Axiomatizations for probabilistic finite-state behaviors ⋆

Yuxin Deng [a], Catuscia Palamidessi [b]

[a] *Shanghai Jiao Tong University, China*

[b] *INRIA Futurs and LIX, École Polytechnique, France*

Abstract

We study a process calculus which combines both nondeterministic and probabilistic behavior in the style of Segala and Lynch's probabilistic automata. We consider various strong and weak behavioral equivalences, and we provide complete axiomatizations for finite-state processes, restricted to guarded recursion in case of the weak equivalences. We conjecture that in the general case of unguarded recursion the "natural" weak equivalences are undecidable.

This is the first work, to our knowledge, that provides a complete axiomatization for weak equivalences in the presence of recursion and both nondeterministic and probabilistic choice.

Key words: Process calculus, Probability, Nondeterminism, Axiomatizations, Behavioral equivalences

1 Introduction

The last decade has witnessed increasing interest in the area of formal methods for the specification and analysis of probabilistic systems [22,5,3,20,26,7]. In [28] van Glabbeek *et al.* classified probabilistic models into *reactive, generative*

⋆ The work of Yuxin Deng was done when he was doing his PhD study at INRIA and Université Paris 7, France, under the support of the EU project PROFUNDIS. The work of Catuscia Palamidessi was partially supported by the Project Rossignol of the ACI Sécurité Informatique (Ministère de la recherche et nouvelles technologies). An extended abstract of this paper appeared at FOSSACS 2005.

Email addresses: `deng-yx@cs.sjtu.edu.cn` (Yuxin Deng), `catuscia@lix.polytechnique.fr` (Catuscia Palamidessi).

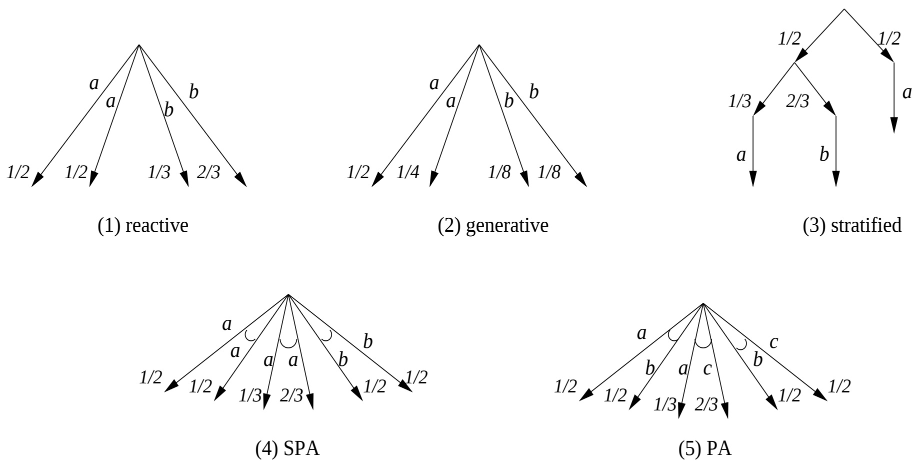

Fig. 1. Probabilistic models

and *stratified*. In reactive models, each labeled transition is associated with a probability, and for each state the sum of the probabilities with the same label is 1. Generative models differ from reactive ones in that for each state the sum of the probabilities of all the outgoing transitions is 1. Stratified models have more structure and for each state either there is exactly one outgoing labeled transition or there are only unlabeled transitions and the sum of their probabilities is 1.

In [22] Segala pointed out that neither reactive nor generative nor stratified models capture real nondeterminism, an essential notion for modeling scheduling freedom, implementation freedom, the external environment and incomplete information. He then introduced a model, the *probabilistic automata* (PA), where both probability and nondeterminism are taken into account. Probabilistic choice is expressed by the notion of *transition*, which, in PA, leads to a probabilistic distribution over pairs (action, state) and deadlock. Nondeterministic choice, on the other hand, is expressed by the possibility of choosing different transitions. Segala proposed also a simplified version of PA called *simple probabilistic automata* (SPA), which are like ordinary automata except that a labeled transition leads to a probabilistic distribution over a set of states instead of a single state.

Figure 1 exemplifies the probabilistic models discussed above. In models where both probability and nondeterminism are present, like those of diagrams (4) and (5), a transition is usually represented as a bundle of arrows linked by a small arc. [24] provides a detailed comparison between the various models, and argues that PA subsume all other models above except for the stratified ones.

In this paper we are interested in investigating axiom systems for a process calculus based on PA, in the sense that the operational semantics of each expression of the language is a probabilistic automaton [1] . Axiom systems are important both at the theoretical level, as they help to gain insight into the calculus and establish its foundations, and at the practical level, as tools for systems specification and verification. Our calculus is basically a probabilistic version of the core CCS used by Milner to express finite-state behaviors [13,15].

We shall consider four types of behavioral equivalences: two strong bisimulation equivalences, a weak equivalence sensitive to divergence, and observational equivalence. For recursion-free expressions we provide complete axiomatizations of all the four equivalences. For the strong equivalences we also give complete axiomatizations for all expressions, while for the weak equivalences we achieve this result only for guarded expressions.

The reason why we are interested in studying a model which expresses both nondeterministic and probabilistic behavior, and an equivalence sensitive to divergence, is that one of the long-term goals of this line of research is to develop a theory which will allow us to reason about probabilistic algorithms used in distributed computing. In that domain it is important to ensure that an algorithm will work under any scheduler, and under other unknown or uncontrollable factors. The nondeterministic component of the calculus enables one to deal with these conditions in a uniform and elegant way. Furthermore, in many distributed computing applications it is important to ensure livelock-freedom (progress), and therefore we will need a semantics which does not simply ignore divergences.

We are interested, in particular, in developing a fully distributed implementation of the (synchronous) π-calculus (π) [16,21] using a probabilistic asynchronous π-calculus (π_{pa}) [12] as an intermediate language. The reason why we need a probabilistic calculus is that it has been shown impossible to implement certain mechanisms of the π-calculus without using randomization [18]. We need also the nondeterministic dimension for the usual reasons: the implementation should be portable and in particular make no assumption about the scheduler. Some preliminary initial results of this project appeared in [19], where preliminary results on implementation were reported. We are now investigating a more realistic and efficient implementation.

We consider it important that an implementation does not introduce livelocks (or other kinds of unintended outcomes), thus the translation from π to π_{pa} should preserve livelock-freedom (see [19] for a discussion on the subject), and hence the semantics should be sensitive to divergence. For this reason, the second author chose (a probabilistic version of) testing semantics in [19]. However, it turned out that probabilistic testing semantics, at least the version

[1] Except for the case of deadlock, which is treated slightly differently: following the tradition of process calculi, in our case deadlock is a state, while in PA it is one of the possible components of a transition.

invented in [19], was rather difficult to use. The correctness proofs were ad-hoc, by hand, and rather complicated. For the realistic (and necessarily more sophisticated) implementation, we need proof methods feasible and (at least in part) automatic. For this reason, we are investigating here a divergence-sensitive *bisimulation-like* semantics. In the future, we plan to extend the results of this paper to π_{pa}.

Related work

In [13] and [15] Milner gave complete axiomatizations for strong bisimulation and observational equivalence, respectively, for a core CCS [14]. These two papers serve as our starting point: in several completeness proofs that involve recursion we adopt Milner's *equational characterization theorem* and *unique solution theorem.* In Section 5.1 and Section 6.2 we extend [13] and [15] (for guarded expressions), respectively, to the setting of probabilistic process algebra.

In [25] Stark and Smolka gave a probabilistic version of the results of [13]. So, our paper extends [25] in that we consider also nondeterminism. Note that, when nondeterministic choice is added, Stark and Smolka's technique of proving soundness of axioms can no longer be used. (See the discussion at the beginning of Appendix A.) The same remark applies also to [1] which follows the approach of [25] but uses some axioms from iteration algebra to characterize recursion. In contrast, our probabilistic version of "bisimulation up to" technique works well when combined with the usual transition induction.

In [17] Mislove *et al* presented a domain model for a process algebra with both probabilistic and nondeterministic choice. Their model is fully abstract with respect to a strong bisimilarity, for which they provided a complete axiomatization. However, weak behavioural equivalences are not considered in that paper.

In [6] Bandini and Segala axiomatized both strong and weak behavioral equivalences for process calculi corresponding to SPA and to an alternating model version of SPA. As their process calculus with non-alternating semantics corresponds to SPA, our results in Section 7 can be regarded as an extension of that work to PA.

For probabilistic process algebra of based on ACP, several complete axiom systems have appeared in the literature. However, in each of the systems either weak bisimulation is not investigated [4,2] or nondeterministic choice is prohibited [4,3].

Contribution of this work

The original contributions of this paper are:

- A complete axiomatization of a calculus which contains both nondeterministic and probabilistic choice, and recursion. We axiomatize both strong and weak behavioral equivalences. This is the first time, as far as we know, that a complete axiomatization of weak behavioral equivalences is presented for a language of this kind.
- The development and the axiomatization of a (probabilistic) weak behavioral equivalence sensitive to divergence.

Plan of the paper

In the next section we briefly recall some basic concepts and definitions about probability distributions. In Section 3 we introduce the calculus, with its syntax and operational semantics. In Section 4 we define the four behavioral equivalences we are interested in, and we extend the "bisimulation up to" technique of [14] to the probabilistic case. This technique is used extensively for the proofs of soundness of some axioms, especially in the case of the weak equivalences. In Sections 5 and 6 we give complete axiomatizations for the strong equivalences and for the weak equivalences respectively, restricted to guarded expressions in the second case. Section 7 gives complete axiomatizations for the four equivalences in the case of the finite fragment of the language. The interest of this section is that we use different and much simpler proof techniques than those in Sections 5 and 6. Finally, Section 8 concludes and illustrates our research plans.

2 Preliminaries

Let S be a set. A function $\eta : S \rightarrow [0,1]$ is called a *discrete probability distribution*, or *distribution* for short, on S if the *support* of η, defined as $spt(\eta) = \{x \in S \mid \eta(x) > 0\}$, is finite or countably infinite and $\sum_{x \in S} \eta(x) = 1$. If η is a distribution with finite support and $V \subseteq spt(\eta)$ we use the set $\{(s_i : \eta(s_i))\}_{s_i \in V}$ to enumerate the probability associated with each element of V.

At last, the completeness part of Theorem 33 (4) can be proved as Lemma 34. Note that for any $k, l \Box I$ with $u_k = u_l$ and $E_k \approx E_l$, we derive $\mathbf{A}_{fo} \Box u_k.E_k = u_l.E_l$ by using **T4** and the promotion lemma instead of using induction hypothesis.

It is worth noticing that rule **T5** is necessary to prove Lemma 34. Consider the following two expressions: $\tau.a$ and $\tau.(a + (\frac{1}{2}\tau.a \Box \frac{1}{2}a))$. It is easy to see that they are observationally equivalent. However, we cannot prove their equality if rule **T5** is excluded from the system $\mathbf{A}_{fo}$. In fact, by using only the other rules and axioms it is impossible to transform $\tau.(a + (\frac{1}{2}\tau.a \Box \frac{1}{2}a$

$\square$

Lemma 39 *Suppose* $d_X(G) > 1$, $\eta = \{(u_i, G_i : p_i)\}_{i \in I}$ *and* $G\{E/X\} \ \square\ \eta$. *Then* $G_i = G'_i\{E/X\}$ *for each* $i \square I$. *Moreover,* $G\{F/X\} \ \square\ \eta'$ *and* $\eta \equiv_{\mathcal{R}^*} \eta'$, *where* $\eta' = \{(u_i, G'_i\{F/X\} : p_i)\}_{i \in I}$ *and*

$$\mathbf{R} = \{(G\{E/X\}, G\{F/X\}) \mid \text{for any } G \square \mathbf{E}\}.$$

Proof. A direct consequence of Lemma 38. $\square$

Lemma 40 *Let* $d_X(G) > 1$. *If* $G\{E/X\} \ \square_c\ \eta$ *then* $G\{F/X\} \ \square_c\ \eta'$ *such that* $\eta \equiv_{\mathcal{R}^*} \eta'$ *where* $\mathbf{R} = \{(G\{E/X\}, G\{F/X\}) \mid$ *for any* $G \square \mathbf{E}\}$.

Proof. Let $\eta = r_1\eta_1 + ... + r_n\eta_n$ and $G\{E/X\} \ \square\ \eta_i$ for each $i \leq n$. By Lemma 39, for each $i \leq n$, there exists η'_i s.t. $G\{F/X\} \ \square\ \eta'_i$ and $\eta_i \equiv_{\mathcal{R}^*} \eta'_i$. Now let $\eta' = r_1\eta'_1 + ... + r_n\eta'_n$, thus $G\{F/X\} \ \square_c\ \eta'$. By lemma 36 it follows that $\eta \equiv_{\mathcal{R}^*} \eta'$. $\square$

Proof of Proposition 15(3). We show that the relation

$$\mathbf{R} = \{(G\{E/X\}, G\{\mu_X F/X\}) \mid \text{for any } G \square \mathbf{E}\}$$

is an observational equivalence up to $\square$. That is, we need to show the following assertions:

(1) if $G\{E/X\} \ \square\ \eta$ then there exists η' s.t. $G\{\mu_X F/X\} \ \square_c\ \eta'$ and $\eta \equiv_{\mathcal{R}_\approx} \eta'$;
(2) if $G\{\mu_X F/X\} \ \square\ \eta'$ then there exists η s.t. $G\{E/X\} \ \square_c\ \eta$ and $\eta \equiv_{\mathcal{R}_\approx} \eta'$;

We concentrate on the first clause as the second one is similar. The proof is carried out by induction on the depth of the inference of $G\{E/X\} \ \square\ \eta$. There are several cases depending on the structure of G. As an example, here we consider the case that $G \equiv X$.

We write $G(E)$ for $G\{E/X\}$ and $G^2(E)$ for $G(G(E))$. Since $E \ \square\ F(E)$, we have $E \ \square\ F^2(E)$ since $\square$ is an congruence relation by Proposition 15. If $E \ \square\ \eta$ then by Lemma 37 there exists θ_1 s.t. $F^2(E) \ \square_c\ \theta_1$ and $\eta \equiv_\approx \theta_1$. Since X is guarded in F, i.e., $d_X(F) > 0$, then it follows that $d_X(F^2(X)) > 1$. By Lemma 40, there exists θ_2 s.t. $F^2(\mu_X F) \ \square_c\ \theta_2$ and $\theta_1 \equiv_{\mathcal{R}^*} \theta_2$. From Proposition 14 we have $\mu_X F \ \square\ F^2(\mu_X F)$, thus $\mu_X F \ \square\ F^2(\mu_X F)$. By Lemma 37 there exists η' s.t. $\mu_X F \ \square_c\ \eta'$ and $\theta_2 \equiv_\approx \eta'$. From Lemma 35 and the transitivity of $\equiv_{\mathcal{R}_\approx}$ it follows that $\eta \equiv_{\mathcal{R}_\approx} \eta'$. $\square$

References

[1] L. Aceto, Z. Ésik, and A. Ingólfsdóttir. Equational axioms for probabilistic bisimilarity (preliminary report). Technical Report RS-02-6, BRICS, 2002.

[2] S. Andova. Process algebra with probabilistic choice. Technical Report CSR 99-12, Eindhoven University of Technology, 1999.

[3] S. Andova and J. C. M. Baeten. Abstraction in probabilistic process algebra. In *Tools and Algorithms for the Construction and Analysis of Systems*, volume 2031 of *LNCS*, pages 204–219. Springer, 2001.

[4] J. C. M. Baeten, J. A. Bergstra, and S. A. Smolka. Axiomatizing probabilistic processes: ACP with generative probabilities. *Information and Computation*, 121(2):234–255, 1995.

[5] C. Baier and H. Hermanns. Weak bisimulation for fully probabilistic processes. In *Proceedings of the 9th International Conference on Computer Aided Verification*, volume 1254 of *LNCS*, pages 119–130. Springer, 1997.

[6] E. Bandini and R. Segala. Axiomatizations for probabilistic bisimulation. In *Proceedings of the 28th International Colloquium on Automata, Languages and Programming*, volume 2076 of *LNCS*, pages 370–381. Springer, 2001.

[7] S. Cattani and R. Segala. Decision algorithms for probabilistic bisimulation. In *Proceedings of the 13th International Conference on Concurrency Theory*, volume 2421 of *LNCS*, pages 371–385. Springer, 2002.

[8] P. R. D'Argenio, H. Hermanns, and J.-P. Katoen. On generative parallel composition. *ENTCS*, 22, 1999.

[9] Y. Deng, C. Palamidessi, and J. Pang. Compositional reasoning for probabilistic finite-state behaviors. In *Processes, Terms and Cycles: Steps on the Road to Infinity, Essays Dedicated to Jan Willem Klop, on the Occasion of His 60th Birthday*, volume 3838 of *LNCS*, pages 309–337. Springer, 2005.

[10] Y. Fu and Z. Yang. Tau laws for pi calculus. *Theoretical Computer Science*, 308:55–130, 2003.

[11] J. Groote and A. Ponse. The syntax and semantics of μCRL. Technical Report CS-R9076, CWI, Amsterdam, 1990.

[12] O. M. Herescu and C. Palamidessi. Probabilistic asynchronous pi-calculus. Technical report, INRIA Futurs and LIX, 2004.

[13] R. Milner. A complete inference system for a class of regular behaviours. *Journal of Computer and System Science*, 28:439–466, 1984.

[14] R. Milner. *Communication and Concurrency*. Prentice-Hall, 1989.

[15] R. Milner. A complete axiomatisation for observational congruence of finite-state behaviours. *Information and Computation*, 81:227–247, 1989.

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

[17] M. W. Mislove, J. Ouaknine, and J. Worrell. Axioms for probability and nondeterminism. *ENTCS*, 96:7–28, 2004.

[18] C. Palamidessi. Comparing the expressive power of the synchronous and the asynchronous pi-calculus. *Mathematical Structures in Computer Science*, 13(5):685–719, 2003.

[19] C. Palamidessi and O. M. Herescu. A randomized encoding of the π-calculus with mixed choice. Technical report, INRIA Futurs and LIX, 2004.

[20] A. Philippou, I. Lee, and O. Sokolsky. Weak bisimulation for probabilistic systems. In *Proceedings of the 11th International Conference on Concurrency Theory*, LNCS, pages 334–349. Springer, 2000.

[21] D. Sangiorgi and D. Walker. *The π-calculus: a Theory of Mobile Processes.* Cambridge University Press, 2001.

[22] R. Segala. Modeling and verification of randomized distributed real-time systems. Technical Report MIT/LCS/TR-676, PhD thesis, MIT, Dept. of EECS, 1995.

[23] R. Segala and N. Lynch. Probabilistic simulations for probabilistic processes. In *Proceedings of the 5th International Conference on Concurrency Theory*, volume 836 of *LNCS*, pages 481–496. Springer, 1994.

[24] A. Sokolova and E. de Vink. Probabilistic automata: system types, parallel composition and comparison. In *Validation of Stochastic Systems: A Guide to Current Research*, volume 2925 of *LNCS*, pages 1–43. Springer, 2004.

[25] E. W. Stark and S. A. Smolka. A complete axiom system for finite-state probabilistic processes. In *Proof, language, and interaction: essays in honour of Robin Milner*, pages 571–595. MIT Press, 2000.

[26] M. Stoelinga. *Alea jacta est: verification of probabilistic, real-time and parametric systems.* PhD thesis, University of Nijmegen, 2002.

[27] J. van de Pol. A prover for the muCRL toolset with applications – version 0.1. Technical Report SEN-R0106, CWI, Amsterdam, 2001.

[28] R. J. van Glabbeek, S. A. Smolka, and B. Steffen. Reactive, generative, and stratified models of probabilistic processes. *Information and Computation*, 121(1):59–80, 1995.