Journal of Logical and Algebraic Methods in Programming 96 (2018) 41–64

Contents lists available at ScienceDirect

Journal of Logical and Algebraic Methods in Programming

www.elsevier.com/locate/jlamp

Reduction semantics in Markovian process algebra

Mario Bravetti

University of Bologna, Department of Computer Science and Engineering / FOCUS INRIA, Mura Anteo Zamboni 7, 40126 Bologna, Italy

A R T I C L E   I N F O

Article history:
Received 26 July 2017
Received in revised form 28 November 2017
Accepted 2 January 2018
Available online 2 February 2018

Keywords:
Process algebra
Structural operational semantics
Markov chains

A B S T R A C T

Markovian process algebras allow for performance analysis by automatic generation of Continuous Time Markov Chains. The inclusion of exponential distribution rate information in process algebra terms, however, causes non-trivial issues to arise in the definition of their semantics. As a consequence, technical settings previously considered do not make it possible to base Markovian semantics on directly computing reductions between communicating processes: this would require the ability to readjust processes, i.e. a commutative and associative parallel operator and a congruence relation on terms enacting such properties. Semantics in reduction style is, however, often used for complex languages, due to its simplicity and conciseness. In this paper we introduce a new technique based on stochastic binders that allows us to define Markovian semantics in the presence of such a structural congruence. As application examples, we define the reduction semantics of Markovian versions of the π-calculus, considering both the cases of Markovian durations: being additional standalone prefixes (as in Interactive Markov Chains) and being, instead, associated to standard synchronizable actions, giving them a duration (as in Stochastic π-calculus). Notably, in the latter case, we obtain a "natural" Markovian semantics for π-calculus (CCS) parallel that preserves, for the first time, its associativity. In both cases we show our technique for defining reduction semantics to be correct with respect to the standard Markovian one (in labeled style) by providing Markovian extensions of the classical π-calculus Harmony theorem.

1. Introduction

The importance of considering probabilistic and timing aspects in system specification and analysis is widely recognized: first of all, the behavior of distributed systems and communication protocols often depends on probability and time; second, expressing duration of system activities makes it possible to estimate system performance. Markovian process algebras (see, e.g., [17,1,21,16,7,4]) extend classical process algebras with probabilistic exponentially distributed time durations denoted by rates λ (positive real numbers), where λ is the parameter of an exponential distribution. Defining an operational semantics for a Markovian process algebra makes it possible to derive Continuous Time Markov Chains from system specifications, which can then be analyzed for performance. Essentially this requires generating, from rates λ expressed syntactically in the prefixes of the process algebra, transitions labeled by rates.

Markovian process algebras in the literature have been defined in technical settings where it is not possible to compute their transitions by readjusting parallel processes, i.e. to represent their evolution by (not action labeled) *reduction* transitions (see, e.g., the reduction semantics of π-calculus [22,19]). This is due to limitations in existing approaches to

E-mail address: mario.bravetti@unibo.it.

https://doi.org/10.1016/j.jlamp.2018.01.002
2352-2208/

Markovian operational semantics that make them not compatible with a *structural congruence relation*. In reduction semantics such a relation is endowed with commutative and associative laws of parallel that allow communicating processes to get syntactically adjacent to each other in order to directly produce reduction transitions. The incompatibility arises from the techniques used for managing *multiplicity of rate transitions*: a Markovian semantics, in order to be correct must express multiplicity of several identical transitions, that is transitions with the same rate λ and source and target terms. Process algebras enriching prefixes with Markovian rates λ previously introduced in the literature use, instead, semantical definitions in the *labeled operational semantics* style: transitions are labeled with actions (representing potential of communication) and actual communications are produced by matching transitions of parallel processes according to their action labels. Reduction semantics is, however, widely used, since, due to its simplicity and conciseness, it is convenient for defining semantics of complex languages, see, e.g., [18].

Moreover, for Markovian process algebras where CCS/π-calculus parallel is used and both output and input actions are quantified by rates/weights, existing approaches in the literature (e.g. stochastic π-calculus of [21]) do not even guarantee associativity of parallel. That is in general, unless specific restrictions on the structure of terms are assumed (see [9]), different rates are obtained for outgoing transitions of terms $(P|Q)|R$ and $P|(Q|R)$.

In this paper we present a technique that allows us, for the first time, to develop semantics for Markovian process algebra in reduction style, i.e., where transitions are not labeled with actions and are directly produced by readjusting the structure of terms via a structural congruence relation. This opens the new possibility to include rates λ in actions of complex languages like that of [18] with no need to completely change the semantics from reduction to labeled style and without losing important properties like associativity of parallel. Notably, for the above mentioned case of Markovian process algebras where both output and input actions are quantified by rates/weights, our reduction semantics is the first Markovian one preserving *associativity of CCS/π-calculus parallel*.

More precisely, we consider stochastic versions of the π-calculus, encompassing both the case of rates being additional standalone prefixes and rates being, instead, attached to standard action prefixes. In the former case, *Markovian delay* prefixes (λ) are expressed separately from standard actions (that are unmodified) as in the Interactive Markov Chains approach of [16]. In the latter case, we instead decorate π-calculus output and τ prefixes with a λ rate, denoting *Markovian actions*. We consider two variants for this case. In the first variant input prefixes are not explicitly quantified (they are syntactically unmodified) and τ resulting from the synchronization of a λ decorated output on channel "a" and an input on "a" simply has rate λ: this corresponds to multiplying λ by the number of possible synchronizations over "a", similarly to what is done in the context of biochemistry, see e.g. [8]. In the second variant input prefixes are, instead, endowed with weights w, i.e. numbers establishing the probability to choose an input on channel "a" given that "a" is the channel of the selected output: this corresponds to subdividing the output rate λ among the possible "a" synchronizing inputs according to their weights, as done in Stochastic π-calculus [21] by taking inputs to have (weight equipped) unspecified rates [17,21].

In all above cases, we first define the semantics of the Markovian calculus in labeled operational semantics style (we use the early semantics of the π-calculus) using the classical approach of [14,21] to deal with multiplicity of identical transitions. We then apply our technique, which, as we will see, is based on the use of so-called *stochastic names* and *stochastic binders* (for both rates and weights) to provide a reduction semantics for the same process algebra.

We finally show our approach to correctly calculate rates (manage their multiplicity) by means of a theorem similar to the Harmony Lemma in [22,19], that is: stochastic reductions of reduction semantics are in correspondence with Markovian (τ) transitions of the labeled operational semantics and, thus, the underlying Markov chains. We show such a theorem to hold both for Markovian delays and actions, but not for the variant with input weights: this is related to the fact that parallel of Stochastic π-calculus [21] is *not* associative. If, e.g., in [21] we consider $P|(P'|P'')$ with $P = \overline{x}{<}y{>}_3.Q$, $P' = x(z)_4.Q'$ and $P'' = x(z')_8.Q''$ (we represent a prefix rate as a subscript), we get two distinguished τ transitions with rates 1 and 2. That is, rate 3 of the output multiplied by the probability to choose, inside $P'|P''$ (the subterm to the right of the parallel where synchronization happens): either input $x(z)_4$, i.e. 4/12 (the rate of the input divided by the total rate of inputs on x in $P'|P''$); or input $x(z')_8$ in $P'|P''$, i.e. 8/12 (similarly calculated). Rates of inputs are therefore treated as weights, according to which the output rate 3 is distributed. If, instead, we consider $(P|P')|P''$ [21] yields two distinguished τ transitions with the same rate 3. That is, rate 3 of the output multiplied by the probability: to choose $x(z)_4$ inside P', i.e. 4/4 (the rate of the input divided by the total rate of inputs on x in P'); and to choose input $x(z')_8$ inside P'', i.e. 8/8.[1]

As shown also in [12] non-associativity of parallel comes from the fact that [21] calculates rates, upon synchronization, in a way that strictly depends on a fixed structure for parallel operators in the term (independently of the particular labeled semantic technique used to deal with rates – [9], [12] or the original [21]): the rate of a synchronization on channel x performable by $P_1 \mid P_2$, depends on the total rate of input/output transitions on x performable by P_1 and P_2, called *apparent rate* in [17,21]. By using our reduction semantics technique, instead, the calculation of the total weight is not based on such a fixed structure, but it is the sum of weights of *all* synchronizable input actions (independently of how parallel is associated), hence *parallel is dealt with in an associative way*: in the example above we consider for the calculation of the total weight all input actions synchronizable with $\overline{x}{<}y{>}_3$, thus getting two reductions with rates 1 and 2 independently on how parallel is associated. This is natural in the context of π-calculus (or CCS) parallel $P_1|P_2$ in which, differently from

[1] In this example rates of inputs are treated as weights because, for all parallel operators $P_1|P_2$ considered, the total rate of outputs on x in P_1 is smaller than the total rate of input on x in P_2 (see [21]): the same is obtained by taking rates of inputs as (weight equipped) unspecified rates of [17,21].