Theoretical Computer Science 869 (2021) 181–194

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

# Model independent approach to probabilistic models

Yuxi Fu

*BASICS, Shanghai Jiao Tong University, China*

A R T I C L E   I N F O

*Article history:*
Received 21 March 2020
Received in revised form 30 March 2021
Accepted 1 April 2021
Available online 7 April 2021
Communicated by R. van Glabbeek

*Keywords:*
Probabilistic process
Bisimulation
Divergence

A B S T R A C T

There is a lot of research on probabilistic transition systems. There are not many studies in probabilistic process models. The lack of investigation into the interactive aspect of probabilistic processes is mainly due to the difficulty caused by the discrepancy between probabilistic choices and nondeterministic actions. The paper proposes a uniform approach to probabilistic process models and a bisimulation theory for probabilistic concurrency.



## 1. Introduction

Randomization plays an indispensable role in computer science. The celebrated result, the PCP Theorem [3], reveals the power of

$$\text{interaction} + \text{randomness} + \text{error}$$

in problem solving. Given an NP complete problem, one may design an interactive proof system consisting of a verifier and a prover [22,4]. Upon receiving a problem instance the verifier accepts or rejects the input with high confidence in polynomial time by using logarithmic random bits and asking a constant number of questions to the prover. The scenario can be generalized to a multi-prover situation with an increased power on the verifier side [10,5,17]. This fundamental result is significant to modern computing systems, which are open, distributed, interactive, and have both nondeterministic behaviors and randomized choices. To formalize models in which results like the PCP Theorem apply, one may introduce randomization to interaction models (process models). There are two kinds of randomness in randomized process models. A process may send a random value to another; and it may randomly choose whom it will send a value to. We call the former *content randomness* and the latter *channel randomness*. Content randomness is basically a computational issue [42], whereas channel randomness has to do with interaction.

What kind of channel randomness are there? In literature one finds basically two answers to the question [28,23,44,35,19], the generative scenario and the reactive scenario. *Generative models* feature probabilistic choice for external actions. The standard syntax for a probabilistic guarded choice term is of the form

$$\bigoplus_{i\in I} p_i \ell_i.T_i, \tag{1}$$

*E-mail address:* fu-yx@cs.sjtu.edu.cn.

https://doi.org/10.1016/j.tcs.2021.04.001
0304-3975/

where $p_i \in (0,1)$ and $\Sigma_{i\in I} p_i = 1$. The infix notation $p_1\ell_1.T_1 \oplus \ldots \oplus p_k\ell_k.T_k$ is frequently used. The semantics is defined by $\bigoplus_{i\in I} p_i\ell.T_i \xrightarrow{\ell_i}_{p_i} T_i$, meaning that $\bigoplus_{i\in I} p_i\ell.T_i$ may evolve into $T_i$ with probability $p_i$ by performing the action $\ell_i$. The generative model is problematic in the presence of the *interleaving* composition operator "|" and the localization operator "$(a)$". Let $A$ be $\frac{1}{2}a \oplus \frac{1}{2}b$ and $C$ be $\frac{2}{3}\overline{b} \oplus \frac{1}{3}\overline{c}$. What is then the behavior of $A \mid C$? And how about $(b)(A \mid C)$, where the channel $b$ is local in the sense that $A$ and $C$ may interact through channel $b$ but neither is allowed to interact with any other process at $b$? What is the probability of $A$ interacting with $C$ at channel $b$ in $(a)(c)(A \mid C)$? In $(c)C$ interaction at channel $c$ is disabled. How does that reconcile with the prescription that $C$ interacts at channel $c$ with probability 1/3? It does not sound right to say that $(c)C$ performs the $\overline{b}$ action with probability one. A reasonable semantics is that $(c)C$ may do the $\overline{b}$ action with probability 2/3 and becomes dead with probability 1/3. If this is indeed the interpretation, $C$ should really be $\frac{2}{3}\overline{b} \oplus \frac{1}{3}\tau.\overline{c}$. Symmetrically one may argue that $\frac{2}{3}\overline{b} \oplus \frac{1}{3}\tau.\overline{c}$ should really be $\frac{2}{3}\tau.\overline{b} \oplus \frac{1}{3}\tau.\overline{c}$. All problems with the probabilistic choice (1) are gone if it is replaced by the *random choice* term

$$\bigoplus_{i\in I} p_i\tau.T_i, \tag{2}$$

where the size of the finite index set $I$ is at least 2 and $\sum_{i\in I} p_i = 1$. Thus $0 < p_i < 1$ for all $i \in I$. Early generative models are *fully probabilistic* [8]. Nondeterminism was considered later [35].

In reactive models, introduced by Larsen and Skou [28] and popularized by the work of van Glabbeek, Smolka and Steffen [19], nondeterministic choice and probabilistic choice come in alternation. Using a suggestive notation one may write for example

$$a.\left(\frac{1}{2}A_1 \uplus \frac{1}{2}A_2\right) + b.\left(\frac{1}{3}B_1 \uplus \frac{2}{3}B_2\right). \tag{3}$$

This is a process that may perform an $a$ action and turns into $A_1$ with probability 1/2 and $A_2$ with probability 1/2. It may also do an interaction at channel $b$ and becomes $B_1$ with probability 1/3 and $B_2$ with probability 2/3. A systematic exposure of the research progress on reactive models is given in Deng's book [14]. In literature authors often think of $\frac{1}{2}A_1 \uplus \frac{1}{2}A_2$ simply as a distribution over $\{A_1, A_2\}$. In this paper we take the view that the distribution can only be achieved by carrying out a certain amount of computation, say invoking a random number generator. The details of the computation can be abstracted away, but it should be formalized as an internal action. In our opinion the best way to understand the process in (3) is to see it as a simplification of

$$a.\left(\frac{1}{2}\tau.A_1 \oplus \frac{1}{2}\tau.A_2\right) + b.\left(\frac{1}{3}\tau.B_1 \oplus \frac{2}{3}\tau.B_2\right). \tag{4}$$

The process in (4) may do an external nondeterministic choice, and then an internal random choice. This is why reactive models are also called (strict) alternating models. However once we have separated the two kinds of choice, there is no point in insisting on the alternation. What it means is that we might as well give up on the generative probabilistic choice and the reactive probabilistic choice altogether in favor of (2) plus nondeterministic choice.

The central issue in defining a probabilistic process model is the treatment of nondeterminism in the presence of probabilistic choice. The philosophy we shall be following in this paper is that nondeterminism is an attribute of interaction while randomness is a computational feature. Nondeterminism is a characteristic of system, which cannot be implemented. Randomness is a process property, which can be implemented with a negligible error. We advocate in this paper a model independent methodology that turns an interaction model into a randomized interaction model by adjoining (2). The semantics of the random operator is defined by the following rule.

$$\frac{}{\bigoplus_{i\in I} p_i\tau.T_i \xrightarrow{p_i\tau} T_i}. \tag{5}$$

We emphasize that the label $p_i\tau$ should be understood as the same thing as $\tau$. The additional information attached by $p_i$ is to help reason with the bisimulation semantics. Talking about bisimulation equivalence it is useful to think of the transitions defined by (5) as a single silent transition. We introduce the *collective silent transition*

$$\bigoplus_{i\in I} p_i\tau.T_i \xrightarrow{\coprod_{i\in I} p_i\tau} \coprod_{i\in I} T_i, \tag{6}$$

where $\coprod$ is an auxiliary notation introduced to indicate a collection of things. It ought to be clear that the collective silent transitions are closed under composition, localization and recursion.

Strong bisimulation for probabilistic labeled transition system, pLTS for short, is well understood [28,23,35,19,14]. Weak bisimulation has been studied for reactive models [37,14] and alternation models [33]. In the presence of probabilistic choice a silent transition sequence appears as a tree of silent transitions. Schedulers, adversaries and strategies are introduced to resolve the nondeterminism when constructing such a tree. Branching bisimulations have also been studied for reactive models [37].

Our current understanding of weak/branching bisimulations for nondeterministic probabilistic process models is not very satisfactory in several accounts. Here are a summary of the major points.

- In the presence of both nondeterminism and probabilistic choice, proving the transitivity of a proposed observational equivalence has been a challenge. A proof of the transitivity of branching bisimilarity in such a model is not available for some time until the publication of [13] in 2020. Even the proof of the transitivity in [13] is conditional, the transitivity only holds for divergence-free processes.
- As far as we are aware of almost all the weak and branching bisimilarities studied in literature fail to be a full scale congruence relation. They are not closed under the composition operator [33,2], or the localization operator, or the recursion operator. In fact some of them are closed in none of the three operators. There are suggestions to look at synchronous probabilistic process models [19,8]. A basic problem in the synchronous scenario is if internal actions are synchronized. A yes answer seems to contradict to the very idea of the observational theory. But if the silent transitions are not synchronized, the composition operator is unlikely to be associative. There has been research on branching bisimilarity for nondeterministic probabilistic processes defined over a signature [29], focusing on rule formats that guarantee congruence. The negative premises in rule formats impose strong restriction on the operational semantics. It is not clear how the congruence result in [29] applies to process calculi whose composition, localization and recursion operators are defined without any negative premises. A different approach to attack the issue of congruence is to introduce weak equivalence up to a behavioral distance defined on a probabilistic metric [27]. The point is to measure closeness rather than equality. Congruence is achieved by trading off precision. There has been research to strive for a congruent branching bisimilarity by imposing additional conditions. In [1] the authors introduce a branching style bisimilarity for alternation models. The equivalence is a congruence, but the additional requirement is so strong that the equivalence fails to identify $\tau.\left(\frac{1}{2}a \uplus \frac{1}{2}b\right)$ to $\frac{1}{2}a \uplus \frac{1}{2}b$, using the reactive notation. In a recent paper [12], a branching bisimulation congruence is defined for closed terms generated over a signature. The study is confined exclusively to divergence-free closed terms. Thus the equivalence is not closed under recursion since it introduces divergence.
- A consequence of the failure to account for the composition and localization is that most results, even definitions, apply to only finite state probabilistic processes [33,2,14]. The coincidence between the weak bisimilarity and the branching bisimilarity for example is only proved for the finite state fully probabilistic processes [8]. In fact in literature probabilistic processes are often defined as finite probabilistic labeled transition system [36], or finite labeled graphs [33], or labeled concurrent Markov chains [43], or Probabilistic/Markov automata [40,41]. These restricted models preempt any study on process operators. It is difficult to judge if an observational equivalence $\asymp$ defined for finite state probabilistic processes is a good equivalence for a full scale process model, since both composition and recursion are ignored when studying the finite state processes, and there are more than one way to extend or modify $\asymp$ to take care of the composition and recursion operators. We will come back to this point in Section 6.
- The divergence issue has not been properly addressed. This is definitely an omission, especially so in the presence of random silent actions. In the probabilistic process theory it is important that a process equality can make a distinction between divergence with probability one and divergence with probability zero. The congruence relations studied in [12, 13] cannot make such a distinction because they are defined for non-divergent processes. In the classical process theory, no process equivalence would rely on the absence of divergence.

In summary in the setting of nondeterministic probabilistic model, no observational divergence sensitive bisimulation relation has been shown to be both an equivalence and a congruence for a full-fledged process model. A full fledged process model should contain at least the concurrent composition operator, the localization operator and some form of recursion operator.

In both probabilistic programming and quantum computing, only internal probabilistic choice is available. The operator defined in (5) does appear to be universal in computation models that admit randomness. The main task of the paper is to justify this model independent methodology. We hope to convince the reader not only that randomization of process calculi ought to be model independent, but also that the bisimulation theory of the randomized version of any process model $\mathbb{M}$ can be derived from the bisimulation theory of $\mathbb{M}$ in a uniform manner. Section 2 defines a randomized process model. For simplicity the model is taken to be a sub-model of Milner's CCS. Section 3 introduces $\epsilon$-tree and showcases its role in transferring bisimulation theory of a model $\mathbb{M}$ to bisimulation theory of randomized $\mathbb{M}$. Section 5 provides a technical justification for two crucial definitions of the paper. Section 4 proves the congruence property of the bisimulation equivalence. Section 6 makes some final comments.

## 2. Random process model

Let *Chan* be the set of channels, ranged over by lowercase letters. Let $\overline{Chan} = \{\overline{a} \mid a \in Chan\}$. The set $Chan \cup \overline{Chan}$ will be ranged over by small Greek letters. We let $\overline{\alpha} = a$ if $\alpha = \overline{a}$. The set of actions is $Act = Chan \cup \overline{Chan} \cup \{\tau\}$. We write $\ell$ and its decorated versions for elements of *Act*. The grammar of CCS [30] is defined as follows:

$$S, T := X \mid \sum_{i \in I} \alpha_i.T_i \mid S \mid T \mid (a)T \mid \mu X.T, \tag{7}$$

where the indexing set $I$ is finite. We write $\mathbf{0}$ for the *nondeterministic term* $\sum_{i\in\emptyset}\alpha_i.T_i$ in which $\emptyset$ is the empty set. A trailing $\mathbf{0}$ is often omitted. We also use the infix notation of $\sum$, writing for example $\alpha_1.T_1 + \alpha_2.T_2 + \alpha_3.T_3$. A process variable $X$ that appears in $\sum_{i\in I}\alpha_i.T_i$ is guarded. We shall assume that in the *fixpoint term* $\mu X.T$ the *bounded* variable $X$ is guarded in $T$. A term is a *process* if it contains no free variables. We write $A, B, C, D, E, F, G, H, P, Q$ for processes. Let $\mathcal{T}_{\mathrm{CCS}}$ be the set of all CCS terms and $\mathcal{P}_{\mathrm{CCS}}$ be the set of all CCS processes. A *finite state* term/process is a term/process that contains neither the composition operator nor the localization operator. We can define $\tau$-prefix in the standard manner. For example $a.A + \tau.B$ can be defined by $(c)(\overline{c} \mid (a.A + c.B))$ for some fresh channel $c$. From now on we shall use this derived notation without further comment. The transition semantics of CCS is generated by the following rules.

$$\frac{}{\sum_{i\in I}\alpha_i.T_i \xrightarrow{\alpha_i} T_i} \qquad \frac{S \xrightarrow{\overline{\alpha}} S' \quad T \xrightarrow{\alpha} T'}{S \mid T \xrightarrow{\tau} S' \mid T'} \qquad \frac{T \xrightarrow{\ell} T'}{S \mid T \xrightarrow{\ell} S \mid T'}$$

$$\frac{S \xrightarrow{\ell} S'}{S \mid T \xrightarrow{\ell} S' \mid T} \qquad \frac{T \xrightarrow{\ell} T'}{(a)T \xrightarrow{\ell} (a)T'}\ a \notin \ell \qquad \frac{T\{\mu X.T/X\} \xrightarrow{\ell} T'}{\mu X.T \xrightarrow{\ell} T'}$$

In the rule defining the semantics for the localization operator, $a \notin \ell$ means that $a$ does not appear in $\ell$.

For an equivalence $\mathcal{E}$ on $\mathcal{P}_{\mathrm{CCS}}$ we write $A\,\mathcal{E}\,B$ for $(A, B) \in \mathcal{E}$. The advantage of the infix notation is that we may write for example $A\mathcal{E}B\mathcal{E}C$ for $A\mathcal{E}B \wedge B\mathcal{E}C$ and $A \xrightarrow{\ell} B\mathcal{E}C$ for $A \xrightarrow{\ell} B \wedge B\mathcal{E}C$. The notation $\mathcal{P}_{\mathrm{CCS}}/\mathcal{E}$ stands for the set of equivalence classes defined by $\mathcal{E}$. The equivalence class containing $A$ is denoted by $[A]_{\mathcal{E}}$, or $[A]$ when the equivalence is clear from context. We write $A \xrightarrow{\tau}_{\mathcal{E}} A'$ if $A \xrightarrow{\tau} A'\mathcal{E}A$, and $\Longrightarrow_{\mathcal{E}}$ for the reflexive and transitive closure of $\xrightarrow{\tau}_{\mathcal{E}}$. For $\mathcal{C} \in \mathcal{P}_{\mathrm{CCS}}/\mathcal{E}$ we write $A \xrightarrow{\ell} \mathcal{C}$ for the fact that $A \xrightarrow{\ell} A' \in \mathcal{C}$ for some $A'$. A process $A$ is *$\mathcal{E}$-divergent* if there is an infinite silent sequence $A \xrightarrow{\tau}_{\mathcal{E}} A_1 \xrightarrow{\tau}_{\mathcal{E}} \ldots \xrightarrow{\tau}_{\mathcal{E}} A_k \xrightarrow{\tau}_{\mathcal{E}} \ldots$.

The *Randomized CCS*, RCCS for short, is defined on top of CCS. The RCCS terms are obtained by extending the definition in (7) with the randomized choice term defined in (2). A variable that appears in $\bigoplus_{i\in I} p_i\tau.T_i$ is also guarded. The transition semantics of RCCS is defined by the above rules of CCS plus the rule defined in (5). The label $\ell$ that appears in these rules ranges over $Act \cup \{p\tau \mid 0 < p < 1\}$. The set of RCCS terms is denoted by $\mathcal{T}_{\mathrm{RCCS}}$ and that of RCCS processes by $\mathcal{P}_{\mathrm{RCCS}}$ or simply $\mathcal{P}$.

We shall find it convenient to interpret $T \xrightarrow{1\tau} T'$ as $T \xrightarrow{\tau} T'$. So $\xrightarrow{p\tau}$ is a random silent transition if $0 < p < 1$ and an interaction if $p = 1$. The (reflexive and) transitive closure of $\xrightarrow{\tau}$ is denoted by $\stackrel{\tau}{\Longrightarrow}$ ($\Longrightarrow$). We shall say that the product $p_1 \ldots p_k$ is the probability of the silent transition sequence $T \xrightarrow{p_1\tau} \ldots \xrightarrow{p_k\tau} T'$.

## 3. Epsilon tree

Weak bisimulation equivalence [30,32] has been a dominant observational equivalence in process theory. A refinement of the equivalence is the branching bisimulation equivalence introduced by van Glabbeek and Weijland [20,21]. The advantage of branching bisimulation over weak bisimulation has been demonstrated in a number of scenarios. Branching bisimulation admits more stable logical characterization [31] and more efficient equivalence checking algorithm [11]. From the point of programming verification there are good reasons to use branching bisimulation rather than weak bisimulation. Suppose *Spec* is a specification and *Impl* claims to be an implementation of *Spec*. The correctness of *Impl* with regard to *Spec* can be defined as $Impl \approx Spec$, where $\approx$ is the weak bisimilarity. Now *Spec* specifies only what to do, not how to do. It is reasonable to assume that *Spec* contains neither the composition operator "|" nor the $\tau$ prefixing operator because the former is an implementation operator whereas the latter is an indication of computation which a specification should not be bothered with. If $Impl \stackrel{\tau}{\Longrightarrow} Impl'$, then it must be simulated by *Spec* vacuously, meaning that $Impl' \approx Spec$. What this equivalence says is that $Impl \simeq Spec$, where $\simeq$ is the branching bisimilarity. We conclude that as far as the correctness of an implementation is concerned, the right equality is a branching bisimulation. How about program equivalence? Suppose *Pr* and *Pr'* are equivalent programs in the sense that $Pr \approx Pr'$. In practice this means that both *Pr* and *Pr'* are implementations of some specification *Sp*, in other words $Pr \approx Sp \approx Pr'$. It follows from the above argument that $Pr \simeq Sp \simeq Pr'$. Hence $Pr \simeq Pr'$. So program equivalence is also a branching bisimulation. From another perspective a well known fact is that the weak bisimilarity coincides with the branching bisimilarity on the finite-state fully probabilistic processes [8]. It makes sense to look at the branching bisimulation when the model is extended to infinite state probabilistic processes with nondeterministic choice.

There are also arguments for branching bisimulations at a definitional level. For any process equality $\asymp$ on $\mathcal{P}_{\mathrm{CCS}}$ one thinks of a silent transition $A \xrightarrow{\tau}_{\asymp} A'$ as *state-preserving*, and a silent transition $A \xrightarrow{\tau} A'$ such that $A' \not\asymp A$ as *state-changing*. The basic idea of van Glabbeek and Weijland is that a state-changing silent action must be explicitly bisimulated whereas state-preserving silent actions are ignorable. If $B \asymp A \xrightarrow{\tau}_{\asymp} A'$ then $B$ does not have to do anything because $B \asymp A'$. If $B \asymp A \xrightarrow{\tau} A' \not\asymp A$ then $A \xrightarrow{\tau} A'$ must be simulated by some $B \stackrel{\tau}{\Longrightarrow} B'$. If $\asymp$ is a weak bisimulation then before reaching to $B'$ the process $B$ may pass some processes/states that are equivalent neither to $B$ nor to $B'$. In other words $B \stackrel{\tau}{\Longrightarrow} B'$ is *not* simulated by $A \xrightarrow{\tau} A'$ in general. Branching bisimulation requires additionally that $B \stackrel{\tau}{\Longrightarrow} B'$ must be simulated by $A \xrightarrow{\tau} A'$. It is in this sense that $A \xrightarrow{\tau} A'$ is *bi*simulated by $B \stackrel{\tau}{\Longrightarrow} B'$ in the branching bisimulation case. A minute's thought would make us believe that $B \stackrel{\tau}{\Longrightarrow} B'$ must be of the form $B \Longrightarrow_{\asymp} \xrightarrow{\tau} B' \asymp A'$. With these remarks in mind let us state the notion of branching bisimulation [20,6].

**Definition 1.** An equivalence $\mathcal{E}$ on $\mathcal{P}_{\text{CCS}}$ is a *branching bisimulation* if for all $A, B \in \mathcal{P}_{\text{CCS}}$, and for all $\ell$ and all $\mathcal{C} \in \mathcal{P}_{\text{CCS}}/\mathcal{E}$ such that $\left(\ell \in Chan \cup \overline{Chan}\right) \vee (\ell = \tau \wedge \mathcal{C} \neq [A]_{\mathcal{E}})$, the following is valid.

- If $B\,\mathcal{E}\,A \Longrightarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$, then $B \Longrightarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$.

Clearly $B\,\mathcal{E}\,A \xrightarrow{\tau}_{\mathcal{E}} A'$ implies $B\,\mathcal{E}\,A'$, meaning that $A \xrightarrow{\tau}_{\mathcal{E}} A'$ is bisimulated by $B$ vacuously. That explains the condition $\left(\ell \in Chan \cup \overline{Chan}\right) \vee (\ell = \tau \wedge \mathcal{C} \neq [A]_{\mathcal{E}})$.

The well-known extensional equality for computation is defined as follows: $f = g$ if and only if for every input $x$, if one of $f(x), g(x)$ is defined then both of $f(x), g(x)$ are defined and $f(x) = g(x)$. This equality never identifies a nonterminating computation to a terminating computation. The best way to formalize this requirement in bisimulation semantics is introduced in [34]. It is the key condition that turns a bisimulation equality for interaction to an equality for both interaction and computation.

**Definition 2.** An equivalence $\mathcal{E}$ on $\mathcal{P}_{\text{CCS}}$ is *divergence-sensitive* if, for every $\mathcal{C} \in \mathcal{P}_{\text{CCS}}/\mathcal{E}$, either all members of $\mathcal{C}$ are $\mathcal{E}$-divergent, or no member of $\mathcal{C}$ is $\mathcal{E}$-divergent.

It is not difficult to prove that the equivalence closure of the union of all divergence-sensitive branching bisimulation on $\mathcal{P}_{\text{CCS}}$ is a divergence-sensitive branching bisimulation. Let $=_{\text{CCS}}$ denote this equivalence. Another way to look at $=_{\text{CCS}}$ is that it is the largest codivergent branching bisimulation [18].

Having motivated the bisimulation equality for CCS, we are in a position to randomize it as it were to a bisimulation equality for RCCS. One finds in literature that bisimulation equivalences have been defined with the help of schedulers, adversaries, or strategies to resolve nondeterministic choice. These are generalizations of environments in the non-probabilistic setting. Just as in the classical process theory where one always looks for characterizations of process equivalences without referring to any environments [30], one seeks definition of bisimulation equivalence for random processes without using schedulers and the like. Such a definition would make equivalence reasoning much more manageable. In RCCS a silent transition is generally a distribution over a finite set of silent transitions. A finite sequence of silent transitions in CCS then turns into a silent transition tree in RCCS. To describe that, we introduce an auxiliary definition.

**Definition 3.** Suppose $\mathcal{E}$ is an equivalence on $\mathcal{P}_{\text{RCCS}}$ and $A \in \mathcal{P}_{\text{RCCS}}$. A *silent tree $t$ of $A$* is a labeled tree rendering true the following statements.

- Every node of $t$ is labeled by an element of $\mathcal{P}_{\text{RCCS}}$. The root of $t$ is labeled by $A$.
- The edges are labeled by elements of $(0, 1]$. If an edge from a node labeled $A'$ to a node labeled $A''$ is labeled $p$, then $A' \xrightarrow{p\tau} A''$.

An *$\mathcal{E}$-tree $t^A$ of $A$* is a silent tree of $A$ such that all the labels of the nodes of $t^A$ are in $[A]_{\mathcal{E}}$.

If we confuse a node with its label, we may say for example that $A' \xrightarrow{q} A''$ is an edge in $t^A$. The following definition formalizes state-preserving silent transition sequence in the probabilistic setting, wherein the notation $[k]$ stands for the set $\{1, \ldots, k\}$.

**Definition 4.** An *$\epsilon$-tree $t^A_{\mathcal{E}}$ of $A$ with regard to $\mathcal{E}$* is an $\mathcal{E}$-tree of $A$ rendering true the following.

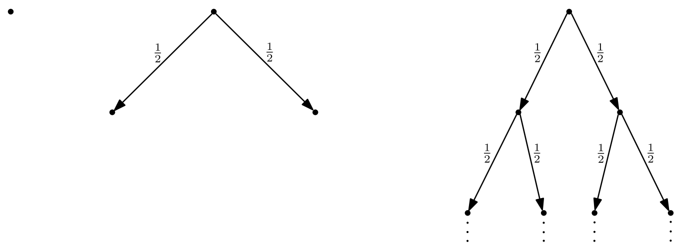
1. If $B \xrightarrow{q} B'$ is in the tree, there must be some collective silent transition $B \xrightarrow{\coprod_{i\in[k]} p_i\tau} \coprod_{i\in[k]} B_i$ such that $B \xrightarrow{p_i} B_i$ is in the tree for every $i \in [k]$ and $B_1, \ldots, B_k$ are the only children of $B$.
2. If $B \xrightarrow{1} B'$ is in the tree, then $B \xrightarrow{\tau} B'$ and $B'$ is the only child of $B$.

Intuitively an $\epsilon$-tree of $A$ with regard to $\mathcal{E}$ is meant to be a random version of $\Longrightarrow_{\mathcal{E}}$. All nodes of an $\epsilon$-tree with regard to $\mathcal{E}$ are equivalent from the viewpoint of $\mathcal{E}$. Condition 1 requires that if one of $B_1, \ldots, B_k$ is in the $\epsilon$-tree then all of $B_1, \ldots, B_k$ are in the $\epsilon$-tree, and $B \xrightarrow{q} B'$ is $B \xrightarrow{p_i} B_i$ for some $i \in I$. This is nothing more than the intuition that

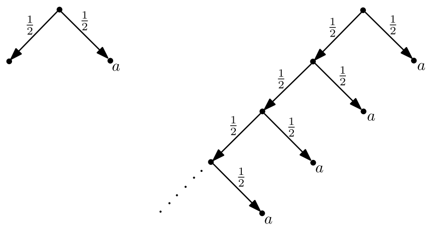
$B \xrightarrow{\coprod_{i\in[k]} p_i\tau} \coprod_{i\in[k]} B_i$ is conceptually a single silent transition. The number of $\epsilon$-trees of $A$ with regard to an equivalence class are in general infinite. Let's see some examples.

**Example 1.** Let $\Omega_a = \mu X.(\tau.a + \tau.X)$. Let $\mathcal{E}_1$ be any equivalence that distinguishes a divergent process from a non-divergent one. A finite $\epsilon$-tree of $\Omega_a$ with regard to $\mathcal{E}_1$ corresponds to a finite transition sequence of the form $\Omega_a \xrightarrow{\tau} \Omega_a \xrightarrow{\tau} \ldots \xrightarrow{\tau} \Omega_a$. In the non-random case an $\epsilon$-tree with regard to $\mathcal{E}_1$ is just an instance of $\Longrightarrow_{\mathcal{E}_1}$. There is an infinite $\epsilon$-tree of $\Omega_a$, corresponding to the divergent sequence $\Omega_a \xrightarrow{\tau} \Omega_a \xrightarrow{\tau} \ldots$.

**Example 2.** Let $\Omega_{\frac{1}{2}} = \mu X.(\frac{1}{2}\tau.X \oplus \frac{1}{2}\tau.X)$. There are infinitely many $\epsilon$-trees of $a \,|\, \Omega_{\frac{1}{2}}$ with regard to any equivalence. An $\epsilon$-tree may be a single node tree (the left diagram below), or a three node tree (the middle diagram below), or an infinite tree (the right diagram below). Unlike Example 1 the divergence in this case is immune from any intervention.

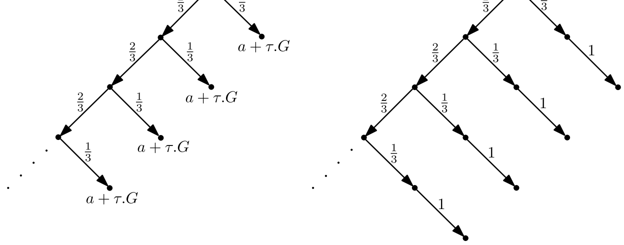


**Example 3.** Let $\Omega_{\frac{1}{2}a} = \mu X.(\frac{1}{2}\tau.a \oplus \frac{1}{2}\tau.X)$. Let $\mathcal{E}_2$ be an equivalence such that $[\Omega_{\frac{1}{2}a}]_{\mathcal{E}_2} = [a]_{\mathcal{E}_2}$. A finite $\epsilon$-tree of $\Omega_{\frac{1}{2}a}$ with regard to $\mathcal{E}_2$ is described by the left diagram below, one of its leaves cannot do an immediate $a$ action. The right diagram describes an infinite $\epsilon$-tree of $\Omega_{\frac{1}{2}a}$ with regard to $\mathcal{E}_2$, all of its leaves can do an immediate $a$ action.

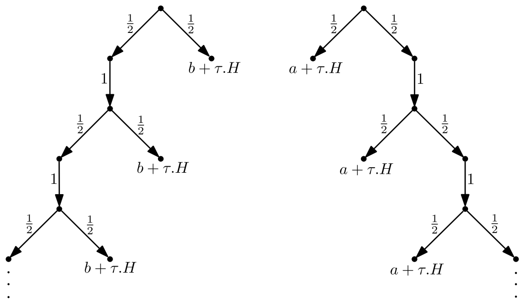


The process $\Omega_{\frac{1}{3}a} = \mu X.(\frac{1}{3}\tau.a \oplus \frac{2}{3}\tau.X)$ has similar $\epsilon$-trees.

**Example 4.** Let $G = \mu X.(\frac{1}{3}\tau.(a + \tau.X) \oplus \frac{2}{3}\tau.X)$. Let $\mathcal{E}_3$ be any equivalence such that $[G]_{\mathcal{E}_3} = [a + \tau.G]_{\mathcal{E}_3}$. Two $\epsilon$-trees of $G$ with regard to $\mathcal{E}_3$ are described by the following infinite diagrams. Every leaf of the left diagram can do an immediate $a$ action, whereas none of the leaves of the right diagram can do an immediate $a$ action.



**Example 5.** Let $H = \mu X.(\frac{1}{2}\tau.(a + \tau.X) \oplus \frac{1}{2}\tau.(b + \tau.X))$. An $\epsilon$-tree of $H$ with regard to an equivalence $\mathcal{E}_4$ rendering true $[H]_{\mathcal{E}_4} = [a + \tau.H]_{\mathcal{E}_4} = [b + \tau.H]_{\mathcal{E}_4}$ is described by the left diagram below. Every leaf of the $\epsilon$-tree can do an immediate $b$ action. Another $\epsilon$-tree of $H$ with regard to $\mathcal{E}_4$ is described by the right diagram below, in which every leaf can do an immediate $a$ action.




These examples bring out a few observations. Firstly $\epsilon$-trees are generalizations of $\Longrightarrow_{\mathcal{E}}$. This is clear from Example 1. However $\epsilon$-trees are a little too general. Two $\epsilon$-trees of a process may differ in that every leaf of one $\epsilon$-tree may do an immediate $a$ action whereas in the other this is not true.

To isolate the $\epsilon$-trees that truly correspond to $\Longrightarrow_{\mathcal{E}}$, we introduce some auxiliary definitions. A *path* in a silent tree $t$ is either a finite path going from the root to a node or an infinite path starting from the root. A *branch* of $t$ is either a path ending in a leaf or an infinite path. The length $|\pi|$ of a path $\pi$ is the number of edges in $\pi$ if $\pi$ is finite; it is $\omega$ otherwise. For $i \le |\pi|$ let $\pi(i)$ be the label of the $i$-th edge. The probability $\mathsf{P}(\pi)$ of a finite path $\pi$ is $\prod\{\pi(i) \mid i \in [|\pi|]\}$. A path of length zero is a single node, and its probability is 1. The probability of an infinite path $A \xrightarrow{p_1\tau} \xrightarrow{p_2\tau} \ldots \xrightarrow{p_k\tau} \ldots$ is the limit of $p_1, p_1p_2, \ldots, \prod_{i\le k} p_i, \ldots$, whose existence is guaranteed because the decreasing sequence is bounded by 0 from below. If $t$ is finite, define $\mathsf{P}(t) = \sum\{\mathsf{P}(\pi) \mid \pi$ is a branch of $t\}$. If $t$ is infinite, we need to define the probability in terms of approximation. Let $t\restriction_k$ be the subtree of $t$ defined by the nodes of height no more than $k$. Inductively

- $t\restriction_0$ is the one node tree defined by the root of $t$; and
- $t\restriction_{k+1}$ is defined by the nodes of $t\restriction_k$ and all the children of these nodes.

It should be clear that $\mathsf{P}(t\restriction_{k+1}) \le \mathsf{P}(t\restriction_k)$. The probability $\mathsf{P}(t)$ of the tree $t$ is defined by the limit $\lim_{k\to\infty} \mathsf{P}(t\restriction_k)$.

**Lemma 3.1.** *$P(t) = 1$ for every $\epsilon$-tree $t$.*

**Proof.** Now $\mathsf{P}(t\restriction_0) = 1$ by definition. If $\mathsf{P}(t\restriction_k) = 1$, then $\mathsf{P}(t\restriction_{k+1}) = 1$ by Definition 4. Thus $\mathsf{P}(t) = \lim_{k\to\infty} \mathsf{P}(t\restriction_k) = 1$. □

The probability of the finite branches of $t$ is defined by $\mathsf{P}^f(t) = \lim_{k\to\infty} \mathsf{P}^k(t)$, where

$$\mathsf{P}^k(t) = \sum\{\mathsf{P}(\pi) \mid \pi \text{ is a finite branch in } t \text{ such that } |\pi| \le k\}. \tag{8}$$

We are now in a position to generalize a branching bisimulation for CCS processes to a branching bisimulation for RCCS processes. First of all we generalize state-preserving silent transition sequences of finite length. Intuitively such a *finite* sequence turns into an $\epsilon$-tree that probabilistically contains no infinite branches.

**Definition 5.** An $\epsilon$-tree $t^A_{\mathcal{E}}$ is *regular* if $\mathsf{P}^f(t^A_{\mathcal{E}}) = 1$.

In the same line of thinking an $\epsilon$-tree is divergent if it does not have any finite branches.

**Definition 6.** An $\epsilon$-tree $t^A_{\mathcal{E}}$ is *divergent* if $\mathsf{P}^f(t^A_{\mathcal{E}}) = 0$.

Notice that an $\epsilon$-tree $t^A_{\mathcal{E}}$ refers to an equivalence $\mathcal{E}$. Definition 6 is consistent with the $\mathcal{E}$-divergence introduced in Section 2. The next definition is the probabilistic counterpart of Definition 2.

**Definition 7.** An equivalence $\mathcal{E}$ on $\mathcal{P}_{\mathrm{RCCS}}$ is *divergence-sensitive* if the following is valid for every $\mathcal{C} \in \mathcal{P}/\mathcal{E}$.

- Either all members of $\mathcal{C}$ have divergent $\epsilon$-trees with regard to $\mathcal{E}$, or no member of $\mathcal{C}$ has any divergent $\epsilon$-tree with regard to $\mathcal{E}$.

To discuss the branching bisimulation for random processes, we need to talk about a transition from a process $A$ to an equivalence class $\mathcal{B} \in \mathcal{P}/\mathcal{E}$. This makes sense because the processes in $\mathcal{B}$ are supposed to be all equivalent. We would like to formalize the idea that after a finite number of state-preserving silent transitions an $\ell$-action is performed and the end processes are in $\mathcal{B}$. Suppose $\left(\ell \in Chan \cup \overline{Chan}\right) \vee (\ell = \tau \wedge \mathcal{B} \ne [A]_{\mathcal{E}})$. An *$\ell$-transition from $A$ to $\mathcal{B}$ with regard to $\mathcal{E}$* consists of a regular $\epsilon$-tree $t^A_{\mathcal{E}}$ of $A$ with regard to $\mathcal{E}$ and a transition $L \xrightarrow{\ell} L' \in \mathcal{B}$ for every leaf $L$ of $t^A_{\mathcal{E}}$. We will write $A \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{B}$ if there is an $\ell$-transition from $A$ to $\mathcal{B}$ with regard to $\mathcal{E}$. By definition $A \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{B}$ whenever $A \xrightarrow{\ell} B \in \mathcal{B}$.

Let's see some examples. For the process $\Omega_{\frac{1}{2}a}$ in Example 3 one has $\Omega_{\frac{1}{2}a} \rightsquigarrow_{\mathcal{E}_2} \xrightarrow{a} \mathbf{0}$, where the regular $\epsilon$-tree is described by the right diagram in Example 3. For the process $G$ in Example 4 one has $G \rightsquigarrow_{\mathcal{E}_3} \xrightarrow{a} \mathbf{0}$, where the regular $\epsilon$-tree is described by the left diagram in Example 4. For the process $H$ in Example 5, $H \rightsquigarrow_{\mathcal{E}_4} \xrightarrow{a} \mathbf{0}$ via the regular $\epsilon$-tree described by the right diagram, and $H \rightsquigarrow_{\mathcal{E}_4} \xrightarrow{b} \mathbf{0}$ via the regular $\epsilon$-tree described by the left diagram.

Now consider the situation where $A$ evolves into processes in $\mathcal{B} \in (\mathcal{P}/\mathcal{E}) \setminus \{[A]_{\mathcal{E}}\}$ with probability greater than 0. Suppose $L$ is a leaf of $t^A_{\mathcal{E}}$ and $L \xrightarrow{\coprod_{i\in[k]} p_i\tau} \coprod_{i\in[k]} L_i$ such that $L_i \in \mathcal{B}$ for some $i \in [k]$. Define

$$\mathsf{P}\left(L \xrightarrow{\coprod_{i\in[k]} p_i\tau} \mathcal{B}\right) = \sum_{i\in[k]} \left\{p_i \mid L \xrightarrow{p_i\tau} L_i \in \mathcal{B}\right\}.$$

Define the *normalized* probability

$$\mathsf{P}_{\mathcal{E}}\left(L \xrightarrow{\coprod_{i\in[k]} p_i\tau} \mathcal{B}\right) = \mathsf{P}\left(L \xrightarrow{\coprod_{i\in[k]} p_i\tau} \mathcal{B}\right) \Big/ \left(1 - \mathsf{P}\left(L \xrightarrow{\coprod_{i\in[k]} p_i\tau} [A]_{\mathcal{E}}\right)\right).$$

Intuitively the normalized probability is the probability that $L$ may leave the class $[A]_{\mathcal{E}}$ silently for elements of $\mathcal{B}$. If one leaf of the regular $t^A_{\mathcal{E}}$ can do a silent transition that leaves $t^A_{\mathcal{E}}$ with a non-zero probability, we require that every leaf of $t^A_{\mathcal{E}}$ is capable of doing a silent transition that leaves $t^A_{\mathcal{E}}$ with that probability. This probabilistic bisimulation property is observed in [8] in the simpler setting of the finite state fully probabilistic processes. In our general setting a process may do several random silent transitions caused by different random operators. Suppose $\mathcal{B} \neq [A]_{\mathcal{E}}$. A *$q$-silent transition from $A$ to $\mathcal{B}$ with regard to $\mathcal{E}$* consists of a regular $\epsilon$-tree $t^A_{\mathcal{E}}$ of $A$ with regard to $\mathcal{E}$ and, for every leaf $L$ of $t^A_{\mathcal{E}}$, a collective silent transition $L \xrightarrow{\coprod_{i\in[k]} p_i\tau} \coprod_{i\in[k]} L_i$ such that

$$\mathsf{P}_{\mathcal{E}}\left(L \xrightarrow{\coprod_{i\in[k]} p_i\tau} \mathcal{B}\right) = q.$$

We will write $A \rightsquigarrow_{\mathcal{E}} \xrightarrow{q} \mathcal{B}$ if there is a $q$-silent transition from $A$ to $\mathcal{B}$ with regard to $\mathcal{E}$.

**Definition 8.** An equivalence $\mathcal{E}$ on $\mathcal{P}_{\text{RCCS}}$ is a *branching bisimulation* if the following statements are valid.

1. If $B \,\mathcal{E}\, A \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C} \in \mathcal{P}/\mathcal{E}$ and $\left(\ell \in Chan \cup \overline{Chan}\right) \vee (\ell = \tau \wedge \mathcal{C} \neq [A]_{\mathcal{E}})$, then $B \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$.
2. If $B \,\mathcal{E}\, A \rightsquigarrow_{\mathcal{E}} \xrightarrow{q} \mathcal{C} \in \mathcal{P}/\mathcal{E}$ such that $\mathcal{C} \neq [A]_{\mathcal{E}}$, then $B \rightsquigarrow_{\mathcal{E}} \xrightarrow{q} \mathcal{C}$.

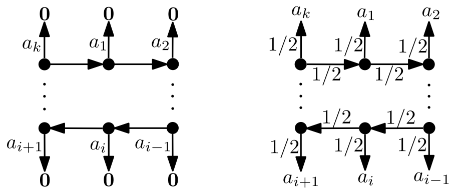
We say that $B \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$ bisimulates $A \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$ in statement 1, and that $B \rightsquigarrow_{\mathcal{E}} \xrightarrow{q} \mathcal{C}$ bisimulates $A \rightsquigarrow_{\mathcal{E}} \xrightarrow{q} \mathcal{C}$ in statement 2.

Let's take a look at an example that explains the subtlety of Definition 8. Define $P_r = r\tau.a \oplus (1-r)\tau.\mu X.(a+\tau.X)$, where $0 < r < 1$. Firstly notice that $a$ and $\mu X.(a+\tau.X)$ cannot be in any divergence-sensitive branching bisimulation because the latter is divergent whereas the former is not. It follows that $P_r$ and $a$ cannot be in any divergence-sensitive branching bisimulation because $P_r$ has the potential to diverge. It also follows that $P_r$ and $\mu X.(a+\tau.X)$ cannot be in any divergence-sensitive branching bisimulation because $P_r \xrightarrow{1-r} \mu X.(a+\tau.X)$ cannot be bisimulated by $\mu X.(a+\tau.X)$. We conclude that the only $\epsilon$-tree of $P_r$ is the trivial tree with one node. Therefore $P_r \rightsquigarrow_{\mathcal{E}} \xrightarrow{r} a$ is the same as $P_r \xrightarrow{r} a$ and $P_r \rightsquigarrow_{\mathcal{E}} \xrightarrow{1-r} \mu X.(a+\tau.X)$ is the same as $P_r \xrightarrow{1-r} \mu X.(a+\tau.X)$. It follows from the second clause of Definition 8 that $P_r$ and $P_{r'}$ cannot be in any divergence-sensitive branching bisimulation whenever $r \neq r'$.

Consider $\mu X.\,(a_1 + \tau.(a_2 + \tau.(\ldots(a_k + \tau.X)\ldots)))$. The behavior of the process can be pictured as a ring (the left diagram below), in which all nodes are equal [21]. Consider a probabilistic version of this process

$$\mu X.\left(\frac{1}{2}\tau.a_1 \oplus \frac{1}{2}\tau.(\frac{1}{2}\tau.a_2 \oplus \frac{1}{2}\tau.(\ldots(\frac{1}{2}\tau.a_k \oplus \frac{1}{2}\tau.X)\ldots))\right).$$

Its behavior is described by the right diagram below. No two nodes in the right ring can be in any branching bisimulation. For example the top middle node in the ring can reach to the process $a_1$ with probability 1/2, or more precisely "the top middle node $\rightsquigarrow_{\mathcal{E}} \xrightarrow{1/2} a_1$" for some equivalence $\mathcal{E}$, whereas it is impossible that "a bottom node $\rightsquigarrow_{\mathcal{E}} \xrightarrow{1/2} a_1$" for any equivalence $\mathcal{E}$.




The process $\Omega_a$ of Example 1 and the process $\Omega_{\frac{1}{2}a}$ of Example 3 cannot be in any divergence-sensitive branching bisimulation because the former is divergent whereas the latter is not.

For a relation $\mathcal{R}$ on $\mathcal{P}_{\text{RCCS}}$, let $\mathcal{R}^*$ be the *equivalence closure* of the relation $\mathcal{R}$. Clearly both $\left\{(\Omega_{\frac{1}{2}a}, a)\right\}^*$ and $\left\{(\Omega_{\frac{1}{3}a}, a)\right\}^*$ are divergence-sensitive bisimulation, where $\Omega_{\frac{1}{2}a}$ and $\Omega_{\frac{1}{3}a}$ are defined in Example 3. And $\{(G, G_a)\}^*$ is a divergence-sensitive branching bisimulation, where $G$ is defined in Example 4 and $G_a \overset{\text{def}}{=} a + \tau.G$. Also $\{(H, H_a), (H, H_b), (H, E)\}^*$ is a divergence-sensitive branching bisimulation, where $H$ is defined in Example 5, $H_a \overset{\text{def}}{=} a + \tau.H$, $H_b \overset{\text{def}}{=} b + \tau.H$ and $E \overset{\text{def}}{=} \mu X.(a + b + \tau.X)$.

## 4. Equality for random process

The following lemma follows immediately from definition.

**Lemma 4.1.** *If $\mathcal{E}_i$ is a divergence-sensitive equivalence for all $i \in I$, then so is $\left(\bigcup_{i\in I} \mathcal{E}_i\right)^*$.*

The proof of the next fact is slightly complicated but standard.

**Proposition 4.2.** *If $\{\mathcal{E}_i\}_{i\in I}$ is a set of branching bisimulation, so is $\left(\bigcup_{i\in I} \mathcal{E}_i\right)^*$.*

**Proof.** Let $\mathcal{E} = \left(\bigcup_{i\in I} \mathcal{E}_i\right)^*$. We only have to prove that for each $i \in I$ the following is valid whenever $A\mathcal{E}_i B$.

1. If $A \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C} \in \mathcal{P}/\mathcal{E}$ and $\left(\ell \in Chan \cup \overline{Chan}\right) \vee (\ell = \tau \wedge \mathcal{C} \neq [A]_{\mathcal{E}})$, then $B \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$.
2. If $A \rightsquigarrow_{\mathcal{E}} \xrightarrow{q} \mathcal{C} \in \mathcal{P}/\mathcal{E}$ for some $\mathcal{C} \neq [A]_{\mathcal{E}}$, then $B \rightsquigarrow_{\mathcal{E}} \xrightarrow{q} \mathcal{C}$.

We emphasize that the pair $(A, B)$ is in $\mathcal{E}_i$, while the above bisimulation property is stated with regards to $\mathcal{E}$. We prove statement 1. The proof of statement 2 is similar. Let $\ell$ be such that $\left(\ell \in Chan \cup \overline{Chan}\right) \vee (\ell = \tau \wedge \mathcal{C} \neq [A]_{\mathcal{E}})$. Consider an $\ell$-transition $A \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$. It consists of a regular $\epsilon$-tree $t_A$ of $A$ with regard to $\mathcal{E}$ and, for every leaf $L$ of $t_A$, a transition $L \xrightarrow{\ell} L' \in \mathcal{C}$. We construct by induction on the structure of $t_A$ an $\ell$-transition $B \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$. The basic idea is to construct an $\epsilon$-tree with regards to $\mathcal{E}$ for every edge of $t_A$ such that the $\epsilon$-tree bisimulates the edge. By sticking these $\epsilon$-trees together we get an $\epsilon$-tree $t_B$ of $B$ with regard to $\mathcal{E}$. We can make sure that $t_B$ is regular, which will become clear by the construction. Formally the bisimulation $B \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$ can be derived by induction.

- Suppose the root of $t_A$ has only one child $A'$. By definition $A \xrightarrow{\tau} A'$. If $A' \in [A]_{\mathcal{E}_i}$, we construct $t_B$ by structural induction on the $\epsilon$-tree of $A'$. If $A' \notin [A]_{\mathcal{E}_i}$ then $A \xrightarrow{\tau} A'$ is bisimulated by some $\tau$-transition $B \rightsquigarrow_{\mathcal{E}_i} \xrightarrow{\tau} [A']_{\mathcal{E}_i}$ consisting of a regular $\epsilon$-tree $t'_B$ of $B$ with regard to $\mathcal{E}_i$ and, for every leaf $B''$ of $t'_B$, a transition $B'' \xrightarrow{\tau} B'\mathcal{E}_i A'$ for some $B'$. Notice that $B''\mathcal{E}_i B \mathcal{E}_i A \mathcal{E} A' \mathcal{E}_i B'$. Thus $B''\mathcal{E}B'\mathcal{E}B$. We then continue to construct an $\epsilon$-tree for $B'$ by induction on the structure of the regular $\epsilon$-tree of $A'$.
- Suppose the root of $t_A$ has $h$ children $A^1, \ldots, A^h$ with the corresponding edges labeled by $p_1, \ldots, p_h$ respectively. By the definition of $\epsilon$-tree,

  $$A \xrightarrow{\coprod_{j\in[h]} p_j \tau} \coprod_{j\in[h]} A^j.$$

  There are two cases. In the first case $A^j \mathcal{E}_i A$ for all $j \in [k]$. We construct $t'_B$ by structural induction on the regular $\epsilon$-tree of say $A^1$. In the second case suppose without loss of generality that $A^1 \notin [A]_{\mathcal{E}_i}$. Let $q = \mathsf{P}_{\mathcal{E}_i}\left(A \xrightarrow{\coprod_{i\in[h]} p_i \tau} [A^1]_{\mathcal{E}_i}\right)$. Then $B \rightsquigarrow_{\mathcal{E}_i} \xrightarrow{q} [A^1]_{\mathcal{E}_i}$ by definition. The $q$-silent transition consists of a regular $\epsilon$-tree $t'_B$ of $B$ with regard to $\mathcal{E}_i$ and, for each leaf $B''$ of $t'_B$, a collective silent transition $B'' \xrightarrow{\coprod_{i'\in[h']} p_{i'} \tau} \coprod_{i'\in[h']} B_{i'}$ such that $\mathsf{P}_{\mathcal{E}_i}\left(B'' \xrightarrow{\coprod_{i'\in[h']} p_{i'} \tau} [A^1]_{\mathcal{E}_i}\right) = q$. For

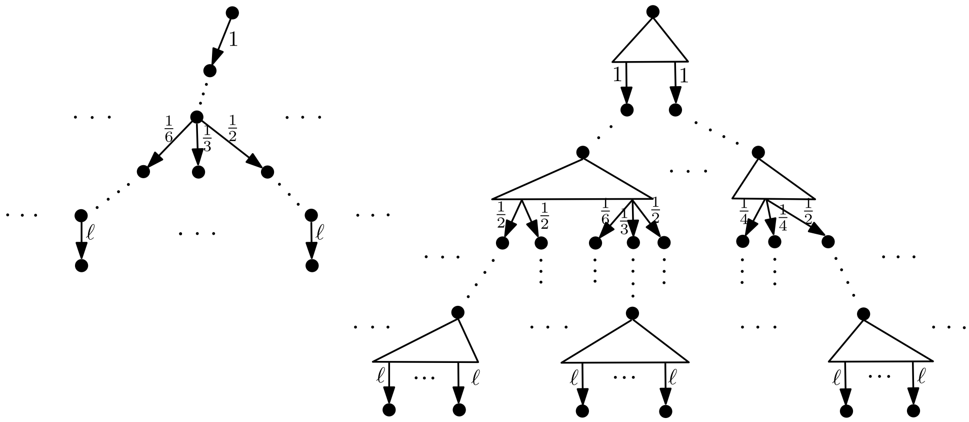



**Fig. 1.** Stepwise bisimulation.

every process $B_{i'}$ the $q$-silent transition $B \rightsquigarrow_{\mathcal{E}_i} \xrightarrow{q} [A^1]_{\mathcal{E}_i}$ reaches, we continue to construct an $\epsilon$-tree of $B_{i'}$ by structural induction on the regular $\epsilon$-tree of $A^1$. Now $B''\mathcal{E}_i B \mathcal{E}_i A \mathcal{E} A^1 \mathcal{E}_i B_{i'}$. So $B \mathcal{E} B'' \mathcal{E} B_{i'}$.

- Suppose the root of $t_A$ does the transition $A \xrightarrow{\ell} L'$. Then $B \rightsquigarrow_{\mathcal{E}_i} \xrightarrow{\ell} [L']_{\mathcal{E}_i}$ by definition. Therefore $B \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} [L']_{\mathcal{E}}$ witnessed by some regular $\epsilon$-tree of $B$ with regards to $\mathcal{E}$.

In Fig. 1 the left is a diagram for $A \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$, while the right is a diagram for the stepwise bisimulation $B \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$. The above itemized cases are described by the upper, middle, and bottom parts of the diagrams respectively. We still need to verify the regularity property. Given $\varepsilon \in (0,1)$, there is a number $K_\varepsilon$ such that $1 - \mathsf{P}^{K_\varepsilon}(t_A) < \varepsilon/2$. Now every edge in $t_A \lceil K_\varepsilon$ is bisimulated either vacuously or by an $\epsilon$-tree $t$. There is a number $N_\varepsilon$ such that for every such $\epsilon$-tree $t$ it holds that $1 - \mathsf{P}^{N_\varepsilon}(t) < \frac{\varepsilon}{2K_\varepsilon}$. It is not difficult to see that $1 - \mathsf{P}^{K_\varepsilon N_\varepsilon}(t_B) < \varepsilon/2 + \varepsilon/2 = \varepsilon$. Therefore $t_B$ is regular. So $A \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$ is bisimulated by $B \rightsquigarrow_{\mathcal{E}} \xrightarrow{\ell} \mathcal{C}$. By transitivity we conclude that $\mathcal{E}$ is a branching bisimulation. $\square$

Proposition 4.2 is reassuring. We may now define the *equality on RCCS processes*, denoted by $=_{\text{RCCS}}$, as the largest divergence-sensitive branching bisimulation on $\mathcal{P}_{\text{RCCS}}$. An obvious corollary of the above proposition is that $=_{\text{RCCS}}$ is an equivalence. We will abbreviate $=_{\text{RCCS}}$ to $\simeq$.

**Theorem 4.3.** *The equality $=_{\text{RCCS}}$ is a congruence.*

**Proof.** It is easy to see that $\simeq$ is closed under both the nondeterministic choice operation and the random choice operation. Now let $\mathcal{R}$ be the relation $\{(A \mid C, B \mid C) \mid A \simeq B \wedge C \in \mathcal{P}\}$. We prove that $\mathcal{R}^\circ \stackrel{\text{def}}{=} (\mathcal{R} \cup \simeq)^*$ is a divergence-sensitive branching bisimulation. Suppose $(A \mid C)\, \mathcal{R}\, (B \mid C)$ and $A \mid C \rightsquigarrow_{\mathcal{R}^\circ} \xrightarrow{\ell} \mathcal{C}$ for some equivalence class $\mathcal{C} \in \mathcal{P}/\mathcal{R}^\circ$ such that

$$\left(\ell \in Chan \cup \overline{Chan}\right) \vee (\ell = \tau \wedge \mathcal{C} \neq [A \mid C]_{\mathcal{R}^\circ}).$$

Let $t_{A|C}$ denote the regular $\epsilon$-tree of $A \mid C$ in the $\ell$-transition. Using the technique explained in the proof of Proposition 4.2 it is routine to build up an $\ell$-transition $B \mid C \rightsquigarrow_{\mathcal{R}^\circ} \xrightarrow{\ell} \mathcal{C}$ that bisimulates $A \mid C \rightsquigarrow_{\mathcal{R}^\circ} \xrightarrow{\ell} \mathcal{C}$. This is inductively described as follows.

- Suppose an edge from $A \mid C$ to $A' \mid C$ labeled 1 is caused by a transition $A \xrightarrow{\tau} A' \simeq A$. In this case $A' \mid C \,\mathcal{R}\, B \mid C$. If it is caused by $A \xrightarrow{\coprod_{i\in I} p_i \tau} \coprod_{i\in I} A_i$ such that $A_i \simeq A$ for all $i \in I$, then $A \mid C \xrightarrow{\coprod_{i\in I} p_i \tau} \coprod_{i\in I} A_i \mid C$ and obviously $A_i \mid C \,\mathcal{R}\, B \mid C$ for each $i \in I$. In neither case $B \mid C$ has to bisimulate anything.
- Suppose an edge from $A \mid C$ to $A' \mid C$ labeled 1 is caused by a transition $A \xrightarrow{\tau} A' \not\simeq A$ such that $A \mid C \simeq A' \mid C$. Then $B \rightsquigarrow_{\simeq} \xrightarrow{\tau} [A']_{\simeq}$. For every leaf $B''$ in the regular $\epsilon$-tree of $B$, $B'' \xrightarrow{\tau} B' \in [A']_{\simeq}$ for some $B'$. It should be clear that $B'' \mid C \,\mathcal{R}\, B \mid C \,\mathcal{R}\, A \mid C \simeq A' \mid C \,\mathcal{R}\, B' \mid C$. So $B \mid C$, $B'' \mid C$ and $B' \mid C$ are related by $\mathcal{R}^\circ$.
- Suppose an edge from $A \mid C$ to $A' \mid C'$ labeled 1 is caused by $A \xrightarrow{\alpha} A'$ and $C \xrightarrow{\overline{\alpha}} C'$ such that $A' \mid C' \simeq A \mid C$. This case is similar to the above case.
- Suppose $A \xrightarrow{\coprod_{i\in[k]} p_i \tau} \coprod_{i\in[k]} A_i$ and $A_1 \not\simeq A \not\simeq A_2 \not\simeq A_1$ and $A_1 \mid C \simeq A_2 \mid C \not\simeq A \mid C$. Define

$$q \stackrel{\text{def}}{=} q_1 + q_2,$$

$$q_1 \stackrel{\text{def}}{=} \mathsf{P}_{\simeq}\left(A \xrightarrow{\coprod_{i\in[k]} p_i\tau} [A_1]_{\simeq}\right),$$

$$q_2 \stackrel{\text{def}}{=} \mathsf{P}_{\simeq}\left(A \xrightarrow{\coprod_{i\in[k]} p_i\tau} [A_2]_{\simeq}\right).$$

Then $q = \mathsf{P}_{\simeq}\left(A \mid C \xrightarrow{\coprod_{i\in[k]} p_i\tau} [A_1 \mid C]_{\simeq}\right)$. By assumption $B \rightsquigarrow_{\simeq}\xrightarrow{q_1} [A_1]_{\simeq}$ and $B \rightsquigarrow_{\simeq}\xrightarrow{q_2} [A_2]_{\simeq}$. It follows that $B \mid C \rightsquigarrow_{\mathcal{R}}\xrightarrow{q}$ $[A_1 \mid C]_{\mathcal{R}}$.

- Suppose $A \mid C \xrightarrow{\ell} \mathcal{C}$. One can show that $B \mid C \rightsquigarrow_{\mathcal{R}^\circ}\xrightarrow{\ell} \mathcal{C}$ by similar argument.

We conclude that $\mathcal{R}^\circ$ is a branching bisimulation. The proof that $\mathcal{R}^\circ$ is divergence-sensitive is simpler. Using almost the same proof one can show that $A \simeq B$ implies $C \mid A \simeq C \mid B$.

Next we argue that $\simeq$ is closed under localization. Define

$$\mathcal{S} \stackrel{\text{def}}{=} \{((a)A, (a)B) \mid A \simeq B\}.$$

We show that $\mathcal{S}^\circ \stackrel{\text{def}}{=} (\mathcal{S} \cup \simeq)^*$ is a divergence-sensitive bisimulation. Suppose $(a)A\,\mathcal{S}\,(a)B$ and that $t_{(a)A}$ is a regular $\epsilon$-tree of $(a)A$. This $\epsilon$-tree is derived from a silent tree of $A$. In the silent tree of $A$ an edge say $A' \xrightarrow{\tau} A''$ may not be state-preserving, even though $(a)A' \xrightarrow{\tau} (a)A''$ is state-preserving. Suppose $B' \simeq A'$ and $A' \xrightarrow{\tau} A''$ is bisimulated by $B' \rightsquigarrow_{\simeq}\xrightarrow{\tau} [A'']_{\simeq}$. It is easily seen that $(a)B' \rightsquigarrow_{\mathcal{S}^\circ}\xrightarrow{\tau} [(a)A'']_{\mathcal{S}^\circ}$ bisimulates $(a)A' \xrightarrow{\tau} (a)A''$. Arguing in this manner and using induction we show that if $\ell \neq \tau \vee \mathcal{C} \neq [(a)A]_{\mathcal{S}^\circ}$, then $(a)A \rightsquigarrow_{\mathcal{S}^\circ}\xrightarrow{\ell} \mathcal{C}$ is bisimulated by some $(a)B \rightsquigarrow_{\mathcal{S}^\circ}\xrightarrow{\ell} \mathcal{C}$. The divergence-sensitive property is easy.

Suppose $S_n, T_n$ are RCCS terms that contain $n$ free variables $X_1, \ldots, X_n$. Define $S_n \simeq T_n$ if for all RCCS terms $P_1, \ldots, P_n$ one has that

$$S_n\{P_1/X_1, \ldots, P_n/X_n\} \simeq T_n\{P_1/X_1, \ldots, P_n/X_n\}.$$

According to this definition $S_n \simeq T_n$ implies $S_n\{P_1/X_1\} \simeq T_n\{P_1/X_1\}$. Now suppose $S, T$ contain a free variable $X$, and $S \simeq T$. We would like to prove that $\mu X.S \simeq \mu X.T$. For clarity we shall abbreviate for example $R\{\mu X.S/X\}$ to $R[\mu X.S]$. Fix $S$ and $T$. Without loss of generality we assume that $S, T$ contain one and only one variable. Let $\mathcal{T}$ be

$$\{(R[\mu X.S], R[\mu X.T]) \mid R \text{ contains one and only one variable}\}.$$

To prove that $\mathcal{T}$ is a subset of $\simeq$, we need to make use of the following facts.

1. $R[R_0] \simeq R'[R_1]$ whenever $R \simeq R'$ and $R_0 \simeq R_1$.
2. If $R, R'$ contain one variable $X$, then $R \simeq R'$ if and only if $R[a] \simeq R'[a]$ for some fresh channel $a$ that does not appear in $R \mid R'$.

Fact 1 is proved similarly. Fact 2 is valid if we can prove that the following relation, denoted by $\mathcal{R}$, is contained in $\simeq$.

$$\left\{(R[P], R'[P]) \mid X \text{ is free in } R, R',\ R[a] \simeq R'[a] \text{ for a fresh } a, \text{ and } P \in \mathcal{P}\right\}.$$

We show that $\mathcal{R}^\circ \stackrel{\text{def}}{=} (\mathcal{R} \cup \simeq)^*$ is a divergence-sensitive branching bisimulation. Consider the typical situation $(R[P], R'[P]) \in \mathcal{R}$. Suppose $s$ is a regular $\epsilon$-tree for $R[P]$ with regards to $\mathcal{R}^\circ$. Let $s'$ be obtained from $s$ by replacing by $a$ every occurrence of $P$ in the labels of $s$. Notice that a subtree of $s$ whose root is labeled $P$ becomes in $s'$ a leaf labeled $a$. Using the method described in the above, we construct a silent tree $t'$ for $R'[a]$ with regards to $(\mathcal{R} \cup \simeq)^*$ that bisimulates the tree $s'$. Now obtain a silent tree $t''$ for $R'[P]$ by substituting $P$ for $a$ in every leaf of $t'$. For each leaf in $t''$ labeled $P$ replicate the $\epsilon$-tree of the corresponding node labeled $P$ in $s$. Let $t$ denote the resulting silent tree. One can check that $t$ is a regular $\epsilon$-tree for $R'[P]$ with regards to $\mathcal{R}^\circ$ that bisimulates $s$. Using this observation one can prove that $\mathcal{R}^\circ$ is a divergence-sensitive branching bisimulation. A simple consequence of Fact 2 is that, for fresh $a, b$, $R[a] \simeq R'[a]$ if and only if $R[b] \simeq R'[b]$.

Finally let $\mathcal{T}^\circ \stackrel{\text{def}}{=} (\mathcal{T} \cup \simeq)^*$. We argue that $\mathcal{T}^\circ$ is a divergence sensitive branching bisimulation. Suppose $(R[\mu X.S], R[\mu X.T]) \in \mathcal{T}$ and $s$ is a regular $\epsilon$-tree of $R[\mu X.S]$ with regards to $\mathcal{T}^\circ$. Let $a$ be a fresh channel. Let $s'$ be obtained from $s$ by replacing $\mu X.S$ in the labels of $s$ by $a$. An *a-tree* in $s'$ is a sub-$\epsilon$-tree of $s'$ that satisfies the following: (i) The root and all the leaves of the subtree are labeled by $a$; (ii) Apart from the root every internal node of the subtree is labeled by a process other than $a$. See the left diagram in Fig. 2. Now construct a tree $t'$ whose root is labeled $R[a]$ in the following manner:

1. Copy the part of the tree $s'$ whose edges are transitions due to $R$.

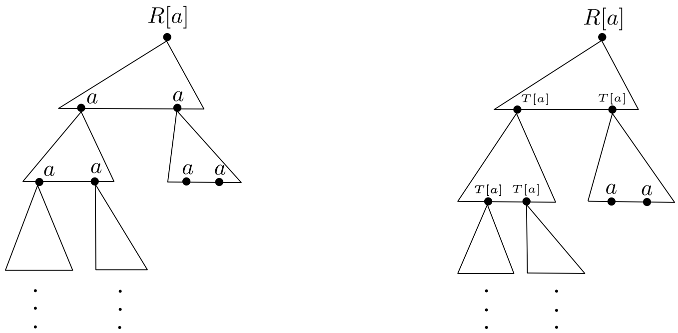



**Fig. 2.** The left is $s'$, the right is $t'$.

2. Let $t_R$ be the tree constructed in the above step. For each leaf of $t_R$ labeled $a$, if the corresponding node in $s'$, also labeled $a$, is an internal node, replace in $t'$ the label $a$ by $T[a]$. Then construct a regular $\epsilon$-tree of $T[a]$ with regards to $\simeq$ that bisimulates the $a$-tree of the corresponding node in $s'$, using the fact that $S[a] \simeq T[a]$. During this process for each internal node labeled $a$ in the constructed tree, repeat the construction just described. See the right diagram in Fig. 2.

Define $t$ to be the tree obtained from $t'$ by substituting $\mu X.T$ for $a$. It follows from Fact 2 that $t$ is an $\epsilon$-tree of $R[\mu X.T]$ with regards to $\mathcal{T}^\circ$ and that $t$ bisimulates $s$ with regards to $\mathcal{T}^\circ$. Using this fact it is not difficult to verify that $\mathcal{T}^\circ$ is a divergence-sensitive branching bisimulation. $\square$

Referring to Example 3 and Example 5 we see that $\Omega_{\frac{1}{2}a} \simeq \Omega_{\frac{1}{3}a}$ and that $\mu X.(\frac{1}{2}\tau.(a+\tau.X) \oplus \frac{1}{2}\tau.(b+\tau.X)) \simeq \mu X.(a+b+\tau.X)$.

## 5. Bisimulation theory justified

Let's address a question the reader might have already raised half way through reading the paper. Definition 5 and Definition 6 deal with two extreme situations to which Definition 8 refers. Instead of referring to $\epsilon$-trees with regards to an equivalence, can Definition 8 be given in terms of $\mathcal{E}$-trees? For an $\mathcal{E}$-tree $t_{\mathcal{E}}$ it is possible that $0 < \mathsf{P}^f(t_{\mathcal{E}}) = 1 - p < 1$. Let us call those trees $p$-divergent. Should $p$-divergence be bisimulated? We argue that the answer is negative. Here are the arguments.

- Firstly consider an RCCS process $D_0$ that contains no composition operators. Without loss of generality we may assume that $D_0$ is of the form $\mu X.E$ and that $E$ contains neither the localization operator nor the recursion operator. There is a constant $h$ defined by the syntax of $E$ such that for every infinite silent transition sequence

$$D_0 \xrightarrow{p_1\tau} D_1 \xrightarrow{p_2\tau} D_2 \xrightarrow{p_3\tau} \ldots \xrightarrow{p_i\tau} D_i \xrightarrow{p_{i+1}\tau} \ldots$$

  and any $i \geq 0$ some $j \in \{i+1, \ldots, j+h\}$ exists such that $D_j \simeq D_0$. Assume that $D_0$ does not have any divergent $\epsilon$-tree with regard to $\simeq$. Let $t^{D_0}$ be the least $\simeq$-tree of $D_0$ that satisfies the followings.
  - If a node $D'$ in $t^{D_0}$ has one child $D''$ in $t^{D_0}$, then $D' \xrightarrow{\tau} D''$.
  - If a node $D'$ in $t^{D_0}$ has $k$ children $D^1, \ldots, D^k$ in $t^{D_0}$, then $D' \xrightarrow{\coprod_{i\in K} p_i\tau} \coprod_{i\in K} D'_i$ for some $\{D'_i\}_{i\in K}$ such that $\{D^1, \ldots, D^k\} = \bigcup_{i\in K}\{D'_i \mid D'_i \simeq D'\}$.

  Since $t^{D_0}$ is not divergent, $D_0 \xrightarrow{\tau}_{\simeq} D_0^1 \xrightarrow{\tau}_{\simeq} D_0^2 \ldots \xrightarrow{\tau}_{\simeq} D_0^g \xrightarrow{q\tau} \mathcal{D}$ for some $g < h$, $q \in (0,1)$ and $\mathcal{D} \neq [D_0]_{\simeq}$. It is then easy to see that $\mathsf{P}(t^{D_0}\lceil_h) \leq 1 - q$, and by induction $\mathsf{P}(t^{D_0}\lceil_{ih}) \leq (1-q)^i$. Let $t_\omega^{D_0}$ be the subtree of $t^{D_0}$ consisting of all the infinite branches of $t^{D_0}$. Clearly $\mathsf{P}(t_\omega^{D_0}) \leq \lim_{i\to\infty}(1-q)^i = 0$.
  More generally one may prove that for RCCS processes without the composition operator, there is no such thing as a $p$-divergent process for any $p \in (0,1)$. We remark that these processes are finite states. Suppose $P_0$ is a finite state process and that the processes it reaches are $P_1, \ldots, P_m$. By structural induction on the syntax of $P_0$ one may derive that there is a number $h$ such that every transition sequence from any of $P_0, P_1, \ldots, P_m$ with length $h$ must contain two occurrences of some $P_i$. Let $t$ be a *full* $\mathcal{E}$-tree of $P_0$, meaning that all process equivalent to $P_0$ and reachable from $P_0$ are in $t$. It is easy to see that there is some $q \in (0,1)$ such that $\mathsf{P}(t\lceil_{mh}) \leq 1 - q$. Moreover this inequality is valid for every subtree of $t$. We can then apply the above argument to derive a contradiction.
- Let $M = \mu X.\left(\frac{1}{3}\tau.X \oplus \frac{1}{3}\tau.a \oplus \frac{1}{3}\tau.b\right)$ and $N = \mu X.\left(\frac{1}{5}\tau.X \oplus \frac{2}{5}\tau.a \oplus \frac{2}{5}\tau.b\right)$. Now $M \simeq N$ for the same reason the two processes defined in Example 3 are equal. Both can do the $a$-action with probability 1/2 and the $b$-action with probability

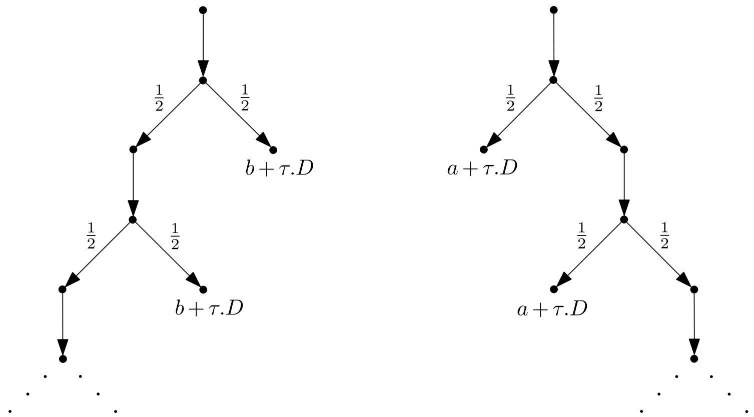



**Fig. 3.** Two $\epsilon$-trees for $D$.

1/2. Thus $\Omega_{\frac{1}{2}} \mid M \simeq \Omega_{\frac{1}{2}} \mid N$, where $\Omega_{\frac{1}{2}}$ is defined in Example 2. There is a $\frac{1}{3}$-divergent $\epsilon$-tree of $\Omega_{\frac{1}{2}} \mid M$ induced by the transition $\Omega_{\frac{1}{2}} \mid M \xrightarrow{\frac{1}{3}\tau} \Omega_{\frac{1}{2}} \mid M$ and the divergent $\epsilon$-tree of $\Omega_{\frac{1}{2}}$. Similarly there is a $\frac{1}{5}$-divergent $\epsilon$-tree of $\Omega_{\frac{1}{2}} \mid N$. The $\frac{1}{3}$-divergence cannot be bisimulated by $\Omega_{\frac{1}{2}} \mid N$, and the $\frac{1}{5}$-divergence cannot be bisimulated by $\Omega_{\frac{1}{2}} \mid M$. Strengthening Definition 8 with the requirement that $p$-divergence should be bisimulated would give rise to a relation that is not a congruence. We conclude that bisimulation of $p$-divergence is not desirable in the presence of the composition operator.

The dichotomy between the regular $\epsilon$-trees and the divergent $\epsilon$-trees now appears natural.

## 6. Comment

The regular $\epsilon$-tree based bisimulation is conceptually simpler than the distribution based bisimulation. The requirements imposed in Definition 8 are standard. The formulation in terms of $\epsilon$-trees makes evident the relationship between Definition 8 and the corresponding non-probabilistic definition. The proofs of Proposition 4.2 and Theorem 4.3 draw a great deal of resemblance to the similar proofs in the classical setting, and for that reason are easy to handle. To see the advantage of our approach, let's see one example from [13]. Suppose $D$ denotes the process $\mu X.\left(c+\tau.\left(\frac{1}{2}\tau.(a+\tau.X) \oplus \frac{1}{2}\tau.(b+\tau.X)\right)\right)$. It reminds one of the process $H$ defined in Example 5. According to the branching bisimilarity defined in [13], $D$ is equivalent to neither $a+\tau.D$ nor $b+\tau.D$. This is counter intuitive. Like $D$, both $a+\tau.D$ and $b+\tau.D$ retain the capacity to do $a$-action ($b$-action, $c$-action) with probability one. No observer can detect any qualitative and quantitative difference among the three processes. By the branching bisimilarity of this paper, one easily sees that

$$D \simeq \tau.\left(\frac{1}{2}\tau.(a+\tau.D) \oplus \frac{1}{2}\tau.(b+\tau.D)\right) \simeq a+\tau.D \simeq b+\tau.D.$$

In Fig. 3 two regular $\epsilon$-trees are given. By the left diagram $D \rightsquigarrow_{\simeq} \xrightarrow{b} \mathbf{0}$, and by the right diagram $D \rightsquigarrow_{\simeq} \xrightarrow{a} \mathbf{0}$. Consequently $D \simeq \mu X.(\tau.X+a+b+c)$. The reason for the well behavior of our branching bisimilarity is that either all the silent transitions in a collective silent transition, see (6), appear in an $\epsilon$-tree or none appear in the $\epsilon$-tree. It differs from the state-based approach in which state-preserving silent transition sequences are considered. It also differs from the distribution-based approach in which the distribution of a composition process, no matter how it is defined, is different from the one defined by collective silent transitions.

We have proposed a model independent approach, at both operational level and conceptual level, that turns a process model $\mathbb{M}$ into a randomized extension of $\mathbb{M}$. We have demonstrated how to build up the bisimulation semantics of the randomized $\mathbb{M}$ on the bisimulation semantics of $\mathbb{M}$. In our approach the bisimulation equality of the randomized $\mathbb{M}$ is a conservative extension of that of $\mathbb{M}$. This is because $\epsilon$-trees of $A$ with regard to an equivalence $\mathcal{E}$ are the same as $A \Longrightarrow_{\mathcal{E}}$ if $A$ is a process in $\mathbb{M}$. Therefore $A =_{\mathrm{CCS}} B$ if and only if $A =_{\mathrm{RCCS}} B$ for all $A, B \in \mathcal{P}_{\mathrm{CCS}}$.

The philosophy of the model independent method is that randomization is a computational property. An external action cannot really be random because it depends on an open-ended environment. An external action may appear random as a consequence of computational randomness. Random computation is the reason; random interaction is a consequence. There is another model independent approach using coalgebra [38]. The algebraic approach has been applied to give unifying semantics to probabilistic models in different application scenarios [25]. In comparison the method advocated in this paper is more operational.

The model independent approach can be studied from the perspective of axiomatization [23,26,7,39,9,15], equivalence checking [8,33], logical characterization [37], other equivalences say testing equivalence [16,24]. In the light of previous works on these topics in the probabilistic setting, results along these lines of investigation could be expected.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgement

We thank NSFC (61772336, 62072299) for financial support and Yuxin Deng and the members of BASICS for discussions. We also thank the two anonymous reviewers for insightful comments and suggestions for improvement.

## References

[1] S. Andova, S. Georgievska, N. Trcka, Branching bisimulation congruence for probabilistic systems, Theor. Comput. Sci. 413 (2012).
[2] S. Andova, A. Willemse, Branching bisimulation for probabilistic systems: characteristics and decidability, Theor. Comput. Sci. 356 (2006) 325–355.
[3] S. Arora, C. Lund, R. Motwani, M. Sudan, M. Szegedy, Proof verification and the hardness of approximation problems, in: FOCS'92, J. ACM (1998).
[4] L. Babai, Trading group theory for randomness, in: STOC'85, ACM, 1985.
[5] L. Babai, L. Fortnow, L. Lund, Nondeterministic exponential time has two prover interactive protocols, in: FOCS'90, Comput. Complex. (1991).
[6] T. Basten, Branching bisimilarity is an equivalence indeed, Inf. Process. Lett. 58 (1996) 141–147.
[7] J. Baeten, J. Bergstra, S. Smolka, Axiomatizing probabilistic processes: ACP with generative probabilities, Inf. Comput. 122 (1995) 234–255.
[8] C. Baier, H. Hermanns, Weak bisimulation for fully probabilistic processes, in: CAV'97, in: Lecture Notes in Computer Science, vol. 1254, Springer, 1997, pp. 119–130.
[9] E. Bandini, R. Segala, Axiomatization for probabilistic bisimulation, in: ICALP'2001, in: Lecture Notes in Computer Science, vol. 2076, Springer, 2001, pp. 370–381.
[10] M. Ben-Or, S. Goldwasser, J. Kilian, A. Wigderson, Multi-prover interactive proofs: how to remove intractability assumptions, in: STOC'88, ACM, 1988.
[11] O. Burkart, D. Caucal, F. Moller, B. Steffen, Verification on infinite structures, in: Handbook of Process Algebra, Elsevier Science, 2001.
[12] V. Castiglioni, S. Tini, Probabilistic divide and congruence: branching bisimilarity, Theor. Comput. Sci. 802 (2020).
[13] V. Castiglioni, S. Tini, Raiders of the lost equivalence: probabilistic branching bisimilarity, Inf. Process. Lett. 159–160 (2020).
[14] Y. Deng, Semantics of Probabilistic Processes: An Operational Approach, Springer-Verlag and Shanghai Jiao Tong University Press, 2015.
[15] Y. Deng, C. Palamidessi, Axiomatizations for probabilistic finite-state behaviors, Theor. Comput. Sci. 373 (1–2) (2007) 92–114.
[16] R. De Nicola, M. Hennessy, Testing equivalence for processes, Theor. Comput. Sci. 34 (1984) 83–133.
[17] L. Fortnow, J. Rompel, M. Sipser, On the power of multi-prover interactive protocols, Theor. Comput. Sci. 21 (1994) 545–557.
[18] Y. Fu, Theory of interaction, Theor. Comput. Sci. 611 (2016) 1–49.
[19] R. van Glabbeek, S. Smolka, B. Steffen, Reactive, generative, and stratified models of probabilistic processes, Inf. Comput. 3 (1995) 59–80.
[20] R. van Glabbeek, W. Weijland, Branching time and abstraction in bisimulation semantics, in: Information Processing'89, North-Holland, 1989, pp. 613–618.
[21] R. van Glabbeek, W. Weijland, Branching time and abstraction in bisimulation semantics, J. ACM 3 (1996) 555–600.
[22] S. Goldwasser, S. Micali, C. Rackoff, The knowledge complexity of interactive proofs, in: STOC'85, ACM, 1985.
[23] H. Hansson, B. Jonsson, A framework for reasoning about time and reliability, in: IEEE Symposium on Real-Time Systems, IEEE, 1989.
[24] M. Hennessy, An Algebraic Theory of Processes, MIT Press, Cambridge, MA, 1988.
[25] H. Hermanns, J. Krčál, J. Křetínský, Probabilistic bisimulation: naturally on distributions, in: CONCUR'14, in: Lecture Notes in Computer Science, vol. 8704, Springer, 2014, pp. 249–265.
[26] C. Jou, S. Smolka, Equivalences and complete axiomatizations for probabilistic processes, in: CONCUR'90, in: Lecture Notes in Computer Science, vol. 458, 1990, pp. 367–383.
[27] R. Lanotte, M. Merro, S. Tini, Compositional weak metrics for group key update, in: MFCS'17, 2017, 72.
[28] K. Larsen, A. Skou, Bisimulation through probabilistic testing, in: POPL'89, ACM, 1989, pp. 344–352.
[29] M. Lee, E. de Vink, Rooted branching bisimulation as a congruence for probabilistic transition systems, in: Proceedings Thirteenth Workshop on Quantitative Aspects of Programming Languages and Systems, London, UK, vol. 344–352, 2015, pp. 79–94.
[30] R. Milner, Communication and Concurrency, Prentice Hall, 1989.
[31] R. De Nicola, F. Vaandrager, Three logics for branching bisimulation, J. ACM (1995) 458–487.
[32] D. Park, Concurrency and automata on infinite sequences, in: TCS'81, in: Lecture Notes in Computer Science, vol. 104, Springer, 1981, pp. 167–183.
[33] A. Philippou, I. Lee, O. Sokolsky, Weak bisimulation for probabilistic systems, in: CONCUR'00, in: Lecture Notes in Computer Science, vol. 1877, Springer, 2000, pp. 334–349.
[34] L. Priese, On the concept of simulation in asynchronous, concurrent systems, Prog. Cybern. Syst. Res. 7 (1978) 85–92.
[35] R. Segala, Modelling and Verification of Randomized Distributed Rela-Time Systems, PhD Thesis, MIT, Dept. of EECS, 1995.
[36] R. Segala, Probability and nondeterminism in operational models of concurrency, in: CONCUR'06, in: Lecture Notes in Computer Science, vol. 4137, Springer, 2006, pp. 64–78.
[37] R. Segala, N. Lynch, Probabilistic simulations for probabilistic processes, in: CONCUR'94, in: Lecture Notes in Computer Science, vol. 836, Springer, 1994, pp. 481–496.
[38] A. Sokolova, Probabilistic systems coalgebraically: a survey, Theor. Comput. Sci. 412 (2011) 5095–5110.
[39] E. Stark, S. Smolka, A complete axiom system for finite-state probabilistic processes, in: Language and Interaction: Essays in Honour of Robin Milner, 1999.
[40] A. Turrini, H. Hermannss, Polynomial time decision algorithms for probabilistic automata, Inf. Comput. 244 (2015).
[41] M. Timmer, J. Katoen, J. van de Pol, M. Stoelinga, Confluence reduction for Markov automata, Theor. Comput. Sci. 655 (2016).
[42] S. Vadhan, Pseudorandomness, Foundations and Trends in Theoretical Computer Science, vol. 7, Now Publishers Inc., 2012.
[43] M. Vardi, Automatic verification of probabilistic concurrent finite-state programs, in: FOCS'85, IEEE, 1985, pp. 327–338.
[44] Y. Wang, K. Larsen, Testing probabilistic and nondeterministic processes, in: Protocol Specification, Testing and Verification XII, 1992, pp. 47–61.