SOS Specifications of Probabilistic Systems by Uniformly Continuous Operators

Daniel Gebler[1] and Simone Tini[2]

1 VU University Amsterdam, The Netherlands
2 University of Insubria, Italy

Abstract

Compositional reasoning over probabilistic systems wrt. behavioral metric semantics requires the language operators to be uniformly continuous. We study which SOS specifications define uniformly continuous operators wrt. bisimulation metric semantics. We propose an expressive specification format that allows us to specify operators of any given modulus of continuity. Moreover, we provide a method that allows to derive from any given specification the modulus of continuity of its operators.

1998 ACM Subject Classification F.3.2 Semantics of Programming Languages

Keywords and phrases SOS, probabilistic process algebra, bisimulation metric semantics, compositional metric reasoning, uniform continuity

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2015.155

1 Introduction

Probabilistic programming languages are languages that incorporate probabilistic choice as a primitive. They allow us to describe probabilistic concurrent communicating systems. The operational semantics of those languages is usually described by Structural Operational Semantics (SOS) specifications. A SOS specification assigns to each language expression a transition system with transitions inductively defined by means of SOS rules [19, 7].

As behavioral semantics we consider bisimulation metric [9, 22], which is the quantitative analogue of bisimulation equivalence [17] and assigns to each pair of processes a distance which measures the proximity of their quantitative properties. Compositional reasoning over probabilistic processes and probabilistic programs requires that the language operators are uniformly continuous [12]. Uniform continuity ensures that a small variance in the behavior of the parts leads to a bounded small variance in the behavior of the composed processes.

A successful approach to study systematically compositionality properties is the structural analysis of SOS language specifications [1, 19]. In this approach one analyses SOS specifications that satisfy desired compositionality properties and proposes syntactic SOS rule and specification templates that ensure by construction the compositionality property.

In this paper we develop an expressive SOS specification format guaranteeing that the specified operators are uniformly continuous. The format allows us to specify for each operator its respective modulus of continuity. Our fundamental insight is that an operator is uniformly continuous if it is Lipschitz continuous for each finite projection. The SOS specification format derives then from the definition of Lipschitz factors of the finite projections the guarantee that the specified operator is uniformly continuous. Furthermore, we develop a method to derive from any modulus of continuity the respective syntactic requirements on the specifications ensuring that the specified operators satisfy this modulus of continuity. Moreover, we develop a novel method to derive from any SOS specification the modulus of

26th International Conference on Concurrency Theory (CONCUR 2015).
Editors: Luca Aceto and David de Frutos Escrig; pp. 155–168
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

continuity of its operators. The Lipschitz factor of some operator wrt. the k-th projection, i.e., wrt. the up-to-k bisimulation metric, is determined by the replication of processes in the first k steps, the probabilistic choices in those steps, and the (step) discount of the bisimulation metric. Hence, our analysis provides further insights in the interplay between those determining factors. Our key contributions are:

1. We develop an expressive SOS specification format guaranteeing that all specified operators are uniformly continuous (Thm. 28).
2. We provide a method that allows us to derive for any uniformly continuous operator its respective modulus of continuity from its specification rules (Thms. 27 and 28).
3. We provide a method that, given any modulus of continuity, determines sufficient syntactic requirements s.t. any specification satisfying these requirements defines an operator with that modulus of continuity (Thm. 32).
4. We show by appropriate examples that our SOS specification formats and syntactic requirements cannot be relaxed in any obvious way (Exs. 10–13).
5. We apply those results and derive an upper bound on the distance between language expressions from the syntactic properties of the operators (Thm. 35). This enables metric compositional reasoning over partial program specification [12].

The paper is organized as follows. In Section 2 we recall the necessary technical definitions. In Section 3 we prove that an operator is uniformly continuous if it is Lipschitz continuous for each finite projection. In Section 4 we discuss which structural patterns of SOS rules define uniformly continuous operators. In Section 5 we present our format for uniformly continuous operators. In Section 6 we develop our method to derive from any modulus of continuity the respective syntactic requirements on the specifications ensuring that the specified operators satisfy this modulus of continuity. In Section 7 we show how to apply our results to derive an upper bound on the distance between language expressions from the syntactic properties of the operators. We conclude in Section 8 and discuss possible future work.

2 Preliminaries

The operational semantics of programming languages and process algebras is usually given as a transition system with language expressions (terms) as states and a transition relation inductively defined by means of SOS rules.

Probabilistic Transition Systems. A *signature* is a structure $\Sigma = (F, \mathsf{r})$, where F is a countable set of *operators*, and $\mathsf{r}\colon F \to \mathbb{N}$ is a *rank function*. We will use n for $\mathsf{r}(f)$ if it is clear from the context. By $f \in \Sigma$ we mean $f \in F$. We assume an infinite set of *state variables* $\mathcal{V}_s$. The set of *state terms* over a signature Σ and a set of state variables $V \subseteq \mathcal{V}_s$, notation $\mathsf{T}(\Sigma, V)$, is defined as usual. The set of *closed state terms* $\mathsf{T}(\Sigma, \emptyset)$ is abbreviated as $\mathsf{T}(\Sigma)$. The set of *open state terms* $\mathsf{T}(\Sigma, \mathcal{V}_s)$ is abbreviated as $\mathbb{T}(\Sigma)$.

Probabilistic transition systems extend transition systems by allowing for probabilistic choices in the transitions. We consider probabilistic nondeterministic labelled transition systems [20]. As state space we take the set of all closed terms $\mathsf{T}(\Sigma)$. Probability distributions over this space are mappings $\pi\colon \mathsf{T}(\Sigma) \to [0,1]$ with $\sum_{t \in \mathsf{T}(\Sigma)} \pi(t) = 1$ that assign to each term $t \in \mathsf{T}(\Sigma)$ its respective probability $\pi(t)$. By $\Delta(\mathsf{T}(\Sigma))$ we denote the set of all probability distributions on $\mathsf{T}(\Sigma)$. We let π, π' range over $\Delta(\mathsf{T}(\Sigma))$.

▶ **Definition 1** (PTS [20]). A *probabilistic nondeterministic labeled transition system* (*PTS*) is given by a triple $(\mathsf{T}(\Sigma), A, \to)$, where Σ is a signature, A is a countable set of *actions*, and $\to \subseteq \mathsf{T}(\Sigma) \times A \times \Delta(\mathsf{T}(\Sigma))$ is a *transition relation*.

We write $t \xrightarrow{a} \pi$ for $(t, a, \pi) \in \rightarrow$. Let $der(t, a) = \{\pi \in \Delta(\mathsf{T}(\Sigma)) \mid t \xrightarrow{a} \pi\}$.

Bisimulation metric. Bisimulation metrics are the quantitative analogue to bisimulation equivalences. Let $([0,1]^{\mathsf{T}(\Sigma)\times\mathsf{T}(\Sigma)}, \sqsubseteq)$ be the complete lattice of functions $d, d' : \mathsf{T}(\Sigma)\times\mathsf{T}(\Sigma) \to [0,1]$ ordered by $d \sqsubseteq d'$ iff $d(t,t') \leq d'(t,t')$ for all terms $t, t' \in \mathsf{T}(\Sigma)$. The bottom element $\mathbf{0}$ is the constant zero function $\mathbf{0}(t,t') = 0$. A function $d \colon \mathsf{T}(\Sigma) \times \mathsf{T}(\Sigma) \to [0,1]$ is a *1-bounded pseudometric* if $d(t,t) = 0$, $d(t,t') = d(t',t)$ and $d(t,t'') \leq d(t,t') + d(t',t'')$ for all $t, t', t'' \in \mathsf{T}(\Sigma)$. Intuitively, the bisimilarity metric will be a 1-bounded pseudometric d with $d(t,t')$ measuring the maximal distance of quantitative properties between t and t'.

We define now the bisimilarity metric as least fixed point of a monotone function over $([0,1]^{\mathsf{T}(\Sigma)\times\mathsf{T}(\Sigma)}, \sqsubseteq)$ [8]. A pseudometric on terms $\mathsf{T}(\Sigma)$ is lifted to a pseudometric on distributions $\Delta(\mathsf{T}(\Sigma))$ by the Kantorovich pseudometric. This lifting corresponds to the lifting of bisimulation equivalence relations on terms to bisimulation equivalence relations on distributions [22]. A *matching* for a pair of distributions $(\pi, \pi') \in \Delta(\mathsf{T}(\Sigma)) \times \Delta(\mathsf{T}(\Sigma))$ is a distribution over the product state space $\omega \in \Delta(\mathsf{T}(\Sigma) \times \mathsf{T}(\Sigma))$ with π and π' as left and right marginal, i.e., $\sum_{t' \in \mathsf{T}(\Sigma)} \omega(t,t') = \pi(t)$ and $\sum_{t \in \mathsf{T}(\Sigma)} \omega(t,t') = \pi'(t')$ for all terms $t, t' \in \mathsf{T}(\Sigma)$. Let $\Omega(\pi, \pi')$ denote the set of all matchings for (π, π'). The *Kantorovich pseudometric* $\mathbf{K}(d) \colon \Delta(\mathsf{T}(\Sigma)) \times \Delta(\mathsf{T}(\Sigma)) \to [0,1]$ of pseudometric $d \colon \mathsf{T}(\Sigma) \times \mathsf{T}(\Sigma) \to [0,1]$ is defined for all distributions $\pi, \pi' \in \Delta(\mathsf{T}(\Sigma))$ by

$$\mathbf{K}(d)(\pi, \pi') = \min_{\omega \in \Omega(\pi,\pi')} \sum_{t,t' \in \mathsf{T}(\Sigma)} d(t,t') \cdot \omega(t,t').$$

In order to capture nondeterministic choices, we need to lift pseudometrics on distributions to pseudometrics on sets of distributions. The *Hausdorff pseudometric* $\mathbf{H}(\hat{d}) \colon P(\Delta(\mathsf{T}(\Sigma))) \times P(\Delta(\mathsf{T}(\Sigma))) \to [0,1]$ is defined for $\Pi_1, \Pi_2 \subseteq \Delta(\mathsf{T}(\Sigma))$ and $\hat{d} \colon \Delta(\mathsf{T}(\Sigma)) \times \Delta(\mathsf{T}(\Sigma)) \to [0,1]$ by

$$\mathbf{H}(\hat{d})(\Pi_1, \Pi_2) = \max\left\{ \sup_{\pi_1 \in \Pi_1} \inf_{\pi_2 \in \Pi_2} \hat{d}(\pi_1, \pi_2), \sup_{\pi_2 \in \Pi_2} \inf_{\pi_1 \in \Pi_1} \hat{d}(\pi_2, \pi_1) \right\}$$

with $\inf \emptyset = 1$, $\sup \emptyset = 0$.

Now we define $\mathbf{B} \colon [0,1]^{\mathsf{T}(\Sigma)\times\mathsf{T}(\Sigma)} \to [0,1]^{\mathsf{T}(\Sigma)\times\mathsf{T}(\Sigma)}$ by

$$\mathbf{B}(d)(t,t') = \sup_{a \in A} \{\mathbf{H}(\lambda \cdot \mathbf{K}(d))(der(t,a), der(t',a))\}$$

for $d \colon \mathsf{T}(\Sigma) \times \mathsf{T}(\Sigma) \to [0,1]$, $t, t' \in \mathsf{T}(\Sigma)$, $\lambda \in (0,1]$ a discount factor[1], and $(\lambda \cdot \mathbf{K}(d))(\pi, \pi') = \lambda \cdot \mathbf{K}(d)(\pi, \pi')$. $\mathbf{B}$ is a monotone function over $([0,1]^{\mathsf{T}(\Sigma)\times\mathsf{T}(\Sigma)}, \sqsubseteq)$. Prefixed points $\mathbf{B}(d) \sqsubseteq d$ are pseudometrics satisfying the bisimulation transfer condition (for all pairs t and t' each transition from t can be mimicked by an equally labelled transition from t' s.t. the distance between the accessible distributions does not exceed the distance between t and t'). By the Knaster-Tarski theorem $\mathbf{B}$ has a least fixed point, which forms the bisimilarity metric.

▶ **Definition 2** (Bisimilarity metric [9, 8]). We call $\mathbf{d}_k = \mathbf{B}^k(\mathbf{0})$ the *up-to-k bisimilarity metric*, and $\mathbf{d} = \lim_{k\to\infty} \mathbf{d}_k$ the *bisimilarity metric*.

By bisimulation distance between t and t' we mean $\mathbf{d}(t,t')$. Bisimulation equivalence [20] is the kernel of the bisimilarity metric [9].

[1] By means of the *discount factor* $\lambda \in (0,1]$ we allow to specify how much the behavioral distance of future transitions is taken into account.

▶ **Example 3.** Consider the probabilistic CCS [23, 15, 12] terms $s = a.a^\omega$ and $t_e = a.([1 - e]a^\omega \oplus [e]0)$, with $e \in (0,1)$. Process a^w performs a forever. The transitions $s \xrightarrow{a} \pi_s$, with $\pi_s(a^\omega) = 1$, and $t_e \xrightarrow{a} \pi_t$, with $\pi_t(a^\omega) = 1-e$ and $\pi_t(0) = e$, are derivable. Then $\mathbf{d}(a^\omega, a^\omega) = 0$ and $\mathbf{d}(a^\omega, 0) = 1$. Hence $\mathbf{K}(\mathbf{d})(\pi_s, \pi_t) = e$. Thus, $\mathbf{d}_0(s, t_e) = 0$ and $\mathbf{d}_k(s, t_e) = \lambda e$ if $k \geq 1$. Finally, we get $\mathbf{d}(s, t_e) = \lambda e$.

Algebra of probability distributions. By $\delta(t)$ with $t \in \mathsf{T}(\Sigma)$ we denote the *Dirac distribution* defined by $(\delta(t))(t) = 1$. The convex combination $\sum_{i\in I} p_i\pi_i$ of a family $\{\pi_i\}_{i\in I}$ of distributions $\pi_i \in \Delta(\mathsf{T}(\Sigma))$ with $p_i \in (0,1]$ and $\sum_{i\in I} p_i = 1$ is defined by $(\sum_{i\in I} p_i\pi_i)(t) = \sum_{i\in I}(p_i\pi_i(t))$ for all $t \in \mathsf{T}(\Sigma)$. The expression $f(\pi_1, \ldots, \pi_n)$ with $f \in \Sigma$ and $\pi_i \in \Delta(\mathsf{T}(\Sigma))$ denotes the product distribution defined by $f(\pi_1, \ldots, \pi_n)(f(t_1, \ldots, t_n)) = \prod_{i=1}^n \pi_i(t_i)$.

In order to describe probabilistic behavior, we need syntactic expressions that denote probability distributions. We assume an infinite set of *distribution variables* $\mathcal{V}_d$. We let μ, ν range over $\mathcal{V}_d$. We denote by $\mathcal{V}$ the set of state and distribution variables $\mathcal{V} = \mathcal{V}_s \cup \mathcal{V}_d$. We let ζ, ζ' range over $\mathcal{V}$. The set of *distribution terms* over a set of state variables $V_s \subseteq \mathcal{V}_s$ and a set of distribution variables $V_d \subseteq \mathcal{V}_d$, notation $\mathsf{DT}(\Sigma, V_s, V_d)$, is the least set satisfying [6]:

(i) $V_d \subseteq \mathsf{DT}(\Sigma, V_s, V_d)$,
(ii) $\{\delta(t) \mid t \in \mathsf{T}(\Sigma, V_s)\} \subseteq \mathsf{DT}(\Sigma, V_s, V_d)$,
(iii) $\sum_{i\in I} p_i\theta_i \in \mathsf{DT}(\Sigma, V_s, V_d)$ whenever $\theta_i \in \mathsf{DT}(\Sigma, V_s, V_d)$ and $p_i \in (0,1]$ with $\sum_{i\in I} p_i = 1$, and
(iv) $f(\theta_1, \ldots, \theta_n) \in \mathsf{DT}(\Sigma, V_s, V_d)$ whenever $f \in \Sigma$ and $\theta_i \in \mathsf{DT}(\Sigma, V_s, V_d)$.

We write $\theta_1 \oplus_p \theta_2$ for $\sum_{i=1}^2 p_i\theta_i$ with $p_1 = p$ and $p_2 = 1-p$. Furthermore, we write $\theta_1 \, f \, \theta_2$ for $f(\theta_1, \theta_2)$. We write $\mathbb{DT}(\Sigma)$ for $\mathsf{DT}(\Sigma, \mathcal{V}_s, \mathcal{V}_d)$ (set of all *open distribution terms*), and $\mathsf{DT}(\Sigma)$ for $\mathsf{DT}(\Sigma, \emptyset, \emptyset)$ (set of all *closed distribution terms*).

Distribution terms have the following meaning. A *distribution variable* $\mu \in \mathcal{V}_d$ is a variable that takes values from $\Delta(\mathsf{T}(\Sigma))$. An *instantiable Dirac distribution* $\delta(t)$ instantiates to $\delta(t')$ if t instantiates to t'. Case (iii) allows to construct convex combinations of distributions. Case (iv) lifts the structural inductive construction of state terms to distribution terms. Substitutions are defined as usual [7].

SOS specification. We specify the operational semantics of operators by SOS rules. SOS rules are syntax-driven inference rules that define the behavior of complex expressions in terms of the behavior of their components. We employ SOS rules of the probabilistic GSOS format [4, 7, 18, 6]. This format uses triples of the form $t \xrightarrow{a} \theta$ that specify in a single literal all probabilistic choices of a transition. Earlier formats [3, 16, 21] used the old fashion quadruple $t \xrightarrow{a,p} t'$ that decorates the transitions with both the action label and the probability in order to partially specify a probabilistic jump. However, this approach required complicated consistency conditions on the set of all rules to ensure that the partially specified probabilistic jumps define in total probabilistic choices.

▶ **Definition 4** (SOS rule). A *SOS rule* r has the form:

$$\frac{\{x_i \xrightarrow{a_{i,k}} \mu_{i,k} \mid i \in I, k \in K_i\} \qquad \{x_i \stackrel{b_{i,l}}{\not\longrightarrow} \mid i \in I, l \in L_i\}}{f(x_1, \ldots, x_n) \xrightarrow{a} \theta}$$

with n the rank of operator $f \in \Sigma$, $I = \{1, \ldots, n\}$ the indices of the arguments of f, K_i, L_i finite index sets, $a_{i,k}, b_{i,l}, a \in A$ actions, $x_i \in \mathcal{V}_s$ state variables, $\mu_{i,k} \in \mathcal{V}_d$ distribution variables, and $\theta \in \mathbb{DT}(\Sigma)$ a distribution term. Furthermore, all $\mu_{i,k}$ for $i \in I, k \in K_i$ are pairwise different, all $x_1, \ldots, x_n$ are pairwise different, and all variables in θ are from $\{\mu_{i,k} \mid i \in I, k \in K_i\} \cup \{x_1 \ldots, x_n\}$.

The expressions $x_i \xrightarrow{a_{i,k}} \mu_{i,k}$, $x_i \stackrel{b_{i,l}}{\not\longrightarrow}$ and $f(x_1,\dots,x_n) \xrightarrow{a} \theta$ are called, resp., *positive premises*, *negative premises* and *conclusion*. The set of all premises is denoted by $\mathrm{prem}(r)$. The term $f(x_1,\dots,x_n)$ is called the *source*, the variables $x_1,\dots,x_n$ are called *source variables*, and the distribution term θ is called the *target* (notation $\mathrm{trgt}(r)$). Let $\mathrm{der}(r,x_i) = \{\mu_{i,k} \mid x_i \xrightarrow{a_{i,k}} \mu_{i,k} \in \mathrm{prem}(r)\}$. We call $\mu \in \mathrm{der}(r,x_i)$ a *derivative* of source variable x_i.

A *probabilistic transition system specification* (PTSS) is a triple $P = (\Sigma, A, R)$, where Σ is a signature, A is a countable set of actions and R is a countable set of SOS rules. We denote by R_f the set of rules specifying operator f, i.e., all rules of R with source $f(x_1,\dots,x_n)$. The unique model of P is a PTS $(\mathsf{T}(\Sigma), A, \to)$, with transitions in $\to$ all and only those for which P offers a justification [7].

Intuitively, a term $f(t_1,\dots,t_n)$ represents the composition of terms $t_1,\dots,t_n$ by operator f. A rule r specifies some transition $f(t_1,\dots,t_n) \xrightarrow{a} \pi$ that represents the evolution of the composed term $f(t_1,\dots,t_n)$ by action a to the distribution π. We say that a rule with conclusion $f(x_1,\dots,x_n) \xrightarrow{a} \theta$ *delays* the evolution of the source term x_i if x_i appears in θ, and that the source term x_i *evolves* to $\mu \in \mathrm{der}(r,x_i)$ if μ appears in θ. We say that r *replicates* a source variable x_i if multiple instances of either x_i or x_i-derivatives in $\mathrm{der}(r,x_i)$ appear in the target θ of rule r.

3 Uniform continuity

In order to specify and reason about probabilistic systems in a compositional manner, it is necessary that the operators describing these systems are uniformly continuous [12]. A uniformly continuous operator ensures that a small variance in the behavior of a system component leads to a bounded small variance in the behavior of the composed system. We assume some fixed PTSS $P = (\Sigma, A, R)$.

▶ **Definition 5** (Modulus of continuity)**.** Let $f \in \Sigma$ be some n-ary operator and d be any 1-bounded pseudometric on $\mathsf{T}(\Sigma)$. A mapping $\omega\colon [0,1]^n \to [0,1]$ is an *upper bound on the distance between f-composed terms wrt. d* if for all terms $s_i, t_i \in \mathsf{T}(\Sigma)$

$$d(f(s_1,\dots,s_n), f(t_1,\dots,t_n)) \le \omega(d(s_1,t_1),\dots,d(s_n,t_n)).$$

An upper bound ω of f wrt. d is a *modulus of continuity of f wrt. d* if ω is continuous at $(0,\dots,0)$, i.e., $\lim_{(\epsilon_1,\dots,\epsilon_n)\to(0,\dots,0)} \omega(\epsilon_1,\dots,\epsilon_n) = \omega(0,\dots,0)$, and $\omega(0,\dots,0) = 0$.

▶ **Definition 6** (Uniformly continuous operator)**.** Let d be any 1-bounded pseudometric on $\mathsf{T}(\Sigma)$. An operator $f \in \Sigma$ is

1. *uniformly continuous wrt. d* if f admits some modulus of continuity wrt. d,
2. *L-Lipschitz continuous wrt. d* with $L \in \mathbb{R}_{\ge 0}$ if $\omega(\epsilon_1,\dots,\epsilon_n) = L\sum_{i=1}^n \epsilon_i$ is a modulus of continuity of f wrt. d, and
3. *Lipschitz continuous wrt. d* if f is L-Lipschitz continuous wrt. d for some $L \in \mathbb{R}_{\ge 0}$.

▶ **Example 7.** Consider the synchronous parallel composition operator specified by

$$\frac{x \xrightarrow{a} \mu \quad y \xrightarrow{a} \nu}{x \mid y \xrightarrow{a} \mu \mid \nu}$$

and terms s and t_e as in Ex. 3. Recall that $\mathbf{d}(s,t_e) = \lambda e$. The transitions $s \mid s \xrightarrow{a} \pi_s$, with $\pi_s = \delta(a^\omega \mid a^\omega)$, and $t_{e_1} \mid t_{e_2} \xrightarrow{a} \pi_t$, with $\pi_t = (1-e_1)(1-e_2)\delta(a^\omega \mid a^\omega) + e_1(1-e_2)\delta(0 \mid a^\omega) + (1-e_1)e_2\delta(a^\omega \mid 0) + e_1e_2\delta(0 \mid 0)$, are derivable. Then, $\mathbf{d}(s \mid s, t_{e_1} \mid t_{e_2}) =$

$\lambda\,\mathbf{K}(\mathbf{d})(\pi_s, \pi_t) = \lambda(1-(1-e_1)(1-e_2)) \leq \lambda e_1 + \lambda e_2 = \mathbf{d}(s, t_{e_1}) + \mathbf{d}(s, t_{e_2})$. Thm. 27 below will confirm that $\omega(\epsilon_1, \epsilon_2) = \epsilon_1 + \epsilon_2$ is a modulus of continuity of the synchronous parallel composition operator wrt. $\mathbf{d}$. Hence, this operator is 1-Lipschitz continuous.

The behavioral distance between two arbitrary terms s and t can be divided in the distance observable by the first k steps and the distance observable after step k. The distance observable after step k is bounded by λ^k.

▶ **Proposition 8.** *Let $s, t \in \mathcal{T}(\Sigma)$. Then $\mathbf{d}(s,t) \leq \mathbf{d}_k(s,t) + \lambda^k$ for all $k \in \mathbb{N}$.*

A fundamental insight that we will use later to define the SOS specification format is that an operator is uniformly continuous wrt. the bisimilarity metric if this operator is Lipschitz continuous wrt. all up-to-k bisimilarity metrics.

▶ **Theorem 9.** *Assume $\lambda < 1$. If an operator $f \in \Sigma$ is Lipschitz continuous wrt. $\mathbf{d}_k$ for each $k \in \mathbb{N}$, then f is uniformly continuous wrt. $\mathbf{d}$.*

Hence, we assume now a strictly discounting bisimulation metric with $\lambda < 1$.

4 Analysis of uniformly continuous operators

We analyze now the structural patterns of SOS rules that define uniformly continuous operators and give representative examples of rules that specify operators that are not uniformly continuous. Moreover, we derive from the structural properties of the rules the moduli of continuity of the specified operators.

▶ **Example 10** (Non-recurring process replication)**.** Consider the rules

$$\frac{x \xrightarrow{a} \mu}{f(x) \xrightarrow{a} \theta} \qquad\qquad \frac{x \xrightarrow{a} \mu \quad y \xrightarrow{a} \nu}{x \mid y \xrightarrow{a} \mu \mid \nu}$$

with $\theta \in \mathbb{DT}(\Sigma)$ some distribution term. We analyze for various distribution terms θ the modulus of continuity of the specified operator f. We use again the terms s and t_e from Ex. 3. Recall that $\mathbf{d}(s, t_e) = \lambda e$.

Consider $\theta = \delta(x \mid x)$. The operator f replicates the source process x, delays both instances, and lets them evolve in parallel. The transitions $f(s) \xrightarrow{a} \delta(s \mid s)$ and $f(t_e) \xrightarrow{a} \delta(t_e \mid t_e)$ are derivable. It follows that $\mathbf{d}(f(s), f(t_e)) = \lambda\,\mathbf{K}(\mathbf{d})(\delta(s \mid s), \delta(t_e \mid t_e)) = \lambda \mathbf{d}(s \mid s, t_e \mid t_e) \leq 2\lambda \mathbf{d}(s, t_e)$ (c.f. Ex. 7). Thm. 27 below will confirm that $\omega(\epsilon) = (2\lambda)\epsilon$ is a modulus of continuity of this specification of f.

Consider $\theta = (\delta(x \mid x) \oplus_r \delta(0))$ for some $r \in (0,1)$. The operator f replicates and delays now only with probability r the source process x. The transitions $f(s) \xrightarrow{a} r\delta(s \mid s) + (1-r)\delta(0)$ and $f(t_e) \xrightarrow{a} r\delta(t_e \mid t_e) + (1-r)\delta(0)$ are derivable. Hence, $\mathbf{d}(f(s), f(t_e)) = \lambda r \mathbf{d}(s \mid s, t_e \mid t_e) \leq 2r\lambda \mathbf{d}(s, t_e)$. Thm. 27 below will confirm that $\omega(\epsilon) = (2r\lambda)\epsilon$ is a modulus of continuity of this specification of f.

Consider $\theta = (\mu \mid \mu) \oplus_r \delta(0)$. The operator f replicates (but does not delay) with probability r the source process x. The evolved instances proceed in parallel. The transitions $f(s) \xrightarrow{a} r\delta(a^\omega \mid a^\omega) + (1-r)\delta(0)$ and $f(t_e) \xrightarrow{a} r((1-e)^2\delta(a^\omega \mid a^\omega) + e(1-e)\delta(0 \mid a^\omega) + (1-e)e\delta(a^\omega \mid 0) + e^2\delta(0 \mid 0)) + (1-r)\delta(0)$ are derivable. Now $\mathbf{d}(f(s), f(t_e)) \leq 2r\lambda e = 2r\mathbf{d}(s, t_e)$. Thm. 27 below will confirm that $\omega(\epsilon) = (2r)\epsilon$ is a modulus of continuity of this specification of f.

In essence, Ex. 10 shows that the number of non-recurring process replications, weighted by the probability of their realization, and weighted by the discount factor if processes are delayed, determines the Lipschitz factor of the operator.

▶ **Example 11** (Linear process replication)**.** We proceed with the analysis of Ex. 10 and analyze the specification of recursive replication behavior.

Consider $\theta = \delta(f(x)) \mid \mu$. Note that this specification of f is precisely the π-calculus bang operator. The transitions $f(s) \xrightarrow{a} \pi_s$, with $\pi_s = \delta(f(s) \mid a^\omega)$, and $f(t_e) \xrightarrow{a} \pi_t$, with $\pi_t = (1-e)\delta(f(t_e) \mid a^\omega) + e\delta(f(t_e) \mid 0)$ are derivable. Then $\mathbf{d}(f(s), f(t_e)) = \lambda e + \lambda(1-e)\mathbf{d}(f(s) \mid a^\omega, f(t_e) \mid a^\omega)) = \lambda e + \lambda(1-e)\mathbf{d}(f(s), f(t_e))$. Hence, $\mathbf{d}(f(s), f(t_e)) = \frac{\lambda e}{1-\lambda+\lambda e} \leq \frac{\lambda e}{1-\lambda} = \frac{1}{1-\lambda}\mathbf{d}(s, t_e)$. Intuitively, the operator f spawns and delays in each computation step a new instance of the source process x. Thus, the total number of spawned (resp. discounted) process copies is $\sum_{k=0}^{\infty} \lambda^k = 1/(1-\lambda)$. Hence, $\omega(\epsilon) = \frac{1}{1-\lambda}\epsilon$ (formally shown below by Thm. 27) is a modulus of continuity of this specification of f.

Consider $\theta = \delta(f(x)) \mid \mu \mid \delta(x)$. The specified operator f has the modulus of continuity $\omega(\epsilon) = \frac{1+\lambda}{1-\lambda}\epsilon$. Similarly, if $\theta = \delta(f(x)) \mid \mu \mid \delta(x) \mid \delta(x)$, then $\omega(\epsilon) = \frac{1+2\lambda}{1-\lambda}\epsilon$ is a modulus of continuity of the specified operator f.

In essence, Ex. 11 shows that if the number of recurring process replications is finitely bounded, then the specified operator is Lipschitz continuous.

▶ **Example 12** (Non-linear process replication)**.** We analyze now the fork operation of operating systems specified by the copy operator of [5, 11] with the rules

$$\frac{x \xrightarrow{a} \mu}{\mathsf{cp}(x) \xrightarrow{a} \mu}\,(a \notin \{l, r\}) \qquad\qquad \frac{x \xrightarrow{l} \mu \quad x \xrightarrow{r} \nu}{\mathsf{cp}(x) \xrightarrow{s} \mathsf{cp}(\mu) \mid \mathsf{cp}(\nu)}$$

Actions l and r are the left and right forking actions, and s is the resulting split action. The fork of t is the process $\mathsf{cp}(t)$ evolving by t to the parallel composition of the left fork (l-derivative of t) and the right fork (r-derivative of t). For all other actions $a \notin \{l, r\}$ the process $\mathsf{cp}(t)$ mimics the behavior of t.

First, we show that the copy operator is not Lipschitz continuous. Formally, for any $L \in \mathbb{R}_{\geq 0}$, we show that $\mathbf{d}(\mathsf{cp}(s), \mathsf{cp}(t)) > L\mathbf{d}(s, t)$ for some CCS processes s, t. Let $s_1 = l.([1-e]a \oplus [e]0) + r.([1-e]a \oplus [e]0)$ and $t_1 = l.a + r.a$, and $s_{k+1} = l.s_k + r.s_k$ and $t_{k+1} = l.t_k + r.t_k$. Clearly $\mathbf{d}(s_k, t_k) = \lambda^k e$. Then $\mathbf{d}(\mathsf{cp}(s_k), \mathsf{cp}(t_k)) = \lambda^k(1-(1-e)^{2^k})$. Hence, for any k with $2^k > L$, $\mathbf{d}(\mathsf{cp}(s), \mathsf{cp}(t))/\mathbf{d}(s, t) = (1-(1-e)^{2^k})/e > L$ holds for $s = s_k$, $t = t_k$ and all $0 < e < (2^k - L)/(2^{k-1}(2^k - 1))$. Thus, the copy operator is not Lipschitz continuous.

However, Thm. 27 below will confirm that $\omega(\epsilon) = \inf_{k \in \mathbb{N}}(2^k\epsilon + \lambda^k)$ is a (non-linear) modulus of continuity of the copy operator. Intuitively, the copy operator creates in k steps at most 2^k copies of the source process x, i.e., the copy operator is 2^k-Lipschitz continuous for the up-to-k bisimilarity metric. Then, by Prop. 8 we derive the modulus of continuity wrt. bisimilarity metric from the moduli of continuity of the up-to-k bisimilarity metrics.

In essence, Ex. 12 shows that an operator is uniformly continuous if in each step only finitely many process copies are spawned.

▶ **Example 13** (Non-uniformly continuous operators)**.** Consider the unary operators f and g specified by the following rules for all $k \in \mathbb{N}$:

$$\frac{x \xrightarrow{a} \mu}{f(x) \xrightarrow{a} \underbrace{\mu \mid \ldots \mid \mu}_{k-\text{times}}} \qquad \frac{}{g(x) \xrightarrow{a} \delta(\underbrace{h(\ldots h(}_{k-\text{times}} x)))} \qquad \frac{x \xrightarrow{a} \mu}{h(x) \xrightarrow{a} \mu \mid \mu}$$

We start with operator f. We get $\mathbf{d}(f(s), f(t_e)) = \sup_{k\in\mathbb{N}} \lambda(1-(1-e)^k) = \lambda$. The least upper bound on the distance between f-composed processes is $\omega(\epsilon) = \lambda$ if $\epsilon > 0$ and $\omega(0) = 0$. However, ω is not a modulus of continuity since it is not continuous at 0. Hence, operator f is not uniformly continuous.

We proceed with operator g. We get $\mathbf{d}(g(s), g(t_e)) = \sup_{k\in\mathbb{N}} \lambda^2(1-(1-e)^{2^k}) = \lambda^2$. Following the same line of reasoning as with operator f we conclude that operator g is not uniformly continuous.

In essence, Ex. 13 shows that an operator may be not uniformly continuous if there is no bound on the number of process copies it can spawn in a single step.

5 Specification of uniformly continuous operators

We develop now a specification format that allows us to specify uniformly continuous operators. We exploit Thm. 9 and specify uniformly continuous operators by defining suitable Lipschitz factors wrt. all up-to-k bisimilarity metrics.

5.1 Finite projection Lipschitz continuous operators

▶ **Definition 14** (Lipschitz factor assignment). We call a mapping[2] $L\colon (\mathbb{N}\times\Sigma) \to \mathbb{R}^{\infty}_{\geq 0}$ a *Lipschitz factor assignment* (*LFA*, for short) for operators in Σ. Let $\mathcal{L}_\Sigma$ be the set of all LFAs for Σ, with $L, M \in \mathcal{L}_\Sigma$ ordered $L \sqsubseteq M$ iff $L_k(f) \leq M_k(f)$ for all $k \in \mathbb{N}$ and $f \in \Sigma$.

Intuitively, $L_k(f)$ is either the Lipschitz factor of operator $f \in \Sigma$ wrt. $\mathbf{d}_k$, or ∞ if f is not Lipschitz continuous wrt. $\mathbf{d}_k$.

▶ **Proposition 15.** *$(\mathcal{L}_\Sigma, \sqsubseteq)$ is a complete lattice.*

It is clear that the bottom element of the lattice $(\mathcal{L}_\Sigma, \sqsubseteq)$ is the LFA $0 \in \mathcal{L}_\Sigma$ given by $0_k(f) = 0$ for all $k \in \mathbb{N}$ and $f \in \Sigma$.

▶ **Definition 16** (Semantic consistency). Let $L \in \mathcal{L}_\Sigma$ be a LFA and $k \in \mathbb{N}$. We call L *consistent with the up-to-k bisimilarity metric* $\mathbf{d}_k$ if

$$\mathbf{d}_k(f(s_1,\dots,s_n), f(t_1,\dots,t_n)) \leq L_k(f)\sum_{i=1}^{n}\mathbf{d}_k(s_i,t_i)$$

for all operators $f \in \Sigma$ and terms $s_i, t_i \in \mathsf{T}(\Sigma)$. Furthermore, we call L *consistent with the bisimilarity metric* $\mathbf{d}$ if L is consistent with $\mathbf{d}_k$ for all $k \in \mathbb{N}$.

Hence $L \in \mathcal{L}_\Sigma$ is consistent with $\mathbf{d}_k$ if each operator f with $L_k(f) < \infty$ is $L_k(f)$-Lipschitz continuous wrt. $\mathbf{d}_k$. We proceed by lifting LFAs from operators to terms.

▶ **Definition 17** (LFA on terms). Let $L \in \mathcal{L}_\Sigma$ be a LFA. The lifting of L is a *Lipschitz factor assignment on terms* given as the mapping $L\colon (\mathbb{N}\times(\mathbb{T}(\Sigma)\cup\mathbb{DT}(\Sigma))\times\mathcal{V}) \to \mathbb{R}^{\infty}_{\geq 0}$ defined by:

$$L_k(t,\zeta) = \begin{cases} 1 & \text{if } t = \zeta \\ L_k(f)\displaystyle\sum_{i=1}^{n} L_k(t_i,\zeta) & \text{if } t = f(t_1,\dots,t_n) \\ 0 & \text{otherwise} \end{cases}$$

[2] We will write the first argument of L as subscript, i.e., $L_k(f)$ for $L(k,f)$, to align with the notation $\mathbf{d}_k$ of up-to-k-bisimilarity metric.

$$L_k(\theta,\zeta) = \begin{cases} 1 & \text{if } \theta = \zeta \\ L_k(t,\zeta) & \text{if } \theta = \delta(t) \\ \sum_{i\in I} p_i \cdot L_k(\theta_i,\zeta) & \text{if } \theta = \sum_{i\in I} p_i\theta_i \\ L_k(f)\sum_{i=1}^{n} L_k(\theta_i,\zeta) & \text{if } \theta = f(\theta_1,\dots,\theta_n) \text{ and } \zeta \in \mathcal{V}_s \\ \overline{L_k(f)}\sum_{i=1}^{n} L_k(\theta_i,\zeta) & \text{if } \theta = f(\theta_1,\dots,\theta_n) \text{ and } \zeta \in \mathcal{V}_d \\ 0 & \text{otherwise} \end{cases}$$

with $\overline{L_k(f)} = \max(L_k(f), 1)$.

The Lipschitz factor of a state term arises from the functional composition of the Lipschitz moduli of continuity of the operators in the state term. Similarly, also for distribution terms except for operators with $L_k(f) < 1$ (case 5 of $L_k(\theta,\zeta)$). As shown in [13, Sec. 4.2], if f has a modulus of continuity on state terms below 1-Lipschitz continuity, then the modulus of continuity of f on distribution terms is 1-Lipschitz continuity (but not smaller).

The lifting of a LFA preserves consistency.

▶ **Proposition 18.** *Let $L \in \mathcal{L}_\Sigma$ be a LFA and $k \in \mathbb{N}$. If L is consistent with $\mathbf{d}_k$, then for any term $t \in \mathbb{T}(\Sigma)$ we have*

$$\mathbf{d}_k(\sigma_1(t), \sigma_2(t)) \leq \sum_{x\in\mathcal{V}_s} L_k(t,x)\cdot \mathbf{d}_k(\sigma_1(x),\sigma_2(x))$$

for all closed substitutions $\sigma_1, \sigma_2 \colon \mathcal{V} \to T(\Sigma)$.

The set of SOS rules R gives rise to a mapping $R\colon \mathcal{L}_\Sigma \to \mathcal{L}_\Sigma$ with $R(L)$ defined as the LFA obtained by applying the rules of R to L.

▶ **Definition 19** (R-extension)**.** The *R-extension of LFAs* is the mapping[3] $R\colon \mathcal{L}_\Sigma \to \mathcal{L}_\Sigma$ defined by

$$R(L)_0(f) = 0$$

$$R(L)_{k+1}(f) = \sup_{r\in R_f} \max_{i=1}^{\mathsf{r}(f)} \left(\lambda \cdot L_k(\mathrm{trgt}(r), x_i) + \sum_{\mu\in\mathrm{der}(r,x_i)} L_k(\mathrm{trgt}(r),\mu) \right)$$

for all $L \in \mathcal{L}_\Sigma$ and $f \in \Sigma$.

Intuitively, the lifted LFA on terms (Def. 17) is obtained by structural induction over terms, while the R-extended LFA (Def. 19) is obtained by operational induction over rules. The R-extension of Lipschitz factor assignments preserves semantic consistency.

▶ **Proposition 20.** *Let $L \in \mathcal{L}_\Sigma$ be a LFA and $k \in \mathbb{N}$. If L is consistent with $\mathbf{d}_k$, then $R(L)$ is consistent with $\mathbf{d}_{k+1}$.*

▶ **Corollary 21.** *If L is consistent with $\mathbf{d}$, then $R(L)$ is consistent with $\mathbf{d}$.*

[3] The symbol R denotes both the set of rules of some specification and the R-extension mapping of LFAs induced by a set of rules R. The meaning of symbol R will always be clear from the application context.

The R-extension mapping allows us to specify a canonical LFA given as the least fixed-point of R. Existence and uniqueness follow by the Knaster-Tarski theorem using that $(\mathcal{L}_\Sigma, \sqsubseteq)$ is a complete lattice (Prop. 15) and that R is monotone (Prop. 22). Since the bottom LFA $0 \in \mathcal{L}_\Sigma$ is consistent with $\mathbf{d}_0$ and R preserves consistency of LFAs (Prop. 20), we get that the canonical LFA is consistent with $\mathbf{d}$. The canonical LFA provides the least restricting syntactic requirements for the specified operators.

▶ **Proposition 22.** *R is order-preserving on $(\mathcal{L}_\Sigma, \sqsubseteq)$.*

▶ **Definition 23** (Canonical LFA)**.** Let $P = (\Sigma, A, R)$ be a PTSS. We call $L_P = \lim_{n\to\infty} R^n(0)$ the *canonical LFA of P*.

Dual to the notion of semantic consistency of LFAs (Def. 16) we introduce now the notion of syntactic consistency of LFAs. Intuitively, a syntactically consistent LFA ensures that the Lipschitz factors are compatible with the rules.

▶ **Definition 24** (Syntactic consistency)**.** Let $P = (\Sigma, A, R)$ be a PTSS and $L \in \mathcal{L}_\Sigma$ some LFA. We call L *consistent with P* (or alternatively L is P-consistent) if $R(L) \sqsubseteq L$.

In other words, all prefixed points of R are consistent with P. In particular, the canonical LFA L_P is consistent with P. Moreover, L_P is the least LFA consistent with P. The syntactic consistency condition $R(L) \sqsubseteq L$ of LFA L with a specification $P = (\Sigma, A, R)$ is a syntactical invariance condition on P that mimics the semantical bisimulation invariance condition $\mathbf{B}(\mathbf{d}) \sqsubseteq \mathbf{d}$ on the induced model $(\mathsf{T}(\Sigma), A, \to)$.

Semantic consistency of a LFA L (Def. 16) means consistency of L with the bisimilarity metric $\mathbf{d}$ on the induced model $(\mathsf{T}(\Sigma), A, \to)$, whereas syntactic consistency of L (Def. 24) means consistency of L with the specification $P = (\Sigma, A, R)$ from which the model is derived. As expected, syntactic consistency implies semantic consistency.

▶ **Proposition 25** (Syntactic consistency implies semantic consistency)**.** *Let $P = (\Sigma, A, R)$ be a PTSS and $L \in \mathcal{L}_\Sigma$ a LFA. If L is consistent with P then L is also consistent with $\mathbf{d}$.*

5.2 Uniformly continuous operators

A P-consistent LFA allows for deriving for each operator f an upper bound on the distance between f-composed terms.

▶ **Definition 26** (Upper bound induced by a LFA)**.** Let $P = (\Sigma, A, R)$ be a PTSS and $L \in \mathcal{L}_\Sigma$ a LFA. We define for any n-ary operator $f \in \Sigma$ the *upper bound on the distance of f-composed processes induced by L* as the mapping $\omega_{L,f} \colon (\mathbb{R}_{\geq 0})^n \to \mathbb{R}_{\geq 0}^{\infty}$ defined by

$$\omega_{L,f}(\epsilon_1, \dots, \epsilon_n) = \inf_{k\in\mathbb{N}} \left(L_k(f) \sum_{i=1}^{n} \epsilon_i + \lambda^k \right)$$

If L is consistent with P, then $\omega_{L,f}$ is an upper bound on the distance between f-composed terms wrt. $\mathbf{d}$.

▶ **Theorem 27.** *Let $P = (\Sigma, A, R)$ be a PTSS and $L \in \mathcal{L}_\Sigma$ a LFA consistent with P. Then*

$$\mathbf{d}(f(s_1, \dots, s_n), f(t_1, \dots, t_n)) \leq \omega_{L,f}(\mathbf{d}(s_1, t_1), \dots, \mathbf{d}(s_n, t_n)).$$

Moreover, if L is consistent with P, then $\omega_{L,f}$ is a modulus of continuity of f wrt. $\mathbf{d}$ if all Lipschitz factors $L_k(f)$ of f are finite.

▶ **Theorem 28.** *Let $P = (\Sigma, A, R)$ be a PTSS and $L \in \mathcal{L}_\Sigma$ a LFA consistent with P. An operator $f \in \Sigma$ is*

1. *uniformly continuous if $L_k(f) < \infty$ for all $k \in \mathbb{N}$,*
2. *Lipschitz continuous if $\sup_{k\in\mathbb{N}} L_k(f) < \infty$, and*
3. *K-Lipschitz continuous if $L_k(f) \le K$ for all $k \in \mathbb{N}$.*

Hence, if f is Lipschitz continuous, then $\sup_{k\in\mathbb{N}} L_k(f)$ is a Lipschitz factor of f. Since the canonical LFA L_P is the least LFA consistent with P it suffices to verify the conditions of Thm. 28 on the canonical LFA.

We provide now an example that shows how to derive the canonical LFA, how to compute the modulus of continuity, and how to determine the resp. compositionality property.

▶ **Example 29.** Let $P = (\Sigma, A, R)$ be the PTSS specifying the synchronous parallel composition operator (Ex. 7) and the copy operator (Ex. 12). Let $L \in \mathcal{L}_\Sigma$ be defined as $L_0(|) = 0$ and $L_0(\mathsf{cp}) = 0$, and $L_k(|) = 1$ and $L_k(\mathsf{cp}) = 2^k$ for any $k \in \mathbb{N}_{>0}$. First we show that L is the canonical LFA $L_P = \lim_{n\to\infty} R^n(0)$ (Def. 23). Observe that $R^{n+1}(0)_k = R^n(0)_k$ for all $k \le n$. By induction over n. Base case $R^0(0)_0 = 0 = L_0(|)$ and $R^0(\mathsf{cp})_0 = 0 = L_0(\mathsf{cp})$ is obvious. The induction step is $R^{n+1}(0)_{n+1}(|) = \max(\lambda \cdot R^n(0)_n(\mu \mid \nu, x) + R^n(0)_n(\mu \mid \nu, \mu), \lambda \cdot R^n(0)_n(\mu \mid \nu, y) + R^n(0)_n(\mu \mid \nu, \nu)) =$ (inductive hypthesis) $\max(\lambda \cdot L_n(\mu \mid \nu, x) + L_n(\mu \mid \nu, \mu), \lambda \cdot L_n(\mu \mid \nu, y) + L_n(\mu \mid \nu, \nu)) = \max(0+1, 0+1) = 1 = L_{n+1}(|)$ and $R^{n+1}(0)_{n+1}(\mathsf{cp}) = \max(R^n(0)_n(\mu,\mu), R^n(0)_n(\mathsf{cp}(\mu) \mid \mathsf{cp}(\nu), \mu) + R^n(0)_n(\mathsf{cp}(\mu) \mid \mathsf{cp}(\nu), \nu)) =$ (inductive hypthesis) $\max(L_n(\mu,\mu), L_n(\mathsf{cp}(\mu) \mid \mathsf{cp}(\nu), \mu) + L_n(\mathsf{cp}(\mu) \mid \mathsf{cp}(\nu), \nu)) = \max(1, 2^n + 2^n) = 2^{n+1} = L_{n+1}(\mathsf{cp})$. Hence by Thm. 27 we get that $\omega_{(L,|)}(\epsilon_1, \epsilon_2) = \epsilon_1 + \epsilon_2$, $\omega_{(L,\mathsf{cp})}(\epsilon) = \inf_{k\in\mathbb{N}}(2^k\epsilon + \lambda^k)$ are upper bounds for $|$ and cp wrt. $\mathbf{d}$. By Thm. 28 get that the operator $|$ is 1-Lipschitz continuous and that the operator cp is uniformly continuous. Moreover, the upper bounds are indeed moduli of continuity.

6 From modulus of continuity to operator specifications

In reverse, we derive now from any modulus of continuity ω a LFA L s.t. any PTSS P consistent with L specifies an operator that has ω as modulus of continuity.. The derived LFA depends on ω and the underlying model of process replication. The model of process replication is given as a mapping $\chi \colon \mathbb{R}_{\ge 0} \times \mathbb{N} \to \mathbb{R}_{\ge 0}$ assigning to each step k an upper bound on the number of spawned process instances. The first argument is a fixed growth factor.

▶ **Definition 30** (Growth function)**.** We define the following *growth functions* $\chi \colon \mathbb{R}_{\ge 0} \times \mathbb{N} \to \mathbb{R}_{\ge 0}$:

1. $\chi(c, k) = c$ (constant),
2. $\chi(c, k) = c \cdot k$ (linear growth),
3. $\chi(c, k) = c^k$ (exponential growth).

The constant growth function expresses that at most c process instances are spawned irrespective of the number of steps performed by the combined process (cf. non-recurring process replication, Ex. 10). The linear growth function will be used to model operators with bounded stepwise replication (cf. recurring step-bounded process replication, Ex. 11). Similarly, the exponential growth function allows us to model continuously replicating operators (cf. recurring step-unbounded process replication, Ex. 12).

▶ **Definition 31** (LFA induced by ω and χ)**.** Assume a function $\omega\colon [0,1]^n \to [0,1]$ s.t. $\omega(0,\dots,0) = 0$ and $\lim_{(\epsilon_1,\dots,\epsilon_n)\to(0,\dots,0)} \omega(\epsilon_1,\dots,\epsilon_n) = \omega(0,\dots,0)$, a growth function χ and an operator $f \in \Sigma$. The *LFA* $L^f_{\omega,\chi}$ *induced by* ω *and* χ *for* f is defined by

$$L^f_{\omega,\chi,k}(g) = \begin{cases} \chi(C,k) & \text{if } g = f \\ \infty & \text{if } g \neq f \end{cases}$$

with $C = \sup\{c \in \mathbb{R}_{\geq 0} \mid \forall L \in \mathcal{L}_\Sigma.\, ((\forall k \in \mathbb{N}. L_k(f) = \chi(c,k)) \Rightarrow \omega_{L,f} \leq \omega)\}$.

The LFA induced by the exponential growth function is the LFA arising from maximal recurring process replications. The recurring process replication factor C is the maximal process replication per single transition step (possibly repeated along the evolution of the combined process).

▶ **Theorem 32.** *Let $P = (\Sigma, A, R)$ be a PTSS and $L^f_{\omega,\chi}$ the LFA induced by ω and χ for f. If there exists a P-consistent LFA $L \in \mathcal{L}_\Sigma$ with $L \sqsubseteq L^f_{\omega,\chi}$, then P specifies f s.t. f admits ω as modulus of continuity.*

▶ **Example 33.** To define an operator that may not increase the behavioral distance of its argument, assume the modulus of continuity $\omega(\epsilon) = \epsilon$ (1-Lipschitz continuity). The LFA $L^f_{\omega,\chi}$ induced by ω and $\chi(1,k) = 1$ for f (Def. 30.1, Def. 31, Def. 26) gives $L^f_{\omega,\chi,k}(f) = 1$. Let the operator f be specified by the rule (with $\theta \in \mathbb{DT}(\Sigma)$ be any distribution term):

$$\frac{x \xrightarrow{a} \mu}{f(x) \xrightarrow{a} \theta}\,.$$

Clearly, $\theta = \mu$ specifies operator f s.t. $L^f_{\omega,\chi}$ is consistent with P (Def. 24) and that operator f admits ω as modulus of continuity (Thm. 32). Let $t \in \mathsf{T}(\Sigma)$ be any closed term describing some alternative process behavior. With the same argument, also $\theta = \delta(\mu \mid \mu) \oplus_p \delta(t)$ with $p \leq 1/2$ (2 instances proceed with probability at most $1/2$), $\theta = \delta(x \mid x) \oplus_p \delta(t)$ with $p \leq 1/(2\lambda)$ (2 instances proceed with one step delay with probability at most $1/(2\lambda)$), and $\theta = \delta(a^n.(x \mid x)) \oplus_p \delta(t)$ with $p \leq 1/(2\lambda^{n+1})$ ($a^n.\_$ is action prefix operator performing n-times action a followed by the argument process) specify each operator f admitting ω as modulus of continuity (Thm. 32).

We conclude by observing that $\theta = f(\mu) \mid \mu$ specifies operator f s.t. P is consistent with $L^f_{\omega,\chi}$ whereby $L^f_{\omega,\chi,k}(f) = 2k$ is obtained from the linear growth function $\chi(2,k) = 2k$ and the modulus of continuity $\omega(\epsilon) = \inf_{k\in\mathbb{N}}(2k\epsilon + \lambda^k)$. In the same way we can derive that $\theta = f(\mu) \mid f(\mu)$ specifies operator f s.t. P is consistent with $L^f_{\omega,\chi,k} = 2^k$ obtained from exponential growth function $\chi(2,k) = 2^k$ and the modulus of continuity $\omega(\epsilon) = \inf_{k\in\mathbb{N}}(2^k\epsilon + \lambda^k)$.

7 Syntactic and semantic compositionality

LFAs induced by moduli of continuity and growth functions (Def. 31) are compositional. This allows us to determine the LFA for multiple operators separately, and then to specify those operators simultaneously in a specification consistent with the composed LFAs.

▶ **Theorem 34.** *Let $P = (\Sigma, A, R)$ be a PTSS and $G \subseteq \Sigma$ be a set of operators. For each $g \in G$ let $L^g_{\omega_g,\chi_g}$ be the LFA induced by some ω_g and χ_g for g. If for each $g \in G$ the LFA $L^g_{\omega_g,\chi_g}$ is consistent with P, then also the LFA $\inf_{g\in G} L^g_{\omega_g,\chi_g}$ is consistent with P.*

Upper bounds of operators (Def. 5) are compositional. Hence, we define now an upper bound on the distance between two closed instances of a term by composing the moduli of continuity of the operators of that term. In essence, the following theorem lifts Thm. 27 to terms.

▶ **Theorem 35.** *Let $P = (\Sigma, A, R)$ be a PTSS, $L \in \mathcal{L}_\Sigma$ a LFA consistent with P and $t \in \mathbb{T}(\Sigma)$ any open term. For all closed substitutions $\sigma_1, \sigma_2 \colon \mathcal{V} \to \mathsf{T}(\Sigma)$ we get*

$$\mathbf{d}(\sigma_1(t), \sigma_2(t)) \leq \inf_{k \in \mathbb{N}} \left(\sum_{x \in \mathcal{V}_s} L_k(t, x) \cdot \mathbf{d}(\sigma_1(x), \sigma_2(x)) + \lambda^k \right).$$

▶ **Example 36.** We start by exemplifying Thm. 34. We consider the specification $P = (\Sigma, A, R)$ of operators $G = \{\_ \mid \_, \mathsf{cp}(\_)\}$. As shown in Ex. 29 the LFAs $L^{|}_{\omega_|, \chi_|, k}(|) = 1$, $L^{|}_{\omega_|, \chi_|, k}(\mathsf{cp}) = \infty$ and $L^{\mathsf{cp}}_{\omega_{\mathsf{cp}}, \chi_{\mathsf{cp}}, k}(\mathsf{cp}) = 2^k$, $L^{\mathsf{cp}}_{\omega_{\mathsf{cp}}, \chi_{\mathsf{cp}}, k}(|) = \infty$ (Def. 31) are consistent with P. Then by Thm. 34 $L = \inf_{g \in G} L^g_{\omega_g, \chi_g}$ with $L_k(|) = 1$ and $L_k(\mathsf{cp}) = 2^k$ is consistent with P.

We proceed by exemplifying Thm. 35. Consider terms $t = \mathsf{cp}(x \mid x)$. By using $L_k(|) = 1$ and $L_k(\mathsf{cp}) = 2^k$ (Ex. 29), we get $L_k(\mathsf{cp}(x \mid x), x) = L_k(\mathsf{cp}) \cdot L_k(x \mid x, x) = 2^k \cdot (L_k(|) \cdot (L_k(x, x) + L_k(x, x)) = 2^{k+1}$ and $L_k(\mathsf{cp}(x) \mid \mathsf{cp}(x), x) = 2^{k+1}$. Hence, by Thm. 35 we get $\mathbf{d}(\sigma_1(t), \sigma_2(t)) \leq \inf_{k \in \mathbb{N}} (2^{k+1} \cdot \mathbf{d}(\sigma_1(x), \sigma_2(x)) + \lambda^k)$ for all closed substitutions $\sigma_1, \sigma_2 \colon \mathcal{V} \to \mathsf{T}(\Sigma)$. Equally, for $t = \mathsf{cp}(x) \mid \mathsf{cp}(x)$ we get $L_k(\mathsf{cp}(x) \mid \mathsf{cp}(x), x) = 2^{k+1}$ and $\mathbf{d}(\sigma_1(t), \sigma_2(t)) \leq \inf_{k \in \mathbb{N}} (2^{k+1} \cdot \mathbf{d}(\sigma_1(x), \sigma_2(x)) + \lambda^k)$. The nesting of the copy operator $\mathsf{cp}(\mathsf{cp}(x))$ induces $L_k(\mathsf{cp}(\mathsf{cp}(x)), x) = 2^{2k}$ with distance bound $\mathbf{d}(\sigma_1(\mathsf{cp}(\mathsf{cp}(x))), \sigma_2(\mathsf{cp}(\mathsf{cp}(x)))) \leq \inf_{k \in \mathbb{N}} (2^{2k} \cdot \mathbf{d}(\sigma_1(x), \sigma_2(x)) + \lambda^k)$.

8 Conclusion

We developed a SOS specification format that allows us to specify simultaneously uniformly continuous operators of arbitrary (and possibly different) moduli of continuity. Our format and results pave the way for a robust and modular approach to specify and verify probabilistic systems using probabilistic process algebras and probabilistic programming languages [9, 14].

We will continue this line of research by developing SOS specification formats for uniformly continuous operators wrt. weak metric semantics [10] and metric variants of branching bisimulation equivalence [2]. Our case studies (partially published in [12]) indicated that concepts such as encapsulation and abstraction are fundamental to perform the metric compositional analysis of systems described by probabilistic process algebras in a scalable manner. A second research direction we plan to investigate is the distance between operators (instead of terms) to describe the behavioral distance whenever one operator needs to be replaced or approximated by another. Intuitively, if an operator becomes unavailable, the distance between operators will suggest an optimal replacement operator to build an alternative system which is closest to the original system.

References

1 Luca Aceto, Wan Fokkink, and Chris Verhoef. Structural operational semantics. In *Handbook of Process Algebra*, pages 197–292. Elsevier, 2001.

2 Suzana Andova and Tim AC Willemse. Branching bisimulation for probabilistic systems: characteristics and decidability. *TCS*, 356(3):325–355, 2006.

3 Falk Bartels. GSOS for probabilistic transition systems. In *Proc. CMCS'02*, volume 65 of *ENTCS*, pages 29–53. Elsevier, 2002.

4 Falk Bartels. *On Generalised Coinduction and Probabilistic Specification Formats*. PhD thesis, VU University Amsterdam, 2004.

5 Bard Bloom, Sorin Istrail, and Albert R. Meyer. Bisimulation can't be traced. *J. ACM*, 42:232–268, 1995.

6 Pedro R. D'Argenio, Daniel Gebler, and Matias David Lee. Axiomatizing Bisimulation Equivalences and Metrics from Probabilistic SOS Rules. In *Proc. FoSSaCS'14*, volume 8412 of *LNCS*, pages 289–303. Springer, 2014.

7 Pedro R. D'Argenio and Matias David Lee. Probabilistic Transition System Specification: Congruence and Full Abstraction of Bisimulation. In *Proc. FoSSaCS'12*, volume 7213 of *LNCS*, pages 452–466. Springer, 2012.

8 Yuxin Deng, Tom Chothia, Catuscia Palamidessi, and Jun Pang. Metrics for Action-labelled Quantitative Transition Systems. In *Proc. QAPL'05*, volume 153 of *EPTCS*, pages 79–96, 2005.

9 Josée Desharnais, Vineet Gupta, Radha Jagadeesan, and Prakash Panangaden. Metrics for Labelled Markov Processes. *TCS*, 318(3):323–354, 2004.

10 Josée Desharnais, Radha Jagadeesan, Vineet Gupta, and Prakash Panangaden. The Metric Analogue of Weak Bisimulation for Probabilistic Processes. In *Proc. LICS'02*, pages 413–422. IEEE, 2002.

11 Wan Fokkink, Rob J. van Glabbeek, and Paulien de Wind. Divide and congruence: From decomposition of modal formulas to preservation of branching and η-bisimilarity. *I&C*, 214:59–85, 2012.

12 Daniel Gebler, Kim G. Larsen, and Simone Tini. Compositional metric reasoning with Probabilistic Process Calculi. In *Proc. FoSSaCS'15*, volume 9034 of *LNCS*, pages 230–245. Springer, 2015.

13 Daniel Gebler and Simone Tini. Fixed-point Characterization of Compositionality Properties of Probabilistic Processes Combinators. In *Proc. EXPRESS/SOS'14*, volume 160 of *EPTCS*, pages 63–78. OPA, 2014.

14 T.A. Henzinger. Quantitative reactive modeling and verification. *Computer Science – R&D*, 28(4):331–344, 2013.

15 Bengt Jonsson and Wang Larsen, Kim G.and Yi. Probabilistic Extensions of Process Algebras. In *Handbook of Process Algebra*, pages 685–710. Elsevier, 2001.

16 Ruggero Lanotte and Simone Tini. Probabilistic bisimulation as a congruence. *ACM TOCL*, 10:1–48, 2009.

17 Kim G. Larsen and Arne Skou. Bisimulation through probabilistic testing. *I&C*, 94:1–28, 1991.

18 Matias David Lee, Daniel Gebler, and Pedro R. D'Argenio. Tree Rules in Probabilistic Transition System Specifications with Negative and Quantitative Premises. In *Proc. EXPRESS/SOS'12*, volume 89 of *EPTCS*, pages 115–130. OPA, 2012.

19 Mohammad Reza Mousavi, Michel A. Reniers, and Jan Friso Groote. SOS formats and meta-theory: 20 years after. *TCS*, 373(3):238–272, 2007.

20 Roberto Segala. *Modeling and Verification of Randomized Distributed Real-Time Systems*. PhD thesis, MIT, 1995.

21 Simone Tini. Non-expansive ϵ-bisimulations for probabilistic processes. *TCS*, 411:2202–2222, 2010.

22 Franck van Breugel and James Worrell. Towards quantitative verification of probabilistic transition systems. In *Proc. ICALP'01*, volume 2076 of *LNCS*, pages 421–432. Springer, 2001.

23 Wang Yi and Kim Guldstrand Larsen. Testing probabilistic and nondeterministic processes. In *PSTV*, volume 12, pages 47–61, 1992.