The Discriminating Power of the Let-in Operator in the Lazy Call-by-Name Probabilistic λ-Calculus

Simona Kašterović
Faculty of Technical Sciences,University of Novi Sad
Trg Dositeja Obradovića 6, 21000 Novi Sad, Serbia
http://imft.ftn.uns.ac.rs/~simona/
simona.k@uns.ac.rs

Michele Pagani
IRIF, University Paris Diderot - Paris 7, France
https://www.irif.fr/~michele/
pagani@irif.fr

Abstract

We consider the notion of probabilistic applicative bisimilarity (PAB), recently introduced as a behavioural equivalence over a probabilistic extension of the untyped λ-calculus. Alberti, Dal Lago and Sangiorgi have shown that PAB is not fully abstract with respect to the context equivalence induced by the lazy call-by-name evaluation strategy. We prove that extending this calculus with a let-in operator allows for achieving the full abstraction. In particular, we recall Larsen and Skou's testing language, which is known to correspond with PAB, and we prove that every test is representable by a context of our calculus.

2012 ACM Subject Classification Mathematics of computing → Lambda calculus; Theory of computation → Lambda calculus

Keywords and phrases Probabilistic lambda calculus, Bisimulation, Howe's technique, Context equivalence, Testing

Digital Object Identifier 10.4230/LIPIcs.FSCD.2019.XXX

1 Introduction

We consider the probabilistic extension $\Lambda_\oplus$ of the untyped λ-calculus, obtained by adding a probabilistic choice primitive $M \oplus N$ representing a term evaluating to M or N with equal probability. This calculus provides a useful although quite simple framework for importing tools and results from the standard theory of the λ-calculus to probabilistic programming.

As well-known, the choice of an evaluation strategy for $\Lambda_\oplus$ plays a crucial role, even for strongly normalising terms. Consider a function $\lambda x.F$ applied to a probabilistic term $M \oplus N$: if we adopt a call-by-name policy, cbn by short, the whole term $M \oplus N$ would be passed to the calling parameter x *before* actually performing the choice between M and N, while in a call-by-value strategy, cbv by short, we first chose between M and N and *then* pass the value associated with this choice to x. If the evaluation of F calls n times the parameter x, then the cbn strategy performs n independent choices between M and N, while the cbv strategy copies n times the result of one single choice. In linear logic semantics [12], this phenomenon can be described by precising that the application is a bilinear function in cbv (so $(\lambda x.F)(M \oplus N)$ is equivalent to $((\lambda x.F)M) \oplus ((\lambda x.F)N)$), while it is not linear in the argument position in cbn (see discussion at Example 3).

In probabilistic programming it is worthwhile to have a cbv operator even in a cbn language, as the most of the randomised algorithms need to sample from a distribution and passing to a sub-procedure the *value* of this sample rather than the *whole distribution*. Consider for example the randomised quicksort: this algorithm takes a pivot randomly from an array and it passes it to the partitioning procedure, which uses this pivot several

4th International Conference on Formal Structures for Computation and Deduction (FSCD 2019).
Editor: Chair of FSCD 2019; Article No. XXX; pp. XXX:1–XXX:29
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

times. The algorithm would be unsound if we allow to make different choices each time the partitioning procedure calls for the same pivot. In [10] the authors enrich the cbn probabilistic PCF with a let-in operator, restricted to the ground values, so that let $x = M \oplus N$ in F behaves like a cbv application of $\lambda x.F$ to $M \oplus N$. In a continuous framework this kind of operator is usually called *sampling* (e.g. [17]), but this is just a different terminology for the same computation mechanism: sampling a value from a distribution before passing it to a parameter.

Both calling policies (cbn and cbv) can be declined with a further attribute which is Abramsky's lazyness [1]: a reduction strategy is *lazy* (sometimes called also *weak*) whenever it does not evaluate the body of a function, i.e. it does not reduce a β-redex under the scope of a λ-abstraction. This notion has been presented in order to provide a formal model of the evaluation mechanism of the lazy functional programming languages.

Two probabilistic programs are context equivalent if they have the same probability of converging to a value in all contexts. Of course, this notion depends on which reduction strategy has been chosen. The prototypical example of diverging term $\mathbf{\Omega} \stackrel{\text{def}}{=} (\lambda x.xx)(\lambda x.xx)$ is context equivalent with $\lambda x.\mathbf{\Omega}$ for a non-lazy strategy, while the two terms can be trivially distinguished by a lazy strategy as $\lambda x.\mathbf{\Omega}$ is a value for such a reduction. Similarly, the term $(\lambda xy.y)\mathbf{\Omega}$ is equivalent to $\mathbf{\Omega}$ for cbv, but it is converging for the cbn policy (lazy or not), because the reduction step $(\lambda xy.y)\mathbf{\Omega} \to \lambda y.y$ is admitted.

One of the major contribution of the already mentioned [1] has been to use the notion of bisimilarity in order to study the context equivalence of the lazy cbn λ-calculus. The idea is to consider a reduction strategy as a labelled transition system where the states and labels of the system are the λ-terms and a transition labelled by a term P goes from a term M to a value M' whenever M' is the result of evaluating the application MP. The benefit of this setting is to be able to transport into λ-calculus the whole theory of bisimilarity (called in [1] *applicative bisimilarity*) and its associated coinduction reasoning, which is one of the main tools for comparing processes in concurrency theory. Basically, two terms M and N are applicative bisimilar whenever their applications MP and NP are applicative bisimilar for any argument P. Abramsky proved that applicative bisimilarity is sound with respect to lazy cbn context equivalence (i.e. the former implies the latter), but it is not fully abstract (there are context equivalent terms that are not bisimilar).

Abramsky's applicative bisimilarity has been recently lifted to $\Lambda_\oplus$ by Dal Lago and his co-authors [4, 7]. The transition system becomes now a Markov Chain (here Definition 12) on the the top of it one can define a notion of probabilistic applicative bisimilarity (PAB). The paper [7] considers a lazy cbn reduction strategy, while [4] focuses on the (lazy) cbv strategy. In both settings, PAB is proven sound with respect to the associated context equivalence, but, surprisingly, the cbv bismilarity is also fully abstract, while the lazy cbn is not. Our paper shows that adding the let-in operator mentioned before is enough for recovering the full abstraction even for the lazy cbn.

Let us discuss more in detail the problem with the lazy cbn operation semantics. The two terms $\lambda xy.(x \oplus y)$ and $(\lambda xy.x) \oplus (\lambda xy.y)$ are context equivalent but not bisimilar (Example 6). The difference is between a process giving a value *allowing* two choices and a process giving two values *after* a choice (see Figure 4 to have a pictorial representation of the two processes). The cbn contexts are not able to discriminate such a subtle difference while bisimilarity does (Examples 14 and 21). In [4] the authors show a cbv context discriminating a variant of these two terms and they conjecture that a kind of sequencing operator can recover the full abstraction for the lazy cbn : our paper proves this conjecture.

The result is not surprising if compared to [4], however let us stress the contrast with the

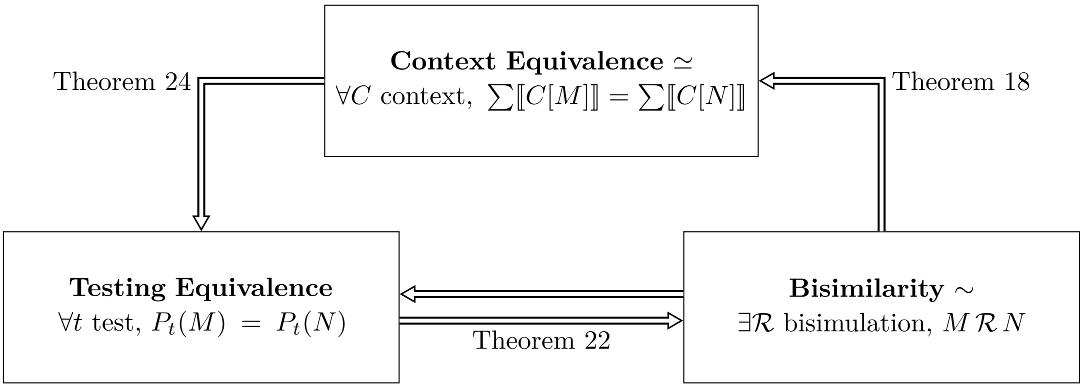

Figure 1 Sketch of the main results in the paper, giving Corollary 25.

non-lazy cbn reduction strategy (i.e. the full head-reduction). We have already mentioned that [10] considers the cbn probabilistic PCF endowed with the let-in operator. The full abstraction result of probabilistic coherence spaces proved in [10] shows that the let-in operator does not change the context equivalence of probabilistic PCF, as this latter corresponds with the equality in probabilistic coherence spaces, regardless of the presence of the let-in in the language. Also, [2, 16] achieve a similar probabilistic coherence spaces full abstraction result for the untyped non-lazy cbn probabilistic λ-calculus without the let-in operator. These considerations show that the need of let-in operator for getting the full abstraction is due to the notion of *lazy* normal form rather than the call-by-name policy.

Structure of the paper. Section 2 defines $\Lambda_{\oplus,\mathsf{let}}$, the lazy call-by-name probabilistic λ-calculus extended with the let-in operator. The operational semantics is given by a notion of big-step approximation, following [8]. An equivalent notion based on Markov chains could be given as in e.g. [9]. The context equivalence is defined by Equation (5) where what we observe is the probability of getting a value. Notice that the notion of *lazyness* plays a crucial role here, since a value is a variable or just an abstraction and not a head-normal form, as it is the case instead in the non-lazy cbn considered in e.g. [2, 9, 15, 16].

Section 3 defines the probabilistic applicative bisimulation and the corresponding bisimilarity by considering $\Lambda_{\oplus,\mathsf{let}}$ as a labelled Markov chain. The definitions and results of this section are an adaptation of the ones in [7]. The main result is the soundness of bisimilarity with respect to the context equivalence (Theorem 18), whose proof is based on Lemma 17 stating that the bisimilarity is a congruence. The proof of this lemma is quite technical but follows the same lines of [4, 5, 7], using Howe's lifting: we postpone the details in the Appendix. The last Section 4 achieves the converse of Theorem 18 by considering Larsen and Skou's testing language (Definition 19) which is well-known to induce an equivalence corresponding with probabilistic bisimilarity (Theorem 22). Lemma 23 states that any test can be represented by a context of $\Lambda_{\oplus,\mathsf{let}}$ (here we are using in an essential way the presence of the let-in operator), so giving Theorem 24 and closing the circle (Corollary 25). Figure 1 sketches the main reasoning of the paper.

2 Preliminaries

In this section we introduce the syntax and operational semantics of $\Lambda_{\oplus,\mathsf{let}}$.

2.1 Probabilistic Lambda Calculus $\Lambda_{\oplus,\mathsf{let}}$

We present the probabilistic lambda calculus $\Lambda_{\oplus,\mathsf{let}}$, that is the pure, untyped lambda calculus endowed with two new operators: a probabilistic binary sum operator $\oplus$, representing a fair choice and a let-in operator, simulating the call-by-value evaluation in a call-by-name calculus. The operational semantics of $\Lambda_{\oplus,\mathsf{let}}$ is defined by a big-step approximation relation as in [8] , we refer to this paper for more details. Given a countable set $X = \{x, y, z, \dots\}$ of variables, term expressions (*terms*) and *values* are generated by the following grammar:

$$\begin{array}{lll} \text{(values)} & V, W & ::= x \mid \lambda x.M, \\ \text{(terms)} & M, N & ::= V \mid MN \mid M \oplus N \mid \mathsf{let}\ x = M\ \mathsf{in}\ N, \end{array} \tag{1}$$

where $x \in X$. The set of all terms (resp. values) is denoted by $\Lambda_{\oplus,\mathsf{let}}$ (resp. $\mathcal{V}_{\oplus,\mathsf{let}}$) and is ranged over by capital Latin letters $M, N, \dots$, the letters V, W being reserved for values. The set of *free variables* of a term M is indicated as $\mathsf{FV}(M)$ and is defined in the usual way. Given a finite set of variables $\Gamma = \{x_1, \dots, x_n\} \subseteq X$, $\Lambda^{\Gamma}_{\oplus,\mathsf{let}}$ (resp. $\mathcal{V}^{\Gamma}_{\oplus,\mathsf{let}}$) denotes the set of terms (resp. values) whose free variables are within Γ. A term M is *closed* if $\mathsf{FV}(M) = \emptyset$, or equivalently if $M \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$. The capture-avoiding substitution of N for the free occurrences of x in M is denoted by $M\{N/x\}$.

▶ **Example 1.** Let us define some terms useful in the sequel. The identity $\mathbf{I} \stackrel{\text{def}}{=} \lambda x.x$, the boolean projections $\mathbf{T} \stackrel{\text{def}}{=} \lambda xy.x$ and $\mathbf{F} \stackrel{\text{def}}{=} \lambda xy.y$ and the duplicator $\boldsymbol{\Delta} \stackrel{\text{def}}{=} \lambda x.xx$, this latter giving the ever looping term $\boldsymbol{\Omega} \stackrel{\text{def}}{=} \boldsymbol{\Delta\Delta}$. The let-in operator allows for a call-by-value duplicator $\boldsymbol{\Delta}^{\ell} \stackrel{\text{def}}{=} \lambda x.\mathsf{let}\ x = x\ \mathsf{in}\ xx$ that will distribute over the probabilistic choice (see Example 3).

Because of the probabilistic operator $\oplus$, a closed term does not evaluate to a single value, but to a discrete distribution of possible outcomes, i.e. to a function assigning a probability to any value. More formally, a *(value) distribution* is a map $\mathscr{D} : \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}} \to \mathbb{R}_{[0,1]}$ such that $\sum_{V \in \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}}} \mathscr{D}(V) \leq 1$. The set of all value distributions is denoted by $\mathcal{P}$. Given a value distribution $\mathscr{D}$, the set of all values to which $\mathscr{D}$ attributes a positive probability is denoted by $\mathsf{S}(\mathscr{D})$ and we will call it *the support of* $\mathscr{D}$. Note that value distributions do not necessarily sum to 1, this allowing to model the possibility of divergence (Example 4). We will use the abbreviation $\sum \mathscr{D}$ to stand for $\sum_{V \in \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}}} \mathscr{D}(V)$. The expression $p_1V_1 + \cdots + p_nV_n$ denotes the distribution $\mathscr{D}$ with finite support $\{V_1, \dots, V_n\}$ such that $\mathscr{D}(V_i) = p_i$, for every $i \in \{1, \dots, n\}$. Note that $\sum \mathscr{D} = \sum_{i=1}^{n} p_i$. In particular, 0 denotes the empty distribution and V can denote both a value and the distribution having all of its mass on V.

The operational semantics of $\Lambda_{\oplus,\mathsf{let}}$ is given in two steps. First, the derivation rules in Figure 2 inductively define a notion of big-step approximation relation $M \Downarrow \mathscr{D}$ between a closed term M and a finite value distribution $\mathscr{D}$. Then, the semantics $[\![M]\!]$ of M is given as:

$$[\![M]\!] = \sup\{\mathscr{D}\ ;\ M \Downarrow \mathscr{D}\}, \tag{2}$$

according to the point-wise order over value distributions ($\mathscr{D} \leq \mathscr{E}$ if and only if $\forall V, \mathscr{D}(V) \leq \mathscr{E}(V)$). The lub in Equation (2) is well-defined since $\leq$ is an ω-complete partial order and the set $\{\mathscr{D}\ ;\ M \Downarrow \mathscr{D}\}$ is directed (for every $M \Downarrow \mathscr{D}$ and $M \Downarrow \mathscr{E}$, then exists a distribution $\mathscr{F} \geq \mathscr{D}, \mathscr{E}$ such that $M \Downarrow \mathscr{F}$).

$$\frac{}{M \Downarrow 0} \qquad \frac{}{V \Downarrow V} \qquad \frac{M \Downarrow \mathscr{D} \quad N \Downarrow \mathscr{E}}{M \oplus N \Downarrow \frac{1}{2}\cdot\mathscr{D} + \frac{1}{2}\cdot\mathscr{E}}$$

$$\frac{M \Downarrow \mathscr{D} \quad \{P\{N/x\} \Downarrow \mathscr{E}_{P,N}\}_{\lambda x.P \in \mathsf{S}(\mathscr{D})}}{MN \Downarrow \sum_{\lambda x.P\in\mathsf{S}(\mathscr{D})} \mathscr{D}(\lambda x.P)\cdot\mathscr{E}_{P,N}} \qquad \frac{N \Downarrow \mathscr{G} \quad \{M\{V/x\} \Downarrow \mathscr{H}_V\}_{V\in\mathsf{S}(\mathscr{G})}}{\mathsf{let}\ x = N\ \mathsf{in}\ M \Downarrow \sum_{V\in\mathsf{S}(\mathscr{G})} \mathscr{G}(V)\cdot\mathscr{H}_V}$$

Figure 2 Rules for the approximation relation $M \Downarrow \mathscr{D}$, with $M \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$ and $\mathscr{D}$ a value distribution.

$$\dfrac{V \Downarrow V \qquad \dfrac{\mathbf{I} \Downarrow \mathbf{I} \qquad \dfrac{V \Downarrow V \qquad \dfrac{\dfrac{\mathbf{I} \Downarrow \mathbf{I} \quad VV \Downarrow 0}{\mathbf{I} \oplus VV \Downarrow \frac{1}{2}\mathbf{I}}}{\vdots}}{VV \Downarrow \sum_{i=1}^{n-1} \frac{1}{2^i}\mathbf{I}}}{\mathbf{I} \oplus VV \Downarrow \sum_{i=1}^{n} \frac{1}{2^i}\mathbf{I}}}{VV \Downarrow \sum_{i=1}^{n} \frac{1}{2^i}\mathbf{I}}$$

Figure 3 A derivation of the big-step approximation $VV \Downarrow \sum_{i=1}^{n} \frac{1}{2^i}\mathbf{I}$ for $V = \lambda x.(\mathbf{I} \oplus xx)$.

Notice that the rules in Figure 2 implement a lazy call-by-name evaluation: they do not reduce within the body of an abstraction, and an application $(\lambda x.M)N$ is evaluated as $M\{N/x\}$ for any term N. However, the let-in operator follows a call-by-value policy: let $x = N$ in M has the same semantics as $M\{N/x\}$ only when N is a value.

▶ **Example 2.** Consider the term $M \overset{\text{def}}{=} \mathbf{\Delta}(\mathbf{T} \oplus \mathbf{F})$. One can easily check that the rules of Figure 2 allows to derive $M \Downarrow \mathscr{D}$ for any $\mathscr{D} \in \{0, \frac{1}{2}\lambda y.(\mathbf{T} \oplus \mathbf{F}), \frac{1}{2}\mathbf{I}, \frac{1}{2}\lambda y.(\mathbf{T} \oplus \mathbf{F}) + \frac{1}{2}\mathbf{I}\}$. The latter distribution is the lub of this set and so it defines the semantics of M.

▶ **Example 3.** Let us replace in Example 2 the duplicator $\mathbf{\Delta}$ with its call-by-value variant $\mathbf{\Delta}^{\ell}$ (Example 1). We have $\mathbf{\Delta}^{\ell}(\mathbf{T} \oplus \mathbf{F}) \Downarrow \mathscr{D}$ for any $\mathscr{D} \in \{0, \frac{1}{2}\lambda y.\mathbf{T}, \frac{1}{2}\mathbf{I}, \frac{1}{2}\lambda y.\mathbf{T} + \frac{1}{2}\mathbf{I}\}$, so $[\![\mathbf{\Delta}^{\ell}(\mathbf{T} \oplus \mathbf{F})]\!] = \frac{1}{2}\lambda y.\mathbf{T} + \frac{1}{2}\mathbf{I}$. Notice that $[\![\mathbf{\Delta}^{\ell}(\mathbf{T} \oplus \mathbf{F})]\!] = [\![\mathbf{\Delta}^{\ell}\mathbf{T} \oplus \mathbf{\Delta}^{\ell}\mathbf{F}]\!] = [\![\mathbf{\Delta}\mathbf{T} \oplus \mathbf{\Delta}\mathbf{F}]\!]$, while $[\![\mathbf{\Delta}(\mathbf{T} \oplus \mathbf{F})]\!] \neq [\![\mathbf{\Delta}\mathbf{T} \oplus \mathbf{\Delta}\mathbf{F}]\!]$, as calculated in Example 2. Let us mention that this phenomenon is well enlightened by the linear logic encoding of the call-by-name application and the call-by-value one, the latter resulting in an operator linear both in the function and the argument position, while the former is linear only in the functional position [12].

▶ **Example 4.** The previous examples are about normalizing terms, in this framework meaning terms M with semantics of total mass $\sum[\![M]\!] = 1$ and such that there exists a unique finite derivation giving $M \Downarrow [\![M]\!]$. Standard non-converging λ-terms gives partiality, as for example $[\![\mathbf{\Omega}]\!] = 0$, so $[\![\mathbf{\Omega} \oplus \mathbf{I}]\!] = \frac{1}{2}\mathbf{I}$. However, probabilistic λ-calculi allow for almost sure terminating terms, that is terms M such that $\sum[\![M]\!] = 1$ but there exists no finite derivation giving $M \Downarrow [\![M]\!]$. For example, consider the term $M \overset{\text{def}}{=} VV$, with $V \overset{\text{def}}{=} \lambda x.(\mathbf{I} \oplus xx)$: any finite approximation of M gives a distribution bounded by $\sum_{i=1}^{n} \frac{1}{2^i}\mathbf{I}$ for some $n \geq 0$, as Figure 3 shows, but only the limit sum $\sup_n \sum_{i=1}^{n} \frac{1}{2^i}\mathbf{I}$ is equal to $[\![M]\!] = \mathbf{I}$.

The following lemma states simple properties of the semantics that can be easily proved by continuity of $[\![\ ]\!]$ and induction over finite approximations (see e.g. [8] for details).

▶ **Lemma 5** ([8]). *For any terms M and N,*

1. $[\![(\lambda x.M)N]\!] = [\![M\{N/x\}]\!]$.
2. $[\![M \oplus N]\!] = \frac{1}{2}[\![M]\!] + \frac{1}{2}[\![N]\!]$.

2.2 Context Equivalence

One standard way of comparing term expressions is by observing their behaviours within programming contexts. A *context* of $\Lambda_{\oplus,\mathsf{let}}$ is a term containing a unique hole $[\cdot]$, generated by the following grammar:

$$C, D ::= [\cdot] \mid \lambda x.C \mid CM \mid MC \mid C \oplus M \mid M \oplus C \mid \mathsf{let}\ x = C\ \mathsf{in}\ M \mid \mathsf{let}\ x = M\ \mathsf{in}\ C \tag{3}$$

If C is a context and M is a $\Lambda_{\oplus,\mathsf{let}}$-term, then $C[M]$ denotes a $\Lambda_{\oplus,\mathsf{let}}$-term obtained by substituting the unique hole in C with M allowing the possible capture of free variables of M. We will work with closing contexts, that is contexts C such that $C[M]$ is a closed term (where M can be an open term). Thus, we want to keep track of the possible variables captured by filling a context hole. Given two finite sets of variables Γ, Δ, we denote by $\mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\Delta)}$ the set of contexts capturing the variables in Γ of a term filling the hole but keeping free the variables in Δ. So for example the context $\lambda x.\mathsf{let}\ y = x \oplus z\ \mathsf{in}\ x[\cdot]$ belongs to $\mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\{x,y\};\Delta)}$ for any Δ containing z.

In a probabilistic setting, the typical observation is the probability to converge to a value, so giving the following standard definition, for every $M, N \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$:

$$M \leq N \text{ iff } \forall C \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}, \sum[\![C[M]]\!] \leq \sum[\![C[N]]\!], \qquad \text{(context preorder)} \tag{4}$$

$$M \simeq N \text{ iff } \forall C \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}, \sum[\![C[M]]\!] = \sum[\![C[N]]\!] \qquad \text{(context equivalence)} \tag{5}$$

Notice that $M \simeq N$ is equivalent to $M \leq N$ and $N \leq M$.

▶ **Example 6.** As mentioned in the Introduction, the terms $M \stackrel{\text{def}}{=} \lambda xy.(x \oplus y)$ and $N \stackrel{\text{def}}{=} (\lambda xy.x) \oplus (\lambda xy.y)$ are context equivalent in the call-by-name probabilistic λ-calculus without the let-in operator [7]. However, they can be discriminated in $\Lambda_{\oplus,\mathsf{let}}$ by, e.g. the context $C \stackrel{\text{def}}{=} (\mathsf{let}\ y = [\cdot]\ \mathsf{in}\ (\mathsf{let}\ z_1 = y\mathbf{I}\mathbf{\Omega}\ \mathsf{in}\ (\mathsf{let}\ z_2 = y\mathbf{I}\mathbf{\Omega}\ \mathsf{in}\ \mathbf{I})))$. In fact, by applying the rules of Figure 2, one gets: $\sum[\![C[M]]\!] = \frac{1}{4}$ and $\sum[\![C[N]]\!] = \frac{1}{2}$.

▶ **Example 7.** The two duplicators $\mathbf{\Delta}$ and $\mathbf{\Delta}^\ell$ (Example 1) are not context equivalent, for example $C \stackrel{\text{def}}{=} [\cdot](\mathbf{I} \oplus \mathbf{\Omega})$ gives $\sum[\![C[\mathbf{\Delta}]]\!] = \frac{1}{4}$ while $\sum[\![C[\mathbf{\Delta}^\ell]]\!] = \frac{1}{2}$.

▶ **Proposition 8.** *Let $M, N \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$, if $[\![M]\!] \leq [\![N]\!]$ then $M \leq N$. So, $[\![M]\!] = [\![N]\!]$ implies $M \simeq N$.*

Proof. First, notice that $[\![M]\!] \leq [\![N]\!]$ is equivalent to $\forall \mathcal{D}, M \Downarrow \mathcal{D}, \exists \mathcal{E} \geq \mathcal{D}, N \Downarrow \mathcal{E}$. Then one proves, by structural induction on a context C that $[\![C(M)]\!] \leq [\![C(N)]\!]$, whenever $[\![M]\!] \leq [\![N]\!]$. The delicate points are in the cases C is an application or a let-in operator. ◀

▶ **Example 9.** Thanks to Proposition 8, one can prove that quite different terms are indeed context equivalent, e.g. the term VV in Example 4 is context equivalent to $\mathbf{I}$. However, not all context equivalent terms have the same semantics, as for example $\lambda x.(x \oplus x)$ and $\mathbf{I}$.

Proving in general that two terms are context equivalent is rather difficult because of the universal quantifier in Equation (5). For example, proving that $\lambda x.(x \oplus x)$ and $\mathbf{I}$ are context equivalent is not immediate. Various other tools are then used to prove context equivalence, as the bisimilarity and testing introduced in the next sections.

3 Probabilistic Applicative Bisimulation

We briefly recall and adapt to $\Lambda_{\oplus,\mathsf{let}}$ the definitions of [7] about probabilistic applicative (bi)simulation. This notion mixes Larsen and Skou's definition of (bi)simulation for labelled Markov chains [14] with Abramsky's applicative (bi)simulation for the lazy call-by-name λ-calculus [1]. The core idea is to look at a closed term M as a state of a transition system, a Markov chain in our setting, having two kinds of transitions. A "solipsistic" transition consisting in evaluating M to a value $\lambda x.P$ (this transition being weighted by the probability $[\![M]\!](\lambda x.P)$ of getting $\lambda x.P$ out of M) and an "interactive" transition consisting in feeding a value $\lambda x.P$ by a new term N representing an input from the environment, so getting the term $P\{N/x\}$. We can then consider the notions of similarity and bisimilarity (resp. (6), (7)) over such probabilistic transition system. The benefit of this approach is to check program equivalence via an existential quantifier (see Equation (7)) rather than a universal one as in context equivalence (Equation (5)). The main result of this section is Theorem 18 stating that similarity implies context preorder. As a consequence we have that bisimilarity implies context equivalence. The key ingredient for achieving this result is to show that the similarity is a precongruence relation (Definition 16 and Lemma 17). The proof of Lemma 17 is quite technical but standard, see the Appendix and [7] for more details.

We start with the definition of a generic labelled Markov chain and following Larsen and Skou [14] we introduce the notions of a probabilistic simulation and bisimulation.

▶ **Definition 10.** *A* labelled Markov chain *is a triple $\mathcal{M} = (\mathcal{S}, \mathcal{L}, P)$ where $\mathcal{S}$ is a countable set of states, $\mathcal{L}$ is a set of labels (actions) and P is a transition probability matrix, i.e. a function $P : \mathcal{S} \times \mathcal{L} \times \mathcal{S} \to \mathbb{R}_{[0,1]}$ satisfying the following condition: $\forall s \in \mathcal{S}, \forall l \in \mathcal{L}, \sum_{t\in\mathcal{S}} P(s, l, t) \leq 1$.*

Given a relation $\mathcal{R}$, $\mathcal{R}(X)$ denotes the $\mathcal{R}$-closure of the set X, namely the set $\{y \mid \exists x \in X \text{ such that } x\mathcal{R}y\}$. If $\mathcal{R}$ is an equivalence relation, then $\mathcal{S}/\mathcal{R}$ stands for the set of all equivalence classes of $\mathcal{S}$ modulo $\mathcal{R}$. The expression $P(s, l, X)$ stands for $\sum_{t\in X} P(s, l, t)$.

▶ **Definition 11.** *Let $(\mathcal{S}, \mathcal{L}, P)$ be a labelled Markov chain and $\mathcal{R}$ be a relation over $\mathcal{S}$:*
- *$\mathcal{R}$ is a* probabilistic simulation *if it is a preorder and $\forall(s, t) \in \mathcal{R}, \forall X \subseteq \mathcal{S}, \forall l \in \mathcal{L}$, $P(s, l, X) \leq P(t, l, \mathcal{R}(X))$.*
- *$\mathcal{R}$ is a* probabilistic bisimulation *if it is an equivalence and $\forall(s, t) \in \mathcal{R}, \forall E \in \mathcal{S}/\mathcal{R}, \forall l \in \mathcal{L}$, $P(s, l, E) = P(t, l, E)$.*

We define the probabilistic (bi)similarity, denoted respectively by $\lesssim$ and $\simeq$, as the union of all probabilistic (bi)simulations which can be proven to be still a (bi)simulation:

$$M \lesssim N \text{ iff } \exists\mathcal{R} \text{ probabilistic simulation s.t. } M\mathcal{R}N, \qquad \text{(probabilistic similarity)} \qquad (6)$$

$$M \sim N \text{ iff } \exists\mathcal{R} \text{ probabilistic bisimulation s.t. } M\mathcal{R}N \qquad \text{(probabilistic bisimilarity)} \qquad (7)$$

One can prove that $M \sim N$ is equivalent to $M \lesssim N$ and $N \lesssim M$, i.e. $\sim = \lesssim \cap \lesssim^{op}$.

As previously stated, we want to see the operational semantics of $\Lambda_{\oplus,\mathsf{let}}$ as a labelled Markov chain defined as follows:

▶ **Definition 12.** *The $\Lambda_{\oplus,\mathsf{let}}$-Markov chain is defined as the triple $(\Lambda^{\emptyset}_{\oplus,\mathsf{let}} \uplus \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}}, \Lambda^{\emptyset}_{\oplus,\mathsf{let}} \cup \{\tau\}, P)$, where the set of states is the disjoint union of the set of closed terms and closed distinguished values, labels (actions) are either closed terms or τ action and the transition probability matrix P is defined in the following way:*
- *for every closed term M and distinguished value $\nu x.N$,*

$$P(M, \tau, \nu x.N) = [\![M]\!](\lambda x.N)\,,$$

- *for every closed term M and distinguished value $\nu x.N$,*

$$P(\nu x.N, M, N\{M/x\}) = 1\,,$$

- *in all other cases, P returns* 0.

For technical reasons the set of states is represented as a disjoint union $\Lambda^{\emptyset}_{\oplus,\mathsf{let}} \uplus \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}}$. For every closed value $V = \lambda x.N \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$ a distinguished value is indicated as $\widetilde{V} = \nu x.N$ and belongs to the set $\mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}}$. As an example, value $\lambda xy.x$ belongs to the set $\Lambda^{\emptyset}_{\oplus,\mathsf{let}}$, while the distinguished value $\nu x.\lambda y.x$ is the element of $\mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}}$.

Since $\Lambda_{\oplus,\mathsf{let}}$ can be seen as a labelled Markov chain, the simulation and bisimulation can be defined as for any labelled Markov chain. *A probabilistic applicative simulation* is a probabilistic simulation on $\Lambda_{\oplus,\mathsf{let}}$ and *a probabilistic applicative bisimulation* is a probabilistic bisimulation on $\Lambda_{\oplus,\mathsf{let}}$. Then, the *probabilistic applicative similarity*, PAS for short, and the *probabilistic applicative bisimilarity*, PAB for short, are defined in the usual way applying Equation (6) and (7). From now on, the symbol $\lesssim$ (resp. $\sim$) will denote the probabilistic applicative similarity (resp. bisimilarity).

The notions of PAS and PAB are defined on closed terms, and we extend these definitions to open terms by requiring the usual closure under substitutions. Let $M, N \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$ where $\Gamma = \{x_1, \ldots, x_n\}$. We say M and N are similar, (denoted $M \lesssim N$), if for all $L_1 \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}, \ldots, L_n \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$, $M\{L_1/x_1, \ldots, L_n/x_n\} \lesssim N\{L_1/x_1, \ldots, L_n/x_n\}$. The analogous terminology is introduced for bisimilarity.

▶ **Example 13.** Let us recall the terms $\lambda x.(x \oplus x)$ and $\lambda x.x$ from Example 9 having different semantics but context equivalent. As mentioned, the proof of their context equivalence is not immediate, because of the universal quantifier in Equation (5). However, we can check easily that they are bimisilar, because we need just to exhibit a bisimulation relation between the two terms. By Theorem 18 we then infer context equivalence from bisimilarity. Let us define the relation $\mathcal{R} = \{(\lambda x.(x \oplus x), \lambda x.x)\} \cup \{(\lambda x.x, \lambda x.(x \oplus x))\} \cup \{(\nu x.(x \oplus x), \nu x.x)\} \cup \{(\nu x.x, \nu x.(x \oplus x))\} \cup \{(N \oplus N, N) \mid N \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}\} \cup \{(N, N \oplus N) \mid N \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}\} \cup \{(M, M) \mid M \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}\} \cup \{(\widetilde{V}, \widetilde{V}) \mid \widetilde{V} \in \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}}\}$. We prove that $\mathcal{R}$ is a bisimulation containing $(\lambda x.(x \oplus x), \lambda x.x)$. The relation is trivially an equivalence, so we have to show that $\forall (M, N) \in \mathcal{R}, \forall E \in (\Lambda^{\emptyset}_{\oplus,\mathsf{let}} \uplus \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}})/\mathcal{R}, \forall \ell \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}} \cup \{\tau\}$, $P(M, \ell, E) = P(N, \ell, E)$ (Definition 11). We prove only for $(\lambda x.(x \oplus x), \lambda x.x) \in \mathcal{R}$ and $(\nu x.(x \oplus x), \nu x.x) \in \mathcal{R}$. First we have that $(\lambda x.(x \oplus x), \lambda x.x) \in \mathcal{R}$ and for all closed terms $F \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$ and all equivalence classes $E \in (\Lambda^{\emptyset}_{\oplus,\mathsf{let}} \uplus \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}})/\mathcal{R}$, $P(\lambda x.(x \oplus x), F, E) = 0 = P(\lambda x.x, F, E)$ holds by Definition 12. If the equivalence class E contains $\nu x.(x \oplus x)$ then $P(\lambda x.(x \oplus x), \tau, E) = 1$, otherwise $P(\lambda x.(x \oplus x), \tau, E) = 0$. Since $(\nu x.(x \oplus x), \nu x.x) \in \mathcal{R}$, we have that $\nu x.(x \oplus x) \in E$ if and only if $\nu x.x \in E$. Hence, $P(\lambda x.(x \oplus x), \ell, E) = P(\lambda x.x, \ell, E)$ for all $\ell \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}} \cup \{\tau\}$ and all $E \in (\Lambda^{\emptyset}_{\oplus,\mathsf{let}} \uplus \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}})/\mathcal{R}$. For all equivalence classes $E \in (\Lambda^{\emptyset}_{\oplus,\mathsf{let}} \uplus \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}})/\mathcal{R}$, $P(\nu x.(x \oplus x), \tau, E) = 0 = P(\nu x.x, \tau, E)$ holds by Definition 12. Further, $P(\nu x.(x \oplus x), F, E) = 1$ for some $F \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$ if $F \oplus F \in E$, otherwise $P(\nu x.(x \oplus x), F, E) = 0$. We have that $F \oplus F \in E$ if and only if $F \in E$, because $(F \oplus F, F) \in \mathcal{R}$ for all $F \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$. Hence, $P(\nu x.(x \oplus x), \ell, E) = P(\nu x.x, \ell, E)$ for all $\ell \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}} \cup \{\tau\}$ and all $E \in (\Lambda^{\emptyset}_{\oplus,\mathsf{let}} \uplus \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}})/\mathcal{R}$. The proof for the other elements of $\mathcal{R}$ is analogous to the cases we considered.

▶ **Example 14.** The terms $M = \lambda xy.(x \oplus y)$ and $N = (\lambda xy.x) \oplus (\lambda xy.y)$ are not bisimilar. Let us suppose the opposite. Then, there exists a bisimulation $\mathcal{R}$ such that $(M, N) \in \mathcal{R}$. By definition $\mathcal{R}$ is an equivalence relation. Let E be an equivalence class of $\Lambda^{\emptyset}_{\oplus,\mathsf{let}} \uplus \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}}$ with respect to $\mathcal{R}$ which contains $\nu x.\lambda y.(x \oplus y)$. Then, we should have that $1 = P(M, \tau, E) =$

$P(N, \tau, E)$. We know that $P(N, \tau, \nu x.\lambda y.x) = \frac{1}{2}$ and $P(N, \tau, \nu x.\lambda y.y) = \frac{1}{2}$. Thus, we can conclude $\nu x.\lambda y.x \in E$ and $\nu x.\lambda y.y \in E$. If $\nu x.\lambda y.x \in E$, then $(\nu x.\lambda y.(x \oplus y), \nu x.\lambda y.x) \in \mathcal{R}$. Hence we have that $1 = P(\nu x.\lambda y.(x \oplus y), \mathbf{\Omega}, E_1) = P(\nu x.\lambda y.x, \mathbf{\Omega}, E_1)$, where E_1 is an equivalence class which contains $\lambda y.(\mathbf{\Omega} \oplus y)$. Using the fact that $P(\nu x.\lambda y.x, \mathbf{\Omega}, \lambda y.\mathbf{\Omega}) = 1$ we obtain $\lambda y.\mathbf{\Omega} \in E_1$. Since $\lambda y.(\mathbf{\Omega} \oplus y)$ and $\lambda y.\mathbf{\Omega}$ belong to the same equivalence class we conclude $(\lambda y.(\mathbf{\Omega} \oplus y), \lambda y.\mathbf{\Omega}) \in \mathcal{R}$. If E_2 is an equivalence class such that $\nu y.(\mathbf{\Omega} \oplus y) \in E_2$, then we have that $1 = P(\lambda y.(\mathbf{\Omega} \oplus y), \tau, E_2) = P(\lambda y.\mathbf{\Omega}, \tau, E_2)$. By a similar reasoning as before we obtain that $(\nu y.(\mathbf{\Omega} \oplus y), \nu y.\mathbf{\Omega}) \in \mathcal{R}$. Let E_3 be an equivalence class which contains $\mathbf{\Omega} \oplus \mathbf{I}$. From $1 = P(\nu y.(\mathbf{\Omega} \oplus y), \mathbf{I}, E_3) = P(\nu y.\mathbf{\Omega}, \mathbf{I}, E_3)$ it follows that $\mathbf{\Omega} \in E_3$, i.e. $(\mathbf{\Omega} \oplus \mathbf{I}, \mathbf{\Omega}) \in \mathcal{R}$. Finally, if E_4 is an equivalence class such that $\nu x.x \in E_4$, then $\frac{1}{2} = P(\mathbf{\Omega} \oplus \mathbf{I}, \tau, E_4) = P(\mathbf{\Omega}, \tau, E_4)$. This is in contradiction with $P(\mathbf{\Omega}, \tau, E_4) = 0$ which is a consequence of the definition of a transition probability matrix. Thus, terms M and N are not bisimilar.

The following proposition is the analogous to Proposition 8, stating the soundness of (bi)simulation with respect to the operational semantics.

▶ **Proposition 15.** *Let $M, N \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$, if $[\![M]\!] \leq [\![N]\!]$ then $M \lesssim N$. So, $[\![M]\!] = [\![N]\!]$ implies $M \sim N$.*

Proof. By checking that the relation $\mathcal{R} = \{(M, N) \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}} \times \Lambda^{\emptyset}_{\oplus,\mathsf{let}} \mid [\![M]\!] \leq [\![N]\!]\} \cup \{(\widetilde{V}, \widetilde{V}) \in \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}} \times \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}}\}$ is a probabilistic applicative simulation. The second part of the statement follows from $\sim = \lesssim \cap (\lesssim)^{op}$. ◀

We introduce a new notion of relations called $\Lambda_{\oplus,\mathsf{let}}$-relations, which are sets of triples in the form (Γ, M, N) where $M, N \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$. Any relation R' on the set of $\Lambda_{\oplus,\mathsf{let}}$-terms can be extended to a $\Lambda_{\oplus,\mathsf{let}}$-relation $\mathcal{R}$, such that whenever $(M, N) \in R'$ and $M, N \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$, we have that $(\Gamma, M, N) \in \mathcal{R}$. We will write $\Gamma \vdash M\mathcal{R}N$ instead of $(\Gamma, M, N) \in \mathcal{R}$.

▶ **Definition 16.** *A $\Lambda_{\oplus,\mathsf{let}}$-relation $\mathcal{R}$ is a* congruence *(respectively,* precongruence*) if it is an equivalence (respectively, a preorder) and for every $\Gamma \cup \Delta \vdash M\mathcal{R}N$ and every context $C \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\Delta)}$, we have that $\Delta \vdash C[M]\mathcal{R}C[N]$.*

It is immediate to check that the context preorder $\leq$ (resp. equivalence $\simeq$) is a precongruence (resp. congruence)(Appendix A.1). Also (bi)similarity is a (pre)congruence, but its proof is more involved (Appendix A.2).

▶ **Lemma 17.** *The similarity $\lesssim$ (resp. bisimilarity $\sim$) is a precongruence (resp. congruence) relation for $\Lambda_{\oplus,\mathsf{let}}$-terms.*

Proof (Sketch). As standard [4, 5, 7], we use Howe's technique to prove that probabilistic similarity is a precongruence, this implying that the probabilistic bisimilarity is also a congruence. The proof is technical and follows the same reasoning as [7], the only difference being in the cases needed to handle the compatibility associated with the let-in operator.

We start with defining Howe's lifting for $\Lambda_{\oplus,\mathsf{let}}$, which turns an arbitrary relation $\mathcal{R}$ to another one $\mathcal{R}^H$. The relation $\mathcal{R}^H$ enjoys some properties with respect to the relation $\mathcal{R}$. In particular, if $\mathcal{R}$ is reflexive, transitive and closed under term-substitution, then it is included in $\mathcal{R}^H$ and the relation $\mathcal{R}^H$ is context closed and also closed under term-substitution. These properties allow to prove that the transitive closure $(\lesssim^H)^+$ of the Howe's lifting $\lesssim^H$ is a precongruence including $\lesssim$. One can conclude then easily that $\lesssim$ is also a precongruence. Finally, from $\sim = \lesssim \cup (\lesssim)^{op}$ we conclude that $\sim$ is a congruence. ◀

Now we can prove that simulation preorder is sound with respect to the context preorder. As a consequence we have that bisimilarity is included in the context equivalence.

▶ **Theorem 18** (Soundness). *For every $M, N \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$, $\Gamma \vdash M \lesssim N$ implies $\Gamma \vdash M \leq N$. Therefore, $M \sim N$ implies $\Gamma \vdash M {\simeq} N$.*

Proof. Suppose that $\Gamma \vdash M \lesssim N$. We have that for every context $C \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}$, $\emptyset \vdash C[M] \lesssim C[N]$ holds as a consequence of Lemma 17. Then by definition there exists a simulation between $C[M]$ and $C[N]$, which implies by Definition 11 that $\sum[\![C[M]]\!] \leq \sum[\![C[N]]\!]$ holds. We conclude $\Gamma \vdash M \leq N$. The second part of the statement follows from the definitions $\sim = \lesssim \cap \lesssim^{op}$ and $\simeq = \leq \cap \leq^{op}$. ◀

4 Full Abstraction

The goal of this section is to prove the converse of Theorem 18, showing that context equivalence and bisimilarity coincide. In order to get this result, it is more convenient to use the notion of testing equivalence, which has been proven to coincide with Markov processes bisimilarity in [20] (here Theorem 22). In this framework we need to consider only Markov chains, which are the discrete-time version of Markov processes, so we simplify the definitions and results of [20] to this discrete setting, following [7]. Notice that Theorem 22 is independent from the particular Markov chain considered, so we recall the general definitions and then we applied them to the $\Lambda_{\oplus,\mathsf{let}}$-Markov chain.

▶ **Definition 19** ([7]). *Let $(\mathcal{S}, \mathcal{L}, \mathcal{P})$ be a labelled Markov chain. The testing language $\mathcal{T}_{(\mathcal{S},\mathcal{L},\mathcal{P})}$ for $(\mathcal{S}, \mathcal{L}, \mathcal{P})$ is given by the grammar*

$$t ::= \omega \mid a.t \mid (t, t),$$

where ω is a symbol for termination and $a \in \mathcal{L}$ is an action (label).

It is easy to see that tests are finite objects. A test is an algorithm for doing an experiment on a program. During the execution of a test on a particular program, one can observe the success or the failure of the experiment with a given probability. The symbol ω represents a test which does not require an experiment at all (it always succeed). The test $a.t$ describes an experiment consisting of performing the action a and in the case of success performing the test t, and the test (t, s) makes two copies of the current state and allows both tests t and s to be performed independently on the same state. The success probability of a test is defined as follows:

▶ **Definition 20** ([7]). *Let $(\mathcal{S}, \mathcal{L}, \mathcal{P})$ be a labelled Markov chain. We define a family $\{P_t(\cdot)\}_{t\in\mathcal{T}_{(\mathcal{S},\mathcal{L},\mathcal{P})}}$ of maps from the set of states $\mathcal{S}$ to $\mathbb{R}_{[0,1]}$, by induction on the structure of t:*

- $P_\omega(s) = 1$;
- $P_{a.t}(s) = \sum_{s'\in\mathcal{S}} \mathcal{P}(s, a, s')P_t(s')$;
- $P_{(t_1,\ldots,t_n)}(s) = \prod_{i=1}^{n} P_{t_i}(s)$.

▶ **Example 21.** The terms $\lambda xy.(x \oplus y)$ and $(\lambda xy.x) \oplus (\lambda xy.y)$ of Example 6 can be discriminated by the test $t = \tau.(\mathbf{I}.\tau.\mathbf{\Omega}.\tau.\omega, \mathbf{I}.\tau.\mathbf{\Omega}.\tau.\omega)$. Figure 4 sketches the computation of $P_t(\lambda xy.(x \oplus y)) = \frac{1}{4}$ and $P_t((\lambda xy.x) \oplus (\lambda xy.y)) = \frac{1}{2}$.

The following theorem states the equivalence between the notion of bisimilarity over $(\mathcal{S}, \mathcal{L}, \mathcal{P})$ and testing equivalence. The theorem has been proven in [20] for a labelled Markov processes. For lack of space, we have omitted a detailed proof of the adaptation of the results from labelled Markov processes to labelled Markov chains.

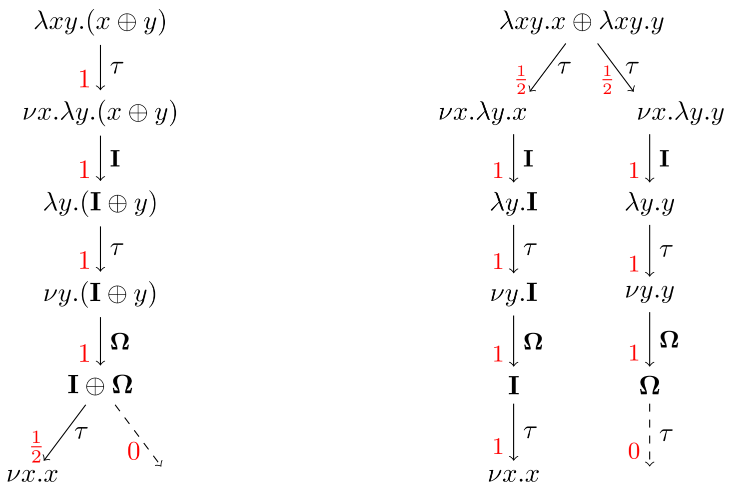

Figure 4 The experiment $t = \tau.(\mathbf{I}.\tau.\mathbf{\Omega}.\tau.\omega, \mathbf{I}.\tau.\mathbf{\Omega}.\tau.\omega)$ over the terms of Example 21.

▶ **Theorem 22** ([7],[20]). *Let $(\mathcal{S}, \mathcal{L}, \mathcal{P})$ be a labelled Markov chain. Then $s, s' \in \mathcal{S}$ are bisimilar if and only if $P_t(s) = P_t(s')$ for every test $t \in \mathcal{T}_{(\mathcal{S},\mathcal{L},\mathcal{P})}$.*

It is known that this theorem does not hold for inequalities [20]. More precisely, it is not true that $s \lesssim s'$ just in case $P_t(s) \leq P_t(s')$ for every test $t \in \mathcal{T}_{(\mathcal{S},\mathcal{L},\mathcal{P})}$.

4.1 Every Test has an Equivalent Context

Here is the main contribution of our paper, showing that for every test t associated with the $\Lambda_{\oplus,\mathsf{let}}$-Markov chain there exists a context C_t expressing t in the syntax of $\Lambda_{\oplus,\mathsf{let}}$, i.e. $P_t(M) = \sum [\![C_t[M]]\!]$ for every term M (Lemma 23). So context equivalence implies testing equivalence (Theorem 24) and hence bisimilarity by Theorem 22. Together with Theorem 18 this achieves the diagram in Figure 1, so Corollary 25.

▶ **Lemma 23.** *For every test $t \in \mathcal{T}_{\Lambda_{\oplus,\mathsf{let}}}$, there are contexts $C_t \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\emptyset;\emptyset)}$ and $D_t \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\emptyset;\emptyset)}$ such that for every term $M \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$ and value $V \in \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}}$ it holds that:*

$$P_t(M) = \sum [\![C_t[M]]\!] \quad \text{and} \quad P_t(\widetilde{V}) = \sum [\![D_t[V]]\!],$$

where $\widetilde{V}$ is a distinguished value from the set $\mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}}$.

Proof. We prove it by induction on the structure of a test t.

- First we consider the case where $t = \omega$. Then, by the definition of $P_t(\cdot)$, we have that for every $M \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$ and $V \in \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}}$, $P_\omega(M) = 1$ and $P_\omega(\widetilde{V}) = 1$. Thus, we can define $C_\omega = (\lambda xy.x)[\cdot]$ and $D_\omega = (\lambda xy.x)[\cdot]$ and we obtain, for every $M \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$

$$\sum [\![C_\omega[M]]\!] = \sum [\![(\lambda xy.x)M]\!] = \sum [\![\lambda y.M]\!] = 1 = P_\omega(M),$$

and for every value $V \in \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}}$

$$\sum [\![D_\omega[V]]\!] = \sum [\![(\lambda xy.x)V]\!] = \sum [\![\lambda y.V]\!] = 1 = P_\omega(\widetilde{V}).$$

- Next, let us consider the case where $t = a.t'$ for some action (label) a. By induction hypothesis there are contexts $C_{t'} \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\emptyset;\emptyset)}$ and $D_{t'} \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\emptyset;\emptyset)}$ such that for every $M \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$ and $V \in \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}}$ we have that $P_{t'}(M) = \sum [\![C_{t'}[M]]\!]$ and $P_{t'}(\widetilde{V}) = \sum [\![D_{t'}[V]]\!]$. An action a can be either a closed term or a τ action, thus depending on it we differ two cases.
  1. If $a = \tau$, then a test t is of the form $\tau.t'$. From Definition 12 and Definition 20 we have $P_{\tau.t'}(\widetilde{V}) = 0$ for any value $V \in \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}}$. Hence, we define $D_{\tau.t'} = \Omega[\cdot]$ and the statement holds. Let M be a closed term. From the definition of a transition probability matrix $(P(M,\tau,\widetilde{V}) = [\![M]\!](V))$ and induction hypothesis $P_{t'}(\widetilde{V}) = \sum [\![D_{t'}[V]]\!]$ it follows that

$$P_{\tau.t'}(M) = \sum_{\widetilde{V} \in \mathsf{V}\Lambda^{\emptyset}_{\oplus,\mathsf{let}}} P(M,\tau,\widetilde{V}) P_{t'}(\widetilde{V}) = \sum_{V \in \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}}} [\![M]\!](V) \cdot \sum [\![D_{t'}[V]]\!].$$

  We define $C_{\tau.t'} = (\mathsf{let}\ y = [\cdot]\ \mathsf{in}\ D_{t'}[y])$. Then, by the definition of operational semantics we get

$$\sum [\![C_{\tau.t'}[M]]\!] = \sum [\![\mathsf{let}\ y = M\ \mathsf{in}\ D_{t'}[y]]\!] = \sum_{V \in \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}}} [\![M]\!](V) \cdot \sum [\![D_{t'}[V]]\!],$$

  for any closed term $M \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$. Thus, $P_{\tau.t'}(M) = \sum [\![C_{\tau.t'}[M]]\!]$.
  2. If $a = F$ for some closed term F, then a test t is of the form $F.t'$. From Definition 12 and Definition 20 we have $P_{F.t'}(M) = 0$ for any term $M \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$. Hence, we define $C_{F.t'} = \Omega[\cdot]$ and the statement holds. Let V be a value $\lambda x.N$ ($\widetilde{V} = \nu x.N$). From the definition of a transition probability matrix ($P(\nu x.N, F, N\{F/x\}) = 1$) and induction hypothesis, $P_{t'}(M) = \sum [\![C_{t'}[M]]\!]$ for every $M \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$, it follows that

$$\begin{aligned} P_{F.t'}(\widetilde{V}) &= \sum_{N' \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}} P(\widetilde{V}, F, N') P_{t'}(N') \\ &= P(\nu x.N, F, N\{F/x\}) \cdot P_{t'}(N\{F/x\}) \\ &= 1 \cdot P_{t'}(N\{F/x\}) = \sum [\![C_{t'}[N\{F/x\}]]\!] \end{aligned}$$

  By Lemma 5 terms $N\{F/x\}$ and $(\lambda x.N)F$ have the same semantics. Hence, they are bisimilar (Proposition 15). Due to the fact that bisimilarity is included in context equivalence (Theorem 18) we have that terms $N\{F/x\}$ and $(\lambda x.N)F$ are context equivalent. More precisely, for any context C, $\sum [\![C[N\{F/x\}]]\!] = \sum [\![C[(\lambda x.N)F]]\!]$. Finally, we obtain that

$$P_{F.t'}(\widetilde{V}) = \sum [\![C_{t'}[N\{F/x\}]]\!] = \sum [\![C_{t'}[(\lambda x.N)F]]\!] = \sum [\![C_{t'}[VF]]\!].$$

  We define $D_{F.t'} = C_{t'}[[\cdot]F]$. Then, we have that $\sum [\![D_{F.t'}[V]]\!] = \sum [\![C_{t'}[VF]]\!]$, holds for any value $V \in \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}}$. Thus, $P_{F.t'}(\widetilde{V}) = \sum [\![D_{F.t'}[V]]\!]$.
- Finally, let $t = (t_1, t_2)$. By induction hypothesis there exist contexts $C_{t_1}, D_{t_1}, C_{t_2}, D_{t_2} \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\emptyset;\emptyset)}$ such that for any closed term M and a value V the following holds:

$$P_{t_1}(M) = \sum [\![C_{t_1}[M]]\!], \quad P_{t_1}(\widetilde{V}) = \sum [\![D_{t_1}[V]]\!],$$

$$P_{t_2}(M) = \sum [\![C_{t_2}[M]]\!] \quad \text{and} \quad P_{t_2}(\widetilde{V}) = \sum [\![D_{t_2}[V]]\!].$$

From Definition 20 we have

$$P_{(t_1,t_2)}(M) = P_{t_1}(M) \cdot P_{t_2}(M) = \sum [\![C_{t_1}[M]]\!] \cdot \sum [\![C_{t_2}[M]]\!],$$

for any closed term $M \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$. We define:

$$C_{(t_1,t_2)} = (\lambda y.(\mathsf{let}\ z_1 = C_{t_1}[y]\ \mathsf{in}\ (\mathsf{let}\ z_2 = C_{t_2}[y]\ \mathsf{in}\ I)))[\cdot] \tag{8}$$

and by the definition of operational semantics we have

$$\sum [\![C_{(t_1,t_2)[M]}]\!] = \sum [\![C_{t_1}[M]]\!] \cdot \sum [\![C_{t_2}[M]]\!].$$

Since, for a value $V \in \mathcal{V}^{\emptyset}_{\oplus,\mathsf{let}}$ it holds that

$$P_{(t_1,t_2)}(\widetilde{V}) = P_{t_1}(\widetilde{V}) \cdot P_{t_2}(\widetilde{V}) = \sum [\![D_{t_1}[V]]\!] \cdot \sum [\![D_{t_2}[V]]\!],$$

we define $D_{(t_1,t_2)} = (\lambda y.(\mathsf{let}\ z_1 = D_{t_1}[y]\ \mathsf{in}\ (\mathsf{let}\ z_2 = D_{t_2}[y]\ \mathsf{in}\ I)))[\cdot]$ and the statement holds.

This concludes the proof. ◀

▶ **Theorem 24.** *Let $M, N \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$, $M \simeq N$ implies that $P_t(M) = P_t(N)$, for every test t.*

Proof. It is a straightforward consequence of Lemma 23. Let us assume that terms M and N are context equivalent, $\emptyset \vdash M \simeq N$. Then, for every context $C \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}^{(\emptyset;\emptyset)}$,we have $\sum [\![C[M]]\!] = \sum [\![C[N]]\!]$. Suppose that there exists test $t \in \mathcal{T}_{\Lambda_{\oplus,\mathsf{let}}}$ such that $P_t(M) \neq P_t(N)$. By Lemma 23, we have that there exists context C_t such that for every term M, $P_t(M) = \sum [\![C_t[M]]\!]$. Then, for this context C_t, it holds that $\sum [\![C_t[M]]\!] = P_t(M) \neq P_t(N) = \sum [\![C_t[N]]\!]$, which is in contradiction with the assumption that M and N are context equivalent. Hence, for every test $t \in \mathcal{T}_{\Lambda_{\oplus,\mathsf{let}}}$ it holds that $P_t(M) = P_t(N)$. ◀

Notice that the let-in operator is crucial in defining the context $C_{(t_1,t_2)}$ associated with the product (t_1, t_2) of tests (Equation (8)) in the proof of Lemma 23. For example, if we consider the call-by-name version of $C_{(t_1,t_2)}$, i.e. the context $C = (\lambda y.(\lambda z_1 z_2.I)D_{t_1}[y]D_{t_2}[y])[\cdot]$, then the semantics of $C[M]$ is independent from the contexts $D_{t_1}[\cdot]$, $D_{t_2}[\cdot]$ and the term M, being $[\![C[M]]\!] = I$. Hence, we cannot have $P_{(t_1,t_2)}(M) = \sum [\![C[M]]\!]$ for every M. Another possibility is to try to use a context not erasing $D_{t_1}[\cdot]$ and $D_{t_2}[\cdot]$ during the evaluation, as for example in $C = (\lambda y.D_{t_1}[y]D_{t_2}[y])[\cdot]$. However this would imply to be able to control the result of $D_{t_1}[M]$ for every term M, for example supposing $[\![D_{t_1}[M]]\!] = P_{t_1}(M)I$, which increases considerably the difficulty of the proof. Anyway, the fact that there are examples of terms distinguished by tests (Example 21) but not by contexts without the let-in operator (Example 6) shows the necessity of this latter.

The following resumes all results in the paper, as sketched in Figure 1:

▶ **Corollary 25** (Full Abstraction). *For any $M, N \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$, the following items are equivalent:*
(context equivalence) *$M \simeq N$,*
(bisimilarity) *$M \sim N$,*
(testing equivalence) *$P_t(M) = P_t(N)$ for all tests t.*

Concerning inequalities, the equivalence of similarity and testing preorder, i.e. a relation which contains (s, s') if and only if $P_t(s) \leq P_t(s')$ for every test $t \in \mathcal{T}_{(\mathcal{S},\mathcal{L},\mathcal{P})}$, does not hold as we stated before. So, we have no clue for proving that similarity is fully abstract with respect to the context preorder. We actually conjecture that full abstraction for similarity does not hold for $\Lambda_{\oplus,\mathsf{let}}$.

5 Conclusion

In this paper we have considered the $\Lambda_{\oplus,\mathsf{let}}$-calculus, a pure untyped λ-calculus extended with two operators: a probabilistic choice operator $\oplus$ and a let-in operator. The calculus implements a lazy call-by-name evaluation strategy, following [1, 7], however the let-in operator allows for a call-by-value passing policy. We prove that context equivalence, bisimilarity and testing equivalence all coincide in $\Lambda_{\oplus,\mathsf{let}}$ (Corollary 25).

Concerning the inequalities associated with these equivalences: it is known that that the probabilistic similarity does not imply the testing approximation [20]. We prove that similarity implies context preorder (Theorem 18), but it remains open whether also the converse holds.

This paper confirms a conjecture stated in [4], showing that the calculus introduced in [7] can be endowed with a fully abstract bisimilarity by adding a let-in operator. As discussed in the Introduction, our feeling is that the need of this operator is due to the lazyness rather than to the cbn policy of the calculus. In order to precise this intuition we plan to investigate the definition of bisimilarity for the non-lazy cbn probabilistic λ-calculus, which has already fully abstract denotational models [2, 16] as well as infinitary normal forms [15] but not a theory of bisimulations.

References

1. Samson Abramsky. *Research Topics in Functional Programming*, chapter The Lazy Lambda Calculus, pages 65–116. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1990. URL: `http://dl.acm.org/citation.cfm?id=119830.119834`.
2. Pierre Clairambault and Hugo Paquet. Fully abstract models of the probabilistic lambda-calculus. In Dan R. Ghica and Achim Jung, editors, *27th EACSL Annual Conference on Computer Science Logic, CSL 2018, September 4-7, 2018, Birmingham, UK*, volume 119 of *LIPIcs*, pages 16:1–16:17. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018. `doi:10.4230/LIPIcs.CSL.2018.16`.
3. Robin Cooper, Simona Dobnik, Shalom Lappin, and Staffan Larsson. A probabilistic rich type theory for semantic interpretation. In *Proceedings of the EACL 2014 Workshop on Type Theory and Natural Language Semantics (TTNLS)*, pages 72–79, 2014.
4. Raphaëlle Crubillé and Ugo Dal Lago. On probabilistic applicative bisimulation and call-by-value λ-calculi. In *ESOP*, volume 8410 of *Lecture Notes in Computer Science*, pages 209–228. Springer, 2014.
5. Raphaëlle Crubillé, Ugo Dal Lago, Davide Sangiorgi, and Valeria Vignudelli. On applicative similarity, sequentiality, and full abstraction. In *Correct System Design*, volume 9360 of *Lecture Notes in Computer Science*, pages 65–82. Springer, 2015.
6. Ugo Dal Lago, Davide Sangiorgi, and Michele Alberti. On coinductive equivalences for higher-order probabilistic functional programs (long version). *CoRR*, abs/1311.1722, 2013.
7. Ugo Dal Lago, Davide Sangiorgi, and Michele Alberti. On coinductive equivalences for higher-order probabilistic functional programs. In *POPL*, pages 297–308. ACM, 2014.
8. Ugo Dal Lago and Margherita Zorzi. Probabilistic operational semantics for the lambda calculus. *RAIRO - Theoretical Informatics and Applications - Informatique Théorique et Applications*, 46(3):413–450, 2012. URL: `http://www.numdam.org/item/ITA_2012__46_3_413_0`, `doi:10.1051/ita/2012012`.
9. Thomas Ehrhard, Michele Pagani, and Christine Tasson. The computational meaning of probabilistic coherence spaces. In *LICS*, pages 87–96. IEEE Computer Society, 2011.
10. Thomas Ehrhard, Michele Pagani, and Christine Tasson. Full abstraction for probabilistic PCF. *J. ACM*, 65(4):23:1–23:44, 2018.

11 Silvia Ghilezan, Jelena Ivetic, Simona Kasterovic, Zoran Ognjanovic, and Nenad Savic. Probabilistic reasoning about simply typed lambda terms. In *LFCS*, volume 10703 of *Lecture Notes in Computer Science*, pages 170–189. Springer, 2018.

12 Jean-Yves Girard. Linear logic. *Theor. Comput. Sci.*, 50:1–102, 1987.

13 Douglas J. Howe. Proving congruence of bisimulation in functional programming languages. *Inf. Comput.*, 124(2):103–112, 1996. URL: `https://doi.org/10.1006/inco.1996.0008`, `doi:10.1006/inco.1996.0008`.

14 Kim Guldstrand Larsen and Arne Skou. Bisimulation through probabilistic testing. *Inf. Comput.*, 94(1):1–28, 1991.

15 Thomas Leventis. Probabilistic böhm trees and probabilistic separation. In *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*, pages 649–658, 2018. `doi:10.1145/3209108.3209126`.

16 Thomas Leventis and Michele Pagani. Strong adequacy and untyped full abstraction for probabilistic coherence spaces. accepted to FOSSACS 2019, 2019.

17 Sungwoo Park, Frank Pfenning, and Sebastian Thrun. A probabilistic language based on sampling functions. *ACM Trans. Program. Lang. Syst.*, 31(1):4:1–4:46, December 2008. URL: `http://doi.acm.org/10.1145/1452044.1452048`, `doi:10.1145/1452044.1452048`.

18 Gordon D. Plotkin. LCF considered as a programming language. *Theor. Comput. Sci.*, 5(3):223–255, 1977.

19 Alex Simpson and Niels F. W. Voorneveld. Behavioural equivalence via modalities for algebraic effects. In *ESOP*, volume 10801 of *Lecture Notes in Computer Science*, pages 300–326. Springer, 2018.

20 Franck Van Breugel, Michael W. Mislove, Joël Ouaknine, and James Worrell. Domain theory, testing and simulation for labelled markov processes. *Theor. Comput. Sci.*, 333(1-2):171–197, 2005.

A Appendix - Proofs

A.1 Context Equivalence is a congruence

We consider $\Lambda_{\oplus,\mathsf{let}}$-relations defined in Section 3. The set $P_{\mathsf{FIN}}(X)$ denotes the set of all finite subsets of X.

▶ **Definition 26.** *A $\Lambda_{\oplus,\mathsf{let}}$-relation $\mathcal{R}$ is compatible if and only if the five conditions below hold:*

(Com1) $\forall\Gamma \in P_{\mathsf{FIN}}(X), x \in \Gamma : \Gamma \vdash x \mathcal{R} x$;

(Com2) $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X - \Gamma, \forall M,N \in \Lambda^{\Gamma\cup\{x\}}_{\oplus,\mathsf{let}} : \Gamma \cup \{x\} \vdash M \mathcal{R} N \Rightarrow \Gamma \vdash \lambda x.M \mathcal{R} \lambda x.N$;

(Com3) $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall M,N,L,P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}} : \Gamma \vdash M \mathcal{R} N \wedge \Gamma \vdash L \mathcal{R} P \Rightarrow \Gamma \vdash ML \mathcal{R} NP$;

(Com4) $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall M,N,L,P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}} : \Gamma \vdash M \mathcal{R} N \wedge \Gamma \vdash L \mathcal{R} P \Rightarrow \Gamma \vdash M \oplus L \mathcal{R} N \oplus P$;

(Com5) $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X, \forall M,N \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}, \forall L,P \in \Lambda^{\Gamma\cup\{x\}}_{\oplus,\mathsf{let}} : \Gamma \vdash M \mathcal{R} N \wedge \Gamma\cup\{x\} \vdash L \mathcal{R} P \Rightarrow \Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L) \mathcal{R} (\mathsf{let}\ x = N\ \mathsf{in}\ P)$.

The following lemmas give us an easier way to establish (Com3), (Com4) and (Com5) under particular assumptions.

▶ **Lemma 27.** *Let us consider the properties*
(Com3L) $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall M,N,L \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}} : \Gamma \vdash M \mathcal{R} N \Rightarrow \Gamma \vdash ML \mathcal{R} NL$
(Com3R) $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall M,N,L \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}} : \Gamma \vdash M \mathcal{R} N \Rightarrow \Gamma \vdash LM \mathcal{R} LN$
If $\mathcal{R}$ is transitive, then (Com3L) *and* (Com3R) *together imply* (Com3).

▶ **Lemma 28.** *Let us consider the properties*
(Com4L) $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall M,N,L \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}} : \Gamma \vdash M \mathcal{R} N \Rightarrow \Gamma \vdash M \oplus L \mathcal{R} N \oplus L$
(Com4R) $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall M,N,L \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}} : \Gamma \vdash M \mathcal{R} N \Rightarrow \Gamma \vdash L \oplus M \mathcal{R} L \oplus N$
If $\mathcal{R}$ is transitive, then (Com4L) *and* (Com4R) *together imply* (Com4).

▶ **Lemma 29.** *Let us consider the properties*
(Com5L) $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X, \forall M,N \in \Lambda_{\oplus,\mathsf{let}}(\Gamma), \forall L \in \Lambda^{\Gamma\cup\{x\}}_{\oplus,\mathsf{let}}, \Gamma \vdash M \mathcal{R} N \Rightarrow \Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L) \mathcal{R} (\mathsf{let}\ x = N\ \mathsf{in}\ L)$
(Com5R) $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X, \forall L \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}, \forall M,N \in \Lambda^{\Gamma\cup\{x\}}_{\oplus,\mathsf{let}}, \Gamma \cup \{x\} \vdash M \mathcal{R} N \Rightarrow \Gamma \vdash (\mathsf{let}\ x = L\ \mathsf{in}\ M) \mathcal{R} (\mathsf{let}\ x = L\ \mathsf{in}\ N)$
If $\mathcal{R}$ is transitive, then (Com5L) *and* (Com5R) *together imply* (Com5).

Proof. To prove (Com5) we have to show that the hypothesis $\Gamma \vdash M \mathcal{R} N$ and $\Gamma \cup \{x\} \vdash L \mathcal{R} P$ imply $\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L) \mathcal{R} (\mathsf{let}\ x = N\ \mathsf{in}\ P)$. If we apply (Com5L) to the first hypothesis, with L as steady term, we get $\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L) \mathcal{R} (\mathsf{let}\ x = N\ \mathsf{in}\ L)$. Similarly, applying (Com5R) to the second hypothesis, with N as steady term we obtain $\Gamma \vdash (\mathsf{let}\ x = N\ \mathsf{in}\ L) \mathcal{R} (\mathsf{let}\ x = N\ \mathsf{in}\ P)$. Then, by transitivity of $\mathcal{R}$ we can conclude the claim. ◀

▶ **Definition 30.** *A $\Lambda_{\oplus,\mathsf{let}}$-relation is a congruence (respectively, precongruence) if it is an equivalence relation (respectively, preorder) and compatible.*

This definition of a (pre)congruence is equivalent to Definition 16.

▶ **Lemma 31.** *The context preorder $\leq$ is a precongruence relation.*

Proof. In order to prove $\leq$ is a precongruence, we need to show that $\leq$ is a preorder (reflexive and transitive) relation, which is compatible. Relation $\leq$ is reflexive by its definition and proving its transitivity means to show: $\forall\Gamma \in P_{\mathsf{FIN}}(X), M, N, L \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$,

$$\Gamma \vdash M \leq N \wedge \Gamma \vdash N \leq L \Rightarrow \Gamma \vdash M \leq L.$$

Let assume that $\Gamma \vdash M \leq N$ and $\Gamma \vdash N \leq L$, then we have the following hypothesis:
(1) $\forall C \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}, \sum[\![C[M]]\!] \leq \sum[\![C[N]]\!]$;
(2) $\forall C \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}, \sum[\![C[N]]\!] \leq \sum[\![C[L]]\!]$.
To prove $\Gamma \vdash M \leq L$ we need to show that for every $D \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}$, $\sum[\![D[M]]\!] \leq \sum[\![D[L]]\!]$. For any such context D, from the hypothesis (1) and (2) we have $\sum[\![D[M]]\!] \leq \sum[\![D[N]]\!] \leq \sum[\![D[L]]\!]$. In order to prove that $\leq$ is compatible, we show it satisfies conditions (Com1), (Com2), (Com3), (Com4) and (Com5). We do not consider (Com1), since it is trivial.

- Proving (Com2) means to show $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X - \Gamma, \forall M, N \in \Lambda^{\Gamma\cup\{x\}}_{\oplus,\mathsf{let}}$,

  $$\Gamma \cup \{x\} \vdash M \leq N \Rightarrow \Gamma \vdash \lambda x.M \leq \lambda x.N.$$

  From the assumption $\Gamma \cup \{x\} \vdash M \leq N$, we have $\forall C \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma\cup\{x\};\emptyset)}, \sum[\![C[M]]\!] \leq \sum[\![C[N]]\!]$ as hypothesis. Let us consider a context $D \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma,\emptyset)}$. Since context $\lambda x.[\cdot]$ belongs to the set $\mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\{x\};\Gamma)}$ we have that $E = D[\lambda x.[\cdot]] \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma\cup\{x\};\emptyset)}$. We can apply the hypothesis for context E and obtain $\sum[\![E[M]]\!] \leq \sum[\![E[N]]\!]$, i.e. $\sum[\![D[\lambda x.M]]\!] \leq \sum[\![D[\lambda x.N]]\!]$. Thus, $\Gamma \vdash \lambda x.M \leq \lambda x.N$.
- As we already proved, $\leq$ is transitive relation, thus by Lemma 27 it is enough to prove two characterizations (Com3L) and (Com3R). Proving (Com3L) means to show $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall M, N, L \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$,

  $$\Gamma \vdash M \leq N \Rightarrow \Gamma \vdash ML \leq NL.$$

  If we assume $\Gamma \vdash M \leq N$, then we have $\forall C \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}, \sum[\![C[M]]\!] \leq \sum[\![C[N]]\!]$ as hypothesis. We want to show that for any context $D \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}$, $\sum[\![D[ML]]\!] \leq \sum[\![D[NL]]\!]$ holds. For an arbitrary context $D \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}$ and $[\cdot]L \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\emptyset;\Gamma)}$ we get $E = D[[\cdot]L] \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}$. From the hypothesis, we can conclude that $\sum[\![E[M]]\!] \leq \sum[\![E[N]]\!]$ holds, i.e. $\sum[\![D[ML]]\!] \leq \sum[\![D[NL]]\!]$. Thus, $\Gamma \vdash ML \leq NL$. We do not write a detailed proof of (Com3R) because it is analogous to the proof of (Com3L).
- As in the previous case, the fact that $\leq$ is transitive and Lemma 28 ensure that (Com4L) and (Com4R) imply (Com4), so it is enough to prove these two characterizations. We omit the proof of (Com4L) and (Com4R), since we prove it by a similar reasoning as in the previous case (proof of (Com3L)).
- As we already proved, $\leq$ is transitive relation, thus by Lemma 29 it is enough to prove two characterizations (Com5L) and (Com5R). Proving (Com5L) means to show $\forall\Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X, \forall M, N \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}, \forall L \in \Lambda^{\Gamma\cup\{x\}}_{\oplus,\mathsf{let}}$,

  $$\Gamma \vdash M \leq N \Rightarrow \Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L) \leq (\mathsf{let}\ x = N\ \mathsf{in}\ L).$$

  If we assume $\Gamma \vdash M \leq N$, then we have $\forall C \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}, \sum[\![C[M]]\!] \leq \sum[\![C[N]]\!]$ as hypothesis. We want to show that for any context $D \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}$, $\sum[\![D[\mathsf{let}\ x = M\ \mathsf{in}\ L]]\!] \leq \sum[\![D[\mathsf{let}\ x = N\ \mathsf{in}\ L]]\!]$ holds. For an arbitrary context $D \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}$ and $\mathsf{let}\ x = [\cdot]\ \mathsf{in}\ L \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\emptyset;\Gamma)}$ we have that $E = D[\mathsf{let}\ x = [\cdot]\ \mathsf{in}\ L] \in \mathsf{C}\Lambda_{\oplus,\mathsf{let}}{}^{(\Gamma;\emptyset)}$. From the hypothesis, we can conclude that $\sum[\![E[M]]\!] \leq \sum[\![E[N]]\!]$ holds, i.e. $\sum[\![D[\mathsf{let}\ x = M\ \mathsf{in}\ L]]\!] \leq \sum[\![D[\mathsf{let}\ x = N\ \mathsf{in}\ L]]\!]$. Thus, $\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L) \leq (\mathsf{let}\ x = N\ \mathsf{in}\ L)$. The characterization (Com5R) can be proved in a similar way.

◀

▶ **Lemma 32.** *The context equivalence $\simeq$ is a congruence relation.*

Proof. This statement follows directly from Lemma 31 and the definition of context equivalence, i.e. $\simeq = \leq \cap (\leq)^{op}$. ◀

A.2 Bisimulation Equivalence is a congruence

We use Howe's technique to prove that probabilistic similarity is a precongruence and as a consequence probabilistic bisimilarity is a congruence. Howe's technique is a commonly used technique for proving (pre)congruence of bisimilarity (similarity). The proof is very technical. It is the adaptation of the technique used in [4, 5, 6] and it has the same structure as the proof in [6]. Contrary to the proof in [6], our proof introduces a new notion of compatibility with the let-in operator.

The property $\sim = \lesssim \cap \lesssim^{op}$ ensures it is enough to show that probabilistic similarity ($\lesssim$) is a precongruence in order to prove that probabilistic bisimilarity ($\sim$) is a congruence. The key part is proving that $\lesssim$ is a compatible relation and it is done by Howe's technique.

We call an $\Lambda_{\oplus,\mathsf{let}}$-relation $\mathcal{R}$ *(term) substitutive* if for all $\Gamma \in P_{\mathsf{FIN}}(X), x \in X - \Gamma, M, N \in \Lambda_{\oplus,\mathsf{let}}^{\Gamma\cup\{\mathsf{x}\}}, L, P \in \Lambda_{\oplus,\mathsf{let}}^{\Gamma}$ the following holds

$$\Gamma \cup \{x\} \vdash M\ \mathcal{R}\ N \wedge \Gamma \vdash L\ \mathcal{R}\ P \Rightarrow \Gamma \vdash M\{L/x\}\ \mathcal{R}\ N\{P/x\}.$$

If a relation $\mathcal{R}$ satisfies

$$\Gamma \cup \{x\} \vdash M\ \mathcal{R}\ N \wedge L \in \Lambda_{\oplus,\mathsf{let}}^{\Gamma} \Rightarrow \Gamma \vdash M\{L/x\}\ \mathcal{R}\ N\{L/x\},$$

we say it is *closed under term-substitution.*

Please notice that if $\mathcal{R}$ is substitutive and reflexive then it is closed under term-substitution. As stated in the paper, open extensions of $\lesssim$ and $\sim$ are closed under term-substitution by definition.

For an arbitrary $\Lambda_{\oplus,\mathsf{let}}$-relation $\mathcal{R}$, Howe's lifting $\mathcal{R}^H$ is defined by the rules in Figure 5. We start with some auxiliary statements.

▶ **Lemma 33.** *If $\mathcal{R}$ is reflexive, then $\mathcal{R}^H$ is compatible.*

Proof. We prove that (Com1), (Com2), (Com3), (Com4) and (Com5) hold for $\mathcal{R}^H$, if $\mathcal{R}$ is a reflexive relation.

- To prove (Com1) we need to show that:

  $$\forall \Gamma \in P_{\mathsf{FIN}}(X), x \in \Gamma : \Gamma \vdash x\ \mathcal{R}^H\ x.$$

  Since $\mathcal{R}$ is reflexive, we have that $\forall \Gamma \in P_{\mathsf{FIN}}(X), x \in \Gamma : \Gamma \vdash x\ \mathcal{R}\ x$. If we apply (How1) to $\Gamma \vdash x\ \mathcal{R}\ x$, we obtain $\Gamma \vdash x\ \mathcal{R}^H\ x$.
- In order to prove (Com2) we need to show that: $\forall \Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X - \Gamma, \forall M, N \in \Lambda_{\oplus,\mathsf{let}}^{\Gamma\cup\{\mathsf{x}\}}$,

  $$\Gamma \cup \{x\} \vdash M\ \mathcal{R}^H\ N \Rightarrow \Gamma \vdash \lambda x.M\ \mathcal{R}^H\ \lambda x.N$$

  Using the reflexivity of $\mathcal{R}$, we obtain $\Gamma \vdash \lambda x.N\ \mathcal{R}\ \lambda x.N$. We have $\Gamma \cup \{x\} \vdash M\ \mathcal{R}^H\ N$ by hypothesis, so we can apply (How2) and conclude $\Gamma \vdash \lambda x.M\ \mathcal{R}^H\ \lambda x.N$ holds.

$$\frac{\Gamma \vdash x \mathrel{\mathcal{R}} M}{\Gamma \vdash x \mathrel{\mathcal{R}^H} M} \text{(How1)} \qquad \frac{\Gamma \cup \{x\} \vdash M \mathrel{\mathcal{R}^H} L \quad \Gamma \vdash \lambda x.L \mathrel{\mathcal{R}} N \quad x \notin \Gamma}{\Gamma \vdash \lambda x.M \mathrel{\mathcal{R}^H} N} \text{(How2)}$$

$$\frac{\Gamma \vdash M \mathrel{\mathcal{R}^H} P \quad \Gamma \vdash N \mathrel{\mathcal{R}^H} Q \quad \Gamma \vdash PQ \mathrel{\mathcal{R}} L}{\Gamma \vdash MN \mathrel{\mathcal{R}^H} L} \text{(How3)}$$

$$\frac{\Gamma \vdash M \mathrel{\mathcal{R}^H} P \quad \Gamma \vdash N \mathrel{\mathcal{R}^H} Q \quad \Gamma \vdash P \oplus Q \mathrel{\mathcal{R}} L}{\Gamma \vdash M \oplus N \mathrel{\mathcal{R}^H} L} \text{(How4)}$$

$$\frac{\Gamma \vdash M \mathrel{\mathcal{R}^H} P \quad \Gamma \cup \{x\} \vdash N \mathrel{\mathcal{R}^H} Q \quad \Gamma \vdash (\mathsf{let}\ x = P\ \mathsf{in}\ Q) \mathrel{\mathcal{R}} L}{\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ N) \mathrel{\mathcal{R}^H} L} \text{(How5)}$$

Figure 5 Howe's lifting for $\Lambda_{\oplus,\mathsf{let}}$

- Proving (Com3) means to show: $\forall \Gamma \in P_{\mathsf{FIN}}(X), \forall M, N, L, P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$,

  $$\Gamma \vdash M \mathrel{\mathcal{R}^H} N \wedge \Gamma \vdash L \mathrel{\mathcal{R}^H} P \Rightarrow \Gamma \vdash ML \mathrel{\mathcal{R}^H} NP.$$

  Since the relation $\mathcal{R}$ is reflexive, we have that $\Gamma \vdash NP \mathrel{\mathcal{R}} NP$ holds. Moreover, $\Gamma \vdash M \mathrel{\mathcal{R}^H} N$ and $\Gamma \vdash L \mathrel{\mathcal{R}^H} P$ hold by hypothesis. Therefore, by (How3), we conclude $\Gamma \vdash ML \mathrel{\mathcal{R}^H} NP$ holds.
- To prove (Com4) we have to show: $\forall \Gamma \in P_{\mathsf{FIN}}(X), \forall M, N, L, P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$,

  $$\Gamma \vdash M \mathrel{\mathcal{R}^H} N \wedge \Gamma \vdash L \mathrel{\mathcal{R}^H} P \Rightarrow \Gamma \vdash M \oplus L \mathrel{\mathcal{R}^H} N \oplus P.$$

  We have that $\Gamma \vdash N \oplus P \mathrel{\mathcal{R}} N \oplus P$ holds, because of the reflexivity of $\mathcal{R}$. Furthermore, $\Gamma \vdash M \mathrel{\mathcal{R}^H} N$ and $\Gamma \vdash L \mathrel{\mathcal{R}^H} P$ hold by hypothesis. Now, by (How4) we obtain that $\Gamma \vdash M \oplus L \mathrel{\mathcal{R}^H} N \oplus P$ holds.
- In order to prove (Com5) we need to show: $\forall \Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X, \forall M, N \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}, \forall L, P \in \Lambda^{\Gamma \cup \{x\}}_{\oplus,\mathsf{let}}$,

  $$\Gamma \vdash M \mathrel{\mathcal{R}^H} N \wedge \Gamma \cup \{x\} \vdash L \mathrel{\mathcal{R}^H} P \Rightarrow \Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L) \mathrel{\mathcal{R}^H} (\mathsf{let}\ x = N\ \mathsf{in}\ P).$$

  Since $\mathcal{R}$ is reflexive, we have $\Gamma \vdash (\mathsf{let}\ x = N\ \mathsf{in}\ P)\ R\ (\mathsf{let}\ x = N\ \mathsf{in}\ P)$. The hypothesis is that $\Gamma \vdash M \mathrel{\mathcal{R}^H} N$ and $\Gamma \cup \{x\} \vdash L \mathrel{\mathcal{R}^H} P$ hold. By applying (How5), we obtain $\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L) \mathrel{\mathcal{R}^H} (\mathsf{let}\ x = N\ \mathsf{in}\ P)$.

This concludes the proof. ◀

▶ **Lemma 34.** *If $\mathcal{R}$ is transitive, then $\Gamma \vdash M \mathrel{\mathcal{R}^H} N$ and $\Gamma \vdash N \mathrel{\mathcal{R}} L$ imply $\Gamma \vdash M \mathrel{\mathcal{R}^H} L$.*

Proof. We prove this by induction on the derivation of $\Gamma \vdash M \mathrel{\mathcal{R}^H} N$, looking at the last rule used, thus on the structure of M.

- Let M be a variable $x \in \Gamma$, then $\Gamma \vdash x \mathrel{\mathcal{R}^H} N$ holds by hypothesis. The last rule used has to be (How1). Hence, we have $\Gamma \vdash x \mathrel{\mathcal{R}} N$ as additional hypothesis. Since $\mathcal{R}$ is transitive, from $\Gamma \vdash x \mathrel{\mathcal{R}} N$ and $\Gamma \vdash N \mathrel{\mathcal{R}} L$ we can conclude $\Gamma \vdash x \mathrel{\mathcal{R}} L$. Now, by applying (How1) to the latter, we obtain $\Gamma \vdash x \mathrel{\mathcal{R}^H} L$, i.e. $\Gamma \vdash M \mathrel{\mathcal{R}^H} L$.
- Let M be an abstraction, say $\lambda x.Q$, then $\Gamma \vdash \lambda x.Q \mathrel{\mathcal{R}^H} N$ holds by hypothesis. The last rule used has to be (How2). Hence, we have $\Gamma \cup \{x\} \vdash Q \mathrel{\mathcal{R}^H} P$ and $\Gamma \vdash \lambda x.P \mathrel{\mathcal{R}} N$ as additional hypothesis. Since $\mathcal{R}$ is transitive, from $\Gamma \vdash \lambda x.P \mathrel{\mathcal{R}} N$ and $\Gamma \vdash N \mathrel{\mathcal{R}} L$ we can conclude $\Gamma \vdash \lambda x.P \mathrel{\mathcal{R}} L$. Now, by applying (How2) to $\Gamma \vdash Q \mathrel{\mathcal{R}^H} P$ and the latter, we obtain $\Gamma \vdash \lambda x.Q \mathrel{\mathcal{R}^H} L$, i.e. $\Gamma \vdash M \mathrel{\mathcal{R}^H} L$.

- Let M be an application, say RS, then $\Gamma \vdash RS\ \mathcal{R}^H\ N$ holds by hypothesis. The last rule used has to be (How3). Hence, we have $\Gamma \vdash R\ \mathcal{R}^H\ P$, $\Gamma \vdash S\ \mathcal{R}^H\ Q$ and $\Gamma \vdash PQ\ \mathcal{R}\ N$ as additional hypothesis. Since $\mathcal{R}$ is transitive, from $\Gamma \vdash PQ\ \mathcal{R}\ N$ and $\Gamma \vdash N\ \mathcal{R}\ L$ we can conclude $\Gamma \vdash PQ\ \mathcal{R}\ L$. Now, by applying (How3) to $\Gamma \vdash R\ \mathcal{R}^H\ P$, $\Gamma \vdash S\ \mathcal{R}^H\ Q$ and the latter, we obtain $\Gamma \vdash RS\ \mathcal{R}^H\ L$, i.e. $\Gamma \vdash M\ \mathcal{R}^H\ L$.
- Let M be a probabilistic sum, say $R \oplus S$, then $\Gamma \vdash R \oplus S\ \mathcal{R}^H\ N$ holds by hypothesis. The last rule used has to be (How4). Hence, we have $\Gamma \vdash R\ \mathcal{R}^H\ P$, $\Gamma \vdash S\ \mathcal{R}^H\ Q$ and $\Gamma \vdash P \oplus Q\ \mathcal{R}\ N$ as additional hypothesis. Since $\mathcal{R}$ is transitive, from $\Gamma \vdash P \oplus Q\ \mathcal{R}\ N$ and $\Gamma \vdash N\ \mathcal{R}\ L$ we can conclude $\Gamma \vdash P \oplus Q\ \mathcal{R}\ L$. Now, by applying (How4) to $\Gamma \vdash R\ \mathcal{R}^H\ P$, $\Gamma \vdash S\ \mathcal{R}^H\ Q$ and the latter, we obtain $\Gamma \vdash R \oplus S\ \mathcal{R}^H\ L$, i.e. $\Gamma \vdash M\ \mathcal{R}^H\ L$.
- Let M be a term $\mathsf{let}\ x = R\ \mathsf{in}\ S$, then $\Gamma \vdash (\mathsf{let}\ x = R\ \mathsf{in}\ S)\ \mathcal{R}^H\ N$ holds by hypothesis. The last rule used has to be (How5). Hence, we have $\Gamma \vdash R\ \mathcal{R}^H\ P$, $\Gamma \cup \{x\} \vdash S\ \mathcal{R}^H\ Q$ and $\Gamma \vdash (\mathsf{let}\ x = P\ \mathsf{in}\ Q)\ \mathcal{R}\ N$ as additional hypothesis. Since $\mathcal{R}$ is transitive, from $\Gamma \vdash (\mathsf{let}\ x = P\ \mathsf{in}\ Q)\ \mathcal{R}\ N$ and $\Gamma \vdash N\ \mathcal{R}\ L$ we can conclude $\Gamma \vdash (\mathsf{let}\ x = P\ \mathsf{in}\ Q)\ \mathcal{R}\ L$. Now, by applying (How5) to $\Gamma \vdash R\ \mathcal{R}^H\ P$, $\Gamma \cup \{x\} \vdash S\ \mathcal{R}^H\ Q$ and the latter, we obtain $\Gamma \vdash (\mathsf{let}\ x = R\ \mathsf{in}\ S)\ \mathcal{R}^H\ L$, i.e. $\Gamma \vdash M\ \mathcal{R}^H\ L$.

This concludes the proof. ◀

▶ **Lemma 35.** *If $\mathcal{R}$ is reflexive, then $\Gamma \vdash M\ \mathcal{R}\ N$ implies $\Gamma \vdash M\ \mathcal{R}^H\ N$.*

Proof. We prove the statement by inspection on the last rule used in the derivation of $\Gamma \vdash M\ \mathcal{R}\ N$, that is on the structure of M.

- First, we consider the case where M is a variable $x \in \Gamma$, then $\Gamma \vdash x\ \mathcal{R}\ N$ holds by hypothesis. We can apply (How1) to this and obtain $\Gamma \vdash x\ \mathcal{R}^H\ N$, i.e. $\Gamma \vdash M\ \mathcal{R}^H\ N$.
- Next, we consider the case where M is an abstraction, say $\lambda x.Q$, then $\Gamma \vdash \lambda x.Q\ \mathcal{R}\ N$ holds by hypothesis. Since $\mathcal{R}$ is reflexive, we have that $\mathcal{R}^H$ is compatible and it is easy to prove that $\mathcal{R}^H$ is also reflexive. Hence, we have that $\Gamma \cup \{x\} \vdash Q\ \mathcal{R}^H\ Q$ holds. If we apply (How2) to the latter and $\Gamma \vdash \lambda x.Q\ \mathcal{R}\ N$ we conclude $\Gamma \vdash \lambda x.Q\ \mathcal{R}^H\ N$ holds, i.e. $\Gamma \vdash M\ \mathcal{R}^H\ N$.
- Let us now look at the case where M is an application, say RS, then $\Gamma \vdash RS\ \mathcal{R}\ N$ holds by hypothesis. Since $\mathcal{R}$ is reflexive, $\mathcal{R}^H$ is also reflexive and we have that $\Gamma \vdash R\ \mathcal{R}^H\ R$ and $\Gamma \vdash S\ \mathcal{R}^H\ S$ hold. If we apply (How3) to the latter and $\Gamma \vdash RS\ \mathcal{R}\ N$ we conclude $\Gamma \vdash RS\ \mathcal{R}^H\ N$ holds, i.e. $\Gamma \vdash M\ \mathcal{R}^H\ N$.
- If M is a probabilistic sum, say $R \oplus S$, then $\Gamma \vdash R \oplus S\ \mathcal{R}\ N$ holds by hypothesis. Since $\mathcal{R}$ is reflexive, $\mathcal{R}^H$ is also reflexive and we have that $\Gamma \vdash R\ \mathcal{R}^H\ R$ and $\Gamma \vdash S\ \mathcal{R}^H\ S$ hold. If we apply (How4) to the latter and $\Gamma \vdash R \oplus S\ \mathcal{R}\ N$ we conclude $\Gamma \vdash R \oplus S\ \mathcal{R}^H\ N$ holds, i.e. $\Gamma \vdash M\ \mathcal{R}^H\ N$.
- Finally, we consider the case where M is a term $\mathsf{let}\ R\ \mathsf{in}\ S$, then $\Gamma \vdash (\mathsf{let}\ R\ \mathsf{in}\ S)\ \mathcal{R}\ N$ holds by hypothesis. Since $\mathcal{R}$ is reflexive, $\mathcal{R}^H$ is also reflexive and we have that $\Gamma \vdash R\ \mathcal{R}^H\ R$ and $\Gamma \vdash S\ \mathcal{R}^H\ S$ hold. If we apply (How5) to the latter and $\Gamma \vdash (\mathsf{let}\ R\ \mathsf{in}\ S)\ \mathcal{R}\ N$ we conclude $\Gamma \vdash (\mathsf{let}\ R\ \mathsf{in}\ S)\ \mathcal{R}^H\ N$ holds, i.e. $\Gamma \vdash M\ \mathcal{R}^H\ N$.

  This concludes the proof.

◀

▶ **Lemma 36.** *If $\mathcal{R}$ is reflexive, transitive and closed under term-substitution, then $\mathcal{R}^H$ is (term) substitutive and hence also closed under term-substitution.*

Proof. We need to show that: $\forall \Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X - \Gamma, \forall M, N \in \Lambda_{\oplus,\mathsf{let}}^{\Gamma\cup\{x\}}, \forall L, P \in \Lambda_{\oplus,\mathsf{let}}^{\Gamma}$,

$$\Gamma \cup \{x\} \vdash M\ \mathcal{R}^H\ N\ \wedge\ \Gamma \vdash L\ \mathcal{R}^H\ P\ \Rightarrow \Gamma \vdash M\{L/x\}\ \mathcal{R}^H\ N\{L/x\}.$$

We prove it by induction on the derivation of $\Gamma \cup \{x\} \vdash M \mathrel{\mathcal{R}^H} N$, thus on the structure of M.

- Let us start with the case where M is a variable, then there are two possibilities: either $M = x$ or $M \in \Gamma$. Suppose that $M \in \Gamma$ and $M = y$. Now, we have that $\Gamma \cup \{x\} \vdash y \mathrel{\mathcal{R}^H} N$ holds by hypothesis and it can only be deduced by the rule (How1) from $\Gamma \cup \{x\} \vdash y \mathrel{\mathcal{R}} N$. Using the fact that $\mathcal{R}$ is closed under term-substitution and $P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$, we can conclude $\Gamma \vdash y\{P/x\} \mathrel{\mathcal{R}} N\{P/x\}$, which is equivalent to $\Gamma \vdash y \mathrel{\mathcal{R}} N\{P/x\}$. Next, by Lemma 35 we obtain $\Gamma \vdash y \mathrel{\mathcal{R}^H} N\{P/x\}$, which is equivalent to $\Gamma \vdash y\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$, i.e. $\Gamma \vdash M\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$. Let us now suppose that $M = x$, then $\Gamma \cup \{x\} \vdash x \mathrel{\mathcal{R}^H} N$ holds. The only way to deduce it is by the rule (How1) from $\Gamma \cup \{x\} \vdash x \mathrel{\mathcal{R}} N$. Since $\mathcal{R}$ is closed under term-substitution and $P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$, we conclude $\Gamma \vdash x\{P/x\} \mathrel{\mathcal{R}} N\{P/x\}$ which is equivalent to $\Gamma \vdash P \mathrel{\mathcal{R}} N\{P/x\}$. If we apply Lemma 34 to $\Gamma \vdash L \mathrel{\mathcal{R}^H} P$ and $\Gamma \vdash P \mathrel{\mathcal{R}} N\{P/x\}$, we deduce $\Gamma \vdash L \mathrel{\mathcal{R}^H} N\{P/x\}$ which is equivalent to $\Gamma \vdash x\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$. Hence, $\Gamma \vdash M\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$ holds.
- Next, we consider the case where M is an abstraction, say $\lambda y.Q$, then $\Gamma \cup \{x\} \vdash \lambda y.Q \mathrel{\mathcal{R}^H} N$ holds by hypothesis. It can only be deduced by the rule (How2) from $\Gamma \cup \{x\} \cup \{y\} \vdash Q \mathrel{\mathcal{R}^H} R$, and $\Gamma \cup \{x\} \vdash \lambda y.R \mathrel{\mathcal{R}} N$, where $x, y \notin \Gamma$. By applying the induction hypothesis to $\Gamma \cup \{x\} \cup \{y\} \vdash Q \mathrel{\mathcal{R}^H} R$, we conclude $\Gamma \cup \{y\} \vdash Q\{L/x\} \mathrel{\mathcal{R}} R\{P/x\}$. From the fact that $\mathcal{R}$ is closed under term-substitution and $P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$, we obtain $\Gamma \vdash (\lambda y.R)\{P/x\} \mathrel{\mathcal{R}} N\{P/x\}$, i.e. $\Gamma \vdash \lambda y.R\{P/x\} \mathrel{\mathcal{R}} N\{P/x\}$. By (How2), we deduce $\Gamma \vdash \lambda y.Q\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$, which is equivalent to $\Gamma \vdash (\lambda y.Q)\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$. Hence, $\Gamma \vdash M\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$.
- If M is an application, say RS, then $\Gamma \cup \{x\} \vdash RS \mathrel{\mathcal{R}^H} N$ holds by hypothesis. It can only be deduced by the rule (How3) from $\Gamma \cup \{x\} \vdash R \mathrel{\mathcal{R}^H} R'$, $\Gamma \cup \{x\} \vdash S \mathrel{\mathcal{R}^H} S'$ and $\Gamma \cup \{x\} \vdash R'S' \mathrel{\mathcal{R}} N$. By applying the induction hypothesis to $\Gamma \cup \{x\} \vdash R \mathrel{\mathcal{R}^H} R'$ and $\Gamma \cup \{x\} \vdash S \mathrel{\mathcal{R}^H} S'$, we conclude $\Gamma \vdash R\{L/x\} \mathrel{\mathcal{R}^H} R'\{P/x\}$ and $\Gamma \vdash S\{L/x\} \mathrel{\mathcal{R}^H} S'\{P/x\}$. From the fact that $\mathcal{R}$ is closed under term-substitution and $P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$, we obtain $\Gamma \vdash (R'S')\{P/x\} \mathrel{\mathcal{R}} N\{P/x\}$, i.e. $\Gamma \vdash R'\{P/x\}S'\{P/x\} \mathrel{\mathcal{R}} N\{P/x\}$. By (How3), we deduce $\Gamma \vdash R\{L/x\}S\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$, which is equivalent to $\Gamma \vdash (RS)\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$. Hence, $\Gamma \vdash M\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$.
- Let M be a probabilistic sum, say $R \oplus S$, then $\Gamma \cup \{x\} \vdash R \oplus S \mathrel{\mathcal{R}^H} N$ holds by hypothesis. It can only be deduced by the rule (How4) from $\Gamma \cup \{x\} \vdash R \mathrel{\mathcal{R}^H} R'$, $\Gamma \cup \{x\} \vdash S \mathrel{\mathcal{R}^H} S'$ and $\Gamma \cup \{x\} \vdash R' \oplus S' \mathrel{\mathcal{R}} N$. By applying the induction hypothesis to $\Gamma \cup \{x\} \vdash R \mathrel{\mathcal{R}^H} R'$ and $\Gamma \cup \{x\} \vdash S \mathrel{\mathcal{R}^H} S'$, we conclude $\Gamma \vdash R\{L/x\} \mathrel{\mathcal{R}^H} R'\{P/x\}$ and $\Gamma \vdash S\{L/x\} \mathrel{\mathcal{R}^H} S'\{P/x\}$. From the fact that $\mathcal{R}$ is closed under term-substitution and $P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$, we obtain $\Gamma \vdash (R' \oplus S')\{P/x\} \mathrel{\mathcal{R}} N\{P/x\}$, i.e. $\Gamma \vdash R'\{P/x\} \oplus S'\{P/x\} \mathrel{\mathcal{R}} N\{P/x\}$. By (How4), we deduce $\Gamma \vdash R\{L/x\} \oplus S\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$, which is equivalent to $\Gamma \vdash (R \oplus S)\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$. Hence, $\Gamma \vdash M\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$.
- Finally, we consider the case where M is a term $\mathsf{let}\ y = R\ \mathsf{in}\ S$, then $\Gamma \cup \{x\} \vdash (\mathsf{let}\ y = R\ \mathsf{in}\ S) \mathrel{\mathcal{R}^H} N$ holds by hypothesis. It can only be deduced by the rule (How5) from $\Gamma \cup \{x\} \vdash R \mathrel{\mathcal{R}^H} R'$, $\Gamma \cup \{x\} \cup \{y\} \vdash S \mathrel{\mathcal{R}^H} S'$ and $\Gamma \cup \{x\} \vdash (\mathsf{let}\ y = R'\ \mathsf{in}\ S') \mathrel{\mathcal{R}} N$. By applying the induction hypothesis to $\Gamma \cup \{x\} \vdash R \mathrel{\mathcal{R}^H} R'$ and $\Gamma \cup \{x\} \cup \{y\} \vdash S \mathrel{\mathcal{R}^H} S'$, we conclude $\Gamma \vdash R\{L/x\} \mathrel{\mathcal{R}} R'\{P/x\}$ and $\Gamma \cup \{y\} \vdash S\{L/x\} \mathrel{\mathcal{R}^H} S'\{P/x\}$. From the fact that $\mathcal{R}$ is closed under term-substitution and $P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$, we obtain $\Gamma \vdash (\mathsf{let}\ y = R'\ \mathsf{in}\ S')\{P/x\} \mathrel{\mathcal{R}} N\{P/x\}$, i.e. $\Gamma \vdash (\mathsf{let}\ y = R'\{P/x\}\ \mathsf{in}\ S'\{P/x\}) \mathrel{\mathcal{R}} N\{P/x\}$. By (How5), we deduce $\Gamma \vdash (\mathsf{let}\ y = R\{L/x\}\ \mathsf{in}\ S\{L/x\}) \mathrel{\mathcal{R}^H} N\{P/x\}$, which is equivalent to $\Gamma \vdash (\mathsf{let}\ y = R\ \mathsf{in}\ S)\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$. Hence, $\Gamma \vdash M\{L/x\} \mathrel{\mathcal{R}^H} N\{P/x\}$.

This concludes the proof. ◀

$$\frac{\Gamma \vdash M \mathcal{R} N}{\Gamma \vdash M \mathcal{R}^+ N} \text{(TC1)}$$

$$\frac{\Gamma \vdash M \mathcal{R}^+ N \qquad \Gamma \vdash N \mathcal{R}^+ L}{\Gamma \vdash M \mathcal{R}^+ L} \text{(TC2)}$$

Figure 6 Transitive closure for $\Lambda_{\oplus,\mathsf{let}}$

The goal is to prove that $\lesssim^H$ is a precongruence, but in order to do that some properties are missing. Hence, following Howe's approach we build a transitive closure of a $\Lambda_{\oplus,\mathsf{let}}$-relation $\mathcal{R}$ as a relation $\mathcal{R}^+$ defined by the rules in Figure 6.

▶ **Lemma 37.** *If $\mathcal{R}$ is compatible, then so is $\mathcal{R}^+$.*

Proof. We need to prove that relation $\mathcal{R}^+$ satisfies conditions: (Com1), (Com2), (Com3), (Com4) and (Com5).

- In order to prove (Com1) we have to show:

  $$\forall \Gamma \in P_{\mathsf{FIN}}(X), x \in \Gamma : \Gamma \vdash x \mathcal{R}^+ x.$$

  From the assumption that $\mathcal{R}$ is compatible, we can conclude that $\mathcal{R}$ is reflexive and $\Gamma \vdash x \mathcal{R} x$ holds. Now, $\Gamma \vdash x \mathcal{R}^+ x$ follows by (TC1).
- Proving (Com2) means to show that: $\forall \Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X - \Gamma, \forall M, N \in \Lambda_{\oplus,\mathsf{let}}^{\Gamma\cup\{x\}}$,

  $$\Gamma \cup \{x\} \vdash M \mathcal{R}^+ N \Rightarrow \Gamma \vdash \lambda x.M \mathcal{R}^+ \lambda x.N.$$

  We prove it by induction on the derivation of $\Gamma \cup \{x\} \vdash M \mathcal{R}^+ N$, looking at the last rule used. First we consider base case, where the last rule used is (TC1) and we have that $\Gamma \cup \{x\} \vdash M \mathcal{R} N$ holds by hypothesis. Using the fact that $\mathcal{R}$ is compatible, we can conclude $\Gamma \vdash \lambda x.M \mathcal{R} \lambda x.N$ holds. By applying (TC1) we obtain $\Gamma \vdash \lambda x.M \mathcal{R}^+ \lambda x.N$. Next, let us look at the case where the last rule used is (TC2). Now, we have that for some $L \in \Lambda_{\oplus,\mathsf{let}}^{\Gamma\cup\{x\}}$, $\Gamma \cup \{x\} \vdash M \mathcal{R}^+ L$ and $\Gamma \cup \{x\} \vdash L \mathcal{R}^+ N$ hold by hypothesis. We can apply the induction hypothesis on both of them and obtain $\Gamma \vdash \lambda x.M \mathcal{R}^+ \lambda x.L$ and $\Gamma \vdash \lambda x.L \mathcal{R}^+ \lambda x.N$. Finally, by applying (TC2) on the latter two, we conclude $\Gamma \vdash \lambda x.M \mathcal{R}^+ \lambda x.N$ holds.
- To prove (Com3) we need to show: $\forall \Gamma \in P_{\mathsf{FIN}}(X), \forall M, N, L, P \in \Lambda_{\oplus,\mathsf{let}}^{\Gamma}$,

  $$\Gamma \vdash M \mathcal{R}^+ N \ \wedge \ \Gamma \vdash L \mathcal{R}^+ P \Rightarrow \Gamma \vdash ML \mathcal{R}^+ NP.$$

  First, we will prove the following two statements:

  (1) $\forall M, N, L, P \in \Lambda_{\oplus,\mathsf{let}}^{\Gamma} : \Gamma \vdash M \mathcal{R}^+ N \ \wedge \ \Gamma \vdash L \mathcal{R} P \Rightarrow \Gamma \vdash ML \mathcal{R}^+ NP$,
  (2) $\forall M, N, L, P \in \Lambda_{\oplus,\mathsf{let}}^{\Gamma} : \Gamma \vdash M \mathcal{R} N \ \wedge \ \Gamma \vdash L \mathcal{R}^+ P \Rightarrow \Gamma \vdash ML \mathcal{R}^+ NP$.

  We prove (1) by induction on the derivation of $\Gamma \vdash M \mathcal{R}^+ N$, looking at the last rule used. First we consider the base case where (TC1) is the last rule used. Then we have that $\Gamma \vdash M \mathcal{R} N$ holds by hypothesis. Since we have assumed that $\mathcal{R}$ is compatible and $\Gamma \vdash L \mathcal{R} P$ holds, we can conclude $\Gamma \vdash ML \mathcal{R} NP$. Now, by applying (TC1) on the latter we obtain $\Gamma \vdash ML \mathcal{R}^+ NP$. Let us now consider the case where (TC2) is the last rule used. In this case we have $\Gamma \vdash M \mathcal{R}^+ Q$ and $\Gamma \vdash Q \mathcal{R}^+ N$ as additional hypothesis, for some $Q \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$. Now, by induction hypothesis on $\Gamma \vdash M\ \mathcal{R}^+\ Q$ and $\Gamma \vdash L\ \mathcal{R}\ P$ we have $\Gamma \vdash ML\ \mathcal{R}^+\ QP$. Using the fact that relation $\mathcal{R}$ is compatible, we can conclude its reflexivity and $\Gamma \vdash P\ \mathcal{R}\ P$ holds. Next, by induction hypothesis on $\Gamma \vdash Q\ \mathcal{R}^+\ N$ and $\Gamma \vdash P\ \mathcal{R}\ P$ we get $\Gamma \vdash QP\ \mathcal{R}^+\ NP$. Finally, we conclude applying (TC2) on $\Gamma \vdash ML\ \mathcal{R}^+\ QP$ and the latter, obtaining $\Gamma \vdash ML\ \mathcal{R}^+\ NP$. Statement (2) can be proved similarly.

Let consider the original statement (Com3). We prove it by induction on two derivations $\Gamma \vdash M\ \mathcal{R}^+\ N$ and $\Gamma \vdash L\ \mathcal{R}^+\ P$. If we look at the last rules used, we have four possible cases:

1. (TC1) is the last used rule in both derivations;
2. the last rule used in the derivation of $\Gamma \vdash M\ \mathcal{R}^+\ N$ is (TC1), and the last rule used in the derivation of $\Gamma \vdash L\ \mathcal{R}^+\ P$ is (TC2);
3. the last rule used in the derivation of $\Gamma \vdash M\ \mathcal{R}^+\ N$ is (TC2), and the last rule used in the derivation of $\Gamma \vdash L\ \mathcal{R}^+\ P$ is (TC1);
4. (TC2) is the last used rule in both derivations.

The first case follows from the fact that relation $\mathcal{R}$ is compatible, and second and third cases follow from the statements (1) and (2) we proved. Thus, we only consider the case where both derivations are concluded by applying the rule (TC2). In this case, as additional hypothesis we get that: for some $Q \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$, $\Gamma \vdash M\ \mathcal{R}^+\ Q$ and $\Gamma \vdash Q\ \mathcal{R}^+\ N$ hold, and for some $R \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$, $\Gamma \vdash L\ \mathcal{R}^+\ R$ and $\Gamma \vdash R\ \mathcal{R}^+\ P$ hold. First by induction hypothesis on $\Gamma \vdash M\ \mathcal{R}^+\ Q$ and $\Gamma \vdash L\ \mathcal{R}^+\ R$ we get $\Gamma \vdash ML\ \mathcal{R}^+\ QR$. Next, by induction hypothesis on $\Gamma \vdash Q\ \mathcal{R}^+\ N$ and $\Gamma \vdash R\ \mathcal{R}^+\ P$ we have $\Gamma \vdash QR\ \mathcal{R}^+\ NP$. Now we can apply (TC2) and obtain $\Gamma \vdash ML\ \mathcal{R}^+\ NP$.

- Proving (Com4) means to show: $\forall \Gamma \in P_{\mathsf{FIN}}(X), \forall M, N, L, P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$,

$$\Gamma \vdash M\ \mathcal{R}^+\ N\ \wedge\ \Gamma \vdash L\ \mathcal{R}^+\ P \Rightarrow \Gamma \vdash M \oplus L\ \mathcal{R}^+\ N \oplus P.$$

We do not write a detailed proof, since it is analogous to the previous case. The idea is to prove the following two statements:

(3) $\forall M, N, L, P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}} : \Gamma \vdash M\ \mathcal{R}^+\ N\ \wedge\ \Gamma \vdash L\ \mathcal{R}\ P \Rightarrow \Gamma \vdash M \oplus L\ \mathcal{R}^+\ N \oplus P$,
(4) $\forall M, N, L, P \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}} : \Gamma \vdash M\ \mathcal{R}\ N\ \wedge\ \Gamma \vdash L\ \mathcal{R}^+\ P \Rightarrow \Gamma \vdash M \oplus L\ \mathcal{R}^+\ N \oplus P$.

Then, we prove (Com 4) by a similar reasoning as in the previous case.

- In order to prove (Com5) we need to show: $\forall \Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X, \forall M, N, \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}, \forall L, P \in \Lambda^{\Gamma \cup \{\mathsf{x}\}}_{\oplus,\mathsf{let}}$,

$$\Gamma \vdash M\ \mathcal{R}^+\ N\ \wedge\ \Gamma \cup \{x\} \vdash L\ \mathcal{R}^+\ P \Rightarrow \Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L)\ \mathcal{R}^+\ (\mathsf{let}\ x = N\ \mathsf{in}\ P).$$

First, we will prove the following two statements:

(5) $\forall M, N \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}, \forall L, P \in \Lambda^{\Gamma \cup \{\mathsf{x}\}}_{\oplus,\mathsf{let}} : \Gamma \vdash M\ \mathcal{R}^+\ N\ \wedge\ \Gamma \cup \{x\} \vdash L\ \mathcal{R}\ P \Rightarrow \Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L)\ \mathcal{R}^+\ (\mathsf{let}\ x = N\ \mathsf{in}\ P)$,
(6) $\forall M, N \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}, \forall L, P \in \Lambda^{\Gamma \cup \{\mathsf{x}\}}_{\oplus,\mathsf{let}} : \Gamma \vdash M\ \mathcal{R}\ N\ \wedge\ \Gamma \cup \{x\} \vdash L\ \mathcal{R}^+\ P \Rightarrow \Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ N)\ \mathcal{R}^+\ (\mathsf{let}\ x = N\ \mathsf{in}\ P)$.

We prove (5) by induction on the derivation of $\Gamma \vdash M\ \mathcal{R}^+\ N$, looking at the last rule used. First we consider the base case where (TC1) is the last rule used. Then we have that $\Gamma \vdash M\ \mathcal{R}\ N$ holds by hypothesis. Since we have assumed that $\mathcal{R}$ is compatible and

$\Gamma \cup \{x\} \vdash L \;\mathcal{R}\; P$ holds, we can conclude $\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L)\ \mathcal{R}\ (\mathsf{let}\ x = N\ \mathsf{in}\ P)$. Now, by applying (TC1) on the latter we obtain $\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L)\ \mathcal{R}^+\ (\mathsf{let}\ x = N\ \mathsf{in}\ P)$. Let us now consider the case where (TC2) is the last rule used. In this case we have $\Gamma \vdash M\ \mathcal{R}^+\ Q$ and $\Gamma \vdash Q\ \mathcal{R}^+\ N$ as additional hypothesis, for some $Q \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$. Now, by induction hypothesis on $\Gamma \vdash M\ \mathcal{R}^+\ Q$ and $\Gamma \cup \{x\} \vdash L\ \mathcal{R}\ P$ we have $\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L)\ \mathcal{R}^+\ (\mathsf{let}\ x = Q\ \mathsf{in}\ P)$. Using the fact that relation $\mathcal{R}$ is compatible, we can conclude its reflexivity and $\Gamma \cup \{x\} \vdash P\ \mathcal{R}\ P$ holds. Next, by induction hypothesis on $\Gamma \vdash Q\ \mathcal{R}^+\ N$ and $\Gamma \cup \{x\} \vdash P\ \mathcal{R}\ P$ we get $\Gamma \vdash (\mathsf{let}\ x = Q\ \mathsf{in}\ P)\ \mathcal{R}^+\ (\mathsf{let}\ x = N\ \mathsf{in}\ P)$. Finally, we conclude applying (TC2) on $\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L)\ \mathcal{R}^+\ (\mathsf{let}\ x = Q\ \mathsf{in}\ P)$ and the latter, obtaining $\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L)\ \mathcal{R}^+\ (\mathsf{let}\ x = N\ \mathsf{in}\ P)$. Statement (6) can be proved similarly.

Let consider the original statement (Com5). We prove it by induction on two derivations $\Gamma \vdash M\ \mathcal{R}^+\ N$ and $\Gamma \cup \{x\} \vdash L\ \mathcal{R}^+\ P$. If we look at the last rules used, we have four possible cases:

1. (TC1) is the last used rule in both derivations;
2. the last rule used in the derivation of $\Gamma \vdash M\ \mathcal{R}^+\ N$ is (TC1), and the last rule used in the derivation of $\Gamma \cup \{x\} \vdash L\ \mathcal{R}^+\ P$ is (TC2);
3. the last rule used in the derivation of $\Gamma \vdash M\ \mathcal{R}^+\ N$ is (TC2), and the last rule used in the derivation of $\Gamma \cup \{x\} \vdash L\ \mathcal{R}^+\ P$ is (TC1);
4. (TC2) is the last used rule in both derivations.

The first case follows from the fact that relation $\mathcal{R}$ is compatible, and second and third cases follow from the statements (5) and (6) we proved. Thus, we only consider the case where both derivations are concluded by applying the rule (TC2). In this case, as additional hypothesis we get that: for some $Q \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$, $\Gamma \vdash M\ \mathcal{R}^+\ Q$ and $\Gamma \vdash Q\ \mathcal{R}^+\ N$ hold, and for some $R \in \Lambda^{\Gamma\cup\{x\}}_{\oplus,\mathsf{let}}$, $\Gamma \cup \{x\} \vdash L\ \mathcal{R}^+\ R$ and $\Gamma \cup \{x\} \vdash R\ \mathcal{R}^+\ P$ hold. First by induction hypothesis on $\Gamma \vdash M\ \mathcal{R}^+\ Q$ and $\Gamma \cup \{x\} \vdash L\ \mathcal{R}^+\ R$ we get $\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L)\ \mathcal{R}^+\ (\mathsf{let}\ x = Q\ \mathsf{in}\ R)$. Next, by induction hypothesis on $\Gamma \vdash Q\ \mathcal{R}^+\ N$ and $\Gamma \cup \{x\} \vdash R\ \mathcal{R}^+\ P$ we have $\Gamma \vdash (\mathsf{let}\ x = Q\ \mathsf{in}\ R)\ \mathcal{R}^+\ (\mathsf{let}\ x = N\ \mathsf{in}\ P)$. Now we can apply (TC2) and obtain $\Gamma \vdash (\mathsf{let}\ x = M\ \mathsf{in}\ L)\ \mathcal{R}^+\ (\mathsf{let}\ x = N\ \mathsf{in}\ P)$.

◀

▶ **Lemma 38.** *If $\mathcal{R}$ is closed under term-substitution, then so is $\mathcal{R}^+$.*

Proof. Proving that $\mathcal{R}^+$ is closed under term-substitution means to show: $\forall \Gamma \in P_{\mathsf{FIN}}(X), \forall x \in X - \Gamma, \forall M, N \in \Lambda^{\Gamma\cup\{x\}}_{\oplus,\mathsf{let}}, \forall L \in \Lambda^{\Gamma}_{\oplus,\mathsf{let}}$,

$$\Gamma \cup \{x\} \vdash M\ \mathcal{R}^+\ N \Rightarrow M\{L/x\}\ \mathcal{R}^+\ N\{L/x\}.$$

We prove this statement by induction on the derivation of $\Gamma \cup \{x\} \vdash M\ \mathcal{R}^+\ N$. As usual, we look at the last rule used in the derivation. First we consider the base case, where the last rule used is (TC1) and we have that $\Gamma \cup \{x\} \vdash M\ \mathcal{R}\ N$ holds. Using the fact that relation $\mathcal{R}$ is closed under term-substitution, we can conclude $\Gamma \vdash M\{L/x\}\ \mathcal{R}\ N\{L/x\}$ holds. Now, we apply (TC1) on the latter and obtain $\Gamma \vdash M\{L/x\}\ \mathcal{R}^+\ N\{L/x\}$. Next, let us consider the case where (TC2) is the last rule used. Then, we have that for some $Q \in \Lambda^{\Gamma\cup\{x\}}_{\oplus,\mathsf{let}}$, $\Gamma \cup \{x\} \vdash M\ \mathcal{R}^+\ Q$ and $\Gamma \cup \{x\} \vdash Q\ \mathcal{R}^+\ N$ hold. Now, by induction hypothesis on both of them, we get $\Gamma \vdash M\{L/x\}\ \mathcal{R}^+\ Q\{L/x\}$ and $\Gamma \vdash Q\{L/x\}\ \mathcal{R}^+\ N\{L/x\}$. We conclude applying (TC2) on the latter two, obtaining $\Gamma \vdash M\{L/x\}\ \mathcal{R}^+\ N\{L/x\}$. ◀

▶ **Lemma 39.** *If a $\Lambda_{\oplus,\mathsf{let}}$-relation $\mathcal{R}$ is a preorder, then so is $(\mathcal{R}^H)^+$.*

Proof. A relation is a preorder if it is reflexive and transitive. We assume that $\mathcal{R}$ is reflexive and transitive. Then, by Lemma 33 and Lemma 37 we conclude $(\mathcal{R}^H)^+$ is compatible and hence reflexive. Relation $(\mathcal{R}^H)^+$ is transitive by its construction, since it is a transitive closure of relation $\mathcal{R}^H$. Thus, we conclude relation $(\mathcal{R}^H)^+$ is a preorder. ◀

The crucial part in proving that probabilistic similarity is a precongruence is Key Lemma (Lemma 44). First, we need the definition of a probability assignment and an auxiliary lemma about it.

▶ **Definition 40.** $\mathbb{P} = (\{p_i\}_{1\le i\le n}, \{r_I\}_{I\subseteq\{1,\dots,n\}})$ *is a probability assignment if for each* $I \subseteq \{1,\dots,n\}$ *it holds that* $\sum_{i\in I} p_i \le \sum_{J\cap I\neq\emptyset} r_J$.

▶ **Lemma 41.** *Let* $\mathbb{P} = (\{p_i\}_{1\le i\le n}, \{r_I\}_{I\subseteq\{1,\dots,n\}})$ *be a probability assignment. Then for every nonempty* $I \subseteq \{1,\dots,n\}$ *and for every* $k \in I$ *there is* $s_{k,I} \in [0,1]$ *which satisfies the following conditions:*
1. *for every* I*, it holds that* $\sum_{k\in I} s_{k,I} \le 1$*;*
2. *for every* $k \in \{1,\dots,n\}$*, it holds that* $p_k \le \sum_{k\in I} s_{k,I} \cdot r_I$*.*

The proof of Lemma 41 is omitted, but it can be found in [6]. Besides Lemma 41, in the proof of Key Lemma we use the following technical Lemmas.

▶ **Lemma 42.** *For every* $X \subseteq \Lambda^{\{x\}}_{\oplus,\mathsf{let}}$*, it holds that* $\lesssim(\lambda x.X) = \lambda x.(\lesssim(X))$ *and* $\lesssim(\nu x.X) = \nu x.(\lesssim(X))$.
$\lambda x.(\lesssim(X))$ *stands for the set* $\{\lambda x.M \mid \exists N \in X, N \lesssim M\}$.

Proof.

$$\begin{aligned}\lambda x.M \in\lesssim(\lambda x.X) &\Leftrightarrow \exists N \in X,\ \lambda x.N \lesssim \lambda x.M\\ &\Leftrightarrow \exists N \in X,\ N \lesssim M,\\ &\Leftrightarrow \lambda x.M \in \lambda x.\lesssim(X).\end{aligned}$$

The second part of the statement can be proved analogously. ◀

▶ **Lemma 43.** *If* $M \lesssim N$*, then for every* $X \subseteq \Lambda^{\{x\}}_{\oplus,\mathsf{let}}$*,* $[\![M]\!](\lambda x.X) \le [\![N]\!](\lambda x.\lesssim(X))$*.*

Proof. It is a straightforward consequence of Lemma 42. ◀

▶ **Lemma 44. (Key Lemma)** *If* $M \lesssim^H N$*, then for every* $X \subseteq \Lambda^{\{x\}}_{\oplus,\mathsf{let}}$ *it holds that* $[\![M]\!](\lambda x.X) \le [\![N]\!](\lambda x.(\lesssim^H(X)))$*.*

Proof. Since $[\![M]\!] = \sup\{\mathscr{D}\ ;\ M \Downarrow \mathscr{D}\}$, it is enough to prove the following statement: if $M \lesssim^H N$ and $M \Downarrow \mathscr{D}$ then for every $X \subseteq \Lambda^{\{x\}}_{\oplus,\mathsf{let}}$ it holds that $\mathscr{D}(\lambda x.X) \le [\![N]\!](\lambda x.(\lesssim^H(X))$. We prove it by induction on the derivation of $M \Downarrow \mathscr{D}$, looking at the last rule used.

- If $M \Downarrow \emptyset$, then we have $\mathscr{D}(\lambda x.X) = 0 \le [\![N]\!](\lambda x.(\lesssim^H(X)))$ for every $X \subseteq \Lambda^{\{x\}}_{\oplus,\mathsf{let}}$.
- Next, we consider the case where M is a value $\lambda x.Q$ and $\mathscr{D} = \lambda x.Q$, that is $\mathscr{D}(\lambda x.Q) = 1$. Since M is a value the last used rule in the derivation of $M \lesssim^H N$ (i.e. $\emptyset \vdash M \lesssim^H N$) has to be (How2). Thus, we have that for some $P \in \Lambda^{\{x\}}_{\oplus,\mathsf{let}}$, $x \vdash Q \lesssim^H P$ and $\emptyset \vdash \lambda x.P \lesssim N$ hold as additional hypothesis. For $X \subseteq \Lambda^{\{x\}}_{\oplus,\mathsf{let}}$ we consider two cases:
  - If $Q \notin X$, then $\mathscr{D}(\lambda x.X) = 0$ and the statement holds.
  - If $Q \in X$, then $\mathscr{D}(\lambda x.X) = 1$ and $P \in\lesssim^H(X)$. For every $L \in\lesssim(P)$, we have that $x \vdash Q \lesssim^H P$ and $x \vdash P \lesssim L$. By Lemma 34 we conclude that $x \vdash Q \lesssim^H L$ holds. Thus, $L \in\lesssim^H(X)$ and it holds that $\lesssim(P) \subseteq\lesssim^H(X)$. From Lemma 43 we obtain the following

$$\mathscr{D}(\lambda x.X) = 1 = [\![\lambda x.P]\!](\lambda x.P) \le [\![N]\!](\lambda x.\lesssim(P)) \le [\![N]\!](\lambda x.\lesssim^H(X)).$$

- Let M be an application LP. Then, we have $\mathscr{D} = \sum_{\lambda x.Q} \mathscr{F}(\lambda x.Q) \cdot \mathscr{H}_{Q,P}$ where $L \Downarrow \mathscr{F}$ and for any $\lambda x.Q \in \mathsf{S}(\mathscr{F})$, $\{Q\{P/x\} \Downarrow \mathscr{H}_{Q,P}\}$. The last rule used in the derivation of $\emptyset \vdash M \precsim^H N$ has to be (How3), thus we get $\emptyset \vdash L \precsim^H R$, $\emptyset \vdash P \precsim^H S$ and $\emptyset \vdash RS \precsim N$ as additional hypothesis. If we apply the induction hypothesis on $L \Downarrow \mathscr{F}$ and $\emptyset \vdash L \precsim^H R$, we obtain that for any $Y \subseteq \Lambda_{\oplus,\mathsf{let}}^{\{\mathsf{x}\}}$ it holds that

$$\mathscr{F}(\lambda x.Y) \leq [\![R]\!](\lambda x. \precsim^H (Y)) \tag{9}$$

Since $\mathscr{F}$ is a finite distribution, distribution $\mathscr{D} = \sum_{\lambda x.Q} \mathscr{F}(\lambda x.Q) \cdot \mathscr{H}_{Q,P}$ is a sum of finitely many summands. Let us assume that $\mathsf{S}(\mathscr{F}) = \{\lambda x.Q_1, \ldots, \lambda x.Q_n\}$. From Equation (9) we conclude

$$\mathscr{F}(\bigcup_{i\in I} \lambda x.Q_i) \leq [\![R]\!](\bigcup_{i\in I} \lambda x. \precsim^H (Q_i)),$$

for every $I \subseteq \{1, \ldots, n\}$ which allows us to apply Lemma 41. Hence, for every $U \in \bigcup_{i=1}^n \precsim^H (Q_i)$ there exist numbers $r_1^{U,R}, \ldots, r_n^{U,R}$ such that:

$$[\![R]\!](\lambda x.U) \geq \sum_{i=1}^n r_i^{U,R}, \qquad \forall U \in \bigcup_{i=1}^n \precsim^H (Q_i);$$

$$\mathscr{F}(\lambda x.Q_i) \leq \sum_{U \in \precsim^H (Q_i)} r_i^{U,R}, \qquad \forall i \in \{1, \ldots, n\}.$$

From these equations we can conclude the following

$$\mathscr{D} \leq \sum_{i=1}^n \left(\sum_{U \in \precsim^H (Q_i)} r_i^{U,R} \right) \cdot \mathscr{H}_{Q_i,P} = \sum_{i=1}^n \sum_{U \in \precsim^H (Q_i)} r_i^{U,R} \cdot \mathscr{H}_{Q_i,P}.$$

Since $Q_i \precsim^H U$ and $P \precsim^H S$ holds, by Lemma 36 we have $Q_i\{P/x\} \precsim^H U\{S/x\}$. Now, by applying the induction hypothesis on the derivations $Q_i\{P/x\} \Downarrow \mathscr{H}_{Q_i,P}$, $i \in \{1, \ldots, n\}$, we obtain that for every $X \subseteq \Lambda_{\oplus,\mathsf{let}}^{\{\mathsf{x}\}}$ it holds that

$$\begin{aligned}
\mathscr{D}(\lambda x.X) &\leq \sum_{i=1}^n \sum_{U \in \precsim^H (Q_i)} r_i^{U,R} \cdot [\![U\{S/x\}]\!](\lambda x. \precsim^H (X)) \\
&\leq \sum_{i=1}^n \sum_{U \in \bigcup_{i=1}^n \precsim^H (Q_i)} r_i^{U,R} \cdot [\![U\{S/x\}]\!](\lambda x. \precsim^H (X)) \\
&= \sum_{U \in \bigcup_{i=1}^n \precsim^H (Q_i)} \sum_{i=1}^n r_i^{U,R} \cdot [\![U\{S/x\}]\!](\lambda x. \precsim^H (X)) \\
&= \sum_{U \in \bigcup_{i=1}^n \precsim^H (Q_i)} \left(\sum_{i=1}^n r_i^{U,R} \right) \cdot [\![U\{S/x\}]\!](\lambda x. \precsim^H (X)) \\
&\leq \sum_{U \in \bigcup_{i=1}^n \precsim^H (Q_i)} [\![R]\!](\lambda x.U) \cdot [\![U\{S/x\}]\!](\lambda x. \precsim^H (X)) \\
&\leq \sum_{U \in \Lambda_{\oplus,\mathsf{let}}^{\{\mathsf{x}\}}} [\![R]\!](\lambda x.U) \cdot [\![U\{S/x\}]\!](\lambda x. \precsim^H (X)) \\
&= [\![RS]\!](\lambda x. \precsim^H (X)) \\
&\leq [\![N]\!](\lambda x. \precsim (\precsim^H (X))) \\
&\leq [\![N]\!](\lambda x. \precsim^H (X)).
\end{aligned}$$

- Let M be a probabilistic sum $L \oplus P$, then $\mathscr{D} = \frac{1}{2}\mathscr{F} + \frac{1}{2}\mathscr{E}$ where $L \Downarrow \mathscr{F}$ and $P \Downarrow \mathscr{E}$. The last used rule in the derivation of $\emptyset \vdash M \lesssim^H N$ has to be (How4) and we have that for some $R, S \in \Lambda^{\emptyset}_{\oplus,\mathsf{let}}$, $\emptyset \vdash L \lesssim^H R$, $\emptyset \vdash P \lesssim^H S$ and $\emptyset \vdash R \oplus S \lesssim N$ hold as additional hypothesis. If we apply the induction hypothesis on $L \Downarrow \mathscr{F}$ and $\emptyset \vdash L \lesssim^H R$, we obtain that for any $X \subseteq \Lambda^{\{x\}}_{\oplus,\mathsf{let}}$, $\mathscr{F}(\lambda x.X) \leq [\![R]\!](\lambda x. \lesssim^H (X))$ holds. Similarly, if we apply the induction hypothesis on $P \Downarrow \mathscr{E}$ and $\emptyset \vdash P \lesssim^H S$, we obtain that for any $X \subseteq \Lambda^{\{x\}}_{\oplus,\mathsf{let}}$, $\mathscr{E}(\lambda x.X) \leq [\![S]\!](\lambda x. \lesssim^H (X))$. Since, $\emptyset \vdash R \oplus S \lesssim N$, it holds that $[\![R \oplus S]\!](\lambda x. \lesssim^H (X)) \leq [\![N]\!](\lambda x. \lesssim^H (X))$. From Lemma 5 and previously concluded statements we obtain the following:

$$\begin{aligned}
\mathscr{D}(\lambda x.X) &= \frac{1}{2}\mathscr{F}(\lambda x.X) + \frac{1}{2}\mathscr{E}(\lambda x.X) \\
&\leq \frac{1}{2}[\![R]\!](\lambda x. \lesssim^H (X)) + \frac{1}{2}[\![S]\!](\lambda x. \lesssim^H (X)) \\
&= [\![R \oplus S]\!](\lambda x. \lesssim^H (X)) \\
&\leq [\![N]\!](\lambda x. \lesssim^H (X)).
\end{aligned}$$

- Let us now consider the case where $M = (\mathsf{let}\ x = L\ \mathsf{in}\ P)$. Then, we have $\mathscr{D} = \sum_{\lambda x.Q} \mathscr{F}(\lambda x.Q) \cdot \mathscr{H}_{Q,P}$ where $L \Downarrow \mathscr{F}$ and for any $\lambda x.Q \in \mathsf{S}(\mathscr{F})$, $\{P\{\lambda x.Q/x\} \Downarrow \mathscr{H}_{Q,P}\}$. The last rule used in the derivation of $\emptyset \vdash M \lesssim^H N$ has to be (How5), thus we get $\emptyset \vdash L \lesssim^H R$, $x \vdash P \lesssim^H S$ and $\emptyset \vdash (\mathsf{let}\ x = R\ \mathsf{in}\ S) \lesssim N$ as additional hypothesis. By applying the induction hypothesis on $L \Downarrow \mathscr{F}$ and $\emptyset \vdash L \lesssim^H R$, we obtain that

$$\mathscr{F}(\lambda x.Y) \leq [\![R]\!](\lambda x. \lesssim^H (Y)), \tag{10}$$

holds for any $Y \subseteq \Lambda^{\{x\}}_{\oplus,\mathsf{let}}$. $\mathscr{F}$ is a finite distribution, hence the distribution $\mathscr{D} = \sum_{\lambda x.Q} \mathscr{F}(\lambda x.Q) \cdot \mathscr{H}_{Q,P}$ is a sum of finitely many summands. Let the support of $\mathscr{F}$ be the set $\mathsf{S}(\mathscr{F}) = \{\lambda x.Q_1, \ldots, \lambda x.Q_n\}$. Equation (10) implies that for every $I \subseteq \{1, \ldots, n\}$ the following holds

$$\mathscr{F}(\bigcup_{i \in I} \lambda x.Q_i) \leq [\![R]\!](\bigcup_{i \in I} \lambda x. \lesssim^H (Q_i)).$$

This allows us to apply Lemma 41. Thus, for every $U \in \bigcup_{i=1}^n \lesssim^H (Q_i)$ there exist numbers $r_1^{U,R}, \ldots, r_n^{U,R}$ such that:

$$[\![R]\!](\lambda x.U) \geq \sum_{i=1}^n r_i^{U,R}, \qquad \forall U \in \bigcup_{i=1}^n \lesssim^H (Q_i);$$

$$\mathscr{F}(\lambda x.Q_i) \leq \sum_{U \in \lesssim^H (Q_i)} r_i^{U,R}, \qquad \forall i \in \{1, \ldots, n\}.$$

Now, we can conclude the following

$$\mathscr{D} \leq \sum_{i=1}^n \left(\sum_{U \in \lesssim^H (Q_i)} r_i^{U,R} \right) \cdot \mathscr{H}_{Q_i,P} = \sum_{i=1}^n \sum_{U \in \lesssim^H (Q_i)} r_i^{U,R} \cdot \mathscr{H}_{Q_i,P}.$$

Since $Q_i \lesssim^H U$ holds and $\lesssim^H$ is compatible by Lemma 33 , $\lambda x.Q_i \lesssim^H \lambda x.U$ holds. By applying Lemma 36 on $P \lesssim^H S$ and the latter we get $P\{\lambda x.Q_i/x\} \lesssim^H S\{\lambda x.U/x\}$. If we apply the induction hypothesis on the derivations $P\{\lambda x.Q_i/x\} \Downarrow \mathscr{H}_{Q_i,P}$, $i \in \{1, \ldots, n\}$, we obtain that for every $X \subseteq \Lambda^{\{x\}}_{\oplus,\mathsf{let}}$ it holds that

$$\mathscr{D}(\lambda x.X) \leq [\![N]\!](\lambda x. \lesssim^H (X)).$$

This concludes the proof. ◀

Proof of Lemma 17.

The proof that similarity is a precongruence consists of two steps: the first step is to show that the relation $(\lesssim^H)^+$ is a precongruence and the second one is to show that it coincide with relation $\lesssim$. Since $\lesssim$ is a preorder, then by Lemma 39, relation $(\lesssim^H)^+$ is also a preorder. Relation $\lesssim$ is reflexive, hence by Lemma 33 we have $\lesssim^H$ is compatible. Furthermore, Lemma 37 ensures that $(\lesssim^H)^+$ is also compatible. So, we can conclude that $(\lesssim^H)^+$ is a precongruence. Next, we want to show that $\lesssim = (\lesssim^H)^+$. From the construction of Howe's lifting $\lesssim^H$ and its transitive closure $(\lesssim^H)^+$ it follows that $\lesssim \subseteq (\lesssim^H)^+$. It remains to show the inclusion $(\lesssim^H)^+ \subseteq \lesssim$. We show that $(\lesssim^H)^+$ is included in some probabilistic simulation $\mathcal{R}$, thus it is also included in the largest one, $\lesssim$. The relation we consider is $\mathcal{R} = \{(M,N) : M\ (\lesssim^H)^+\ N\} \cup \{(\nu x.M, \nu x.N) : M\ (\lesssim^H)^+\ N\}$. It is obvious that $(\lesssim^H)^+ \subseteq \mathcal{R}$, so it only remains to show that $\mathcal{R}$ is a probabilistic simulation. Relation $(\lesssim^H)^+$ is closed under term-substitution (by Lemma 36 and Lemma 38), hence it is enough to consider only closed terms and distinguished values. Since $(\lesssim^H)^+$ is a preorder relation (reflexive and transitive), it is easy to see $\mathcal{R}$ is also a preorder. We show the following two points:

1. If $M\ (\lesssim^H)^+\ N$, then for every $X \subseteq \Lambda_{\oplus,\mathsf{let}}^{\{x\}}$ it holds that
   $P(M,\tau,\nu x.X) \le P(N,\tau,\mathcal{R}(\nu x.X))$.
2. If $M\ (\lesssim^H)^+\ N$, then for every $L \in \Lambda_{\oplus,\mathsf{let}}^{\emptyset}$ and for every $X \subseteq \Lambda_{\oplus,\mathsf{let}}^{\{x\}}$,
   $P(\nu x.M, L, X) \le P(\nu x.N, L, \mathcal{R}(X))$.

The first point we prove by induction on the derivation of $M\ (\lesssim^H)^+\ N$. We look at the last rule used. Let us start with the base case where (TC1) is the last rule used. Then, we have $M\ \lesssim^H\ N$ holds by hypothesis. By Key Lemma we conclude the following:

$$\begin{aligned}
P(M,\tau,\nu x.X) &= [\![M]\!](\lambda x.X)\\
&\le [\![N]\!](\lambda x. \lesssim^H (X))\\
&\le [\![N]\!](\lambda x.(\lesssim^H)^+(X))\\
&\le [\![N]\!](\mathcal{R}(\nu x.X))\\
&= P(N,\tau,\mathcal{R}(\nu x.X)).
\end{aligned}$$

Next, we consider the case where (TC2) is the last rule used and we have that for some $P \in \Lambda_{\oplus,\mathsf{let}}^{\emptyset}$, $M\ (\lesssim^H)^+\ P$ and $P\ (\lesssim^H)^+\ N$ hold. By induction hypothesis on both of them, we obtain:

$$P(M,\tau,X) \le P(P,\tau,\mathcal{R}(X)),$$

$$P(P,\tau,\mathcal{R}(X)) \le P(N,\tau,\mathcal{R}(\mathcal{R}(X))).$$

It is easy to show that $\mathcal{R}(\mathcal{R}(X)) \subseteq \mathcal{R}(X)$, thus we can conclude

$$P(M,\tau,X) \le P(N,\tau,\mathcal{R}(X)).$$

This concludes the proof of the first point.

If $M\ (\lesssim^H)^+\ N$ and $L \in \Lambda_{\oplus,\mathsf{let}}^{\emptyset}$, then because of the fact that $(\lesssim^H)^+$ closed under term-substitution, we have that $M\{L/x\}\ (\lesssim^H)^+\ N\{L/x\}$ holds. As a consequence, we have that whenever $M\{L/x\} \in X$, then $N\{L/x\} \in (\lesssim^H)^+(X)$ and it holds that

$$\begin{aligned} P(\nu x.M, L, X) &= 1 \\ &= P(\nu x.N, L, (\lesssim^H)^+(X)) \\ &= P(\nu x.N, L, \mathcal{R}(X)). \end{aligned}$$

On the other hand, if $M\{L/x\} \notin X$, then $P(\nu x.M, L, X) = 0 \leq P(\nu x.N, L, \mathcal{R}(X))$.

To prove that bisimilarity is a congruence we need to prove that $\sim$ is an equivalence relation, which is compatible. Relation $\sim$ is an equivalence relation by its definition. Since we know that $\sim = \lesssim \cap \lesssim^{op}$ holds, from the fact that similarity is a precongruence it follows that $\sim$ is also compatible. This concludes the proof.

◀