# On the Lattice of Program Metrics

**Ugo Dal Lago**
University of Bologna, Italy
INRIA Sophia Antipolis, France

**Naohiko Hoshino**
Sojo University, Japan

**Paolo Pistone**
University of Bologna, Italy

## Abstract

In this paper we are concerned with understanding the nature of program metrics for calculi with higher-order types, seen as natural generalizations of program equivalences. Some of the metrics we are interested in are well-known, such as those based on the interpretation of terms in metric spaces and those obtained by generalizing observational equivalence. We also introduce a new one, called the interactive metric, built by applying the well-known Int-Construction to the category of metric complete partial orders. Our aim is then to understand how these metrics relate to each other, i.e., whether and in which cases one such metric refines another, in analogy with corresponding well-studied problems about program equivalences. The results we obtain are twofold. We first show that the metrics of semantic origin, i.e., the denotational and interactive ones, lie *in between* the observational and equational metrics and that in some cases, these inclusions are strict. Then, we give a result about the relationship between the denotational and interactive metrics, revealing that the former is less discriminating than the latter. All our results are given for a linear lambda-calculus, and some of them can be generalized to calculi with graded comonads, in the style of Fuzz.

**2012 ACM Subject Classification** Theory of computation → Program semantics

**Keywords and phrases** Metrics, Lambda Calculus, Linear Types

**Digital Object Identifier** 10.4230/LIPIcs.FSCD.2023.20

**Related Version** *Extended Version*: `https://arxiv.org/abs/2302.05022`

**Funding** The first and third authors are partially supported by the ERC CoG DIAPASoN, GA 818616. The second author is supported by JSPS KAKENHI Grant Number JP21K11762.

## 1 Introduction

Program equivalence is one of the most important concepts in the semantics of programming languages: every way of giving semantics to programs induces a notion of equivalence, and the various notions of equivalence available for the same language, even when very different from each other, help us understanding the deep nature of the language itself. Indeed, there is not *one* single, preferred way to construct a notion of equivalence for programs. The latter is especially true in presence of higher-order types or in scenarios in which programs have a fundamentally interactive behavior, e.g. in process algebras. For example, the relationship between observational equivalence, the most coarse-grained congruence relation among those which are coherent with the underlying notion of observation, and denotational semantics has led in some cases to so-called full-abstraction results (e.g. [20, 13]), which are known to hold only for *some* denotational models and in *some* programming languages. A similar argument applies to applicative bisimilarity, which, e.g., is indeed fully abstract in presence of *probabilistic* effects [5, 8] but not so in presence of *nondeterministic* effects [23].

$$
\begin{tikzcd}
 & d^{\mathrm{obs}} = d^{\mathrm{log}} & \\
d^{\mathrm{den}} \arrow[ur] & & d^{\mathrm{int}} \arrow[ul, thick] \arrow[ll, thick] \\
 & d^{\mathrm{equ}} \arrow[ul, thick] \arrow[ur] &
\end{tikzcd}
$$

**Figure 1** Illustration of our comparison results for program metrics: an arrow $d^{\mathbf{a}} \to d^{\mathbf{b}}$ indicates that $d^{\mathbf{b}}$ is *coarser* (i.e. less discriminating) than $d^{\mathbf{a}}$. Thick arrows indicate *strict* domination.

Equivalences, although central to the theory of programming languages, do not allow us to say anything about all those pairs of programs which, while qualitatively exhibiting different behaviors, behave *similarly* in a quantitative sense. This has led to the study of notions of *distance* between programs, which often take the form of (pseudo-)metrics on the space of programs or of their denotations. In this sense we can distinguish at least three defining styles:

- First, observational equivalence can be generalized to a metric, maintaining the intrinsic quantification across all contexts, but observing a difference rather than an equality [6, 7].
- There is also an approach obtained by generalizing equational logic, recently introduced by Mardare et al. [25], which has been adapted to higher-order computations with both linear [9] and non-linear [11] types.
- Finally, linear calculi admit a denotational interpretation in the category of metric complete partial orders [2], and this is well-known to work well in presence of graded comonads.

In other words, various definitional styles for program equivalences for higher-order calculi have been proved to have a meaningful metric counterpart, at least when the underlying type system is based on linear or graded types. Actually, metric semantics for non-linear [15, 28] as well as effectful [14, 10] higher-order calculi have also been recently explored. However, there is a missing tale in this picture, namely the one provided by interactive semantic models akin to game semantics and the geometry of interaction [17], which were key ingredients towards the aforementioned full-abstraction results. Moreover, the relationship between the various notions of distance in the literature has been studied only superficially, and the overall situation is currently less clear than for program equivalences.

The aim of this work is to shed light on the landscape about metrics in higher-order programs. Notably, a new metric between programs inspired by Girard's geometry of interaction [17] is defined, being obtained by applying the so-called Int-construction [21, 1] to the category of metric complete partial orders. The result is a denotational model, which, while fundamentally different from existing metric models, provides a natural way to measure the distance between programs, which we call the *interactive metric*. In the interactive metric, differences between two programs can be observed incrementally, by interacting with the underlying denotational interpretation in the question-answer protocol typical of game semantics and the geometry of interaction.

Technically, the main contribution of the paper is an in-depth study of the relationships between the various metrics existing in the literature, including the interactive metric. Overall, the result of this analysis is the one in Figure 1. The observational metric remains the least discriminating, while the equational metric is proved to be the one assigning the greatest distances to (pairs of) programs. The two metrics of a semantic nature, namely the denotational one and the interactive one, stand in between the two metrics mentioned above, with the interactive metric being more discriminating than the denotational one.

The remainder of this manuscript is structured as follows. After recalling some basic facts about metric spaces in Section 2, in Section 3 we introduce a basic linear programming language over the reals and its associated notion of program metrics; in Section 4, we discuss the logical relation metric and the observational metric; in Section 5, we discuss the equational metric; in Section 6, we introduce the two denotational metrics; Sections 7 and 8 contain our main comparison results, and in Section 9, we shortly discuss the case of graded exponentials. Some proofs are omitted and can be found in an extended version of this paper [12].

## 2 Preliminaries

In this section, we recall the notions of extended pseudo-metric spaces and non-expansive functions. Let $\mathbb{R}^{\infty}_{\geq 0}$ be the set $\{a \in \mathbb{R} \mid a \geq 0\} \cup \{\infty\}$ of non-negative real numbers and infinity. An *extended pseudo-metric space* $X$ consists of a set $|X|$ and a function $d_X : |X| \times |X| \to \mathbb{R}^{\infty}_{\geq 0}$ satisfying the following conditions:

- For all $x \in |X|$, we have $d_X(x, x) = 0$;
- For all $x, y \in |X|$, we have $d_X(x, y) = d_X(y, x)$;
- For all $x, y, z \in |X|$, we have $d_X(x, z) \leq d_X(x, y) + d_X(y, z)$.

In the sequel, we simply refer to extended pseudo-metric spaces as metric spaces, and we denote the underlying set $|X|$ by $X$.

For metric spaces $X$ and $Y$, a function $f : X \to Y$ is said to be *non-expansive* when for all $x, y \in X$, we have $d_Y(fx, fy) \leq d_X(x, y)$. We write **Met** for the category of metric spaces and non-expansive functions. The category **Met** has a symmetric monoidal closed structure $(1, \otimes, \multimap)$ where the metric of the tensor product $X \otimes Y$ is given by

$$d_{X\otimes Y}((x, y), (z, w)) = d_X(x, z) + d_Y(y, w).$$

We suppose that the monoidal product is left associative, and we denote the $n$-fold monoidal product of $X$ by $X^{\otimes n}$. In the sequel, $\mathbb{R}$ denotes the metric space of real numbers equipped with the absolute distance $d_{\mathbb{R}}(a, b) = |a - b|$.

## 3 A Linear Programming Language

### 3.1 Syntax and Operational Semantics

We introduce our target language that is a linear lambda calculus equipped with constant symbols for real numbers and non-expansive functions. We fix a set $S$ of non-expansive functions $f : \mathbb{R}^{\otimes n} \to \mathbb{R}$ with $n \geq 1$. We call $n$ the *arity* of $f$. For example, $S$ may include addition $+ : \mathbb{R} \otimes \mathbb{R} \to \mathbb{R}$ and trigonometric functions such as $\sin, \cos : \mathbb{R} \to \mathbb{R}$. We assume function symbols $\overline{f}$ for $f \in S$ and constant symbols $\overline{a}$ for real numbers $a \in \mathbb{R}$.

Our language, denoted by $\Lambda_S$, is given as follows. Types and environments are given by

$$\text{Types} \quad \tau, \sigma := \mathbf{R} \mid \mathbf{I} \mid \tau \multimap \sigma \mid \tau \otimes \sigma, \qquad \text{Environments} \quad \Gamma, \Delta := \varnothing \mid \Gamma, x : \tau.$$

We denote the set of types by **Ty** and denote the set of environments by **Env**. We always suppose that every variable appears at most once in any environment. For environments $\Gamma$ and $\Delta$ that do not share any variable, we write $\Gamma \# \Delta$ for a *merge* [3, 18] of $\Gamma$ and $\Delta$, that is an environment obtained by shuffling variables in $\Gamma$ and $\Delta$ preserving the order of variables in $\Gamma$ and the order of variables in $\Delta$. For example, $(x : \tau, y : \sigma, y' : \sigma', x' : \tau')$ is a merge of $(x : \tau, x' : \tau')$ and $(y : \sigma, y' : \sigma')$. When we write $\Gamma \# \Delta$, we implicitly suppose that no variable is shared by $\Gamma$ and $\Delta$. Terms, values and contexts are given by the following BNF.

$$\frac{}{x:\tau\vdash x:\tau}\qquad\frac{a\in\mathbb{R}}{\vdash\overline{a}:\mathbf{R}}\qquad\frac{}{\vdash *:\mathbf{I}}\qquad\frac{f\in S\quad\Gamma_1\vdash M_1:\mathbf{R}\quad\dots\quad\Gamma_{\mathrm{ar}(f)}\vdash M_{\mathrm{ar}(f)}:\mathbf{R}}{\Gamma_1\#\cdots\#\Gamma_{\mathrm{ar}(f)}\vdash\overline{f}(M_1,\dots,M_{\mathrm{ar}(f)}):\mathbf{R}}$$

$$\frac{\Gamma,x:\sigma\vdash M:\tau}{\Gamma\vdash\lambda x:\sigma.\,M:\sigma\multimap\tau}\qquad\frac{\Gamma\vdash M:\sigma\multimap\tau\quad\Delta\vdash N:\sigma}{\Gamma\#\Delta\vdash M\,N:\tau}\qquad\frac{\Gamma\vdash M:\tau\quad\Delta\vdash N:\sigma}{\Gamma\#\Delta\vdash M\otimes N:\tau\otimes\sigma}$$

$$\frac{\Gamma\vdash M:\mathbf{I}\quad\Delta\vdash N:\tau}{\Gamma\#\Delta\vdash\mathbf{let}\ *\ \mathbf{be}\ M\ \mathbf{in}\ N:\tau}\qquad\frac{\Gamma\vdash M:\sigma_1\otimes\sigma_2\quad\Delta,x:\sigma_1,y:\sigma_2\vdash N:\tau}{\Gamma\#\Delta\vdash\mathbf{let}\ x\otimes y\ \mathbf{be}\ M\ \mathbf{in}\ N:\tau}$$

**Figure 2** Typing Rules.

$$\frac{}{V\hookrightarrow V}\qquad\frac{M_1\hookrightarrow\overline{a_1}\quad\dots\quad M_n\hookrightarrow\overline{a_n}}{\overline{f}(M_1,\dots,M_n)\hookrightarrow\overline{f(a_1,\dots,a_n)}}\qquad\frac{M\hookrightarrow\lambda x:\tau.\,L\quad N\hookrightarrow V\quad L[V/x]\hookrightarrow U}{M\,N\hookrightarrow U}$$

$$\frac{M\hookrightarrow V\quad N\hookrightarrow U}{M\otimes N\hookrightarrow V\otimes U}\qquad\frac{M\hookrightarrow *\quad N\hookrightarrow V}{\mathbf{let}\ *\ \mathbf{be}\ M\ \mathbf{in}\ N\hookrightarrow V}\qquad\frac{M\hookrightarrow V\otimes U\quad N[V/x,U/y]\hookrightarrow W}{\mathbf{let}\ x\otimes y\ \mathbf{be}\ M\ \mathbf{in}\ N\hookrightarrow W}$$

**Figure 3** Evaluation Rules.

$$\begin{array}{ll}
\text{Terms} & M,N := x\in\mathbf{Var}\mid\overline{a}\mid *\mid\overline{f}(M_1,\dots,M_{\mathrm{ar}(f)})\mid M\,N\mid\lambda x:\tau.\,M\mid\\
& \qquad\quad M\otimes N\mid\mathbf{let}\ *\ \mathbf{be}\ M\ \mathbf{in}\ N\mid\mathbf{let}\ x\otimes y\ \mathbf{be}\ M\ \mathbf{in}\ N\\
\text{Values} & V,U := \overline{a}\mid *\mid\lambda x:\tau.\,M\mid V\otimes U\\
\text{Contexts} & C[-] := [-]\mid\overline{f}(M,\dots,M',C[-],N',\dots,N)\mid C[-]\,M\mid M\,C[-]\mid\lambda x:\tau.\,C[-]\mid\\
& \qquad\quad C[-]\otimes M\mid M\otimes C[-]\mid\mathbf{let}\ *\ \mathbf{be}\ C[-]\ \mathbf{in}\ M\mid\mathbf{let}\ *\ \mathbf{be}\ M\ \mathbf{in}\ C[-]\mid\\
& \qquad\quad \mathbf{let}\ x\otimes y\ \mathbf{be}\ C[-]\ \mathbf{in}\ M\mid\mathbf{let}\ x\otimes y\ \mathbf{be}\ M\ \mathbf{in}\ C[-]
\end{array}$$

Here, $a$ ranges over $\mathbb{R}$, $f$ ranges over $S$, and $x$ ranges over a countably infinite set $\mathbf{Var}$ of variables. We write $\Gamma\vdash M:\tau$ when the typing judgement is derived from the rules given in Figure 2. Evaluation rules are given in Figure 3. Since $\Lambda_S$ is a purely linear programming language, for any closed term $\vdash M:\tau$, there is a value $\vdash V:\tau$ such that $M\hookrightarrow V$. For an environment $\Gamma$ and a type $\tau$, we define $\mathbf{Term}(\Gamma,\tau)$ to be the set of all terms $M$ such that $\Gamma\vdash M:\tau$, and we define $\mathbf{Value}(\tau)$ to be the set of closed values of type $\tau$. We simply write $\mathbf{Term}(\tau)$ for $\mathbf{Term}(\varnothing,\tau)$, that is the set of closed terms of type $\tau$. For a context $C[-]$, we write $C[-]\colon(\Gamma,\tau)\to(\Delta,\sigma)$ when for all terms $\Gamma\vdash M:\tau$, we have $\Delta\vdash C[M]:\sigma$.

We adopt Church-style lambda abstraction so that every type judgement $\Gamma\vdash M:\tau$ has a unique derivation, which makes it easier to define denotational semantics for $\Lambda_S$. Except for this point, our language can be understood as a fragment of Fuzz [29] – the typing judgment $x:\sigma,\dots,y:\rho\vdash M:\tau$ corresponds to $x:_1\sigma,\dots,y:_1\rho\vdash M:\tau$ in Fuzz. In Section 9, we discuss extending our results in this paper to a richer language, closer to the one from [29].

## 3.2 Equational Theory

In this paper, we consider an equational theory for $\Lambda_S$, which will turn out to be instrumental to define a notion of well-behaving family of metrics for $\Lambda_S$ called admissibility (Section 3.3) and to give a quantitative equational theory for $\Lambda_S$ (Section 5). In both cases, if two terms are to be considered equal, then the distance between them is required to be 0. Here, we adopt the standard equational theory for the linear lambda calculus [24] extended with the following axiom

$$\frac{f\in S\quad f(a_1,\dots,a_{\mathrm{ar}(f)})=b}{\vdash\overline{f}(\overline{a_1},\dots,\overline{a_{\mathrm{ar}(f)}})=\overline{b}:\tau}\ .$$

For terms $\Gamma\vdash M:\tau$ and $\Gamma\vdash N:\tau$, we write $\Gamma\vdash M=N:\tau$ when the equality is derivable.

We may add some other axioms to the equational theory as long as the axioms are valid when we interpret function symbols $\overline{f}$ as $f$ and constant symbols $\overline{a}$ as $a$. For example, when $\mathrm{add}\colon \mathbb{R}\otimes\mathbb{R}\to\mathbb{R}$ is in $S$, we may add the commutativity law $x : \mathbf{R}, y : \mathbf{R} \vdash \overline{\mathrm{add}}(x,y) = \overline{\mathrm{add}}(y,x) : \mathbf{R}$ to the equational theory. The rest of this paper is not affected by such extensions to the equational theory.

## 3.3 Admissibility

Let us call a family $\{d_{\Gamma,\tau}\}_{\Gamma\in\mathbf{Env},\tau\in\mathbf{Ty}}$ in which $d_{\Gamma,\tau}$ is a metric on $\mathbf{Term}(\Gamma,\tau)$ a *metric on* $\Lambda_S$. We introduce a class of metrics on $\Lambda_S$, which is the object of study of this paper.

▶ **Definition 1** (Admissible Metric). *Let $\{d_{\Gamma,\tau}\}_{\Gamma\in\mathbf{Env},\tau\in\mathbf{Ty}}$ be a metric on $\Lambda_S$. We say that $\{d_{\Gamma,\tau}\}_{\Gamma\in\mathbf{Env},\tau:\mathbf{Ty}}$ is* admissible *when the following conditions hold:*

**(A1)** *For any environment $\Gamma$, any type $\tau$, any pair of terms $\Gamma\vdash M:\tau$, $\Gamma\vdash N:\tau$ and any context $C[-]\colon(\Gamma,\tau)\to(\Delta,\sigma)$, we have $d_{\Delta,\sigma}(C[M],C[N])\le d_{\Gamma,\tau}(M,N)$.*

**(A2)** *For all $a,b\in\mathbb{R}$, we have $d_{\varnothing,\mathbf{R}}(a,b)=|a-b|$.*

**(A3)** *For all $a_1,\dots,a_n,b_1,\dots,b_n\in\mathbb{R}$ and all closed values $\vdash V:\tau$ and $\vdash U:\tau$, we have*

$$d_{\varnothing,\mathbf{R}^{\otimes n}\otimes\tau}\left(\overline{a_1}\otimes\cdots\otimes\overline{a_n}\otimes V,\overline{b_1}\otimes\cdots\otimes\overline{b_n}\otimes U\right)\ge|a_1-b_1|+\cdots+|a_n-b_n|.$$

**(A4)** *If $\Gamma\vdash M=N:\tau$, then $d_{\Gamma,\tau}(M,N)=0$.*

The first condition (A1) states that all contexts are non-expansive, and the second condition (A2) states that the metric on $\mathbf{R}$ coincides with the absolute metric on $\mathbb{R}$. (A3) states that the distance between two terms $\overline{a_1}\otimes\cdots\otimes\overline{a_n}\otimes V$ and $\overline{b_1}\otimes\cdots\otimes\overline{b_n}\otimes U$ is bounded (from below) by the distance between their “observable fragments” $d_{\mathbb{R}^{\otimes n}}((a_1,\dots,a_n),(b_1,\dots,b_n))$. The last condition (A4) states that $d_{\Gamma,\tau}$ subsumes the equational theory for $\Lambda_S$.

The definition of admissibility is motivated by the study of Fuzz [29], which is a linear type system for verifying differential privacy [4]. There, Reed and Pierce introduce a syntactically defined metric on Fuzz using a family of relations called metric relations, and they prove that all programs are non-expansive with respect to the syntactic metric (Theorem 6.4 in [29]). (A1) is motivated by this result. Furthermore, in the definition of the metric relation, the tensor product of types is interpreted as the monoidal product of metric spaces, and the type of real numbers is interpreted as $\mathbb{R}$ with the absolute distance. (A2) and (A3) are motivated by these definitions. In fact, given an admissible metric $\{d_{\Gamma,\tau}\}_{\Gamma\in\mathbf{Env},\tau\in\mathbf{Ty}}$ on $\Lambda_S$, we can show that $d_{\varnothing,\mathbf{R}^{\otimes n}}$ coincides with the metric of $\mathbb{R}^{\otimes n}$:

▶ **Lemma 2.** *If a metric $\{d_{\Gamma,\tau}\}_{\Gamma\in\mathbf{Env},\,\tau\in\mathbf{Ty}}$ is admissible, then for all $a_1,b_1,\dots,a_n,b_n\in\mathbb{R}$,*

$$d_{\varnothing,\mathbf{R}^{\otimes n}}(\overline{a_1}\otimes\cdots\otimes\overline{a_n},\overline{b_1}\otimes\cdots\otimes\overline{b_n})=|a_1-b_1|+\cdots+|a_n-b_n|. \tag{1}$$

The reason that we do not take (1) as the third condition of admissibility and instead rely on the stronger condition (A3) above is that requiring (1) would not allow us to characterize the observational metric (Section 4.2) as the least admissible metric on $\Lambda_S$.

# 4 Logical Metric and Observational Metric

We give two syntactically defined metrics on $\Lambda_S$: one is based on logical relations, and the other is given in the style of Morris observational equivalence [27]. We then show that the two metrics coincide. This can be seen as a metric variant of Milner's context lemma [26].

## 4.1 Logical Metric

The first metric on $\Lambda_S$ is given by means of a quantitative form of logical relations [29] called *metric logical relations*. Here, we directly define metric logical relations, and then, we define the induced metric on $\Lambda_S$. The metric logical relations

$$\{(-) \simeq_r^\tau (-) \subseteq \mathbf{Term}(\tau) \times \mathbf{Term}(\tau)\}_{\tau \in \mathbf{Ty},\, r \in \mathbb{R}_{\geq 0}^\infty}$$

are given by induction on $\tau$ as follows.

$$\begin{aligned}
M \simeq_r^{\mathbf{R}} N &\iff M \hookrightarrow \overline{a} \text{ and } N \hookrightarrow \overline{b} \text{ and } |a-b| \leq r\\
M \simeq_r^{\mathbf{I}} N &\iff M \hookrightarrow * \text{ and } N \hookrightarrow *\\
M \simeq_r^{\tau\otimes\sigma} N &\iff M \hookrightarrow V \otimes V' \text{ and } N \hookrightarrow U \otimes U' \text{ and}\\
&\qquad \exists s, s' \in \mathbb{R}_{\geq 0}^\infty,\ V \simeq_s^\tau U \text{ and } V' \simeq_{s'}^\sigma U' \text{ and } s + s' \leq r\\
M \simeq_r^{\tau\multimap\sigma} N &\iff M \hookrightarrow \lambda x : \tau.\, M' \text{ and } N \hookrightarrow \lambda x : \tau.\, N' \text{ and}\\
&\qquad \forall V, U \in \mathbf{Value}(\tau), \text{ if } V \simeq_s^\tau U, \text{ then } M'[V/x] \simeq_{r+s}^\sigma N'[U/x]
\end{aligned}$$

Then for an environment $\Gamma = (x : \sigma, \dots, y : \rho)$ and a pair of terms $\Gamma \vdash M : \tau$ and $\Gamma \vdash N : \tau$, we define $d^{\log}_{\Gamma,\tau}(M, N) \in \mathbb{R}_{\geq 0}^\infty$ by

$$d^{\log}_{\Gamma,\tau}(M, N) = \inf\{r \in \mathbb{R}_{\geq 0}^\infty \mid \lambda x : \sigma. \cdots \lambda y : \rho.\, M \simeq_r^{\sigma\multimap\cdots\multimap\rho\multimap\tau} \lambda x : \sigma. \cdots \lambda y : \rho.\, N\}.$$

We give a consequence of our results in this paper, namely, Theorem 6 and Theorem 18. For the detail of the proof of Proposition 3, see [12].

▶ **Proposition 3.** *For any environment $\Gamma$ and any type $\tau$, the function $d^{\log}_{\Gamma,\tau}$ is a metric on $\mathbf{Term}(\Gamma, \tau)$. Furthermore, $\{d^{\log}_{\Gamma,\tau}\}_{\Gamma\in\mathbf{Env},\tau\in\mathbf{Ty}}$ is admissible.*

We call $d^{\log}$ *logical metric*. We note that we can directly check that $d^{\log}$ satisfies (A2) and (A3), and we need Theorem 6 and Theorem 18 to show that $d^{\log}$ is a metric on $\Lambda_S$ and satisfies (A1) and (A4).

▶ **Example 4.** In this example, we suppose that the addition $\mathrm{add}\colon \mathbb{R} \otimes \mathbb{R} \to \mathbb{R}$ is in $S$. Let $M$ be $\lambda k : \mathbf{R} \multimap \mathbf{R}.\, k\,\overline{1}$, and let $N$ be $\lambda k : \mathbf{R} \multimap \mathbf{R}.\, \overline{\mathrm{add}}(\overline{-2}, k\,\overline{0})$. Then

$$d^{\log}_{\varnothing,\mathbf{R}\multimap\mathbf{R}}(M, N) \leq 1 + 2 = 3$$

because we have $V\,\overline{1} \simeq_{s+1}^{\mathbf{R}} U\,\overline{0}$ for any pair $V \simeq_s^{\mathbf{R}\multimap\mathbf{R}} U$. In fact, we have $d^{\log}_{\varnothing,\mathbf{R}\multimap\mathbf{R}}(M, N) = 3$, which follows from Theorem 6. See Example 7. ⌟

▶ **Example 5.** For $a \in \mathbb{R}$, we define a term $M_a$ to be

$$\vdash \overline{a} \otimes \overline{a} \otimes V : \mathbf{R} \otimes \mathbf{R} \otimes ((\mathbf{R} \otimes \mathbf{R} \multimap \mathbf{R}) \multimap \mathbf{R}) \qquad \text{where } V = \lambda k : \mathbf{R} \otimes \mathbf{R} \multimap \mathbf{R}.\, k\,(\overline{0} \otimes \overline{0}).$$

Since $d^{\log}_{\varnothing,\mathbf{R}}(\overline{0}, \overline{1}) = 1$, we obtain $d^{\log}_{\varnothing,\mathbf{R}\otimes\mathbf{R}\otimes((\mathbf{R}\otimes\mathbf{R}\multimap\mathbf{R})\multimap\mathbf{R})}(M_0, M_1) = 1 + 1 + 0 = 2$. ⌟

## 4.2 Observational Metric

We next give a metric, which we call the *observational metric*, that measures distances between terms by observing concrete values produced by any possible context. For terms $\Gamma \vdash M : \tau$ and $\Gamma \vdash N : \tau$, we define $d^{\mathrm{obs}}_{\Gamma,\tau}(M, N) \in \mathbb{R}_{\geq 0}^\infty$ by

$$d^{\mathrm{obs}}_{\Gamma,\tau}(M, N) = \sup_{(n,\sigma,C[-])\in\mathcal{K}(\Gamma,\tau)} \left\{ |a_1 - b_1| + \cdots + |a_n - b_n| \;\middle|\; \begin{array}{l} C[M] \hookrightarrow \overline{a_1} \otimes \cdots \otimes \overline{a_n} \otimes V \\ \text{and } C[N] \hookrightarrow \overline{b_1} \otimes \cdots \otimes \overline{b_n} \otimes U \end{array} \right\}$$

where $(n, \sigma, C[-]) \in \mathcal{K}(\Gamma, \tau)$ if and only if $C[-]$ is a context from $(\Gamma, \tau)$ to $(\varnothing, \mathbf{R}^{\otimes n} \otimes \sigma)$.

$$\frac{\Gamma \vdash M = N : \tau}{\Gamma \vdash M \approx_0 N : \tau} \qquad \frac{\Gamma \vdash M \approx_r N : \tau}{\Gamma \vdash N \approx_r M : \tau} \qquad \frac{\Gamma \vdash M \approx_r N : \tau \quad \Gamma \vdash N \approx_s L : \tau}{\Gamma \vdash M \approx_{r+s} L : \tau}$$

$$\frac{|a-b| \leq r}{\vdash \overline{a} \approx_r \overline{b} : \mathbf{R}} \qquad \frac{\Gamma \vdash M \approx_r N : \tau \quad C[-] \colon (\Gamma, \tau) \to (\Delta, \sigma)}{\Delta \vdash C[M] \approx_r C[N] : \sigma}$$

**Figure 4** Derivation Rules for $\Gamma \vdash M \approx_r N : \tau$.

▶ **Theorem 6.** *For any environment* $\Gamma$ *and any type* $\tau$*, we have* $d^{\mathrm{obs}}_{\Gamma,\tau} = d^{\mathrm{log}}_{\Gamma,\tau}$.

This theorem follows from coincidence of the metric logical relations with the metric relations [29] and the fundamental lemma for metric logical relations.

▶ **Example 7.** Let $M$ and $N$ be terms given in Example 4. By observing these terms by the context $[-]\,(\lambda x : \mathbf{R}.\, x)$, we see that $d^{\mathrm{log}}_{\varnothing,(\mathbf{R}\multimap\mathbf{R})\multimap\mathbf{R}}(M, N) \geq 3$. By Theorem 6 and Example 4, we obtain $d^{\mathrm{log}}_{\varnothing,(\mathbf{R}\multimap\mathbf{R})\multimap\mathbf{R}}(M, N) = 3$. ⌟

▶ **Example 8.** We consider the term $\vdash M_a : \tau$ given in Example 5 again. By observing $M_0$ and $M_1$ by the trivial context $[-]$, we can directly check that $d^{\mathrm{obs}}_{\varnothing,\mathbf{R}\otimes\mathbf{R}\otimes((\mathbf{R}\otimes\mathbf{R}\multimap\mathbf{R})\multimap\mathbf{R})}(M_0, M_1) \geq 2$. (In fact, it follows from Theorem 6 that the distance is equal to 2.) The purpose of the auxiliary type $\sigma$ in the definition of $\mathcal{K}(\Gamma, \tau)$ is to enable observations of this type. In this case, while the logical metric distinguishes $M_0$ from $M_1$, we can not observationally distinguish $M_0$ from $M_1$ by means of observations at types $\mathbf{R}^{\otimes n}$ when $S$ is empty. This is because there is no closed term of type $\mathbf{R} \otimes \mathbf{R} \multimap \mathbf{R}$ when $S$ is empty. For a detailed explanation, see [12]. ⌟

## 5 Equational Metric

We give another syntactic metric on $\Lambda_S$, which we call the *equational metric*. This is essentially the quantitative equational theory from [9] without the rules called **weak**, **join** and **arch**. We exclude these rules since they do not affect the equational metric $d^{\mathrm{equ}}$ given below.

For terms $\Gamma \vdash M : \tau$ and $\Gamma \vdash N : \tau$, and for $r \in \mathbb{R}^{\infty}_{\geq 0}$, we write

$$\Gamma \vdash M \approx_r N : \tau$$

when we can derive the judgement from the rules in Figure 4. Then, for terms $\Gamma \vdash M : \tau$ and $\Gamma \vdash N : \tau$ we define $d^{\mathrm{equ}}_{\Gamma,\tau}(M, N) \in \mathbb{R}^{\infty}_{\geq 0}$ by

$$d^{\mathrm{equ}}_{\Gamma,\tau}(M, N) = \inf\{r \in \mathbb{R}^{\infty}_{\geq 0} \mid \Gamma \vdash M \approx_r N : \tau\}.$$

▶ **Proposition 9.** *For any environment* $\Gamma$ *and any type* $\tau$*, the function* $d^{\mathrm{equ}}_{\Gamma,\tau}$ *is a metric on* $\mathbf{Term}(\Gamma, \tau)$*. Furthermore,* $\{d^{\mathrm{equ}}_{\Gamma,\tau}\}_{\Gamma\in\mathbf{Env},\tau\in\mathbf{Ty}}$ *is admissible.*

▶ **Example 10.** The equational metric measures differences between terms by comparing their subterms. For example, we have $\vdash \overline{2} \approx_1 \overline{3} : \mathbf{R}$, and therefore, $k : \mathbf{R} \multimap \mathbf{R} \vdash k\,\overline{2} \approx_1 k\,\overline{3} : \mathbf{R}$ holds. From this, we see that $d^{\mathrm{equ}}_{(k:\mathbf{R}\multimap\mathbf{R}),\mathbf{R}}(k\,\overline{2}, k\,\overline{3}) \leq 1$. In fact, this is an equality. This follows from $d^{\mathrm{obs}}_{(k:\mathbf{R}\multimap\mathbf{R}),\mathbf{R}}(k\,\overline{2}, k\,\overline{3}) \geq 1$, which is easy to check, and Theorem 18 below. ⌟

In general, we have $d^{\mathrm{obs}}_{\Gamma,\tau}(M, N) < d^{\mathrm{equ}}_{\Gamma,\tau}(M, N)$, i.e., the equational metric is strictly more discriminating than the observational metric (Theorem 18), which is proved by semantically inspired metrics in the next section.

# 6 Models of $\Lambda_S$ and Associated Metrics

Now, we move our attention to semantically derived metrics on $\Lambda_S$. We first give a notion of models of $\Lambda_S$ based on **Met**-enriched symmetric monoidal closed categories. **Met**-enriched symmetric monoidal closed categories are studied in [9] as models of quantitative equational theories for the linear lambda calculus. Then, we give two examples of semantic metrics on $\Lambda_S$: one is based on domain theory, and the other is based on the Geometry of Interaction.

## 6.1 Met-enriched Symmetric Monoidal Closed Category

We say that a symmetric monoidal closed category $(\mathcal{C}, I, \otimes, \multimap)$ is **Met**-*enriched* when every hom-set $\mathcal{C}(X,Y)$ has the structure of a metric space subject to the following conditions:

- the composition is a morphism in **Met** from $\mathcal{C}(X,Y) \otimes \mathcal{C}(Z,X)$ to $\mathcal{C}(Z,Y)$; and
- the tensor is a morphism in **Met** from $\mathcal{C}(X,Y) \otimes \mathcal{C}(Z,W)$ to $\mathcal{C}(X \otimes Z, Y \otimes W)$; and
- the currying operation is an isomorphism in **Met** from $\mathcal{C}(X \otimes Y, Z)$ to $\mathcal{C}(X, Y \multimap Z)$.

For morphisms $f, g\colon X \to Y$ in $\mathcal{C}$, we write $d(f,g)$ for the distance between $f$ and $g$.

▶ **Definition 11.** *A* pre-model $\mathcal{M} = (\mathcal{C}, I, \otimes, \multimap, \lfloor - \rfloor)$ *of* $\Lambda_S$ *is a* **Met**-*enriched symmetric monoidal closed category* $(\mathcal{C}, I, \otimes, \multimap)$ *equipped with an object* $\lfloor \mathbf{R} \rfloor \in \mathcal{C}$ *and families of morphisms* $\{\lfloor a \rfloor \colon I \to \lfloor \mathbf{R} \rfloor\}_{a \in \mathbb{R}}$ *and* $\{\lfloor f \rfloor \colon \lfloor \mathbf{R} \rfloor^{\otimes \mathrm{ar}(f)} \to \lfloor \mathbf{R} \rfloor\}_{f \in S}$*.*

For a pre-model $\mathcal{M} = (\mathcal{C}, I, \otimes, \multimap, \lfloor - \rfloor)$ of $\Lambda_S$, we interpret types as follows:

$$[\![\mathbf{R}]\!]^{\mathcal{M}} = \lfloor \mathbf{R} \rfloor, \quad [\![\mathbf{I}]\!]^{\mathcal{M}} = I, \quad [\![\tau \otimes \sigma]\!]^{\mathcal{M}} = [\![\tau]\!]^{\mathcal{M}} \otimes [\![\sigma]\!]^{\mathcal{M}}, \quad [\![\tau \multimap \sigma]\!]^{\mathcal{M}} = [\![\tau]\!]^{\mathcal{M}} \multimap [\![\sigma]\!]^{\mathcal{M}}.$$

For an environment $\Gamma = (x : \tau, \ldots, y : \sigma)$, we define $[\![\Gamma]\!]^{\mathcal{M}}$ to be $[\![\tau]\!]^{\mathcal{M}} \otimes \cdots \otimes [\![\sigma]\!]^{\mathcal{M}}$. Then, the interpretation $[\![\Gamma \vdash M : \tau]\!]^{\mathcal{M}} \colon [\![\Gamma]\!]^{\mathcal{M}} \to [\![\tau]\!]^{\mathcal{M}}$ in $\mathcal{M}$ is given in the standard manner following [24], and constants are interpreted as follows: $[\![\vdash \overline{a} : \mathbf{R}]\!]^{\mathcal{M}} = \lfloor a \rfloor$,

$$[\![\Gamma \# \cdots \# \Delta \vdash \overline{f}(M, \ldots, N) : \mathbf{R}]\!]^{\mathcal{M}} = \lfloor f \rfloor \circ ([\![M]\!]^{\mathcal{M}} \otimes \cdots \otimes [\![N]\!]^{\mathcal{M}}) \circ \theta$$

where $\theta \colon [\![\Gamma \# \Delta]\!]^{\mathcal{M}} \xrightarrow{\cong} [\![\Gamma]\!]^{\mathcal{M}} \otimes [\![\Delta]\!]^{\mathcal{M}}$ swaps objects following the merge $\Gamma \# \Delta$.

▶ **Definition 12.** *We say that a pre-model* $\mathcal{M} = (\mathcal{C}, I, \otimes, \multimap, \lfloor - \rfloor)$ *of* $\Lambda_S$ *is a* model *of* $\Lambda_S$ *if* $\mathcal{M}$ *satisfies the following three conditions.*

- *(M1) For any* $f \in S$*, if* $f(a_1, \ldots, a_{\mathrm{ar}(f)}) = b$*, then* $[\![\overline{f}(\overline{a_1}, \ldots, \overline{a_n})]\!]^{\mathcal{M}} = [\![\overline{b}]\!]^{\mathcal{M}}$*.*
- *(M2) For all* $a, b \in \mathbb{R}$*,* $d(\lfloor a \rfloor, \lfloor b \rfloor) = |a - b|$*.*
- *(M3) For all* $x, y \colon I \to X$ *in* $\mathcal{C}$ *and all finite sequences* $a_1, \ldots, a_n, b_1, \ldots, b_n \in \mathbb{R}$*, we have*

$$d(\lfloor a_1 \rfloor \otimes \cdots \otimes \lfloor a_n \rfloor \otimes x, \lfloor b_1 \rfloor \otimes \cdots \otimes \lfloor b_n \rfloor \otimes y) \geq |a_1 - b_1| + \cdots + |a_n - b_n|.$$

The first condition corresponds to the reduction rule for function symbols and is necessary to prove soundness for models of $\Lambda_S$. The remaining conditions are for admissibility of the metric derived from models of $\Lambda_S$.

▶ **Proposition 13** (Soundness)**.** *Let* $\mathcal{M}$ *be a model of* $\Lambda_S$*. For any term* $M \in \mathbf{Term}(\tau)$ *and any value* $V \in \mathbf{Value}(\tau)$*, if* $M \hookrightarrow V$*, then* $[\![M]\!]^{\mathcal{M}} = [\![V]\!]^{\mathcal{M}}$*.*

Let $\mathcal{M} = (\mathcal{C}, I, \otimes, \multimap, \lfloor - \rfloor)$ be a model of $\Lambda_S$. For an environment $\Gamma$ and a type $\tau$, we define $d^{\mathcal{M}}_{\Gamma,\tau}$ to be the function $d([\![-]\!]^{\mathcal{M}}, [\![-]\!]^{\mathcal{M}})$ from $\mathbf{Term}(\Gamma, \tau) \times \mathbf{Term}(\Gamma, \tau)$ to $\mathbb{R}^{\infty}_{\geq 0}$.

▶ **Proposition 14.** *For any environment* $\Gamma$ *and any type* $\tau$*, the function* $d^{\mathcal{M}}_{\Gamma,\tau}$ *is a metric on* $\mathbf{Term}(\Gamma, \tau)$*. Furthermore,* $\{d^{\mathcal{M}}_{\Gamma,\tau}\}_{\Gamma \in \mathbf{Env}, \tau \in \mathbf{Ty}}$ *is admissible.*

▶ **Example 15.** The symmetric monoidal closed category **Met** of metric spaces and non-expansive functions can be extended to a model $(\mathbf{Met}, I, \otimes, \multimap, \lfloor - \rfloor)$ of $\Lambda_S$ where we define $\lfloor \mathbf{R} \rfloor \in \mathbf{Met}$ to be $\mathbb{R}$, and for $f \in S$, we define $\lfloor f \rfloor \colon \mathbb{R}^{\otimes \mathrm{ar}(f)} \to \mathbb{R}$ to be $f$. ◀

## 6.2 Denotational Metric

In this section, we recall the notion of metric cpos introduced in [2] as a denotational model of Fuzz, and we give a model of $\Lambda_S$ based on metric cpos. While we do not need the domain-theoretic nature of metric cpos to model $\Lambda_S$, we believe that the category of metric cpos is a good place to explore how metrics from denotational models and metrics from interactive semantic models are related. This is because the domain theoretic structure of the category of metric cpos directly gives rise to an interactive semantic model via Int-construction as we show in Section 6.3.2.

Let us recall the notion of (pointed) metric cpos [2].

▶ **Definition 16.** *A* (pointed) metric cpo $X$ *consists of a metric space* $(|X|, d_X)$ *with a partial order* $\leq_X$ *on* $|X|$ *such that* $(|X|, \leq_X)$ *is a (pointed) cpo, and for all ascending chains* $(x_i)_{i\in\mathbb{N}}$ *and* $(x'_i)_{i\in\mathbb{N}}$ *on* $X$*, we have* $d_X\left(\bigvee_{i\in\mathbb{N}} x_i, \bigvee_{i\in\mathbb{N}} x'_i\right) \leq \bigvee_{i\in\mathbb{N}} d_X(x_i, x'_i)$.

For metric cpos $X$ and $Y$, a function $f\colon |X| \to |Y|$ is said to be continuous and non-expansive when $f$ is a continuous function from $(|X|, \leq_X)$ to $(|Y|, \leq_Y)$ and is a non-expansive function from $(|X|, d_X)$ to $(|Y|, d_Y)$. Below, we simply write $X$ for the underlying set $|X|$.

Pointed metric cpos and continuous and non-expansive functions form a category, which is denoted by **MetCppo**. The unit object $I$ of **MetCppo** is the unit object of **Met** equipped with the trivial partial order. The tensor product $X \otimes Y$ is given by the tensor product of metric spaces with the componentwise order. The hom-object $X \multimap Y$ is given by the set of continuous and non-expansive functions equipped with the pointwise order and

$$d_{X\multimap Y}(f, g) = \sup_{x\in X} d_Y(fx, gx).$$

We associate **MetCppo** with the structure of a model of $\Lambda_S$ as follows. We define $R = \lfloor \mathbf{R} \rfloor$ to be $(\mathbb{R} \cup \{\bot\}, d_R, \leq_R)$ where $d_R$ is the extension of the metric on $\mathbb{R}$ given by $d_R(a, \bot) = \infty$ for all $a \in \mathbb{R}$, and $(\mathbb{R} \cup \{\bot\}, \leq_R)$ is the lifting of the discrete cpo $\mathbb{R}$. For $f \in S$, we define $\lfloor f \rfloor\colon R^{\otimes \mathrm{ar}(f)} \to R$ to be the function satisfying $\lfloor f \rfloor(x_1, \ldots, x_{\mathrm{ar}(f)}) = y \in \mathbb{R}$ if and only if $x_1, \ldots, x_{\mathrm{ar}(f)} \in \mathbb{R}$ and $f(x_1, \ldots, x_{\mathrm{ar}(f)}) = y$. In the sequel, we denote the metric on $\Lambda_S$ induced by this model by $d^{\mathrm{den}}$, and we call the metric $d^{\mathrm{den}}$ the *denotational metric.*

## 6.3 Interactive Metric

We describe another model of $\Lambda_S$, which we call the *interactive semantic model.* Categorically speaking, the construction is based on the notion of *trace operator* and on the related *Int-construction* [21]. Below, we first sketch how terms are interpreted, and then, we formally describe the construction of the interactive semantic model.

### 6.3.1 How Terms are Interpreted, Informally

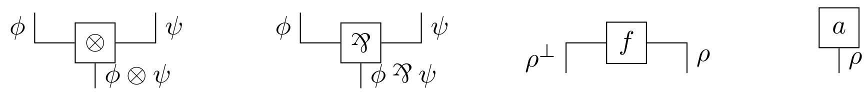
Via the Curry-Howard correspondence, our target language can be considered as the (intuitionistic) multiplicative fragment of linear logic equipped with an atomic proposition $\rho$ and derivation rules

$$\frac{}{\vdash \rho^\perp, \ldots, \rho^\perp, \rho}\, f, \qquad \frac{}{\vdash \rho}\, a$$

for each $f \in S$ and each $a \in \mathbb{R}$. Roughly speaking, the metric induced by the interactive semantic model measures distances between terms by measuring distances between the proof structures [16] associated to the corresponding proofs in this extension of linear logic; and the distances between proof structures are given by the distances between the non-expansive functions associated to proof structures. We will use the Int-construction as a categorical machinery to associate non-expansive functions to proof structures. Below, without going into categorical detail, we illustrate how we measure distances between terms using concrete examples.

For simplicity, we suppose that $S$ is the set of all non-expansive functions from $\mathbb{R}$ to $\mathbb{R}$. Then the proof structures associated to proofs are generated by the axiom links and the cut-links:

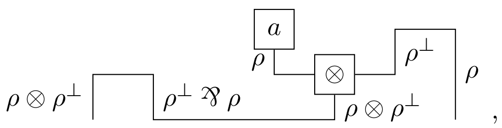

with the following nodes:



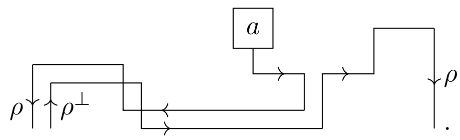

labeled by $f\colon \mathbb{R} \to \mathbb{R}$ in $S$ and $a \in \mathbb{R}$. The first two nodes are called tensor-node and par-node, respectively. The latter two nodes corresponds to derivation rules for $f \in S$ and $a \in \mathbb{R}$. Labels $\phi, \psi, \dots$ on edges are formulae of the multiplicative fragment of linear logic given by: $\phi, \psi ::= \rho \mid \rho^\perp \mid \phi \otimes \psi \mid \phi \parr \psi$. As usual, $(-)^\perp$ is an involutive operator inductively defined by $\rho^{\perp\perp} = \rho$, $(\phi \otimes \psi)^\perp = \phi^\perp \parr \psi^\perp$, and $(\phi \parr \psi)^\perp = \phi^\perp \otimes \psi^\perp$.

As an example, let us consider $M_a = \lambda k : \mathbf{R} \multimap \mathbf{R}.\, k\,\overline{a}$ for $a \in \mathbf{R}$. The proof and the proof structure corresponding to $M_a$ are

$$\dfrac{\dfrac{}{\vdash \rho \otimes \rho^\perp, \rho^\perp \parr \rho} \qquad \dfrac{\dfrac{}{\vdash \rho}\,a \qquad \dfrac{}{\vdash \rho^\perp, \rho}}{\vdash \rho \otimes \rho^\perp, \rho}}{\vdash \rho \otimes \rho^\perp, \rho} \quad ,$$



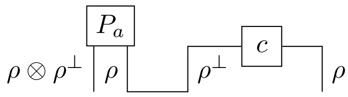

and the interactive semantic model associates this proof structure with a non-expansive function $f_a\colon \mathbb{R} \to \mathbb{R} \otimes \mathbb{R}$ given by $f_a(x) = (a, x)$. Intuitively, this non-expansive function $f$ represents information flow on the proof structure. In this case, the information flow on the proof structure can be visualized by replacing edges labeled by $\rho \otimes \rho^\perp$ and $\rho^\perp \parr \rho$ with pairs of directed wires and removing the tensor node:



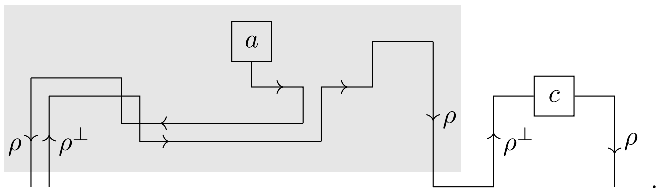

Here, we have one incoming edge labeled by $\rho^\perp$, and given an input $x$ to this edge, we will obtain a pair of outputs: one is $a$ from the left outgoing edge and the other is $x$ from the right outgoing edge. This way, we obtain the function $f_a(x) = (a, x)$. Technically speaking, this graph transformation is precisely what the Int-construction does: the Int-construction provides a way to represent undirected graphs as bidirectional information flows. Finally, we can compute the distance between $M_a$ and $M_b$ by comparing $f_a$ and $f_b$. Since $d_{\mathbb{R} \multimap \mathbb{R} \otimes \mathbb{R}}(f_a, f_b) = |a - b|$, the distance between $M_a$ and $M_b$ in the interactive semantic model is $|a - b|$.

An important feature of the interactive semantic model is that it provides an intensional view. For example, for $a \in \mathbb{R}$, let $L_a$ be $\lambda k : \mathbf{R} \multimap \mathbf{R}. \overline{c}(k\,\overline{a})$ where $c \in S$ is the constant function given by $c(x) = 0$. Then, for all $a, b \in \mathbb{R}$, while $L_a$ and $L_b$ are extensionally equivalent, the interactive semantic model distinguishes $L_a$ from $L_b$ if $a \neq b$. Below, we explain how the interactive semantic model distinguishes these terms. The proof and the proof structure corresponding to $L_a$ are

$$\dfrac{\dfrac{\Pi_a}{\vdash \rho \otimes \rho^\perp, \rho} \quad \dfrac{}{\vdash \rho^\perp, \rho}\,c}{\vdash \rho \otimes \rho^\perp, \rho} ,$$

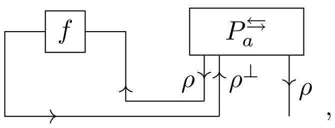



where $\Pi_a$ in the left hand side denotes the proof associated to $M_a$, and $P_a$ in the right hand side denotes the proof structure associated to $M_a$. By replacing edges in $P_a$ with bidirectional edges (depicted in the gray region in the following graph), we obtain




This graph represents $g_a \colon \mathbb{R} \to \mathbb{R} \otimes \mathbb{R}$ given by $g_a(x) = (0, a)$. Because $d_{\mathbb{R} \multimap \mathbb{R} \otimes \mathbb{R}}(g_a, g_b) = |a - b|$, we see that the distance between $L_a$ and $L_b$ in the interactive semantic model is $|a - b|$. In other words, the interactive semantic model distinguishes $L_a$ and $L_b$ by looking at their computational processes, namely, applying either $a$ or $b$ to its argument and then returning 0, rather than their extensional behavior.

Finally, we give a remark on the name *interactive* semantic model. Interaction in the interactive semantic model can be found when we consider terms that have $\beta$-redexes. For example, the bidirectional graph associated to $M_a$ $(\lambda x : \mathbf{R}. \overline{f}(x))$ is




where $P_a^{\leftrightarrows}$ is the bidirectional graph obtained from $P_a$. In this graph, we can find interaction between the node $f$ and the node $P_a^{\leftrightarrows}$.

### 6.3.2 The Interactive Semantic Model, Formally

In order to formally describe the interactive semantic model, we first observe that the category **MetCppo** has a trace operator, which is necessary to apply the Int-construction to **MetCppo**. For $f \colon X \otimes Z \to Y \otimes Z$ in **MetCppo**, we define $\mathrm{tr}^Z_{X,Y}(f) \colon X \to Y$ by

$$\mathrm{tr}^Z_{X,Y}(f)(x) = \text{the first component of } f(x, z)$$

where $z$ is the least fixed point of the continuous function $f(x, -) \colon Z \to Z$. When we ignore metric enrichment, the definition of $\mathrm{tr}^Z_{X,Y}(f)$ coincides with the definition of the trace operator associated to the least fixed point operator on the category of pointed cpos and continuous functions [18]. Hence, in order to show that $\mathrm{tr}^Z_{X,Y}$ is a trace operator, it is enough to check non-expansiveness of $\mathrm{tr}^Z_{X,Y}(f)$.

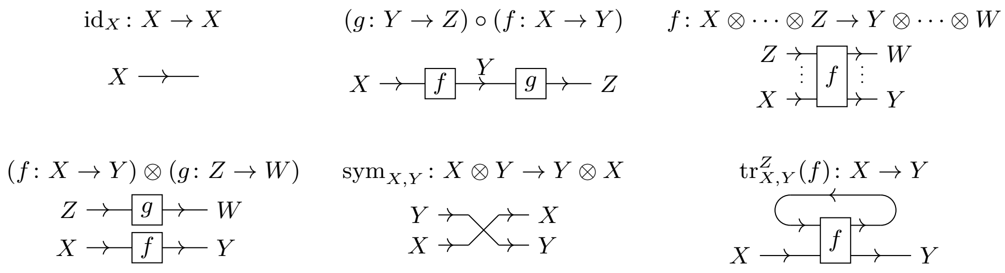



**Figure 5** String Diagrams for the Traced Symmetric Monoidal Structure.

▶ **Proposition 17.** *The symmetric monoidal category* $(\mathbf{MetCppo}, I, \otimes)$ *equipped with the family of operators* $\{\mathrm{tr}^Z_{X,Y}\}_{X,Y,Z \in \mathbf{MetCppo}}$ *is a traced symmetric monoidal category.*

Now, we can apply the Int-construction to **MetCppo** and obtain a symmetric monoidal *closed* category $\mathbf{Int}(\mathbf{MetCppo})$. (In fact, what we obtain is a *compact closed category*, and we only need its symmetric monoidal closed structure to interpret $\Lambda_S$.) Objects in $\mathbf{Int}(\mathbf{MetCppo})$ are pairs $X = (X_+, X_-)$ consisting of objects $X_+$ and $X_-$ in **MetCppo**, and a morphism from $X$ to $Y$ in $\mathbf{Int}(\mathbf{MetCppo})$ is a morphism from $X_+ \otimes Y_-$ to $X_- \otimes Y_+$ in **MetCppo**. The identity on $(X_+, X_-)$ is the symmetry $X_+ \otimes X_- \cong X_- \otimes X_+$, and the composition of $f \colon (X_+, X_-) \to (Y_+, Y_-)$ is given by

$$\mathrm{tr}^{Y_- \otimes Y_+}_{X_+ \otimes Z_-, X_- \otimes Z_+}((X_- \otimes \theta) \circ (f \otimes g) \circ (X_+ \otimes \theta'))$$

where $\theta \colon Y_+ \otimes Y_- \otimes Z_+ \to Z_+ \otimes Y_- \otimes Y_+$ and $\theta' \colon Y_- \otimes Y_+ \otimes Z_- \to Z_- \otimes Y_- \otimes Y_+$ are the canonical isomorphisms, and we omit some coherence isomorphisms. The symmetric monoidal closed structure of $\mathbf{Int}(\mathbf{MetCppo})$ is given as follows. The tensor unit is $(I, I)$, and the tensor product $X \otimes Y$ is $(X_+ \otimes Y_+, X_- \otimes Y_-)$. The hom-object $X \multimap Y$ is $(X_- \otimes Y_+, X_+ \otimes Y_-)$. For more details on the categorical structure of $\mathbf{Int}(\mathbf{MetCppo})$, see [21, 30].

We associate $\mathbf{Int}(\mathbf{MetCppo})$ with the structure of a model of $\Lambda_S$ as follows. We define $\lfloor \mathbf{R} \rfloor$ to be $(R, I)$, and for each $f \in S$, we define $\lfloor f \rfloor \colon (R, I)^{\otimes \mathrm{ar}(f)} \to (R, I)$ by

$$R^{\otimes \mathrm{ar}(f)} \otimes I \xrightarrow{\cong} R^{\otimes \mathrm{ar}(f)} \xrightarrow{\text{the interpretation of } \overline{f} \text{ in } \mathbf{MetCppo}} R \xrightarrow{\cong} I^{\otimes \mathrm{ar}(f)} \otimes R.$$

We write $d^{\mathrm{int}}$ for the metric on $\Lambda_S$ induced by the interactive semantic model, and we call $d^{\mathrm{int}}$ the *interactive metric.*

In Figure 6, we describe the interpretation of $\Lambda_S$ in $\mathbf{Int}(\mathbf{MetCppo})$ in terms of string diagrams. Here, we write $(\!|\tau|\!)_+$ and $(\!|\tau|\!)_-$ for the positive part and the negative part of the interpretation of $\tau$, and we write $(\!|\Gamma \vdash M : \tau|\!)$ for the interpretation of a term $\Gamma \vdash M : \tau$. See Figure 5 (and [30]) for the meaning of string diagrams. The interpretation $(\!|\vdash * : \mathbf{I}|\!)$ is not in Figure 6 since $(\!|\vdash * : \mathbf{I}|\!)$ is the identity on the unit object $I$, which is presented by zero wires. In the interpretation of $\overline{f}(M_1, \ldots, M_{\mathrm{ar}(f)})$, we suppose that $\mathrm{ar}(f) = 2$ for legibility.

## 7 Finding Your Way Around the Zoo

We describe how the admissible metrics on $\Lambda_S$ considered in this paper are related. Below, for metrics $d = \{d_{\Gamma,\tau}\}_{\Gamma \in \mathbf{Env}, \tau \in \mathbf{Ty}}$ and $d' = \{d'_{\Gamma,\tau}\}_{\Gamma \in \mathbf{Env}, \tau \in \mathbf{Ty}}$ on $\Lambda_S$, we write $d \leq d'$ when for all terms $\Gamma \vdash M : \tau$ and $\Gamma \vdash N : \tau$, we have $d_{\Gamma,\tau}(M, N) \leq d'_{\Gamma,\tau}(M, N)$. We write $d < d'$ when we have $d \leq d'$ and $d \neq d'$. Our main results are about the relationships between the various metrics on $\Lambda_S$, as from Figure 1.

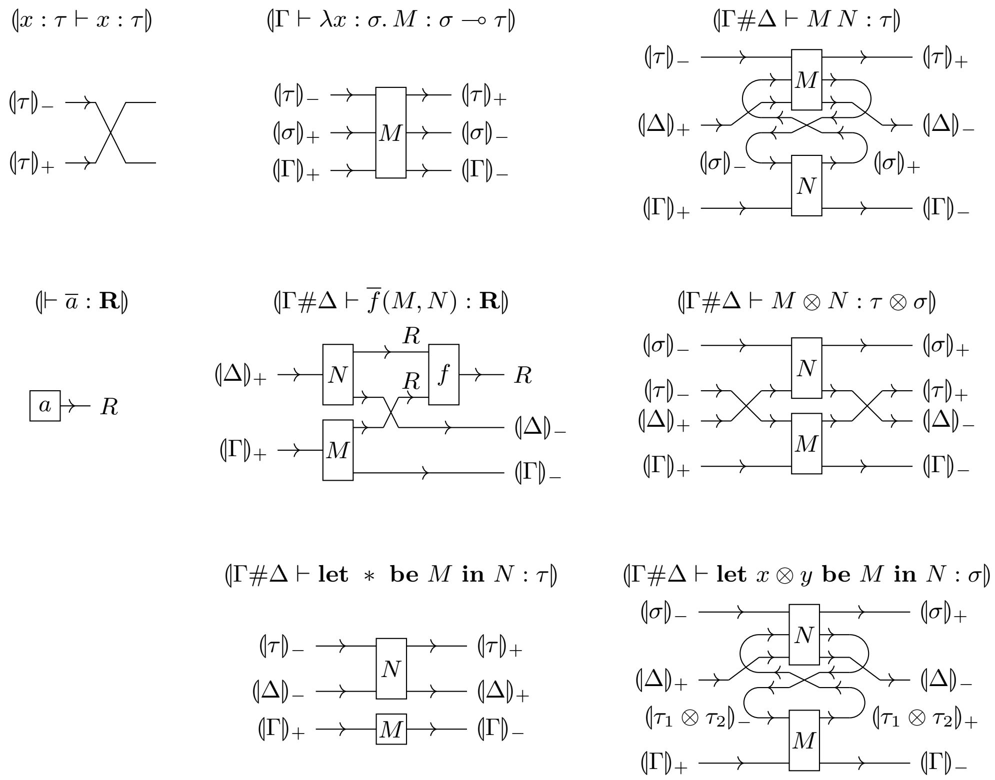



**Figure 6** The Interpretation of $\Lambda_S$ in $\mathbf{Int}(\mathbf{MetCppo})$.

▶ **Theorem 18.** *The following inclusions hold.*

1. *For any admissible metric $d$ on $\Lambda_S$, we have $d^{\log} = d^{\mathrm{obs}} \leq d \leq d^{\mathrm{equ}}$.*
2. *$d^{\log} = d^{\mathrm{obs}} \leq d^{\mathrm{den}} < d^{\mathrm{int}} \leq d^{\mathrm{equ}}$.*

The first claim in the above theorem states that the observational metric is the least admissible metric and the equational metric is the greatest admissible metric. In the proof, the conditions (A1), (A2), (A3) and (A4) in the definition of admissibility play different roles. While $d^{\mathrm{obs}} \leq d$ follows from (A1), (A3) and (A4), $d \leq d^{\mathrm{equ}}$ follows from (A1), (A2) and (A4). In the long version [12], we also show the converse of this statement. Namely, if a metric $d$ on $\Lambda_S$ satisfies (A1) and $d^{\mathrm{obs}} \leq d \leq d^{\mathrm{equ}}$, then $d$ is admissible. This implies the notion of admissibility captures reasonable class of metrics on $\Lambda_S$. The second claim in the main theorem is what illustrated in Figure 1. The inequalities $d^{\mathrm{obs}} \leq d^{\mathrm{den}}$ and $d^{\mathrm{int}} \leq d^{\mathrm{equ}}$ follow from the first claim; the proof of the strict inequality $d^{\mathrm{den}} < d^{\mathrm{int}}$ is deferred to the next section.

Concrete metrics in-between $d^{\mathrm{obs}}$ and $d^{\mathrm{equ}}$ are useful to approximately compute $d^{\mathrm{obs}}$ and $d^{\mathrm{equ}}$. For example, it is not easy to *directly* prove $d^{\mathrm{equ}}_{(k:\mathbf{R}\multimap\mathbf{I}),\mathbf{I}}(k\,\overline{2}, k\,\overline{3}) \geq 1$ since we need to know that *whenever* $k : \mathbf{R} \multimap \mathbf{I} \vdash k\,\overline{2} \approx_r k\,\overline{3} : \mathbf{I}$ is derivable, we have $r \geq 1$. Let us give a semantic proof for the inequality $d^{\mathrm{equ}}_{(k:\mathbf{R}\multimap\mathbf{I}),\mathbf{I}}(k\,\overline{2}, k\,\overline{3}) \geq 1$. Here, we use the interactive semantic model. The interpretations of these terms in the interactive semantic model are


where we can *directly* see the values applied to $k$. Hence, we obtain $d^{\mathrm{int}}_{(k:\mathbf{R}\multimap\mathbf{I}),\mathbf{I}}(k\,\overline{2}, k\,\overline{3}) = 1$. Then, the claim follows from $d^{\mathrm{int}} \leq d^{\mathrm{equ}}$.

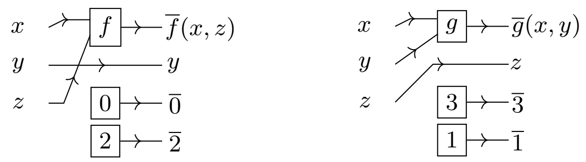



**Figure 7** String diagrams with Int-terms for $M = \overline{f}(x(y\overline{0}), z\overline{2})$ and $N = \overline{g}(x(z\overline{1}), y\overline{3})$.

## 8 Comparing the Two Denotational Viewpoints

We show that by switching from **MetCppo** to the interactive semantic model via the Int-construction, one obtains a more discriminating metric. In other words, our goal is to establish that $d^{\mathrm{den}} < d^{\mathrm{int}}$. In this section, besides the standard equational theory from Section 3, we will also make reference to the standard $\beta$-reduction relation on $\Lambda_S$.

Let us start by making the interactive semantic metric more explicit. Notably, in the case of $\beta$-normal terms, computing distances in $\mathbf{Int}(\mathbf{MetCppo})$ can be reduced to computing distances in **MetCppo** as follows: a morphism from $\Gamma$ to $\sigma$ in $\mathbf{Int}(\mathbf{MetCppo})$ is a morphism in **MetCppo** from $(\!|\Gamma|\!)_+ \otimes (\!|\sigma|\!)_+$ to $(\!|\Gamma|\!)_- \otimes (\!|\sigma|\!)_+$, where these two objects correspond to tensors of the form $\mathbf{U} \otimes \cdots \otimes \mathbf{U}$, with $\mathbf{U} \in \{\mathbf{I}, \mathbf{R}\}$, More precisely, with any list of types $\Gamma$ one can associate two natural numbers $\Gamma^+, \Gamma^-$ defined inductively as $(\emptyset)^+ = (\emptyset)^- = 0$, $(\mathbf{U} * \Gamma)^+ = 1 + \Gamma^+$, $(\mathbf{U} * \Gamma)^- = \Gamma^-$, $(\sigma \multimap \tau * \Gamma)^+ = \sigma^- + \tau^+ + \Gamma^+$, $(\sigma \multimap \tau * \Gamma)^- = \sigma^+ + \tau^- + \Gamma^-$, $(\sigma \otimes \tau * \Gamma)^+ = \sigma^+ + \tau^+ + \Gamma^+$, $(\sigma \otimes \tau * \Gamma)^- = \sigma^- + \tau^- + \Gamma^-$. Then one has the following:

▶ **Proposition 19** (First-Order Int-Terms). *Let $M, N$ be $\beta$-normal terms such that $\Gamma \vdash M, N : \sigma$ and let $m = \Gamma^+ + \sigma^-$, $n = \Gamma^- + \sigma^+$. Then there exist* first-order *linear terms $H^M_1, \dots, H^M_n$, depending on variables $x_1, \dots, x_m$, and a partition $I_1, \dots, I_m$ of $\{1, \dots, m\}$ such that:*
- *$\Gamma_j \vdash H^M_j : \mathbf{U}$, for all $j = 1, \dots, n$, where $\Gamma_j = \{x_l : \mathbf{U} \mid l \in I_j\}$, with $\mathbf{U} \in \{\mathbf{I}, \mathbf{R}\}$;*
- *$[\![M]\!]^{\mathbf{Int}(\mathbf{MetCppo})} = \bigotimes_j [\![H^M_j]\!]^{\mathbf{MetCppo}}$.*

Intuitively, the variables occurring in the left-hand of $\Gamma_j \vdash H^M_j : \mathbf{U}$ correspond to the left-hand "wires" of the string diagram representation of $[\![M]\!]^{\mathbf{Int}(\mathbf{MetCppo})}$, and the first-order term $H^M_j$ describes what exits from $i$-th right-hand "wire" of $[\![M]\!]^{\mathbf{Int}(\mathbf{MetCppo})}$.

▶ **Example 20.** Let $M = \overline{f}(x(y\overline{0}), z\overline{2})$ and $N = \overline{g}(x(z\overline{1}), y\overline{3})$, so that $\Gamma \vdash M, N : \mathbf{R}$, where $\Gamma = \{x : \mathbf{R} \multimap \mathbf{R}, y : \mathbf{R} \multimap \mathbf{R}, z : \mathbf{R} \multimap \mathbf{R}\}$. The string diagram representations of $M$ and $N$, with the associated Int-terms, are illustrated in Fig. 7. ⌟

From Proposition 19 we can now deduce the following:

▶ **Corollary 21.** *For all $\beta$-normal terms $M, N$, $d^{\mathrm{int}}(M, N) = \sum_{j=1}^{n} d^{\mathrm{den}}(H^M_j, H^N_j)$.*

For instance, in the case of Example 20, the distance $d^{\mathrm{int}}(M, N)$ coincides with the sum of the distances, computed in **MetCppo**, between the Int-terms illustrated in Fig. 7.

We can use Corollary 21 to show that the equality $d^{\mathrm{int}} = d^{\mathrm{den}}$ cannot hold. For instance, while $d^{\mathrm{den}}_{(k:\mathbf{R} \multimap \mathbf{I}), \mathbf{I}}(k\overline{2}, k\overline{3}) = 0$, by computing the Int-terms $H^{k\overline{2}}_1(x) = H^{k\overline{3}}_1(x) = x$, $H^{k\overline{3}}_2 = \overline{2}$, $H^{k\overline{3}}_2 = \overline{3}$ we deduce $d^{\mathrm{int}}_{(k:\mathbf{R} \multimap \mathbf{I}), \mathbf{I}}(k\overline{2}, k\overline{3}) = 0 + 1 = 1$.

It remains to prove then that $d^{\mathrm{den}} \leq d^{\mathrm{int}}$.

▶ **Theorem 22.** *For all $M, N$ such that $\Gamma \vdash M, N : \sigma$ holds, $d^{\mathrm{den}}(M, N) \leq d^{\mathrm{int}}(M, N)$.*

**Proof sketch.** It suffices to prove the claim for $M, N$ $\beta$-normal, using the fact that, if $M^*$ and $N^*$ are the $\beta$-normal forms of $M, N$, then $d^{\mathrm{den}}(M, M^*) = d^{\mathrm{den}}(N, N^*) = 0$, and moreover $d^{\mathrm{int}}(M^*, N^*) \leq d^{\mathrm{int}}(M, N)$, as a consequence of the non-expansiveness of the trace operator. Recall that

$$d^{\mathrm{den}}(M,N) = \sup\{d^{\mathrm{den}}_\sigma(\llbracket M\rrbracket^{\mathbf{MetCppo}}(\vec{a}), \llbracket N\rrbracket^{\mathbf{MetCppo}}(\vec{a})) \mid \vec{a} \in \llbracket\Gamma\rrbracket^{\mathbf{MetCppo}}\},$$
$$d^{\mathrm{int}}(M,N) = \sup\left\{\sum_{i=1}^n d^{\mathrm{den}}_{\mathbf{R}}(H^M_i[\vec{r}], H^N_i[\vec{r}]) \;\middle|\; \vec{r} \in \mathbb{R}^m\right\}.$$

For fixed $\vec{a} \in \llbracket\Gamma\rrbracket^{\mathbf{MetCppo}}$ we will construct reals $\vec{r} \in \mathbb{R}^m$, a sequence of terms $M = M_0, \dots, M_k = N$, where $k = \Gamma^- + \sigma^+$, and a bijection $\rho : \{1, \dots, k\} \to \{1, \dots k\}$ such that the distance between $M_i[\vec{a}]$ and $M_{i+1}[\vec{a}]$ is bounded by the distance between the Int-terms $H^M_{\rho(i+1)}[\vec{r}]$ and $H^N_{\rho(i+1)}[\vec{r}]$. In this way we can conclude by a finite number of applications of the triangular law that

$$\begin{aligned} d^{\mathrm{den}}_\sigma(M[\vec{a}], N[\vec{a}]) &\leq d^{\mathrm{den}}_\sigma(M_0[\vec{a}], M_1[\vec{a}]) + \dots + d^{\mathrm{den}}_\sigma(M_{k-1}[\vec{a}], M_k[\vec{a}]) \\ &\leq d^{\mathrm{den}}_{\mathbf{R}}(H^M_{\rho(1)}[\vec{r}], H^N_{\rho(1)}[\vec{r}]) + \dots + d^{\mathrm{den}}_{\mathbf{R}}(H^M_{\rho(k)}[\vec{r}], H^N_{\rho(k)}[\vec{r}]) \leq d^{\mathrm{int}}(M,N). \end{aligned}$$

We observe (see [12] for more details) that:

- the bound or free variables $x_i$ of $M$ are bijectively associated with the subterms $\phi_i M$ of $M$ of the form $x_i\vec{Q}$ and with first-order variables $\alpha_i$;
- the Int-terms $H^M_i[\alpha_{i_1}, \dots, \alpha_{i_s}]$ are bijectively associated with the subterms $\psi_i M$ of $M$ of the form $H^M_i[\phi_{i_1}M, \dots, \phi_{i_s}M]$.

Similar observations hold for $N$, and $\rho$ is defined so that, whenever $\psi_i N$ is a subterm of $\psi_j N$, $\phi(j) \leq \phi(i)$. Let us set $r_{\alpha_j} := \phi_j M[\vec{a}]$. The desired sequence is defined by letting $M_0 = M$ and $M_{i+1}$ be obtained from $M_i$ by replacing the subterm $\psi_{\rho(i)}M = H^M_{\rho(i)}[\phi_{i_1}M, \dots, \phi_{i_s}M]$ by $H^N_{\rho(i)}[\phi_{i_1}M, \dots, \phi_{i_s}M]$. Using the properties of $\rho$, one can check that this replacement is well-defined at each step, and that $M_k$ actually coincides with $N$. Moreover, at each step the passage from $M_i$ to $M_{i+1}$ is bounded in distance by

$$\begin{aligned} d^{\mathrm{den}}(H^M_{\rho(i)}[\dots\phi_j M[\vec{a}]\dots], H^N_{\rho(i)}[\dots\phi_j M[\vec{a}]\dots]) &= d^{\mathrm{int}}(H^M_{\rho(i)}[\dots r_{\alpha_j}\dots], H^N_{\rho(i)}[\dots r_{\alpha_j}\dots]) \\ &\leq d^{\mathrm{int}}(H^M_{\rho(i)}, H^N_{\rho(i)}). \end{aligned}$$ ◀

▶ **Example 23.** For the terms $M$ and $N$ from Example 20, the procedure just sketched defines the sequence: $M = \overline{f}(x(y\overline{0}), z\overline{2}) \overset{\overline{f}(x,z)\mapsto\overline{g}(x,y)}{\to} \overline{g}(x(y\overline{0}), y\overline{0}) \overset{y\mapsto z}{\to} \overline{g}(x(z\overline{2}), y\overline{0}) \overset{\overline{0}\mapsto\overline{3}}{\to} \overline{g}(x(z\overline{2}), y\overline{3}) \overset{\overline{2}\mapsto\overline{1}}{\to} \overline{g}(x(z\overline{1}), y\overline{3}) = N$, where at each step the replacement is of the form $H^M_i[\dots\varphi_j M\dots] \mapsto H^N_i[\dots\varphi_j M\dots]$. ⌟

While the argument above holds in the linear case, it does not seem to scale to graded exponentials, and in this last case we are not even sure if a result like Theorem 22 may actually hold (see also the discussion in the next section).

## 9 On Graded Exponentials

One of the (original) motivations of this paper was to study distances between programs in Fuzz [29], which is a linear type system designed to track *program sensitivity* . A function $f : \mathbb{R} \to \mathbb{R}$ is said to be $r$-sensitive for $r \in \mathbb{R}^{\infty}_{\geq 0}$ when $f$ is a non-expansive function from $!_r\mathbb{R} \to \mathbb{R}$ where $!_r\mathbb{R}$ is the metric space of real numbers with $d_{!_r\mathbb{R}}(a, b) = r|a - b|$. Sensitivity tells how much a function depends on its arguments. Since Fuzz adopts the scaling modalities

$!_r(-)$ as type constructors, when we are to extend our results in this paper to Fuzz, we need to take care of the scaling modalities. In this section, we give some hints about which ones of our results can be extended to Fuzz.

As for Theorem 6, we can show that the metric given in [29] in terms of metric logical relations equals the observational metric $d^{\mathrm{obs}}_{\Gamma,\tau}(M,N)$ given by

$$\sup_{C[-]\colon(\Gamma,\tau)\to(\varnothing,\mathbf{R})} \inf\left\{ r\in\mathbb{R}^{\infty}_{\geq 0} \;\middle|\; \begin{array}{l}\text{if } C[M]\hookrightarrow a,\ \text{then } C[N]\hookrightarrow b \text{ and } |a-b|\leq r,\\ \text{and if } C[N]\hookrightarrow a,\ \text{then } C[M]\hookrightarrow b \text{ and } |a-b|\leq r\end{array}\right\}$$

as long as we equip Fuzz with unary multiplications $r\times(-)\colon !_r\mathbf{R}\to\mathbf{R}$ for all $r\in\mathbb{R}_{\geq 0}$. For Fuzz *without* unary multiplications, we need to define observational metric by observations at types $!_{r_1}\mathbf{R}\otimes\cdots\otimes !_{r_n}\mathbf{R}$ in order to prove that the observational metric coincides with the logical metric. We can also extend the over-approximation of the observational metric by the denotational metric to Fuzz. This follows from the adequacy theorem of **MetCppo** with respect to Fuzz shown in [2]. In **MetCppo**, the scaling modalities $!_r(-)$ of Fuzz are interpreted as scaling operators $r\cdot(-)$: for any metric space $X$, $r\cdot X$ is a metric space defined to be $|X|$ with $d_{r\cdot X}(x,y)=rd_X(x,y)$. Unfortunately, generalizing the over-approximation of the observational metric by the interactive semantic metric to Fuzz is not done yet. The main difficulty lies in the interpretation of the scaling modalities $!_r(-)$. Since the scaling modalities can be understood as a graded variant of the linear exponential comonad in linear logic [22], it is reasonable to explore graded variants of Abramsky and Jagadeesan's a model of linear logic based on **Int**(**Cppo**) [1]. However, at this point, we could only accommodate grades as non-negative possibly infinite *integers* [12]. We believe that this restriction is not so strong because, for example, closed terms of type $!_{k/n}\mathbf{R}\multimap !_{h/m}\mathbf{R}$ in Fuzz are "definable" as closed terms of type $!_{km}\mathbf{R}\multimap !_{hn}\mathbf{R}$.

## 10 Conclusion

In this paper, we study quantitative reasoning about linearly typed higher-order programs. We introduce a notion of admissibility for families of metrics on a purely linear programming language $\Lambda_S$, and among them, we investigate five notions of program metrics and how these are related, namely the logical, observational, equational, denotational, and interactive metrics. Some of our results can be seen as quantitative analogues of well-known results about program equivalences: the observational metric is never more discriminating than the semantic metrics, and non-definable functionals in the semantics are the source of inclusions. We list some open problems:

- Does the denotational metric coincide with the observational metric?
- Does the interactive metric coincide with the equational metric?
- We may define another observational metric $d^{\mathrm{obsbase}}$ where we observe terms only at $\mathbf{R}$. This observational metric $d^{\mathrm{obsbase}}$ is less than or equal to the logical metric and depends on the choice of $S$. For which choices of $S$ does $d^{\mathrm{obsbase}}$ coincide with the logical metric?
- We have another semantic metric obtained from the category of metric spaces. How is the metric related to the denotational metric?
- There is a symmetric monoidal coreflection between **Int**(**MetCppo**) and **MetCppo** [19]. This is a strong connection between the two models. However, we do not know whether this categorical structure sheds any light on their relationship at the level of higher-order programs.

On the last point, we can say that our study reveals the intrinsic difficulty of comparing denotational models with interactive semantic models obtained by applying the Int-construction. Indeed, their relationship is not trivial already at the level of program equivalences.

Some of our results can be extended to a fragment of Fuzz where grading is restricted to extended natural numbers. Providing a quantitative equational theory and an interactive metric for full Fuzz is another very interesting topic for future work. There are some notions of metric that we have not taken into account in this paper. In [14], Gavazzo gives coinductively defined metrics for an extension of Fuzz with algebraic effects and recursive types, which we do not consider here. The so-called observational quotient [20] can be seen as a way to construct less discriminating program metrics from fine-grained ones. A thorough comparison of these notions of program distance with the ones we introduce here is another intriguing problem on which we plan to work in the future.

## References


1 S. Abramsky and R. Jagadeesan. New foundations for the geometry of interaction. *Information and Computation*, 111(1):53–119, 1994. doi:10.1006/inco.1994.1041.

2 Arthur Azevedo de Amorim, Marco Gaboardi, Justin Hsu, Shin-ya Katsumata, and Ikram Cherigui. A semantic account of metric preservation. In *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL 2017)*, pages 545–556, New York, NY, USA, 2017. Association for Computing Machinery.

3 Andrew Barber and Gordon D. Plotkin. Dual intuitionistic linear logic, 1996. Unpublished draft. An early version appeared as a technical report ECS-LFCS-96-347, LFCS, University of Edinburgh.

4 Avrim Blum, Cynthia Dwork, Frank McSherry, and Kobbi Nissim. Practical privacy: The sulq framework. In *Proceedings of the 24th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS 2005)*, pages 128–138, New York, NY, USA, 2005. Association for Computing Machinery. doi:10.1145/1065167.1065184.

5 Raphaëlle Crubillé and Ugo Dal Lago. On probabilistic applicative bisimulation and call-by-value $\lambda$-calculi. In Zhong Shao, editor, *Proceedings of the 23rd European Symposium on Programming Languages and Systems (ESOP 2014)*, volume 8410 of *Lecture Notes in Computer Science*, pages 209–228. Springer, 2014. doi:10.1007/978-3-642-54833-8_12.

6 Raphaëlle Crubillé and Ugo Dal Lago. Metric reasoning about $\lambda$-terms: The affine case. In *Proceedings of the 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2015)*, pages 633–644, USA, 2015. IEEE Computer Society. doi:10.1109/LICS.2015.64.

7 Raphaëlle Crubillé and Ugo Dal Lago. Metric reasoning about $\lambda$-terms: The general case. In Hongseok Yang, editor, *Proceedings of the 26th European Symposium on Programming Languages and Systems (ESOP 2017)*, Lecture Notes in Computer Science, pages 341–367, Berlin, Heidelberg, 2017. Springer Berlin Heidelberg. doi:10.1007/978-3-662-54434-1_13.

8 Raphaëlle Crubillé, Ugo Dal Lago, Davide Sangiorgi, and Valeria Vignudelli. On applicative similarity, sequentiality, and full abstraction. In Roland Meyer, André Platzer, and Heike Wehrheim, editors, *Proceedings of Correct System Design - Symposium in Honor of Ernst-Rüdiger Olderog on the Occasion of His 60th Birthday, Oldenburg, Germany, September 8-9, 2015*, volume 9360 of *Lecture Notes in Computer Science*, pages 65–82. Springer, 2015. doi:10.1007/978-3-319-23506-6_7.

9 Fredrik Dahlqvist and Renato Neves. An internal language for categories enriched over generalised metric spaces. In Florin Manea and Alex Simpson, editors, *Proceedings of the 30th EACSL Annual Conference on Computer Science Logic, (CSL 2022)*, volume 216 of *LIPIcs*, pages 16:1–16:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.CSL.2022.16.

10 Ugo Dal Lago and Francesco Gavazzo. Effectul program distancing. *Proceedings of the ACM on Programming Languages*, 6(POPL):1–30, 2022. doi:10.1145/3498680.

11 Ugo Dal Lago, Furio Honsell, Marina Lenisa, and Paolo Pistone. On Quantitative Algebraic Higher-Order Theories. In Amy P. Felty, editor, *Proceedings of the 7th International Conference on Formal Structures for Computation and Deduction (FSCD 2022)*, volume 228 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 4:1–4:18, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.FSCD.2022.4.

12 Ugo Dal Lago, Naohiko Hoshino, and Paolo Pistone. On the lattice of program metrics, 2023. URL: https://arxiv.org/abs/2302.05022.

13 Thomas Ehrhard, Michele Pagani, and Christine Tasson. Full abstraction for probabilistic PCF. *Journal of the ACM*, 65(4), 2018. doi:10.1145/3164540.

14 Francesco Gavazzo. Quantitative behavioural reasoning for higher-order effectful programs: Applicative distances. In *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2018)*, pages 452–461, New York, NY, USA, 2018. Association for Computing Machinery. doi:10.1145/3209108.3209149.

15 Guillaume Geoffroy and Paolo Pistone. A partial metric semantics of higher-order types and approximate program transformations. In *Proceedings of the 29th EACSL Annual Conference on Computer Science Logic (CSL 2021)*, volume 183 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 35:1–35:18, Dagstuhl, Germany, 2021. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.CSL.2021.23.

16 Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50(1):1–101, 1987.

17 Jean-Yves Girard. Geometry of interaction 1: Interpretation of System F. In R. Ferro, C. Bonotto, S. Valentini, and A. Zanardo, editors, *Proceedings of the Logic Colloquium '88*, volume 127 of *Studies in Logic and the Foundations of Mathematics*, pages 221–260. North-Holland, 1989.

18 Masahito Hasegawa. *Models of Sharing Graphs: A Categorical Semantics of Let and Letrec.* Springer-Verlag, Berlin, Heidelberg, 1999.

19 Masahito Hasegawa. On traced monoidal closed categories. *Mathematical Structures in Computer Science*, 19(2):217–244, 2009. doi:10.1017/S0960129508007184.

20 J. M. E. Hyland and C. H. L. Ong. On Full Abstraction for PCF: I, II, and III. *Information and Computation*, 163(2):285–408, 2000. doi:10.1006/inco.2000.2917.

21 André Joyal, Ross Street, and Dominic Verity. Traced monoidal categories. *Mathematical Proceedings of the Cambridge Philosophical Society*, 119(3):447–468, 1996.

22 Shin-ya Katsumata. A double category theoretic analysis of graded linear exponential comonads. In Christel Baier and Ugo Dal Lago, editors, *Proceedings of the 21st International Conference on Foundations of Software Science and Computation Structures (FOSSACS 2018)*, pages 110–127, Cham, 2018. Springer International Publishing. doi:10.1007/978-3-319-89366-2_6.

23 Soren Bogh Lassen. *Relational Reasoning about Functions and Nondeterminism.* PhD thesis, University of Aarhus, available at https://www.brics.dk/DS/98/2/BRICS-DS-98-2.pdf, 1998.

24 Ian Mackie, Leopoldo Román, and Samson Abramsky. An internal language for autonomous categories. *Applied Categorical Structures*, 1:311–343, 1993.

25 Radu Mardare, Prakash Panangaden, and Gordon Plotkin. Quantitative algebraic reasoning. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2016)*, pages 1–10, New York, NY, USA, 2016.

26 Robin Milner. Fully abstract models of typed $\lambda$-calculi. *Theoretical Computer Science*, 4(1):1–22, 1977. doi:10.1016/0304-3975(77)90053-6.

27 James Hiram Morris. *Lambda-calculus models of programming languages.* PhD thesis, Massachusetts Institute of Technology, available at https://dspace.mit.edu/handle/1721.1/64850, 1969.

28 Paolo Pistone. On generalized metric spaces for the simply typed $\lambda$-calculus. In *Proceedings of the 36th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2021)*, pages 1–14. IEEE Computer Society, 2021. doi:10.1109/LICS52264.2021.9470696.

**29** Jason Reed and Benjamin C. Pierce. Distance makes the types grow stronger: A calculus for differential privacy. In *Proceedings of the 15th ACM SIGPLAN International Conference on Functional Programming (ICFP 2010)*, volume 45, pages 157–168, New York, NY, USA, 2010. Association for Computing Machinery. `doi:10.1145/1932681.1863568`.

**30** Peter Selinger. A survey of graphical languages for monoidal categories. In Bob Coecke, editor, *New Structures for Physics*, volume 813 of *Lecture Notes in Physics*, pages 289–355. Springer-Verlag Berlin Heidelberg, 2011.