Book Introduction by the Authors

Invited by

Luca Aceto

luca.aceto@gmail.com
President of EATCS
Reykjavik University, Reykjavik, Iceland

Semantics of Probabilistic Processes
An Operational Approach

Yuxin Deng
Shanghai Jiaotong University
yuxindeng@sjtu.edu.cn

Subject matter

With the rapid development of computer network and communication technology, the study of concurrent and distributed systems has become increasingly important. Among various models of concurrent computation, process calculi have been widely investigated and successfully used in the specification, design, analysis and verification of practical concurrent systems. In recent years, probabilistic process calculi have been proposed to describe and analyse quantitative behaviour of concurrent systems, which calls for the study of semantic foundations of probabilistic processes.

In “Semantics of Probabilistic Processes” [4] we adopt an operational approach to describing the behaviour of nondeterministic and probabilistic processes. The semantic comparison of different systems is based on appropriate behavioural relations such as bisimulation equivalences and testing preorders.

This book mainly consists of two parts. The first part provides an elementary account of bisimulation semantics for probabilistic processes from metric, logical and algorithmic perspectives. The second part sets up a general testing framework and specialises it to probabilistic processes with nondeterministic behaviour. The resulting testing semantics is treated in depth. A few variants of it are shown to coincide, and they can be characterised in terms of modal logics and coinductively defined simulation relations. Although in the traditional (nonprobabilistic) setting, simulation semantics is in general finer (i.e. it distinguishes more processes) than testing semantics, for a large class of probabilistic processes, the gap between simulation and testing semantics disappears. Therefore, in this case, we have a semantics where both negative and positive results can be easily proved: to show that two processes are not related in the semantics, we just give a witness test, and to prove that two processes are related, we only need to establish a simulation relation.

Why yet another book?

Three decades have passed since the well-known books on process algebras by Hoare [8], Milner [10], Baeten and Weijland [3], and Hennessy [7] were published. In the meanwhile some excellent textbooks have appeared, including those by Roscoe [13, 14], Milner [11], Fokkink[6], Sangiorgi and Walker [17], Aceto et al. [1], Sangiorgi [15], as well as Sangiorgi and Rutten [16]. They are mainly about classical (nonprobabilistic) process algebras. For probabilistic concurrency theory, the book by Panangaden [12] is dedicated to the model of labelled Markov Processes and the book by Doberkat [5] treats stochastic logics in depth. Probabilistic model checking is well covered in the books by Baier and Katoen [2], and Kwiatkowska et al. [9]. This book, however, collects some recent developments in probabilistic testing semantics and gives an elementary account of probabilistic bisimulation semantics. Below we give a rough overview of the book's contents.

Mathematical preliminaries

In order to study the semantics of probabilistic processes, several mathematical concepts and results turn out to be very useful. They are collected in Chapter 2, including, for example, continuous functions over complete lattices, the Knaster-Tarski fixed-point theorem, induction and coinduction proof principles, compact sets in topological spaces, the separation theorem in geometry, the Banach fixed-point theorem in metric spaces, the π-λ theorem in probability spaces and the duality theorem in linear programming. The purpose of introducing these contents is to make the proofs in later chapters more accessible to postgraduate students and junior researchers entering the discipline of theoretical computer science.

Probabilistic bisimulation

In this book we work within a framework that features the co-existence of probability and nondeterminism. More specifically, we deal with *probabilistic labelled transition systems (pLTS's)* that are an extension of the usual *labelled transition systems* so that a step of transition is in the form $s \xrightarrow{a} \Delta$, meaning that state s can perform action a and evolve into a distribution Δ over some successor states. The diagram in Figure 1 describes a pLTS; states are represented by nodes of the form $\bullet$ and distributions by nodes of the form $\circ$.

Let s and t be two states in a pLTS. They are related by *probabilistic simulation* $\mathcal{R}$, written $s \mathrel{\mathcal{R}} t$, if for each transition $s \xrightarrow{a} \Delta$ from s there exists a transition $t \xrightarrow{a} \Theta$ from t such that Θ can somehow mimic the behaviour of Δ according to $\mathcal{R}$. To formalise the mimicking of Δ by Θ, we have to *lift* $\mathcal{R}$ to be a relation $\mathcal{R}^\dagger$ between distributions over states so that we can require $\Delta \mathrel{\mathcal{R}^\dagger} \Theta$.

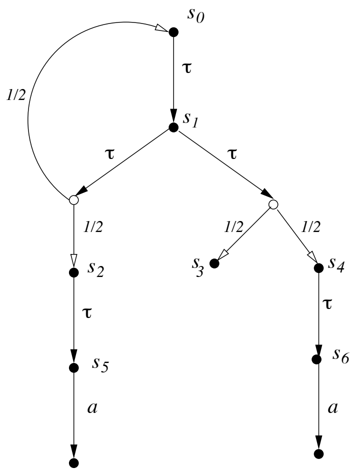

Figure 1: A pLTS

Various methods of lifting relations have appeared in the literature, but they can be reconciled. Essentially, there is only one lifting operation, which has been rediscovered in different occasions and presented in different forms. Moreover, we argue that the lifting operation is interesting in itself. This is justified by its intrinsic connection with some fundamental concepts in mathematics, notably *the Kantorovich metric*. For example, it turns out that our lifting of binary relations from states to distributions nicely corresponds to the lifting of metrics from states to distributions by using the Kantorovich metric. In addition, the lifting operation is closely related to *the maximum flow problem* in optimisation theory.

In Chapter 3 we provide three characterisations of probabilistic bisimulation, from the perspectives of modal logics, metrics, and decision algorithms.

- Our logical characterisation of probabilistic bisimulation consists of two aspects: *adequacy* and *expressivity*. A logic $\mathcal{L}$ is adequate when two states are bisimilar if and only if they satisfy exactly the same set of formulae in $\mathcal{L}$. The logic is expressive when each state s has a characteristic formula φ_s in $\mathcal{L}$ such that state t is bisimilar to s if and only if t satisfies φ_s. We introduce a probabilistic-choice modality to capture the behaviour of distributions. Intuitively, distribution Δ satisfies the formula $\bigoplus_{i\in I} p_i \cdot \varphi_i$ if there is a decomposition of Δ into a convex combination of some distributions, $\Delta = \sum_{i\in I} p_i \cdot \Delta_i$, and each Δ_i conforms to the property specified by φ_i. When the new modality is added to the Hennessy-Milner logic we obtain an adequate logic for probabilistic bisimilarity; when it is added to the modal mu-calculus we obtain an expressive logic.

- By metric characterisation of probabilistic bisimulation, we mean to give a pseudometric such that two states are bisimilar if and only if their distance is 0 when measured by the pseudometric. More specifically, we show that bisimulations correspond to pseudometrics that are postfixed points of a monotone function, and in particular bisimilarity corresponds to a pseudometric that is the greatest fixed point of the monotone function.

- As to the algorithmic characterisation, we first introduce a partition refinement algorithm to check whether two states are bisimilar. Then we provide an "on-the-fly" algorithm that checks whether two states are related by probabilistic bisimilarity. The schema of the algorithm is to approximate probabilistic bisimilarity by iteratively accumulating information about state pairs (s, t) where s and t are not bisimilar. In each iteration we dynamically construct a relation $\mathcal{R}$ as an approximant. Then we verify that every transition from one state should be matched by a transition from the other state, and that their resulting distributions are related by the lifted relation $\mathcal{R}^\dagger$. The latter involves solving the maximum flow problem of an appropriately constructed network, by taking advantage of the close relationship between our lifting operation and the above mentioned maximum flow problem.

Probabilistic testing semantics

It is natural to view the semantics of processes as being determined by their ability to pass tests; two processes are deemed to be semantically equivalent unless there is a test that can distinguish them. The actual tests used typically represent the ways in which users, or indeed other processes, can interact with the processes under examination. To formulate this idea, in Chapter 4 we set up a general testing scenario. It assumes

- a set of processes $\mathcal{P}roc$
- a set of tests $\mathcal{T}$, which can be applied to processes
- a set of outcomes O, the possible results from applying a test to a process
- a function $\mathcal{A} : \mathcal{T} \times \mathcal{P}roc \to \mathcal{P}(O)$, representing the possible results of applying a specific test to a specific process.

Here $\mathcal{P}(O)$ denotes the collection of non-empty subsets of O; so the result of applying a test T to a process P, $\mathcal{A}(T, P)$, is in general a *non-empty set* of outcomes, representing the fact that the behaviour of processes, and indeed tests, may be nondeterministic.

Moreover, some outcomes are considered better then others; for example, the application of a test may simply succeed, or it may fail, with success being better than failure. So we can assume that O is endowed with a partial order, in which $o_1 \le o_2$ means that o_2 is a better outcome than o_1.

When comparing the results of applying tests to processes we need to compare subsets of O. There are two standard approaches to make this comparison, based on viewing these sets as elements of either the Hoare or Smyth powerdomain of O[1]. Consequently, we have two different semantic preorders for processes:

(i) For $P, Q \in \mathcal{P}roc$ let $P \sqsubseteq_{\text{may}} Q$ if for any test T and every outcome $o_1 \in \mathcal{A}(T, P)$ there exists some $o_2 \in \mathcal{A}(T, Q)$ such that $o_1 \le o_2$.

(ii) Similarly, let $P \sqsubseteq_{\text{must}} Q$ if for any test T and every $o_2 \in \mathcal{A}(T, Q)$ there exists some $o_1 \in \mathcal{A}(T, P)$ such that $o_1 \le o_2$.

Let us have a look at two typical instances of the set O and its associated partial order $\le$.

1. For probabilistic processes we consider an application of a test to a process to succeed with a given probability. Thus we take as the set of outcomes the unit interval $[0, 1]$, with the standard ordering: if $0 \le p < q \le 1$ then succeeding with probability q is considered better than succeeding with probability p. We refer to this approach as *scalar testing*.

2. Another approach of testing, as originally proposed by Segala, employs a countable set of special actions $\Omega = \{\omega_1, \omega_2, ...\}$ to report success. When applied to probabilistic processes, this approach uses the function space $[0, 1]^\Omega$ as the set of outcomes and the standard partial order for real functions: for any $o_1, o_2 \in O$, we have $o_1 \le o_2$ if and only if $o_1(\omega) \le o_2(\omega)$ for every $\omega \in \Omega$. When Ω is fixed, an outcome $o \in O$ can be considered as a vector $\langle o(\omega_1), o(\omega_2), ...\rangle$, with $o(\omega_i)$ representing the probability of success observed by action ω_i. Therefore, this approach is called *vector-based testing*.

Surprisingly, it turns out that for *finitary* systems, i.e. finite-state and finitely branching systems, scalar testing is equally powerful as vector-based testing. This is the main result shown in Chapter 4. Other variants of testing approaches, such as reward testing and extremal reward testing are also discussed. They all coincide with vector-based testing as far as finitary systems are concerned.

[1] A third approach is to use the Plotkin powerdomain, which can be obtained by combining the Hoare and Smyth powerdomains.

Testing finite probabilistic processes

Chapter 5 investigates the connection between testing and simulation semantics. The simulation semantics is based on a notion of failure simulation and a notion of forward simulation; a distinguishing feature of them is to allow for the comparison of states with distributions. We say a relation $\mathcal{R} \subseteq S \times \mathcal{D}(S)$ is a *failure simulation* if $s \mathrel{\mathcal{R}} \Theta$ implies

(i) for any action α, if $s \xrightarrow{\alpha} \Delta$ then there exists some Θ' such that $\Theta \overset{\alpha}{\Longrightarrow} \Theta'$ and $\Delta \mathrel{\mathcal{R}^\dagger} \Theta'$

(ii) for any set of actions A, if s fails to perform any action in A, then so does some Θ' with $\Theta \overset{\tau}{\Longrightarrow} \Theta'$.

Here we write $\overset{\alpha}{\Longrightarrow}$ for a weak transition that abstracts away the internal action τ. Similarly, we define *forward simulation* by dropping the clause (ii) above.

For *finite processes*, i.e. processes whose behaviour can be described by pLTS's with finite tree structures, testing semantics is not only sound but also complete for simulation semantics. More specifically, may testing preorder coincides with forward simulation preorder and must testing preorder coincides with failure simulation preorder. Therefore, unlike the traditional (nonprobabilistic) setting, here there is no gap between testing and simulation semantics. To prove this result we make use of logical characterisations of testing preorders. For example, each state s has a characteristic formula φ_s in the sense that another state t can simulate s if and only if t satisfies φ_s. We can then turn this formula φ_s into a characteristic test T_s so that if t is not related to s via the may testing preorder then T_s is a witness test that distinguishes t from s. Similarly for the case of failure simulation and must testing. We also give a complete axiom system for the testing preorders in the finite fragment of a probabilistic process algebra.

Testing finitary probabilistic processes

In Chapter 6 we extend the results in the previous chapter from finite processes to finitary processes. Testing preorders can still be characterised as simulation preorders and admit modal characterisations. The soundness and completeness proofs inherit the general schemata from Chapter 5. However, the technicalities are much more subtle and more interesting. For example, the weak transition relation $\overset{\tau}{\Longrightarrow}$ needs to be carefully defined so as to abstract away infinitely many internal transition steps, and we make a significant use of subdistributions. A crucial topological property shown in this chapter is that from any given subdistribution, the set of stable subdistributions reachable from it by weak transitions can be finitely generated. Consider the pLTS in Figure 1 again. Both the point distribution $\overline{s_5}$ and the distribution $(\frac{1}{2}\overline{s_3} + \frac{1}{2}\overline{s_6})$ are stable and reachable from s_0. In fact, by taking linear combinations of them we obtain the set of all stable subdistributions reachable from s_0. The proof is highly non-trivial and involves techniques from *Markov decision processes* such as rewards and static policies. This result enables us to approximate coinductively defined relations by stratified inductive relations. As a consequence, if two processes behave differently we can tell them apart by a finite test.

We also introduce a notion of real-reward testing that allows for negative rewards. It turns out that real-reward may preorder is the inverse of real-reward must preorder, and vice versa. More interestingly, for finitary convergent processes, real-reward must testing preorder coincides with nonnegative-reward testing preorder.

Weak probabilistic bisimulation

In Chapter 7 we introduce a notion of weak probabilistic bisimulation simply by taking the symmetric form of the forward simulation preorder given in Chapter 6. It provides a sound and complete proof methodology for an extensional behavioural equivalence, a probabilistic variant of the traditional *reduction barbed congruence* well-known in concurrency theory.

More information

More information on this book can be found at

`www.springer.com/978-3-662-45197-7`

Acknowledgements Most of the work reported in this book was carried out during the last few years with a number of colleagues including Rob van Glabbeek, Matthew Hennessy, Carroll Morgan, Chenyi Zhang, and Wenjie Du. Thanks go also to Barry Jay, Matthew Hennessy and Carroll Morgan for having read parts of the first draft and offered useful feedback. The Springer staff have provided wonderful cooperation in the process of editing and publishing the book. In particular, I am thankful to Jane Li. My research on probabilistic concurrency theory has been sponsored by the National Natural Science Foundation of China under grants 61173033 and 61261130589. Finally, I thank Luca Aceto for his comments on this summary.

References

[1] L. Aceto, A. Ingólfsdóttir, K.G. Larsen and J. Srba. *Reactive Systems: Modelling, Specification and Verification.* Cambridge University Press, 2007.

[2] C. Baier and J.-P. Katoen. *Principles of Model Checking*. MIT Press, 2008.

[3] J.C.M. Baeten and W.P. Weijland. *Process Algebra*. Cambridge Tracts in Theoretical Computer Science, vol. 18. Cambridge University Press, 1990.

[4] Y. Deng. *Semantics of Probabilistic Processes: An Operational Approach*. Springer, 2014.

[5] E.E. Doberkat. *Stochastic Coalgebraic Logic*. Springer, 2010.

[6] W. Fokkink. *Introduction to Process Algebra*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2000.

[7] M. Hennessy. *Algebraic Theory of Processes*. The MIT Press, 1988.

[8] C.A.R. Hoare. *Communicating Sequential Processes*. Prentice Hall, 1985.

[9] M. Kwiatkowska, G. Norman, D. Parker and J.J.M.M. Rutten. *Mathematical Techniques for Analyzing Concurrent and Probabilistic Systems*. Crm Monograph Series. American Mathematical Society, 2004.

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

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

[12] P. Panangaden. *Labelled Markov Processes*. Imperial College Press, London, 2009.

[13] A.W. Roscoe. *The Theory and Practice of Concurrency*. Prentice-Hall, 1997.

[14] A.W. Roscoe. *Understanding Concurrent Systems*. Springer, 2010.

[15] D. Sangiorgi. *Introduction to Bisimulation and Coinduction*. Cambridge University Press, 2011.

[16] D. Sangiorgi and J. Rutten (editors). *Advanced Topics in Bisimulation and Coinduction*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2011.

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