

**Deng, Yuxin**

**Semantics of probabilistic processes. An operational approach.** (English) Zbl 1315.68002

Shanghai: Shanghai Jiao Tong University Press; Heidelberg: Springer (ISBN 978-3-662-45197-7/hbk; 978-3-662-45198-4/ebook). xiv, 249 p. (2014).

Probabilistic labelled transition systems (pLTS) are a generalization of LTS by adding probabilities. These are essentially the same as probabilistic automata in the presentation by *R. Segala* and *N. Lynch* [Nord. J. Comput. 2, No. 2, 250–273 (1995; [Zbl 0839.68067](#))]. The present monograph studies probabilistic processes as modeled by finite-state and finitely branching pLTS and considers the variety of approaches to this subject: Behavior, as given by the notions of probabilistic bisimulation and simulation; process-algebraic, by studying an extension of CSP [*C. A. R. Hoare*, Communicating sequential processes. Repr. New York, NY: Prentice Hall (1995; [Zbl 0841.68042](#))] with a probabilistic construct, defining an appropriate semantics, and studying congruences; testing, including scalar- and vector-based versions, as well as a real reward testing study; and finally modal logic, including the presentation of characteristic formulas.

The book is organized in seven chapters, each with its own abstract, selection of keywords, and reference list (there is no comprehensive bibliography).

The first chapter consists of a concise introduction to the subject matter of the text, and the second includes essentially all mathematical preliminaries that will be used afterwards. This includes some material on complete lattices, fixpoints, and the related inductive and coinductive proof principles; a look into [*D. Sangiorgi*, Introduction to bisimulation and coinduction. Cambridge: Cambridge University Press (2012; [Zbl 1252.68008](#))] might be beneficial, as the author recommends. Also metric and probability spaces are briefly reviewed.

The main model, pLTS, is defined in Chapter 3. In order to define bisimulation for pLTS, a lifting of relations from states to distributions is studied; this lifting is then related to the Kantorovich metric on the space of probabilities and to network flow problems. Then a modal logic that characterizes bisimulation is introduced. This logic contains two kinds of formulas, where the first kind is interpreted on the state space, and the other kind on the set of distributions over the state space, as it was first developed in [*B. Jonsson* et al., in: Handbook of process algebra. Amsterdam: North-Holland/Elsevier (2001; [Zbl 1062.68081](#))]. In subsequent sections, an expressive logic (having formulas that characterize each bisimilarity class) is presented. Finally, metric characterizations and two algorithms (partition refinement and ‘on-the-fly’) for probabilistic bisimilarity are expounded.

Chapter 4 presents the testing framework where, as usual, tests are conceived as processes of the same kind being tested. Testing preorders (in ‘must’ and ‘may’ flavors) are defined. As a key result in the context of the book, it is shown that for finitary processes, vector-based testing and scalar testing are equally powerful. Several technical lemmas, including some material on convex subsets of Euclidean  $n$ -space, are developed.

In Chapter 5, testing is applied to finite (i.e., finitary without loops, or terminating) processes. Here, the process terms of pCSP are defined. The *state-base* terms (a subset of the process terms) are the ones that are used as the state space of the operational semantics of this language. Several identities between CSP terms on plain LTS cease to hold for pLTS, and there is a wealth of examples in this chapter (and in other parts of the book) that illustrate these facts thoroughly. In this chapter, forward and failure simulation preorders are defined, and expressive logics are developed for each of these preorders. By assigning characteristic tests to formulas, it is shown that (failure) simulation preorder corresponds to (must) may testing preorder. The results are rounded off by presenting (in)equational calculi for both preorders.

Chapter 6 generalizes the previous one from finite to finitary processes, strengthening pCSP by adding a recursive construct, obtaining the language rpCSP. The exposition scheme goes in parallel lines with that of Chapter 5, but there is an intensive use of subdistributions. One of the highlights in this chapter is the zero-one law, that states that whenever one finds a transition to a true subdistribution (i.e., with total mass less than 1), there must exist a wholly divergent state. The generalization to finitary processes requires a careful definition of weak  $\tau$ -transitions that involves a potentially infinite decomposition of a

distribution.

The last chapter introduces weak probabilistic bisimulation, by using weak  $\tau$ -transitions, and reduction-barbed congruence (see [*K. Honda* and *N. Yoshida*, *Theor. Comput. Sci.* 151, No. 2, 437–486 (1995; [Zbl 0871.68122](#))] and Sangiorgi's book [loc. cit.]). It is proved that the former implies the latter for finitary pLTS. In one particular case, namely, the pLTS given by the full rpCSP, the converse is shown to hold as well.

As a final remark, the book provides a complete introduction to the subject, including several recent results and many interesting examples that delimit the former. It is written in a research-level style, but proofs are detailed and it provides a smooth read.

Reviewer: [Pedro Sánchez Terraf](#) (Córdoba)

**MSC:**

- 68-02 Research exposition (monographs, survey articles) pertaining to computer science
- 03B70 Logic in computer science
- 68Q10 Modes of computation (nondeterministic, parallel, interactive, probabilistic, etc.)
- 68Q55 Semantics in the theory of computing
- 68Q85 Models and methods for concurrent and distributed computing (process algebras, bisimulation, transition nets, etc.)
- 68Q87 Probability in computer science (algorithm analysis, random structures, phase transitions, etc.)

Cited in 2 Documents

**Keywords:**

[probabilistic bisimulation](#); [probabilistic labelled transition system](#); [testing semantics](#); [reward testing](#); [probabilistic simulation](#); [modal logic](#); [characteristic formula](#); [weak probabilistic bisimulation](#); [reduction-barbed congruence](#)

**Full Text:** [DOI](#)