(4)	$(P + Q) + R = P + (Q + R)$
Hide(1)	$\text{NIL} \backslash I = \text{NIL}$
Hide(2)	$(P + Q) \backslash I = (P \backslash I) + (Q \backslash I)$ if $\text{imr}(P) \cap \text{imr}(Q) \cap I = \emptyset$
Hide(3)	$(A:P) \backslash I = A:(P \backslash I)$ if $\text{res}(A) \cap I = \emptyset$
Hide(4)	$P \backslash I \backslash J = P \backslash I \cup J$
Hide(5)	$P \backslash \emptyset = P$
Hide(6)	$P \backslash I = P \backslash I \cup \{r\}$ if $r \notin \text{res}(P)$
Down	$A : P = \text{NIL},$ if for some $r \in A, \text{p}(r) = 0$
Rename	$P \backslash I = (P \backslash I)[r'/r]$ if $r \in I, r' \notin \text{res}(P)$ and $\text{p}(r) = \text{p}(r')$

Table 2: Laws for sum, recursion and hiding

Extend	$(\sum_{i \in I} A_i : P_i) \backslash V = (\sum_{j \in I, r \notin A_j} (A_j \cup \{r\}) : P_j + (A_j \cup \{\bar{r}\}) : P_j) + \sum_{j \in I, r \in A_j} A_j : P_j \backslash V$ where $r \in V$
Standard(1)	$(\sum_{i=1}^I \sum_{j=1}^{J_i} \sum_{k=1}^{K_i} (A_{ik} \cup B_{ij}) : P_{ik}) \backslash V$ $= (\sum_{i=1}^I \sum_{k=1}^{K_i} (A_{ik} \cup C_i) : P_{ik}) \backslash V \cup \{\rho_1, \dots, \rho_I\}$ if $\exists W \in \mathcal{W}(\bigcup_{i,j,k} A_{ik} \cup B_{ij}) \forall i, j, k \cdot A_{ik} \cup B_{ij} \not\subseteq W$, and whenever $i, j \neq m, n, \text{res}(B_{ij}) = \text{res}(B_{mn}), B_{ij} \neq B_{mn}$ and where $\bigcup_{i,j} B_{ij} \subseteq V, (\bigcup_{i,k} A_{ik}) \cap V = \emptyset,$ $C_i = \bigcup_{1 \leq j < i} \{\bar{\rho}_j\} \cup \bigcup_{i \leq j < I} \{\rho_j\},$ where ρ_1, \dots, ρ_I are fresh resources, such that $\text{p}(C_i) = \sum_{j=1}^{J_i} \text{p}(B_{ij})$
Standard(2)	$(\sum_{i=1}^I \sum_{j=1}^{J_i} \sum_{k=1}^{K_i} (A_{ik} \cup B_{ij}) : P_{ik}) \backslash V$ $= (\sum_{i=1}^{I-1} \sum_{k=1}^{K_i} (A_{ik} \cup C_i) : P_{ik} +$ $\sum_{D \in \mathcal{D}} \sum_{k=1}^{K_i} (A_{Ik} \cup D) : P_{Ik}) \backslash V \cup \{\rho_1, \dots, \rho_{I-1}\}$ if $\forall W \in \mathcal{W}(\bigcup_{i,j,k} A_{ik} \cup B_{ij}) \exists i, j, k \cdot A_{ik} \cup B_{ij} \subseteq W$, and whenever $i, j \neq m, n, \text{res}(B_{ij}) = \text{res}(B_{mn}), B_{ij} \neq B_{mn}$ and where $\bigcup_{i,j} B_{ij} \subseteq V, (\bigcup_{i,k} A_{ik}) \cap V = \emptyset,$ $C_i = \bigcup_{1 \leq j < i} \{\bar{\rho}_j\} \cup \bigcup_{i \leq j < I-1} \{\rho_j\},$ where $\rho_1, \dots, \rho_{I-1}$ are fresh resources, such that $\text{p}(C_i) = \sum_{j=1}^{J_i} \text{p}(B_{ij}),$ $\mathcal{D} = \mathcal{W}(\rho_1, \dots, \rho_{I-1}) - \{C_1, \dots, C_{I-1}\}$

Table 3: Laws for reintroduction of hidden resources