On Higher-Order Probabilistic Subrecursion (Long Version)*

Flavien Breuvart Ugo Dal Lago Agathe Herrou

Abstract

We study the expressive power of subrecursive probabilistic higher-order calculi. More specifically, we show that endowing a very expressive deterministic calculus like Gödel's $\mathbb{T}$ with various forms of probabilistic choice operators may result in calculi which are not equivalent as for the class of distributions they give rise to, although they all guarantee *almost-sure* termination. Along the way, we introduce a probabilistic variation of the classic reducibility technique, and we prove that the simplest form of probabilistic choice leaves the expressive power of $\mathbb{T}$ essentially unaltered. The paper ends with some observations about functional expressivity: expectedly, all the considered calculi represent precisely the functions which $\mathbb{T}$ itself represents.

1 Introduction

Probabilistic models are more and more pervasive in computer science and are among the most powerful modeling tools in many areas like computer vision [19], machine learning [18] and natural language processing [16]. Since the early times of computation theory [7], the very concept of an algorithm has been itself generalised from a purely deterministic process to one in which certain elementary computation steps can have a probabilistic outcome. This has further stimulated research in computation and complexity theory [10], but also in programming languages [20].

Endowing programs with probabilistic primitives (e.g. an operator which models sampling from a distribution) poses a challenge to programming language semantics. Already for a minimal, imperative probabilistic programming language, giving a denotational semantics is nontrivial [15]. When languages also have higher-order constructs, everything becomes even harder [13] to the point of disrupting much of the beautiful theory known in the deterministic case [1]. This has stimulated research on denotational semantics of higher-order probabilistic programming languages, with some surprising positive results coming out recently [8, 3].

Not much is known about the expressive power of *probabilistic* higher-order calculi, as opposed to the extensive literature on the same subject about *deterministic* calculi (see, e.g. [23, 22]). What happens to the class of representable functions if one enrich, say, a deterministic λ-calculus $\mathbb{X}$ with certain probabilistic choice primitives? Are the expressive power or the good properties of $\mathbb{X}$ somehow preserved? These questions have been given answers in the case in which $\mathbb{X}$ is the pure, untyped, λ-calculus [5]: in that case, univesality continues to hold, mimicking what happens in Turing machines [21]. But what if $\mathbb{X}$ is one of the many typed λ-calculi ensuring strong normalisation for typed terms [11]?

But let us do a step back, first: when should a higher-order probabilistic program be considered terminating? The question can be given a satisfactory answer being inspired by, e.g., recent works on probabilistic termination in imperative languages and term rewrite systems [17, 2]: one could ask the probability of divergence to be 0, called *almost sure termination* property, or the stronger *positive almost sure termination*, in which one requires the average number of evaluation steps to be finite. That termination is desirable property, even in a probabilistic setting can be seen, e.g. in

*The authors are partially supported by ANR project 14CE250005 ELICA and ANR project 12IS02001 PACE.

the field of languages like CHURCH and ANGLICAN, in which programs are often assumed to be almost surely terminating, e.g. when doing inference by MH algorithms [12].

In this paper, we initiate a study on the expressive power of terminating higher-order calculi, in particular those obtained by endowing Gödel's $\mathbb{T}$ with various forms of probabilistic choice operators. In particular, three operators will be analyzed in this paper:

- A binary probabilistic operator $\oplus$ such that for every pair of terms M, N, the term $M \oplus N$ evaluates to either M or N, each with probability $\frac{1}{2}$. This is a rather minimal option, which, however, guarantees universality if applied to the untyped λ-calculus [5] (and, more generally, to universal models of computation [21]).
- A combinator $\mathtt{R}$, which evaluates to any natural number $n \geq 0$ with probability $\frac{1}{2^{n+1}}$. This is the natural generalization of $\oplus$ to sampling from a distribution having *countable* rather than *finite* support. This apparently harmless generalization (which is absolutely non-problematic in a universal setting) has dramatic consequences in a subrecursive scenario, as we will discover soon.
- A combinator $\mathtt{X}$ such that for every pair of values V, W, the term $\mathtt{X}\, V\, W$ evaluates to either W or $V(\mathtt{X}\, V\, W)$, each with probability $\frac{1}{2}$. The operator $\mathtt{X}$ can be seen as a probabilistic variation on $\mathbb{PCF}$'s fixpoint combinator. As such, $\mathtt{X}$ is potentially problematic to termination, giving rise to infinite trees.

This way, various calculi can be obtained, like $\mathbb{T}^{\oplus}$, namely a minimal extension of $\mathbb{T}$, or the full calculus $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$, in which the three operators are all available. In principle, the only obvious fact about the expressive power of the above mentioned operators is that both $\mathtt{R}$ and $\mathtt{X}$ are at least as expressive as $\oplus$: binary choice can be easily expressed by either $\mathtt{R}$ or $\mathtt{X}$. Less obvious but still easy to prove is the equivalence between $\mathtt{R}$ and $\mathtt{X}$ in presence of a recursive operator (see Section 3.4). But how about, say, $\mathbb{T}^{\oplus}$ vs. $\mathbb{T}^{\mathtt{R}}$?

Traditionally, the expressivities of such languages are compared by looking at the set of functions $f : \mathbb{N} \to \mathbb{N}$ defined by typable programs $M : \mathtt{NAT} \to \mathtt{NAT}$. However, in probabilistic setting, programs $M : \mathtt{NAT} \to \mathtt{NAT}$ computes functions from natural numbers into *distributions* of natural numbers. In order to fit usual criterions, we need to fix a notion of observation. There are at least two relevant notions of observations, corresponding to two randomised programming paradigms, namely *Las Vegas* and *Monte Carlo* observations. The main question, then, consists in understanding how the obtained classes relate to each other, and to the class of $\mathbb{T}$-representable functions. Along the way, however, we manage to understand how to capture the expressive power of probabilistic calculi *per se*. This paper's contributions can be summarised as follows:

- We first take a look at the full calculus $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$, and prove that it enforces almost-sure termination, namely that the probability of termination of any typable term is 1. This is done by appropriately adapting the well-known reducibility technique [11] to a probabilistic operational semantics. We then observe that while $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ cannot be *positively* almost surely terminating, $\mathbb{T}^{\oplus}$ indeed is. This already shows that there must be a gap in expressivity. This is done in Section 3.
- In Section 4, we look more precisely at the expressive power of $\mathbb{T}^{\oplus}$, proving that the mere presence of probabilistic choice does not add much to the expressive power of $\mathbb{T}$: in a sense, probabilistic choice can be "lifted up" to the ambient deterministic calculus.
- We look at other fragments of $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ and at their expressivity. More specifically, we will prove that (the equiexpressive) $\mathbb{T}^{\mathtt{R}}$ and $\mathbb{T}^{\mathtt{X}}$ represent precisely what $\mathbb{T}^{\oplus}$ can do *at the limit*, in a sense which will be made precise in Section 3. This part, which is the most challenging, is done in Section 5.
- Section 6 is devoted to proving that both for *Monte Carlo* and for *Las Vegas* observations, the class of representable functions of $\mathbb{T}^{\mathtt{R}}$ coincides with the $\mathbb{T}$-representable ones.

2 Probabilistic Choice Operators, Informally

Any term of Gödel's $\mathbb{T}$ can be seen as a purely deterministic computational object whose dynamics is finitary, due to the well-known strong normalization theorem (see, e.g., [11]). In particular, the apparent non-determinism due to multiple redex occurrences is completely harmless because of confluence. In this paper, indeed, we even neglect this problem, and work with a reduction strategy, namely weak call-by-value reduction (keeping in mind that all what we will say also holds in call-by-name). Evaluation of a $\mathbb{T}$-term M of type $\mathtt{NAT}$ can be seen as a finite sequence of terms ending in the normal form $\boldsymbol{n}$ of M (see Figure 1). More generally, the unique normal form of any term $\mathbb{T}$ term M will be denoted as $[\![M]\!]$. Noticeably, $\mathbb{T}$ is computationally very powerful. In particular, the $\mathbb{T}$-representable functions from $\mathbb{N}$ to $\mathbb{N}$ coincide with the functions which are provably total in Peano's arithmetic [11].

As we already mentioned, the most natural way to enrich deterministic calculi and turn them into probabilistic ones consists in endowing their syntax with one or more probabilistic choice operators. Operationally, each of them models the essentially stochastic process of sampling from a distribution and proceeding depending on the outcome. Of course, one has many options here as for *which* of the various operators to grab. The aim of this work is precisely to study to which extend this choice have consequences on the overall expressive power of the underlying calculus.

Suppose, for example, that $\mathbb{T}$ is endowed with the binary probabilistic choice operator $\oplus$ described in the Introduction, whose evaluation corresponds to tossing a fair coin and choosing one of the two arguments accordingly. The presence of $\oplus$ has indeed an impact on the dynamics of the underlying calculus: the evaluation of any term M is not deterministic anymore, but can be modeled as a finitely branching tree (see, e.g. Figure 3 for such a tree when M is $(\mathbf{3} \oplus \mathbf{4}) \oplus \mathbf{2}$). The fact that all branches of this tree have finite height (and the tree is thus finite) is intuitive, and a proof of it can be given by adapting the well-known reducibility proof of termination for $\mathbb{T}$. In this paper, we in fact prove much more, and establish that $\mathbb{T}^{\oplus}$ can be embedded into $\mathbb{T}$.

If $\oplus$ is replaced by $\mathtt{R}$, the underlying tree is not finitely branching anymore, but, again, there is not (at least apparently) any infinitely long branch, since each of them can somehow be seen as a $\mathbb{T}$ computation (see Figure 2 for an example). What happens to the expressive power of the obtained calculus? Intuition tells us that the calculus should not be too expressive viz. $\mathbb{T}^{\oplus}$. If $\oplus$ is replaced by $\mathtt{X}$, on the other hand, the underlying tree *is* finitely branching, but its height can well be infinite. Actually, $\mathtt{X}$ and $\mathtt{R}$ are easily shown to be equiexpressive in presence of higher-order recursion, as we show in Section 3.4. On the other hand, for $\mathtt{R}$ and $\oplus$, no such encoding is available. Nonetheless, $\mathbb{T}^{\mathtt{R}}$ can still be somehow encoded embedded into $\mathbb{T}$ (just that we need an infinite structure) as we will detail in Section 5. From this embeding, we can show that applying Monte Carlo or Las Vegas algorithm on $\mathbb{T}^{\oplus,\mathtt{X},\mathtt{R}}$ results do not add any expressive power to $\mathbb{T}$, This is done in Section 6.

3 The Full Calculus $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$

All along this paper, we work with a calculus $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ whose *terms* are the ones generated by the following grammar:

$$M, N, L ::= x \mid \lambda x.M \mid M\ N \mid \langle M, N\rangle \mid \pi_1 \mid \pi_2 \mid \mathtt{rec} \mid \mathbf{0} \mid \mathtt{S} \mid M \oplus N \mid \mathtt{R} \mid \mathtt{X}.$$

Please observe the presence of the usual constructs from the untyped λ-calculus, but also of primitive recursion, constants for natural numbers, pairs, and the three choice operators we have described in the previous sections.

As usual, terms are taken modulo α-equivalence. Terms in which no variable occurs free are, as usual, dubbed *closed*, and are collected in the set $\mathbb{T}_C^{\oplus,\mathtt{R},\mathtt{X}}$. A *value* is simply a closed term from the following grammar

$$U, V ::= \lambda x.M \mid \pi_1 \mid \pi_2 \mid \langle U, V\rangle \mid \mathtt{rec} \mid \mathbf{0} \mid \mathtt{S} \mid \mathtt{S}\ V \mid \mathtt{X},$$

and the set of values is $\mathbb{T}_V^{\oplus,\mathtt{R},\mathtt{X}}$. *Extended values* are (not necessarily closed) terms generated by the same grammar as values with the addition of variables. Closed terms that are not values are called *reducible* and their set is denoted $\mathbb{T}_R^{\oplus,\mathtt{R},\mathtt{X}}$. A *context* is a term with a unique hole:

$$C := (\!|\cdot|\!) \mid \lambda x.C \mid C\ M \mid M\ C \mid \langle C, M\rangle \mid \langle M, C\rangle \mid C \oplus M \mid M \oplus C$$

We write $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}_{(\!|\cdot|\!)}$ for the set of all such contexts.

Termination of Gödel's $\mathbb{T}$ is guaranteed by the presence of types, which we also need here. *Types* are expressions generated by the following grammar

$$A, B ::= \mathtt{NAT} \;\Big|\; A \to B \;\Big|\; A \times B.$$

Environmental contexts are expressions of the form $\Gamma = x_1 : A_1, \ldots, x_n : A_n$, while *typing judgments* are of the form $\Gamma \vdash M : A$. *Typing rules* are given in Figure 5. From now on, only typable terms will be considered. We denote by $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}(A)$ the set of terms of type A, and similarly for $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}_C(A)$ and $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}_V(A)$. We use the shortcut $\boldsymbol{n}$ for values of type NAT: $\mathbf{0}$ is already part of the language of terms, while $\boldsymbol{n+1}$ is simply $\mathtt{S}\ \boldsymbol{n}$. For simplicity of notations, we also denote SS for $\lambda x.\mathtt{S}(\mathtt{S}\ x)$; we do the same for SSS...

3.1 Operational Semantics

While evaluating terms in a deterministic calculus ends up in a *value*, the same process leads to a *distribution* of values when performed on terms in a probabilistic calculus. Keep in mind that despite speaking of distribution, we are only treating countable distributions and not continuous one, which is much simpler. Formalizing all this requires some care, but can be done following one of the many definitions from the literature (e.g., [5]).

Given a countable set X, a *distribution* $\mathcal{L}$ *on* X is a probabilistic (sub)distribution over elements of X:

$$\mathcal{L}, \mathcal{M}, \mathcal{N} \in \mathfrak{D}(X) = \Big\{ f : X \to [0,1] \;\Big|\; \sum_{x \in X} f(x) \leq 1 \Big\}$$

We are especially concerned with distributions over terms here. In particular, a *distribution of type A* is simply an element of $\mathfrak{D}(\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}(A))$. The set $\mathfrak{D}(\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}_V)$ is ranged over by metavariables like $\mathcal{U}, \mathcal{V}, \mathcal{W}$. We will use the pointwise order $\leq$ on distributions, which turns them into an $\omega\boldsymbol{CPO}$. Moreover, we use the following notation for Dirac's distributions over terms: $\{M\} := \left\{ \begin{matrix} M \mapsto 1 \\ N \mapsto 0 \text{ if } M \neq N \end{matrix} \right\}$. The support of a distribution is indicated as $|\mathcal{M}|$; we also define the reducible and value supports fragments as $|\mathcal{M}|_R := |\mathcal{M}| \cap \mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}_R$ and $|\mathcal{M}|_V := |\mathcal{M}| \cap \mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}_V$. Notions like $\mathcal{M}^R$ and $\mathcal{M}^V$ have an obvious and natural meaning: for any $\mathcal{M} \in \mathfrak{D}(X)$ and $Y \subseteq X$, then $\mathcal{M}^Y(x) = \mathcal{M}(x)$ if $x \in Y$ and $\mathcal{M}^Y(x) = 0$ otherwise.

As syntactic sugar, we use integral notations to manipulate distributions, *i.e.*, for any family of distributions $(\mathcal{N}_M)_{M \in \mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}} : \mathfrak{D}(\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}})^{\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}}$, the expression $\int_{\mathcal{M}} \mathcal{N}_M.dM$ stands for

$$\int_{\mathcal{M}} \mathcal{N}_M.dM \;:=\; \sum_{M \in \mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}} \mathcal{M}(M) \cdot \mathcal{N}_M.$$

y abuse of notation, we may define $\mathcal{N}_M$ only for $M \in |\mathcal{M}|$, since the others are not used anyway.

Example 1. *Suppose that* $\mathcal{M} = \{\boldsymbol{n} \mapsto \frac{1}{2^{n+1}} \mid n \in \mathbb{N}\} \in \mathfrak{D}(\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}(\mathtt{NAT}))$ *is the exponential distribution over the natural numbers. Suppose, moreover that for any* n, $\mathcal{N}_{\boldsymbol{n}} = \left\{ \begin{matrix} \boldsymbol{n+1} & \mapsto \frac{1}{2} \\ \mathbf{0} & \mapsto \frac{1}{2} \end{matrix} \right\}$ *gives as value either* $n+1$ *or* 0 *with uniform probability. Then*

$$\begin{aligned} \int_{\mathcal{M}} \mathcal{N}_M.dM &= \sum_n \frac{1}{2^{n+1}} \cdot \left\{ \begin{matrix} \boldsymbol{n+1} & \mapsto \frac{1}{2} \\ \mathbf{0} & \mapsto \frac{1}{2} \end{matrix} \right\} \\ &= \left\{ \begin{matrix} \boldsymbol{n+1} & \mapsto & \frac{1}{2^{n+1}} \cdot \frac{1}{2} & \textit{for } n \in \mathbb{N} \\ \mathbf{0} & \mapsto & \sum_m \frac{1}{2^{m+1}} & \end{matrix} \right\} \\ &= \{\boldsymbol{n} \mapsto \frac{1}{2^{n+1}} \mid n \in \mathbb{N}\} \end{aligned}$$

which is exactly the same as $\mathcal{M}$.

The notation can be easily adapted, e.g., to families of real numbers $(p_M)_{M\in\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}}$ and to other kinds of distributions. We indicate as $C(\!|\mathcal{M}|\!)$ the push-forward distribution $\int_{\mathcal{M}}\{C(\!|M|\!)\}dM$ induced by a context C, and as $\|\mathcal{M}\|$ the norm $\int_{\mathcal{M}} 1dM$ of $\mathcal{M}$. Remark, finally, that we have the useful equality $\mathcal{M} = \int_{\mathcal{M}}\{M\}dM$. The integral can be manipulated as usual integrals which respect the less usual equation:

$$\int_{(\int_{\mathcal{M}}\mathcal{N}_M dM)} \mathcal{L}_N dN \quad = \quad \int_{\mathcal{M}}\left(\int_{\mathcal{N}_M} \mathcal{L}_N dN\right) dM \tag{1}$$

Reduction rules of $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ are given by Figure 6. For simplicity, we use the notation $M \to^? \mathcal{M}$ for $\{M\} \to \mathcal{M}$, *i.e.*, $M \to \mathcal{M}$ whenever M is reducible and $\mathcal{M} = \{M\}$ whenever M is a value. The notation permit to rewrite rule (r-$\in$) as a monadic lifting:

$$\frac{\forall M \in |\mathcal{M}|,\ M \to^? \mathcal{N}_M}{\mathcal{M} \to \int_{\mathcal{M}} \mathcal{N}_M.dM}\ (r\text{-}\in)$$

Example 2. *First, notice that we can simulate any reduction of the usual system $\mathbb{T}$. For example, take the following term:*

$$\mathtt{Expo} \ \coloneqq\ \lambda n.\mathtt{rec}\ \langle \mathbf{2}\ ,\ \lambda xy.\mathtt{rec}\ \langle \mathbf{0}, \lambda x.\mathtt{SS}, y\rangle\ ,\ \mathtt{S}n\rangle \quad : \mathtt{NAT} \to \mathtt{NAT}$$

This term is computing the function $n \mapsto 2^{n+1}$ in time $\sim 2^{n+1}$. Indeed, when applied to a natural number $\boldsymbol{n}$, it will get the following reduction where we denote $E_k \coloneqq \mathtt{rec}\ \langle \mathbf{1}, \lambda xy.\mathtt{rec}\ \langle \mathbf{0}, \lambda x.\mathtt{SS}, y\rangle, \boldsymbol{k}\rangle$:

$$\begin{aligned}
& (\lambda n.\mathtt{rec}\ \langle \mathbf{1}\ ,\ \lambda xy.\mathtt{rec}\ \langle \mathbf{0}, \lambda x.\mathtt{SS}, y\rangle\ ,\ \mathtt{S}n\rangle)\ \boldsymbol{n} \\
\to\ & \Big\{E_{n+1}\Big\} \\
\to\ & \Big\{(\lambda xy.\mathtt{rec}\ \langle \mathbf{0}, \lambda x.\mathtt{SS}, y\rangle)\ \boldsymbol{n}\ E_n\Big\} \\
\to\ & \Big\{(\lambda y.\mathtt{rec}\ \langle \mathbf{0}, \lambda x.\mathtt{SS}, y\rangle)\ E_n\Big\} \\
\to\ & \Big\{\mathtt{rec}\ \langle \mathbf{0}, \lambda x.\mathtt{SS}, E_n\rangle\Big\} \\
& \dots \\
\to\ & \Big\{\mathtt{rec}\ \langle \mathbf{0}, \lambda x.\mathtt{SS}, \mathbf{2}^{\boldsymbol{n}}\rangle\Big\} \\
\to\ & \Big\{(\lambda x.\mathtt{SS})\ (\mathbf{2}^{\boldsymbol{n}}\mathbf{-1})(\mathtt{rec}\ \langle \mathbf{0}, \lambda x.\mathtt{SS}, \mathbf{2}^{\boldsymbol{n}}\mathbf{-1}\rangle)\Big\} \\
\to\ & \Big\{\mathtt{SS}(\mathtt{rec}\ \langle \mathbf{0}, \lambda x.\mathtt{SS}, \mathbf{2}^{\boldsymbol{n}}\mathbf{-1}\rangle)\Big\} \\
& \dots \\
\to\ & \Big\{\mathbf{2}^{\boldsymbol{n+1}}\Big\}
\end{aligned}$$

Remark that we are only considering Diracs here; this is because the reduction is completely determinist: there is no probabilistic choice involved.

Example 3. *As a second example, we are presenting the term $\mathtt{X}\langle\mathtt{S},\mathbf{0}\rangle$ that is essentially reduced by probabilistic derivations:*

$$\mathtt{X}\langle\mathtt{S},\mathbf{0}\rangle \to \begin{Bmatrix} \mathtt{S}\ (\mathtt{X}\ \langle\mathtt{S},\mathbf{0}\rangle) & \mapsto & \frac{1}{2} \\ \mathbf{0} & \mapsto & \frac{1}{2} \end{Bmatrix} \to \begin{Bmatrix} \mathtt{SS}\ (\mathtt{X}\ \langle\mathtt{S},\mathbf{0}\rangle) & \mapsto & \frac{1}{4} \\ \mathbf{1} & \mapsto & \frac{1}{4} \\ \mathbf{0} & \mapsto & \frac{1}{2} \end{Bmatrix} \to \begin{Bmatrix} \mathtt{SSS}\ (\mathtt{X}\ \langle\mathtt{S},\mathbf{0}\rangle) & \mapsto & \frac{1}{8} \\ \mathbf{2} & \mapsto & \frac{1}{8} \\ \mathbf{1} & \mapsto & \frac{1}{4} \\ \mathbf{0} & \mapsto & \frac{1}{2} \end{Bmatrix} \to \cdots$$

$$M \to \cdots \to \boldsymbol{n}$$

Figure 1: A Reduction in $\mathbb{T}$

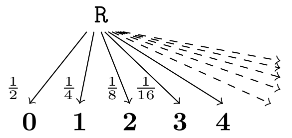


Figure 2:A Reduction in $\mathbb{T}^{\mathtt{R}}$

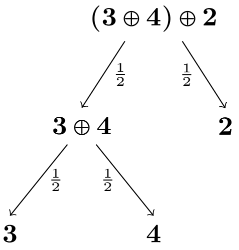


Figure 3:A Reduction in $\mathbb{T}^{\oplus}$

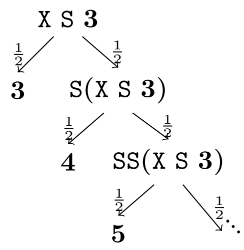


Figure 4:A Reduction in $\mathbb{T}^{\mathtt{X}}$

$$\frac{}{\Gamma, x : A \vdash x : A} \qquad \frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x.M : A \to B} \qquad \frac{\Gamma \vdash M : A \to B \qquad \Gamma \vdash N : A}{\Gamma \vdash M\ N : B}$$

$$\frac{}{\Gamma \vdash \mathbf{0} : \mathtt{NAT}} \qquad \frac{}{\Gamma \vdash \mathtt{S} : \mathtt{NAT} \to \mathtt{NAT}} \qquad \frac{}{\Gamma \vdash \mathtt{rec} : (A \times (\mathtt{NAT} \to A \to A) \times \mathtt{NAT}) \to A}$$

$$\frac{\Gamma \vdash M : A \qquad \Gamma \vdash N : B}{\Gamma \vdash \langle M, N \rangle : A \times B} \qquad \frac{}{\Gamma \vdash \pi_1 : (A \times B) \to A} \qquad \frac{}{\Gamma \vdash \pi_2 : (A \times B) \to B}$$

$$\frac{\Gamma \vdash M : A \qquad \Gamma \vdash N : A}{\Gamma \vdash M \oplus N : A} \qquad \frac{}{\Gamma \vdash \mathtt{R} : \mathtt{NAT}} \qquad \frac{}{\Gamma \vdash \mathtt{X} : (A \to A) \times A \to A}$$

Figure 5: Typing Rules.

$$\frac{}{(\lambda x.M)\ V \to \left\{M[V/x]\right\}}\ (r\text{-}\beta) \qquad \frac{M \to \mathcal{M}}{M\ V \to \mathcal{M}\ V}\ (r\text{-}@L)$$

$$\frac{N \to \mathcal{N}}{M\ N \to M\ \mathcal{N}}\ (r\text{-}@R) \qquad \frac{M \to \mathcal{M}}{\langle M, N \rangle \to \langle \mathcal{M}, N \rangle}\ (r\text{-}\langle\cdot\rangle L) \qquad \frac{M \to \mathcal{M}}{\langle V, M \rangle \to \langle V, \mathcal{M} \rangle}\ (r\text{-}\langle\cdot\rangle R)$$

$$\frac{}{\mathtt{rec}\langle U, V, \mathbf{0} \rangle \to \left\{U\right\}}\ (r\text{-}\mathtt{rec}\mathbf{0}) \qquad \frac{}{\mathtt{rec}\langle U, V, \mathtt{S}\ \boldsymbol{n} \rangle \to \left\{V\ \boldsymbol{n}\ (\mathtt{rec}\langle U, V, \boldsymbol{n} \rangle)\right\}}\ (r\text{-}\mathtt{recS})$$

$$\frac{}{\pi_1\ \langle V, U \rangle \to \left\{V\right\}}\ (r\text{-}\pi_1) \qquad \frac{}{\pi_2\ \langle V, U \rangle \to \left\{U\right\}}\ (r\text{-}\pi_2) \qquad \frac{}{\mathtt{R} \to \left\{\boldsymbol{n} \mapsto \frac{1}{2^{n+1}}\right\}_{n \in \mathbb{N}}}\ (r\text{-}\mathtt{R})$$

$$\frac{}{M \oplus N \to \left\{\begin{matrix} M & \mapsto & \frac{1}{2} \\ N & \mapsto & \frac{1}{2} \end{matrix}\right\}}\ (r\text{-}\oplus) \qquad \frac{}{\mathtt{X}\langle V, W \rangle \to \left\{\begin{matrix} V\ (\mathtt{X}\ \langle V, W \rangle) & \mapsto & \frac{1}{2} \\ W & \mapsto & \frac{1}{2} \end{matrix}\right\}}\ (r\text{-}\mathtt{X})$$

$$\frac{\forall M \in |\mathcal{M}|_R,\ M \to \mathcal{N}_M \qquad \forall V \in |\mathcal{M}|_V,\ \mathcal{N}_V = \{V\}}{\mathcal{M} \to \int_{\mathcal{M}} \mathcal{N}_M . dM}\ (r\text{-}\in)$$

Figure 6: Operational Semantics.

Notice that the reduction $\to$ (resp. $\to^?$) is deterministic. We can easily define $\to^n$ (resp. $\to^{\leq n}$) as the n^{th} exponentiation of $\to$ (resp. $\to^?$) and $\to^*$ as the reflexive and transitive closure of $\to$ (and $\to^?$). In probabilistic systems, we might want to consider infinite reductions such as the ones induced by $\mathtt{X}\ (\lambda x.x)\ \mathbf{0}$ which reduces to $\{\mathbf{0}\}$, but in an infinite number of steps. Remark that for any value V, and whenever $\mathcal{M} \to \mathcal{N}$, it holds that $\mathcal{M}(V) \leq \mathcal{N}(V)$. As a consequence, we can proceed as follows:

Definition 4. *Let M be a term and let $(\mathcal{M}_n)_{n\in\mathbb{N}}$ be the unique distribution family such that $M \to^{\leq n} \mathcal{M}_n$. The* evaluation *of M is the value distribution*

$$[\![M]\!] := \{V \mapsto \lim_{n\to\infty} \mathcal{M}_n(V)\} \in \mathfrak{D}(\mathbb{T}_V^{\oplus,\mathtt{R},\mathtt{X}}).$$

The success *of $\mathcal{M}$ is its probability of normalization, which is formally defined as the norm of its evaluation, i.e., $Succ(M) := \|\,[\![M]\!]\,\|$. $\mathcal{M}_n^{\Delta V}$ stands for $\{V \mapsto \mathcal{M}_n(V) - \mathcal{M}_{n-1}(V)\}$, the distributions of values reachable in exactly n steps. The* average reduction length *from M is then*

$$[M] \ := \ \sum_n \left(n \cdot \|\mathcal{M}_n^{\Delta V}\|\right) \quad \in \ \mathbb{N} \cup \{+\infty\}$$

Example 5. *Take as example the term $\mathtt{X}\langle\mathtt{S},\mathbf{0}\rangle$ of example 3. We have, for all n:*

$$\mathtt{X}\langle\mathtt{S},\mathbf{0}\rangle \to^n \mathcal{M}_n = \left\{\begin{matrix} \mathtt{S}^n\ (\mathtt{X}\ \langle\mathtt{S},\mathbf{0}\rangle) & \mapsto & \frac{1}{2^n} & \\ \boldsymbol{i} & \mapsto & \frac{1}{2^{i+1}} & \text{for } i < n \end{matrix}\right\},$$

so that $\mathcal{M}_n(\boldsymbol{i}) = \mathbf{0}$ if $i \geq n$ and $\mathcal{M}_n(\boldsymbol{i}) = \frac{1}{2^{i+1}}$ otherwise. Thus:

$$\begin{aligned} [\![\mathtt{X}\langle\mathtt{S},\mathbf{0}\rangle]\!] &= \left\{\boldsymbol{n} \mapsto \lim_{n\to\infty} \mathcal{M}_n(\boldsymbol{n}) \;\middle|\; n \in \mathbb{N}\right\} \\ &= \left\{\boldsymbol{n} \mapsto \frac{1}{2^{n+1}} \;\middle|\; n \in \mathbb{N}\right\} \end{aligned}$$

Notice that, by Rule (r- $\in$), the evaluation is continuous:

$$[\![\mathcal{M}]\!] = \int_{\mathcal{M}} [\![M]\!]\, dM.$$

Any term M of type $\mathtt{NAT} \to \mathtt{NAT}$ *represents* a function $g : \mathbb{N} \to \mathfrak{D}(\mathbb{N})$ iff for every m, n it holds that $g(n)(m) = [\![M\ \boldsymbol{n}]\!]\,(\boldsymbol{m})$.

3.2 Technical Properties Of The Operational Semantics

Before giving the main results, some auxiliary lemmas are necessary. In particular, we are looking for Lemma 10 which is a fundamental continuity lemma stating that it is equivalent to apply a call by value strategy at the level of distributions, provided that we allow infinite reductions.

The first lemma is stating that the (one step) reduction of a sum (or an integral) is the sum (or the integral) of all the addends. Of course, the different reductions of the addends can have interpolations which makes the decomposition nontrivial:

Lemma 6. *If it exist, the reduction $\mathcal{L}$ of any integral $\int_{\mathcal{M}} \mathcal{N}_M dM \to \mathcal{L}$, is the integral $\int_{\mathcal{M}} \mathcal{L}_M dM$ such that $\mathcal{N}_M \to \mathcal{L}_M$ for any $M \in |\mathcal{M}|$.*

Proof.
- Let $(\mathcal{L}_M)_{M\in|\mathcal{M}|}$ such that $\mathcal{N}_M \to \mathcal{L}_M$ for any $M \in |\mathcal{M}|$. Then in order to derive $\mathcal{N}_M \to \mathcal{L}_M$, the only rule we can apply is (r-$\in$) so that there is $(\mathcal{L}'_{M,N})_{M\in|\mathcal{M}|, N\in|\mathcal{N}_M|}$ such that $\mathcal{L}_M = \int_{\mathcal{N}_M} \mathcal{L}'_{M,N} dN$ and $N \to^? \mathcal{L}'_{M,N}$. Notice that for $N \in |\mathcal{N}_M| \cap |\mathcal{N}_{M'}|$, the determinism of $\to^?$ gives that $\mathcal{L}'_{M,N} = \mathcal{L}'_{M',N}$, thus we can define $\mathcal{L}'_N$ without ambiguity for any $N \in \bigcup_{M\in|\mathcal{M}|} |\mathcal{N}_M| = \left|\int_{\mathcal{M}} \mathcal{N}_M dM\right|$. Then we have $N \to^? \mathcal{L}'_N$ and $\int_{\int_{\mathcal{M}} \mathcal{N}_M dM} \mathcal{L}'_N dN = \int_{\mathcal{M}} \left(\int_{\mathcal{N}_M} \mathcal{L}'_N dN\right) dM = \int_{\mathcal{M}} \mathcal{L}_M dM$ (we use Eq (1)). Thus, by applying rule (r-$\in$) we get $\int_{\mathcal{M}} \mathcal{N}_M dM \to \int_{\mathcal{M}} \mathcal{L}_M dM$.

- Conversely, we assume that $\int_{\mathcal{M}} \mathcal{N}_M dM \to \mathcal{L}$. In order to derive it, the only rule we can apply is $(r\text{-}\epsilon)$ so that there is $(\mathcal{L}'_N)_{N\in|\int_{\mathcal{M}}\mathcal{N}_M dM|}$ such that $\mathcal{L} = \int_{(\int_{\mathcal{M}}\mathcal{N}_M dM)} \mathcal{L}'_N dN = \int_{\mathcal{M}} \left(\int_{\mathcal{N}_M} \mathcal{L}'_N dN\right) dM$ and $N \to^? \mathcal{L}'_N$. We conclude by setting $\mathcal{L}_M := \int_{\mathcal{N}_M} \mathcal{L}'_N dN$.

□

This decomposition is off course iterable to any lengths of reduction:

Lemma 7. *The n^{th} reductions $\mathcal{L}$ of any integral $\int_{\mathcal{M}} \mathcal{N}_M dM \to^n \mathcal{L}$, is (if it exists) the integral $\int_{\mathcal{M}} \mathcal{L}_M dM$ such that $\mathcal{N}_M \to^n \mathcal{L}_M$ for any $M \in |\mathcal{M}|$.*

Proof. By induction on n:
- If $n = 0$ then $\mathcal{L}_M = \mathcal{N}$.
- Otherwise, $\int_{\mathcal{M}} \mathcal{N}_M dM \to \mathcal{L}' \to^{n-1} \mathcal{L}$. By Lemma 6, the first step is possible iff $\mathcal{L}' = \int_{\mathcal{M}} \mathcal{L}'_M dM$ with $\mathcal{N}_M \to \mathcal{L}'_M$ for any $M \in |\mathcal{M}|$. By IH, the remaining steps are then possible iff $\mathcal{L} = \int_{\mathcal{M}} \mathcal{L}_M dM$ such that $\mathcal{L}'_M \to^{n-1} \mathcal{L}_M$ for any $M \in |\mathcal{M}|$.

□

Conversely, it is possible to track back values obtained from an application as an application of values obtained from reducing both applicants.

Lemma 8. *If $(M\ N) \to^n \mathcal{L} \geq \mathcal{W}$ for some $\mathcal{W} \in \mathfrak{D}(\mathbb{T}_V^{\oplus,\mathtt{R},\mathtt{X}})$, then there is $\mathcal{U}, \mathcal{V} \in \mathfrak{D}(\mathbb{T}_V^{\oplus,\mathtt{R},\mathtt{X}})$ such that*

$$M \to^* \mathcal{M} \geq \mathcal{U}, \qquad N \to^* \mathcal{N} \geq \mathcal{V}, \qquad (\mathcal{U}\ \mathcal{V}) \to^* \mathcal{L}' \geq \mathcal{W}.$$

Proof. By induction on n.
Trivial if $n = 0$.
If $n \geq 1$, we proceed differently depending whether M and N are values or not.
- If N is not a value then $N \to \mathcal{N}'$ and $(M\ N) \to (M\ \mathcal{N}') \to^{n-1} \mathcal{L} \geq \mathcal{W}$.
  Notice that $(M\ \mathcal{N}') := \int_{\mathcal{N}'} \{M\ N'\} dN'$. Using Lemma 7, we can decompose $\mathcal{W} = \int_{\mathcal{N}'} \mathcal{W}_{N'} dN'$ and $\mathcal{L} = \int_{\mathcal{N}'} \mathcal{L}_{N'} dN'$ in such a way that for any $N' \in |\mathcal{N}'|$, $(M\ N') \to^{\leq n-1} \mathcal{L}_{N'} \geq \mathcal{W}_{N'}$. By induction we have $N' \to^* \mathcal{N}_{N'} \geq \mathcal{V}_{N'}$ and $M \to^* \mathcal{M}_{N'} \geq \mathcal{U}_{N'}$ with $(\mathcal{U}_{N'}\ \mathcal{V}_{N'}) \to^* \mathcal{L}'_{N'} \geq \mathcal{W}_{N'}$ for all $N' \in |\mathcal{N}'|$. Summing up we have $\mathcal{M} := \int_{\mathcal{N}'} \mathcal{M}_{N'} dN'$, $\mathcal{U} := \int_{\mathcal{N}'} \mathcal{U}_{N'} dN'$, $\mathcal{N} := \int_{\mathcal{N}'} \mathcal{N}_{N'} dN'$ and $\mathcal{V} := \int_{\mathcal{N}'} \mathcal{V}_{N'} dN'$.
- If N is a value and $M \to \mathcal{M}'$ then $(M\ N) \to (\mathcal{M}'\ N) \to^{n-1} \mathcal{L} \geq \mathcal{W}$. Thus we can decompose the equation similarly along $\mathcal{M}'$ and apply our IH.
- If both M and N are values it is trivial since $\mathcal{U} = \{M\}$ and $\mathcal{V} = \{N\}$.

□

Lemma 9. *For every $m, n \in \mathbb{N}$ and every $\mathcal{M}, \mathcal{N} \in \mathfrak{D}(\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}})$, whenever $M \to^m \mathcal{M}$ and $N \to^n \mathcal{N} \geq \mathcal{V}$, we have $[\![M\ N]\!] \geq [\![\mathcal{M}\ \mathcal{V}]\!]$. In particular, if moreover $\mathcal{M} \geq \mathcal{U}$, then $[\![M\ N]\!] \geq [\![\mathcal{U}\ \mathcal{V}]\!]$.*

Proof. By induction on $m + n$.
Trivial if $m + n = 0$.
If $n \geq 1$, we proceed differently depending whether M and N are values or not.
- If $N \to \mathcal{L} \to^{n-1} \mathcal{N} \geq \mathcal{V}$ then N is not a value.
  Using Lemma 7, we can decompose $\mathcal{N} = \int_{\mathcal{L}} \mathcal{N}_L dL$ and $\mathcal{V} = \int_{\mathcal{L}} \mathcal{V}_L dL$ in such a way that for any $L \in |\mathcal{L}|$, $L \to^{\leq n-1} \mathcal{N}_L \geq \mathcal{V}_L$ (with $\to^{\leq n-1}$ dubbing either $=$ or $\to^{n-1}$ depending whether L is a value or not).
  Then we get
$$[\![M\ N]\!] = [\![M\ \mathcal{L}]\!] = \int_{\mathcal{L}} [\![M\ L]\!]\, dL \geq \int_{\mathcal{L}} [\![\mathcal{M}\ \mathcal{V}_L]\!]\, dL = \left[\!\!\left[\mathcal{M}\ \left(\int_{\mathcal{L}} \mathcal{V}_L dL\right)\right]\!\!\right] = [\![\mathcal{M}\ \mathcal{V}]\!]\ .$$
- If $\mathcal{V}$ is the empty (sub)distribution, this is trivial.
- If $n = 0$ and $\mathcal{V}$ is not empty, then $\mathcal{V} = \{N\}$, thus N is a value and $M \to \mathcal{L} \to^{m-1} \mathcal{M}$, then we can decompose the equation similarly along $\mathcal{L}$ and apply our IH.

□

The following is a crucial intermediate step towards Theorem 12, the main result of this section.

Lemma 10. *For any M, N: $[\![M\ N]\!] = [\![[\![M]\!]\ [\![N]\!]]\!]$. In particular, if the application $M\ N$ is almost-surely terminating, so are M and N.*

Proof. ($\leq$) There is $(\mathcal{L}_n, \mathcal{W}_n)_{n\geq 1}$ such that $(M\ N) \to^n \mathcal{L}_n \geq \mathcal{W}_n$ for all n and such that $[\![M\ N]\!] = \lim_n(\mathcal{W}_n)$. Applying Lemma 8 gives $\mathcal{M}_n$, $\mathcal{N}_n$, $\mathcal{L}'_n \in \mathfrak{D}(\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}})$ and $\mathcal{U}_n, \mathcal{V}_n \in \mathfrak{D}(\mathbb{T}_V^{\oplus,\mathtt{R},\mathtt{X}})$ such that $M \to^* \mathcal{M}_n \geq \mathcal{U}_n$, $N \to^* \mathcal{N}_n \geq \mathcal{V}_n$ and $(\mathcal{U}_n\ \mathcal{V}_n) \to^* \mathcal{L}'_n \geq \mathcal{W}_n$. Thus $\lim_n \mathcal{U}_n \leq [\![M]\!]$ and $\lim_n \mathcal{V}_n \leq [\![N]\!]$ with $\mathcal{W}_n \leq [\![\mathcal{U}_n\ \mathcal{V}_n]\!]$ leading to the required inequality:

$$[\![M\ N]\!] = \lim_n \mathcal{W}_n \leq \lim_n [\![\mathcal{U}_n\ \mathcal{V}_n]\!] \leq \left[\!\!\left[\lim_n (\mathcal{U}_n\ \mathcal{V}_n)\right]\!\!\right] \leq \left[\!\!\left[\left(\lim_n \mathcal{U}_n\right)\ \left(\lim_n \mathcal{V}_n\right)\right]\!\!\right] \leq [\![[\![M]\!]\ [\![N]\!]]\!]\ .$$

($\geq$) There is $(\mathcal{M}_n, \mathcal{N}_n, \mathcal{U}_n, \mathcal{V}_n)_{n\geq 1}$ such that $M \to^n \mathcal{M}_n \geq \mathcal{U}_n$ and $N \to^n \mathcal{N}_n \geq \mathcal{V}_n$ for all n and such that $[\![M]\!] = \lim_n(\mathcal{U}_n)$ and $[\![N]\!] = \lim_n(\mathcal{V}_n)$. This leads to the equality $[\![[\![M]\!]\ [\![N]\!]]\!] = \lim_{m,n} [\![\mathcal{U}_m\ \mathcal{V}_n]\!]$. Finally, by Lemma 9, for any m, n, each approximant of $[\![\mathcal{U}_m\ \mathcal{V}_n]\!]$ is below $[\![M\ N]\!]$, so is their sup. □

3.3 Almost-Sure Termination

We now have all the necessary ingredients to specify a quite powerful notion of probabilistic computation. When, precisely, should such a process be considered *terminating*? Do all probabilistic branches (see figures 1-4) need to be finite? Or should we stay more liberal? The literature on the subject is unanimously pointing to the notion of *almost-sure termination*: a probabilistic computation should be considered terminating if the set of infinite computation branches, although not necessarily empty, has null probability [17, 9, 14]. This has the following incarnation in our setting:

Definition 11. *A term M is said to be* almost-surely terminating *(AST) iff $Succ(M) = 1$.*

This section is concerned with proving that $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ indeed guarantees almost-sure termination. This will be done by adapting Girard-Tait's reducibility technique.

Theorem 12. *The full system $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ is almost-surely terminating (AST),* i.e.,

$$\forall M \in \mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}, \quad Succ(M) = 1.$$

Proof. The proof is is based on the the notion of a *reducible* term, which is given as follows by induction on the structure of types:

$$\begin{aligned} Red_{\mathtt{NAT}} &:= \left\{M \in \mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}(\mathtt{NAT}) \mid M \text{ is AST}\right\} \\ Red_{A\to B} &:= \{M \mid \forall V \in Red_A \cap \mathbb{T}_V^{\oplus,\mathtt{R},\mathtt{X}},\ (M\ V) \in Red_B\} \\ Red_{A\times B} &:= \{M \mid (\pi_1\ M) \in Red_A,\ (\pi_2\ M) \in Red_B\} \end{aligned}$$

Then we can observe that:

1. *The reducibility candidates over Red_A are $\to$-saturated:* by induction on A we can indeed show that if $M \to \mathcal{M}$ then $|\mathcal{M}| \subseteq Red_A$ iff $M \in Red_A$.
   - Trivial for $A = \mathtt{NAT}$.
   - If $A = B \to C$: then for all value $V \in Red_B$, $(M\ V) \to (\mathcal{M}\ V)$, thus by IH $(M\ V) \in Red_C$ iff $|\mathcal{M}\ V| = \{M'\ V \mid M' \in |\mathcal{M}|\} \subseteq Red_C$; which exactly means that $|\mathcal{M}| \subseteq Red_{B\to C}$ iff $M \in Red_{B\to C}$.
   - If $A = A_1 \times A_2$: then for $i \in \{1, 2\}$, $(\pi_i\ M) \to (\pi_i\ \mathcal{M})$ so that by IH, $(\pi_i\ M) \in Red_{A_i}$ iff $(\pi_i\ \mathcal{M}) \subseteq Red_{A_i}$; which exactly means that $|\mathcal{M}| \subseteq Red_{A_1\times A_2}$ iff $M \in Red_{A_1\times A_2}$.
2. *The reducibility candidates over Red_A are precisely the AST terms M such that $[\![M]\!] \subseteq Red_A$:* this goes by induction on A.

- Trivial for $A = \mathtt{NAT}$.
- For $A = B \to C$:

  Let $M \in Red_{B\to C}$. Remark that there is a value $V \in Red_B$, thus $(M\ V) \in Red_C$ and $(M\ V)$ is AST by IH; using Lemma 10 we get M AST and it is easy to see that if $U \in |[\![M]\!]|$ then $U \in |\mathcal{M}|$ for some $M \to^* \mathcal{M}$ so that $U \in Red_{B\to C}$ by saturation.

  Conversely, let M be AST with $|[\![M]\!]| \subseteq Red_{B\to C}$ and let $V \in Red_B$ be a value. By IH, for any $U \in |[\![M]\!]| \subseteq Red_{B\to C}$ we have $(U\ V)$ AST with an evaluation supported by elements of Red_C; by Lemma 10 $[\![M\ V]\!] = [\![[\![M]\!]\ V]\!]$ meaning that $(M\ V)$ is AST and has an evaluation supported by elements of Red_C, so that we can conclude by IH.
- For $A = A_1 \times A_2$:

  Let $M \in Red_{A_1\times A_2}$. then $(\pi_1\ M) \in Red_{A_1}$ and $(\pi_1\ M)$ is AST by IH; using Lemma 10 we get M AST and it is easy to see that if $U \in |[\![M]\!]|$ then $U \in |\mathcal{M}|$ for some $M \to^* \mathcal{M}$ so that $U \in Red_{A_1\to A_2}$ by saturation.

  Conversely, let M be AST with $|[\![M]\!]| \subseteq Red_{A_1\to A_2}$ and let $i \in \{1,\}2$. By IH, for any $U \in |[\![M]\!]| \subseteq Red_{A_1\to A_2}$ we have $(\pi_i U)$ AST with an evaluation supported by elements of Red_{A_i}; by Lemma 10 $[\![\pi_i\ M]\!] = [\![\pi_i\ [\![M]\!]]\!]$ meaning that $(\pi_i\ M)$ is AST and has an evaluation supported by elements of Red_{A_i}, so that we can conclude by IH.

3. *Every term M such that $x_1 : A_1, \dots, x_n : A_n \vdash M : B$ is a candidate in the sense that if $V_i \in Red_{A_i}$ for every $1 \le i \le n$, then $M[V_1/x_1, \dots, V_n/x_n] \in Red_B$:*
by induction on the type derivation. The only difficult cases are the recursion, the application, the binary choice $\oplus$ and the denumerable choice $\mathtt{X}$:

   - For the operator $\mathtt{rec}$: We have to show that if $U \in Red_A$ and $V \in Red_{\mathtt{NAT}\to A\to A}$ then for all $n \in \mathbb{N}$, $(\mathtt{rec}\ \langle U, V, \boldsymbol{n}\rangle) \in Red_A$. We proceed by induction on n:
     - If $n = 0$: $\mathtt{rec}\ \langle U, V, \mathbf{0}\rangle \to \{U\} \subseteq Red_A$ and we conclude by saturation.
     - Otherwise: $\mathtt{rec}\ \langle U, V, (\boldsymbol{n}+\mathbf{1})\rangle \to V\ \boldsymbol{n}\ (\mathtt{rec}\ \langle U, V, \boldsymbol{n}\rangle) \in Red_A$ since $(\mathtt{rec}\ \langle U, V, \boldsymbol{n}\rangle) \in Red_A$ by IH and since $\boldsymbol{n} \in Red_{\mathbb{N}}$ and $V \in Red_{\mathbb{N}\to A\to A}$, we conclude by saturation.
   - For the application: we have to show that if $M \in Red_{A\to B}$ and $N \in Red_A$ then $(M\ N) \in Red_B$. But since $N \in Red_A$, this means that it is AST and for every $V \in |[\![N]\!]|$, $(M\ V) \in Red_B$. In particular, by Lemma 10, we have $[\![M\ N]\!] = [\![M\ [\![N]\!]]\!]$ so that $(M\ N)$ is AST and $|[\![M\ N]\!]| \subseteq \bigcup_{V\in|[\![N]\!]|} |[\![M\ V]\!]| \subseteq Red_B$.
   - For the operator $\oplus$: If $M, N \in Red_A$ then $\left|\begin{Bmatrix} M \mapsto \frac{1}{2} \\ N \mapsto \frac{1}{2} \end{Bmatrix}\right| \subseteq Red_A$, and, by $\to$-saturation, $(M \oplus N) \in Red_A$.
   - For the operator $\mathtt{X}$: we have to show that for any value $U{\in}Red_{A\to A}$ and $V{\in}Red_A$ if holds that $(\mathtt{X}\ U\ V) \in Red_A$.
   By an easy induction on n, $(U^n\ V) \in Red_A$ since $U^0\ V = V \in Red_B$ and $U^{n+1}\ V = U\ (U^n\ V) \in Red_B$ whenever $U^n\ V \in Red_B$ and $U \in Red_{B\to B}$.
   Moreover, by an easy induction on n we have

   $$[\![\mathtt{X}\ U\ V]\!] = \frac{1}{2^{n+1}} [\![U^n\ (\mathtt{X}\ U\ V)]\!] + \sum_{i\le n} \frac{1}{2^{i+1}} [\![U^i\ V]\!] \ .$$

     - trivial for $n = 0$,
     - we know that $[\![\mathtt{X}\ U\ V]\!] = \frac{1}{2}[\![V]\!] + \frac{1}{2}[\![V(\mathtt{X}\ U\ V)]\!]$ and we get $[\![V(\mathtt{X}\ U\ V)]\!] = [\![V\ [\![\mathtt{X}\ U\ V]\!]]\!] = \frac{1}{2^{n+1}} \left[\!\left[U^{n+1}(\mathtt{X}\ U\ V)\right]\!\right] + \sum_{i\le n} \frac{1}{2^{i+1}} \left[\!\left[U^{i+1}V\right]\!\right]$ by Lemma 10 and IH, which is sufficient to conclude.

   At the limit, we get $[\![\mathtt{X}\ U\ V]\!] = \sum_{i\in\mathbb{N}} \frac{1}{2^{i+1}} \left[\!\left[U^i\ V\right]\!\right]$. We can then conclude that $(\mathtt{X}\ U\ V)$ is AST (since each of the $(U^i\ V) \in Red_B$ are AST and $\sum_i \frac{1}{2^{i+1}} = 1$) and that $|[\![M\ N]\!]| = \bigcup_i \left|\left[\!\left[U^i\ V\right]\!\right]\right| \subseteq Red_A$.

□

Almost-sure termination could however be seen as too weak a property: there is no guarantee about the average computation length. For this reason, another stronger notion is often considered, namely *positive* almost-sure termination:

Definition 13. *A term M is said to be* positively almost-surely terminating *(or* PAST*) iff the average reduction length $[M]$ is finite.*

Gödel's $\mathbb{T}$, when paired with $\mathtt{R}$, is combinatorially too powerful to guarantee positive almost sure termination. More precisely, the issue is triggered by the ability to describe programs with exponential reduction time such as the term **Expo** from Example 2 which is computing the function $n \mapsto 2^{n+1}$ in time $\sim 2^{n+1}$.

Theorem 14. *$\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ is not positively almost-surely terminating.*

Proof. The term $(\mathtt{Expo}\ \mathtt{R}) : \mathtt{NAT}$ is computing, with probability $\frac{1}{2^{n+1}}$ the number 2^{n+1} in time 2^{n+1}; the average reduction length is thus

$$[\mathtt{Expo}\ \mathtt{R}] \;=\; \sum_n \frac{2^{n+1}}{2^{n+1}} \;=\; +\infty \;.$$

□

3.4 On Fragments of $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$: a Roadmap

The calculus $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ contains at least four fragments, namely Gödel's $\mathbb{T}$ and the three fragments $\mathbb{T}^{\oplus}$, $\mathbb{T}^{\mathtt{R}}$ and $\mathbb{T}^{\mathtt{X}}$ corresponding to the three probabilistic choice operators we consider. It is then natural to ask oneselves how these fragments relate to each other as for their respective expressive power. At the end of this paper, we will have a very clear picture in front of us.

The first such result is the equivalence between the apparently dual fragments $\mathbb{T}^{\mathtt{R}}$ and $\mathbb{T}^{\mathtt{X}}$. The embeddings are in fact quite simple: getting $\mathtt{X}$ from $\mathtt{R}$ only requires "guessing" the number of iterations via $\mathtt{R}$ and then use `rec` to execute them; conversely, capturing $\mathtt{R}$ from $\mathtt{X}$ is even easier: it corresponds to counting the recursive loops done by some executions of $\mathtt{X}$:

Proposition 15. *$\mathbb{T}^{\mathtt{R}}$ and $\mathbb{T}^{\mathtt{X}}$ are both equiexpressive with $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$.*

Proof. The calculus $\mathbb{T}^{\mathtt{R}}$ embeds the full system $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ via the encoding:[1]

$$M \oplus_{\mathtt{R}} N := \mathtt{rec}\ \langle \lambda z.N, \lambda xyz.M, \mathtt{R}\rangle\ \mathbf{0}; \qquad\qquad \mathtt{X}_{\mathtt{R}} \;:=\; \lambda x.\mathtt{rec}\ \langle \pi_2 x, \lambda z.\pi_1 x, \mathtt{R}\rangle.$$

The fragment $\mathbb{T}^{\mathtt{X}}$ embeds the full system $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ via the encoding:

$$M \oplus_{\mathtt{X}} N := \mathtt{X}\ \langle \lambda xy.M, \lambda y.N\rangle\ \mathbf{0}; \qquad\qquad \mathtt{R}_{\mathtt{X}} \;:=\; \mathtt{X}\ \langle \mathtt{S}, \mathbf{0}\rangle.$$

In both cases, the embedding is compositional and preserves types.

We have to prove the correctedness of the two embeddings:

- For any M and N:

$$[\![M \oplus_{\mathtt{R}} N]\!] \;=\; [\![M \oplus_{\mathtt{X}} N]\!] \;=\; [\![M \oplus N]\!] \;=\; \frac{1}{2}\,[\![M]\!] + \frac{1}{2}\,[\![N]\!] \;.$$

[1]Notice that the dummy abstractions on z and the $\mathbf{0}$ at the end ensure the correct reduction order by making $\lambda z.N$ a value.

Indeed, we only have to perform a few reductions:

$$\begin{aligned}
[\![M \oplus_{\mathtt{R}} N]\!] &= \sum_{n\geq 0} \frac{1}{2^{n+1}} [\![\mathtt{rec}\ \langle \lambda z.N, \lambda xyz.M, \boldsymbol{n}\rangle\ \mathbf{0}]\!] \\
&= \frac{1}{2} [\![\mathtt{rec}\ \langle \lambda z.N, \lambda xyz.M, \mathbf{0}\rangle\ \mathbf{0}]\!] + \sum_{n\geq 0} \frac{1}{2^{n+2}} [\![\mathtt{rec}\ \langle \lambda z.N, \lambda xyz.M, \mathtt{S}\boldsymbol{n}\rangle\ \mathbf{0}]\!] \\
&= \frac{1}{2} [\![(\lambda z.N)\ \mathbf{0}]\!] + \sum_{n\geq 0} \frac{1}{2^{n+2}} [\![(\lambda xyz.M)\ \boldsymbol{n}\ (\mathtt{rec}\ \langle \lambda z.N, \lambda xyz.M, \boldsymbol{n}\rangle)\ \mathbf{0}]\!] \\
&= \frac{1}{2} [\![N]\!] + \sum_{n\geq 0} \frac{1}{2^{n+2}} [\![M]\!] \\
&= \frac{1}{2} [\![N]\!] + \frac{1}{2} [\![M]\!] \\
[\![M \oplus_{\mathtt{X}} N]\!] &= \frac{1}{2} [\![(\lambda y.N)\ \mathbf{0}]\!] + \frac{1}{2} [\![(\lambda xy.M)\ \mathtt{R}_{\mathtt{X}}\ \mathbf{0}]\!] \\
&= \frac{1}{2} [\![(\lambda y.N)\ \mathbf{0}]\!] + \frac{1}{2} [\![(\lambda xy.M)\ \mathtt{R}_{\mathtt{X}}\ \mathbf{0}]\!] \\
&= \frac{1}{2} [\![N]\!] + \frac{1}{2} [\![M]\!]
\end{aligned}$$

- For any U and V:

$$[\![\mathtt{X}_{\mathtt{R}}\langle U, V\rangle]\!] \ = \ [\![\mathtt{X}\langle U, V\rangle]\!]$$

  Indeed, both of them are the unique fixedpoint of the following contractive function:

$$f(\mathcal{X}) \ \coloneqq \ \frac{1}{2} [\![U]\!] + \frac{1}{2} [\![V\ \mathcal{X}]\!] \ .$$

  That $[\![\mathtt{X}\langle U, V\rangle]\!] = f([\![\mathtt{X}\langle U, V\rangle]\!])$ is immediate after a reduction, as for the other, we have:

$$\begin{aligned}
[\![\mathtt{X}_{\mathtt{R}}\langle U, V\rangle]\!] &= \sum_{n\geq 0} \frac{1}{2^{n+1}} [\![\mathtt{rec}\ \langle U, \lambda z.V, \boldsymbol{n}\rangle]\!] \\
&= \frac{1}{2} [\![U]\!] + \frac{1}{2} \sum_{n\geq 0} \frac{1}{2^{n+1}} [\![U\ (\mathtt{rec}\ \langle U, \lambda z.V, \boldsymbol{n}\rangle)]\!] \\
&= \frac{1}{2} [\![U]\!] + \frac{1}{2} \sum_{n\geq 0} \frac{1}{2^{n+1}} [\![U\ \ [\![\mathtt{rec}\ \langle U, \lambda z.V, \boldsymbol{n}\rangle]\!]]\!] && \text{by Lemma 10} \\
&= \frac{1}{2} [\![U]\!] + \frac{1}{2} \left[\!\!\left[U\ \left(\sum_{n\geq 0} \frac{1}{2^{n+1}} [\![\mathtt{rec}\ \langle U, \lambda z.V, \boldsymbol{n}\rangle]\!]\right)\right]\!\!\right] \\
&= \frac{1}{2} [\![U]\!] + \frac{1}{2} [\![U\ \ [\![\mathtt{X}_{\mathtt{R}}\langle U, V\rangle]\!]]\!]
\end{aligned}$$

- Finally:

$$[\![\mathtt{R}_{\mathtt{X}}]\!] \ = \ [\![\mathtt{R}]\!] \ = \ \left\{\boldsymbol{n} \mapsto \frac{1}{2^{n+1}} \middle| n \geq 0\right\} \ .$$

  That $[\![\mathtt{R}]\!] = \left\{\boldsymbol{n} \mapsto \frac{1}{2^{n+1}} \middle| n \geq 0\right\}$ is just one step of reduction, while $[\![\mathtt{R}_{\mathtt{X}}]\!] = \left\{\boldsymbol{n} \mapsto \frac{1}{2^{n+1}} \middle| n \geq 0\right\}$ was shown in Example 5.

□

Notice how simulating $\mathtt{X}$ by $\mathtt{R}$ requires the presence of recursion, while the converse is not true. The implications of this fact are intriguing, but lie outside the scope of this work.

In the following, we will no longer consider $\mathbb{T}^{\mathtt{X}}$ nor $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ but only $\mathbb{T}^{\mathtt{R}}$, keeping in mind that all these are equiexpressive due to Proposition 15. The rest of this paper, thus, will be concerned with understanding the relative expressive power of the three fragments $\mathbb{T}$, $\mathbb{T}^{\oplus}$, and $\mathbb{T}^{\mathtt{R}}$. Can any of the (obvious) strict *syntactical* inclusions between them be turned into a strict *semantic* inclusion? Are the three systems equiexpressive?

In order to compare probabilistic calculi to deterministic ones, several options are available. The most common one is to consider notions of observations over the probabilistic outputs; this will be the purpose of Section 6. In this section, we will look at whether applying a Monte Carlos or a Las Vegas algorithm on the output of a randomized function in $\mathbb{T}^{\oplus}$ or $\mathbb{T}^{\mathtt{R}}$ can enriched the set of deterministicaly $\mathbb{T}$-definable functions.

Notice that neither Monte Carlos nor Las Vegas algorithms are definable inside $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$. Indeed, for those algorithms to be applicable, they require restrictions on the resulting distribution that we are not able to describe in the calculus. For example, Las Vegas is only applicable to functions $M : \mathtt{NAT} \to \mathtt{NAT}$ such that $[\![M\ \boldsymbol{n}]\!](0) \leq \frac{1}{2}$ for any n.

Since those algorithms are not representable, we have to perform an external study that can be quite heavy. Hopefully, we where able to define an internal property, namely the (parameterized) *functional representability*, that is sufficient to collapse the results of both algorithms into the deterministic system $\mathbb{T}$.

We say that the distribution $\mathcal{M} \in \mathfrak{D}(\mathbb{N})$ is *finitely represented* by[2] $f : \mathbb{N} \to \mathbb{B}$, if there exists a q such that for every $k \geq q$ it holds that $f(k) = 0$ and

$$\mathcal{M} = \{\boldsymbol{k} \mapsto f(k)\}$$

Moreover, the definition can be extended to families of distributions $(\mathcal{M}_n)_n$ by requiring the existence of $f : (\mathbb{N} \times \mathbb{N}) \to \mathbb{B}$, $q : \mathbb{N} \to \mathbb{N}$ such that $\forall k \geq q(n), f(n,k) = 0$ and

$$\forall n, \quad \mathcal{M}_n = \ \{\boldsymbol{k} \mapsto f(n,k)\}\,.$$

In this case, we say that the representation is *parameterized*.

We will see in Section 4 that the distributions computed by $\mathbb{T}^{\oplus}$ are exactly the (parametrically) finitely representable by $\mathbb{T}$ terms. Concretely, this means that for any $M \in \mathbb{T}^{\oplus}(\mathtt{NAT})$ or any $M \in \mathbb{T}^{\oplus}(\mathtt{NAT} \to \mathtt{NAT})$, the distributions $[\![M]\!]$ and $([\![N\boldsymbol{n}]\!])_n$ are (parametrically) finitely representable.

In $\mathbb{T}^{\mathtt{R}}$, however, distributions are more complex (infinite, non-rational). That is why only a characterization in terms of approximations is possible. More specifically, a distribution $\mathcal{M} \in \mathfrak{D}(\mathtt{NAT})$ is said to be *functionally represented* by two functions $f : (\mathbb{N}\times\mathbb{N}) \to \mathbb{B}$ and $g : \mathbb{N} \to \mathbb{N}$ iff for every $n \in \mathbb{N}$ and for every $k \geq g(n)$ it holds that $f(n,k) = 0$ and

$$\sum_{k\in\mathbb{N}} \Big|\ \mathcal{M}(\boldsymbol{k})\ -\ f(n,k)\ \Big| \ \leq\ \frac{1}{n}.$$

In other words, the distribution $\mathcal{M}$ can be approximated arbitrarily well, and uniformly, by finitely representable ones. Similarly, we can define a parameterized version of this definition at first order.

In Section 5 , we show that distributions generated by $\mathbb{T}^{\mathtt{R}}$ terms are indeed uniform limits over those of $\mathbb{T}^{\oplus}$; using our result on $\mathbb{T}^{\oplus}$ this give their (parametric) functional representability in the deterministic $\mathbb{T}$.

4 Binary Probabilistic Choice

This section is concerned with two theoretical results on the expressive power of $\mathbb{T}^{\oplus}$. Taken together, they tell us that this fragment is not far from $\mathbb{T}$.

4.1 Positive Almost-Sure Termination

The average number of steps to normal form can be infinite for terms of $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$. We will prove that, on the contrary, $\mathbb{T}^{\oplus}$ is *positive* almost-surely terminating. This will be done by adapting (and strengthening!) the reducibility-based result from Section 3.3.

[2] Here we denote $\mathbb{B}$ for binomial numbers $\frac{m}{2^n}$ (where $m, n \in \mathbb{N}$) and BIN for their representation in system $\mathbb{T}$ encoded by pairs $\langle m, n\rangle$ of natural numbers.

$$\dfrac{}{M \Rightarrow \left\{M\right\}}\,(R\text{-}refl) \qquad \dfrac{M \to \mathcal{M} \quad \mathcal{M} \Rightarrow \mathcal{N}}{M \Rightarrow \mathcal{N}}\,(R\text{-}tran) \qquad \dfrac{\forall M \in |\mathcal{M}|,\ M \Rightarrow \mathcal{N}_M}{\mathcal{M} \Rightarrow \int_{\mathcal{M}} \mathcal{N}_M . dM}\,(R\text{-}\epsilon)$$

Figure 7: Multistep reduction. All terms and distributions are closed.

For this, we will first give a formalization of the notion of execution tree discussed in Section 2 in the form of a multistep reduction procedure. Then we will formally show that this tree is finite.

We can give another notion of evaluation via a notion of multistep reduction. We will see later on that the multistep reduction is none other than $\to^*$ for the system T with binary choice, but this is not always the case.

Definition 16. *The multistep reduction $\Rightarrow$ is defined by induction in Figure 7. Due to the (potentially) countably many preconditions of the rule (R-ϵ), the derivation tree of a multistep reduction $\Rightarrow$ can be infinitely wide and even of unbounded height, but each path have to be finite.*

The infinitness of the width and the fact that the height is unbounded is essentially in order to annalise $\mathbb{T}^{\mathtt{R}}$ in the same way. In fact, most theorems in this section will be given for both $\mathbb{T}^{\oplus}$ and $\mathbb{T}^{\mathtt{R}}$. But, for now, we will focus on $\mathbb{T}^{\oplus}$ and finite derivations, while $\mathbb{T}^{\oplus}$ and transfinite derivations will be analyzed in details in Section 5.1.

Example 17. *Take the example $rec\langle \mathbf{0}, \lambda xy.y \oplus (\mathtt{S}y), \mathbf{2}\rangle$; the execution tree is the following where we denote $U_n := rec\langle \mathbf{0}, \lambda xy.y \oplus (\mathtt{S}y), \boldsymbol{n}\rangle$:*

$$\begin{tikzcd}
 & & & & & & & & U_0 \arrow[r] & \mathbf{0} \\
 & & & & U_1 \arrow[r] & \bullet \arrow[r] & \bullet \arrow[r] & U_0 \oplus \mathtt{S}U_0 \arrow[ur, "\frac{1}{2}"] \arrow[dr, "\frac{1}{2}"'] & & \\
 & & & & & & & & \mathtt{S}U_0 \arrow[r] & \mathbf{1} \\
U_2 \arrow[r] & \bullet \arrow[r] & \bullet \arrow[r] & U_1 \oplus \mathtt{S}U_1 \arrow[uur, "\frac{1}{2}"] \arrow[ddr, "\frac{1}{2}"'] & & & & & & \\
 & & & & & & & & \mathtt{S}U_0 \arrow[r] & \mathbf{1} \\
 & & & & \mathtt{S}U_1 \arrow[r] & \bullet \arrow[r] & \bullet \arrow[r] & \mathtt{S}(U_0 \oplus \mathtt{S}U_0) \arrow[ur, "\frac{1}{2}"] \arrow[dr, "\frac{1}{2}"'] & & \\
 & & & & & & & & \mathtt{SS}U_0 \arrow[r] & \mathbf{2}
\end{tikzcd}$$

This tree is subsumed by the (more complex) derivation of $U_2 \Rightarrow \frac{1}{4}\{\mathbf{0}\} + \frac{1}{2}\{\mathbf{1}\} + \frac{1}{4}\{\mathbf{2}\}$, with every arrow of the exectution tree replaced by a (R-$tran$) rule followed by a (R-ϵ) rule:

$$\dfrac{U_2 \to \dfrac{\dfrac{(\lambda xy.y \oplus (\mathtt{S}y))\mathbf{1}U_1 \to \dfrac{\dfrac{(\lambda y.y \oplus (\mathtt{S}y))U_1 \to \dfrac{\dfrac{U_1 \oplus (\mathtt{S}U_1) \to \dfrac{\dfrac{U_1 \to \dfrac{\dfrac{(\lambda xy.y \oplus (\mathtt{S}y)\mathbf{1}U_0) \to \dfrac{\dfrac{(\lambda y.y \oplus (\mathtt{S}y))U_0 \to \dfrac{\dfrac{U_0 \oplus (\mathtt{S}U_0) \to \dfrac{\dfrac{U_0 \to \dfrac{\dfrac{}{\mathbf{1} \Rightarrow \{\mathbf{1}\}}\,(R\text{-}refl)}{\{\mathbf{1}\} \Rightarrow \{\mathbf{1}\}}\,(R\text{-}\epsilon)}{U_0 \Rightarrow \{\mathbf{1}\}}\,(R\text{-}tran) \quad \dfrac{\mathtt{S}U_0 \to \dfrac{\dfrac{}{\mathbf{2} \Rightarrow \{\mathbf{2}\}}\,(R\text{-}refl)}{\{\mathbf{2}\} \Rightarrow \{\mathbf{2}\}}\,(R\text{-}\epsilon)}{\mathtt{S}U_0 \Rightarrow \{\mathbf{2}\}}\,(R\text{-}tran)}{\frac{1}{2}\{U_0\} + \frac{1}{2}\{\mathtt{S}U_0\} \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}\epsilon)}{U_0 \oplus (\mathtt{S}U_0) \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}tran)}{\{U_0 \oplus (\mathtt{S}U_0)\} \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}\epsilon)}{(\lambda y.y \oplus (\mathtt{S}y))U_0 \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}tran)}{\{(\lambda y.y \oplus (\mathtt{S}y))U_0\} \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}}{(\lambda xy.y \oplus (\mathtt{S}y))\mathbf{1}U_0 \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}}{\{(\lambda xy.y \oplus (\mathtt{S}y))\mathbf{1}U_0\} \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}\epsilon)}{U_1 \Rightarrow \frac{1}{2}\{\mathbf{0}\} + \frac{1}{2}\{\mathbf{1}\}}\,(R\text{-}tran) \quad \dfrac{\mathtt{S}U_1 \to \dfrac{\dfrac{\mathtt{S}((\lambda xy.y \oplus (\mathtt{S}y))\mathbf{1}U_0) \to \dfrac{\dfrac{\mathtt{S}((\lambda y.y \oplus (\mathtt{S}y))U_0) \to \dfrac{\dfrac{\mathtt{S}(U_0 \oplus (\mathtt{S}U_0)) \to \dfrac{\dfrac{\mathtt{S}U_0 \to \dfrac{\dfrac{}{\mathbf{1} \Rightarrow \{\mathbf{1}\}}\,(R\text{-}refl)}{\{\mathbf{1}\} \Rightarrow \{\mathbf{1}\}}\,(R\text{-}\epsilon)}{\mathtt{S}U_0 \Rightarrow \{\mathbf{1}\}}\,(R\text{-}tran) \quad \dfrac{\mathtt{SS}U_0 \to \dfrac{\dfrac{}{\mathbf{2} \Rightarrow \{\mathbf{2}\}}\,(R\text{-}refl)}{\{\mathbf{2}\} \Rightarrow \{\mathbf{2}\}}\,(R\text{-}\epsilon)}{\mathtt{SS}U_0 \Rightarrow \{\mathbf{2}\}}\,(R\text{-}tran)}{\frac{1}{2}\{\mathtt{S}U_0\} + \frac{1}{2}\{\mathtt{SS}U_0\} \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}\epsilon)}{\mathtt{S}(U_0 \oplus (\mathtt{S}U_0)) \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}tran)}{\{\mathtt{S}(U_0 \oplus (\mathtt{S}U_0))\} \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}\epsilon)}{\mathtt{S}((\lambda y.y \oplus (\mathtt{S}y))U_0) \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}tran)}{\{\mathtt{S}((\lambda y.y \oplus (\mathtt{S}y))U_0)\} \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}\epsilon)}{\mathtt{S}((\lambda xy.y \oplus (\mathtt{S}y))\mathbf{1}U_0) \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}tran)}{\{\mathtt{S}((\lambda xy.y \oplus (\mathtt{S}y))\mathbf{1}U_0)\} \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}\epsilon)}{\mathtt{S}U_1 \Rightarrow \frac{1}{2}\{\mathbf{1}\} + \frac{1}{2}\{\mathbf{2}\}}\,(R\text{-}tran)}{\frac{1}{2}\{U_1\} + \frac{1}{2}\{\mathtt{S}U_1\} \Rightarrow \frac{1}{4}\{\mathbf{0}\} + \frac{1}{2}\{\mathbf{1}\} + \frac{1}{4}\{\mathbf{2}\}}\,(R\text{-}\epsilon)}{U_1 \oplus (\mathtt{S}U_1) \Rightarrow \frac{1}{4}\{\mathbf{0}\} + \frac{1}{2}\{\mathbf{1}\} + \frac{1}{4}\{\mathbf{2}\}}\,(R\text{-}tran)}{\{U_1 \oplus (\mathtt{S}U_1)\} \Rightarrow \frac{1}{4}\{\mathbf{0}\} + \frac{1}{2}\{\mathbf{1}\} + \frac{1}{4}\{\mathbf{2}\}}\,(R\text{-}\epsilon)}{(\lambda y.y \oplus (\mathtt{S}y))U_1 \Rightarrow \frac{1}{4}\{\mathbf{0}\} + \frac{1}{2}\{\mathbf{1}\} + \frac{1}{4}\{\mathbf{2}\}}\,(R\text{-}tran)}{\{(\lambda y.y \oplus (\mathtt{S}y))U_1\} \Rightarrow \frac{1}{4}\{\mathbf{0}\} + \frac{1}{2}\{\mathbf{1}\} + \frac{1}{4}\{\mathbf{2}\}}\,(R\text{-}\epsilon)}{(\lambda xy.y \oplus (\mathtt{S}y))\mathbf{1}U_1 \Rightarrow \frac{1}{4}\{\mathbf{0}\} + \frac{1}{2}\{\mathbf{1}\} + \frac{1}{4}\{\mathbf{2}\}}\,(R\text{-}tran)}{\{(\lambda xy.y \oplus (\mathtt{S}y))\mathbf{1}U_1\} \Rightarrow \frac{1}{4}\{\mathbf{0}\} + \frac{1}{2}\{\mathbf{1}\} + \frac{1}{4}\{\mathbf{2}\}}\,(R\text{-}\epsilon)}{U_2 \Rightarrow \frac{1}{4}\{\mathbf{0}\} + \frac{1}{2}\{\mathbf{1}\} + \frac{1}{4}\{\mathbf{2}\}}\,(R\text{-}tran)$$

Notice that we are contracting the rule (R-$tran$) for readability. More important, notice also that the derivation is correct and finite because the execution tree is finite.

Lemma 18. *The multistep semantics $\Rightarrow$ is confluent.*

Proof. By an easy induction, we show that if $\mathcal{N}_1 \Leftarrow M \Rightarrow \mathcal{N}_2$ (resp. $\mathcal{N}_1 \Leftarrow \mathcal{M} \Rightarrow \mathcal{N}_2$), then there is $\mathcal{L}$ such that $\mathcal{N}_1 \Rightarrow \mathcal{L} \Leftarrow \mathcal{N}_2$. Now:

- If both reductions are using the same rule (either (R-$refl$), (R-$tran$), or (R-$\in$)), then it is an immediate use of the induction hypothesis on the premises as those rules are determinist.
- If one of them use the rule (R-$refl$), then it is trivial.
- No other case is possible as (R-$tran$) and (R-$\in$) cannot apply together (one require a term as source and the other a distribution).

$\square$

Lemma 19. *If $(M\ V) \Rightarrow \mathcal{U} \in \mathfrak{D}(\mathbb{T}_V^{\oplus,\mathtt{R},\mathtt{X}})$ then there is $\mathcal{W} \in \mathfrak{D}(\mathbb{T}_V^{\oplus,\mathtt{R},\mathtt{X}})$ such that $M \Rightarrow \mathcal{W}$.*

Proof. By induction on $\Rightarrow$ we can show that if $(M\ V) \Rightarrow \mathcal{N}$ then $\mathcal{N} = \mathcal{L} + (\mathcal{M}\ V)$ with $M \Rightarrow \mathcal{W} + \mathcal{M}$ and $(\mathcal{W}\ V) \Rightarrow \mathcal{L}$ (and similarly if M is a distribution):

- the (R-$refl$) and (R- $\in$) are trivial,
- if $(M\ V) \to \mathcal{N}' \Rightarrow \mathcal{N}$ then there is two cases:
  - either $M \to \mathcal{M}'$ and $\mathcal{N}' = (\mathcal{M}'\ V) \Rightarrow \mathcal{N}$ and we can conclude by induction hypothesis,
  - or $M = W$ is a value and $M \Rightarrow \{W\}$ with $(\{W\}\ V) \to \mathcal{N}' \Rightarrow \mathcal{N} = \mathcal{L}$.

Notice that if $\mathcal{N}$ is a value distribution then $\mathcal{M}$ has to be one. $\square$

The following lemma is an alternative version of Lemma 10 where the evaluation is obtained after a finite number of steps (in both hypothesis and conclusions).

Lemma 20. *If $N \Rightarrow \mathcal{V} \in \mathfrak{D}(\mathbb{T}_V^{\oplus,\mathtt{R},\mathtt{X}})$ and $M\ \mathcal{V} \Rightarrow \mathcal{U} \in \mathfrak{D}(\mathbb{T}_V^{\oplus,\mathtt{R},\mathtt{X}})$ then $M\ N \Rightarrow \mathcal{U}$.*

Proof. By induction on the derivation of $N \Rightarrow V$ (generalizing the property for any distribution $\mathcal{N}$ in place of N):

- If $V = \{N\}$ this is trivial.
- If $M \to \mathcal{N} \Rightarrow \mathcal{V}$ then by IH, $M\ \mathcal{N} \Rightarrow U$ and thus $M\ N \to M\ \mathcal{N} \Rightarrow U$ so that we can conclude by rule (R-$trans$).
- If for all $N \in |\mathcal{N}|$, $N \Rightarrow \mathcal{V}_N$ with $\mathcal{V} = \int_{\mathcal{N}} \mathcal{V}_N dN$ then by applying the IH on each $N \in |\mathcal{N}|$, we get that $M\ N \Rightarrow \mathcal{U}_N$ for some $\mathcal{U}_N$ and we conclude by rule (R- $\in$) using $\mathcal{U} = \int_{\mathcal{N}} \mathcal{U}_N dN$.

$\square$

Theorem 21. *For any term $M \in \mathbb{T}^{\oplus,\mathtt{R}}$, there exists a value distribution $[\![M]\!]_a \in \mathfrak{D}(\mathbb{T}_V^{\oplus,\mathtt{R}})$ such that $M \Rightarrow [\![M]\!]_a$. We call it the accessible evaluation.*

Proof. When it exists, $[\![M]\!]_a$ is unique due to confluence. Thus we only have to prove its existence. The proof goes by reducibility over the candidates:

$$\begin{aligned}
Red_{\mathtt{NAT}} &:= \{M \in \mathbb{T}^{\oplus}(\mathtt{NAT}) \mid \exists\, [\![M]\!]_a \in \mathfrak{D}(\mathbb{T}_V^{\oplus}),\ M \Rightarrow [\![M]\!]_a\} \\
Red_{A\to B} &:= \{M \mid \forall V \in Red_A \cap \mathbb{T}_V^{\oplus},\ (M\ V) \in Red_B\} \\
Red_{A\times B} &:= \{M \mid (\pi_1\ M) \in Red_A,\ (\pi_2\ M) \in Red_B\}
\end{aligned}$$

1. *The reducibility candidates over Red_A are $\to$-saturated:*
   By induction on A we can show that if $M \to \mathcal{M}$ then $|\mathcal{M}| \subseteq Red_A$ iff $M \in Red_A$.
   - If $A = \mathtt{NAT}$: then whenever $M' \Rightarrow [\![M']\!]_a$ for all $M' \in |\mathcal{M}|$ we get $\mathcal{M} \Rightarrow \int_{\mathcal{M}} [\![M']\!]_a dM' = [\![\mathcal{M}]\!]_a$ by (R- $\in$) and thus $M \Rightarrow [\![\mathcal{M}]\!]_a = [\![M]\!]_a$ by (R-$trans$). Conversely, whenever $M \Rightarrow [\![M]\!]_a$, this reduction cannot comes from rule (R-$refl$) since $[\![M]\!]_a$ is a value distribution and M is reducible, thus it comes from rule (R-$trans$) and $\mathcal{M} \Rightarrow [\![\mathcal{M}]\!]_a$ which itself necessarily comes from an application of (R- $\in$) so that $M' \Rightarrow [\![M']\!]_a$ for any $M' \in |\mathcal{M}|$.

- If $A = B \to C$: then for all value $V \in Red_B$, $(M\ V) \to (\mathcal{M}\ V)$, thus by IH $(M\ V) \in Red_C$ iff $|\mathcal{M}\ V| = \{M'\ V \mid M' \in |\mathcal{M}|\} \subseteq Red_C$; which exactly means that $|\mathcal{M}| \subseteq Red_{B\to C}$ iff $M \in Red_{B\to C}$.
- If $A = A_1 \times A_2$: then for $i \in \{1,2\}$, $(\pi_i\ M) \to (\pi_i\ \mathcal{M})$ so that by IH, $(\pi_i\ M) \in Red_{A_i}$ iff $(\pi_i\ \mathcal{M}) \subseteq Red_{A_i}$; which exactly means that $|\mathcal{M}| \subseteq Red_{A_1\times A_2}$ iff $M \in Red_{A_1\times A_2}$.

2. *The reducibility candidates over Red_A are $\Rightarrow$-saturated:*
By a trivial induction on $\Rightarrow$ using the $\to$-saturation for the (*R-trans*) case.

3. *Red_A is inhabited by a value:*
By induction on A: $\mathbf{0} \in Red_{\mathtt{NAT}}$, $\lambda x.V \in Red_{A\to B}$ and $\langle U, V\rangle \in Red_{A\times B}$ whenever $U \in Red_A$ and $V \in Red_B$.

4. *The reducibility candidates M over Red_A $\Rightarrow$-reduce to $[\![M]\!]_a$:*
By induction on A:
   - Trivial for $A = \mathtt{NAT}$.
   - Let $M \in Red_{B\to C}$, there is a value $V \in Red_B$, thus $(M\ V) \in Red_C$ and $M\ V \Rightarrow [\![M\ V]\!]_a$ by IH; we can conclude using Lemma 19.
   - Similar for products.

5. *Every term $\vec{x} : \vec{A} \vdash M : B$ is a candidate in the sense that if $\vec{V} \in \vec{Red_A}$ then $M[\vec{V}/\vec{x}] \in Red_B$:*
By induction on the type derivation: The only difficult cases, the application and the recursion and the binary probabilistic operator:
   - For the application: we have to show that if $M \in Red_{A\to B}$ and $N \in Red_A$ then $(M\ N) \in Red_B$. But since $N \in Red_A$ we get that $N \Rightarrow [\![N]\!]_a$ with $|\,[\![N]\!]_a\,| \subseteq Red_A$. The means that $|M\ \ [\![N]\!]_a| \subseteq Red_B$ and that $M\ \ [\![N]\!]_a \Rightarrow [\![M\ \ [\![N]\!]_a]\!]_a$ supported into Red_B. We conclude by Lemma 20 that $\mathcal{U} = [\![M\ N]\!]_a$ and thus that $(M\ N) \in Red_B$.
   - For the operator $\mathtt{rec}$: We have to show that if $U \in Red_A$ and $V \in Red_{\mathtt{NAT}\to A\to A}$ then for all $n \in \mathbb{N}$, $(\mathtt{rec}\ \langle U, V, \boldsymbol{n}\rangle) \in Red_A$. We proceed by induction on n:
     - If $n = 0$: $\mathtt{rec}\ \langle U, V, \mathbf{0}\rangle \to \{U\} \subseteq Red_A$ and we conclude by saturation.
     - Otherwise: $\mathtt{rec}\ \langle U, V, (\boldsymbol{n}+\mathbf{1})\rangle \to \{V\ \boldsymbol{n}\ (\mathtt{rec}\ \langle U, V, \boldsymbol{n}\rangle)\} \subseteq Red_A$ since $(\mathtt{rec}\ \langle U, V, \boldsymbol{n}\rangle) \in Red_A$ by IH and since $\boldsymbol{n} \in Red_{\mathtt{NAT}}$ and $V \in Red_{\mathtt{NAT}\to A\to A}$, we conclude by saturation.
   - For the operator $\oplus$: If $M, N \in Red_A$ then $\left|\left\{\begin{matrix} M \mapsto \frac{1}{2} \\ N \mapsto \frac{1}{2}\end{matrix}\right\}\right| \subseteq Red_A$, and, by $\to$-saturation, $(M \oplus N) \in Red_A$.

□

Notice that this theorem dose not applies in $\mathbb{T}^{\mathtt{X}}$ (and a fortiori in $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$) because the step (5) of the proof would not be verified.

Theorem 22. *The accessible evaluation of a term $M \in \mathbb{T}^{\oplus,\mathtt{R}}$ is its evaluation. Moreover, any term M is almost surely terminating.*

Proof. By a trivial induction on $\Rightarrow$, we can easily show that if $M \Rightarrow \mathcal{M}$ then $[\![M]\!] = [\![\mathcal{M}]\!]$. In particular, $[\![M]\!] = [\![[\![M]\!]_a]\!] = [\![M]\!]_a$.

By a trivial induction on $\Rightarrow$ we can show that if $M \Rightarrow \mathcal{N}$ then $||\mathcal{N}|| = 1$ and that if $\mathcal{M} \Rightarrow \mathcal{N}$ then $||\mathcal{M}|| = ||\mathcal{N}||$. □

Corollary 23. *Any term $M \in \mathbb{T}^{\oplus}$ is positively almost-surely terminating.*

Proof. By an induction on $\Rightarrow$ we can show that if $M \Rightarrow [\![M]\!]$ (resp. $\mathcal{M} \Rightarrow [\![\mathcal{M}]\!]$ for $\mathcal{M}$ finitely supported) then $M \to^* [\![M]\!]$ (resp. $\mathcal{M} \to^* [\![\mathcal{M}]\!]$):
- (*R-refl*) is trivial.
- (*R-trans*) is immediate once we remark that in $\mathbb{T}^{\oplus}$ whenever $M \to \mathcal{M}$, necessarily $\mathcal{M}$ is finitely supported and we can use our induction hypothesis.

- If for all $M \in |\mathcal{M}|$, $M \Rightarrow [\![M]\!]$, then by IH, $M \to^{n_M} [\![M]\!]$ for some $n_M \in \mathbb{N}$. Moreover, since $|\mathcal{M}|$ is finite, we can set $n = sup_{M\in|\mathcal{M}|}$ so that $M \to^{\leq n} [\![M]\!]$ and $\mathcal{M} \to^{\leq n} \int_{\mathcal{M}} [\![M]\!]\, dM = [\![\mathcal{M}]\!]$.

The reduction time of a term is then bounded by n such that $M \to^{n} [\![M]\!]$. □

Notice that this theorem dose not applies in $\mathbb{T}^{\mathtt{R}}$ (and a fortiori in $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$) because the second bullet of the proof would not be verified.

4.2 Mapping to $\mathbb{T}$

This positive almost sure convergence is not the only consequence of Theorem 22. In fact, the finitedness of the resulting distribution over values allows a finite representation of $\mathbb{T}^{\oplus}$-distributions by $\mathbb{T}$-definable functions.

Indeed, we can consider an extension of the usual system T with a single memory cell of type `NAT` that we use to store binary encoding of the determination of every probabilistic choices during the execution. If we denote c the memory-cell, this means that the $\oplus$ can be encoded:[3]

$$(M \oplus N)^* \quad := \quad \mathtt{if}\ (\mathtt{mod}_2\ c)\ \mathtt{then}\ (c{:=}\frac{c}{2}\ ;\ M^*)\ \mathtt{else}\ (c{:=}\frac{c}{2}\ ;\ N^*)$$

From theorem 21, we know that for any $M \in \mathbb{T}^{\oplus}(\mathtt{NAT})$, there is $n \in \mathbb{N}$ such that $M \to^{n} [\![M]\!]$. Since the execution is bounded by n, there cannot be more than n successive probabilistic choice so that:

$$[\![M]\!] = \left\{ k \mapsto \frac{\#\{m < 2^n \mid k = \mathbb{NF}(c{:=}\boldsymbol{m}\ ;\ M^*)\}}{2^n} \right\}.$$

Using a well known state-passing style transformation, we can enforce $(c{:=}\boldsymbol{m}\ ;\ M^*)$ into a term of $\mathbb{T}$. Then, using a simple recursive operation on m, we can represent the whole $\#\{m < 2^n \mid k = \mathbb{NF}(c{:=}\boldsymbol{m}\ ;\ M^*)\}$ into the result of a term $k : \mathbb{N} \vdash M' : \mathbb{N}$ so that $\lambda k.M'$ define a function that represent the distribution $[\![M]\!]$.

Example 24. *Take the term* $M\ =\ \mathtt{rec}\ \langle \mathbf{0},\ \lambda xy.y \oplus \mathtt{S}y, \mathbf{2}\rangle$ *from Example 17. Then the encoding is*

$$M^* = \mathtt{rec}\langle \mathbf{0},\ \lambda xy.\,\mathtt{if}\ (\mathtt{mod}\,c\,\mathbf{2})\,\mathtt{then}\,(c{:=}\frac{c}{2}\ ;\ y)\ \mathtt{else}\ (c{:=}\frac{c}{2}\ ;\ \mathtt{S}y),\ \mathbf{2}\rangle$$

with a state passing style (and a few simplifications) we obtain the term:

$$M^{\sim}\ =\ \mathtt{rec}\ \langle \lambda c.(\mathbf{0}, c),\ \lambda xyc.\,\mathtt{if}\ (\mathtt{mod}\,c\,\mathbf{2})\,\mathtt{then}\,y\ (\mathtt{div}\,c\,\mathbf{2})\,\mathtt{else}\,\mathtt{S}\ (y\ (\mathtt{div}\,c\,\mathbf{2})),\ \mathbf{2}\rangle$$

As we know that there is at most two choices, we can count the number of c below 4 *which result to a certain u, we get:*

$$M^{\$}\ :=\ \lambda u.\,\mathtt{rec}\ \langle \mathbf{0},\ \lambda xy.\,\mathtt{if}\ (\pi_1(M^{\sim}x) == u)\ \mathtt{then}\ \mathtt{S}y\ \mathtt{else}\ y,\ \mathbf{4}\rangle$$

Then we have:

$$[\![M]\!] = \left\{ k \mapsto \frac{\mathbb{NF}(M^{\$}\ \boldsymbol{k})}{4} \right\}.$$

Remains to show that this encoding can be parameterized in the sens that for any $M \in \mathbb{T}^{\oplus}(\mathtt{NAT} \to \mathtt{NAT})$, we can generate $M_{\downarrow} \in \mathbb{T}(\mathtt{NAT} \to \mathtt{NAT} \to \mathtt{NAT})$ and $M_{\#} \in \mathbb{T}(\mathtt{NAT} \to \mathtt{NAT})$ such that for all $n \in \mathbb{N}$:

$$[\![M\ \boldsymbol{n}]\!] \quad = \quad \left\{ \boldsymbol{k} \mapsto \frac{\#\{m < 2^{\mathbb{NF}(M_{\#}\ \boldsymbol{n})} \mid k = \mathbb{NF}(M_{\downarrow}\ \boldsymbol{n})\}}{2^{\mathbb{NF}(M_{\#}\ \boldsymbol{n})}} \right\}.$$

The supplementary difficulty, here, comes from the bound $M_{\#}$ that have to be computed dynamically in a more complex monadic encoding.

[3]Notice that conditionals, parity and fractions are easily implementable in $\mathbb{T}$.

First of all, let us define two maps $(\!(\cdot)\!)$ and $(\!(\cdot)\!)_{\boldsymbol{V}}$ on types as follows:

$$(\!(A)\!) := \big(\mathtt{NAT} \to (\!(A)\!)_{\boldsymbol{V}}\big) \times \mathtt{NAT}\,; \qquad (\!(\mathtt{NAT})\!)_{\boldsymbol{V}} := \mathtt{NAT}\,;$$
$$(\!(A \to B)\!)_{\boldsymbol{V}} := (\!(A)\!)_{\boldsymbol{V}} \to (\!(B)\!)\,; \qquad (\!(A \times B)\!)_{\boldsymbol{V}} := (\!(A)\!)_{\boldsymbol{V}} \times (\!(B)\!)_{\boldsymbol{V}}\,.$$

This can be seen as the monadic lifting of the probabilistic monad. The embedding is centered around two maps, $(\!(\cdot)\!)$ from $\mathbb{T}^{\oplus}$ terms to $\mathbb{T}$ terms and $(\!(\cdot)\!)_{\boldsymbol{V}}$ from $\mathbb{T}^{\oplus}$ extended values to $\mathbb{T}$ extended values. These maps are such that, whenever $\Gamma \vdash M : A$ and $\Gamma \vdash V : A$, it holds that

$$(\!(\Gamma)\!)_{\boldsymbol{V}} \vdash (\!(M)\!) : (\!(A)\!) \qquad (\!(\Gamma)\!)_{\boldsymbol{V}} \vdash (\!(V)\!)_{\boldsymbol{V}} : (\!(A)\!)_{\boldsymbol{V}}\,.$$

With a slight abuse of notation, we see the type $(\!(A)\!)$, which by definition is a product type, as given through two components $(\!(A)\!)_{\downarrow} := \mathtt{NAT} \to (\!(A)\!)_{\boldsymbol{V}}$ and $(\!(A)\!)_{\#} := \mathtt{NAT}$. Accordingly, we denote $M_{\downarrow} := \pi_1 M : (\!(A)\!)_{\downarrow}$ and $M_{\#} := \pi_2 M : (\!(A)\!)_{\#}$ whenever $M : (\!(A)\!)$. Similarly, we may directly define $(\!(M)\!)_{\downarrow}$ and $(\!(M)\!)_{\#}$ whenever $(\!(M)\!) = \langle (\!(M)\!)_{\downarrow}, (\!(M)\!)_{\#} \rangle$.

We give a relatively precise (although laborious) definition of the maps above. What is important for the rest of the development is that for every natural number n, $(\!(\boldsymbol{n})\!)_{\boldsymbol{V}} = \boldsymbol{n}$, and that $(\!(\lambda y.M)\!)_{\boldsymbol{V}} = \lambda y.(\!(M)\!)$.

The encoding $(\!(\cdot)\!)_{\boldsymbol{V}}$ of extended values is given by:

$$(\!(\mathtt{S})\!)_{\boldsymbol{V}} := \lambda y.(\!(\mathtt{S}\ y)\!)_{\boldsymbol{V}} \qquad (\!(\mathtt{S}\ V)\!)_{\boldsymbol{V}} := \mathtt{S}\ (\!(V)\!)_{\boldsymbol{V}} \qquad (\!(\mathbf{0})\!)_{\boldsymbol{V}} := \mathbf{0}$$
$$(\!(\langle M, N\rangle)\!)_{\boldsymbol{V}} := \langle (\!(M)\!)_{\boldsymbol{V}}, (\!(N)\!)_{\boldsymbol{V}} \rangle \qquad (\!(\lambda x.M)\!)_{\boldsymbol{V}} := \lambda x.(\!(M)\!) \qquad (\!(x)\!)_{\boldsymbol{V}} := x$$
$$(\!(\pi_i)\!)_{\boldsymbol{V}} := \lambda x.\mathtt{ret}(\pi_i x) \qquad (\!(\mathtt{rec})\!)_{\boldsymbol{V}} := \lambda\langle u, v, w\rangle.\mathtt{rec}\,\langle \mathtt{ret}\,u,\ \lambda xy.(vx) \mathbin{=\!\!\ll} y,\ w\rangle$$

Where $\mathtt{ret}$ and $\mathbin{=\!\!\ll}$ are the return and the bind of the considered monad:[4]

$$\mathtt{ret} : (\!(A)\!)_{\boldsymbol{V}} \to (\!(A)\!) \qquad \mathtt{ret} := \lambda x.\langle \lambda y.x,\ \mathbf{0}\rangle$$
$$\_ \mathbin{=\!\!\ll} \_ : (\!(A \to B)\!) \times (\!(A)\!) \to (\!(B)\!) \qquad M \mathbin{=\!\!\ll} N := \langle M \mathbin{=\!\!\ll}_{\downarrow} N, M \mathbin{=\!\!\ll}_{\#} N\rangle$$

However, the computation of the bind is extremely complex. Intuitively, the right part is computing the number of choices in M in N and in all the possibles outcomes $(U\ V)$ for $U \in |[\![M]\!]|$ and $V \in |[\![N]\!]|$; off course we take the sup of those outcomes as they are independent:

$$\_ \mathbin{=\!\!\ll}_{\#} \_ : (\!(A \to B)\!) \times (\!(A)\!) \to \mathtt{NAT}$$
$$M \mathbin{=\!\!\ll}_{\#} N := M_{\#} + N_{\#} + \max_{x < 2^{M_{\#}}} \max_{y < 2^{N_{\#}}} (M_{\downarrow}\ x\ (N_{\downarrow}\ y))_{\#}$$

where we use the following syntactical sugar:

$$\_ + \_ : \mathtt{NAT} \times \mathtt{NAT} \to \mathtt{NAT} \qquad M + N := \mathtt{rec}\langle M,\ \lambda\_x.\mathtt{S}x,\ N\rangle$$
$$2^{\_} : \mathtt{NAT} \to \mathtt{NAT} \qquad 2^{M} := \mathtt{rec}\langle \mathbf{1},\ \lambda\_y.\mathtt{rec}\langle y, \lambda\_v.\mathtt{S}v, y\rangle,\ M\rangle$$
$$\max_{x < M}(N) := \mathtt{rec}\langle \mathbf{0},\ \lambda\_y.\ N \vee y,\ M\rangle$$
$$\_ \vee \_ : \mathtt{NAT} \times \mathtt{NAT} \to \mathtt{NAT} \qquad M \vee N := \mathtt{rec}\langle \lambda u.u,\ \lambda xyu.\mathtt{S}(\mathtt{rec}\langle x, \lambda a\_.ya, u\rangle),\ M\rangle\ N$$

The first member of the bind is also complex. It is taking a streams of probabilistic choices s, computing M over this choices, then we have to shift the stream s in order to remove the choices relative to M, this way we can compute the result from N, that we can give to the result from N. After this, we obtain an object of type $(\!(B)\!)$; but we have not finished: we have to select the first member, to which we give what remain of the stream (computed by shifting x twice).

$$\_ \mathbin{=\!\!\ll}_{\downarrow} \_ : (\!(A \to B)\!) \times (\!(A)\!) \to \mathtt{NAT} \to (\!(B)\!)_{\boldsymbol{V}}$$
$$M \mathbin{=\!\!\ll}_{\downarrow} N := \lambda s.(M_{\downarrow}\ s\ (N_{\downarrow}\ (\mathtt{shift}\ s\ M_{\#})))_{\downarrow}\ (\mathtt{shift}\ s\ (M_{\#} + N_{\#}))$$

[4]Technically, this bind is not the usual bind, but a lifted version.

where we use the following syntactical sugar:

$$\begin{array}{ll} \texttt{shift} : \texttt{NAT} \to \texttt{NAT} \to \texttt{NAT} & \texttt{shift} := \lambda sy.\texttt{rec}\langle s,\ \lambda u.div_2,\ y\rangle \\ \texttt{div}_2 : \texttt{NAT} \to \texttt{NAT} & \texttt{div}_2 := \lambda x.\texttt{rec}\langle \lambda\_.\mathbf{0},\ \lambda\_vw.\texttt{ite}\langle w, \texttt{S}(v\mathbf{0}), v\mathbf{1}\rangle,\ \ x\rangle\ \mathbf{0} \end{array}$$

The encoding $(\!(\cdot)\!)$ is given by the return and the bind operations :

$$(\!(V)\!) := \texttt{ret}\ (\!(V)\!)_{\boldsymbol{V}} \qquad\qquad (\!(M\ N)\!) := (\!(M)\!) {=\!\!\ll} (\!(N)\!)$$

The binary choice is the defined as intuited:

$$(\!(M \oplus N)\!)_{\#} := (\!(M)\!)_{\#} \vee (\!(N)\!)_{\#} \qquad (\!(M \oplus N)\!)_{\downarrow} := \lambda x.\texttt{ite}\langle \texttt{mod}_2 x,\ (\!(M)\!)_{\downarrow}(\texttt{div}_2 x),\ (\!(N)\!)_{\downarrow}(\texttt{div}_2 x)\rangle$$

where we use the following syntactical sugar:

$$\begin{array}{ll} \texttt{ite} : \texttt{NAT} \times A \times A \to A & \texttt{ite} := \lambda x.\texttt{rec}\ \langle \pi_3 x,\ \lambda\_\_.\pi_2 x,\ \pi_1 x\rangle\ . \\ \texttt{mod}_2 : \texttt{NAT} \to \texttt{NAT} & \texttt{mod}_2 := \lambda x.\texttt{rec}\langle \mathbf{0},\ \lambda\_v.\texttt{ite}\langle v, \mathbf{0}, \mathbf{1}\rangle,\ \ x\rangle \end{array}$$

Theorem 25. *Distributions in $\mathbb{T}^{\oplus}$ are finitely parameterically representable by $\mathbb{T}$-definable functions,* i.e. *for any $M : \texttt{NAT} \to \texttt{NAT}$ in $\mathbb{T}^{\oplus}$ there is $F : \texttt{NAT} \to \texttt{NAT} \to \texttt{BIN}$ and $Q : \texttt{NAT} \to \texttt{NAT}$ in $\mathbb{T}$ such that for all n:*

$$[\![M\ \boldsymbol{n}]\!]\ =\ \{\boldsymbol{k} \mapsto \mathbb{NF}(F\ \boldsymbol{n}\ \boldsymbol{k})\} \qquad\qquad \forall k \geq \mathbb{NF}(Q\ \boldsymbol{n}),\ \mathbb{NF}(F\ \boldsymbol{n}\ \boldsymbol{k}) = \mathbf{0}.$$

5 Countable Probabilistic Choice

5.1 Multistep semantics

We have seen that none of Theorems 21, 22 and 23, about the multistep semantics, hold anymore for the $\mathbb{T}^{\texttt{X}}$. Indeed Theorem 21, which is a premise to the two others, is broken by the term $\texttt{X}\langle \mathbf{0}, \texttt{S}\rangle$ that will never $\Rightarrow$-reduce to a value distribution.

The fragment $\mathbb{T}^{\texttt{R}}$ is more interesting as both Theorem 21 and Theorem 22 hold. However, as we have seen in Theorem 14, the positive almost sure normalization (and Theorem 23) does not hold. This is because we are manipulating infinitely supported distributions (due to the reduction rule of $\texttt{R}$).

Example 26. *Recall that $\texttt{X}_{\texttt{R}}\ :=\ \lambda x.\texttt{rec}\ \langle \pi_2 x, \lambda z.\pi_1 x, \texttt{R}\rangle$ is the encoding of $\texttt{X}$ into $\mathbb{T}^{\texttt{R}}$. We have $\texttt{X}_{\texttt{R}}\ \langle \mathbf{0}, \texttt{SS}\rangle \Rightarrow \{\boldsymbol{2n} \mapsto \frac{1}{2^{n+1}} \mid n \geq 0\}$; indeed, if we fix $U_n = \texttt{rec}\ \langle \pi_2\langle \mathbf{0}, \texttt{SS}\rangle, \lambda z.\pi_1\langle \mathbf{0}, \texttt{SS}\rangle, \boldsymbol{n}\rangle$:*

$$\dfrac{\dfrac{\dfrac{U_{\texttt{R}} \to \dfrac{\dfrac{\dfrac{U_0 \to \dfrac{\dfrac{\pi_2\langle \mathbf{0}, \texttt{SS}\rangle \to \dfrac{\dfrac{}{\mathbf{0} \Rightarrow \{\mathbf{0}\}}\,(R\text{-}refl)}{\{\mathbf{0}\} \Rightarrow \{\mathbf{0}\}}\,(R\text{-}\epsilon)}{\pi_2\langle \mathbf{0}, \texttt{SS}\rangle \Rightarrow \{\mathbf{0}\}}\,(R\text{-}tran)}{\{\pi_2\langle \mathbf{0}, \texttt{SS}\rangle\} \Rightarrow \{\mathbf{0}\}}\,(R\text{-}\epsilon)}{U_0 \Rightarrow \{\mathbf{0}\}}\,(R\text{-}tran) \quad \dfrac{U_1 \to \dfrac{\dfrac{\pi_1\langle \mathbf{0}, \texttt{SS}\rangle\, U_0 \to \dfrac{\dfrac{\pi_1\langle \mathbf{0}, \texttt{SS}\rangle\,(\pi_2\langle \mathbf{0}, \texttt{SS}\rangle) \to \dfrac{\dfrac{\pi_1\langle \mathbf{0}, \texttt{SS}\rangle\,\mathbf{0} \to \dfrac{\dfrac{}{\texttt{SS}\ \mathbf{0} \Rightarrow \{\mathbf{2}\}}\,(R\text{-}refl)}{\{\texttt{SS}\ \mathbf{0}\} \Rightarrow \{\mathbf{2}\}}\,(R\text{-}\epsilon)}{\pi_1\langle \mathbf{0}, \texttt{SS}\rangle\,\mathbf{0} \Rightarrow \{\mathbf{2}\}}\,(R\text{-}tran)}{\{\pi_1\langle \mathbf{0}, \texttt{SS}\rangle\,\mathbf{0}\} \Rightarrow \{\mathbf{2}\}}\,(R\text{-}\epsilon)}{\pi_1\langle \mathbf{0}, \texttt{SS}\rangle\,(\pi_2\langle \mathbf{0}, \texttt{SS}\rangle) \Rightarrow \{\mathbf{2}\}}\,(R\text{-}tran)}{\{\pi_1\langle \mathbf{0}, \texttt{SS}\rangle\,(\pi_2\langle \mathbf{0}, \texttt{SS}\rangle)\} \Rightarrow \{\pi_1\langle \mathbf{0}, \texttt{SS}\rangle\,\mathbf{0}\}}\,(R\text{-}\epsilon)}{\pi_1\langle \mathbf{0}, \texttt{SS}\rangle\, U_0 \Rightarrow \{\mathbf{2}\}}\,(R\text{-}tran)}{\{\pi_1\langle \mathbf{0}, \texttt{SS}\rangle\, U_0\} \Rightarrow \{\mathbf{2}\}}\,(R\text{-}\epsilon)}{U_1 \Rightarrow \{\mathbf{2}\}}\,(R\text{-}tran) \quad \cdots}{\{U_n \mapsto \frac{1}{2^{n+1}} \mid n \geq 0\} \Rightarrow \{\boldsymbol{2n} \mapsto \frac{1}{2^{n+1}} \mid n \geq 0\}}\,(R\text{-}\epsilon)}{U_{\texttt{R}} \Rightarrow \mathcal{N}}\,(R\text{-}tran)}{\texttt{X}\langle \mathbf{0}, \texttt{SS}\rangle \to \{U_{\texttt{R}}\} \Rightarrow \{\boldsymbol{2n} \mapsto \frac{1}{2^{n+1}} \mid n \geq 0\}}\,(R\text{-}\epsilon)}{\texttt{X}\langle \mathbf{0}, \texttt{SS}\rangle \Rightarrow\ \{\boldsymbol{2n} \mapsto \frac{1}{2^{n+1}} \mid n \geq 0\}}\,(R\text{-}tran)$$

we can see that the tree is infinite due to the second application of $(R\text{-}\epsilon)$; but, despite being infinite, and with an infinite height, each subtree above the second application of $(R\text{-}\epsilon)$ is finite, making the derivation correct.

However, $\mathtt{X}_\mathtt{R}\langle \mathbf{0}, \mathtt{SS}\rangle \not\Rightarrow \{\mathbf{2n} \mapsto \frac{1}{2^{n+1}} \mid n \geq 0\}$. *We can aproach this distribution, but it is impossible to $\Rightarrow$-reduce to a value distribution.*

Remember that $\mathbb{T}^\mathtt{R}$ and $\mathbb{T}^\mathtt{X}$ are equivalent, so why such a difference? This is due to the difference of nature in their execution trees. Indeed, we have seen that the execution trees of $\mathbb{T}^\mathtt{X}$ are finitely branching, but with infinite paths, while those of $\mathbb{T}^\mathtt{R}$ are infinitely branching, but with finite paths. The inference of the multistep reduction following those execution trees, we can see that we only need derivations with infinite arity to get a correct multistep semantics for $\mathbb{T}^\mathtt{R}$.

The whole point is that we can perform transfinite structural inductions over these trees. Indeed, by considering the reduction trees themselves with the inclusion (or subtree) order gives you a well-founded poset, recalling that there is no infinite path. If one want to unfold this well-founded poset into an ordinal, then it should be the smallest ordinal o such that $o = 1 + \omega o$, *i.e.* $o = \omega^\omega$.

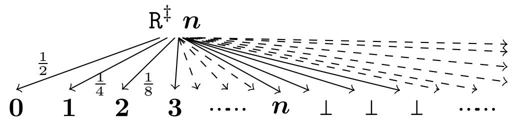
Remark that, due to the encoding of $\oplus$ and $\mathtt{X}$ into $\mathbb{T}^\mathtt{R}$, Theorem 22 is subsuming Theorem 12. Remark, moreover, that we did not have to go thought the definition of approximants. Nonetheless, those approximants exists and point out that morally $\mathbb{T}^\oplus$ should be approximating $\mathbb{T}^\mathtt{R}$ is some way or another.

5.2 The approximants: State-Bounded Random Integers

In this section, we show that $\mathbb{T}^\oplus$ approximates $\mathbb{T}^\mathtt{R}$: for any term $M \in \mathbb{T}^\mathtt{R}(\mathtt{NAT})$, there is a term $N \in \mathbb{T}^\oplus(\mathtt{NAT} \to \mathtt{NAT})$ that represents a sequence approximating M uniformly. We will here make strong use of the fact that M has type $\mathtt{NAT}$. This is a natural drawback when we understand that the encoding $(\cdot)^\dagger$ on which the result above is based is not direct, but goes through an other state passing style transformation. Nonetheless, everything can be lifted easily to the first order, achieving the parameterization of our theorem.

A naive idea would be to use $\mathbb{T}^\mathtt{X}$ and to stope the evaluation after a given reduction time as schematised in Figure 8. Despite the encoding to be a nightmare, this should be encodable in $\mathbb{T}^\oplus$. However, for the convergence time to be independant from the term and uniform, ther is vitually no hope. That is why we have swiched to $\mathbb{T}^\mathtt{R}$ which carries much nicer properties as seen in the previous chapter.

The basic idea behind the embedding $(\cdot)^\dagger$ is to mimic any instance of the $\mathtt{R}$ operator in the source term by some term $\mathbf{0} \oplus (\mathbf{1} \oplus (\cdots(\boldsymbol{n} \oplus \bot)\cdots))$, where n is *sufficiently large*, and $\bot$ is an arbitrary value of type $\mathtt{NAT}$. Of course, the semantics of this term is *not* the same as that of $\mathtt{R}$, due to the presence of $\bot$; however, n will be chosen sufficiently large for the difference to be negligible. Notice, moreover, that this term can be generalized into the following parametric form $\mathtt{R}^\ddagger := \lambda n.\mathtt{rec}\ \bot\ (\lambda x.\mathtt{S} \oplus (\lambda y.\mathbf{0}))\ n$.



Once $\mathtt{R}^\ddagger$ is available, a natural candidate for the encoding $(\cdot)^\dagger$ would be to consider something like $M^\ddagger := \lambda z.M[(\mathtt{R}^\ddagger\ z)/\mathtt{R}]$. In the underlying execution tree, $(M^\ddagger\ \boldsymbol{n})$ correctly simulates the first n branches of each $\mathtt{R}$ (which had infinite-arity), but truncates the rest with garbage terms $\bot$. As schematized in Figure 9, by increasing n, we can hope to obtain the M at the limit.

The question is whether the remaining untruncated tree has a "sufficient weight", i.e., that there is a minimal bound to the probability to stay in this untruncated tree. However, in general $(\cdot)^\ddagger$ fails on this point, not achieving to approximate M uniformly. In fact, this probability is basically $(1 - \frac{1}{2^n})^d$ where d is its depth. Since in general the depth of the untruncated tree can grow very rapidly on n in a powerful system like $\mathbb{T}$, there is no hope for this transformation to perform a uniform approximation.

It may well be possible to perform a complex monadic transformation in the syle of Section 4.2, that computes a function relating the size n of the truncature to the depth d of the execution tree. But there is a much easyer solution.

The solution we are using is to have the precision m of $\mathbf{0} \oplus (\mathbf{1} \oplus (\cdots(\boldsymbol{m} \oplus \bot)\cdots))$ to dynamically grow along the computation, as schematized in Figure 10. More specifically, in the approximants $M^\dagger \boldsymbol{n}$, the growing speed of m will increase with n: in the n-th approximant $M^\dagger \boldsymbol{n}$, $\mathtt{R}$ will be simulated as $\mathbf{0} \oplus (1 \oplus (\cdots(\boldsymbol{m} \oplus \bot)\cdots))$ *and*, somehow, m will be updated to $m+n$. Why does it work? Simply because even for an (hypothetical) infinite and complete execution tree of M, we would stay inside the n^{th} untruncated tree with probability $\prod_{k\geq 0}(1-\frac{1}{2^{k*n}})$ which is asymptotically above $(1-\frac{1}{n})$.

Implementing this scheme in $\mathbb{T}^\oplus$ requires a feature which is not available (but which can be encoded), namely ground-type references. We then prefer to show that the just described scheme can be realized in an intermediate language called $\mathbb{T}^{\bar{\mathtt{R}}}$, whose operational semantics is formulated not on *terms*, but rather on triples in the form (M,m,n), where M is the term currently being evaluated, m is the current approximation threshold value, and n is the value of which m is incremented whenever $\mathtt{R}$ is simulated. The operational semantics is standard, except for the following rule:

$$\frac{}{(\bar{\mathtt{R}},m,n) \to \left\{(k, m{+}n, n) \mapsto \frac{1}{2^{k+1}} \mid k < m\right\}}\ (r\text{-}\bar{\mathtt{R}})$$

Notice how this operator behaves similarly to $\mathtt{R}$ with the exception that it fails when drawing too big of a number (*i.e.*, bigger that the fist state m). Notice that the failure is represented by the fact that the resulting distribution does not necessarily sum to 1. The intermediate language $\mathbb{T}^{\bar{\mathtt{R}}}$ is able to approximate $\mathbb{T}^{\mathtt{R}}$ at every order (Theorem 31 below). Moreover, the two memory cells can be shown to be expressible in $\mathbb{T}^\oplus$, again by way of a continuation-passing transformation. Crucially, the initial value of n can be passed as an argument to the encoded term.

Definition 27. *For any $M \in \mathbb{T}^{\mathtt{R}}$ we denote $M^* := M[\bar{\mathtt{R}}/\mathtt{R}]$. We say that $(M,m,n) \in \mathbb{T}^{\bar{\mathtt{R}}}$ if $m,n \in \mathbb{N}$ and $M = N^*$ for some $N \in \mathbb{T}^{\mathtt{R}}$. Similarly, $\mathfrak{D}(\mathbb{T}^{\bar{\mathtt{R}}})$ is the set of probabilistic distributions over $\mathbb{T}^{\bar{\mathtt{R}}} \times \mathbb{N}^2$,* i.e.*, over the terms plus states. The reduction rules of the system T with state-bounded random integers are given by Figure 11.*

For any m and n, the behavior of M and (M^*,m,n) are similar, except that (M^*,m,n) will "fail" more often. In other words, all $(M^*,m,n)_{m,n\in\mathbb{N}}$ are good candidates in order to approach M from bellow:

Lemma 28. *For any $M \in \mathbb{T}^{\mathtt{R}}$ and any $m,n \in \mathbb{N}$, $[\![M]\!] \geq [\![M^*,m,n]\!]$, i.e., for every $V \in \mathbb{T}^{\mathtt{R}}_V$, we have*

$$[\![M]\!](V) \geq \sum_{m',n'} [\![M^*,m,n]\!](V^*,m',n').$$

Proof. By an easy induction, one can show that for any $\mathcal{M} \in \mathfrak{D}(\mathbb{T}^{\mathtt{R}})$ and $\mathcal{N} \in \mathfrak{D}(\mathbb{T}^{\bar{\mathtt{R}}})$ if $\mathcal{M} \geq \mathcal{N}$, $\mathcal{M} \to \mathcal{M}'$ and $\mathcal{N} \to \mathcal{N}'$, then $\mathcal{M}' \geq \mathcal{N}'$. This ordering is then preserved at the limit so that we get our result. □

In fact, the probability of "failure" of any $(M,m,n)_{m,n\in\mathbb{N}}$ can be upper-bounded explicitly. More precisely, we can find an infinite product underapproximating the success rate of (M,m,n) by reasoning inductively over $(M,m,n) \Rightarrow [\![(M,m,n)]\!]$, which is possible because of PAST.

Lemma 29. *For any $M \in \mathbb{T}^{\bar{\mathtt{R}}}$ and any $m,n \geq 1$*

$$Succ(M,m,n) \geq \prod_{k\geq 0}\left(1 - \frac{1}{2^{m+kn}}\right).$$

Proof. We denote

$$\#(m,n) := \prod_{k\geq 0}\left(1-\frac{1}{2^{m+kn}}\right) \qquad \text{and} \qquad \#\mathcal{M} := \int_{\mathcal{M}} \#(m,n)\ dMdmdn.$$

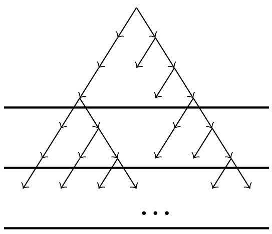


Figure 8:
Horizontal truncature of $\mathbb{T}^{\oplus}$

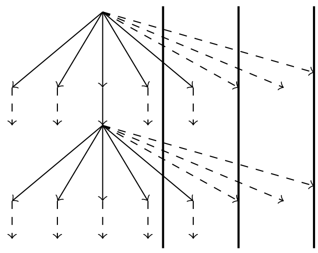


Figure 9:
Vertical truncatur of $\mathbb{T}^{\mathtt{R}}$

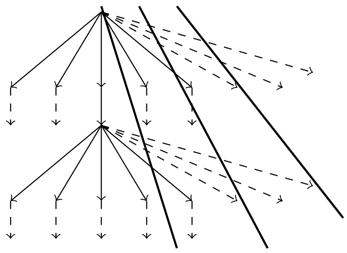


Figure 10:
Biaised Tuncatur of $\mathbb{T}^{\mathtt{R}}$

$$\frac{}{(\lambda x.M,m,n)\ V \to \Big\{(M[V/x],m,n)\Big\}}\ (s\text{-}\beta) \qquad \frac{(M,m,n)\to\mathcal{M}}{(M\ V,m,n)\to\mathcal{M}\ V}\ (s\text{-}c@L)$$

$$\frac{(N,m,n)\to\mathcal{N}}{(M\ N,m,n)\to M\ \mathcal{N}}\ (s\text{-}c@R) \qquad \frac{}{(\mathtt{rec}\ U\ V,m,n)\ \mathbf{0}\to\Big\{(U,m,n)\Big\}}\ (s\text{-}\mathtt{rec}\mathbf{0})$$

$$\frac{}{(\mathtt{rec}\ U\ V\ (\mathtt{S}\ \boldsymbol{k}),\ m,n)\to\Big\{(V\ \boldsymbol{k}\ (\mathtt{rec}\ U\ V\ \boldsymbol{k}),m,n)\Big\}}\ (s\text{-}\mathtt{recS})$$

$$\frac{}{(\pi_1\ \langle M,N\rangle,m,n)\to\Big\{(M,m,n)\Big\}}\ (s\text{-}\pi_1) \qquad \frac{}{(\pi_2\ \langle M,N\rangle,m,n)\to\Big\{(N,m,n)\Big\}}\ (s\text{-}\pi_2)$$

$$\frac{}{(\bar{\mathtt{R}},m,n)\to\left\{\begin{matrix}(k,m+n,n)\mapsto\frac{1}{2^{k+1}},\ \text{if } k<m \\ (N,m',n')\mapsto 0,\quad\text{otherwise}\end{matrix}\right\}}\ (s\text{-}\bar{\mathtt{R}})$$

$$\frac{\forall(M,m,n)\in|\mathcal{M}|,\ (M,m,n)\to^{?}\mathcal{N}_{(M,m,n)}}{\mathcal{M}\to\int_{\mathcal{M}}\mathcal{N}_{M,m,n}.dMdmdn}\ (s\text{-}\epsilon)$$

Figure 11: Operational semantics. Except for the M in (b-β) that can have x as a free variable, all terms and distributions are closed. Notice that $\to^{?}$ stands for the identity if the premise is a value and for $\to$ otherwise.

By induction on $\Rightarrow$, we can show that if $(M,m,n) \Rightarrow \mathcal{M}$ then $\#\mathcal{M} = \#(m,n)$ and that if $\mathcal{N} \Rightarrow \mathcal{M}$ then $\#\mathcal{M} = \#\mathcal{N}$.

- (R-$refl$) and (R-int) are immediate.
- If $(M,m,n) \to \mathcal{N} \Rightarrow \mathcal{M}$ then $\mathcal{N}$ is either of the form $\{(N,m,n)\}$ or $\{(N_i, m{+}n, n) \mapsto \frac{1}{2^{i+1}} \mid i < m\}$ for some N of $(N_i)_{i \leq m}$. In the first case it is clear that $\#\mathcal{N} = \#(m,n)$, but the equality hold also in the second:

$$\#\mathcal{N} \quad = \quad \sum_{i \leq m} \frac{1}{2^{i+1}} \#(m+n,n) \quad = \quad (1 - \frac{1}{2^m}) \#(m+n,n) \quad = \quad \#(m,n).$$

By IH, we conclude that $\#\mathcal{M} = \#\mathcal{N} = \#(m,n)$.

In particular we have

$$\begin{aligned} Succ(M,m,n) &:= \int_{[\![M,m,n]\!]} 1 \; dM dm dn && \\ &\geq \# [\![M]\!] && \text{since } \forall m,n, \;\; 1 \geq \#(m,n) \\ &= \#(m,n) && \text{since } (M,m,n) \Rightarrow [\![M,m,n]\!]. \end{aligned}$$

□

This gives us an analytic lower bound to the success rate of (M,m,n). However, it is not obvious that this infinite product is an interesting bound, it is not even clear that it can be different from 0. This is why we will further underapproximate this infinite product to get a simpler expression whenever $m = n$:

Lemma 30. *For any $M \in \mathbb{T}^{\bar{\mathtt{R}}}$ and any $n \geq 4$*

$$Succ(M,n,n) \quad \geq \quad 1 - \frac{1}{n}.$$

Proof. By Lemma 29 we have that $Succ(M,n,n) \geq \prod_{k \geq 1} \big(1 - \frac{1}{2^{k*n}}\big)$ which is above the product $\prod_{k \geq 1} \big(1 - \frac{1}{n^2 k^2}\big)$ whenever $n \geq 4$. This infinite product has been shown by Euler to be equal to $\frac{sin(\frac{\pi}{n})}{\frac{\pi}{n}}$. By an easy numerical analysis we then obtain that $\frac{sin(\frac{\pi}{n})}{\frac{\pi}{n}} \geq 1 - \frac{1}{n}$. □

This lemma can be restated by saying that the probability of "failure" of (M^*, n, n), *i.e.* the difference between $[\![M^*, n, n]\!]$ and $[\![M]\!]$, is bounded by $\frac{1}{n}$. With this we then get our first theorem, which is the uniform approximability of elements of $\mathbb{T}^{\mathtt{R}}$ by those of $\mathbb{T}^{\bar{\mathtt{R}}}$:

Theorem 31. *For any $M \in \mathbb{T}^{\mathtt{R}}$ and any $n \in \mathbb{N}$,*

$$\sum_V \Big| \; [\![M]\!](V) \; - \; \Sigma_{m',n'} [\![M^*, n, n]\!](V^*, m', n') \; \Big| \quad \leq \quad \frac{1}{n}.$$

Proof. By Lemma 28, for each V the difference is positive, thus we can remove the absolute value and distribute the sum. We conclude by using the fact that $Succ M = 1$ and $Succ(M^*, n, n) \geq 1 - \frac{1}{n}$. □

The second theorem, *i.e.*, the uniform approximability of ground elements of $\mathbb{T}^{\mathtt{R}}$ by those of $\mathbb{T}^{\oplus}$, follows immediately:

Theorem 32. *Distributions in $\mathbb{T}^{\mathtt{R}}(\mathtt{NAT})$ can be approximated by $\mathbb{T}^{\oplus}$-distributions (which are finitely $\mathbb{T}$-representable), i.e., for any $M \in \mathbb{T}^{\mathtt{R}}(\mathtt{NAT})$, there is $M^\dagger \in \mathbb{T}^{\oplus}(\mathbb{N})$ such that:*

$$\forall n, \quad \sum_k \Big| \; [\![M]\!](\boldsymbol{k}) \; - \; [\![M^\dagger \; \boldsymbol{n}]\!](\boldsymbol{k}) \; \Big| \quad \leq \quad \frac{1}{n}.$$

Moreover:

- *the encoding is parameterizable in the sens that for all $M \in \mathbb{T}^{\mathtt{R}}(\mathtt{NAT} \to \mathtt{NAT})$, there is $M^\dagger \in \mathbb{T}^{\oplus}(\mathtt{NAT} \to \mathtt{NAT})$ such that $(M \; \boldsymbol{n})^\dagger = M^\dagger \; \boldsymbol{n}$ for all $n \in \mathbb{N}$,*

- *the encoding is such that* $[\![M]\!](\boldsymbol{k}) \leq [\![M^{\dagger}\ \boldsymbol{n}]\!](\boldsymbol{k})$ *possible only for* $k = 0$.

Proof. It is clear that in an extension of $\mathbb{T}^{\oplus}$ with two global memory cells m, n and with exceptions, the $\bar{\mathtt{R}}$ operator can be encoded by

$$\bar{\mathtt{R}} \quad := \quad \mathtt{rec}\langle \lambda u.\bot,\ \lambda xyu.\,\mathbf{0} \oplus \mathtt{S}(y\ u),\ m := !m + !n \rangle\ \mathbf{0},$$

where $\bot$ is raising an error/exception and $m := !m + !n$ is returning the value of m before changing the memory cell to $m + n$. Remark that the only objective of the dummy abstraction over u and of the dummy aplication to $\mathbf{0}$, is to block the evaluation of the $\bot$. We can conclude by referring to the usual state passing style encoding of exceptions and state-monads into $\mathbb{T}$ (and thus into $\mathbb{T}^{\oplus}$).

In fact, we do not have any requirement over the $\bot$ since we are just majoring the divergence toward a required result. This means that we can replace $\bot$ by any value $\bot_A$ of the correct type A (which is possible since every type is inhabited). Then we do not need to implement the exception monad, but only the state monad which we can present easily here:

$$(\!(A)\!) := \mathtt{NAT}^3 \to ((\!(A)\!)_{\boldsymbol{V}} \times \mathtt{NAT}^3) \qquad (\!(\mathtt{NAT})\!)_{\boldsymbol{V}} := \mathtt{NAT}$$
$$(\!(A \to B)\!)_{\boldsymbol{V}} := (\!(A)\!)_{\boldsymbol{V}} \to (\!(B)\!) \qquad (\!(A \times B)\!)_{\boldsymbol{V}} := (\!(A)\!) \times (\!(B)\!)$$

Where the first state is monitoring the presence of an error along the reduction, the second represents the state m and the third represents the state n.

The encoding $(\!(-)\!)_{\boldsymbol{V}}$ of extended values is the same as for the encoding of Section 4.2:

$$(\!(\mathtt{S})\!)_{\boldsymbol{V}} := \lambda y.(\!(\mathtt{S}\ y)\!)_{\boldsymbol{V}} \qquad (\!(\mathtt{S}\ V)\!)_{\boldsymbol{V}} := \mathtt{S}\ (\!(V)\!)_{\boldsymbol{V}} \qquad (\!(\mathbf{0})\!)_{\boldsymbol{V}} := \mathbf{0}$$
$$(\!(\langle M, N\rangle)\!)_{\boldsymbol{V}} := \langle (\!(M)\!)_{\boldsymbol{V}}, (\!(N)\!)_{\boldsymbol{V}} \rangle \quad (\!(\lambda x.M)\!)_{\boldsymbol{V}} := \lambda x.(\!(M)\!) \qquad (\!(x)\!)_{\boldsymbol{V}} := x$$
$$(\!(\pi_i)\!)_{\boldsymbol{V}} := \lambda x.\mathtt{ret}(\pi_i x) \qquad (\!(\mathtt{rec})\!)_{\boldsymbol{V}} := \lambda\langle u, v, w\rangle.\mathtt{rec}\ \langle \mathtt{ret}\ u,\ \lambda xy.(vx) =\!\!\ll y,\ w\rangle$$

Where $\mathtt{ret}$ and $=\!\!\ll$ are the return and the bind of the considered monad:[5]

$$\mathtt{ret} : (\!(A)\!)_{\boldsymbol{V}} \to (\!(A)\!) \qquad \mathtt{ret} := \lambda xs.\langle x, s\rangle$$
$$\_ =\!\!\ll \_ : (\!(A \to B)\!) \times (\!(A)\!) \to (\!(B)\!) \qquad M =\!\!\ll N := \lambda s_1.(\lambda\langle x, s_2\rangle.(\lambda\langle y, s_3\rangle.xys_3)\ (Ns_2))\ (Ms_1)$$

What $M =\!\!\ll N$ does is looking at the current state s_1, evaluating M under the state s_1 which results to $\langle x, s_2\rangle$, then evaluating N under the state s_2 which results to $\langle y, s_3\rangle$, and, finally, evaluating $(x\ y) : (\!(B)\!)$ under the state s_3.

The encoding $(\!(\cdot)\!)$ is given by the return and the bind operations as well as the encoding of effectfull operations:

$$(\!(V)\!) := \mathtt{ret}\ (\!(V)\!)_{\boldsymbol{V}} \qquad (\!(M\ N)\!) := (\!(M)\!) =\!\!\ll (\!(N)\!)$$
$$(\!(M \oplus N)\!) := \lambda s.\ ((\!(M)\!)\ s)\ \oplus\ ((\!(N)\!)\ s) \qquad (\!(m := !m + !n)\!) := \lambda\langle e, m, n\rangle.\ \langle\ m\ ,\ \langle e, m + n, n\rangle\ \rangle$$
$$(\!(\bot)\!) := \lambda\langle e, m, n\rangle.\langle *, \langle \mathbf{1}, \langle m, n\rangle\rangle\rangle.$$

where $*$ is any term of correct type.

In the end, we set $M^{\dagger} := \lambda x.(\lambda\langle y, e, m, n\rangle.\mathtt{rec}\ y\ (\lambda uv.\mathbf{0})\ e)\ ((\!(M)\!)\ \langle x, x\rangle)$ for $M \in \mathbb{T}^{\mathtt{R}}(\mathtt{NAT})$. The parameterizability is obtained by using the equality $(\!(\boldsymbol{n})\!)_{\boldsymbol{V}} = \boldsymbol{n}$.

□

Corollary 33. *Distributions in* $\mathbb{T}^{\mathtt{R}}$ *are functionally parameterically representable by* $\mathbb{T}$*-definable functions,* i.e. *for any* $M : \mathtt{NAT} \to \mathtt{NAT}$ *in* $\mathbb{T}^{\mathtt{R}}$ *there is* $F : \mathtt{NAT} \to \mathtt{NAT} \to \mathtt{NAT} \to \mathtt{BIN}$ *and* $Q : \mathtt{NAT} \to \mathtt{NAT} \to \mathtt{NAT}$ *in* $\mathbb{T}$ *such that for all* m *and* n*:*

$$\sum_{k \in \mathbb{N}} \Big| [\![M\ \boldsymbol{m}]\!](\boldsymbol{k}) - \mathbb{NF}(F\ \boldsymbol{m}\ \boldsymbol{n}\ \boldsymbol{k}) \Big| \ \leq\ \frac{1}{n} \qquad \forall k \geq \mathbb{NF}(Q\ \boldsymbol{m}\ \boldsymbol{n}),\ \mathbb{NF}(F\ \boldsymbol{m}\ \boldsymbol{n}\ \boldsymbol{k}) = \mathbf{0}.$$

[5] Technically, this bind is not the usual bind, but a lifted version.

6 Subrecursion

We recall that a function $f : \mathbb{N} \to \mathbb{N}$ is $\mathbb{T}$-definable if there is a program $\vdash M : \mathbb{N} \to \mathbb{N}$ in $\mathbb{T}$ such that $(M\ \boldsymbol{n}) \to^* f(n)$ for all n. We denote $\boldsymbol{DT}$ the set of $\mathbb{T}$-definissable functions.

If one wishes to define $\mathbb{T}^{\oplus}$-definable or $\mathbb{T}^{\mathtt{R}}$-definable functions as a set of ordinary set-theoretic functions (say from $\mathbb{N}$ to $\mathbb{N}$), it is necessary to collapse the random output into a deterministic one. As already acknowledged by the complexity community, there are at least two reasonable ways to do so: by using a either Monte Carlo or Las Vegas observations.

6.1 Monte Carlo

We call *Monte Carlo* observation on $\mathbb{T}^{\oplus}$ (rep. $\mathbb{T}^{\mathtt{R}}$) the class $\boldsymbol{BPT}^{\oplus}$ (rep. $\boldsymbol{BPT}^{R}$) of functions f definable by a program $\vdash M : \mathtt{NAT} \to \mathtt{NAT}$ (in $\mathbb{T}^{\oplus}$ and $\mathbb{T}^{\mathtt{R}}$ resp.) in the sens that $(M\ \boldsymbol{n})$ evaluate into $f(n)$ with probability $p \geq \frac{2}{3}$.

$$f \in \boldsymbol{BPT}^{\oplus} \qquad \text{if} \qquad \exists M \in \mathbb{T}^{\oplus}(\mathtt{NAT} \to \mathtt{NAT}),\quad \forall n \in \mathbb{N},\quad [\![M\boldsymbol{n}]\!]\,(\boldsymbol{f(n)}) \geq \frac{2}{3}$$

In fact, the bound $\frac{2}{3}$ is arbitrary and we could have equivalently use any bound strictly above $\frac{1}{2}$. That is why it is natural to also consider the limit. We call *probabilistic* observation on $\mathbb{T}^{\oplus}$ (rep. $\mathbb{T}^{\mathtt{R}}$) the class $\boldsymbol{PT}^{\oplus}$ (rep. $\boldsymbol{PT}^{R}$) of functions f definable by a program $M : \mathtt{NAT} \to \mathtt{NAT}$ of $\mathbb{T}^{\oplus}$ (rep. $\mathbb{T}^{\mathtt{R}}$) in the sens that $(M\ \boldsymbol{n})$ evaluate into $f(n)$ with probability $p > \frac{1}{2}$:

$$f \in \boldsymbol{PT}^{\oplus} \qquad \text{if} \qquad \exists M \in \mathbb{T}^{\oplus}(\mathtt{NAT} \to \mathtt{NAT}),\quad \forall n \in \mathbb{N},\quad [\![M\boldsymbol{n}]\!]\,(\boldsymbol{f(n)}) > \frac{1}{2}$$

In its all generality, the pertinence of this class is dubitative. Indeed, we can see it as the class where $(M\ \boldsymbol{n})$ evaluates into $f(n)$ with probability $p \geq \frac{1}{2} + h(n)$ for an arbitrary non null n, in particular for h non commutable. In this case, the result may not make much sense in practice.

Due to the functional aspect of the considered objects,[6] we can nonetheless consider subclasses of $\boldsymbol{PT}^{\oplus}$ (rep. $\boldsymbol{PT}^{R}$) for a reasonable dynamic bound h.

We call *dynamic Monte Carlo* observation on $\mathbb{T}^{\oplus}$ (rep. $\mathbb{T}^{\mathtt{R}}$) the class $\boldsymbol{BPT}^{\oplus}_{\geq\mathbb{T}}$ (rep. $\boldsymbol{BPT}^{R}_{\geq\mathbb{T}}$) of functions f definable by a program $\vdash M : \mathtt{NAT} \to \mathtt{NAT}$ (in $\mathbb{T}^{\oplus}$ and $\mathbb{T}^{\mathtt{R}}$ resp.) in the sens that $(M\ \boldsymbol{n})$ evaluate into $f(n)$ with probability $p \geq \frac{1}{2} + \frac{1}{h(n)}$ for a $\mathbb{T}$-definable function h.

$$f \in \boldsymbol{PT}^{\oplus} \qquad \text{if} \qquad \exists h \in \boldsymbol{DT},\ \ \exists M \in \mathbb{T}^{\oplus}(\mathtt{NAT} \to \mathtt{NAT}),\quad \forall n \in \mathbb{N},\quad [\![M\boldsymbol{n}]\!]\,(\boldsymbol{f(n)}) > \frac{1}{2} + \frac{1}{h(n)}$$

Theorem 34.

$$\boldsymbol{PT}^{\oplus} \ = \ \boldsymbol{DT},$$

in particular we also have $\boldsymbol{BPT}^{\oplus} = \boldsymbol{BPT}^{\oplus}_{\geq\mathbb{T}} = \boldsymbol{DT}$.

Proof. Let $f \in \boldsymbol{PT}^{\oplus}$.

There is $M \in \mathbb{T}^{\oplus}(\mathtt{NAT} \to \mathtt{NAT})$ such that $[\![M\ \boldsymbol{m}]\!]\,(f(m)) > \frac{1}{2}$.

By Theorem 25, there is $F \in \mathbb{T}(\mathtt{NAT} \to \mathtt{NAT} \to \mathtt{BIN})$ and $G \in \mathbb{T}(\mathtt{NAT} \to \mathtt{NAT})$ such that

$$\forall k \leq \mathtt{NF}(G\,n), \qquad \mathtt{NF}(F\,\boldsymbol{n}\,\boldsymbol{k}) > \frac{1}{2} \quad\Leftrightarrow\quad k = f(n)\ .$$

In this case can we set $M' \in \mathbb{T}(\mathtt{NAT} \to \mathtt{NAT})$ such that $\mathtt{NF}(M'\ \boldsymbol{n}) = f(n)$, by:

$$M' := \lambda n.\mathtt{rec}\langle Fn\mathbf{0},\ \lambda ky.\mathtt{ite}\langle \mathtt{sup}_{\frac{1}{2}}(F\,n\,k),\ k,\ y\rangle,\ G\,n\rangle$$

where $\mathtt{sup}_{\frac{1}{2}}$ is testing whether the input is above $\frac{1}{2}$:

$$\begin{array}{ll} \mathtt{sup}_{\frac{1}{2}} : \mathtt{NAT} \times \mathtt{NAT} \to \mathtt{NAT} & \qquad \mathtt{sup}_{\frac{1}{2}} := \lambda\langle m,n\rangle.(m+m) > 2^n \\ \_>\_ : \mathtt{NAT} \times \mathtt{NAT} \to \mathtt{NAT} & \qquad M > N := \mathtt{rec}\langle \lambda u.\mathbf{0},\ \lambda xyu.\mathtt{rec}\langle \mathbf{1}, \lambda a\_.ya, u\rangle,\ M\rangle\ N \end{array}$$

□

[6] in contrast to what happened for the polynomial classes.

Theorem 35.

$$\boldsymbol{BPT}^{R}_{\geq\mathbb{T}} \;=\; \boldsymbol{DT},$$

in particular we also have $\boldsymbol{BPT}^{R} = \boldsymbol{DT}$.

Proof. Let $f \in \boldsymbol{BPT}^{R}_{\geq\mathbb{T}}$.
There is $M \in \mathbb{T}^{\mathtt{R}}(\mathtt{NAT} \to \mathtt{NAT})$ and $H \in \mathbb{T}(\mathtt{NAT} \to \mathtt{NAT})$ such that

$$[\![M\ \boldsymbol{m}]\!]\,(f(m)) \geq \frac{1}{2} + \frac{1}{\mathbb{NF}(H\ \boldsymbol{m})}.$$

By Theorem 33, there exists $F : \mathtt{NAT}{\to}\mathtt{NAT}{\to}\mathtt{NAT}{\to}\mathtt{BIN}$ and $Q : \mathtt{NAT}{\to}\mathtt{NAT}{\to}\mathtt{NAT}$ in $\mathbb{T}$ such that:

$$\sum_{k\in\mathbb{N}} \Big|\, [\![M\ \boldsymbol{m}]\!]\,(\boldsymbol{k}) - \mathbb{NF}(F\ \boldsymbol{m}\ \boldsymbol{n}\ \boldsymbol{k}) \,\Big| \;\leq\; \frac{1}{n} \qquad \forall k \geq \mathbb{NF}(Q\ \boldsymbol{m}\ \boldsymbol{n}),\ \mathbb{NF}(F\ \boldsymbol{m}\ \boldsymbol{n}\ \boldsymbol{k}) = \boldsymbol{0}.$$

In particular, $f(m)$ is the only $k \leq \mathbb{NF}(Q\ \boldsymbol{m}\ (H\ \boldsymbol{m}))$ such that:

$$\mathbb{NF}(F\ \boldsymbol{m}\ (H\ \boldsymbol{m})\ \boldsymbol{k}) > \frac{1}{2}$$

In this case can we set M' such that $\mathbb{NF}(M'\ \boldsymbol{n}) = f(n)$ by:

$$M' := \lambda m.\mathtt{rec}\langle F\,m\,(H\,m)\,\boldsymbol{0},\ \lambda ky.\mathtt{ite}\langle \mathtt{sup}_{\frac{1}{2}}(F\,m\,(H\,m)\,k), k, y\rangle,\ G\,m\,(H\,m)\rangle$$

□

6.2 Las Vegas

We call *Las Vegas* observation on $\mathbb{T}^{\oplus}$ (rep. $\mathbb{T}^{\mathtt{R}}$) the class[7] $\boldsymbol{LVT}^{\oplus}$ (rep. $\boldsymbol{LVT}^{R}$) of functions f definable by a program $M : \mathtt{NAT} \to \mathtt{NAT}$ (in $\mathbb{T}^{\oplus}$ and $\mathbb{T}^{\mathtt{R}}$ resp.) in the sens that $(M\ \boldsymbol{n})$ evaluate either to $\boldsymbol{0}$ representing a failure or to $(\mathtt{S}\boldsymbol{f(n)})$, the later happening with probability at least $\frac{1}{3}$:

$$f \in \boldsymbol{LVT}^{\oplus} \qquad \text{if} \qquad \exists M \in \mathbb{T}^{\oplus}(\mathtt{NAT} \to \mathtt{NAT}),\ [\![M\boldsymbol{n}]\!] = \begin{Bmatrix} \mathtt{S}\boldsymbol{f(n)} & \mapsto & p \\ \boldsymbol{0} & \mapsto & (1-p) \end{Bmatrix} \text{ with } p \geq \frac{1}{3}$$

Similarly, the bound $\frac{1}{3}$ is arbitrary and we can consider the limit class (which corresponds to a non-deterministic interpretation on $\oplus$) and the bounded restrictions.

We call *non-deterministic* observation on $\mathbb{T}^{\oplus}$ (rep. $\mathbb{T}^{\mathtt{R}}$) the class $\boldsymbol{NT}^{\oplus}$ (rep. $\boldsymbol{NT}^{R}$) of functions f definable by a program $\vdash_{\mathbb{T}^{\oplus},\mathtt{R},\mathtt{X}} M : \mathtt{NAT} \to \mathtt{NAT}$ in the sens that $(M\ \boldsymbol{n})$ evaluate either to $\boldsymbol{0}$ representing a failure or to $\mathtt{S}f(n)$, the later eventually happening.

$$f \in \boldsymbol{LVT}^{\oplus} \qquad \text{if} \qquad \exists M \in \mathbb{T}^{\oplus}(\mathtt{NAT} \to \mathtt{NAT}),\ [\![M\boldsymbol{n}]\!] = \begin{Bmatrix} \mathtt{S}\boldsymbol{f(n)} & \mapsto & p \\ \boldsymbol{0} & \mapsto & (1-p) \end{Bmatrix} \text{ with } p > 0$$

We call *dynamic Las Vegas* observation on $\mathbb{T}^{\oplus}$ (rep. $\mathbb{T}^{\mathtt{R}}$) the class $\boldsymbol{LVT}^{\oplus}_{\geq\mathbb{T}}$ (rep. $\boldsymbol{LVT}^{R}_{\geq\mathbb{T}}$) of functions f definable by a program $\vdash_{\mathbb{T}^{\oplus},\mathtt{R},\mathtt{X}} M : \mathtt{NAT} \to \mathtt{NAT}$ (in $\mathbb{T}^{\oplus}$ and $\mathbb{T}^{\mathtt{R}}$ resp.)in the sens that $(M\ \boldsymbol{n})$ evaluate either to $\boldsymbol{0}$ representing a failure or to $\mathtt{S}f(n)$, the later happening with probability $p \geq \frac{1}{h(n)}$ for some $\mathbb{T}$-definable function h.

$$f \in \boldsymbol{LVT}^{\oplus} \qquad \text{if} \qquad \exists h \in \boldsymbol{DT}, \exists M \in \mathbb{T}^{\oplus}(\mathtt{NAT} \to \mathtt{NAT}),\ [\![M\boldsymbol{n}]\!] = \begin{Bmatrix} \mathtt{S}\boldsymbol{f(n)} & \mapsto & p \\ \boldsymbol{0} & \mapsto & (1-p) \end{Bmatrix} \text{ with } p \geq \frac{1}{h(n)}$$

Theorem 36.

$$\boldsymbol{NT}^{\oplus} \;=\; \boldsymbol{DT},$$

in particular we also have $\boldsymbol{LVT}^{\oplus} = \boldsymbol{LVT}^{\oplus}_{\geq\mathbb{T}} = \boldsymbol{DT}$.

[7]The polynomial equivalent to this class is the class $\boldsymbol{ZPP}$, which name comes from an equivalent representation of zero-error probabilistic programs in polynomial mid-time. Notice that the equivalence does not hold her.

Proof. Let $f \in \boldsymbol{NT}^{\oplus}$.

There is $M \in \mathbb{T}^{\oplus}(\mathtt{NAT} \to \mathtt{NAT})$ such that $f(m)$ is the only $k \in \mathbb{N}$ such that $[\![M\ \boldsymbol{m}]\!]\,(\mathtt{S}\boldsymbol{k}) > 0$.

By Theorem 25, there is $F \in \mathbb{T}(\mathtt{NAT} \to \mathtt{NAT} \to \mathtt{BIN})$ and $G \in \mathbb{T}(\mathtt{NAT} \to \mathtt{NAT})$ such that

$$\forall k \leq \mathbb{NF}(G\,n), \qquad \mathbb{NF}(F\,\boldsymbol{n}\,(\mathtt{S}\boldsymbol{k})) > 0 \quad \Leftrightarrow \quad k = f(n)\ .$$

In this case can we set $M' \in \mathbb{T}(\mathtt{NAT} \to \mathtt{NAT})$ such that $\mathbb{NF}(M'\ \boldsymbol{n}) = f(n)$, by:

$$M' := \lambda n.\mathtt{rec}\langle Fn\mathbf{0},\ \lambda ky.\mathtt{ite}\langle \mathtt{sup}_0(F\,n\,(\mathtt{S}k)),\ k,\ y\rangle,\ G\,n\rangle$$

where $\mathtt{sup}_0$ is testing whether the input is above $\frac{1}{2}$:

$$\mathtt{sup}_0 : \mathtt{NAT} \times \mathtt{NAT} \to \mathtt{NAT} \qquad\qquad \mathtt{sup}_0 := \lambda\langle m, n\rangle.m > \mathbf{0}$$

□

Theorem 37.

$$\boldsymbol{LVT}^{R}_{\geq\mathbb{T}} \ = \ \boldsymbol{DT},$$

in particular we also have $\boldsymbol{LVT}^{R} = \boldsymbol{DT}$.

Proof. Let $f \in \boldsymbol{LVT}^{R}_{\geq\mathbb{T}}$.
There is $M \in \mathbb{T}^{\mathrm{R}}(\mathtt{NAT} \to \mathtt{NAT})$ and $H \in \mathbb{T}(\mathtt{NAT} \to \mathtt{NAT})$ such that

$$[\![M\ \boldsymbol{m}]\!]\,(\mathtt{S}\ \boldsymbol{k}) > \mathbf{0} \quad \Leftrightarrow \quad [\![M\ \boldsymbol{m}]\!]\,(\mathtt{S}\ \boldsymbol{k}) > \frac{1}{\mathbb{NF}(H\ \boldsymbol{m})} \quad \Leftrightarrow \quad k = f(m)$$

By Theorem 33, there exists $F : \mathtt{NAT}{\to}\mathtt{NAT}{\to}\mathtt{NAT}{\to}\mathtt{BIN}$ and $Q : \mathtt{NAT}{\to}\mathtt{NAT}{\to}\mathtt{NAT}$ in $\mathbb{T}$ such that:

$$\sum_{k\in\mathbb{N}} \Big|\ [\![M\ \boldsymbol{m}]\!]\,(\boldsymbol{k}) - \mathbb{NF}(F\ \boldsymbol{m}\ \boldsymbol{n}\ \boldsymbol{k})\ \Big| \ \leq\ \frac{1}{n} \qquad\qquad \forall k \geq \mathbb{NF}(Q\ \boldsymbol{m}\ \boldsymbol{n}),\ \mathbb{NF}(F\ \boldsymbol{m}\ \boldsymbol{n}\ \boldsymbol{k}) = \mathbf{0}.$$

In particular, for $n = 2 * \mathbb{NF}(H\ \boldsymbol{m})$, we get that $f(m)$ is the only $k \leq \mathbb{NF}(Q\ \boldsymbol{m}\ (\mathbf{2}{*}(H\ \boldsymbol{m})))$ such that:

$$\mathbb{NF}(F\ \boldsymbol{m}\ (2 * (H\ \boldsymbol{m}))\ (\mathtt{S}\ \boldsymbol{k})) > \frac{1}{2 * \mathbb{NF}(H\ \boldsymbol{m})}$$

In this case can we set M' such that $\mathbb{NF}(M'\ \boldsymbol{n}) = f(n)$ by:

$$\begin{aligned} M' := \lambda m.\mathtt{rec}\langle & F\,m\,(\mathbf{2}{*}(H\,m))\,\mathbf{1}, \\ & \lambda ky.\mathtt{ite}\langle\ \mathtt{sup}_{\frac{1}{2}}((H\ \boldsymbol{m}) *_b (F\,m\,(\mathbf{2}{*}(H\,m))\,(\mathtt{S}k)))\ ,\ k,\ y\rangle, \\ & G\,m\,(\mathbf{2}{*}(H\,m)) \\ & \rangle \end{aligned}$$

Where

$$\begin{aligned} \_ * \_ : \mathtt{NAT} \times \mathtt{NAT} \to \mathtt{NAT} \qquad\qquad & M * N := \mathtt{rec}\langle \mathbf{0}, \lambda\_y.M{+}y, N\rangle \\ \_ *_b \_ : \mathtt{NAT} \times \mathtt{BIN} \to \mathtt{BIN} \qquad\qquad & (M *_b) := \lambda\langle m, n\rangle.\langle (M * m, n)\rangle \end{aligned}$$

□

6.3 Probabilistic and non-deterministic observations

In the two studies above, we have not considered $\boldsymbol{PT}^{\mathrm{R}}$ and $\boldsymbol{NT}^{\mathrm{R}}$. As we previously mentioned, the practical pertinence of these classes is kind of dubitative, in the sens as the result will be obtained after an unbounded number of tries and the proof that the algorithm is correct is a given as an oracle. In this section, we formalize this intuition and we show that both of them contain recursive bot not provably-recursive functions.

More precisely, we show that $\boldsymbol{NT}^{\mathtt{R}}$, the non-deterministic observation over $\mathbb{T}^{\mathtt{R}}$, corresponds exactly to recursive functions; While $\boldsymbol{PT}^{\mathtt{R}}$, the probabilistic observation over $\mathbb{T}^{\mathtt{R}}$, is a bit more complex and corresponds to a recursive choice over two $\mathbb{T}$-definable possible results.

Remark: Contrary to the polynomial case where $\boldsymbol{NP} \subset \boldsymbol{PP}$, we have $\boldsymbol{PT} \subset \boldsymbol{NT}$. In fact for decisionel (when the codomain is the Booleans), both collapse to recursive decisions, this is on the functional structure that they differ (while this functional structure is not considered for the polynomial classes).

Theorem 38.

$$\boldsymbol{NT}^{\mathtt{R}} = \boldsymbol{Rec}^{\mathbb{N}}$$

Proof.

- $\boldsymbol{NT}^{\mathtt{R}} \subseteq \boldsymbol{Rec}^{\mathbb{N}}$:
Let $f \in \boldsymbol{NT}^{\mathtt{R}}\boldsymbol{NT}^{\mathtt{X}}$; there is $M \in \mathbb{T}^{\mathtt{X}}(\mathtt{NAT} \to \mathtt{NAT})$ such that $f(m)$ is the only $k \in \mathbb{N}$ such that $[\![M\ \boldsymbol{m}]\!]\,(\mathtt{S}\boldsymbol{k}) > 0$. This means that there is a finite execution of M converging to $(\mathtt{S}\boldsymbol{k})$. Thus we only have to perform a simple Breadth-first search in the binary execution tree of M.

- $\boldsymbol{NT}^{\mathtt{R}} \supseteq \boldsymbol{Rec}^{\mathbb{N}}$:
Let $f \in \boldsymbol{Rec}^{\mathbb{N}}$; then f is computed by a program $M : \mathbb{N} \to \mathbb{N}$ that can use the operators of system $\mathbb{T}$ plus an unguarded recursion $\mathtt{Y} : (A \to A) \to A$; since the execution of $M\boldsymbol{n}$ is finite, there exists an error-free execution of $M[\mathtt{Y} := \lambda x.\mathtt{X}\langle x, \bot\rangle]\,\boldsymbol{n} \in \mathbb{T}^{\mathtt{X}}_{\bot}$ that gives the same result; using a an encoding of the error monad, we can easily get a term $N \in \mathbb{T}^{\mathtt{X}}(\mathbb{N} \to \mathbb{N})$ such that $f(m)$ is the only $k \in \mathbb{N}$ such that $[\![N\ \boldsymbol{m}]\!]\,(\mathtt{S}\boldsymbol{k}) > 0$. We conclude by $\boldsymbol{NT}^{\mathtt{R}} = \boldsymbol{NT}^{\mathtt{X}}$

□

Theorem 39.

$$\boldsymbol{PT}^{\mathtt{R}} = \boldsymbol{DT} \circ \boldsymbol{Rec}^{\{1,2\}}$$

in the sens that $f \in \boldsymbol{PT}^{\mathtt{R}}$ iff there is two functions $g_1, g_2 \in \boldsymbol{DT}$ and a recursive function $h : \mathbb{N} \to \{1,2\}$ such that $f(n) = g_{h(n)}(n)$.

Proof.

- $\boldsymbol{PT}^{\mathtt{R}} \subseteq \boldsymbol{DT} \circ \boldsymbol{Rec}^{\{1,2\}}$:
Let $f \in \boldsymbol{PT}^{\mathtt{R}}$; there is $M \in \mathbb{T}^{\mathtt{R}}(\mathtt{NAT} \to \mathtt{NAT})$ such that $[\![M\ \boldsymbol{m}]\!]\,(f(m)) > \frac{1}{2}$. By Theorem 33, there exists $F : \mathtt{NAT}{\to}\mathtt{NAT}{\to}\mathtt{NAT}{\to}\mathtt{BIN}$ and $Q : \mathtt{NAT}{\to}\mathtt{NAT}{\to}\mathtt{NAT}$ in $\mathbb{T}$ such that:

$$\sum_{k \in \mathbb{N}} \left| [\![M\ \boldsymbol{m}]\!]\,(\boldsymbol{k}) - \mathtt{NF}(F\ \boldsymbol{m}\ \boldsymbol{n}\ \boldsymbol{k}) \right| \leq \frac{1}{n} \qquad \forall k \geq \mathtt{NF}(Q\ \boldsymbol{m}\ \boldsymbol{n}),\ \mathtt{NF}(F\ \boldsymbol{m}\ \boldsymbol{n}\ \boldsymbol{k}) = \boldsymbol{0}.$$

Then for $n = 8$, we get that $\mathtt{NF}(F\ \boldsymbol{m}\ \boldsymbol{n}\ \boldsymbol{k}) > \frac{3}{8}$ for $k = f(m)$ and for at most one other value (since the total has to be bellow $\frac{9}{8}$), both bellow $\mathtt{NF}(Q\ \boldsymbol{m}\ \boldsymbol{n})$. We can thus construct two terms $N_1, N_2 : \mathbb{N} \to \mathbb{N}$ in $\mathbb{T}$ such that

  - $N_1\boldsymbol{m}$ gives the smaller of those k such that $\mathtt{NF}(F\ \boldsymbol{m}\ \boldsymbol{n}\ \boldsymbol{k}) > \frac{3}{8}$
  - and $N_2\boldsymbol{m}$ the bigger.

A recursive procedure can then easily choose which one of the failure is the correct one.

- $\boldsymbol{PT}^{\mathtt{R}} \supseteq \boldsymbol{DT} \circ \boldsymbol{Rec}^{\{1,2\}}$:
Let $g_1, g_2 \in \boldsymbol{DT}$ and a recursive function $h : \mathbb{N} \to \{1,2\}$. Trivially, we can write $G : \mathtt{NAT} \to \mathtt{NAT} \to \mathtt{NAT}$ in $\mathbb{T} \subseteq \mathbb{T}^{\mathtt{R}}$ such that $\mathtt{NF}(G\boldsymbol{1}\boldsymbol{n}) = g_1(n)$ and $\mathtt{NF}(G\boldsymbol{2}\boldsymbol{n}) = g_2(n)$. As we have seen in Theorem 38, $h \in \boldsymbol{NT}^{\mathtt{R}}$ and thus there is $M \in \mathbb{T}^{\mathtt{R}}(\mathtt{NAT} \to \mathtt{NAT})$ such that $f(m)$ is the only $k \in \mathbb{N}$ such that $[\![M\ \boldsymbol{m}]\!]\,(\mathtt{S}\boldsymbol{k}) > 0$. We thus set:

$$N := \lambda n.\mathtt{ite}\langle Mn,\ G\,(Mn)\,n,\ (G\,\boldsymbol{1}\,n) {\oplus} (G\,\boldsymbol{2}\,n)\rangle$$

□

7 Conclusions

This paper is concerned with the impact of adding various forms of probabilistic choice operators to a higher-order subrecursive calculus in the style of Gödel's $\mathbb{T}$. One may wonder why we have put ourselves in such a context, and whether the results in this paper can be adapted to other scenarios.

The three probabilistic choice operators we analyze in this paper are equivalent if employed in the context of untyped or Turing-powerful λ-calculi [5]. As an example, $\mathtt{X}$ can be easily expressed by way of $\oplus$, thanks to fixpoints. Moreover, there is no hope to get termination in any of those settings.

On the other hand, we claim that all we have said in this paper could have been spelled out in a probabilistic variation of Kleene's algebra of primitive recursive functions, e.g. [6]. Going higher-order makes our results, and in particular the termination results from Sections 3 and 4, significantly stronger. This is one of the reasons why we have proceeded this way. Classically, subrecursion refers to the study of relatively small classes of computable functions lying strictly below the partially recursive ones, and typically consisting of *total* functions. In this paper, we have initiated a study of the corresponding notion of subrecursive computability in presence of probabilistic choice operators, where computation itself becomes a stochastic process.

However, we barely scratched the tip of the iceberg, since the kinds of probabilistic choice operators we consider here are just examples of how one can turn a deterministic calculus like $\mathbb{T}$ into a probabilistic model of computation. The expressiveness of $\mathbb{T}^{\oplus,\mathtt{R},\mathtt{X}}$ is sufficient to encode most reasonable probabilistic operators, but what can we say about their own expressive power? For example, what about a ternary operator in which either of the first two operators is chosen with a probability *which depends* on the value of the third operator?

This ternary operator would have the type $\mathtt{Ter} : A{\to}A{\to}(\mathtt{NAT}{\to}\mathtt{NAT}){\to}A$ where the third argument $z : \mathtt{NAT}{\to}\mathtt{NAT}$ is seen as a probability $p \in [0,1]$ (whose n^{th} binary component is given by $(z\ \boldsymbol{n})$). The expressivity of $\mathbb{T}^{\mathtt{R}}$ is sufficient to encode $\mathtt{Ter} := \lambda xyz.\mathtt{rec}\ x\ (\lambda uv.y)\ (z\ \mathtt{R})$. The expressivity of $\mathbb{T}^{\mathtt{Ter}}$, however, strictly lies between that of $\mathbb{T}^{\oplus}$ and of $\mathbb{T}^{\mathtt{R}}$: $\mathbb{T}^{\mathtt{Ter}}$ can construct non binomial distributions[8] while enforcing PAST. A general theory of probabilistic choice operators and of their expressive power is still lacking.

Another research direction to which this paper hints at consists in studying the logical and proof-theoretical implications of endowing a calculus like $\mathbb{T}$ with probabilistic choice operators. The calculus $\mathbb{T}$ was born as a language of realizers for arithmetical formulas, and indeed the class of first-order functions $\mathbb{T}$ can express precisely corresponds to the ones which are provably total in Peano's arithmetic. But how about, e.g., $\mathbb{T}^{\mathtt{R}}$? Is there a way to characterize the functions (from natural numbers to *distributions* of natural numbers) which can be represented in it? Or even better: to which extent do *real* numbers in the codomain of a distribution in the form $[\![M]\!]$ (where M is, say, a $\mathbb{T}^{\mathtt{R}}$ term of type $\mathtt{NAT}$) are computable? They are of course computable in the sense of Turing computability, but how about subrecursive notions of real-number computability?

What is even more exciting, however, is the application of the ideas presented here to polynomial time computation. This would allow to go towards a characterization of expected polynomial time computation, thus greatly improving on the existing works on the implicit complexity of probabilistic systems [4, 6], which only deals with worst-case execution time. The authors are currently engaged in that.

Acknowledgements:

We would like to thank Martin Avanzini and Charles Grellois for their precise comments and for their careful proofreading.

[8]Such as $\mathtt{Ter}\ \mathbf{0}\ \mathbf{1}\ (\mathtt{rec}\ \mathbf{0}\ (\lambda x.\mathtt{rec}\ \mathbf{1}(\lambda yz.\mathbf{0})))$.

References

[1] Henk P. Barendregt. *The Lambda Calculus, Its Syntax and Semantics.* 1984.

[2] Olivier Bournez and Florent Garnier. Proving positive almost-sure termination. In *RTA*, volume 3467 of *Lecture Notes in Computer Science*, pages 323–337, 2005.

[3] Raphaëlle Crubillé and Ugo Dal Lago. On probabilistic applicative bisimulation and call-by-value λ-calculi. In *ESOP*, pages 209–228, 2014.

[4] Ugo Dal Lago and Paolo Parisen Toldin. A higher-order characterization of probabilistic polynomial time. *Inf. Comput.*, 241:114–141, 2015.

[5] Ugo Dal Lago and Margherita Zorzi. Probabilistic operational semantics for the lambda calculus. *RAIRO - Theor. Inf. and Applic.*, 46(3):413–450, 2012.

[6] Ugo Dal Lago, Sara Zuppiroli, and Maurizio Gabbrielli. Probabilistic recursion theory and implicit computational complexity. *Sci. Ann. Comp. Sci.*, 24(2):177–216, 2014.

[7] Karel De Leeuw, Edward F Moore, Claude E Shannon, and Norman Shapiro. Computability by probabilistic machines. *Automata studies*, 34:183–198, 1956.

[8] Thomas Ehrhard, Michele Pagani, and Christine Tasson. Probabilistic Coherence Spaces are Fully Abstract for Probabilistic PCF. In P. Sewell, editor, *POPL*. ACM, 2014.

[9] Luis María Ferrer Fioriti and Holger Hermanns. Probabilistic termination: Soundness, completeness, and compositionality. In *POPL*, pages 489–501, 2015.

[10] John T. Gill, III. Computational complexity of probabilistic turing machines. In *Proceedings of STOC 1974*, pages 91–95. ACM, 1974.

[11] Jean-Yves Girard, Paul Taylor, and Yves Lafont. *Proofs and Types.* Cambridge University Press, 1989.

[12] Noah D. Goodman, Vikash K. Mansinghka, Daniel M. Roy, Keith Bonawitz, and Joshua B. Tenenbaum. Church: a language for generative models. In *UAI*, pages 220–229, 2008.

[13] Achim Jung and Regina Tix. The troublesome probabilistic powerdomain. *Electr. Notes Theor. Comput. Sci.*, 13:70–91, 1998.

[14] Benjamin Lucien Kaminski and Joost-Pieter Katoen. On the hardness of almost-sure termination. In *MFCS*, volume 9234 of *LNCS*, pages 307–318, 2015.

[15] Dexter Kozen. Semantics of probabilistic programs. *J. Comput. Syst. Sci.*, 22(3):328–350, 1981.

[16] Christopher D Manning and Hinrich Schütze. *Foundations of statistical natural language processing*, volume 999. MIT Press, 1999.

[17] Annabelle McIver and Carroll Morgan. *Abstraction, Refinement and Proof for Probabilistic Systems.* Monographs in Computer Science. Springer, 2005.

[18] Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference.* Morgan Kaufmann, 1988.

[19] Simon J. D. Prince. *Computer Vision: Models, Learning, and Inference.* Cambridge University Press, New York, NY, USA, 2012.

[20] N. Saheb-Djahromi. Probabilistic LCF. In *MFCS*, pages 442–451, 1978.

[21] Eugene S Santos. Probabilistic Turing machines and computability. *Proceedings of the American Mathematical Society*, 22(3):704–710, 1969.

[22] Morten Heine Sørensen and Pawel Urzyczyn. *Lectures on the Curry-Howard Isomorphism,*. Elsevier Science Inc., New York, NY, USA, 2006.

[23] Richard Statman. The typed lambda-calculus is not elementary recursive. *Theor. Comput. Sci.*, 9:73–81, 1979.