Compositional metric reasoning with Probabilistic Process Calculi

Daniel Gebler[1], Kim G. Larsen[2⋆], and Simone Tini[3]

[1] VU University Amsterdam (NL)
[2] Aalborg University (DK)
[3] University of Insubria (IT)

Abstract. We study which standard operators of probabilistic process calculi allow for compositional reasoning with respect to bisimulation metric semantics. We argue that uniform continuity (generalizing the earlier proposed property of non-expansiveness) captures the essential nature of compositional reasoning and allows now also to reason compositionally about recursive processes. We characterize the distance between probabilistic processes composed by standard process algebra operators. Combining these results, we demonstrate how compositional reasoning about systems specified by continuous process algebra operators allows for metric assume-guarantee like performance validation.

1 Introduction

Probabilistic process algebras describe probabilistic concurrent communicating systems (probabilistic processes for short). In this paper we study compositional reasoning over probabilistic processes, specified by terms of probabilistic process algebras.

Behavioral equivalences equate processes that are indistinguishable to any external observer. The most prominent example is bisimulation equivalence [15], which provides a well-established theory of the behavior of probabilistic nondeterministic transition systems. However, bisimulation equivalence is too sensitive to the exact probabilities of transitions. The slightest perturbation of the probabilities can destroy bisimilarity. Bisimulation metric [3, 7, 8] provides a robust semantics for probabilistic processes. It is the quantitative analogue to bisimulation equivalence and assigns to each pair of processes a distance which measures the proximity of their quantitative properties. The distances form a pseudometric[4] where bisimilar processes are in distance 0.

In order to specify and verify systems in a compositional manner, it is necessary that the behavioral semantics is compatible with all operators of the language that describe these systems. For behavioral equivalence semantics there is common agreement that compositional reasoning requires that the considered behavioral equivalence is a congruence wrt. all operators. On the other hand, for behavioral metric semantics there are several proposals of properties that operators should satisfy in order to facilitate compositional reasoning. Most prominent examples are non-expansiveness [8] and non-extensiveness [1]. We discuss these properties and propose uniform continuity

⋆ This research is partially supported by the European FET projects SENSATION and CASSTING and the Sino-Danish Center IDEA4CPS.

[4] A bisimulation metric is in fact a pseudometric. For convenience we use the term bisimulation metric instead of bisimulation pseudometric.

as the most natural property of process operators to facilitate compositional reasoning wrt. behavioral metric semantics especially in presence of recursion. Uniform continuity generalizes non-extensiveness and non-expansiveness and captures the essential nature of compositional reasoning wrt. behavioral metric semantics. A uniformly continuous binary process operator f ensures that for any non-zero bisimulation distance ϵ (understood as the admissible tolerance from the operational behavior of the composed process $f(p_1, p_2)$) there are non-zero bisimulation distances δ_1 and δ_2 (understood as the admissible tolerances from the operational behavior of the processes p_1 and p_2) such that the distance between the composed processes $f(p_1, p_2)$ and $f(p'_1, p'_2)$ is at most ϵ whenever the component p'_1 (resp. p'_2) is in distance of at most δ_1 from p_1 (resp. at most δ_2 from p_2). Our key contributions are as follows:

1. We develop for many non-recursive and recursive process operators used in various probabilistic process algebras tight upper bounds on the distance between processes combined by those operators (Sec. 3.2 and 4.2).
2. We show that non-recursive process operators, esp. (nondeterministic and probabilistic variants of) sequential, alternative and parallel composition, allow for compositional reasoning wrt. the compositionality criteria of non-expansiveness and hence also wrt. uniform continuity (Sec. 3).
3. We show that recursive process operators, e.g. (nondeterministic and probabilistic variants of) Kleene-star iteration and π-calculus bang replication, allow for compositional reasoning wrt. the compositionality criterion of uniform continuity, but not wrt. non-expansiveness and non-extensiveness (Sec. 4).
4. We demonstrate the usefulness of compositional reasoning using a network protocol build from uniformly continuous operators. In particular, we show how it is possible to derive performance guarantees of the entire system from performance assumptions about individual components. Conversely, we show how it is also possible to derive performance requirements on individual components from performance requirements of the complete system (Sec. 5).

2 Preliminaries

We consider transition systems with process terms as states and a transition relation inductively defined by means of SOS rules. Process terms are inductively defined by the process combinators. The SOS rules are syntax-driven inference rules that define the behavior of complex processes in terms of the behavior of their components.

Probabilistic Transition Systems A *signature* is a structure $\Sigma = (F, \mathsf{r})$, where F is a countable set of *operators*, or *process combinators*, and $\mathsf{r}\colon F \to \mathbb{N}$ is a *rank function*, which gives the arity of an operator. By $f \in \Sigma$ we mean $f \in F$. We assume an infinite set of *process variables* (or *state variables*) $\mathcal{V}_s$ disjoint from F. The set of *process terms* (or *state terms*) over a signature Σ and a set $V \subseteq \mathcal{V}_s$ of variables, notation $\mathsf{T}(\Sigma, V)$, is the least set satisfying: (i) $V \subseteq \mathsf{T}(\Sigma, V)$, and (ii) $f(t_1, \ldots, t_n) \in \mathsf{T}(\Sigma, V)$ whenever $f \in \Sigma$, $t_1, \ldots, t_n \in \mathsf{T}(\Sigma, V)$ and $n = \mathsf{r}(f)$. We will use n for $\mathsf{r}(f)$ if it is clear from the context. We write $\mathsf{T}(\Sigma)$ for $\mathsf{T}(\Sigma, \emptyset)$ (set of all *closed process terms*) and $\mathbb{T}(\Sigma)$ for $\mathsf{T}(\Sigma, \mathcal{V}_s)$ (set of all *open process terms*). We may refer to closed process terms as *processes*.

Probabilistic transition systems extend transition systems by allowing for probabilistic choices in the transitions. We consider probabilistic nondeterministic labelled

transition systems [15]. The state space is defined as the set $\mathsf{T}(\Sigma)$ of all processes. Probability distributions over this state 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 process t 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). *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* $(t, a, \pi) \in \to$.

Bisimulation metric on PTS We define now bisimulation metric as the quantitative analogue to bisimulation equivalence. A 1-*bounded pseudometric* on the set of processes $\mathsf{T}(\Sigma)$ is a function $d\colon \mathsf{T}(\Sigma) \times \mathsf{T}(\Sigma) \to [0,1]$ with $d(t,t) = 0$, $d(t,t') = d(t',t)$, and $d(t,t') \leq d(t,t'') + d(t'',t')$, for all $t, t', t'' \in \mathsf{T}(\Sigma)$. We will use 1-bounded pseudometrics to describe the behavioral distances between processes. We order 1-bounded pseudometrics by $d_1 \sqsubseteq d_2$ iff $d_1(t,t') \leq d_2(t,t')$ for all $t, t' \in \mathsf{T}(\Sigma)$.

A 1-bounded pseudometric on processes $\mathsf{T}(\Sigma)$ is lifted to a 1-bounded pseudometric on distributions $\Delta(\mathsf{T}(\Sigma))$ by means of the Kantorovich pseudometric. A *matching* for $(\pi, \pi') \in \Delta(\mathsf{T}(\Sigma)) \times \Delta(\mathsf{T}(\Sigma))$ is a distribution $\omega \in \Delta(\mathsf{T}(\Sigma) \times \mathsf{T}(\Sigma))$ with $\sum_{t'\in\mathsf{T}(\Sigma)} \omega(t,t') = \pi(t)$ and $\sum_{t\in\mathsf{T}(\Sigma)} \omega(t,t') = \pi'(t')$ for all $t, t' \in \mathsf{T}(\Sigma)$. Let $\Omega(\pi,\pi')$ be the set of all matchings for (π,π'). The *Kantorovich pseudometric* $\mathbf{K}(d)\colon \Delta(\mathsf{T}(\Sigma)) \times \Delta(\mathsf{T}(\Sigma)) \to [0,1]$ for a pseudometric $d\colon \mathsf{T}(\Sigma) \times \mathsf{T}(\Sigma) \to [0,1]$ is given by $\mathbf{K}(d)(\pi,\pi') = \min_{\omega\in\Omega(\pi,\pi')} \sum_{t,t'\in\mathsf{T}(\Sigma)} d(t,t') \cdot \omega(t,t')$ for all $\pi, \pi' \in \Delta(\mathsf{T}(\Sigma))$.

A 1-bounded pseudometric is a bisimulation metric if for all pairs of process terms t and t' each transition of t can be mimicked by a transition of t' with the same label and the distance between the accessible distributions does not exceed the distance between t and t'. By means of a *discount factor* $\lambda \in (0,1]$ we allow to specify how much the behavioral distance of future transitions is taken into account [6, 8]. The discount factor $\lambda = 1$ expresses no discount, meaning that the differences in the behavior between t and t' are considered irrespective of after how many steps they can be observed.

Definition 2 (Bisimulation metric [8]). *A* 1*-bounded pseudometric* d *on* $\mathsf{T}(\Sigma)$ *is a* λ-bisimulation metric *for* $\lambda \in (0,1]$ *if for all process terms* $t, t' \in \mathsf{T}(\Sigma)$ *with* $d(t,t') < 1$*, if* $t \xrightarrow{a} \pi$ *then there exists a transition* $t' \xrightarrow{a} \pi'$ *such that* $\lambda \cdot \mathbf{K}(d)(\pi,\pi') \leq d(t,t')$.

The smallest λ-bisimulation metric, notation $\mathbf{d}_\lambda$, is called *λ-bisimilarity metric* [3, 7, 8]. By λ-bisimulation distance between t and t' we mean $\mathbf{d}_\lambda(t,t')$. Bisimilarity equivalence [15] is the kernel of $\mathbf{d}_\lambda$ [8], i.e. $\mathbf{d}_\lambda(t,t') = 0$ iff t and t' are bisimilar. We may write $\mathbf{d}$ for $\mathbf{d}_1$.

Remark 3. Clearly, $\mathbf{d}_\lambda(t,t') \in [0,\lambda] \cup \{1\}$ for all $t, t' \in \mathsf{T}(\Sigma)$. Let $\lambda < 1$. Then, $\mathbf{d}_\lambda(t,t') = 1$ iff t can perform an action which t' cannot (or vice versa), $\mathbf{d}_\lambda(t,t') = 0$ iff t and t' have the same reactive behavior, and $\mathbf{d}_\lambda(t,t') \in (0,\lambda]$ iff t and t' have different reactive behavior after performing the same initial action.

Algebra of probability distributions We start with some notations and operations on probability distributions. We denote by $\delta(t)$ with $t \in \mathsf{T}(\Sigma)$ the *Dirac distribution* defined by $(\delta(t))(t) = 1$ and $(\delta(t))(t') = 0$ if $t \neq t'$. The convex combination $\sum_{i\in I} p_i\pi_i$ of a family $\{\pi_i\}_{i\in I}$ of probability 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 of $\pi_1,\ldots,\pi_n$ defined by $f(\pi_1,\ldots,\pi_n)(f(t_1,\ldots,t_n)) = \prod_{i=1}^n \pi_i(t_i)$ and $f(\pi_1,\ldots,\pi_n)(t) = 0$ for all $t \in \mathsf{T}(\Sigma)$ not in the form $t = f(t_1,\ldots,t_n)$. For binary operators f we may write $\pi_1\, f\, \pi_2$ for $f(\pi_1,\pi_2)$.

In order to describe probabilistic behavior, we need syntactic expressions that denote probability distributions. To be precise, each closed expression will denote some probability distribution, and each open expression instantiates by a closed substitution to some probability distribution. 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 process and distribution variables $\mathcal{V} = \mathcal{V}_s \cup \mathcal{V}_d$. The set of *distribution terms* over process variables $V_s \subseteq \mathcal{V}_s$ and distribution variables $V_d \subseteq \mathcal{V}_d$, notation $\mathsf{DT}(\Sigma, V_s, V_d)$, is the least set satisfying [12]: (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 $\mathbb{DT}(\Sigma)$ 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* $\delta(t)$ is an expression that takes as value the Dirac distribution $\delta(t')$ when variables in t are substituted so that t becomes the closed term t'. Case iii allows to construct convex combinations of distributions. We write $\theta_1 \oplus_p \theta_2$ for $\sum_{i=1}^2 p_i\theta_i$ with $p_1 = p$ and $p_2 = 1-p$. Case iv lifts the structural inductive construction of state terms to distribution terms.

A *substitution* is a mapping $\sigma\colon \mathcal{V} \to \mathbb{T}(\Sigma) \cup \mathbb{DT}(\Sigma)$ s.t. $\sigma(x) \in \mathbb{T}(\Sigma)$ if $x \in \mathcal{V}_s$ and $\sigma(\mu) \in \mathbb{DT}(\Sigma)$ if $\mu \in \mathcal{V}_d$. σ extends to a mapping from process terms to process terms as usual and to a mapping from distribution terms to distribution terms by $\sigma(\delta(t)) = \delta(\sigma(t))$, $\sigma(\sum_{i\in I} p_i\theta_i) = \sum_{i\in I} p_i\sigma(\theta_i)$, and $\sigma(f(\theta_1,\ldots,\theta_n)) = f(\sigma(\theta_1),\ldots,\sigma(\theta_n))$. A substitution σ is *closed* if $\sigma(x) \in \mathsf{T}(\Sigma)$ for all $x \in \mathcal{V}_s$ and $\sigma(\mu) \in \mathsf{DT}(\Sigma)$ for all $\mu \in \mathcal{V}_d$.

Specification of Process Combinators We specify the operational semantics of process combinators by SOS rules in the probabilistic GSOS format [2,12]. The operational semantics of a process term is given by inductively applying the respective SOS rules.

Definition 4 (PGSOS rule [2,12]). *A* PGSOS rule *has the form:*

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

with n the rank of operator $f \in \Sigma$, $I = \{1,\ldots,n\}$ the indices of the arguments of f, finite index sets K_i, L_i, actions $a_{i,k}, b_{i,l}, a \in A$, process variables $x_i \in \mathcal{V}_s$, distribution variables $\mu_{i,k} \in \mathcal{V}_d$, distribution term $\theta \in \mathbb{DT}(\Sigma)$, and constraints:

1. *all $\mu_{i,k}$ for $i \in I, k \in K_i$ are pairwise different;*
2. *all $x_1,\ldots,x_n$ are pairwise different;*
3. *$Var(\theta) \subseteq \{\mu_{i,k} \mid i \in I, k \in K_i\} \cup \{x_1 \ldots, x_n\}$.*

The expressions $x_i \xrightarrow{a_{i,k}} \mu_{i,k}$ and $x_i \xrightarrow{b_{i,l}} \not\to$ above the line, and $f(x_1,\ldots,x_n) \xrightarrow{a} \theta$ below the line, are called, resp., *positive premises*, *negative premises* and *conclusion* of the rule.

A *probabilistic transition system specification* (PTSS) in PGSOS format is a triple $P = (\Sigma, A, R)$, where Σ is a signature, A is a countable set of actions and R is a countable

$$\frac{}{\varepsilon \xrightarrow{\surd} \delta(0)} \qquad \frac{}{a.\bigoplus_{i=1}^{n}[p_i]x_i \xrightarrow{a} \sum_{i=1}^{n} p_i\delta(x_i)} \qquad \frac{x \xrightarrow{a} \mu \quad a \neq \surd}{x;y \xrightarrow{a} \mu;\delta(y)} \qquad \frac{x \xrightarrow{\surd} \mu \quad y \xrightarrow{a} \nu}{x;y \xrightarrow{a} \nu}$$

$$\frac{x \xrightarrow{a} \mu}{x+y \xrightarrow{a} \mu} \qquad \frac{y \xrightarrow{a} \nu}{x+y \xrightarrow{a} \nu} \qquad \frac{x \xrightarrow{a} \mu \quad y \xrightarrow{a} \nu}{x \mid y \xrightarrow{a} \mu \mid \nu} \qquad \frac{x \xrightarrow{a} \mu}{x \,|||\, y \xrightarrow{a} \mu \,|||\, \delta(y)} \qquad \frac{y \xrightarrow{a} \nu}{x \,|||\, y \xrightarrow{a} \delta(x) \,|||\, \nu}$$

$$\frac{x \xrightarrow{a} \mu \quad y \xrightarrow{a} \nu \quad a \in B \setminus \{\surd\}}{x \parallel_B y \xrightarrow{a} \mu \parallel_B \nu} \qquad \frac{x \xrightarrow{a} \mu \quad a \notin B \cup \{\surd\}}{x \parallel_B y \xrightarrow{a} \mu \parallel_B \delta(y)} \qquad \frac{y \xrightarrow{a} \nu \quad a \notin B \cup \{\surd\}}{x \parallel_B y \xrightarrow{a} \delta(x) \parallel_B \nu} \qquad \frac{x \xrightarrow{\surd} \mu \quad y \xrightarrow{\surd} \nu}{x \parallel_B y \xrightarrow{\surd} \delta(0)}$$

Tab. 1: Standard non-recursive process combinators

set of PGSOS rules. A *supported model* of P is a PTS $(\mathsf{T}(\Sigma), A, \rightarrow)$ such that the transition relation $\rightarrow$ contains all and only those transitions for which P offers a justification, i.e. $t \xrightarrow{a} \pi \in \rightarrow$ iff for some rule $r \in R$ and some closed substitution σ all premises of r hold, i.e. for all positive premises $x_i \xrightarrow{a_{i,k}} \mu_{i,k}$ we have $\sigma(x_i) \xrightarrow{a_{i,k}} \sigma(\mu_{i,k}) \in \rightarrow$ and for all negative premises $x_i \xrightarrow{b_{i,l}}\!\!\!\!\!\!/\;\;$ we have $\sigma(x_i) \xrightarrow{b_{i,l}} \pi \notin \rightarrow$ for all $\pi \in \Delta(\mathsf{T}(\Sigma))$, and the conclusion $f(x_1, \ldots, x_n) \xrightarrow{a} \theta$ instantiates to $\sigma(f(x_1, \ldots, x_n)) = t$ and $\sigma(\theta) = \pi$. Each PTSS in PGSOS format has a supported model which is moreover unique [2].

Intuitively, a term $f(t_1, \ldots, t_n)$ represents the composition of processes $t_1, \ldots, t_n$ by operator f. A rule r specifies some transition $f(t_1, \ldots, t_n) \xrightarrow{a} \pi$ that represents the evolution of the composed process $f(t_1, \ldots, t_n)$ by action a to the distribution π.

Definition 5 (Disjoint extension). *Let $P = (\Sigma, A, R)$ and $P' = (\Sigma', A, R')$ be two PTSSs in PGSOS format. P' is a* disjoint extension *of P, notation $P \sqsubseteq P'$, iff $\Sigma \subseteq \Sigma'$, $R \subseteq R'$ and R' introduces no new rule for any operator in Σ.*

The disjoint extension of the specification of some process combinator allows to specify arbitrary processes while the operational semantics of the process combinator remains unchanged. This allows us to study the compositionality properties of concrete process combinators which hold for the composition of arbitrary processes.

3 Non-recursive processes

We start by discussing compositional reasoning over probabilistic processes that are composed by non-recursive process combinators. First we introduce the most common non-recursive process combinators, then study the distance between composed processes, and conclude by analyzing their compositionality properties. Our study of compositionality properties generalizes earlier results of [7, 8] which considered only a small set of process combinators and only the property of non-expansiveness. The development of tight bounds on the distance between composed process (necessary for effective metric assume-guarantee performance validation) is novel.

3.1 Non-recursive process combinators

We introduce a probabilistic process algebra that comprises many of the probabilistic CCS [2] and CSP [4] process combinators. Let Σ_{PA} be a signature with the following

$$\frac{x \xrightarrow{a} \mu \quad y \not\xrightarrow{a}}{x +_p y \xrightarrow{a} \mu} \qquad \frac{x \not\xrightarrow{a} \quad y \xrightarrow{a} \nu}{x +_p y \xrightarrow{a} \nu} \qquad \frac{x \xrightarrow{a} \mu \quad y \xrightarrow{a} \nu}{x +_p y \xrightarrow{a} \mu \oplus_p \nu}$$

$$\frac{x \xrightarrow{a} \mu \quad y \not\xrightarrow{a}}{x \,|||_p\, y \xrightarrow{a} \mu \,|||_p\, \delta(y)} \qquad \frac{x \not\xrightarrow{a} \quad y \xrightarrow{a} \nu}{x \,|||_p\, y \xrightarrow{a} \delta(x) \,|||_p\, \nu} \qquad \frac{x \xrightarrow{a} \mu \quad y \xrightarrow{a} \nu}{x \,|||_p\, y \xrightarrow{a} \mu \,|||_p\, \delta(y) \oplus_p \delta(x) \,|||_p\, \nu}$$

Tab. 2: Standard non-recursive probabilistic process combinators

operators: i) constants 0 (stop process) and ε (skip process); ii) a family of n-ary probabilistic prefix operators $a.([p_1]\_ \oplus \ldots \oplus [p_n]\_)$ with $a \in A$, $n \geq 1$, $p_1, \ldots, p_n \in (0, 1]$ and $\sum_{i=1}^{n} p_i = 1$; iii) binary operators $\_;\_$ (sequential composition), $\_ + \_$ (alternative composition), $\_ +_p \_$ (probabilistic alternative composition), $\_ \mid \_$ (synchronous parallel composition), $\_ \,|||\, \_$ (asynchronous parallel composition), $\_ \,|||_p\, \_$ (probabilistic parallel composition), and $\_ \,\|_B\, \_$ for each for each $B \subseteq A$ (CSP parallel composition). The PTSS $P_{\mathrm{PA}} = (\Sigma_{\mathrm{PA}}, A, R_{\mathrm{PA}})$ is given by the rules R_{PA} in Tab. 1 and Tab. 2. We write $a.\bigoplus_{i=1}^{n}[p_i]\_$ for $a.([p_1]\_ \oplus \ldots \oplus [p_n]\_)$ and $a.\_$ for $a.([1]\_)$. Moreover, by process a we mean $a.0$.

3.2 Distance between non-recursive processes

We develop now tight bounds on the distance between processes combined by the non-recursive process combinators. This allows us later to derive the compositionality properties of those operators. As we will discuss two different compositionality properties for non-recursive processes, we split in this section the discussion on the distance bounds accordingly. We use disjoint extensions of the specification of the process combinators in order to reason over the composition of arbitrary processes.

We will express the bound on the distance between composed processes $f(s_1, \ldots, s_n)$ and $f(t_1, \ldots, t_n)$ in terms of the distance between their respective components s_i and t_i. Intuitively, given a probabilistic process $f(s_1, \ldots, s_n)$ we provide a bound on the distance to the respective probabilistic process $f(t_1, \ldots, t_n)$ where each component s_i is replaced by the component t_i. We start with those process combinators that satisfy the later discussed compositionality property of non-extensiveness (Def. 9).

Proposition 6. *Let $P = (\Sigma, A, R)$ be any PTSS with $P_{PA} \sqsubseteq P$. For all $s_i, t_i \in T(\Sigma)$*

(a) $\mathbf{d}_\lambda(a.\bigoplus_{i=1}^{n}[p_i]s_i, a.\bigoplus_{i=1}^{n}[p_i]t_i) \leq \lambda \sum_{i=1}^{n} p_i \mathbf{d}_\lambda(s_i, t_i)$*;*
(b) $\mathbf{d}_\lambda(s_1 + s_2, t_1 + t_2) \leq \max(\mathbf{d}_\lambda(s_1, t_1), \mathbf{d}_\lambda(s_2, t_2))$*;*
(c) $\mathbf{d}_\lambda(s_1 +_p s_2, t_1 +_p t_2) \leq \max(\mathbf{d}_\lambda(s_1, t_1), \mathbf{d}_\lambda(s_2, t_2))$.

The distance between action prefixed processes (Prop. 6.a) is discounted by λ since the processes $a.\bigoplus_{i=1}^{n}[p_i]s_i$ and $a.\bigoplus_{i=1}^{n}[p_i]t_i$ perform first the action a before s_i and t_i may evolve. The distances between processes composed by either the nondeterministic alternative composition operator or by the probabilistic alternative composition are both bounded by the maximum of the distances between their respective arguments (Prop. 6.b and Prop. 6.c). The distance bounds for these operators coincide since the first two rules specifying the probabilistic alternative composition define the same operational behavior as the nondeterministic alternative composition and the third rule defines a convex combination of these transitions.

We proceed with those process combinators that satisfy the later discussed compositionality property of non-expansiveness (Def. 12).

Proposition 7. *Let $P = (\Sigma, A, R)$ be any PTSS with $P_{PA} \sqsubseteq P$. For all $s_i, t_i \in \mathsf{T}(\Sigma)$*

(a) $\mathbf{d}_\lambda(s_1; s_2, t_1; t_2) \leq \begin{cases} 1 & \text{if } \mathbf{d}_\lambda(s_1, t_1) = 1 \\ \max(d^a_{1,2}, \mathbf{d}_\lambda(s_2, t_2)) & \text{if } \mathbf{d}_\lambda(s_1, t_1) \in [0, 1) \end{cases}$

(b) $\mathbf{d}_\lambda(s_1 \mid s_2, t_1 \mid t_2) \leq d^s$

(c) $\mathbf{d}_\lambda(s_1 \,|||\, s_2, t_1 \,|||\, t_2) \leq d^a$

(d) $\mathbf{d}_\lambda(s_1 \parallel_B s_2, t_1 \parallel_B t_2) \leq \begin{cases} d^s & \text{if } B \setminus \{\surd\} \neq \emptyset \\ d^a & \text{otherwise} \end{cases}$

(e) $\mathbf{d}_\lambda(s_1 \,|||_p\, s_2, t_1 \,|||_p\, t_2) \leq d^a$, *with*

$$d^s = \begin{cases} 1 & \text{if } \mathbf{d}_\lambda(s_1, t_1) = 1 \text{ or } \mathbf{d}_\lambda(s_2, t_2) = 1 \\ \mathbf{d}_\lambda(s_1, t_1) + (1 - \mathbf{d}_\lambda(s_1, t_1)/\lambda)\mathbf{d}_\lambda(s_2, t_2) & \text{otherwise} \end{cases}$$

$$d^a = \begin{cases} 1 & \text{if } \mathbf{d}_\lambda(s_1, t_1) = 1 \\ 1 & \text{if } \mathbf{d}_\lambda(s_2, t_2) = 1 \\ \max(d^a_{1,2}\,,\, d^a_{2,1}) & \text{otherwise} \end{cases} \qquad \begin{aligned} d^a_{1,2} &= \mathbf{d}_\lambda(s_1, t_1) + \lambda(1 - \mathbf{d}_\lambda(s_1, t_1)/\lambda)\mathbf{d}_\lambda(s_2, t_2) \\ d^a_{2,1} &= \mathbf{d}_\lambda(s_2, t_2) + \lambda(1 - \mathbf{d}_\lambda(s_2, t_2)/\lambda)\mathbf{d}_\lambda(s_1, t_1) \end{aligned}$$

The expression d^s captures the distance bound between the synchronously evolving processes s_1 and s_2 on the one hand and the synchronously evolving processes t_1 and t_2 on the other hand. We remark that distances $\mathbf{d}_\lambda(s_1, t_1)$ and $\mathbf{d}_\lambda(s_2, t_2)$ contribute symmetrically to d^s since $\mathbf{d}_\lambda(s_1, t_1) + (1 - \mathbf{d}_\lambda(s_1, t_1)/\lambda)\mathbf{d}_\lambda(s_2, t_2) = \mathbf{d}_\lambda(s_2, t_2) + (1 - \mathbf{d}_\lambda(s_2, t_2)/\lambda)\mathbf{d}_\lambda(s_1, t_1) = \mathbf{d}_\lambda(s_1, t_1) + \mathbf{d}_\lambda(s_2, t_2) - \mathbf{d}_\lambda(s_1, t_1)\mathbf{d}_\lambda(s_2, t_2)/\lambda$. The expressions $d^a_{1,2}, d^a_{2,1}, d^a$ cover different scenarios of the asynchronous evolution of those processes. The expression $d^a_{1,2}$ (resp. $d^a_{2,1}$) denotes the distance bound between the asynchronously evolving processes s_1 and s_2 on the one hand and the asynchronously evolving processes t_1 and t_2 on the other hand, at which the first transition is performed by the processes s_1 and t_1 (resp. the first transition is performed by processes s_2 and t_2). Hence, the distances of the asynchronously evolving processes $d^a_{1,2}$ and $d^a_{2,1}$ differ from the distance d^s of the synchronously evolving processes only by the discount factor λ that is applied to the delayed process. Finally, d^a captures the distance between asynchronously evolving processes independent of which of those processes moves first. If $\mathbf{d}_\lambda(s_i, t_i) = 1$ the processes may disagree on the initial actions they can perform and the composed processes have then also the maximal distance of 1 (cf. Rem. 3).

We consider now the process combinators in detail. The distance between sequentially composed processes $s_1; s_2$ and $t_1; t_2$ (Prop. 7.a) is given if $\mathbf{d}_\lambda(s_1, t_1) \in [0, 1)$ as the maximum of (i) the distance $d^a_{1,2}$ (which captures the case that first the processes s_1 and t_1 evolve followed by s_2 and t_2), and (ii) the distance $\mathbf{d}_\lambda(s_2, t_2)$ (which captures the case that the processes s_2 and t_2 evolve immediately because both s_1 and t_1 terminate successfully). The distance $d^a_{1,2}$ weights the distance between s_2 and t_2 by $\lambda(1 - \mathbf{d}_\lambda(s_1, t_1)/\lambda)$. The discount λ expresses that the distance between processes s_2 and t_2 is observable just after s_1 and t_1 have performed at least one step. Additionally, note that the difference between s_2 and t_2 can only be observed when s_1 and t_1 agree to terminate. When processes s_1 and t_1 evolve by one step, they disagree by $\mathbf{d}_\lambda(s_1, t_1)/\lambda$ on their behavior. Hence they agree by $1 - \mathbf{d}_\lambda(s_1, t_1)/\lambda$. Thus, the distance between processes s_2 and t_2 needs to be additionally weighted by $(1 - \mathbf{d}_\lambda(s_1, t_1)/\lambda)$. In case (ii) the distance between s_2 and t_2 is not discounted since both processes start immediately.

The distance between synchronous parallel composed processes $s_1 \mid s_2$ and $t_1 \mid t_2$ is $\mathbf{d}_\lambda(s_1,t_1)+(1-\mathbf{d}_\lambda(s_1,t_1)/\lambda)\mathbf{d}_\lambda(s_2,t_2) = \lambda(1-(1-\mathbf{d}_\lambda(s_1,t_1)/\lambda)(1-\mathbf{d}_\lambda(s_2,t_2)/\lambda))$. The distance between $s_1 \mid s_2$ and $t_1 \mid t_2$ is bounded by the sum of the distance between s_1 and t_1 (degree of dissimilarity between s_1 and t_1) and the distance between s_2 and t_2 weighted by the probability that s_1 and t_1 agree on their behavior (degree of dissimilarity between s_2 and t_2 under equal behavior of s_1 and t_1). Alternatively, the distance between $s_1 \mid s_2$ and $t_1 \mid t_2$ can be understood as composing processes on the behavior they agree upon, i.e. $s_1 \mid s_2$ and $t_1 \mid t_2$ agree on their behavior if s_1 and t_1 agree (probability of similarity $1 - \mathbf{d}_\lambda(s_1,t_1)/\lambda$) and if s_2 and t_2 agree (probability of similarity $1 - \mathbf{d}_\lambda(s_2,t_2)/\lambda$). The resulting distance is then the probability of dissimilarity of the respective behavior expressed by $1-(1-\mathbf{d}_\lambda(s_1,t_1)/\lambda)(1-\mathbf{d}_\lambda(s_2,t_2)/\lambda)$ multiplied by the discount factor λ.

The distance between asynchronous parallel composed processes $s_1 \mid\mid\mid s_2$ and $t_1 \mid\mid\mid t_2$ is exactly the expression d^a. The distance between processes composed by the probabilistic parallel composition operator $s_1 \mid\mid\mid_p s_2$ and $t_1 \mid\mid\mid_p t_2$ is bounded by the same expression d^a since the first two rules specifying the probabilistic parallel composition define the same operational behavior as the nondeterministic parallel composition. The third rule defining a convex combination of these transitions applies only for those actions that can be performed by both processes s_1 and s_2 and resp. t_1 and t_2.

Processes that are composed by the CSP parallel composition operator $\_ \|_B \_$ evolve synchronously for actions in $B \setminus \{\surd\}$, evolve asynchronously for actions in $A \setminus (B \cup \{\surd\})$, and the action $\surd$ leads always to the stop process if both processes can perform $\surd$. Since $d^s \geq d^a$, the distance is bounded by d^s if there is at least one action $a \in B$ with $a \neq \surd$ for which the composed processes can evolve synchronously, and otherwise by d^a.

The distance bounds for non-recursive process combinators are tight.

Proposition 8. *Let $\epsilon_i \in [0,1]$. There are $s_i, t_i \in \mathsf{T}(\Sigma_{PA})$ with $\mathbf{d}_\lambda(s_i,t_i) = \epsilon_i$ such that the inequalities in Prop. 6 and 7 become equalities.*

3.3 Compositional reasoning over non-recursive processes

In order to specify and verify systems in a compositional manner, it is necessary that the behavioral semantics is compatible with all operators of the language that describe these systems. There are multiple proposals which properties of process combinators facilitate compositional reasoning. In this section we discuss non-extensiveness [1] and non-expansiveness [7, 8]), which are compositionality properties based on the p-norm. They allow for compositional reasoning over probabilistic processes that are built of non-recursive process combinators. Non-extensiveness and non-expansiveness are very strong forms of uniform continuity. For instance, a non-expansive operator ensures that the distance between the composed processes is at most the sum of the distances between its parts. Later in Sec. 4.3 we will propose uniform continuity as generalization of these properties that allows also for compositional reasoning over recursive processes.

Definition 9 (Non-extensive process combinator). *A process combinator $f \in \Sigma$ is* non-extensive *wrt. λ-bisimulation metric $\mathbf{d}_\lambda$ if or all closed process terms $s_i, t_i \in \mathsf{T}(\Sigma)$*

$$\mathbf{d}_\lambda(f(s_1,\ldots,s_n), f(t_1,\ldots,t_n)) \leq \max_{i=1}^{n} \mathbf{d}_\lambda(s_i,t_i)$$

Theorem 10. *The process combinators probabilistic action prefix $a.\bigoplus_{i=1}^{n}[p_i]\_$, nondeterministic alternative composition $\_ + \_$ and probabilistic alternative composition $\_ +_p \_$ are non-extensive wrt. $\mathbf{d}_\lambda$ for any $\lambda \in (0,1]$.*

Proposition 11. *The process combinators sequential composition* $\_;\_$*, synchronous parallel composition* $\_ \mid \_$*, asynchronous parallel composition* $\_ \mid\mid\mid \_$*, CSP-like parallel composition* $\_ \parallel_B \_$ *and probabilistic parallel composition* $\_ \mid\mid\mid_p \_$ *are not non-extensive wrt.* $\mathbf{d}_\lambda$ *for any* $\lambda \in (0, 1]$.

Note that Thm. 10 follows from Prop. 6, and that Prop. 11 follows from Prop. 7 and Prop. 8. We proceed now with the compositionality property of non-expansiveness.

Definition 12 (Non-expansive process combinator). *A process combinator* $f \in \Sigma$ *is* non-expansive *wrt.* λ*-bisimulation metric* $\mathbf{d}_\lambda$ *if for all closed process terms* $s_i, t_i \in \mathsf{T}(\Sigma)$

$$\mathbf{d}_\lambda(f(s_1, \ldots, s_n), f(t_1, \ldots, t_n)) \leq \sum_{i=1}^{n} \mathbf{d}_\lambda(s_i, t_i)$$

If f is non-extensive, then f is non-expansive.

Theorem 13. *All non-recursive process combinators of* Σ_{PA} *are non-expansive wrt.* $\mathbf{d}_\lambda$ *for any* $\lambda \in (0, 1]$.

Note that Thm. 13 follows from Prop. 6 and Prop. 7. Thm. 13 generalizes a similar result of [8] which considered only PTSs without nondeterministic branching and only a small set of combinators. The analysis which operators are non-extensive (Thm. 10) and the tight distance bounds (Prop. 6 and 7) are novel.

4 Recursive processes

Recursion is necessary to express infinite behavior in terms of finite process expressions. Moreover, recursion allows to express repetitive finite behavior in a compact way. We will discuss now compositional reasoning over probabilistic processes that are composed by recursive process combinators. We will see that the compositionality properties used for non-recursive process combinators (Sec. 3.3) fall short for recursive process combinators. We will propose the more general property of uniform continuity (Sec. 4.3) that captures the inherent nature of compositional reasoning over probabilistic processes. In fact, it allows to reason compositionally over processes that are composed by both recursive and non-recursive process combinators. In the next section we apply these results to reason compositionally over a communication protocol and derive its respective performance properties. To the best of our knowledge this is the first study which explores systematically compositional reasoning over recursive processes in the context of bisimulation metric semantics.

4.1 Recursive process combinator

We define $P_{\mathrm{PA}^\cup}$ as disjoint extension of P_{PA} with the operators finite iteration $\_^n$, infinite iteration $\_^\omega$, binary Kleene-star iteration $\_^*\_$, probabilistic Kleene-star iteration $\_^{*_p}\_$, finite replication $!^n\_$, infinite replication (bang) operator $!\_$, and probabilistic bang operator $!_p\_$. The operational semantics of these operators is specified by the rules in Tab. 3. The finite iteration t^n (resp. infinite iteration t^ω) of process t expresses that t is performed n times (resp. infinitely often) in sequel. The binary Kleene-star is as usual. The bang operator expresses for $!t$ (resp. finite replication $!^n t$) that infinitely many copies (resp. n copies) of t evolve asynchronously. The probabilistic variants of Kleene-star iteration [2, Sec. 5.2.4(vi)] and bang replication [14, Fig. 1] substitute the nondeterministic choice of the non-probabilistic variants by a respective probabilistic choice.

$$\frac{x \xrightarrow{a} \mu}{x^{n+1} \xrightarrow{a} \mu; \delta(x^n)} \qquad \frac{x \xrightarrow{a} \mu}{x^{\omega} \xrightarrow{a} \mu; \delta(x^{\omega})} \qquad \frac{x \xrightarrow{a} \mu}{x^* y \xrightarrow{a} \mu; \delta(x^* y)} \qquad \frac{y \xrightarrow{a} \nu}{x^* y \xrightarrow{a} \nu}$$

$$\frac{x \xrightarrow{a} \mu \quad y \xrightarrow{a} \nu}{x^{*_p} y \xrightarrow{a} \nu \oplus_p \mu; \delta(x^{*_p} y)} \qquad \frac{x \xrightarrow{a} \mu \quad y \not\xrightarrow{a}}{x^{*_p} y \xrightarrow{a} \mu; \delta(x^{*_p} y)} \qquad \frac{x \not\xrightarrow{a} \quad y \xrightarrow{a} \nu}{x^{*_p} y \xrightarrow{a} \nu}$$

$$\frac{x \xrightarrow{a} \mu}{!^{n+1} x \xrightarrow{a} \mu \mid\mid\mid \delta(!^n x)} \qquad \frac{x \xrightarrow{a} \mu}{!x \xrightarrow{a} \mu \mid\mid\mid \delta(!x)} \qquad \frac{x \xrightarrow{a} \mu}{!_p x \xrightarrow{a} \mu \oplus_p (\mu \mid\mid\mid \delta(!_p x))}$$

Tab. 3: Standard recursive process combinators

4.2 Distance between recursive processes

We develop now tight bounds for recursive process combinators.

Proposition 14. *Let $P = (\Sigma, A, R)$ be any PTSS with $P_{PA^{\circlearrowleft}} \sqsubseteq P$. For all $s, t \in \mathsf{T}(\Sigma)$*

(a) $\mathbf{d}_\lambda(s^n, t^n) \leq d^n$
(b) $\mathbf{d}_\lambda(!^n s, !^n t) \leq d^n$
(c) $\mathbf{d}_\lambda(s^\omega, t^\omega) \leq d^\omega$
(d) $\mathbf{d}_\lambda(!s, !t) \leq d^\omega$
(e) $\mathbf{d}_\lambda({s_1}^* s_2, {t_1}^* t_2) \leq \max(\mathbf{d}_\lambda({s_1}^\omega, {t_1}^\omega), \mathbf{d}_\lambda(s_2, t_2))$
(f) $\mathbf{d}_\lambda(s_1^{*_p} s_2, t_1^{*_p} t_2) \leq \mathbf{d}_\lambda({s_1}^* s_2, {t_1}^* t_2)$
(g) $\mathbf{d}_\lambda(!_p s, !_p t) \leq \begin{cases} \mathbf{d}_\lambda(s,t)\frac{1}{1-(1-p)(\lambda-\mathbf{d}_\lambda(s,t))} & \text{if } \mathbf{d}_\lambda(s,t) \in (0,1) \\ \mathbf{d}_\lambda(s,t) & \text{if } \mathbf{d}_\lambda(s,t) \in \{0,1\} \end{cases}$, *with*

$$d^n = \begin{cases} \mathbf{d}_\lambda(s,t)\frac{1-(\lambda-\mathbf{d}_\lambda(s,t))^n}{1-(\lambda-\mathbf{d}_\lambda(s,t))} & \text{if } \mathbf{d}_\lambda(s,t) \in (0,1) \\ \mathbf{d}_\lambda(s,t) & \text{if } \mathbf{d}_\lambda(s,t) \in \{0,1\} \end{cases} \quad d^\omega = \begin{cases} \mathbf{d}_\lambda(s,t)\frac{1}{1-(\lambda-\mathbf{d}_\lambda(s,t))} & \text{if } \mathbf{d}_\lambda(s,t) \in (0,1) \\ \mathbf{d}_\lambda(s,t) & \text{if } \mathbf{d}_\lambda(s,t) \in \{0,1\} \end{cases}$$

First we explain the distance bounds of the nondeterministic recursive process combinators. To understand the distance bound between processes that iterate finitely many times (Prop. 14.a), observe that s^n and $s; \ldots; s$ (where $s; \ldots; s$ denotes n sequentially composed instances of s) denote the same PTSs (up to renaming of states). Recursive application of the distance bound Prop. 7.a yields $\mathbf{d}_\lambda(s^n, t^n) = \mathbf{d}_\lambda(s; \ldots; s, t; \ldots; t) \leq \mathbf{d}_\lambda(s,t) \sum_{k=0}^{n-1} (\lambda - \mathbf{d}_\lambda(s,t)) = d^n$. The same reasoning applies to the finite replication operator (Prop.14.b) by observing that $!^n s$ and $s \mid\mid\mid \ldots \mid\mid\mid s$ denote the same PTSs (up to renaming of states) and that the bounds in Prop. 7.a and 7.c coincide if $s_1 = s_2 = s$ and $t_1 = t_2 = t$. The distance between processes that may iterate infinitely many times (Prop. 14.c), and the distance between processes that may spawn infinite many copies that evolve asynchronously (Prop. 14.d) are the limit of the respective finite iteration and replication bounds. The distance between the Kleene-star iterated processes ${s_1}^* s_2$ and ${t_1}^* t_2$ is bounded by the maximum of the distance $\mathbf{d}_\lambda({s_1}^\omega, {t_1}^\omega)$ (infinite iteration of s_1 and t_1 s.t. s_2 and t_2 never evolve), and the distance $\mathbf{d}_\lambda(s_2, t_2)$ (s_2 and t_2 evolve immediately). The case where s_1 and t_1 iterate n-times and then s_2 and t_2 evolve leads always to a distance $\mathbf{d}_\lambda({s_1}^n, {t_1}^n) + (\lambda - \mathbf{d}_\lambda(s_1, t_1))^n \mathbf{d}_\lambda(s_2, t_2) \leq \max(\mathbf{d}_\lambda({s_1}^\omega, {t_1}^\omega), \mathbf{d}_\lambda(s_2, t_2))$.

Now we explain the bounds of the probabilistic recursive process combinators. The distance between processes composed by the probabilistic Kleene star is bounded by

the distance between those processes composed by the nondeterministic Kleene star (Prop. 14.f), since the second and third rule specifying the probabilistic Kleene star define the same operational behavior as the nondeterministic Kleene star. The first rule which defines a convex combination of these transitions applies only for those actions that both of the combined processes can perform. In fact, $\mathbf{d}_\lambda(s_1 {}^*{}_p\, s_2, t_1 {}^*{}_p\, t_2) = \mathbf{d}_\lambda(s_1{}^* s_2, t_1{}^* t_2)$ if the initial actions that can be performed by processes s_1, t_1 are disjoint from the initial actions that can be performed by processes s_2, t_2 (and hence the first rule defining $\_ {}^*{}_p\, \_$ cannot be applied). Thus, the distance bound of the probabilistic Kleene star coincides with the distance bound of the nondeterministic Kleene star. The bound on the distance of processes composed by the probabilistic bang operator can be understood by observing that $!_p s$ behaves as $!^{n+1} s$ with probability $p(1-p)^n$. Hence, by Prop. 14.b we get $\mathbf{d}_\lambda(!_p s, !_p t) \leq \sum_{n=0}^{\infty} p(1-p)^n \mathbf{d}_\lambda(!^{n+1} s, !^{n+1} t) \leq \sum_{n=0}^{\infty} p(1-p)^n d^{n+1} = \mathbf{d}_\lambda(s,t)/(1-(1-p)(\lambda - \mathbf{d}_\lambda(s,t)))$.

The distance bounds for recursive process combinators are tight.

Proposition 15. *Let $\epsilon_i \in [0,1]$. There are $s_i, t_i \in \mathcal{T}(\Sigma_{PA})$ with $\mathbf{d}_\lambda(s_i, t_i) = \epsilon_i$ such that the inequalities in Prop. 14 become equalities.*

4.3 Compositional reasoning over recursive processes

From Prop. 14 and Prop. 15 it follows that none of the recursive process combinators discussed in this section satisfies the compositionality property of non-expansiveness.

Proposition 16. *All recursive process combinators of $\Sigma_{PA^\cup}$ (unbounded recursion and bounded recursion with $n \geq 2$) are not non-expansive wrt. $\mathbf{d}_\lambda$ for any $\lambda \in (0,1]$.*

However, a weaker property suffices to facilitate compositional reasoning. To reason compositionally over probabilistic processes it is enough if the distance of the composed processes can be related to the distance of its parts. In essence, compositional reasoning over probabilistic processes is possible whenever a small variance in the behavior of the parts leads to a bounded small variance in the behavior of the composed processes.

We introduce uniform continuity as the compositionality property for both recursive and non-recursive process combinators. Uniform continuity generalizes the properties non-extensiveness and non-expansiveness for non-recursive process combinators.

Definition 17 (Uniformly continuous process combinator). *A process combinator $f \in \Sigma$ is* uniformly continuous *wrt. λ-bisimulation metric $\mathbf{d}_\lambda$ if for all $\epsilon > 0$ there are $\delta_1, \ldots, \delta_n > 0$ such that for all closed process terms $s_i, t_i \in \mathcal{T}(\Sigma)$*

$$\forall i = 1, \ldots, n.\ \mathbf{d}_\lambda(s_i, t_i) < \delta_i \implies \mathbf{d}_\lambda(f(s_1, \ldots, s_n), f(t_1, \ldots, t_n)) < \epsilon.$$

Note that by definition each non-expansive operator is also uniformly continuous (by $\delta_i = \epsilon/n$). A uniformly continuous combinator f ensures that for any non-zero bisimulation distance ϵ there are appropriate non-zero bisimulation distances δ_i s.t. for any composed process $f(s_1, \ldots, s_n)$ the distance to the composed process where each s_i is replaced by any t_i with $\mathbf{d}_\lambda(s_i, t_i) < \delta_i$ is $\mathbf{d}_\lambda(f(s_1, \ldots, s_n), f(t_1, \ldots, t_n)) < \epsilon$. We consider the uniform notion of continuity (technically, the δ_i depend only on ϵ and are independent of the concrete states s_i) because we aim at universal compositionality guarantees.

The distance bounds of Sec. 4.2 allow us to derive that finitely recursing process combinators are uniformly continuous wrt. both non-discounted and discounted bisimulation metric (Thm. 18). On the contrary, unbounded recursing process combinators are uniformly continuous only wrt. discounted bisimulation metric (Thm. 19 and Prop. 20).

$$BRP(N,T,p,q) = RC(N,T,p,q) \parallel_B TV, \text{ where } B = \{c(d,b) \mid d \in D, b \in \{0,1\}\} \cup \{ack, lost\}$$

$$RC(N,T,p,q) = \Bigg[\sum_{0 \le n \le N, n=2k} i(n).\Big(CH(0,T,p,q)\,;\, CH(1,T,p,q)\Big)^{\frac{n}{2}} + \sum_{0 \le n \le N, n=2k+1} i(n).\Big(\Big(CH(0,T,p,q)\,;\, CH(1,T,p,q)\Big)^{\frac{n-1}{2}}\,;\, CH(0,T,p,q)\Big)\Bigg]\,;\, res(\mathrm{OK}).\surd$$

$$CH(b,t,p,q) = \sum_{d \in D} i(d).CH'(d,b,t,p,q)$$

$$CH'(d,b,t,p,q) = \begin{cases} (\bot.\, CH'(d,b,t-1,p,q)) \;\oplus_p\; (c(d,b).CH_2(d,b,t,p,q)) & \text{if } t > 0 \\ res(NOK) & \text{if } t = 0 \end{cases}$$

$$CH_2(d,b,t,p,q) = \begin{cases} (lost.CH'(d,b,t-1,p,q)) \;\oplus_q\; (ack.\surd) & \text{if } t > 0 \\ res(NOK) & \text{if } t = 0 \end{cases}$$

$$TV = \Bigg[\Big(\Big(\sum_{d \in D} c(d,1).(ack.\surd + lost.\surd)\Big)^*\Big(\sum_{d \in D} c(d,0).o(d).(ack.\surd + lost.\surd)\Big)\Big)\,; \Big(\Big(\sum_{d \in D} c(d,0).(ack.\surd + lost.\surd)\Big)^*\Big(\sum_{d \in D} c(d,1).o(d).(ack.\surd + lost.\surd)\Big)\Big)\Bigg]^\omega$$

Figure 1: Specification of the Bounded Retransmission Protocol

Theorem 18. *The process combinators finite iteration $\_^n$, finite replication $!^n\_$, and probabilistic replication (bang) $!_p\_$ are uniformly continuous wrt.* $\mathbf{d}_\lambda$ *for any* $\lambda \in (0,1]$.

Note that the probabilistic bang is uniformly continuous wrt. non-discounted bisimulation metric $\mathbf{d}_1$ because in each step there is a non-zero probability that the process is not copied. On contrary, the process $s_1 {}^*{}_p\, s_2$ applying the probabilistic Kleene star creates with probability 1 a copy of s_1 for actions that s_1 can and s_2 cannot perform. Hence, $\_^*{}_p\_$ is uniformly continuous only for discounted bisimulation metric $\mathbf{d}_\lambda$ with $\lambda < 1$.

Theorem 19. *The process combinators infinite iteration $\_^\omega$, nondeterministic Kleene-star iteration $\_^*\_$, probabilistic Kleene-star iteration $\_^*{}_p\_$, and infinite replication (bang) $!\_$ are uniformly continuous wrt.* $\mathbf{d}_\lambda$ *for any* $\lambda \in (0,1)$.

Proposition 20. *The process combinators $\_^\omega$, $\_^*\_$, $!\_$ and $\_^*{}_p\_$ are not uniformly continuous wrt.* $\mathbf{d}_1$.

5 Application

To advocate both uniform continuity as adequate property for compositional reasoning as well as bisimulation metric semantics as a suitable distance measure for performance validation of communication protocols, we exemplify the discussed compositional reasoning method by analyzing the bounded retransmission protocol (BRP) as a case study.

The BRP allows to transfer streams of data from a sender (e.g. a remote control RC) to a receiver (e.g. a TV). The RC tries to send to the TV a stream of n data, $d_0, \ldots, d_{n-1}$, with each d_i a member of the finite data domain D. The length n of the stream is bounded by a given N. Each d_i is sent separately and has probability p to get

lost. When the TV receives d_i, it sends back an acknowledgment message (ack), which may also get lost, with probability q. If the RC does not receive the ack for d_i within a given time, it assumes that d_i got lost and retries to transmit it. However, the maximal number of attempts is T. Since the ack may get lost, it may happen that the RC sends more than once the same datum d_i notwithstanding that it was correctly received by the TV. Therefore the RC attaches a control bit b to each datum d_i s.t. the TV can recognize if this datum is original or already received. Data items at even positions, i.e. d_{2k} for some $k \in \mathbb{N}$, get control bit 0 attached, and data items d_{2k+1} get control bit 1 attached.

The BRP is specified in Fig. 1. Our specification adapts the nondeterministic process algebra specification of [10] by refining the configuration of lossy channels. While in the nondeterministic setting a lossy channel (nondeterministically) either successfully transmits a datum or looses it, we attached a success and failure probability to this choice. The protocol specification $BRP(N, T, p, q)$ represents a system consisting of the RC modeled as process $RC(N, T, p, q)$, the TV modeled as process TV, and the channels $CH(b, t, p, q)$ for data transmission and $CH_2(d, b, t, p, q)$ for acknowledgment. The processes $RC(N, T, p, q)$ and TV synchronize over the actions: (i) $c(d, b)$, modeling the correct transmission of pair (d, b) from the RC to the TV; (ii) ack, modeling the correct transmission of the ack from the TV to the RC, and (iii) $lost$, used to model the timeout due to loss of the ack. Timeout due to the loss of pair (d, b) is modeled by action $\perp$ by the RC. $RC(N, T, p, q)$ starts by receiving the size $n \leq N$ of the data stream, by means of action $i(n)$. Then, for n times it reads the datum d_i by means of action $i(d)$ and tries to send it to the TV. If all data are sent successfully, then the other RC components are notified by means of action $res(OK)$. In case of T failures for one datum, the whole transmission fails and emits $res(NOK)$. If TV receives a pair (d, b) by action $c(d, b)$ then, if d is original, namely b is the expected control bit, then d is sent to other TV components by $o(d)$, otherwise (d, b) is ignored.

To advocate bisimulation metric semantics as a suitable distance measure for performance validation of communication protocols we translate performance properties of a BRP implementation with lossy channels $BRP(N, T, p, q)$ to the bisimulation distance between this implementation and the specification with perfect channels $BRP(N, T, 0, 0)$.

Proposition 21. *Let $N, T \in \mathbb{N}$ and $p, q \in [0, 1]$.*

(a) Bisimulation distance $\mathbf{d}(BRP(N, T, 0, 0), BRP(N, T, p, q)) = \epsilon$ relates as follows to the protocol performance properties:
- *The likelihood that N data items are sent and acknowledged without any retry (i.e. $BRP(N, T, p, q)$ behaves as $BRP(N, T, 0, 0)$) is $1 - \epsilon$.*
- *The likelihood that N data items are sent and acknowledged with at most $k \leq N \cdot T$ retries is $(1 - \epsilon)\frac{1-(1-(1-\epsilon)^{1/N})^k}{(1-\epsilon)^{1/N}}$.*
- *The likelihood that N items are sent and acknowledged is $(1 - \epsilon)\frac{1-(1-(1-\epsilon)^{1/N})^{N \cdot T}}{(1-\epsilon)^{1/N}}$.*

(b) Bisimulation distance $\mathbf{d}(CH(b, T, 0, 0), CH(b, T, p, q)) = \delta$ relates as follows to the channel performance properties:
- *The likelihood that one datum is sent and acknowledged without retry is $1 - \delta$.*
- *The likelihood that one datum is sent and acknowledged with at most $k \leq T$ retries is $1 - \delta^k$.*

Now we show that by applying the compositionality results in Prop. 6, 7, 14 we can relate the bisimulation distance between the specification $BRP(N,T,0,0)$ and some implementation $BRP(N,T,p,q)$ of the entire protocol with the distances between the specification and some implementation of its respective components. On the one hand, this allows to derive from specified performance properties of the entire protocol individual performance requirements of its components (compositional verification). On the other hand, it allows to infer from performance properties of the protocol components suitable performance guarantees on the entire protocol (compositional specification).

Proposition 22. *Let $N,T \in \mathbb{N}$ and $p,q \in [0,1]$. For all $d \in D$ and $b \in \{0,1\}$*

(a) $\mathbf{d}(BRP(N,T,0,0), BRP(N,T,p,q)) \leq 1-(1-\mathbf{d}(CH(b,T,0,0), CH(b,T,p,q)))^N$
(b) $\mathbf{d}(CH(b,T,0,0), CH(b,T,p,q)) = 1-(1-p)(1-q)$

Prop. 22.a follows from Props. 6, 7, 14 and Prop. 22.b from Props. 6, 7.

To advocate uniform continuity as adequate property for compositional reasoning, we show that the uniform continuity of process combinators in $BRP(N,T,p,q)$ allows us to relate the distance between this implementation and the specification $BRP(N,T,0,0)$ (which relates by Prop. 21 to performance properties of the entire protocol) to the concrete parameters p,q and N of the system. In detail, by Thm. 10, 13, 18 and Prop. 22 we get $\mathbf{d}(BRP(N,T,p,q), BRP(N,T,0,0)) \leq N/2 \cdot (\mathbf{d}(CH(0,T,p,q), CH(0,T,0,0)) + \mathbf{d}(CH(1,T,p,q), CH(1,T,0,0))) \leq N(1-(1-p)(1-q))$. We infer the following result.

Proposition 23. *Let $N,T \in \mathbb{N}$ and $p,q \in [0,1]$. For all $\epsilon \geq 0$, $p+q-pq < \epsilon/N$ ensures*

$$\mathbf{d}(BRP(N,T,p,q), BRP(N,T,0,0)) < \epsilon$$

Combining Prop. 21 – 23 allows us now to reason compositionally over a concrete scenario. We derive from a given performance requirement to transmit a stream of data the necessary performance properties of the channel components.

Example 24. Consider the following scenario. We want to transmit a data stream of $N = 20$ data items with at most $T = 1$ retry per data item. We want to build an implementation that should satisfy the performance property 'The likelihood that all 20 data items are successfully transmitted is at least 99%'. By Prop. 21.a we translate this performance property to the resp. bisimulation distance $\mathbf{d}(BRP(N,T,0,0), BRP(N,T,p,q)) \leq 0.01052$ on the entire system. By Prop. 22.a we derive the bisimulation distance for its channel component $\mathbf{d}(CH(b,T,0,0), CH(b,T,p,q) \leq 0.00053$. By Prop. 22.b this distance can be translated to appropriate parameters of the channel component, e.g. $p = 0.0002$ and $q = 0.00032$ or equivalently $p = 0.020\%$ and $q = 0.032\%$. Finally, Prop. 21.b allows to translate the distance between the specification and implementation of the channel component back to an appropriate performance requirement, e.g. 'The likelihood that one datum is successfully transmitted is at least 99.95%'. ■

6 Conclusion

We argued that uniform continuity is an appropriate property of process combinators to facilitate compositional reasoning wrt. bisimulation metric semantics. We showed that all standard (non-recursive and recursive) process algebra operators are uniformly continuous. In addition, we provided tight bounds on the distance between the composed

processes. We exemplified how these results can be used to reason compositionally over protocols. In fact, they allow to derive from performance requirements on the entire system appropriate performance properties of the respective components, and in reverse to induce from performance assumptions on the system components performance guarantees on the entire system.

We will continue this line of research as follows. First, we generalize the analysis of concrete process algebra operators as discussed in this paper to general SOS rule and specification formats. Preliminary results show that in essence, a process combinator is uniformly continuous if the combined processes are copied only finitely many times along their evolution [11]. Then, we explore further (as initiated in Sec. 5) the relation between various behavioral distance measures, e.g. convex bisimulation metric [5], trace metric [9], and total-variation distance based metrics [13] with performance properties of communication and security protocols. This will provide further practical means to apply process algebraic methods and compositional metric reasoning wrt. uniformly continuous process combinators.

References

1. Bacci, G., Bacci, G., Larsen, K.G., Mardare, R.: Computing Behavioral Distances, Compositionally. In: Proc. MFCS'13, pp. 74–85. Springer (2013)
2. Bartels, F.: On Generalised Coinduction and Probabilistic Specification Formats. Ph.D. thesis, VU University Amsterdam (2004)
3. van Breugel, F., Worrell, J.: A Behavioural Pseudometric for Probabilistic Transition Systems. TCS 331(1), 115–142 (2005)
4. D'Argenio, P.R., Lee, M.D.: Probabilistic Transition System Specification: Congruence and Full Abstraction of Bisimulation. In: Proc. FoSSaCS'12. LNCS, vol. 7213, pp. 452–466. Springer (2012)
5. De Alfaro, L., Majumdar, R., Raman, V., Stoelinga, M.: Game relations and metrics. In: Proc. LICS'07. pp. 99–108. IEEE (2007)
6. De Alfaro, L., Henzinger, T.A., Majumdar, R.: Discounting the Future in Systems Theory. In: Proc. ICALP'03, pp. 1022–1037. ICALP '03, Springer (2003)
7. Deng, Y., Chothia, T., Palamidessi, C., Pang, J.: Metrics for Action-labelled Quantitative Transition Systems. ENTCS 153(2), 79–96 (2006)
8. Desharnais, J., Gupta, V., Jagadeesan, R., Panangaden, P.: Metrics for Labelled Markov Processes. TCS 318(3), 323–354 (2004)
9. Fahrenberg, U., Legay, A.: The quantitative linear-time-branching-time spectrum. TCS 538(0), 54 – 69 (2014)
10. Fokkink, W.: Modelling Distributed Systems. Springer (2007)
11. Gebler, D., Tini, S.: Fixed-point Characterization of Compositionality Properties of Probabilistic Processes Combinators. In: Proc. EXPRESS/SOS'14. EPTCS, vol. 160, pp. 63–78 (2014)
12. Lee, M.D., Gebler, D., D'Argenio, P.R.: Tree Rules in Probabilistic Transition System Specifications with Negative and Quantitative Premises. In: Proc. EXPRESS/SOS'12. EPTCS, vol. 89, pp. 115–130 (2012)
13. Mio, M.: Upper-Expectation Bisimilarity and Łukasiewicz μ-Calculus. In: Proc. FoSSaCS'14. pp. 335–350. Springer (2014)
14. Mio, M., Simpson, A.: A Proof System for Compositional Verification of Probabilistic Concurrent Processes. In: Proc. FoSSaCS'13. LNCS, vol. 7794, pp. 161–176. Springer (2013)
15. Segala, R.: Modeling and Verification of Randomized Distributed Real-Time Systems. Ph.D. thesis, MIT (1995)