On Bar Recursion and Choice in a Classical Setting

Valentin Blot and Colin Riba

ENS de Lyon, Université de Lyon, LIP ⋆
valentin.blot@ens-lyon.fr http://perso.ens-lyon.fr/valentin.blot/
colin.riba@ens-lyon.fr http://perso.ens-lyon.fr/colin.riba/

Abstract. We show how Modified Bar-Recursion, a variant of Spector's Bar-Recursion due to Berger and Oliva can be used to realize the Axiom of Countable Choice in Parigot's Lambda-Mu-calculus, a direct-style language for the representation and evaluation of classical proofs.
We rely on Hyland-Ong innocent games. They provide a model for the instances of the axiom of choice usually used in the realization of classical choice with Bar-Recursion, and where, moreover, the standard datatype of natural numbers is in the image of a CPS-translation.

1 Introduction

Peano's Arithmetic in all finite types (PA^ω) is a multisorted version of first-order Peano's Arithmetic, with one sort for each simple type, together with the constants of Gödel's System T and their defining equations. When augmenting PA^ω with the Axiom of Countable Choice (CAC), we obtain a system known to contain large parts of classical analysis (see e.g. [9, 16]). A similar system can be obtained by extending Peano's Arithmetic to Second-Order Logic (see e.g. [16]).

We are interested here in the realizability interpretation of $\mathsf{PA}^\omega + \mathsf{CAC}$. Realizability is a mathematical tool, part of the Curry-Howard correspondence, used to extract computational content from formal proofs.

The usual route to get a computational interpretation of (some extension of) PA^ω is to apply a negative translation, yielding proofs in (some extension of) Heyting's Arithmetic in all finite types (HA^ω, the intuitionist variant of PA^ω, see e.g. [19]), followed by a computational interpretation of the translated proofs. Realizability for HA^ω can be obtained in simply-typed settings, typically using Gödel's System T. In this way, CAC is translated to a formula which can be realized by combining a realizer of the *Intuitionistic Axiom of Choice* (IAC) with a realizer of the *Double Negation Shift* (DNS, see Sect. 3). Intuitionistic choice is easily realizable, and realizers of DNS can be obtained by adapting Spector's *Bar-Recursion* to realizability [3, 4].

We are interested here in a computational interpretation of $\mathsf{PA}^\omega + \mathsf{CAC}$ based on a realizability interpretation directly for classical proofs. It has been noted by Griffin [6] that the control operator `call/cc` of the functional language Scheme

⋆ UMR 5668 CNRS ENS Lyon UCBL INRIA

can be typed using *Peirce's Law*, which gives full Classical Logic when added to Intuitionistic Logic. Since then, there have been much work on calculi for Classical Logic, starting from Parigot's $\lambda\mu$-calculus [14]. Moreover, Krivine has developed a notion of Classical Realizability for Second-Order Peano's Arithmetic which relies on Girard's System F [10] (see also [13, 12]).

In this paper, we investigate a version of Spector's Bar-Recursion in a classical realizability setting for PA^ω, obtained by adapting Krivine's Realizability to a simply-typed extension of Parigot's $\lambda\mu$-calculus. Handling Bar-Recursion in realizability (typically to show that it realizes DNS) usualy involves some form of the axiom of choice (typically bar-induction). Suitable instances of bar-induction can be applied to some programming language extended with infinite terms, as in [3]. Another possibility, as done in [4], is to internalize realizability in the logic, reason within the logic on finite terms using bar-induction, and provide a suitable model (typically a model of PCF). Similarly to [3] and contrary to [4], our notion of realizability is not internalized in the logic. For extraction of programs from proofs, our approach is similar to [4]: we separate the programming language from the model in which the realizability argument is made.

Most non-degenerate models and operational semantics for the $\lambda\mu$-calculus rely on CPS translations (see e.g. [15]). We work here with the call-by-name translation of Lafont-Reus-Streicher (see e.g. [18, 15]). In the coproduct completion of the innocent unbracketed Hyland-Ong game model of PCF [8, 11], the usual flat game arena of natural numbers is in the image of such a CPS translation (this was observed in [11]).

We define a notion of classical realizability in this game model. Our main result is that the usual realizer of classical choice obtained by combining a realizer of IAC with Berger-Oliva's variant of Bar-Recursion [4], is indeed a realizer of classical countable choice in our framework. We then obtain an extraction result for the $\lambda\mu$-terms by a logical relation argument (see e.g. [2]), relating the operational semantics and the model.

The paper is organized as follows: We begin by presenting PA^ω in Sect. 2. We then briefly discuss the usual computational interpretation of CAC by negative translation in Sect. 3. In Sect. 4, we present the bare minimum we need on Hyland-Ong games. Parigot's $\lambda\mu$-calculus, as well as its game interpretation and its operational semantics are discussed in Sect. 5. We then devise our notion of realizability in Sect. 6 and discuss the realization of CAC in Sect. 7.

2 Peano's Arithmetic in All Finite Types

In this section, we briefly discuss the logical system on which we work in this paper, namely PA^ω (Peano's Arithmetic in all finite types), as well as its extension with the axiom of countable choice. We build on usual versions of HA^ω (see e.g. [19, 9]), with ideas of [14, 10] for classical logic.

Language. The language of PA^ω is multisorted, with one sort for each simple type. We use the following syntax of simple types, where ι is intended to be the base type of natural numbers:

$$\sigma, \tau \in \mathcal{T} \quad ::= \quad \iota \quad | \quad \sigma \to \tau \quad | \quad \sigma \times \tau$$

We assume given, for each simple type τ, a countable set $\mathcal{V}_\tau = \{x^\tau, y^\tau, \dots\}$ of *individual variables of type* τ. Individuals are simply-typed terms

$$a, b \in \mathcal{I} \quad ::= \quad x^\tau \quad | \quad ab \quad | \quad \mathsf{c}$$

where $(ab)^\tau$ provided $a^{\sigma\to\tau}$, b^σ for some σ, and c ranges over the constants $\mathsf{0}^\iota$, $\mathsf{S}^{\iota\to\iota}$, $\mathsf{Rec}^{\tau\to(\iota\to\tau\to\tau)\to\iota\to\tau}$, $\mathsf{Pair}^{\sigma\to\tau\to\sigma\times\tau}$, $\mathsf{P}_i^{\tau_1\times\tau_2\to\tau_i}$ $(i = 1, 2)$, $\mathsf{k}^{\sigma\to\tau\to\sigma}$ and $\mathsf{s}^{(\rho\to\sigma\to\tau)\to(\rho\to\sigma)\to\rho\to\tau}$. Let $\mathcal{I}_0$ be the set of closed individuals and $\mathcal{I}_0^\tau$ be the set of closed individuals of type τ.

Formulas are defined as follows:

$$A, B \in \mathcal{F} \quad ::= \quad (a^\tau \neq_\tau b^\tau) \quad | \quad \bot \quad | \quad A \Rightarrow B \quad | \quad A \wedge B \quad | \quad \forall x^\tau A$$

Note the atomic inequality $(\_ \neq_\tau \_)$. It is inspired from Krivine's work [10] and will greatly ease our realizability interpretation (see Sect. 6).

We use the following abbreviations:

$$\begin{array}{rcl@{\qquad}rcl} \neg A & := & A \Rightarrow \bot & \exists x^\tau A & := & \neg\forall x^\tau \neg A \\ (a =_\tau b) & := & \neg(a \neq_\tau b) & A \vee B & := & \neg(\neg A \wedge \neg B) \end{array}$$

Deduction. We consider the following deduction system (see e.g. [14]). It is parametrized by a set Ax of axioms (containing only closed formulas).

$$\frac{}{\Gamma, A \vdash A \mid \Delta} \qquad \frac{}{\Gamma \vdash A \mid \Delta}\ (A \in Ax) \qquad \frac{\Gamma \vdash \bot \mid \Delta}{\Gamma \vdash a^\tau \neq_\tau b^\tau \mid \Delta}$$

$$\frac{\Gamma, A \vdash B \mid \Delta}{\Gamma \vdash A \Rightarrow B \mid \Delta} \qquad \frac{\Gamma \vdash A \Rightarrow B \mid \Delta \qquad \Gamma \vdash A \mid \Delta}{\Gamma \vdash B \mid \Delta}$$

$$\frac{\Gamma \vdash A \mid \Delta \qquad \Gamma \vdash B \mid \Delta}{\Gamma \vdash A \wedge B \mid \Delta} \qquad \frac{\Gamma \vdash A_1 \wedge A_2 \mid \Delta}{\Gamma \vdash A_i \mid \Delta}\ (i = 1, 2)$$

$$\frac{\Gamma \vdash A \mid \Delta}{\Gamma \vdash \forall x^\tau A \mid \Delta}\ (x \notin \mathrm{FV}(\Gamma, \Delta)) \qquad \frac{\Gamma \vdash \forall x^\tau A \mid \Delta}{\Gamma \vdash A[a^\tau/x] \mid \Delta}$$

$$\frac{\Gamma \vdash A \mid \Delta, A}{(\Gamma \vdash \Delta, A)} \qquad \frac{(\Gamma \vdash \Delta, A)}{\Gamma \vdash A \mid \Delta}$$

This system is chosen so as to have a direct extraction of realizers in Parigot's $\lambda\mu$-calculus (see Sect. 5 and 6).

Note that the Ex Falso rule is restricted to atomic formulas. For each formula A one can easily derive $\Gamma \vdash A \mid \Delta$ from $\Gamma \vdash \bot \mid \Delta$. The introduction rules for existential quantification and disjunction are easy to derive:

$$\frac{\Gamma \vdash A \mid \Delta}{\Gamma \vdash A \vee B \mid \Delta} \qquad \frac{\Gamma \vdash A[a^\tau/x] \mid \Delta}{\Gamma \vdash \exists x^\tau A \mid \Delta}$$

One can also derive Peirce's Law and Double Negation Elimination (see e.g. [14]):

$$\overline{\Gamma \vdash ((A \Rightarrow B) \Rightarrow A) \Rightarrow A \mid \Delta} \qquad \overline{\Gamma \vdash ((A \Rightarrow \bot) \Rightarrow \bot) \Rightarrow A \mid \Delta}$$

as well as the elimination rules of disjunction and existential quantification: $\Gamma \vdash C \mid \Delta$ provided $\Gamma \vdash A \vee B \mid \Delta$, $\Gamma, A \vdash C \mid \Delta$ and $\Gamma, B \vdash C \mid \Delta$; and $\Gamma \vdash C \mid \Delta$ provided $\Gamma \vdash \exists x^\tau A \mid \Delta$ and $\Gamma, A \vdash C \mid \Delta$ with x not free in Γ, C, Δ.

Axioms for Equality and Arithmetic. The axioms of PA^ω are the universal closures of the following formulas:

– Equality axioms are *reflexivity* $\forall x^\tau (x =_\tau x)$ and *Leibniz's scheme*:

$$\text{for all formula } A, \qquad \forall x^\tau y^\tau (A[x/z] \;\Rightarrow\; \neg A[y/z] \;\Rightarrow\; x \neq_\tau y)$$

Note that the usual version of Leibniz's scheme is derivable:

$$\forall x^\tau y^\tau (x =_\tau y \;\Rightarrow\; A[x/z] \;\Rightarrow\; A[y/z])$$

– Equational axioms (with variables of the appropriate types):

$$\mathsf{k}\, x\, y \;=_\tau\; x \qquad \mathsf{s}\, x\, y\, z \;=_\tau\; x\, z\, (y\, z) \qquad \mathsf{P}_i\, (\mathsf{Pair}\, x_1\, x_2) \;=_{\tau_i}\; x_i \quad (i = 1, 2)$$
$$\mathsf{Rec}\, x\, y\, \mathsf{0} \;=_\tau\; x \qquad \mathsf{Rec}\, x\, y\, (\mathsf{S}\, z) \;=_\tau\; y\, z\, (\mathsf{Rec}\, x\, y\, z)$$

– Arithmetic axioms are $\forall x^\iota (\mathsf{S}\, x \neq_\iota \mathsf{0})$ and the *Induction scheme*:

$$\text{for all formula } A, \qquad A[\mathsf{0}/x] \;\Rightarrow\; \forall x^\iota (A \;\Rightarrow\; A[\mathsf{S}x/x]) \;\Rightarrow\; \forall x^\iota A$$

We write $\mathsf{PA}^\omega \vdash A$ if $\vdash A|$ is derivable using the axioms of PA^ω.

Axiom of Countable Choice. Given $\tau \in \mathcal{T}$, we write $\mathsf{CAC}^{\iota,\tau}$ for the following version of the axiom (scheme) of countable choice:

$$\text{for all formula } A, \qquad (\forall x^\iota \exists y^\tau A) \;\Rightarrow\; \exists f^{\iota \to \tau} \forall x^\iota A[fx/y]$$

Note that this unfolds to

$$\forall x^\iota (\forall y^\tau (A \Rightarrow \bot) \Rightarrow \bot) \;\Rightarrow\; \forall f^{\iota \to \tau} (\forall x^\iota A[fx/y] \Rightarrow \bot) \;\Rightarrow\; \bot$$

We write $\mathsf{PA}^\omega + \mathsf{CAC}^{\iota,-}$ for provability in PA^ω using any $\mathsf{CAC}^{\iota,\tau}$ for $\tau \in \mathcal{T}$.

3 Intuitionistic Modified Realizability and Bar-Recursion

In this section, we briefly and informally recall the realization of CAC via negative translation to $\mathsf{HA}^\omega + \mathsf{DNS}$, and discuss some aspects of our realization of CAC.

HA^ω can be obtained from our presentation of PA^ω by restricting deduction to *intuitionistic sequents*, *i.e.* sequents of the form $\Gamma \vdash A|$. One also has to take a primitive notion of equality (instead of our primitive $(\_ \neq_\tau \_)$), and primitive existential quantification (disjunction can be coded). Gödel's negative translation maps PA^ω to HA^ω: let $(\_)^\neg$ commute over the connectives of PA^ω (remember that there is no $\vee$, $\exists$ in $\mathcal{F}$), and put $\neg\neg$ in front of atomic formulas, after having replaced $(a \neq_\tau b)$ by $\neg(a =_\tau b)$. It is equivalent to leave $\bot$ unchanged and map $(a \neq_\tau b)$ to $\neg(a =_\tau b)$.

Let us briefly discuss Modified Realizability. To each closed formula A is associated a simple type A^* of potential realizers of A. Actual realizers of A are closed terms of type A^* satisfying a property, usually written $t \Vdash A$, defined by induction on A. Typical clauses are:

$$\begin{array}{rclrcl} t^\iota \Vdash \bot & := & \bot & t^\iota \Vdash (a =_\tau b) & := & (a =_\tau b) \\ t \Vdash (A \Rightarrow B) & := & \forall u(u \Vdash A \ \Rightarrow \ tu \Vdash B) & t \Vdash \forall x^\tau A & := & \forall x^\tau (tx \Vdash A) \\ t \Vdash (A \wedge B) & := & (\mathsf{P}_1 t \Vdash A \ \wedge \ \mathsf{P}_2 t \Vdash B) & t \Vdash \exists x^\tau A & := & (\mathsf{P}_2 t \Vdash A[\mathsf{P}_1 t/x]) \end{array}$$

Note that this provides a realizer, written t_{IAC}, of intuitionistic choice $(\mathsf{IAC}^{\sigma,\tau})$[1]:

$$\lambda z.\mathsf{Pair}\,(\lambda x.\mathsf{P}_1(zx))\,(\lambda x.\mathsf{P}_2(zx)) \quad \Vdash \quad (\forall x^\sigma \exists y^\tau A) \ \Rightarrow \ \exists f^{\sigma \to \tau} \forall x^\sigma A[fx/y]$$

A proof in PA^ω of a formula A can be mapped to a realizer of the negative translation $A^\neg$ of A[2]. For $\mathsf{CAC}^{\iota,\tau}$, this leads (modulo the intuitionistic equivalence $\neg\forall\neg \longleftrightarrow \neg\neg\exists$) to find a realizer of

$$\forall x^\iota \neg\neg\exists y^\tau A^\neg \ \Rightarrow \ \neg\neg\exists f^{\iota\to\tau} \forall x^\iota A^\neg[fx/y]$$

It is well-known (see e.g. [3, 4, 9]) that such a realizer can be obtained by combining a realizer of $\mathsf{IAC}^{\iota,\tau}$ with a realizer of the *Double Negation Shift*

$$(\forall x^\iota \neg\neg B) \ \Rightarrow \ \neg\neg\forall x^\iota B \qquad (\mathsf{DNS})$$

for the instance $B := \exists y^\tau A$. Assuming Ψ realizes this instance of DNS, we get

$$\lambda z.\lambda k.\Psi z(\lambda a.k(t_{\mathsf{IAC}} a)) \quad \Vdash \quad \forall x^\iota \neg\neg\exists y^\tau A^\neg \ \Rightarrow \ \neg\neg\exists f^{\iota\to\tau} \forall x^\iota A^\neg[fx/y]$$

The reader can check that we obtain the following realizer of CAC:

$$\begin{array}{rcl} t_{\mathsf{CAC}} & := & \lambda z.\lambda c.\Psi(t_{\neg\neg\exists} z)(\lambda a.c(\lambda x.\mathsf{P}_1(ax))(\lambda x.\mathsf{P}_2(ax))) \quad \Vdash \\ & & \quad \forall x^\iota(\forall y^\tau(A \Rightarrow \bot) \Rightarrow \bot) \ \Rightarrow \ \forall f^{\iota\to\tau}(\forall x^\iota A[fx/y] \Rightarrow \bot) \ \Rightarrow \ \bot \end{array}$$

$$\text{with } t_{\neg\neg\exists} \ := \ \lambda a.\lambda x.\lambda k.ax(\lambda y.\lambda z.k(\mathsf{Pair}\,y\,z)) \ \Vdash \ \forall x^\iota \neg\forall y^\tau \neg A \ \Rightarrow \ \forall x^\iota \neg\neg\exists y^\tau A$$

Realizers Ψ of DNS can be obtained by adapting Spector's *Bar-Recursion* to realizability [3, 4].

The purpose of this paper is to realize CAC using Bar-Recursion, directly in a language for classical proofs. We show that (the interpretation in a suitable model

[1] We use the λ-notation for individual terms in $\mathcal{I}$.

[2] To get extraction for Π^0_2-formulas, one can adapt Friedman's trick by defining $(t^\iota \Vdash \bot)$ as $\perp\!\!\!\perp(t)$, where $\perp\!\!\!\perp$ is a given predicate, see e.g. [4] and also Sect. 6.

of) t_{CAC} realizes $\mathsf{CAC}^{\iota,\tau}$, for a notion of realizability defined for (the interpretation in a suitable model of) an extension of Parigot's $\lambda\mu$-calculus [14].

Most non-degenerate models and operational semantics for the $\lambda\mu$-calculus rely on CPS translations (see e.g. [15]). If we CPS-translate Bar-Recursion we obtain a term of type

$$(\iota^{\neg} \to (\tau \to \iota^{\neg}) \to \iota^{\neg}) \to ((\iota^{\neg} \to \tau) \to \iota^{\neg}) \to \iota^{\neg}$$

with $\iota^{\neg} := (\iota \to \mathtt{R}) \to \mathtt{R}$. Obvious choices for $\mathtt{R}$ besides (a model of) ι, e.g. a one-point object, tend to give degenerated results: typically, in domains (and even predomains [18]), taking $\mathtt{R} = \{\bot\}$ ($\mathtt{R} = \emptyset$) gives a unique inhabitant in $\iota^{\neg}$. We use the fact, observed in [11], that in the coproduct completion (given by the Fam construction, see e.g. [1]) of Hyland-Ong innocent unbracketed games for PCF, the basic type of natural numbers is of the form $([\![\mathbb{N}]\!] \to \mathtt{R}) \to \mathtt{R}$, for the one-move game $\mathtt{R}$ and the countable family of empty games $[\![\mathbb{N}]\!]$ (see Sect. 4). We then reason using the usual argument [4, 3].

4 The Model of Hyland-Ong Games

In this section, we present the bare minimum we need on Hyland-Ong games. We use innocent unbracketed games, combined with the coproduct completion provided by the Fam construction. Details can be found in e.g. [8, 7, 11, 1].

4.1 Arenas and Strategies

Definition 4.1 (Arena). *An* arena *is a countable forest of* moves*. Each move is given a polarity O (for Opponent) or P (for Player or Proponent):*

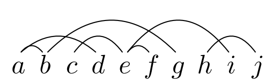
- *A root is of polarity O.*
- *A move which is not a root has the inverse polarity of that of his parent.*

A root of an arena is also called an initial move. We will often identify an arena with its set of moves.

Definition 4.2 (Justified sequence). *Given an arena $\mathcal{A}$, we define a* justified sequence *on $\mathcal{A}$ to be a finite word s on $\mathcal{A}$ together with a partial* justifying function *$f : |s| \rightharpoonup |s|$ such that:*

- *If $f(i)$ is undefined, then s_i is an initial move.*
- *If $f(i)$ is defined, then $f(i) < i$ and s_i is a child of $s_{f(i)}$.*

We denote the empty justified sequence by ϵ. Remark here that by definition of the polarity, if $f(i)$ is undefined (s_i is initial), then s_i is of polarity O, and if $f(i)$ is defined, then s_i and $s_{f(i)}$ are of opposite polarities. Also, $f(0)$ is never defined, and so s_0 is always an initial O-move. A justified sequence is represented for example as:



If $\mathcal{A}$ is an arena, X is a subset of $\mathcal{A}$ and s is a justified sequence on $\mathcal{A}$, then $s_{|X}$ is the subsequence of s consisting of the moves of s which are in X.

Definition 4.3 (Play). *A* play *s on $\mathcal{A}$ is an even and alternating justified sequence of $\mathcal{A}$, i.e., for any i, s_{2i} is a O-move and s_{2i+1} is a P-move. We denote the set of plays of $\mathcal{A}$ by $\mathcal{P}_{\mathcal{A}}$.*

A play on an arena is the trace of an interaction between a program and a context, each one performing an action alternatively.

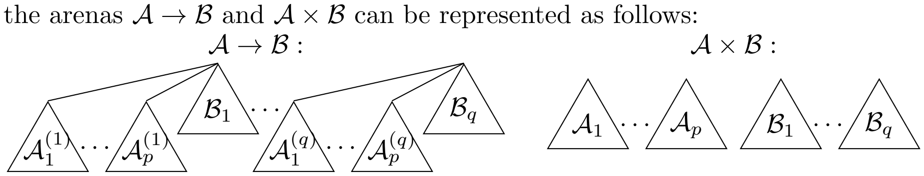
Definition 4.4 (Strategy). *A* strategy *σ on $\mathcal{A}$ is a non-empty even-prefix-closed set of finite plays on $\mathcal{A}$ such that:*

- *σ is deterministic*
- *σ is innocent*

The definitions of determinism and innocence are standard and can be found for example in [7, 8].

Cartesian Closed Structure. The constructions we use will sometimes contain multiple copies of the same arena (for example $\mathcal{A} \to \mathcal{A}$), so we distinguish the instances with superscripts (for example $\mathcal{A}^{(1)} \to \mathcal{A}^{(2)}$).

Let $\mathcal{U}$ be the empty arena and $\mathcal{V}$ be the arena with only one (opponent) move. If $\mathcal{A}$ and $\mathcal{B}$ are arenas consisting of the trees $\mathcal{A}_1 \dots \mathcal{A}_p$ and $\mathcal{B}_1 \dots \mathcal{B}_q$, then the arenas $\mathcal{A} \to \mathcal{B}$ and $\mathcal{A} \times \mathcal{B}$ can be represented as follows:



The constructions described here define a cartesian closed category whose objects are arenas and morphisms are innocent strategies. Details of the construction can be found in [7, 8]. In the following this category will be denoted as $\mathcal{G}$.

4.2 The Fam Construction

Our model is built as a continuation category [18]. In order to make explicit the double negation translation of the base types, we base the model on the category of continuations $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$, where $\mathsf{Fam}(\mathcal{G})$ is a variant of the coproduct completion described in [1] applied to the category $\mathcal{G}$ defined in Sect. 4.1.

Definition 4.5 ($\mathsf{Fam}(\mathcal{G})$). *The objects of $\mathsf{Fam}(\mathcal{G})$ are families of objects of $\mathcal{G}$ indexed by at most countable sets, and a morphism from $\{A_i \mid i \in I\}$ to $\{B_j \mid j \in J\}$ is a function $f : I \to J$ together with a family of morphisms of $\mathcal{G}$ from A_i to $B_{f(i)}$, for $i \in I$.*

See [5] for details on the differences with [1]. Note that $\mathsf{Fam}(\mathcal{G})$ is a distributive category with finite products and coproducts, and has exponentials of all singleton families. The empty product and terminal object is the singleton family $\{\mathcal{U}\}$, the empty sum and initial object is the empty family $\{\}$, and:

$$\begin{aligned}
\{A_i \mid i \in I\} \times \{B_j \mid j \in J\} &:= \{A_i \times B_j \mid (i,j) \in I \times J\} \\
\{A_i \mid i \in I\} + \{B_j \mid j \in J\} &:= \{C_k \mid k \in I \uplus J\} \text{ where } C_k := \begin{cases} A_k \text{ if } k \in I \\ B_k \text{ if } k \in J \end{cases} \\
\{B_0\}^{\{A_i \mid i \in I\}} &:= \{\Pi_{i \in I} B_0^{A_i}\}
\end{aligned}$$

We fix once and for all:

$$\mathtt{R} \quad := \quad \{\mathcal{V}\}$$

which is an object of $\mathsf{Fam}(\mathcal{G})$ as a singleton family. $\mathtt{R}$ has all exponentials as stated above. Note that the canonical morphism $\delta_A : A \to \mathtt{R}^{(\mathtt{R}^A)}$ is a mono.

The category of continuations $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$ is the full subcategory of $\mathsf{Fam}(\mathcal{G})$ consisting of the objects of the form $\mathtt{R}^A$. The objects of $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$ are singleton families, and $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$ is isomorphic to $\mathcal{G}$. We will consider that objects and morphisms of $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$ are arenas and strategies and we will use the vocabulary defined at the end of Sect. 4.1 on $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$ also.

4.3 The Type Structure

We use the lambda notation in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$, *i.e.* we build simply-typed λ-terms with constants in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$. We write them using bold symbols (such as $\boldsymbol{\lambda}$, $\langle\_,\_\rangle$ etc) in order make no confusion with the syntactic $\lambda\mu$-terms of Section 5.

Interpretation of Simple Types. Let $[\![\mathbb{N}]\!]$ be the object $\{\mathcal{U}_n \mid n \in \mathbb{N}\}$ of $\mathsf{Fam}(\mathcal{G})$. We use the interpretation of simple types proposed in [18] (see also [15]). Given a simple type $\tau \in \mathcal{T}$, we associate two objects of $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$: the object $[\tau]$ of *programs* of type τ, and the object $[\![\tau]\!]$ of *continuations* of type τ. We let

$$[\![\iota]\!] := \mathtt{R}^{[\![\mathbb{N}]\!]} \qquad [\![\sigma \to \tau]\!] := \mathtt{R}^{[\![\sigma]\!]} \times [\![\tau]\!] \qquad [\![\sigma \times \tau]\!] := [\![\sigma]\!] + [\![\tau]\!] \qquad [\tau] := \mathtt{R}^{[\![\tau]\!]}$$

Note that $[\![\sigma \to \tau]\!] = [\sigma] \times [\![\tau]\!]$, and moreover

$$[\sigma \to \tau] \quad = \quad \mathtt{R}^{[\sigma] \times [\![\tau]\!]} \quad \simeq \quad \mathtt{R}^{[\![\tau]\!]^{\mathtt{R}^{[\![\sigma]\!]}}} \qquad \text{and} \qquad [\sigma \times \tau] \quad \simeq \quad \mathtt{R}^{[\![\sigma]\!]} \times \mathtt{R}^{[\![\tau]\!]}$$

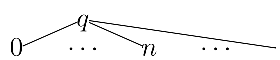
Representation of Arithmetic Constants. In $\mathsf{Fam}(\mathcal{G})$ a morphism from the terminal object $\{\mathcal{U}\}$ to $[\![\mathbb{N}]\!] = \{\mathcal{U}_n \mid n \in \mathbb{N}\}$ is given by a function from the singleton set to $\mathbb{N}$ together with a strategy from $\mathcal{U}$ to $\mathcal{U}$. Since there is only one such strategy, such a morphism is given by a natural number. We will call this morphism $\widetilde{n}$. Similarly a morphism from $[\![\mathbb{N}]\!]$ to $[\![\mathbb{N}]\!]$ is given by a function from $\mathbb{N}$ to $\mathbb{N}$. This leads to a morphism $\widetilde{\mathsf{succ}} : [\![\mathbb{N}]\!] \to [\![\mathbb{N}]\!]$ for the successor function on $[\![\mathbb{N}]\!]$.

Moreover, given $a : [\tau]$ (officially, $a : \{\mathcal{U}\} \to [\tau]$ in $\mathsf{Fam}(\mathcal{G})$), and $b : [\![\mathbb{N}]\!] \to [\tau] \to [\tau]$, we can define by induction on $n \in \mathbb{N}$ a morphism $\widetilde{r}_{a,b} : [\![\mathbb{N}]\!] \to [\tau]$ such that $\widetilde{r}_{a,b}\widetilde{0} = a$ and $\widetilde{r}_{a,b}(\widetilde{n+1}) = b\widetilde{n}(\widetilde{r}_{a,b}(\widetilde{n}))$. This leads to $\widetilde{\mathsf{rec}} := \boldsymbol{\lambda} a.\boldsymbol{\lambda} b.\widetilde{r}_{a,b}$ in $[\tau] \to ([\![\mathbb{N}]\!] \to [\tau] \to [\tau]) \to [\![\mathbb{N}]\!] \to [\tau]$.

We now discuss the object of $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$ associated to the base type ι. We have:

$$[\iota] \quad := \quad \mathtt{R}^{\mathtt{R}^{[\![\mathbb{N}]\!]}} \quad = \quad \mathtt{R}^{\mathtt{R}^{\{\mathcal{U}_n \ \mid \ n \in \mathbb{N}\}}} \quad \simeq \quad \mathtt{R}^{\Pi_{n\in\mathbb{N}}\mathtt{R}} \quad \simeq \quad \{\mathcal{V}^{\Pi_{n\in\mathbb{N}}\mathcal{V}}\}$$

Note that this is the usual flat arena of natural numbers:



It is easy to see that $\boldsymbol{\lambda} k.k\widetilde{n}$ corresponds to the strategy answering n to the initial opponent question q. Moreover, the only inhabitants of $[\iota]$ are the empty strategy $\perp_{[\iota]}$ and the strategies $\boldsymbol{\lambda} k.k\widetilde{n}$ for $n \in \mathbb{N}$.

The arithmetical constants of System T will be interpreted in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$ using $\mathbf{succ} : [\iota] \to [\iota]$ defined as $\mathbf{succ} := \boldsymbol{\lambda} n.\boldsymbol{\lambda} k.n(\boldsymbol{\lambda} x.k(\widetilde{\mathsf{succ}}\, x))$ and $\mathbf{rec} : [\tau] \to [\iota \to \tau \to \tau] \to [\iota] \to [\tau]$ with $\mathbf{rec} := \boldsymbol{\lambda} u.\boldsymbol{\lambda} v.\boldsymbol{\lambda} n.\boldsymbol{\lambda} k.n(\boldsymbol{\lambda} x.\widetilde{\mathsf{rec}}\, u(\boldsymbol{\lambda} y.v^{\bullet}(\boldsymbol{\lambda} k.ky))xk)$, where $v^{\bullet} := \boldsymbol{\lambda} x.\boldsymbol{\lambda} y.\boldsymbol{\lambda} z.v\langle x, y, z\rangle$ (see [5] for details).

It is convenient to use the notations $(\_)^{\bullet}$ and $(\_)^{\circ}$ for resp. curryfication and uncurryfication. Note that as with $v^{\bullet}$ above, the amount to which an expression is curryfied/uncurryfied depends on the context, and moreover that in $\mathcal{G}$, $(\_)^{\bullet}$ and $(\_)^{\circ}$ are the identity.

5 Lambda-Mu-Calculus

We present here an extension of Parigot's $\lambda\mu$-calculus [14] that we will use as a programming language for our realizers. We begin by a basic language, which essentially adds pairs and products to the original calculus. We then present an extension with the arithmetic constants of Gödel's System T, which will be used for the realization of PA^{ω}. Finally, we discuss the interpretation, along the lines of [15], of the calculus in the model $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$, and present an operational semantics using an abstract machine adapted from [18].

Syntax and Typing. We assume given two countable sets $\mathrm{Var} = \{x, y, z, \dots\}$ and $\mathrm{CVar} = \{\alpha, \beta, \gamma, \dots\}$ of respectively *term* and *continuation* variables. The $\lambda\mu$-terms are defined as follows:

$$t, u \in \Lambda \quad ::= \quad x \quad | \quad \lambda x.t \quad | \quad tu \quad | \quad \mu\alpha.v \quad | \quad \langle t, u\rangle \quad | \quad \mathsf{p}_1(t) \quad | \quad \mathsf{p}_2(t)$$

where v is a *named term*: $\quad v \quad ::= \quad [\alpha]t$

They are typed by extending Parigot's system [14] with rules for product types:

$$\frac{}{\Gamma, x : \tau \vdash x : \tau \mid \Delta} \qquad \frac{\Gamma \vdash t : \tau \mid \Delta, \alpha : \tau}{[\alpha]t : (\Gamma \vdash \Delta, \alpha : \tau)} \qquad \frac{v : (\Gamma \vdash \Delta, \alpha : \tau)}{\Gamma \vdash \mu\alpha.v : \tau \mid \Delta}$$

$$\frac{\Gamma, x : \tau \vdash t : \sigma \mid \Delta}{\Gamma \vdash \lambda x.t : \tau \to \sigma \mid \Delta} \qquad \frac{\Gamma \vdash t : \sigma \to \tau \mid \Delta \qquad \Gamma \vdash u : \sigma \mid \Delta}{\Gamma \vdash tu : \tau \mid \Delta}$$

$$\frac{\Gamma \vdash t : \tau \mid \Delta \qquad \Gamma \vdash u : \sigma \mid \Delta}{\Gamma \vdash \langle t, u\rangle : \tau \times \sigma \mid \Delta} \qquad \frac{\Gamma \vdash t : \tau_1 \times \tau_2 \mid \Delta}{\Gamma \vdash \mathsf{p}_i(t) : \tau_i \mid \Delta}\ (i = 1, 2)$$

Extension with Arithmetic Constants. We write Λ_T for the set of $\lambda\mu$-terms obtained by extending the grammar of Λ with the following productions:

$$t, u \quad ::= \quad \dots \quad \mid \quad \overline{n} \quad \mid \quad \mathsf{succ} \quad \mid \quad \mathsf{rec}(t, u)$$

where $n \in \mathbb{N}$. We extend the typing rules of Λ with the following ones:

$$\frac{}{\Gamma \vdash \overline{n} : \iota \mid \Delta} \qquad \frac{}{\Gamma \vdash \mathsf{succ} : \iota \to \iota \mid \Delta} \qquad \frac{\Gamma \vdash t : \tau \mid \Delta \qquad \Gamma \vdash u : \iota \to \tau \to \tau \mid \Delta}{\Gamma \vdash \mathsf{rec}(t, u) : \iota \to \tau \mid \Delta}$$

Interpretation in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$. The interpretation of Λ_T in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$ follows the lines of [15]. A term $\vdash t : \tau|$ is interpreted by $[t] \in [\tau]$. To make the presentation simpler, we use $\boldsymbol{\lambda}$-expressions in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$ build from the variables of Λ_T with the following convention: a term variable x of type τ (resp. a continuation variable α of type σ) in Λ_T becomes a variable x of type $[\tau]$ (resp. a variable α of type $[\![\sigma]\!]$) in the $\boldsymbol{\lambda}$-calculus of $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$:

$$\begin{array}{rcl} [x] & := & x \\ [\lambda x.t] & := & \boldsymbol{\lambda}\langle x, k\rangle.[t]k \\ [tu] & := & \boldsymbol{\lambda}k.[t]\langle[u], k\rangle \\ [\mathsf{succ}] & := & \boldsymbol{\lambda}\langle n, k\rangle.\,\mathbf{succ}\; n\, k \end{array} \qquad \begin{array}{rcl} [\overline{n}] & := & \boldsymbol{\lambda}k.k\widetilde{n} \\ [\langle t, u\rangle] & := & \boldsymbol{\lambda}k.\mathbf{case}\, k\{[t], [u]\} \\ [\mathsf{p}_i(t)] & := & \boldsymbol{\lambda}k.[t](\mathbf{in}_i k) \\ [\mathsf{rec}(t, u)] & := & \boldsymbol{\lambda}\langle n, k\rangle.\mathbf{rec}\,[t][u]\, n\, k \end{array} \qquad [\mu\alpha.[\beta]t] \;:=\; \boldsymbol{\lambda}\alpha.[t]\beta$$

Operational Semantics. We now present an operational semantics for Λ_T using an abstract machine. The machine is derived from the interpretation of Λ_T in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$, following the method of [18]. Our machine is actually an adaptation of the machine of [18] to a typed language with arithmetic constants.

The machine evaluates triples of the form (t, e, π), where t is a $\lambda\mu$-term, e is an environment and π is a stack. Environments map term variables to closures and continuation variables to stacks. Stacks, closures and environments are defined by mutual induction as usual:

$$\begin{array}{rrcl} \text{Env.} & e \in \mathrm{E} & ::= & \varepsilon \;\mid\; (x, c) :: e \;\mid\; (\alpha, \pi) :: e \\ \text{Closures} & c \in \mathrm{C} & ::= & (t, e) \\ \text{Stacks} & \pi \in \Pi & ::= & \star \mid \langle c, \pi\rangle \mid \mathsf{kp}_i(\pi) \mid \mathsf{ksucc}(\pi) \mid \mathsf{krec}(t, u, c, \pi) \end{array}$$

We let $e(x) := c$ if (x, c) is the first occurrence of the form (x, c') in e, and define $e(\alpha)$ similarly. Let $\mathrm{dom}(e)$ be the domain of the partial map $e(\_)$.

The evaluation rules are the following:

$$
\begin{array}{rcll}
(x, e, \pi) & \succ & (t, e', \pi) & \text{if } e(x) = (t, e') \\
\\
(tu, e, \pi) & \succ & (t, e, \langle (u, e), \pi \rangle) & \\
(\lambda x.t, e, \langle c, \pi \rangle) & \succ & (t, (x, c) :: e, \pi) & \\
\\
(\mu\alpha.[\beta]t, e, \pi) & \succ & (t, (\alpha, \pi) :: e, \pi') & \text{if } ((\alpha, \pi) :: e)(\beta) = \pi' \\
\\
(\mathsf{p}_i(t), e, \pi) & \succ & (t, e, \mathsf{kp}_i(\pi)) & i = 1, 2 \\
(\langle t_1, t_2 \rangle, e, \mathsf{kp}_i(\pi)) & \succ & (t_i, e, \pi) & i = 1, 2 \\
\\
(\mathsf{succ}, e, \langle (t, e'), \pi \rangle) & \succ & (t, e', \mathsf{ksucc}(\pi)) & \\
(\overline{n}, e, \mathsf{ksucc}(\pi)) & \succ & (\overline{n+1}, e, \pi) & \\
\\
(\mathsf{rec}(t, u), e, \langle (v, e'), \pi \rangle) & \succ & (v, e', \mathsf{krec}(t, u, e, \pi)) & \\
(\overline{0}, e, \mathsf{krec}(t, u, e', \pi)) & \succ & (t, e', \pi) & \\
(\overline{n+1}, e, \mathsf{krec}(t, u, e', \pi)) & \succ & (u, e', \langle (\overline{n}, e), \langle (\mathsf{rec}(t, u)\overline{n}, e'), \pi \rangle \rangle) &
\end{array}
$$

The correctness of the machine (*i.e.* reduction preserves semantics) can be proved as usual[3] (see e.g. [18]). For extraction, we actually only need the property stated in Prop. 7.3, to be discussed in presence of Bar-Recursion.

6 Classical Realizability

In this section, we present our notion of realizability. It is highly inspired from Krivine's Realizability [10], but adapted to the simply-typed model $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$.

The main idea, adapting Krivine's ideas to the typed continuation category $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$, would be to fix a *Pole* $\perp\!\!\!\perp \subseteq \{[\overline{n}] \mid n \in \mathbb{N}\}$, and then associate to each formula A a type A^* and a set $\mathcal{A} \subseteq [\![A^*]\!]$ defined by induction on A. Realizers would then be strategies in $\mathcal{A}^{\perp\!\!\!\perp} \subseteq [A^*]$, the *Orthogonal* of $\mathcal{A}$.

We choose to have $\perp\!\!\!\perp \subseteq [\iota]$ to get extraction (see Prop. 7.4). This causes difficulties since $[\iota] = \mathtt{R}^{[\![\iota]\!]}$ is not a base type in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$. Roughly speaking, our choice for $\perp\!\!\!\perp$ leads to $\perp^* := \iota$, but there are not enough contexts in $[\![\iota]\!] = \{\perp_{[\![\iota]\!]}\}$, since applying $\perp_{[\![\iota]\!]}$ to a numeral $[\overline{n}]$ gives the empty strategy on $\mathtt{R}$. A solution is to add some space in the interpretations, and have $\mathcal{A} \subseteq [\![\iota]\!] \to [\![A^*]\!]$ and $\mathcal{A}^{\perp\!\!\!\perp} \subseteq [\![\iota]\!] \to [A^*]$ for a formula A. For instance, we can then have $\boldsymbol{\lambda} k.k$ as a basic context "at type" $[\![\iota]\!]$ (actually $[\![\iota]\!] \to [\![\iota]\!]$).

The definition of realizability involves two additional translations, that we present now. First, to each formula A, we associate the simple type A^*:

$$
(a^\tau \neq_\tau b^\tau)^* := \iota \qquad \perp^* := \iota \qquad (\forall x^\tau A)^* := \tau \to A^*
$$
$$
(A \Rightarrow B)^* := A^* \to B^* \qquad (A \wedge B)^* := A^* \times B^*
$$

[3] Since the model $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$ is typed, this would involve typing rules for environments and stacks.

Moreover, we map each individual term $a \in \mathcal{I}$ to a $\lambda\mu$-term $a^\dagger \in \Lambda_T$:

$$
\begin{array}{rclrclrcl}
x^{\tau\dagger} & := & x & (ab)^\dagger & := & a^\dagger b^\dagger & \mathsf{s}^\dagger & := & \lambda xyz.xz(yz) \\
\mathsf{k}^\dagger & := & \lambda xy.x & \mathsf{0}^\dagger & := & \overline{0} & \mathsf{S}^\dagger & := & \mathsf{succ} \\
\mathsf{Rec}^\dagger & := & \lambda xy.\mathsf{rec}(x,y) & \mathsf{Pair}^\dagger & := & \lambda xy.\langle x,y\rangle & \mathsf{P}_i{}^\dagger & := & \lambda x.\mathsf{p}_i(x)
\end{array}
$$

The Realizability Construction. To a formula A, we will associate two sets $||A|| \subseteq [\![\iota]\!] \to [\![A^*]\!]$ and $|A| \subseteq [\![\iota]\!] \to [A^*]$. These sets will only be defined for *closed* formulas. It is convenient (and necessary to deal with CAC in Sect. 7) to allow parameters in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$. In order to realize the induction axiom, we must restrict to the *total* elements of $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$. For a simple type τ, the set $\tau^{\mathtt{t}} \subseteq [\tau]$ of its total elements is defined by induction on τ. Let $\iota^{\mathtt{t}} := \{[\overline{n}] \mid n \in \mathbb{N}\}$, and using curryfied notation:

$$
\begin{array}{rcl}
(\sigma \to \tau)^{\mathtt{t}} & := & \{a \mid \forall b \in \sigma^{\mathtt{t}},\ ab \in \tau^{\mathtt{t}}\} \\
(\sigma \times \tau)^{\mathtt{t}} & := & \{a \mid \mathbf{p}_1(a) \in \sigma^{\mathtt{t}} \ \&\ \mathbf{p}_2(a) \in \tau^{\mathtt{t}}\}
\end{array}
$$

Lemma 6.1. *For all $a \in \mathcal{I}_0^\tau$, $[a^\dagger] \in \tau^{\mathtt{t}}$.*

We now only consider closed formulas with parameters of the appropriate type in $\tau^{\mathtt{t}}$ ($\tau \in \mathcal{T}$). Let $\perp\!\!\!\perp \subseteq \iota^{\mathtt{t}}$.

First, given $\mathcal{A} \subseteq [\![\iota]\!] \to [\![A^*]\!]$, we define $\mathcal{A}^{\perp\!\!\!\perp} \subseteq [\![\iota]\!] \to [A^*]$ as

$$
\mathcal{A}^{\perp\!\!\!\perp} \quad := \quad \{a \in [\![\iota]\!] \to [A^*] \mid \forall b \in \mathcal{A},\ \boldsymbol{\lambda} k.ak(bk) \in \perp\!\!\!\perp\}
$$

If moreover $\mathcal{B} \subseteq [\![\iota]\!] \to [\![\mathcal{B}^*]\!]$, we let

$$
\mathcal{A}^{\perp\!\!\!\perp} \cdot \mathcal{B} \quad := \quad \{\boldsymbol{\lambda} k.\langle ak, bk\rangle \in [\![\iota]\!] \to [A^*] \times [\![B^*]\!] \mid a \in \mathcal{A}^{\perp\!\!\!\perp} \ \&\ b \in \mathcal{B}\}
$$

We now define the sets $|A| \subseteq [\![\iota]\!] \to [A^*]$ and $||A|| \subseteq [\![\iota]\!] \to [\![A^*]\!]$ for a formula A. We let $|A| \subseteq [\![\iota]\!] \to [A^*]$ be $||A||^{\perp\!\!\!\perp}$, and define $||A|| \subseteq [\![\iota]\!] \to [\![A^*]\!]$ by induction on A as follows:

$$
\begin{array}{rcl}
||\bot|| & := & \{\boldsymbol{\lambda} k.k\} \qquad\qquad ||A \Rightarrow B|| \quad := \quad |A| \cdot ||B|| \\
||a \neq_\tau b|| & := & \begin{cases} \emptyset & \text{if } [a^\dagger] \neq [b^\dagger] \\ \{\boldsymbol{\lambda} k.k\} & \text{otherwise} \end{cases} \\
||A \wedge B|| & := & \{\boldsymbol{\lambda} k.\mathbf{in}_1(ak) \mid a \in ||A||\} \cup \{\boldsymbol{\lambda} k.\mathbf{in}_2(bk) \mid b \in ||B||\} \\
||\forall x^\tau A|| & := & \bigcup_{a \in \tau^{\mathtt{t}}} \{\boldsymbol{\lambda} k.\langle a, bk\rangle \mid b \in ||A[a/x]||\}
\end{array}
$$

Realization of Equality and Arithmetic Axioms. We now discuss the realization of the axioms of PA^ω.

First, it is easy to see that all equational axioms (including reflexivity) are realized by the identity:

Lemma 6.2. *We have $\boldsymbol{\lambda} k.[\lambda x.x] \in |a =_\tau a|$. Moreover,*

$$
\begin{array}{cc}
\boldsymbol{\lambda} k.[\lambda x.x] \in |\mathsf{k}\, a\, b =_\tau a| & \boldsymbol{\lambda} k.[\lambda x.x] \in |\mathsf{s}\, a\, b\, c =_\tau ac(bc)| \\
\boldsymbol{\lambda} k.[\lambda x.x] \in |\mathsf{Rec}\, a\, b\, \mathsf{0} =_\tau a| & \boldsymbol{\lambda} k.[\lambda x.x] \in |\mathsf{Rec}\, a\, b\, (\mathsf{S}\, c) =_\tau bc(\mathsf{Rec}\, a\, b\, c)|
\end{array}
$$

where in each case, individuals a, b, c are in the appropriate $\tau^{\mathtt{t}}$, $\sigma^{\mathtt{t}}$, $\rho^{\mathtt{t}}$.

The realization of our version of Leibniz's scheme is obtained by applying realizers of the first premise to realizers of the second premise.

Lemma 6.3. $\boldsymbol{\lambda} k.[\lambda x.\lambda y.yx] \in |A[a^\tau/z^\tau] \Rightarrow \neg A[b^\tau/z^\tau] \Rightarrow a^\tau \neq_\tau b^\tau|$.

For the Arithmetic axioms, it is easy to see that $(\mathsf{S}a \neq_\iota \mathsf{0})$ is realized by any natural number. As expected, the recursor $\mathsf{rec}(\_,\_)$ realizes induction.

Lemma 6.4. *(i) For all $n \in \mathbb{N}$ and all $a \in \iota^{\mathtt{t}}$, we have $\boldsymbol{\lambda} k.[\overline{n}] \in |\mathsf{S}a \neq_\iota \mathsf{0}|$.*
(ii) $\boldsymbol{\lambda} k.[\lambda x.\lambda y.\mathsf{rec}(x,y)] \in |A[\mathsf{0}/x] \Rightarrow \forall x^\iota(A \Rightarrow A[\mathsf{S}x/x]) \Rightarrow \forall x^\iota A|$.

Adequacy for Classical Proofs. Adequacy of the realizability interpretation is proved as usual (see App. A).

Theorem 6.5. *Let Γ, A, Δ with $\Gamma = A_1,\ldots,A_n$, $\Delta = B_1,\ldots,B_m$, and such that $FV(\Gamma,A,\Delta) \subseteq \{x_1^{\tau_1},\ldots,x_k^{\tau_k}\}$.*

From a proof of $\Gamma \vdash A \mid \Delta$ in PA^ω one can build a term

$$x_1 : \tau_1, \ldots, x_k : \tau_k, y_1 : {A_1}^*, \ldots, y_n : {A_n}^* \vdash t : A^* \mid \alpha_1 : {B_1}^*, \ldots, \alpha_m : {B_m}^*$$

such that for all $c_1 \in {\tau_1}^{\mathtt{t}},\ldots,c_k \in {\tau_k}^{\mathtt{t}}$, all $a_1 \in |A_1[\boldsymbol{c}/\boldsymbol{x}]|,\ldots,a_n \in |A_n[\boldsymbol{c}/\boldsymbol{x}]|$, and all $b_1 \in ||B_1[\boldsymbol{c}/\boldsymbol{x}]||,\ldots,b_m \in ||B_m[\boldsymbol{c}/\boldsymbol{x}]||$, we have

$$\boldsymbol{\lambda} k.[t][\boldsymbol{c}/\boldsymbol{x}][a_1k/y_1,\ldots,a_nk/y_n,b_1k/\alpha_1,\ldots,b_mk/\alpha_m] \in |A[\boldsymbol{c}/\boldsymbol{x}]|$$

In particular, from a proof of $\vdash A|$ in PA^ω with A closed, one can build a term $\vdash t : A^*|$ such that $\boldsymbol{\lambda} k.[t] \in |A|$.

Extraction. Extraction of witnessing programs from realizable (and hence from provable) Π^0_2 statements is performed as usual. We come back on this point in Sect. 7 (Prop. 7.4) in presence of CAC and Bar-Recursion.

7 Realization of Classical Countable Choice

In this section we discuss the realization of the classical axiom of countable choice $\mathsf{CAC}^{\iota,-}$. Our realizer is based on Berger & Oliva's variant of Spector's Bar-Recursion [4].

Extension of the $\boldsymbol{\lambda\mu}$-Calculus with Bar-Recursion. We extend the set Λ_T with constants for bar-recursion: $t, u \in \Lambda_\Psi ::= \ldots \mid \Psi_\tau(t,u)\langle s_0,\ldots,s_n\rangle$, where $n \in \mathbb{N}$ and $\tau \in \mathcal{T}$.

These constants are typed as follows: $\Gamma \vdash \Psi_\tau(t,u)\langle s_0,\ldots,s_n\rangle : \iota \mid \Delta$ whenever $\Gamma \vdash t : \iota \to (\tau \to \iota) \to \iota \mid \Delta$, $\Gamma \vdash u : (\iota \to \tau) \to \iota \mid \Delta$ and $\Gamma \vdash s_i : \tau \mid \Delta$ for all $0 \leq i \leq n$.

The operational semantics uses some auxiliary terms. We define by induction on τ the terms $\vdash \mathrm{ex}_\tau : \iota \to \tau|$. Let $\mathrm{ex}_\iota := \lambda x.x$, $\mathrm{ex}_{\tau\to\sigma} := \lambda x.\lambda\_.\mathrm{ex}_\sigma x$ and $\mathrm{ex}_{\tau\times\sigma} := \lambda x.\langle \mathrm{ex}_\tau x, \mathrm{ex}_\sigma x\rangle$.

Moreover, given $n \in \mathbb{N}$, $s_0, \dots, s_n, t \in \Lambda_\Psi$, we let $\langle s_0, \dots, s_n \rangle @ t$ be a term (written using rec) such that for all $e, e' \in \mathrm{E}$, $\pi \in \Pi$ and $m \in \mathbb{N}$,

$$(\langle s_0, \dots, s_n \rangle @ t, e, \langle (\overline{m}, e'), \pi \rangle) \;\succ\; \begin{cases} (s_m, e, \pi) & \text{if } m \leq n \\ (t, e, \langle (\overline{m - (n+1)}, e'), \pi \rangle) & \text{otherwise} \end{cases}$$

The operational semantics of $\Psi_\tau(t,u)\langle s_1, \dots, s_n \rangle$ is given by:

$$\begin{array}{l} (\Psi_\tau(t,u)\langle s_0, \dots, s_n \rangle, e, \pi) \quad \succ \\ \qquad (u, e, \langle ((\langle s_0, \dots, s_n \rangle @ \lambda\_.\mathrm{ex}_\tau(t\, \overline{n+1}\, \lambda x.\Psi_\tau(t,u)\langle s_0, \dots, s_n, x \rangle), e), \pi \rangle) \end{array}$$

The Bar-Recursor in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$. We now define the strategies interpreting Ψ_τ in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$. Fix $\tau \in \mathcal{T}$. First, given $a_0, \dots, a_n \in [\tau]$, and $b \in [\iota \to \tau]$, let

$$\langle a_0, \dots, a_n \rangle \boldsymbol{@} b \quad := \quad [\langle x_0, \dots, x_n \rangle @ y][a_0/x_0, \dots, a_n/x_n, b/y]$$

For each $m \in \mathbb{N}$, we will define by induction on m a family of strategies $(\widetilde{\Psi}^m_n)_{n \in \mathbb{N}}$. Each $\widetilde{\Psi}^m_n$ will be in $[\tau]^n \to \widetilde{\tau_\Psi}$, where $[\tau]^0 := \{\mathcal{U}\}$, $[\tau]^{n+1} := [\tau] \times [\tau]^n$ and

$$\widetilde{\tau_\Psi} \quad := \quad [\iota \to (\tau \to \iota) \to \iota] \to [(\iota \to \tau) \to \iota] \to [\iota]$$

We let $\widetilde{\Psi}^0_n \;:=\; \boldsymbol{\lambda}\langle x_1, \dots, x_n \rangle.\bot_{\widetilde{\tau_\Psi}}$ and

$$\begin{array}{rcl} \widetilde{\Psi}^{m+1}_n & := & \boldsymbol{\lambda}\langle x_1, \dots, x_n \rangle.\boldsymbol{\lambda} b.\boldsymbol{\lambda} c.c^\bullet(\langle x_1, \dots, x_n \rangle \boldsymbol{@} \\ & & \qquad \boldsymbol{\lambda}\_.[\mathrm{ex}_\tau]^\bullet(b^\bullet\, [\overline{n+1}]\, (\boldsymbol{\lambda} x.\widetilde{\Psi}^m_n \langle x_1, \dots, x_n, x \rangle\; b\; c)^\circ)) \end{array}$$

Given $a_0, \dots, a_n \in [\tau]$, we now define a strategy $\widetilde{\Psi}^\tau_{\langle a_0, \dots, a_n \rangle}$ using the CPO structure on $\mathcal{G}$ (and hence on $\mathsf{Fam}(\mathcal{G})$). Note that the family $(\widetilde{\Psi}^m_{n+1}\langle a_0, \dots, a_n \rangle)_{m \in \mathbb{N}}$ is directed. We let $\widetilde{\Psi}^\tau_{\langle a_0, \dots, a_n \rangle} \;:=\; \bigvee_{m \in \mathbb{N}} \widetilde{\Psi}^m_{n+1}\langle a_0, \dots, a_n \rangle$
and $[\Psi_\tau(t,u)\langle s_0, \dots, s_n \rangle] \;:=\; \widetilde{\Psi}^\tau_{\langle [s_0], \dots, [s_n] \rangle}[t][u]$.

Realization of $\mathsf{CAC}^{\iota,-}$. We discuss here the realization of $\mathsf{CAC}^{\iota,\tau}$ using (the interpretation in Λ_Ψ) of the term t_{CAC} build in Sect. 3, where we take suitable instances of $\Psi_\tau(\_,\_)\langle \dots \rangle$ for Bar-Recursion. We let

$$\begin{array}{rrcl} & t^{\tau,A}_{\mathsf{CAC}} & := & \lambda z.\lambda c.\Psi_{\tau \times A^*}(t_{\neg\neg\exists} z,\; \lambda a.c(\lambda x.\mathsf{p}_1(ax))(\lambda x.\mathsf{p}_2(ax)))\langle\rangle \\ \text{where} & t_{\neg\neg\exists} & := & \lambda a.\lambda x.\lambda k.ax(\lambda y.\lambda z.k\langle y, z \rangle) \end{array}$$

Proposition 7.1. $\boldsymbol{\lambda} k.[t^{\tau,A}_{\mathsf{CAC}}] \in$
$|\forall x^\iota(\forall y^\tau(A \Rightarrow \bot) \Rightarrow \bot) \;\Rightarrow\; \forall f^{\iota \to \tau}(\forall x^\iota A[fx/y] \Rightarrow \bot) \;\Rightarrow\; \bot|$.

The proof of Prop. 7.1 is deferred to App. C. Contrary to e.g. [3, 4], we do not use the decomposition of CAC as $\mathsf{IAC} + \mathsf{DNS}$ discussed in Sect. 3. Rather, we show directly that Bar-Recursion realizes a form of choice.

The main point is to decompose the notion of realizability proposed in Sect. 6 w.r.t. the relativization of quantifiers. We first extend the formulas:

$$A, B \quad ::= \quad \dots \quad | \quad \tilde{\forall} x^\tau A \quad | \quad (\mathsf{r}_\tau(a^\tau) \times A) \Rightarrow B$$

Hence, in extended formulas, the construction $(\mathsf{r}_\tau(a) \times A)$ is only allowed to appear to the left of an implication. Realizability is extended as follows:

$$\begin{array}{rcl} ||\tilde{\forall} x^\tau A|| & := & \bigcup_{a \in \tau^{\mathfrak{t}}} ||A[a/x]|| \\ ||(\mathsf{r}_\tau(c) \times A) \Rightarrow B|| & := & \{\boldsymbol{\lambda} k.\langle \boldsymbol{\lambda} k'.\mathbf{case}\, k'\{c, ak\}, bk\rangle \mid a \in |A| \ \&\ b \in ||B||\} \end{array}$$

Extended formulas and their realizability interpretation rely on ideas introduced in Krivine's Realizability [10] (see also [12]). We also extend the mapping $(\_)^*$: $(\tilde{\forall} x^\tau A)^* := A^*$ and $((\mathsf{r}_\tau(a) \times A) \Rightarrow B)^* := \tau \times A^* \to B^*$. The following is the key for Prop. 7.1. It is shown as usual, see e.g. [3, 4].

Lemma 7.2. *Let B such that $(B \Rightarrow \bot)$ is an extended formula.*
Assume $b \in |\forall x^\iota(\tilde{\forall} y^\tau(B \Rightarrow \bot) \Rightarrow \bot)|$ and $c \in |\tilde{\forall} f^{\iota\to\tau}(\forall x^\iota B[fx/y] \Rightarrow \bot)|$.
Then $\boldsymbol{\lambda} k.\widetilde{\Psi}^{B^*}_{\langle\rangle}(bk)(ck) \in |\bot|$.

Computational Adequacy and Extraction. For extraction, we rely on the following property relating the evaluation of $\lambda\mu$-terms with their interpretation in $\mathsf{R}^{\mathsf{Fam}(\mathcal{G})}$. The proof is deferred to App. B.

Proposition 7.3. *(i) If $\vdash t : \iota|$ in Λ_Ψ, then for all $n \in \mathbb{N}$ we have $(t, \varepsilon, \star) \succ (\overline{n}, e, \star)$ if $[t] = [\overline{n}]$.*
(ii) Let $\vdash t : \iota \to \iota$ in Λ_Ψ. For all $n, m \in \mathbb{N}$, if $\boldsymbol{\lambda} k.[t]\langle[\overline{n}], k\rangle = [\overline{m}]$ then $(t\overline{n}, \varepsilon, \star) \succ (\overline{m}, e, \star)$.

Extraction of witnessing programs from realizable (and hence from provable) Π^0_2 statements is performed as usual:

Proposition 7.4. *From a proof of $\mathsf{PA}^\omega + \mathsf{CAC}^{\iota,-} \vdash \forall x^\iota \exists y^\iota (a =_\iota \mathsf{0})$ (where $FV(a) \subseteq \{x, y\}$), we can extract a term $\vdash t : \iota \to \iota|$ such that for all $n \in \mathbb{N}$, there is $m \in \mathbb{N}$ such that $(t\overline{n}, \varepsilon, \star) \succ (\overline{m}, e, \star)$ and $[a^\dagger][[\overline{n}]/x, [\overline{m}]/y] = [\overline{0}]$.*

Proof (sketch). By adequacy, we get u s.t. $\boldsymbol{\lambda} k.[u] \in |\forall x^\iota \neg \forall y^\iota (a \neq_\iota \mathsf{0})|$. Let $n \in \mathbb{N}$ and fix $\perp\!\!\!\perp := \{[\overline{m}] \mid [a^\dagger][[\overline{n}]/x, [\overline{m}]/y] = [\overline{0}]\}$. We thus have $\boldsymbol{\lambda} k.[u\overline{n}(\lambda x.x)] \in |\bot|$. This implies $[u\overline{n}(\lambda x.x)] = [\overline{m}]$ with $[\overline{m}] \in \perp\!\!\!\perp$. We conclude by Prop. 7.3.(ii). □

8 Conclusion

We presented a notion of classical realizability for $\mathsf{PA}^\omega + \mathsf{CAC}$ based on Hyland-Ong innocent unbracketed games for a simply-typed extension of Parigot's $\lambda\mu$-calculus. For PA^ω, these realizers seem to CPS translate to the same realizers as obtained by a negative translation from PA^ω to HA^ω followed by Friedman's translation and a realizability interpretation, as devised in Sect. 3 It is not clear whether this extends to the decomposition of CAC as IAC + DNS, because of the interaction of the CPS translation with Friedman's trick.

Further works will concern this question, a comparison with [17], where Bar-Recursion is used in an untyped Classical Realizablity model, as well as trying to extend the result to non-innocent games (along the lines of [5]), known to raise problems with Bar-Recursion [3].

References

[1] S. Abramsky and G. McCusker. Call-by-Value Games. In *Proceedings of CSL'97*, LNCS, pages 1–17. Springer, 1997. 6, 7, 8

[2] R. M. Amadio and P.-L. Curien. *Domains and Lambda-Calculi*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 1998. 2, 29, 38

[3] S. Berardi, M. Bezem, and T. Coquand. On the Computational Content of the Axiom of Choice. *Journal of Symbolic Logic*, 63(2):600–622, 1998. 1, 2, 5, 6, 14, 15, 16, 40

[4] U. Berger and P. Oliva. Modified bar recursion and classical dependent choice. *Lecture Notes in Logic*, 20:89–107, 2005. 1, 2, 5, 6, 13, 14, 15, 40

[5] V. Blot. Realizability for Peano Arithmetic with Winning Conditions in HO Games. To appear in *Proceedings of TLCA'13*, 2013. 8, 9, 16

[6] T. Griffin. A Formulae-as-Types Notion of Control. In *POPL'90*, pages 47–58. ACM Press, 1990. 1

[7] R. Harmer. *Games and Full Abstraction for Nondeterministic Languages*. PhD thesis, Imperial College, London, 1999. 6, 7

[8] J. M. E. Hyland and C.-H. Ong. On Full Abstraction for PCF: I, II, and III. *Information and Computation*, 163(2):285–408, 2000. 2, 6, 7

[9] U. Kohlenbach. *Applied Proof Theory: Proof Interpretations and their Use in Mathematics*. Springer Monographs in Mathematics. Springer, 2008. 1, 2, 5

[10] J.-L. Krivine. Realizability in classical logic. In *Interactive models of computation and program behaviour*, volume 27 of *Panoramas et synthèses*, pages 197–229. Société Mathématique de France, 2009. 2, 3, 11, 15, 40

[11] J. Laird. *A Semantic analysis of control*. PhD thesis, University of Edimbourgh, 1998. 2, 6

[12] A. Miquel. Existential witness extraction in classical realizability and via a negative translation. *Logical Methods in Computer Science*, 7(2), 2011. 2, 15

[13] P. Oliva and T. Streicher. On Krivine's Realizability Interpretation of Classical Second-Order Arithmetic. *Fundam. Inform.*, 84(2):207–220, 2008. 2

[14] M. Parigot. Lambda-Mu-Calculus: An Algorithmic Interpretation of Classical Natural Deduction. In *LPAR*, volume 624 of *LNCS*, pages 190–201. Springer, 1992. 2, 3, 4, 6, 9, 10

[15] P. Selinger. Control Categories and Duality: on the Categorical Semantics of the Lambda-Mu Calculus. *Mathematical Structures in Computer Science*, 11:207–260, 2001. 2, 6, 8, 9, 10

[16] S. G. Simpson. *Subsystems of Second Order Arithmetic*. Perspectives in Logic. Cambridge University Press, 2nd edition, 2010. 1

[17] T. Streicher. A Classical Realizability Model araising from a Stable Model of Untyped Lambda-Calculus. Unpublished Notes, 2013. 16

[18] T. Streicher and B. Reus. Classical Logic, Continuation Semantics and Abstract Machines. *J. Funct. Program.*, 8(6):543–572, 1998. 2, 6, 7, 8, 9, 10, 11, 29

[19] A. S. Troelstra. *Metamathematical Investigation of Intuitionistic Arithmetic and Analysis*, volume 344 of *LNM*. Springer Verlag, 1973. 1, 2

A Adequacy of the Realizability Interpretation

In this appendix, we give a proof of adequacy of our realizability interpretation (Thm. 6.5).

Lemma A.1. *If $c \in [\![\iota]\!] \to [\tau \to A^*]$, then:*

$$c \in |\forall x^\tau A| \Leftrightarrow \forall e \in \tau^{\mathtt{t}}, \boldsymbol{\lambda} k.[xy][ck/x, e/y] \in |A[e/x]|$$

Proof. One one hand:

$$\begin{array}{rcl} c \in |\forall x^\tau A| & \Leftrightarrow & \forall d \in ||\forall x^\tau A||, \boldsymbol{\lambda} k.ck(dk) \in \perp\!\!\!\perp \\ & \Leftrightarrow & \forall e \in \tau^{\mathtt{t}}, \forall b \in ||A[e/x]||, \boldsymbol{\lambda} k.ck((\boldsymbol{\lambda} k'.\langle e, bk'\rangle)k) \in \perp\!\!\!\perp \\ & \Leftrightarrow & \forall e \in \tau^{\mathtt{t}}, \forall b \in ||A[e/x]||, \boldsymbol{\lambda} k.ck\langle e, bk\rangle \in \perp\!\!\!\perp \end{array}$$

and on the other hand if $e \in \tau^{\mathtt{t}}$ and $b \in ||A[e/x]||$ then:

$$\begin{array}{rcl} \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.[xy][ck'/x, e/y])k(bk) & = & \boldsymbol{\lambda} k.([xy][ck/x, e/y])(bk) \\ & = & \boldsymbol{\lambda} k.((\boldsymbol{\lambda} k'.x\langle y, k'\rangle)[ck/x, e/y])(bk) \\ & = & \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.ck\langle e, k'\rangle)(bk) \\ & = & \boldsymbol{\lambda} k.ck\langle e, bk\rangle \end{array}$$

therefore:

$$c \in |A \Rightarrow B| \quad \Leftrightarrow \quad \forall e \in \tau^{\mathtt{t}}, \forall b \in ||A[e/x]||, \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.[xy][ck'/x, e/y])k(bk) \in \perp\!\!\!\perp$$

that is:

$$c \in |A \Rightarrow B| \quad \Leftrightarrow \quad \forall e \in \tau^{\mathtt{t}}, \boldsymbol{\lambda} k'.[xy][ck'/x, e/y] \in |A[e/x]|$$

$\square$

Lemma A.2. *If A is such that $FV(A) = \{x^\tau\}$ and if for all $e \in \tau^{\mathtt{t}}$, $b \in |A[e/x]|$, then $\boldsymbol{\lambda} k.[\lambda x.y][bk/y] \in |\forall x^\tau A|$.*

Proof. From lemma A.1 we have:

$$\boldsymbol{\lambda} k.[\lambda x.y][bk/y] \in |\forall x^\tau A| \Leftrightarrow \forall e \in \tau^{\mathtt{t}}, \boldsymbol{\lambda} k.[xy'][(\boldsymbol{\lambda} k'.[\lambda x.y][bk'/y])k/x, e/y'] \in |A[e/x]|$$

but since:

$$\begin{array}{rcl} \boldsymbol{\lambda} k.[xy'][(\boldsymbol{\lambda} k'.[\lambda x.y][bk'/y])k/x, e/y'] & = & \boldsymbol{\lambda} k.[xy'][[\lambda x.y][bk/y]/x, e/y'] \\ & = & \boldsymbol{\lambda} k.[(\lambda x.y)y'][bk/y, e/y'] \\ & = & \boldsymbol{\lambda} k.[y][bk/y, e/y'] \\ & = & \boldsymbol{\lambda} k.y[bk/y, e/y'] \\ & = & \boldsymbol{\lambda} k.bk \\ & = & b \end{array}$$

We have:

$$\boldsymbol{\lambda} k.[\lambda x.y][bk/y] \in |\forall x^\tau A| \Leftrightarrow \forall e \in \tau^{\mathtt{t}}, b \in |A[e/x]|$$

from which we conclude since $b \in |A[e/x]|$. $\square$

Lemma A.3. *If $c \in [\![\iota]\!] \to [A^* \to B^*]$, then:*

$$c \in |A \Rightarrow B| \Leftrightarrow \forall a \in |A|, \boldsymbol{\lambda} k.[xy][ck/x, ak/y] \in |B|$$

Proof. One one hand:

$$\begin{array}{rcl} c \in |A \Rightarrow B| & \Leftrightarrow & \forall d \in ||A \Rightarrow B||, \boldsymbol{\lambda} k.ck(dk) \in \perp\!\!\!\perp \\ & \Leftrightarrow & \forall a \in |A|, \forall b \in ||B||, \boldsymbol{\lambda} k.ck((\boldsymbol{\lambda} k'.\langle ak', bk'\rangle)k) \in \perp\!\!\!\perp \\ & \Leftrightarrow & \forall a \in |A|, \forall b \in ||B||, \boldsymbol{\lambda} k.ck\langle ak, bk\rangle \in \perp\!\!\!\perp \end{array}$$

and on the other hand if $a \in |A|$ and $b \in ||B||$ then:

$$\begin{array}{rcl} \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.[xy][ck'/x, ak'/y])k(bk) & = & \boldsymbol{\lambda} k.([xy][ck/x, ak/y])(bk) \\ & = & \boldsymbol{\lambda} k.((\boldsymbol{\lambda} k'.x\langle y, k'\rangle)[ck/x, ak/y])(bk) \\ & = & \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.ck\langle ak, k'\rangle)(bk) \\ & = & \boldsymbol{\lambda} k.ck\langle ak, bk\rangle \end{array}$$

therefore:

$$c \in |A \Rightarrow B| \quad \Leftrightarrow \quad \forall a \in |A|, \forall b \in ||B||, \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.[xy][ck'/x, ak'/y])k(bk) \in \perp\!\!\!\perp$$

that is:

$$c \in |A \Rightarrow B| \quad \Leftrightarrow \quad \forall a \in |A|, \boldsymbol{\lambda} k'.[xy][ck'/x, ak'/y] \in |B|$$

□

Lemma A.4. *If $|A| \subseteq |B|$, then $\boldsymbol{\lambda} k.[\lambda x.x] \in |A \Rightarrow B|$*

Proof. From lemma A.3 we have:

$$\boldsymbol{\lambda} k.[\lambda x.x] \in |A \Rightarrow B| \Leftrightarrow \forall a \in |A|, \boldsymbol{\lambda} k.[xy][(\boldsymbol{\lambda} k'.[\lambda x.x])k/x, ak/y] \in |B|$$

but since:

$$\begin{array}{rcl} \boldsymbol{\lambda} k.[xy][(\boldsymbol{\lambda} k'.[\lambda x.x])k/x, ak/y] & = & \boldsymbol{\lambda} k.[xy][[\lambda x.x]/x, ak/y] \\ & = & \boldsymbol{\lambda} k.[(\lambda x.x)y][ak/y] \\ & = & \boldsymbol{\lambda} k.[y][ak/y] \\ & = & \boldsymbol{\lambda} k.y[ak/y] \\ & = & \boldsymbol{\lambda} k.ak \\ & = & a \end{array}$$

We have:

$$\boldsymbol{\lambda} k.[\lambda x.x] \in |A \Rightarrow B| \Leftrightarrow \forall a \in |A|, a \in |B|$$

from which we conclude using $|A| \subseteq |B|$. □

Lemma A.5. *If $c \in [\![\iota]\!] \to [A^* \times B^*]$, then:*

$$\begin{array}{rl} c \in |A \wedge B| \Leftrightarrow & \forall a \in |A|, \boldsymbol{\lambda} k.[\mathsf{p}_1(x)][ck/x] \in |A| \\ & \textit{and } \forall b \in |B|, \boldsymbol{\lambda} k.[\mathsf{p}_2(x)][ck/x] \in |B| \end{array}$$

Proof. One one hand:

$$\begin{array}{rcl} c \in |A \wedge B| & \Leftrightarrow & \forall d \in ||A \wedge B||, \boldsymbol{\lambda} k.ck(dk) \in \perp\!\!\!\perp \\ & \Leftrightarrow & \forall a \in ||A||, \boldsymbol{\lambda} k.ck((\boldsymbol{\lambda} k'.\mathsf{in}_1(ak'))k) \in \perp\!\!\!\perp \\ & & \quad \text{and } \forall b \in ||B||, \boldsymbol{\lambda} k.ck((\boldsymbol{\lambda} k'.\mathsf{in}_2(bk'))k) \in \perp\!\!\!\perp \\ & \Leftrightarrow & \forall a \in ||A||, \boldsymbol{\lambda} k.ck(\mathsf{in}_1(ak)) \in \perp\!\!\!\perp \\ & & \quad \text{and } \forall b \in ||B||, \boldsymbol{\lambda} k.ck(\mathsf{in}_2(bk)) \in \perp\!\!\!\perp \end{array}$$

and on the other hand if $a \in ||A||$ then:

$$\begin{array}{rcl} \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.[\mathsf{p}_1(x)][ck'/x])k(ak) & = & \boldsymbol{\lambda} k.([\mathsf{p}_1(x)][ck/x])(ak) \\ & = & \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.x(\mathsf{in}_1 k'))[ck/x])(ak) \\ & = & \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.ck(\mathsf{in}_1 k'))(ak) \\ & = & \boldsymbol{\lambda} k.ck(\mathsf{in}_1(ak)) \end{array}$$

and similarly, if $b \in ||B||$ then:

$$\boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.[\mathsf{p}_2(x)][ck'/x])k(bk) = \boldsymbol{\lambda} k.ck(\mathsf{in}_2(2k))$$

therefore:

$$\begin{array}{rcl} c \in |A \wedge B| & \Leftrightarrow & \forall a \in ||A||, \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.[\mathsf{p}_1(x)][ck'/x])k(ak) \in \perp\!\!\!\perp \\ & & \quad \text{and } \forall B \in ||B||, \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.[\mathsf{p}_2(x)][ck'/x])k(bk) \in \perp\!\!\!\perp \end{array}$$

that is:

$$c \in |A \wedge B| \quad \Leftrightarrow \quad \boldsymbol{\lambda} k'.[\mathsf{p}_1(x)][ck'/x] \in |A| \text{ and } \boldsymbol{\lambda} k'.[\mathsf{p}_2(x)][ck'/x] \in |B|$$

□

Lemma A.6. *We have* $\boldsymbol{\lambda} k.[\lambda x.x] \in |e =_\tau e|$. *Moreover,*

$$\begin{array}{cc} \boldsymbol{\lambda} k.[\lambda x.x] \in |\mathsf{k}\, e\, f \ =_\tau e| & \boldsymbol{\lambda} k.[\lambda x.x] \in |\mathsf{s}\, e\, f\, g =_\tau eg(fg)| \\ \boldsymbol{\lambda} k.[\lambda x.x] \in |\mathsf{Rec}\, e\, f\, 0 =_\tau e| & \boldsymbol{\lambda} k.[\lambda x.x] \in |\mathsf{Rec}\, e\, f\, (\mathsf{S}\, g) =_\tau fg(\mathsf{Rec}\, e\, f\, g)| \end{array}$$

where in each case, individuals $e, f, g \in \tau^{\mathtt{t}}$ *have the appropriate types* τ.

Proof. First remark that if $e, f \in \tau^{\mathtt{t}}$ are such that $e = f$, then $||e \neq_\tau f|| = \{\boldsymbol{\lambda} k.k\} = ||\bot||$, so $|e \neq_\tau f| = |\bot|$. Since $|e =_\tau f|$ is $|e \neq_\tau f \Rightarrow \bot|$, we have $\boldsymbol{\lambda} k.[\boldsymbol{\lambda} x.x] \in |e =_\tau f|$ by lemma A.4. All the results of the lemma are then instances of this, using the interpretation of the terms in the model and its adequacy. For example if $e \in \tau^{\mathtt{t}}$ and $f \in \sigma^{\mathtt{t}}$, then:

$$[\mathsf{k}^\dagger xy][e/x, f/y] = [(\lambda xy.x)xy][e/x, f/y] = [x][e/x, f/y] = x[e/x, f/y] = e$$

□

Lemma A.7. $\boldsymbol{\lambda} k.[\lambda x.\lambda y.yx] \in |A[e/z] \ \Rightarrow\ \neg A[f/z] \ \Rightarrow\ e \neq_\tau f|$ *for* $e, f \in \tau^{\mathtt{t}}$.

Proof. From lemma A.3 we have:

$$\begin{aligned}&\boldsymbol{\lambda} k.[\lambda x.\lambda y.yx] \in |A[e/z] \Rightarrow \neg A[f/z] \Rightarrow e \neq_\tau f| \\ &\quad\Leftrightarrow \forall a \in |A[e/z]|, \boldsymbol{\lambda} k.[x'y'][[\lambda x.\lambda y.yx]/x', ak/y'] \in |\neg A[f/z] \Rightarrow e \neq_\tau f|\end{aligned}$$

but:

$$\begin{aligned}\boldsymbol{\lambda} k.[x'y'][[\lambda x.\lambda y.yx]/x', ak/y'] &= \boldsymbol{\lambda} k.[(\lambda x.\lambda y.yx)y'][ak/y'] \\ &= \boldsymbol{\lambda} k.[\lambda y.yy'][ak/y'] \\ &= \boldsymbol{\lambda} k.[\lambda y.yx][ak/x]\end{aligned}$$

Then we have:

$$\begin{aligned}&\boldsymbol{\lambda} k.[\lambda x.\lambda y.yx] \in |A[e/z] \Rightarrow \neg A[f/z] \Rightarrow e \neq_\tau f| \\ &\quad\Leftrightarrow \forall a \in |A[e/z]|, \boldsymbol{\lambda} k.[\lambda y.yx][ak/x] \in |\neg A[f/z] \Rightarrow e \neq_\tau f| \\ &\quad\Leftrightarrow \forall a \in |A[e/z]|, \forall b \in |\neg A[f/z]|, \boldsymbol{\lambda} k.[x'y'][[\lambda y.yx][ak/x]/x', bk/y'] \in |e \neq_\tau f|\end{aligned}$$

but:

$$\begin{aligned}\boldsymbol{\lambda} k.[x'y'][[\lambda y.yx][ak/x]/x', bk/y'] &= \boldsymbol{\lambda} k.[(\lambda y.yx)y'][ak/x, bk/y'] \\ &= \boldsymbol{\lambda} k.[y'x][ak/x, bk/y'] \\ &= \boldsymbol{\lambda} k.[yx][ak/x, bk/y] \\ &= \boldsymbol{\lambda} k.[xy][bk/x, ak/y]\end{aligned}$$

therefore:

$$\begin{aligned}&\boldsymbol{\lambda} k.[\lambda x.\lambda y.yx] \in |A[e/z] \Rightarrow \neg A[f/z] \Rightarrow e \neq_\tau f| \\ &\quad\Leftrightarrow \forall a \in |A[e/z]|, \forall b \in |\neg A[f/z]|, \boldsymbol{\lambda} k.[xy][bk/x, ak/y] \in |e \neq_\tau f|\end{aligned}$$

If $e \neq f$, then $|e \neq_\tau f| = [\![\iota]\!] \to [\iota]$ and we are done. If $e = f$, then $|\neg A[f/z]| = |\neg A[e/z]| = |A[e/z] \Rightarrow \bot|$, so by lemma A.3, if $a \in |A[e/z]|$ and $b \in |\neg A[f/z]|$, then $\boldsymbol{\lambda} k.[xy][bk/x, ak/y] \in |\bot| = |e \neq_\tau f|$ (since $e = f$). □

Lemma A.8. *If $e, f \in \mathcal{I}_0^\tau$ are closed first order terms with parameters such that $[e^\dagger] = [f^\dagger]$, then for any formula A with $FV(A) = \{x^\tau\}$, we have $||A[e/x]|| = ||A[f/x]||$.*

Proof. We first prove that for any $g \in \mathcal{I}^\sigma$ with parameters and $\mathrm{FV}(g) = \{x^\tau\}$, we have $[g[e/x]^\dagger] = [g[f/x]^\dagger]$ by induction on the structure of the term g. Then the proof goes by induction on the structure of the formula A. □

Lemma A.9. (i) *For all $n \in \mathbb{N}$ and all $e \in \iota^{\mathfrak{t}}$, we have $\boldsymbol{\lambda} k.[\overline{n}] \in |\mathsf{S}e \neq_\iota \mathsf{0}|$.*
(ii) $\boldsymbol{\lambda} k.[\lambda x.\lambda y.\mathsf{rec}(x, y)] \in |A[\mathsf{0}/x] \Rightarrow \forall x^\iota(A \Rightarrow A[\mathsf{S}x/x]) \Rightarrow \forall x^\iota A|$.

Proof. (i) Since for any $e \in \iota^{\mathfrak{t}}$ there is some $m \in \mathbb{N}$ such that $e = [\overline{m}]$, we have:

$$\begin{aligned}[\mathsf{S}^\dagger x][e/x] &= [\mathsf{S}^\dagger x][[\overline{n}]/x] \\ &= [\mathsf{S}^\dagger \overline{n}] \\ &= [\mathsf{succ}\, \overline{n}] \\ &= [\overline{n+1}]\end{aligned}$$

so $[\mathsf{S}^\dagger x][e/x] = [\overline{n+1}] \neq [\overline{0}] = [\mathsf{0}^\dagger]$. Therefore $||\mathsf{S}e \neq_\iota \mathsf{0}|| = \emptyset$, so $|\mathsf{S}e \neq_\iota \mathsf{0}| = [\![\iota]\!] \to [\iota]$, and therefore for any $n \in \mathbb{N}$ we have $\boldsymbol{\lambda} k.[\overline{n}] \in |\mathsf{S}e \neq_\iota \mathsf{0}|$.

(ii) We have by lemmas A.3 and A.1:

$$
\begin{aligned}
&\boldsymbol{\lambda} k.[\lambda x.\lambda y.\mathsf{rec}(x,y)] \in |A[\mathsf{0}/x] \Rightarrow \forall x^\iota (A \Rightarrow A[\mathsf{S}x/x]) \Rightarrow \forall x^\iota A| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \\
&\qquad \boldsymbol{\lambda} k.[xy][(\boldsymbol{\lambda} k'.[\lambda x'.\lambda y'.\mathsf{rec}(x',y')])k/x, a_0k/y] \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x]) \Rightarrow \forall x^\iota A| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \\
&\qquad \boldsymbol{\lambda} k.[xy][[\lambda x'.\lambda y'.\mathsf{rec}(x',y')]/x, a_0k/y] \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x]) \Rightarrow \forall x^\iota A| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \\
&\qquad \boldsymbol{\lambda} k.[(\lambda x'.\lambda y'.\mathsf{rec}(x',y'))y][a_0k/y] \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x]) \Rightarrow \forall x^\iota A| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \\
&\qquad \boldsymbol{\lambda} k.[\lambda y'.\mathsf{rec}(y,y')][a_0k/y] \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x]) \Rightarrow \forall x^\iota A| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \\
&\qquad \boldsymbol{\lambda} k.[\lambda y.\mathsf{rec}(x,y)][a_0k/x] \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x]) \Rightarrow \forall x^\iota A| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \forall a_1 \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x])|, \\
&\qquad \boldsymbol{\lambda} k.[xy][(\boldsymbol{\lambda} k'.[\lambda y'.\mathsf{rec}(x,y')][a_0k/x])k/x, a_1k/y] \in |\forall x^\iota A| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \forall a_1 \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x])|, \\
&\qquad \boldsymbol{\lambda} k.[xy][[\lambda y'.\mathsf{rec}(x,y')][a_0k/x]/x, a_1k/y] \in |\forall x^\iota A| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \forall a_1 \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x])|, \\
&\qquad \boldsymbol{\lambda} k.[(\lambda y'.\mathsf{rec}(x,y'))y][a_0k/x, a_1k/y] \in |\forall x^\iota A| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \forall a_1 \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x])|, \\
&\qquad \boldsymbol{\lambda} k.[\mathsf{rec}(x,y)][a_0k/x, a_1k/y] \in |\forall x^\iota A| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \forall a_1 \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x])|, \forall e \in \iota^{\mathsf{t}}, \\
&\qquad \boldsymbol{\lambda} k.[xz][(\boldsymbol{\lambda} k'.[\mathsf{rec}(x,y)][a_0k'/x, a_1k'/y])k/x, e/z] \in |A[e/x]| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \forall a_1 \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x])|, \forall e \in \iota^{\mathsf{t}}, \\
&\qquad \boldsymbol{\lambda} k.[xz][[\mathsf{rec}(x,y)][a_0k/x, a_1k/y]/x, e/z] \in |A[e/x]| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \forall a_1 \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x])|, \forall a \in \iota^{\mathsf{t}}, \\
&\qquad \boldsymbol{\lambda} k.[\mathsf{rec}(x,y)z][a_0k/x, a_1k/y, e/z] \in |A[e/x]| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \forall a_1 \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x])|, \forall n \in \mathbb{N}, \\
&\qquad \boldsymbol{\lambda} k.[\mathsf{rec}(x,y)z][a_0k/x, a_1k/y, [\overline{n}]/z] \in |A[[\overline{n}]/x]| \\
&\quad \Leftrightarrow \forall a_0 \in |A[\mathsf{0}/x]|, \forall a_1 \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x])|, \forall n \in \mathbb{N}, \\
&\qquad \boldsymbol{\lambda} k.[\mathsf{rec}(x,y)\overline{n}][a_0k/x, a_1k/y] \in |A[[\overline{n}]/x]|
\end{aligned}
$$

Since for all $e \in \iota^{\mathsf{t}}$ there is some $n \in \mathbb{N}$ such that $e = [\overline{n}]$, and therefore $|A[e/x]| = |A[[\overline{n}]/x]|$. Let us fix $a_0 \in |A[\mathsf{0}/x]|$ and $a_1 \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x])|$. We have:

$$
\begin{aligned}
&a_1 \in |\forall x^\iota (A \Rightarrow A[\mathsf{S}x/x])| \\
&\quad \Leftrightarrow \forall e \in \iota^{\mathsf{t}}, \boldsymbol{\lambda} k.[xy][a_1k/x, e/y] \in |A[e/x] \Rightarrow A[\mathsf{S}e/x]| \\
&\quad \Leftrightarrow \forall n \in \mathbb{N}, \boldsymbol{\lambda} k.[xy][a_1k/x, [\overline{n}]/y] \in |A[[\overline{n}]/x] \Rightarrow A[\mathsf{S}[\overline{n}]/x]| \\
&\quad \Leftrightarrow \forall n \in \mathbb{N}, \boldsymbol{\lambda} k.[x\overline{n}][a_1k/x] \in |A[[\overline{n}]/x] \Rightarrow A[\mathsf{S}[\overline{n}]/x]| \\
&\quad \Leftrightarrow \forall n \in \mathbb{N}, \forall a_2 \in |A[[\overline{n}]/x]|, \boldsymbol{\lambda} k.[xy][(\boldsymbol{\lambda} k'.[x\overline{n}][a_1k'/x])k/x, a_2k/y] \in |A[\mathsf{S}[\overline{n}]/x]| \\
&\quad \Leftrightarrow \forall n \in \mathbb{N}, \forall a_2 \in |A[[\overline{n}]/x]|, \boldsymbol{\lambda} k.[xy][[x\overline{n}][a_1k/x]/x, a_2k/y] \in |A[\mathsf{S}[\overline{n}]/x]| \\
&\quad \Leftrightarrow \forall n \in \mathbb{N}, \forall a_2 \in |A[[\overline{n}]/x]|, \boldsymbol{\lambda} k.[x\overline{n}y][a_1k/x, a_2k/y] \in |A[\mathsf{S}[\overline{n}]/x]| \\
&\quad \Leftrightarrow \forall n \in \mathbb{N}, \forall a_2 \in |A[[\overline{n}]/x]|, \boldsymbol{\lambda} k.[x\overline{n}y][a_1k/x, a_2k/y] \in |A[[\overline{n+1}]/x]|
\end{aligned}
$$

by lemma A.8, since $[(\mathsf{S}[\overline{n}])^\dagger] = [(\mathsf{S}x)^\dagger][[\overline{n}]/x] = [\mathsf{succ}\, x][[\overline{n}]/x] = [\mathsf{succ}\, \overline{n}] = [\overline{n+1}]$. We now prove by induction on $n \in \mathbb{N}$ that:

$$\boldsymbol{\lambda} k.[\mathsf{rec}(x,y)\overline{n}][a_0 k/x, a_1 k/y] \in |A[[\overline{n}]/x]|$$

– $n = 0$:

$$\begin{array}{rcl} \boldsymbol{\lambda} k.[\mathsf{rec}(x,y)\overline{0}][a_0 k/x, a_1 k/y] & = & \boldsymbol{\lambda} k.[x][a_0 k/x, a_1 k/y] \\ & = & \boldsymbol{\lambda} k.x[a_0 k/x, a_1 k/y] \\ & = & \boldsymbol{\lambda} k.a_0 k \\ & = & a_0 \in |A[\mathsf{0}/x]| = |A[[\mathsf{0}^\dagger]/x]| = |A[[\overline{0}]/x]| \end{array}$$

– $n = m+1$: the induction hypothesis gives:

$$\boldsymbol{\lambda} k.[\mathsf{rec}(x,y)\overline{m}][a_0 k/x, a_1 k/y] \in |A[[\overline{m}]/x]|$$

and a_1 is such that:

$$\forall a_2 \in |A[[\overline{m}]/x]|, \boldsymbol{\lambda} k.[x\overline{m}y][a_1 k/x, a_2 k/y] \in |A[[\overline{m+1}]/x]|$$

therefore:

$$\boldsymbol{\lambda} k.[x\overline{m}y][a_1 k/x, (\boldsymbol{\lambda} k'.[\mathsf{rec}(x',y')\overline{m}][a_0 k'/x', a_1 k'/y'])k/y] \in |A[[\overline{m+1}]/x]|$$

but:

$$\begin{array}{l} \boldsymbol{\lambda} k.[x\overline{m}y][a_1 k/x, (\boldsymbol{\lambda} k'.[\mathsf{rec}(x',y')\overline{m}][a_0 k'/x', a_1 k'/y'])k/y] \\ \quad = \boldsymbol{\lambda} k.[x\overline{m}y][a_1 k/x, [\mathsf{rec}(x',y')\overline{m}][a_0 k/x', a_1 k/y']/y] \\ \quad = \boldsymbol{\lambda} k.[x\overline{m}(\mathsf{rec}(x',y')\overline{m})][a_1 k/x, a_0 k/x', a_1 k/y'] \\ \quad = \boldsymbol{\lambda} k.[y'\overline{m}(\mathsf{rec}(x',y')\overline{m})][a_0 k/x', a_1 k/y'] \\ \quad = \boldsymbol{\lambda} k.[y\overline{m}(\mathsf{rec}(x,y)\overline{m})][a_0 k/x, a_1 k/y] \\ \quad = \boldsymbol{\lambda} k.[\mathsf{rec}(x,y)\overline{m+1})][a_0 k/x, a_1 k/y] \end{array}$$

so we conclude:

$$\boldsymbol{\lambda} k.[\mathsf{rec}(x,y)\overline{n})][a_0 k/x, a_1 k/y] \in |A[[\overline{n}]/x]|$$

□

We associate to each axiom A a closed typed $\lambda\mu$-term ξ_A as follows:

$$\begin{array}{rcl} \xi_{\forall x^\tau (x =_\tau x)} & = & \lambda xy.y : \tau \to \iota \to \iota \\ \xi_{\forall x^\tau \forall y^\tau (A[x] \Rightarrow \neg A[y] \Rightarrow x \neq_\tau y)} & = & \lambda xyuv.vu : \tau \to \tau \to A^* \to (A^* \to \iota) \to \iota \\ \xi_{\forall x^\tau \forall y^\sigma (\mathsf{k}xy =_\tau x)} & = & \lambda xyu.u : \tau \to \sigma \to \iota \to \iota) \to \iota \\ \xi_{\forall x^\tau \forall y^\sigma \forall z^\nu (\mathsf{s}xyz =_\nu xz(yz))} & = & \lambda xyzu.u : \tau \to \sigma \to \nu \to \iota \to \iota \\ \xi_{\forall x^\iota (Sx \neq_\iota \mathsf{0})} & = & \lambda x.\overline{0} : \iota \to \iota \\ \xi_{\forall x^\tau \forall y^{\iota\to\tau\to\tau} (\mathsf{Rec}xy\mathsf{0} =_\tau x)} & = & \lambda xyu.u : \tau \to (\iota \to \tau \to \tau) \to \iota \to \iota \\ \xi_{\forall x^\tau \forall y^{\iota\to\tau\to\tau} \forall z^\iota (\mathsf{Rec}xy\mathsf{S}z =_\tau yz(\mathsf{Rec}xyz))} & = & \lambda xyzu.u : \tau \to (\iota \to \tau \to \tau) \to \iota \to \iota \to \iota \\ \xi_{A[\mathsf{0}] \Rightarrow \forall y^\tau (A[y] \Rightarrow A[\mathsf{S}y]) \Rightarrow \forall x^\iota A[x]} & = & \lambda uv.\mathsf{rec}(u,v) : A^* \to (\iota \to A^* \to A^*) \to \iota \to A^* \end{array}$$

We extend the translation $(\_)^*$ to contexts in the obvious way: $(A_1, \ldots, A_n)^*$ is translated to $x_1 : {A_1}^*, \ldots, x_n : {A_n}^*$.

We translate each derivation $\dfrac{\varepsilon}{\Gamma \vdash A \mid \Delta}$ to a typing derivation of $\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \varepsilon^* : A^* \mid \Delta^*$ and each derivation $\dfrac{\varepsilon}{(\Gamma \vdash \Delta)}$ to a typing derivation of $\varepsilon^* : (\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \Delta^*)$, where the free variables x^τ of Γ, A, Δ occur as $x : \tau$ in $\boldsymbol{x} : \boldsymbol{\tau}$:

$$\left(\frac{}{\Gamma, A \vdash A \mid \Delta}\right)^* = \frac{}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^*, x : A^* \vdash x : A^* \mid \Delta^*} \quad \text{where } \boldsymbol{x} = \mathrm{FV}(\Gamma, A, \Delta)$$

$$\left(\frac{}{\Gamma \vdash A \mid \Delta}\ (A \text{ axiom})\right)^* = \frac{\vdots}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \xi_A : A^* \mid \Delta^*} \quad \text{where } \boldsymbol{x} = \mathrm{FV}(\Gamma, \Delta)$$

$$\left(\frac{\dfrac{\varepsilon}{\Gamma \vdash \bot \mid \Delta}}{\Gamma \vdash a^\tau \neq_\tau b^\tau \mid \Delta}\right)^* = \frac{\vdots}{\boldsymbol{x} : \boldsymbol{\tau}, \boldsymbol{y} : \boldsymbol{\sigma}, \Gamma^* \vdash \varepsilon^* : \iota \mid \Delta^*} \quad \text{where } \boldsymbol{y} = \mathrm{FV}(a^\tau, b^\tau)$$

$$\left(\frac{\dfrac{\varepsilon}{\Gamma, A \vdash B \mid \Delta}}{\Gamma \vdash A \Rightarrow B \mid \Delta}\right)^* = \frac{\dfrac{\vdots}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^*, y : A^* \vdash \varepsilon^* : B^* \mid \Delta^*}}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \lambda y.\varepsilon^* : A^* \to B^* \mid \Delta^*}$$

$$\left(\frac{\dfrac{\varepsilon}{\Gamma \vdash A \Rightarrow B \mid \Delta} \qquad \dfrac{\zeta}{\Gamma \vdash A \mid \Delta}}{\Gamma \vdash B \mid \Delta}\right)^* = \frac{\dfrac{\vdots}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \varepsilon^* : A^* \to B^* \mid \Delta^*} \qquad \dfrac{\vdots}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \zeta^* : A^* \mid \Delta^*}}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \varepsilon^* \zeta^* : B^* \mid \Delta^*}$$

$$\left(\frac{\dfrac{\varepsilon}{\Gamma \vdash A \mid \Delta} \qquad \dfrac{\zeta}{\Gamma \vdash B \mid \Delta}}{\Gamma \vdash A \wedge B \mid \Delta}\right)^* = \frac{\dfrac{\vdots}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \varepsilon^* : A^* \mid \Delta^*} \qquad \dfrac{\vdots}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \zeta^* : B^* \mid \Delta^*}}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \langle \varepsilon^*, \zeta^* \rangle : A^* \times B^* \mid \Delta^*}$$

$$\left(\frac{\dfrac{\varepsilon}{\Gamma \vdash A_1 \wedge A_2 \mid \Delta}}{\Gamma \vdash A_i \mid \Delta}\ (i = 1, 2)\right)^* = \frac{\dfrac{\vdots}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \varepsilon^* : A^* \times B^* \mid \Delta^*}}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \mathsf{p}_i(\varepsilon^*) : {A_i}^* \mid \Delta^*}$$

$$\left(\frac{\dfrac{\varepsilon}{\Gamma \vdash A \mid \Delta}}{\Gamma \vdash \forall x^\tau A \mid \Delta}\ (x \notin \mathrm{FV}(\Gamma, \Delta))\right)^* = \frac{\dfrac{\vdots}{\boldsymbol{x} : \boldsymbol{\tau}, x : \tau, \Gamma^* \vdash \varepsilon^* : A^* \mid \Delta^*}}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \lambda x.\varepsilon^* : \tau \to A^* \mid \Delta^*}$$

$$\left(\frac{\dfrac{\varepsilon}{\Gamma \vdash \forall x^\tau A \mid \Delta}}{\Gamma \vdash A[a^\tau / x] \mid \Delta}\right)^* = \frac{\dfrac{\vdots}{\boldsymbol{x} : \boldsymbol{\tau}, \boldsymbol{y} : \boldsymbol{\sigma}, \Gamma^* \vdash \varepsilon^* : \tau \to A^* \mid \Delta^*}}{\boldsymbol{x} : \boldsymbol{\tau}, \boldsymbol{y} : \boldsymbol{\sigma}, \Gamma^* \vdash \varepsilon^* a^{\tau\dagger} : A^* \mid \Delta^*} \quad \text{where } \boldsymbol{y} = \mathrm{FV}(a^\tau)$$

$$\left(\frac{\dfrac{\varepsilon}{\Gamma \vdash A \mid \Delta, A}}{(\Gamma \vdash \Delta, A)}\right)^* = \frac{\dfrac{\vdots}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \varepsilon^* : A^* \mid \Delta^*, \alpha : A^*}}{[\alpha]\varepsilon^* : (\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \Delta^*, \alpha : A^*)}$$

$$\left(\frac{\dfrac{\varepsilon}{(\Gamma \vdash \Delta, A)}}{\Gamma \vdash A \mid \Delta}\right)^* = \frac{\dfrac{\vdots}{\varepsilon^* : (\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \Delta^*, \alpha : A^*)}}{\boldsymbol{x} : \boldsymbol{\tau}, \Gamma^* \vdash \mu\alpha.\varepsilon^* : A^* \mid \Delta^*}$$

Theorem A.10. *Let* $\dfrac{\varepsilon}{\Gamma \vdash A \mid \Delta}$ *in* PA^ω, *with* $FV(\Gamma, A, \Delta) \subseteq \{x_1^{\tau_1}, \ldots, x_k^{\tau_k}\}$, $\Gamma = A_1, \ldots, A_n$ *and* $\Delta = B_1, \ldots, B_m$.

Then

$$x_1 : \tau_1, \ldots, x_k : \tau_k, y_1 : {A_1}^*, \ldots, y_n : {A_n}^* \vdash \varepsilon^* : A^* \mid \alpha_1 : {B_1}^*, \ldots, \alpha_m : {B_m}^*$$

is such that for all $c_1 \in {\tau_1}^{\mathtt{t}}, \ldots, c_k \in {\tau_k}^{\mathtt{t}}$, *all* $a_1 \in |A_1[\boldsymbol{c}/\boldsymbol{x}]|, \ldots, a_n \in |A_n[\boldsymbol{c}/\boldsymbol{x}]|$, *and all* $b_1 \in ||B_1[\boldsymbol{c}/\boldsymbol{x}]||, \ldots, b_m \in ||B_m[\boldsymbol{c}/\boldsymbol{x}]||$, *we have*

$$\boldsymbol{\lambda}k.[\varepsilon^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |A[\boldsymbol{c}/\boldsymbol{x}]|$$

Proof. By induction on the structure of the derivation:

- $\varepsilon = \dfrac{}{\Gamma, A \vdash A \mid \Delta}$:

$$\boldsymbol{\lambda}k.[\varepsilon^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, ak/x, \boldsymbol{b}k/\boldsymbol{\alpha}] = \boldsymbol{\lambda}k.[x][ak/x] = \boldsymbol{\lambda}k.ak = a \in |A[\boldsymbol{c}/\boldsymbol{x}]|$$

- $\varepsilon = \dfrac{}{\Gamma \vdash A \mid \Delta}$ (A axiom): then by lemmas A.6 A.7 A.9 we have $\boldsymbol{\lambda}k.[\varepsilon^*] = \boldsymbol{\lambda}k.[\xi_A] \in |A| = |A[\boldsymbol{c}/\boldsymbol{x}]|$ since A is closed. Moreover, since ξ_A is closed,

$$\boldsymbol{\lambda}k.[\varepsilon^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] = \boldsymbol{\lambda}k.[\varepsilon^*] \in |A[\boldsymbol{c}/\boldsymbol{x}]|$$

- $\varepsilon = \dfrac{\dfrac{\varepsilon'}{\Gamma \vdash \bot \mid \Delta}}{\Gamma \vdash a^\tau \neq_\tau b^\tau \mid \Delta}$: this comes from the fact that since $||a \neq_\tau b|| \subseteq \{\boldsymbol{\lambda}k.k\} = ||\perp\!\!\!\perp||$, we have $|\perp\!\!\!\perp| \subseteq |a \neq_\tau b|$
- $\varepsilon = \dfrac{\dfrac{\varepsilon'}{\Gamma, A \vdash B \mid \Delta}}{\Gamma \vdash A \Rightarrow B \mid \Delta}$: the induction hypothesis gives for any $c_1 \in {\tau_1}^{\mathtt{t}}, \ldots, c_k \in {\tau_k}^{\mathtt{t}}, a_1 \in |A_1[\boldsymbol{c}/\boldsymbol{x}]|, \ldots, a_n \in |A_n[\boldsymbol{c}/\boldsymbol{x}]|, b_1 \in ||B_1[\boldsymbol{c}/\boldsymbol{x}]||, \ldots, b_m \in ||B_m[\boldsymbol{c}/\boldsymbol{x}]||$:

$$\forall a \in |A[\boldsymbol{c}/\boldsymbol{x}]|, \boldsymbol{\lambda}k.[{\varepsilon'}^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, ak/y, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |B[\boldsymbol{c}/\boldsymbol{x}]|$$

  using lemma A.3, in order to prove:

$$\boldsymbol{\lambda}k.[\lambda y.{\varepsilon'}^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |A[\boldsymbol{c}/\boldsymbol{x}] \Rightarrow B[\boldsymbol{c}/\boldsymbol{x}]|$$

it suffices to prove:

$$\forall a \in |A[\boldsymbol{c}/\boldsymbol{x}]|, \boldsymbol{\lambda} k.[xy][(\boldsymbol{\lambda} k'.[\lambda y.\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/x, ak/y] \in |B[\boldsymbol{c}/\boldsymbol{x}]|$$

but if $a \in |A[\boldsymbol{c}/\boldsymbol{x}]|$, then:

$$\begin{aligned}
&\boldsymbol{\lambda} k.[xy][(\boldsymbol{\lambda} k'.[\lambda y.\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}(k')/\boldsymbol{y}, \boldsymbol{b}(k')/\boldsymbol{\alpha}])k/x, ak/y] \\
&\quad = \boldsymbol{\lambda} k.[xy][[\lambda y.\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]/x, ak/y] \\
&\quad = \boldsymbol{\lambda} k.[(\lambda y.\varepsilon'^*)y][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, ak/y] \\
&\quad = \boldsymbol{\lambda} k.[(\lambda y.\varepsilon^*)y][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, ak/y] \\
&\quad = \boldsymbol{\lambda} k.[\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, ak/y]
\end{aligned}$$

so we conclude using the induction hypothesis.

– $\varepsilon = \dfrac{\dfrac{\varepsilon'}{\Gamma \vdash A \Rightarrow B \mid \Delta} \qquad \dfrac{\zeta}{\Gamma \vdash A \mid \Delta}}{\Gamma \vdash B \mid \Delta}$: let $c_1 \in {\tau_1}^{\mathtt{t}}, \ldots, c_k \in {\tau_k}^{\mathtt{t}}$, $a_1 \in |A_1[\boldsymbol{c}/\boldsymbol{x}]|$, $\ldots$, $a_n \in |A_n[\boldsymbol{c}/\boldsymbol{x}]|$, $b_1 \in ||B_1[\boldsymbol{c}/\boldsymbol{x}]||$, $\ldots$, $b_m \in ||B_m[\boldsymbol{c}/\boldsymbol{x}]||$. By induction hypothesis we have:

$$\boldsymbol{\lambda} k.[\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |A[\boldsymbol{c}/\boldsymbol{x}] \Rightarrow B[\boldsymbol{c}/\boldsymbol{x}]|$$

so by lemma A.3 we get:

$$\forall a \in |A[\boldsymbol{c}/\boldsymbol{x}]|, \boldsymbol{\lambda} k.[xy][(\boldsymbol{\lambda} k'.[\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/x, ak/y] \in |B[\boldsymbol{c}/\boldsymbol{x}]|$$

but:

$$\begin{aligned}
&\boldsymbol{\lambda} k.[xy][(\boldsymbol{\lambda} k'.[\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/x, ak/y] \\
&\quad = \boldsymbol{\lambda} k.[xy][[\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]/x, ak/y] \\
&\quad = \boldsymbol{\lambda} k.[\varepsilon'^* y][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, ak/y]
\end{aligned}$$

and since again by induction hypothesis we have:

$$\boldsymbol{\lambda} k.[\zeta^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |A[\boldsymbol{c}/\boldsymbol{x}]|$$

we get:

$$\boldsymbol{\lambda} k.[\varepsilon'^* y][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, (\boldsymbol{\lambda} k'.[\zeta^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/y] \in |B[\boldsymbol{c}/\boldsymbol{x}]|$$

so we can conclude since:

$$\begin{aligned}
&\boldsymbol{\lambda} k.[\varepsilon'^* y][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, (\boldsymbol{\lambda} k'.[\zeta^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/y] \\
&\quad = \boldsymbol{\lambda} k.[\varepsilon'^* y][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, [\zeta^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]/y] \\
&\quad = \boldsymbol{\lambda} k.[\varepsilon'^* \zeta^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]
\end{aligned}$$

– $\varepsilon = \dfrac{\dfrac{\varepsilon'}{\Gamma \vdash A \mid \Delta} \qquad \dfrac{\zeta}{\Gamma \vdash B \mid \Delta}}{\Gamma \vdash A \wedge B \mid \Delta}$: by induction hypothesis we have:

$$\boldsymbol{\lambda} k.[\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |A[\boldsymbol{c}/\boldsymbol{x}]|$$

and

$$\boldsymbol{\lambda}k.[\zeta^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |B[\boldsymbol{c}/\boldsymbol{x}]|$$

Using lemma A.5, we have:

$$\begin{aligned}
&\boldsymbol{\lambda}k.[\langle {\varepsilon'}^*, \zeta^*\rangle][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |A[\boldsymbol{c}/\boldsymbol{x}] \wedge B[\boldsymbol{c}/\boldsymbol{x}]| \\
&\quad \Leftrightarrow \boldsymbol{\lambda}k.[\mathsf{p}_1(x)][(\boldsymbol{\lambda}k'.[\langle {\varepsilon'}^*, \zeta^*\rangle][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/x] \in A[\boldsymbol{c}/\boldsymbol{x}]^* \\
&\qquad \text{and } \boldsymbol{\lambda}k.[\mathsf{p}_2(x)][(\boldsymbol{\lambda}k'.[\langle {\varepsilon'}^*, \zeta^*\rangle][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/x] \in B[\boldsymbol{c}/\boldsymbol{x}]^* \\
&\quad \Leftrightarrow \boldsymbol{\lambda}k.[\mathsf{p}_1(x)][[\langle {\varepsilon'}^*, \zeta^*\rangle][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]/x] \in A[\boldsymbol{c}/\boldsymbol{x}]^* \\
&\qquad \text{and } \boldsymbol{\lambda}k.[\mathsf{p}_2(x)][[\langle {\varepsilon'}^*, \zeta^*\rangle][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]/x] \in B[\boldsymbol{c}/\boldsymbol{x}]^* \\
&\quad \Leftrightarrow \boldsymbol{\lambda}k.[\mathsf{p}_1(\langle {\varepsilon'}^*, \zeta^*\rangle)][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in A[\boldsymbol{c}/\boldsymbol{x}]^* \\
&\qquad \text{and } \boldsymbol{\lambda}k.[\mathsf{p}_2(\langle {\varepsilon'}^*, \zeta^*\rangle)][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in B[\boldsymbol{c}/\boldsymbol{x}]^* \\
&\quad \Leftrightarrow \boldsymbol{\lambda}k.[{\varepsilon'}^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in A[\boldsymbol{c}/\boldsymbol{x}]^* \\
&\qquad \text{and } \boldsymbol{\lambda}k.[\zeta^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in B[\boldsymbol{c}/\boldsymbol{x}]^*
\end{aligned}$$

which is true by induction hypothesis.

- $\varepsilon = \dfrac{\dfrac{\varepsilon'}{\Gamma \vdash A_1 \wedge A_2 \mid \Delta}}{\Gamma \vdash A_i \mid \Delta}$ $(i = 1, 2)$: let us take $i = 1$ (the other case is similar). The induction hypothesis gives us:

$$\boldsymbol{\lambda}k.[{\varepsilon'}^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |A_1[\boldsymbol{c}/\boldsymbol{x}] \wedge A_2[\boldsymbol{c}/\boldsymbol{x}]|$$

so we get by lemma A.5:

$$\boldsymbol{\lambda}k.[\mathsf{p}_1(x)][(\boldsymbol{\lambda}k'.[{\varepsilon'}^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/x] \in |A_1[\boldsymbol{c}/\boldsymbol{x}]|$$

but:

$$\begin{aligned}
&\boldsymbol{\lambda}k.[\mathsf{p}_1(x)][(\boldsymbol{\lambda}k'.[{\varepsilon'}^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/x] \\
&\quad = \boldsymbol{\lambda}k.[\mathsf{p}_1(x)][[{\varepsilon'}^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]/x] \\
&\quad = \boldsymbol{\lambda}k.[\mathsf{p}_1({\varepsilon'}^*)][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]
\end{aligned}$$

so we can conclude.

- $\varepsilon = \dfrac{\dfrac{\varepsilon'}{\Gamma \vdash A \mid \Delta}}{\Gamma \vdash \forall x^\tau A \mid \Delta}$ $(x \notin \mathrm{FV}(\Gamma, \Delta))$: by induction hypothesis we have:

$$\forall c \in \tau^{\mathtt{t}}, \boldsymbol{\lambda}k.[{\varepsilon'}^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |A[c/x, \boldsymbol{c}/\boldsymbol{x}]|$$

First, since x is not free in $A_1, \ldots, A_n, B_1, \ldots, B_m$ we have $|A_i[c/x, \boldsymbol{c}/\boldsymbol{x}]| = |A_i[\boldsymbol{c}/\boldsymbol{x}]|$ and $||B_i[c/x, \boldsymbol{c}/\boldsymbol{x}]|| = ||B_i[\boldsymbol{c}/\boldsymbol{x}]||$, so we still have $a_i \in |A_i[\boldsymbol{c}/\boldsymbol{x}]|$ and $b_i \in ||B_i[\boldsymbol{c}/\boldsymbol{x}]||$. Using lemma A.1, we have:

$$\begin{aligned}
&\boldsymbol{\lambda}k.[\lambda x.{\varepsilon'}^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |\forall x^\tau A[\boldsymbol{c}/\boldsymbol{x}]| \\
&\quad \Leftrightarrow \forall c \in \tau^{\mathtt{t}}, \boldsymbol{\lambda}k.[zx][(\boldsymbol{\lambda}k'.[\lambda x.{\varepsilon'}^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/z, c/x] \in |A[\boldsymbol{c}/\boldsymbol{x}, c/x]| \\
&\quad \Leftrightarrow \forall c \in \tau^{\mathtt{t}}, \boldsymbol{\lambda}k.[zx][[\lambda x.{\varepsilon'}^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]/z, c/x] \in |A[\boldsymbol{c}/\boldsymbol{x}, c/x]| \\
&\quad \Leftrightarrow \forall c \in \tau^{\mathtt{t}}, \boldsymbol{\lambda}k.[(\lambda x.{\varepsilon'}^*)x][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, c/x] \in |A[\boldsymbol{c}/\boldsymbol{x}, c/x]|
\end{aligned}$$

and we conclude by induction hypothesis.

– $\varepsilon = \dfrac{\dfrac{\varepsilon'}{\Gamma \vdash \forall x^\tau A \mid \Delta}}{\Gamma \vdash A[a^\tau/x] \mid \Delta}$: by induction hypothesis we have:

$$\boldsymbol{\lambda} k.[\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |\forall x^\tau A[\boldsymbol{c}/\boldsymbol{x}]|$$

Using lemma A.1, we have:

$$\forall e \in \tau^{\mathtt{t}}, \boldsymbol{\lambda} k.[zx][(\boldsymbol{\lambda} k'.[\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/z, e/x] \in |A[\boldsymbol{c}/\boldsymbol{x}, e/x]|$$

so by taking $e = [a^\dagger][\boldsymbol{c}/\boldsymbol{x}]$ we have:

$$\begin{aligned}
&\boldsymbol{\lambda} k.[zx][(\boldsymbol{\lambda} k'.[\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k/z, e/x] \\
&\quad = \boldsymbol{\lambda} k.[zx][[\varepsilon'^*][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]/z, [a^\dagger][\boldsymbol{c}/\boldsymbol{x}]/x] \\
&\quad = \boldsymbol{\lambda} k.[\varepsilon'^* x][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, [a^\dagger][\boldsymbol{c}/\boldsymbol{x}]/x] \\
&\quad = \boldsymbol{\lambda} k.[\varepsilon'^* a^\dagger][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]
\end{aligned}$$

since on the other hand we have $|A[\boldsymbol{c}/\boldsymbol{x}, [a^\dagger][\boldsymbol{c}/\boldsymbol{x}]/x]| = |A[a^\tau/x][\boldsymbol{c}/\boldsymbol{x}]|$ we obtain:

$$\boldsymbol{\lambda} k.[\varepsilon'^* a^\dagger][\boldsymbol{c}/\boldsymbol{x}, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |A[a^\tau/x][\boldsymbol{c}/\boldsymbol{x}]|$$

– $\varepsilon = \dfrac{\dfrac{\varepsilon'}{\Gamma \vdash A \mid \Delta, A}}{(\Gamma \vdash \Delta, A)}$: we prove here that for any $c_1 \in {\tau_1}^{\mathtt{t}}, \ldots, c_k \in {\tau_k}^{\mathtt{t}}$, $a_1 \in |A_1[\boldsymbol{c}/\boldsymbol{x}]|, \ldots, a_n \in |A_n[\boldsymbol{c}/\boldsymbol{x}]|$, $b_1 \in ||B_1[\boldsymbol{c}/\boldsymbol{x}]||, \ldots, b_m \in ||B_m[\boldsymbol{c}/\boldsymbol{x}]||$ and $b \in ||A[\boldsymbol{c}/\boldsymbol{x}]||$, we have:

$$\boldsymbol{\lambda} k.[[\alpha]\varepsilon'^*][\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, bk/\alpha] \in \perp\!\!\!\perp$$

where $[[\alpha]t]$ is $[t]\alpha$. The induction hypothesis gives:

$$\boldsymbol{\lambda} k.[\varepsilon'^*][\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, bk/\alpha] \in |A[\boldsymbol{c}/\boldsymbol{x}]|$$

and we have:

$$\begin{aligned}
&\boldsymbol{\lambda} k.[[\alpha]\varepsilon'^*][\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, bk/\alpha] \\
&\quad = \boldsymbol{\lambda} k.([\varepsilon'^*]\alpha)[\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, bk/\alpha] \\
&\quad = \boldsymbol{\lambda} k.([\varepsilon'^*](bk))[\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, bk/\alpha] \\
&\quad = \boldsymbol{\lambda} k.[\varepsilon'^*][\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, bk/\alpha](bk) \\
&\quad = \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.[\varepsilon'^*][\boldsymbol{c}/x, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}, bk'/\alpha])k(bk)
\end{aligned}$$

which is in $\perp\!\!\!\perp$ using the induction hypothesis and the fact that $b \in ||A[\boldsymbol{c}/\boldsymbol{x}]||$.

– $\varepsilon = \dfrac{\dfrac{\varepsilon'}{(\Gamma \vdash \Delta, A)}}{\Gamma \vdash A \mid \Delta}$: the previous point gives us:

$$\forall b \in ||A[\boldsymbol{c}/\boldsymbol{x}]||, \boldsymbol{\lambda} k.[\varepsilon'^*][\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, bk/\alpha] \in \perp\!\!\!\perp$$

so if we define $[\mu\alpha.t] = \boldsymbol{\lambda}\alpha.[t]$ (so we still have $[\mu\alpha.[\beta]t] = \boldsymbol{\lambda}\alpha[t]\beta$), we prove:

$$\boldsymbol{\lambda} k.[\mu\alpha.\varepsilon'^*][\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}] \in |A[\boldsymbol{c}/\boldsymbol{x}]|$$

In order to do that we choose $b \in ||A[\boldsymbol{c}/\boldsymbol{x}]||$ and we prove:

$$\boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.[\mu\alpha.\varepsilon'^*][\boldsymbol{c}/x, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k(bk) \in \perp\!\!\!\perp$$

but since we have:

$$\begin{aligned}
&\boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.[\mu\alpha.\varepsilon'^*][\boldsymbol{c}/x, \boldsymbol{a}k'/\boldsymbol{y}, \boldsymbol{b}k'/\boldsymbol{\alpha}])k(bk)\\
&\quad = \boldsymbol{\lambda} k.[\mu\alpha.\varepsilon'^*][\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}](bk)\\
&\quad = \boldsymbol{\lambda} k.[\mu\alpha.\varepsilon'^*](bk)[\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}]\\
&\quad = \boldsymbol{\lambda} k.[\mu\alpha.\varepsilon'^*]\alpha[\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, bk/\alpha]\\
&\quad = \boldsymbol{\lambda} k.(\boldsymbol{\lambda}\alpha.[\varepsilon'^*])\alpha[\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, bk/\alpha]\\
&\quad = \boldsymbol{\lambda} k.[\varepsilon'^*][\boldsymbol{c}/x, \boldsymbol{a}k/\boldsymbol{y}, \boldsymbol{b}k/\boldsymbol{\alpha}, bk/\alpha]
\end{aligned}$$

which is in $\perp\!\!\!\perp$ by the previous point.

□

B Computational Adequacy

The correctness of the machine of Sect. 5 (*i.e.* reduction preserves semantics) can be proved as usual (see e.g. [18]). Note that since the model $\mathsf{Fam}(\mathcal{G})$ is typed, this would involve typing rules for environments and stacks.

For extraction, we acutally only need to show Proposition 7.3:

(i) If $\vdash t : \iota|$ in Λ_Ψ, then for all $n \in \mathbb{N}$ we have $(t, \varepsilon, \star) \succ (\overline{n}, e, \star)$ if $[t] = [n]$.
(ii) Let $\vdash t : \iota \to \iota$ in Λ_Ψ. For all $n, m \in \mathbb{N}$, if $\boldsymbol{\lambda} k.[t]\langle[\overline{n}], k\rangle = [\overline{m}]$ then $(t\overline{n}, \varepsilon, \star) \succ (\overline{m}, e, \star)$.

We prove this property here.

We use the usual technique of logical relations, and deal with Bar-Recursion using the usual technique for the PCF fixpoint operator (see e.g. [2]). As suggested by the interpretation of the calculus in $\mathtt{R}^{\mathsf{Fam}(\mathcal{G})}$, and similarly to what we have done for realizability (see Sect. 6 and App A), our logical relations will be build by orthogonality.

Idealy, we would process as follows: We would fix a binary relation $\top\!\!\!\top$ between $(\Lambda_\Psi \times \mathrm{E} \times \Pi)$ and (strategies on) $[\iota]$, and as usual assume that $\top\!\!\!\top$ is closed under *anti-evaluation*, *i.e.*

$$(t, e, \pi) \top\!\!\!\top a \quad \Longrightarrow \quad (t', e', \pi') \succ (t, e, \pi) \quad \Longrightarrow \quad (t', e', \pi') \top\!\!\!\top a$$

and that for all (t, e, π) we have

$$(t, e, \pi) \top\!\!\!\top \perp_{[\iota]}$$

We would then devise a relation $\mathcal{R}_\tau \subseteq \Pi \times [\![\tau]\!]$ for each simple type τ, and obtain by orthogonality a relation $\mathcal{R}_\tau^{\top\!\!\!\top} \subseteq \mathrm{C} \times [\tau]$. However, we face a similar problem as with realizability in Sect. 6: we want to observe termination at $[\iota] = \mathtt{R}^{[\![\iota]\!]}$, which is not a basic type. Moreover, the only strategy on $[\![\iota]\!]$ is the empty strategy; and applying it to (the interpretation of) a numeral $[\overline{n}]$ gives the empty strategy on R (which is the only possible one since R is the arena with just one (initial) oponent move).

Hence, we preceed similarly as for realizability, and consider for each simple type τ relations $\mathcal{R}_\tau \subseteq \Pi \times (\llbracket\iota\rrbracket \to \llbracket\tau\rrbracket)$ and $\mathcal{R}_\tau^{\top\!\!\!\top} \subseteq \mathrm{C} \times (\llbracket\iota\rrbracket \to \llbracket\tau\rrbracket)$. The main point, given by Lemma B.5, is that we can define $\mathcal{R}_\iota$ so that for all $n \in \mathbb{N}$, we have $((\overline{n}, e), \boldsymbol{\lambda} k.[\overline{n}]) \in \mathcal{R}_\iota^{\top\!\!\!\top}$.

B.1 The Logical Relation

Let $\top\!\!\!\top$ be a binary relation between $(\Lambda_\Psi \times \mathrm{E} \times \Pi)$ and (strategies on) $[\iota]$. We assume that $\top\!\!\!\top$ is closed under *anti-evaluation*, *i.e.*

$$(t, e, \pi) \top\!\!\!\top a \quad \Longrightarrow \quad (t', e', \pi') \succ (t, e, \pi) \quad \Longrightarrow \quad (t', e', \pi') \top\!\!\!\top a$$

and that for all (t, e, π) we have

$$(t, e, \pi) \top\!\!\!\top \bot_{[\iota]}$$

Recall from Section 4 that we use a simply-typed $\boldsymbol{\lambda}$-calculus with constants in $\mathsf{Fam}(\mathcal{G})$.

To each simple type τ, we will associate two binary relations

$$\mathcal{R}_\tau \subseteq \Pi \times (\llbracket\iota\rrbracket \to \llbracket\tau\rrbracket) \qquad \text{and} \qquad \mathcal{R}_\tau^{\top\!\!\!\top} \subseteq \mathrm{C} \times (\llbracket\iota\rrbracket \to [\tau])$$

First, given any $\mathcal{A} \subseteq \Pi \times (\llbracket\iota\rrbracket \to \llbracket\tau\rrbracket)$, we let $\mathcal{A}^{\top\!\!\!\top} \subseteq \mathrm{C} \times (\llbracket\iota\rrbracket \to [\tau])$ be

$$\{((t, e), a) \mid \forall (\pi, b) \in \mathcal{A},\ (t, e, \pi) \top\!\!\!\top \boldsymbol{\lambda} k.ak(bk)\}$$

If moreover $\mathcal{B} \subseteq \Pi \times (\llbracket\iota\rrbracket \to \llbracket\sigma\rrbracket)$, we define $(\mathcal{A}^{\top\!\!\!\top} \cdot \mathcal{B}) \subseteq \Pi \times (\llbracket\iota\rrbracket \to \llbracket\tau \to \sigma\rrbracket)$ as

$$\{(\langle c, \pi\rangle, \boldsymbol{\lambda} k.\langle ak, bk\rangle) \mid (c, a) \in \mathcal{A}^{\top\!\!\!\top} \text{ and } (\pi, b) \in \mathcal{B}\}$$

For the moment, we assume given some $\mathcal{R}_\iota \subseteq \Pi \times (\llbracket\iota\rrbracket \to \llbracket\iota\rrbracket)$, and define $\mathcal{R}_\tau$ by induction on τ as follows:

$$\begin{array}{ll} \mathcal{R}_{\sigma\times\tau} & := \{(\mathsf{kp}_1(\pi), \boldsymbol{\lambda} k.\mathbf{in}_1(ak)) \mid (\pi, a) \in \mathcal{R}_\sigma\} \\ & \ \cup \{(\mathsf{kp}_2(\pi), \boldsymbol{\lambda} k.\mathbf{in}_2(ak)) \mid (\pi, a) \in \mathcal{R}_\tau\} \\ \mathcal{R}_{\sigma\to\tau} & := (\mathcal{R}_\sigma^{\top\!\!\!\top} \cdot \mathcal{R}_\tau) \end{array}$$

This definition of $\mathcal{R}_\tau$ with $\mathcal{R}_\iota$ arbitrary is sufficient to deal with the $\lambda\mu$-calculus with products. The actual definition on $\mathcal{R}_\iota$ will be given in App. B.2, when discussing arithmetic constants.

Theorem B.1 (Adequacy for the $\lambda\mu$-Calculus with Products). *If*

$$x_1 : \tau_1, \ldots, x_n : \tau_n \vdash t : \tau \mid \alpha_1 : \sigma_1, \ldots, \alpha_m : \sigma_m$$

then for all

$$(u_1, e_1)\, \mathcal{R}_{\tau_1}^{\top\!\!\!\top}\, b_1, \ldots, (u_n, e_n)\, \mathcal{R}_{\tau_n}^{\top\!\!\!\top}\, b_n$$

and all

$$\pi_1 \, \mathcal{R}_{\sigma_1} \, a_1, \ldots, \pi_m \, \mathcal{R}_{\sigma_m} \, a_m$$

we have

$$(t, e) \, \mathcal{R}_\tau^{\top\!\!\top} \, \boldsymbol{\lambda} k.\boldsymbol{t}(k)$$

where

$$e \quad := \quad (x_1, (u_1, e_1)) :: \cdots :: (x_n, (u_n, e_n)) :: (\alpha_1, \pi_1) :: \cdots :: (\alpha_m, \pi_m) :: \varepsilon$$

and

$$\boldsymbol{t}(k) \quad := \quad [t][b_1(k)/x_1, \ldots, b_n(k)/x_n, a_1(k)/\alpha_1, \ldots, a_m(k)/\alpha_m]$$

In particular, if $\vdash t : \tau|$*, then we have* $((t, \varepsilon), \boldsymbol{\lambda} k.[t]) \in \mathcal{R}_\tau^{\top\!\!\top}$*.*

Proof. By induction on typing judgments. In the following, we let Γ be the context $x_1 : \tau_1, \ldots, x_n : \tau_n$ and Δ be $\alpha_1 : \sigma_1, \ldots, \alpha_m : \sigma_m$. Unless stated otherwise, we will always assume given $(u_i, e_i, b_i)_{1 \leq i \leq n}$ and $(\pi_j, a_j)_{1 \leq j \leq m}$ as in the statement of the theorem.

We reason by cases on the last applied typing rule.

$$\frac{}{\Gamma, x_0 : \tau \vdash x_0 : \tau \mid \Delta}$$

Let $((u_0, e_0), b_0) \in \mathcal{R}_\tau^{\top\!\!\top}$ and $(\pi, a) \in \mathcal{R}_\tau$. We have to show

$$((u_0, e_0), b_0) \top\!\!\top \boldsymbol{\lambda} k.\boldsymbol{t}(k)(ak)$$

We have

$$(t, e, \pi) \quad \succ \quad (u_0, e_0, \pi)$$

and

$$\boldsymbol{\lambda} k.\boldsymbol{t}(k) \quad = \quad \boldsymbol{\lambda} k.b_0 k \quad = \quad b_0$$

We are done since by assumption,

$$(u_0, e_0, \pi) \top\!\!\top \boldsymbol{\lambda} k.b_0 k(ak)$$

$$\frac{\Gamma, x : \tau \vdash t : \sigma \mid \Delta}{\Gamma \vdash \lambda x.t : \tau \to \sigma \mid \Delta}$$

Let $((u_0, e_0), b_0) \in \mathcal{R}_\tau^{\top\!\!\top}$ and $(\pi, a) \in \mathcal{R}_\sigma$. We have to show

$$(\lambda x.t, e, \langle (u_0, e_0), \pi \rangle) \top\!\!\top \boldsymbol{\lambda} k.(\boldsymbol{\lambda} \langle x, k' \rangle.\boldsymbol{t}(k)k')\langle b_0 k, ak \rangle$$

We have

$$(\lambda x.t, e, \langle (u_0, e_0), \pi \rangle) \quad \succ \quad (t, ((x, (u_0, e_0)) :: e, \pi)$$

and

$$\boldsymbol{\lambda} k.(\boldsymbol{\lambda} \langle x, k' \rangle.\boldsymbol{t}(k)k')\langle b_0 k, ak \rangle \quad = \quad \boldsymbol{\lambda} k.\boldsymbol{t}(k)[b_0 k/x](ak)$$

and we are donce since by induction hypothesis

$$(t, (x, (u_0, e_0)) :: e, \pi) \perp\!\!\!\perp \boldsymbol{\lambda} k.\boldsymbol{t}(k)[b_0 k/x](ak)$$

$$\frac{\Gamma \vdash t : \sigma \to \tau \mid \Delta \qquad \Gamma \vdash u : \sigma \mid \Delta}{\Gamma \vdash tu : \tau \mid \Delta}$$

Let $(\pi, a) \in \mathcal{R}_\tau$. We have to show

$$(tu, e, \pi) \perp\!\!\!\perp \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.\boldsymbol{t}(k)\langle \boldsymbol{u}(k), k'\rangle)(ak)$$

We have

$$(tu, e, \pi) \quad \succ \quad (t, e, \langle (u, e), \pi\rangle)$$

and

$$\boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.\boldsymbol{t}(k)\langle \boldsymbol{u}(k), k'\rangle)(ak) \quad = \quad \boldsymbol{\lambda} k.\boldsymbol{t}(k)\langle \boldsymbol{u}(k), ak\rangle$$

We are done, since on the one hand by induction hypothesis

$$((t, e), \boldsymbol{\lambda} k.\boldsymbol{t}(k)) \in \mathcal{R}^{\perp\!\!\!\perp}_{\sigma \to \tau}$$

and on the other hand

$$(\langle (u, e), \pi\rangle, \boldsymbol{\lambda} k.\langle \boldsymbol{u}(k), ak\rangle) \in \mathcal{R}_{\sigma \to \tau}$$

since by induction hypothesis

$$((u, e), \boldsymbol{\lambda} k.\boldsymbol{u}(k)) \in \mathcal{R}^{\perp\!\!\!\perp}_{\sigma}$$

$$\frac{\Gamma \vdash t_1 : \tau_1 \mid \Delta \qquad \Gamma \vdash t_2 : \tau_2 \mid \Delta}{\Gamma \vdash \langle t_1, t_2\rangle : \tau_1 \times \tau_2 \mid \Delta}$$

Let $(\pi, a) \in \mathcal{R}_{\tau_i}$. Hence $(\mathsf{kp}_i(\pi), \boldsymbol{\lambda} k.\textbf{in}_i(ak)) \in \mathcal{R}_{\tau_1 \times \tau_2}$. We have to show

$$(\langle t_1, t_2\rangle, e, \mathsf{kp}_i(\pi)) \perp\!\!\!\perp \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.\textbf{case}\, k'\{\boldsymbol{t}_1(k), \boldsymbol{t}_2(k)\})(\textbf{in}_i(ak))$$

We have

$$(\langle t_1, t_2\rangle, e, \mathsf{kp}_i(\pi)) \quad \succ \quad (t_i, e, \pi)$$

and

$$\boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.\textbf{case}\, k'\{\boldsymbol{t}_1(k), \boldsymbol{t}_2(k)\})(\textbf{in}_i(ak)) \quad = \quad \boldsymbol{\lambda} k.\boldsymbol{t}_i(k)(ak)$$

and we are donce since by induction hypothesis

$$((t_i, e), \boldsymbol{\lambda} k.\boldsymbol{t}_i(k)) \in \mathcal{R}^{\perp\!\!\!\perp}_{\tau_i}$$

$$\frac{\Gamma \vdash t : \tau_1 \times \tau_2 \mid \Delta}{\Gamma \vdash \mathsf{p}_i(t) : \tau_i \mid \Delta}\ (i = 1, 2)$$

Let $(\pi, a) \in \mathcal{R}_{\tau_i}$. We have have to show

$$(\mathsf{p}_i(t), e, \pi) \perp\!\!\!\perp \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.\boldsymbol{t}(k)(\textbf{in}_i k'))(ak)$$

We have

$$(\mathsf{p}_i(t), e, \pi) \quad \succ \quad (t, e, \mathsf{kp}_i(\pi))$$

and we are done since

$$(\mathsf{kp}_i(\pi), \boldsymbol{\lambda} k.\mathbf{in}_i(ak)) \in \mathcal{R}_{\tau_1 \times \tau_2}$$

and since by induction hypothesis

$$((t, e), \boldsymbol{\lambda} k.\boldsymbol{t}(k)) \in \mathcal{R}^{\top\!\!\top}_{\tau_1 \times \tau_2}$$

$$\frac{\Gamma \vdash t : \tau \mid \Delta, \alpha : \tau}{[\alpha]t : (\Gamma \vdash \Delta, \alpha : \tau)} \qquad\qquad \frac{t : (\Gamma \vdash \Delta, \alpha : \tau)}{\Gamma \vdash \mu\alpha.t : \tau \mid \Delta}$$

It is actually sufficent to consider the case of

$$\dfrac{\dfrac{\Gamma \vdash t : \sigma \mid \Delta, \beta : \sigma}{\vdots}}{\Gamma \vdash \mu\alpha.[\beta]t : \tau \mid (\Delta, \beta : \sigma) \setminus \{\alpha : \tau\}}$$

Let $(\pi, a) \in \mathcal{R}_\tau$. Note that we can have either $\alpha = \beta$ (in which case $\tau = \sigma$) or $\alpha = \beta$. In both cases, by assumption we can assume given $(\pi', a') \in \mathcal{R}_\sigma$. We have to show

$$(\mu\alpha.[\beta]t, e, \pi) \top\!\!\top \boldsymbol{\lambda} k.(\boldsymbol{\lambda}\alpha.\boldsymbol{t}(k)(a'k))(ak)$$

We have

$$(\mu\alpha.[\beta]t, e, \pi) \quad \succ \quad (t, (\alpha, \pi) :: e, \pi')$$

and we are done since by induction hypothesis

$$((t, (\alpha, \pi) :: e), \boldsymbol{\lambda} k.\boldsymbol{t}(k)[ak/\alpha]) \in \mathcal{R}^{\top\!\!\top}_\sigma$$

□

B.2 Adequacy for Arithmetical Constants

From now on, we let $(t, e, \pi) \top\!\!\top a$ iff either $a = \bot_{[\iota]}$ or $a = [\overline{n}]$ and $(t, e, \pi) \succ (\overline{n}, e', \star)$ for some $n \in \mathbb{N}$ and $e' \in \mathrm{E}$.

We now proceed to the definition of $\mathcal{R}_\iota$. It will be defined as a least fixpoint in the complete lattice $\mathcal{P}(\Pi \times ([\![\iota]\!] \to [\![\iota]\!]))$. Given $X \subseteq \Pi \times ([\![\iota]\!] \to [\![\iota]\!])$, let $F(X) \subseteq \Pi \times ([\![\iota]\!] \to [\![\iota]\!])$ be

$$\begin{aligned}
&\{(\star, \boldsymbol{\lambda} k.k)\} \\
\cup\ &\{(\mathsf{ksucc}(\pi), \boldsymbol{\lambda} k.\widetilde{\mathsf{ksucc}}\,(ak)) \mid (\pi, a) \in X\} \\
\cup\ &\textstyle\bigcup_{\mathcal{A} \subseteq \Pi \times ([\![\iota]\!] \to [\![\tau]\!])} \{(\mathsf{krec}(u, v, e, \pi), \boldsymbol{\lambda} k.\widetilde{\mathsf{krec}}\,(bk)\,(ck)\,(ak)) \mid \\
&\quad ((u, e), b) \in \mathcal{A}^{\top\!\!\top},\ ((v, e), c) \in (X^{\top\!\!\top} \cdot \mathcal{A}^{\top\!\!\top} \cdot \mathcal{A})^{\top\!\!\top}\ \&\ (\pi, a) \in \mathcal{A}\}
\end{aligned}$$

where

$$\begin{array}{rcl} \widetilde{\mathsf{ksucc}} & := & \boldsymbol{\lambda} a.\boldsymbol{\lambda} n.a(\widetilde{\mathsf{succ}}\, n) \\ \widetilde{\mathsf{krec}} & := & \boldsymbol{\lambda} b.\boldsymbol{\lambda} c.\boldsymbol{\lambda} a.\boldsymbol{\lambda} n.\widetilde{\mathsf{rec}}\, b\, (\boldsymbol{\lambda} m.c^{\bullet}(\boldsymbol{\lambda} k.km))\, n\, a \\ & & \quad \text{where } c^{\bullet} := \boldsymbol{\lambda} x.\boldsymbol{\lambda} y.\boldsymbol{\lambda} z.c\langle x,y,z\rangle \end{array}$$

Note that

$$\begin{array}{rcl} \textbf{succ} & = & \boldsymbol{\lambda} n \boldsymbol{\lambda} k.n\,(\widetilde{\mathsf{ksucc}}\, k) \\ \textbf{rec} & = & \boldsymbol{\lambda} u.\boldsymbol{\lambda} v.\boldsymbol{\lambda} n.\boldsymbol{\lambda} k.n\,(\widetilde{\mathsf{krec}}\, u\, v\, k) \end{array}$$

Lemma B.2. *If $X \subseteq Y$ then $F(X) \subseteq F(Y)$.*

Proof. Let $(\pi, a) \in F(X)$. We show $(c, a) \in F(Y)$ by cases on the form of π. If π is of the form $\star$ or $\mathsf{ksucc}(\pi')$ then the result is trivial. The remaining case follows from the following (usual) observation: since $X \subseteq Y$ implies $Y^{\top\!\!\!\top} \subseteq X^{\top\!\!\!\top}$ we have $(Y^{\top\!\!\!\top} \cdot \mathcal{A}^{\top\!\!\!\top} \cdot \mathcal{A}) \subseteq (X^{\top\!\!\!\top} \cdot \mathcal{A}^{\top\!\!\!\top} \cdot \mathcal{A})$ hence $(X^{\top\!\!\!\top} \cdot \mathcal{A}^{\top\!\!\!\top} \cdot \mathcal{A})^{\top\!\!\!\top} \subseteq (Y^{\top\!\!\!\top} \cdot \mathcal{A}^{\top\!\!\!\top} \cdot \mathcal{A})^{\top\!\!\!\top}$. □

Using Tarski's fixpoint theorem, we let $\mathcal{R}_\iota$ be least fixed-point of F. Hence,

$$\mathcal{R}_\iota \quad = \quad F^{\lambda}(\emptyset)$$

for some ordinal λ. For ordinals $\alpha \leq \lambda$, let

$$\mathcal{R}_\iota^{\alpha} \quad := \quad F^{\alpha}(\emptyset)$$

Note that

$$\mathcal{R}_\iota^{\beta+1} \quad = \quad F(\mathcal{R}_\iota^{\beta})$$

In other words, $\mathcal{R}_\iota$ is the smallest subset of $\mathit{\Pi} \times (\llbracket \iota \rrbracket \to \llbracket \iota \rrbracket)$ such that

- $(\star, \boldsymbol{\lambda} k.k) \in \mathcal{R}_\iota$,
- if $(\pi, a) \in \mathcal{R}_\iota$, then $(\mathsf{ksucc}(\pi), \boldsymbol{\lambda} k.\widetilde{\mathsf{ksucc}}(ak)) \in \mathcal{R}_\iota$, and
- if $\mathcal{A} \subseteq \mathit{\Pi} \times (\llbracket \iota \rrbracket \to \llbracket \tau \rrbracket)$, then for all u, v, e, π and all a, b, c such that $((u, e), b) \in \mathcal{A}^{\top\!\!\!\top}$, $((v, e), c) \in (\mathcal{R}_\iota^{\top\!\!\!\top} \cdot \mathcal{A}^{\top\!\!\!\top} \cdot \mathcal{A})^{\top\!\!\!\top}$, and $(\pi, a) \in \mathcal{A}$, we have

$$(\mathsf{krec}(u, v, e, \pi), \boldsymbol{\lambda} k.\widetilde{\mathsf{krec}}(bk)(ck)(ak))) \in \mathcal{R}_\iota$$

Lemma B.3. $((\mathsf{succ}, e), \boldsymbol{\lambda} k.[\mathsf{succ}]) \in \mathcal{R}_{\iota\to\iota}^{\top\!\!\!\top}$

Proof. We have to show

$$(\mathsf{succ}, e, \pi) \perp\!\!\!\perp \boldsymbol{\lambda} k.\boldsymbol{\lambda}\langle n, y\rangle.(\textbf{succ}\ n\, y)(ak)$$

for all $(\pi, a) \in \mathcal{R}_{\iota\to\iota}$. Since $\mathcal{R}_{\iota\to\iota} = \mathcal{R}_\iota^{\top\!\!\!\top} \cdot \mathcal{R}_\iota$, this amounts to show

$$(\mathsf{succ}, e, \langle (t, e'), \pi\rangle) \perp\!\!\!\perp \boldsymbol{\lambda} k.\,\textbf{succ}\,(ak)\,(bk)$$

for all $((t, e'), a) \in \mathcal{R}_\iota^{\top\!\!\!\top}$ and all $(\pi, b) \in \mathcal{R}_\iota$. But

$$(\mathsf{succ}, e, \langle (t, e'), \pi\rangle) \quad \succ \quad (t, e', \mathsf{ksucc}(\pi))$$

On the other hand,

$$\begin{array}{rcl} \boldsymbol{\lambda} k.\,\textbf{succ}\,(ak)\,(bk) & = & \boldsymbol{\lambda} k.ak(\boldsymbol{\lambda} n.bk(\widetilde{\mathsf{succ}}\, n)) \\ & = & \boldsymbol{\lambda} k.(ak)(\widetilde{\mathsf{ksucc}}\,(bk)) \end{array}$$

and we are done since $((t, e'), a) \in \mathcal{R}_\iota^{\top\!\!\!\top}$ and $(\mathsf{ksucc}(\pi), \boldsymbol{\lambda} k.\widetilde{\mathsf{ksucc}}\,(bk)) \in \mathcal{R}_\iota$. □

Lemma B.4. *Let $\tau \in \mathcal{T}$, $\mathcal{A} \subseteq \Pi \times (\llbracket\iota\rrbracket \to \llbracket\tau\rrbracket)$, and $\alpha \leq \lambda$. For all $u, v \in \Lambda, e \in$ E, $b \in \llbracket\iota\rrbracket \to [\tau]$, and $c \in \llbracket\iota\rrbracket \to [\iota \to \tau \to \tau]$ such that $((u,e),b) \in \mathcal{A}^{\top\!\!\top}$ and $((v,e),c) \in (\mathcal{R}_\iota^{\alpha\,\top\!\!\top} \cdot \mathcal{A}^{\top\!\!\top} \cdot \mathcal{A})^{\top\!\!\top}$, we have*

$$((\mathsf{rec}(u,v),e), \boldsymbol{\lambda}k.\boldsymbol{\lambda}\langle n,y\rangle.\mathbf{rec}\,(bk)\,(ck)\,n\,y) \in (\mathcal{R}_\iota^{\alpha\,\top\!\!\top} \cdot \mathcal{A})^{\top\!\!\top}$$

In partcular, if $\mathcal{A} = \mathcal{R}_\tau$ and $\alpha = \lambda$, then we get

$$((\mathsf{rec}(u,v),e), \boldsymbol{\lambda}k.\boldsymbol{\lambda}\langle n,y\rangle.\mathbf{rec}\,(bk)\,(ck)\,n\,y) \in \mathcal{R}_{\iota\to\tau}^{\top\!\!\top}$$

Proof. We have to show

$$(\mathsf{rec}(u,v), e, \langle(t,e'),\pi\rangle) \;\top\!\!\top\; \boldsymbol{\lambda}k.\mathbf{rec}\,(bk)\,(ck)\,(ak)\,(dk)$$

for all $((t,e'),a) \in \mathcal{R}_\iota^{\alpha\,\top\!\!\top}$ and all $(\pi,d) \in \mathcal{A}$. We have

$$(\mathsf{rec}(u,v), e, \langle(t,e'),\pi\rangle) \quad \succ \quad (t, e', \mathsf{krec}(u,v,e,\pi))$$

On the other hand,

$$\begin{array}{rcl} \boldsymbol{\lambda}k.\mathbf{rec}\,(bk)\,(ck)\,(ak)\,(dk) & = & \boldsymbol{\lambda}k.ak(\boldsymbol{\lambda}n.\widetilde{\mathsf{rec}}(bk)(\boldsymbol{\lambda}y.(ck)^{\bullet}(\boldsymbol{\lambda}k'.k'y))n(dk)) \\ & & \quad \text{where } (ck)^{\bullet} := \boldsymbol{\lambda}x.\boldsymbol{\lambda}y.\boldsymbol{\lambda}z.(ck)\langle x,y,z\rangle \\ & = & \boldsymbol{\lambda}k.ak(\widetilde{\mathsf{krec}}(bk)(ck)(dk)) \end{array}$$

We are done as $((t,e'),a) \in \mathcal{R}_\iota^{\top\!\!\top}$ and $(\mathsf{krec}(u,v,e,\pi), \boldsymbol{\lambda}k.\widetilde{\mathsf{krec}}(bk)(ck)(dk)) \in \mathcal{R}_\iota$. ☐

Lemma B.5. *For all $n \in \mathbb{N}$ and all $e \in$ E, we have $((\overline{n},e), \boldsymbol{\lambda}k.[\overline{n}]) \in \mathcal{R}_\iota^{\top\!\!\top}$.*

Proof. We show by induction on ordinals $\alpha \leq \lambda$ that for all $(\pi,a) \in \mathcal{R}_\iota^\alpha$, we have $(\overline{n},e,\pi) \;\top\!\!\top\; \boldsymbol{\lambda}k.[\overline{n}](ak)$ for all $n \in \mathbb{N}$ and all $e \in$ E.

First, if α is a limit ordinal, then

$$\mathcal{R}_\iota^\alpha \quad = \quad \bigcup_{\beta<\alpha} \mathcal{R}_\iota^\beta.$$

Hence $(\pi,a) \in \mathcal{R}_\iota^\alpha$ iff $(\pi,a) \in \mathcal{R}_\iota^\beta$ for some $\beta < \alpha$ and the result follows directly by induction hypothesis.

Otherwise, α is either $\emptyset$ or a limit ordinal $\beta + 1$. We reason by cases on $(\pi,a) \in \mathcal{R}_\iota^\alpha$.

- **Case of $(\star, \boldsymbol{\lambda}k.k)$.** Given $n \in \mathbb{N}$ and $e \in$ E, we have to show

$$(\overline{n}, e, \star) \;\top\!\!\top\; \boldsymbol{\lambda}k.[\overline{n}]((\boldsymbol{\lambda}k'.k')k)$$

  But we are done since

$$\boldsymbol{\lambda}k.[\overline{n}]((\boldsymbol{\lambda}k'.k')k) \quad = \quad \boldsymbol{\lambda}k.[\overline{n}]k \quad = \quad [\overline{n}]$$

– **Case of** $(\mathsf{ksucc}(\pi), \boldsymbol{\lambda} k.\widetilde{\mathsf{ksucc}}(ak))$**.** In this case we have $\alpha = \beta{+}1$ and $(\pi, a) \in \mathcal{R}_\iota^\beta$. Given $n \in \mathbb{N}$ and $e \in \mathrm{E}$, we have to show

$$(\overline{n}, e, \mathsf{ksucc}(\pi)) \mathrel{\top\!\!\!\top} \boldsymbol{\lambda} k.[\overline{n}]((\boldsymbol{\lambda} k'.\widetilde{\mathsf{ksucc}}(ak'))k)$$

First, we have

$$(\overline{n}, e, \mathsf{ksucc}(\pi)) \quad \succ \quad (\overline{n+1}, e, \pi)$$

On the other hand

$$\begin{array}{rcl} \boldsymbol{\lambda} k.[\overline{n}]((\boldsymbol{\lambda} k'.\widetilde{\mathsf{ksucc}}(ak'))k) & = & \boldsymbol{\lambda} k.[\overline{n}](\widetilde{\mathsf{ksucc}}(ak)) \\ & = & \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.k'\widetilde{n})(\widetilde{\mathsf{ksucc}}(ak)) \\ & = & \boldsymbol{\lambda} k.\widetilde{\mathsf{ksucc}}(ak)\widetilde{n} \\ & = & \boldsymbol{\lambda} k.(\boldsymbol{\lambda} a.\boldsymbol{\lambda} n.a(\widetilde{\mathsf{succ}}n))(ak)\widetilde{n} \\ & = & \boldsymbol{\lambda} k.ak(\widetilde{\mathsf{succ}}\widetilde{n}) \\ & = & \boldsymbol{\lambda} k.ak\widetilde{n+1} \\ & = & \boldsymbol{\lambda} k.(\boldsymbol{\lambda} k'.k'\widetilde{n+1})(ak) \\ & = & \boldsymbol{\lambda} k.[\overline{n+1}](ak) \end{array}$$

Now we are done since by induction hypothesis, we have

$$(\overline{n+1}, e, \pi) \mathrel{\top\!\!\!\top} \boldsymbol{\lambda} k.[\overline{n+1}](ak)$$

– **Case of** $(\mathsf{krec}(u, v, e, \pi), \boldsymbol{\lambda} k.\widetilde{\mathsf{krec}}(bk)(ck)(ak))$**.** In this case we have $\alpha = \beta{+}1$ and $((u, e), b) \in \mathcal{A}^{\top\!\!\!\top}$, $((v, e), c) \in (\mathcal{R}_\iota^{\beta\,\top\!\!\!\top} \cdot \mathcal{A}^{\top\!\!\!\top} \cdot \mathcal{A})^{\top\!\!\!\top}$, and $(\pi, a) \in \mathcal{A}$ for some $\mathcal{A} \subseteq \Pi \times (\llbracket\iota\rrbracket \to \llbracket\tau\rrbracket)$.
We have to show that for all $n \in \mathbb{N}$ and all $e' \in \mathrm{E}$, we have

$$(\overline{n}, e', \mathsf{krec}(u, v, e, \pi)) \mathrel{\top\!\!\!\top} \boldsymbol{\lambda} k.[\overline{n}](\widetilde{\mathsf{krec}}(bk)(ck)(ak))$$

We reason by cases on $n \in \mathbb{N}$.

- If $n = 0$, then we have

$$(\overline{0}, e', \mathsf{krec}(u, v, e, \pi)) \quad \succ \quad (u, e, \pi)$$

  On the other hand,

$$\begin{array}{rcl} \boldsymbol{\lambda} k.[\overline{0}](\widetilde{\mathsf{krec}}(bk)(ck)(ak)) & = & \boldsymbol{\lambda} k.\widetilde{\mathsf{rec}}(bk)(\boldsymbol{\lambda} m.(ck)^\bullet(\boldsymbol{\lambda} k'.k'm))\widetilde{0}(ak) \\ & & \text{where } (ck)^\bullet := \boldsymbol{\lambda} x.\boldsymbol{\lambda} y.\boldsymbol{\lambda} z.ck\langle x, y, z\rangle \\ & = & \boldsymbol{\lambda} k.bk(ak) \end{array}$$

  We are done since $((u, e), b) \in \mathcal{A}^{\top\!\!\!\top}$ and $(\pi, a) \in \mathcal{A}$ by assumption.
- Otherwise, $n = m + 1$. We have

$$(\overline{m+1}, e', \mathsf{krec}(u, v, e, \pi)) \quad \succ \quad (v, e, \langle(\overline{m}, e'), \langle(\mathsf{rec}(u, v)\overline{m}, e), \pi\rangle\rangle)$$

On the other hand,

$$\begin{array}{rcl}\boldsymbol{\lambda} k.[\overline{m+1}](\widetilde{\mathsf{krec}}(bk)(ck)(ak)) & = & \boldsymbol{\lambda} k.[\overline{m+1}](\boldsymbol{\lambda} x.\widetilde{\mathsf{rec}}(bk)(\boldsymbol{\lambda} m.(ck)^\bullet(\boldsymbol{\lambda} k'.k'm))x(ak)) \\ & = & \boldsymbol{\lambda} k.\widetilde{\mathsf{rec}}(bk)(\boldsymbol{\lambda} m.(ck)^\bullet(\boldsymbol{\lambda} k'.k'm))\widetilde{m+1}(ak) \\ & = & \boldsymbol{\lambda} k.(ck)^\bullet[\overline{m}](\widetilde{\mathsf{rec}}(bk)(\boldsymbol{\lambda} m.(ck)^\bullet(\boldsymbol{\lambda} k'.k'm))\widetilde{m})(ak) \\ & = & \boldsymbol{\lambda} k.(ck)^\bullet[\overline{m}](\boldsymbol{\lambda} y.\widetilde{\mathsf{rec}}(bk)(\boldsymbol{\lambda} m.(ck)^\bullet(\boldsymbol{\lambda} k'.k'm))\widetilde{m}y)(ak) \\ & = & \boldsymbol{\lambda} k.(ck)^\bullet[\overline{m}](\boldsymbol{\lambda} y.[\overline{m}](\boldsymbol{\lambda} x.\widetilde{\mathsf{rec}}(bk)(\boldsymbol{\lambda} m.(ck)^\bullet(\boldsymbol{\lambda} k'.k'm))xy))(ak) \\ & = & \boldsymbol{\lambda} k.(ck)\langle[\overline{m}],\langle\boldsymbol{\lambda} y.\mathbf{rec}\,(bk)\,(ck)\,[\overline{m}]\,y, ak\rangle\rangle\end{array}$$

where $(ck)^\bullet := \boldsymbol{\lambda} x.\boldsymbol{\lambda} y.\boldsymbol{\lambda} z.ck\langle x,\langle y,z\rangle\rangle$

Since $((v,e),c) \in (\mathcal{R}_\iota^{\beta\,\top\!\!\top}\cdot\mathcal{A}^{\top\!\!\top}\cdot\mathcal{A})^{\top\!\!\top}$, we are done if

$$(\langle(\overline{m},e'),\langle(\mathsf{rec}(u,v)\overline{m},e),\pi\rangle\rangle,\boldsymbol{\lambda} k.\langle[\overline{m}],\langle\boldsymbol{\lambda} y.\mathbf{rec}\,(bk)\,(ck)\,[\overline{m}]\,y,ak\rangle\rangle) \in \mathcal{R}_\iota^{\beta\,\top\!\!\top}\cdot\mathcal{A}^{\top\!\!\top}\cdot\mathcal{A}$$

Now, by induction hypothesis we have $((\overline{m},e),\boldsymbol{\lambda} k.[\overline{m}]) \in \mathcal{R}_\iota^{\beta\,\top\!\!\top}$. We thus have to show

$$(\langle(\mathsf{rec}(u,v)\overline{m},e),\pi\rangle,\boldsymbol{\lambda} k.\langle\boldsymbol{\lambda} y.\mathbf{rec}\,(bk)\,(ck)\,[\overline{m}]\,y,ak\rangle) \in \mathcal{A}^{\top\!\!\top}\cdot\mathcal{A}$$

Since $(\pi,a)\in\mathcal{A}$, we are done if

$$((\mathsf{rec}(u,v)\overline{m},e),\boldsymbol{\lambda} k.\boldsymbol{\lambda} y.\mathbf{rec}\,(bk)\,(ck)\,[\overline{m}]\,y) \in \mathcal{A}^{\top\!\!\top}$$

Let $(\pi',a')\in\mathcal{A}$. Note that

$$(\mathsf{rec}(u,v)\overline{m},e,\pi') \quad \succ \quad (\mathsf{rec}(u,v),e,\langle(\overline{m},e),\pi'\rangle)$$

On the other hand,

$$\boldsymbol{\lambda} k.(\boldsymbol{\lambda} y.\mathbf{rec}\,(bk)\,(ck)\,[\overline{m}]\,y)(a'k) \quad = \quad \boldsymbol{\lambda} k.(\boldsymbol{\lambda}\langle x,y\rangle.\mathbf{rec}\,(bk)\,(ck)\,x\,y)\langle[\overline{m}],a'k\rangle$$

Now, we are done since by induction hypothesis we have $((\overline{m},e),\boldsymbol{\lambda} k.[\overline{m}]) \in \mathcal{R}_\iota^{\beta\,\top\!\!\top}$, hence

$$(\langle(\overline{m},e),\pi'\rangle,\boldsymbol{\lambda} k.\langle[\overline{m}],a'k\rangle) \in \mathcal{R}_\iota^{\beta\,\top\!\!\top}\cdot\mathcal{A}$$

and by Lemma B.4:

$$((\mathsf{rec}(u,v),e),\boldsymbol{\lambda} k.\boldsymbol{\lambda}\langle x,y\rangle.\mathbf{rec}\,(bk)\,(ck)\,x\,y) \in (\mathcal{R}_\iota^{\beta\,\top\!\!\top}\cdot\mathcal{A})^{\top\!\!\top}$$

□

It is now easy to extend computational adequacy (Thm. B.1) to the type system of the $\lambda\mu$-calculus with products extended with the typing rules for arithmetical constants of Section 5. We only detail the case of the recursor:

$$\frac{\Gamma\vdash t:\tau\mid\Delta \qquad \Gamma\vdash u:\iota\to\tau\to\tau\mid\Delta}{\Gamma\vdash\mathsf{rec}(t,u):\iota\to\tau\mid\Delta}$$

Assuming the conventions used in the proof of Thm. B.1, we have to show

$$(\mathsf{rec}(t,u),e),\boldsymbol{\lambda} k.\boldsymbol{\lambda}\langle n,k'\rangle.\mathbf{rec}\ \boldsymbol{t}(k)\ \boldsymbol{u}(k)\ n\ k') \in \mathcal{R}_{\iota\to\tau}^{\top\!\!\top}$$

This directly follows from Lemma B.4, since by hypothesis we have $((t,e),\boldsymbol{\lambda} k.\boldsymbol{t}(k)) \in \mathcal{R}_\tau^{\top\!\!\top}$ and $((u,e),\boldsymbol{\lambda} k.\boldsymbol{u}(k)) \in \mathcal{R}_{\iota\to\tau\to\tau}^{\top\!\!\top}$.

B.3 Adequacy for Bar-Recursion

We now discuss computational adequacy for the bar-recursor Ψ. We use the well-known technique of fixpoint induction, as in the usual proofs of computational adequacy for PCF (see e.g. [2]).
We rely on the following remarks (for all type σ):

1. For all (t, e) we have $((t, e), \bot_{[\![\iota]\!] \to [\sigma]}) \in \mathcal{R}_\sigma^{\top\!\!\top}$.

   Proof. Given $(\pi, a) \in \mathcal{R}_\sigma$, we have
   $$((t, e), \pi) \top\!\!\top \boldsymbol{\lambda} k.\bot_{[\![\iota]\!] \to [\sigma]} k(ak)$$
   since $\boldsymbol{\lambda} k.\bot_{[\![\iota]\!] \to [\sigma]} k(ak) = \bot_{[\iota]}$. □

2. Let (t, e) and let $(b_m)_{m \in \mathbb{N}} \in [\![\iota]\!] \to [\sigma]$ be a directed family such that $((t, e), b_m) \in \mathcal{R}_\sigma^{\top\!\!\top}$ for all $m \in \mathbb{N}$. Then $((t, e), \bigvee_{m \in \mathbb{N}} b_m) \in \mathcal{R}_\sigma^{\top\!\!\top}$.

   Proof. Given $(\pi, a) \in \mathcal{R}_\sigma$, we have to show
   $$(t, e, \pi) \top\!\!\top \boldsymbol{\lambda} k.(\bigvee_{m \in \mathbb{N}} b_m) k(ak)$$
   We have
   $$\boldsymbol{\lambda} k. \bigvee_{m \in \mathbb{N}} (b_m k(ak))$$
   If $\boldsymbol{\lambda} k.b_m k(ak) = \bot_{[\iota]}$ for all m, then $\boldsymbol{\lambda} k. \bigvee_{m \in \mathbb{N}} (b_m k(ak)) = \bot_{[\iota]}$ and we are done.
   Otherwise, there is some $m \in \mathbb{N}$ such that $\boldsymbol{\lambda} k.b_m k(ak) = [\overline{n}]$ for some $n \in \mathbb{N}$, and we have $\boldsymbol{\lambda} k. \bigvee_{m \in \mathbb{N}} (b_m k(ak)) = [\overline{n}]$. But we are done since by assumption, $\boldsymbol{\lambda} k.b_m k(ak) = [\overline{n}]$ implies $(t, e, \pi) \succ (\overline{n}, e', \star)$. □

We now fix $\tau \in \mathcal{T}$, $e \in \mathrm{E}$, $((t, e), a) \in \mathcal{R}_{\iota \to (\tau \to \iota) \to \tau}^{\top\!\!\top}$ and $((u, e), b) \in \mathcal{R}_{(\iota \to \tau) \to \tau}^{\top\!\!\top}$.

Given $c_0, \dots, c_n \in [\tau]$, we let
$$\widetilde{\Psi}^m_{\langle c_0, \dots, c_n \rangle} \quad := \quad \widetilde{\Psi}^m_{n+1} \langle c_0, \dots, c_n \rangle$$

We show that for all $m \in \mathbb{N}$, we have
$$((\Psi_\tau(t, u)\langle s_0, \dots, s_n \rangle, e), \boldsymbol{\lambda} k.\widetilde{\Psi}^m_{\langle a_0 k, \dots, a_n k \rangle}(ak)(bk)) \in \mathcal{R}_\iota^{\top\!\!\top}$$
for all $n \in \mathbb{N}$ and all $((s_0, e), b_0), \dots, ((s_n, e), b_n) \in \mathcal{R}_\tau^{\top\!\!\top}$.

We reason by induction on $m \in \mathbb{N}$. The base case $m = 0$ follows from Rem (1) above.

For the induction step, first note that thanks to the results of App. B.1 and B.2, we have the adequacy for the construction $\langle \dots \rangle @\_$:
$$(((\langle s_0, \dots, s_n \rangle @c, e), \boldsymbol{\lambda} k.\langle a_0 k, \dots, a_n k \rangle \mathbf{@}(ck)) \in \mathcal{R}_{\iota \to \tau}^{\top\!\!\top} \qquad (((v, e), c) \in \mathcal{R}_{\iota \to \tau}^{\top\!\!\top})$$

Let now $(\pi, d) \in \mathcal{R}_\iota$. We have to show:

$$(\Psi_\tau(t,u)\langle s_0,\ldots,s_n\rangle, e, \pi) \perp\!\!\!\perp \boldsymbol{\lambda} k.\widetilde{\Psi}^{m+1}_{\langle a_0k,\ldots,a_nk\rangle}(ak)(bk)(dk)$$

We have

$$\begin{array}{l}(\Psi_\tau(t,u)\langle s_0,\ldots,s_n\rangle, e, \pi) \quad \succ \\ (u, e, \langle((\langle s_0,\ldots,s_n\rangle @\lambda\_.\mathrm{ex}_\tau(t\,\overline{n+1}\,\lambda x.\Psi_\tau(t,u)\langle s_0,\ldots,s_n,x\rangle), e), \pi\rangle)\end{array}$$

and $\boldsymbol{\lambda} k.\widetilde{\Psi}^{m+1}_{\langle a_0k,\ldots,a_nk\rangle}(ak)(bk)(dk) \quad =$

$$\begin{array}{l}\boldsymbol{\lambda} k.(bk)\langle\langle a_0k,\ldots,a_nk\rangle \mathbf{@} \\ \boldsymbol{\lambda}\langle\_,k_0\rangle.[\mathrm{ex}_\tau]\langle\boldsymbol{\lambda} k_1.(ak)\langle[\overline{n+1}], \boldsymbol{\lambda}\langle x,k_2\rangle.\widetilde{\Psi}^{m}_{\langle a_0k,\ldots,a_nk,x\rangle}(bk)(ck)k_2, k_1\rangle, k_0\rangle, (dk)\rangle\end{array}$$

Now we are done since by induction hypothesis:

$$(\lambda x.\Psi_\tau(t,u)\langle s_0,\ldots,s_n,x\rangle, \boldsymbol{\lambda} k.\boldsymbol{\lambda}\langle x,k'\rangle.\widetilde{\Psi}^{m}_{\langle a_0k,\ldots,a_nk,x\rangle}k') \in \mathcal{R}^{\perp\!\!\!\perp}_{\tau\to\iota}$$

Using Rem (2) above we conclude:

Lemma B.6. *Let $\tau \in \mathcal{T}$, $e \in \mathrm{E}$, $((t,e),a) \in \mathcal{R}^{\perp\!\!\!\perp}_{\iota\to(\tau\to\iota)\to\tau}$ and $((u,e),b) \in \mathcal{R}^{\perp\!\!\!\perp}_{(\iota\to\tau)\to\tau}$.*

Let moreover $n \in \mathbb{N}$ and $((s_0,e),b_0),\ldots,((s_n,e),b_n) \in \mathcal{R}^{\perp\!\!\!\perp}_\tau$.

We have:

$$((\Psi_\tau(t,u)\langle s_0,\ldots,s_n\rangle, e), \boldsymbol{\lambda} k.\widetilde{\Psi}^{\tau}_{\langle a_0k,\ldots,a_nk\rangle}(ak)(bk)) \in \mathcal{R}^{\perp\!\!\!\perp}_\iota$$

B.4 Proof of Proposition 7.3

Corollary B.7. *(i) If $\vdash t : \iota|$ in Λ_Ψ, then for all $n \in \mathbb{N}$ we have $(t,\varepsilon,\star) \succ (\overline{n}, e, \star)$ if $[t] = [n]$.*

(ii) Let $\vdash t : \iota \to \iota$ in Λ_Ψ. For all $n,m \in \mathbb{N}$, if $\boldsymbol{\lambda} k.[t]\langle[\overline{n}],k\rangle = [\overline{m}]$ then $(t\overline{n},\varepsilon,\star) \succ (\overline{m}, e, \star)$.

Proof. (i) By Thm. B.1, together with Lemmas B.3, B.4, B.5 and B.6, we have $(t,\varepsilon)\,\mathcal{R}^{\perp\!\!\!\perp}_\iota\,\boldsymbol{\lambda} k.[n]$. Since $(\star, \boldsymbol{\lambda} k.k) \in \mathcal{R}_\iota$, it follows that

$$(t,\varepsilon,\star) \perp\!\!\!\perp \boldsymbol{\lambda} k.[\overline{n}]((\boldsymbol{\lambda} k'.k')k)$$

and we are done by definition of $\perp\!\!\!\perp$ since $\boldsymbol{\lambda} k.[\overline{n}]((\boldsymbol{\lambda} k'.k')k) = \boldsymbol{\lambda} k.[\overline{n}]k = [\overline{n}]$ (note that the later holds even without using the η-rule, since $[\overline{n}] = \boldsymbol{\lambda} k.k\widetilde{n}$).

(ii) Follows from (i) since for all $n \in \mathbb{N}$, we have $((\overline{n},\varepsilon), \boldsymbol{\lambda} k.[\overline{n}]) \in \mathcal{R}^{\perp\!\!\!\perp}_\iota$ by Lemma B.5. □

C Realization of Countable Choice Using Bar-Recursion

In this Appendix, we prove Proposition 7.1. We recall its statement:

Proposition C.1. $\boldsymbol{\lambda} k.[t^{\tau,A}_{\mathsf{CAC}}] \in$
$|\forall x^\iota(\forall y^\tau(A \Rightarrow \bot) \Rightarrow \bot) \;\Rightarrow\; \forall f^{\iota\to\tau}(\forall x^\iota A[fx/y] \Rightarrow \bot) \;\Rightarrow\; \bot|$

Recall that

$$\begin{array}{lrcl} & t^{\tau,A}_{\mathsf{CAC}} & = & \lambda z.\lambda c.\Psi_{\tau\times A^*}(t_{\neg\neg\exists} z,\, \lambda a.c(\lambda x.\mathsf{p}_1(ax))(\lambda x.\mathsf{p}_2(ax)))\langle\rangle \\ \text{where} & t_{\neg\neg\exists} & = & \lambda a.\lambda x.\lambda k.ax(\lambda y.\lambda z.k\langle y, z\rangle) \end{array}$$

The main point is to decompose the notion of realizability proposed in Sect. 6 w.r.t. the relativization of quantifications. It is convenient to extend the formulas defined in Sect. 2:

$$A, B \quad ::= \quad \dots \quad | \quad \tilde{\forall} x^\tau A \quad | \quad (\mathsf{r}_\tau(a^\tau) \times A) \Rightarrow B$$

Hence, in extended formulas, the construction $(\mathsf{r}_\tau(a) \times A)$ is only allowed to appear to the left of an implication. The definition of realizability is extended as follows:

$$\begin{array}{rcl} \|\tilde{\forall} x^\tau A\| & := & \bigcup_{a\in\tau^{\mathfrak{t}}} \|A\|[a/x] \\ \|(\mathsf{r}_\tau(c) \times A) \Rightarrow B\| & := & \{\boldsymbol{\lambda} k.\langle \boldsymbol{\lambda} k'.\mathbf{case}\, k'\{c, ak\}, bk\rangle \mid a \in |A| \;\&\; b \in \|B\|\} \end{array}$$

Extended formulas and their realizability interpretation are inspired from ideas used in Krivine's Realizability [10]. We also extend the mapping $(\_)^*$ of Section 6, mapping extended formulas to simple types:

$$\begin{array}{rcl} (\tilde{\forall} x^\tau A)^* & := & A^* \\ ((\mathsf{r}_\tau(a) \times A) \Rightarrow B)^* & := & \tau \times A^* \to B^* \end{array}$$

The following is the key for Proposition C.1. The argument is the usual one for bar-recursion, see e.g. [3, 4].

Lemma C.2. *Let B such that $B \Rightarrow \bot$ is an extended formula. Assume*

$$b \in |\forall x^\iota(\tilde{\forall} y^\tau(B \Rightarrow \bot) \Rightarrow \bot)| \qquad \text{and} \qquad c \in |\tilde{\forall} f^{\iota\to\tau}(\forall x^\iota B[fx/y] \Rightarrow \bot)|$$

Then $\boldsymbol{\lambda} k.\widetilde{\Psi}^{B^*}_{\langle\rangle}(bk)(ck) \in |\bot|$.

Recall that we use notations $(\_)^\bullet$ and $(\_)^\circ$ for resp. currfication and uncurryfication. Recall also that the amount to which an expression is curryfied/uncurryfied depends on the context, and moreover that in $\mathcal{G}$, $(\_)^\bullet$ and $(\_)^\circ$ are the identity.

Proof. First, note that for all extended formula A, we have $\mathrm{ex}_{A^*} \in |\tilde{\forall}\boldsymbol{x}(\bot \Rightarrow A)|$. This can be easily proved by induction on A.

Recall that for all $a_0, \dots, a_n \in [B^*]$, we have

$$\begin{aligned}\widetilde{\Psi}^{B^*}_{\langle a_0,\dots,a_n\rangle}(bk)(ck) \quad = \quad & (ck)^\bullet(\langle a_0,\dots,a_n\rangle @ \\ & \boldsymbol{\lambda}\_.[\mathrm{ex}_{B^*}]^\bullet((bk)^\bullet\,[\overline{n+1}]\,(\boldsymbol{\lambda}x.\widetilde{\Psi}^{B^*}_{\langle a_0,\dots,a_n,x\rangle}\ (bk)\ (ck))^\circ))\end{aligned}$$

Assume now that $\boldsymbol{\lambda}k.\widetilde{\Psi}^{B^*}_{\langle\rangle}(bk)(ck) \notin |\bot|$. Since $||\bot|| = \{\boldsymbol{\lambda}k.k\}$, this means that $\boldsymbol{\lambda}k.\widetilde{\Psi}^{B^*}_{\langle\rangle}(bk)(ck)k \notin \perp\!\!\!\perp$. By assumption on c, again since $||\bot|| = \{\boldsymbol{\lambda}k.k\}$, this implies that there is $f \in (\iota \to \tau)^{\mathtt{t}}$ such that

$$\boldsymbol{\lambda}k.\boldsymbol{\lambda}\_.[\mathrm{ex}_{B^*}]^\bullet((bk)^\bullet\,[\overline{0}]\,(\boldsymbol{\lambda}x.\widetilde{\Psi}^{B^*}_{\langle x\rangle}\ (bk)\ (ck))^\circ) \quad \notin \quad |\forall x^\iota B[fx/y]|$$

By assumption on b, this implies that

$$\boldsymbol{\lambda}k.\boldsymbol{\lambda}\langle x,k'\rangle.\widetilde{\Psi}^{B^*}_{\langle x\rangle}\ (bk)\ (ck)\ k' \quad \notin \quad |\tilde{\forall} y(B[[\overline{0}]/x] \Rightarrow \bot)|$$

Hence, there is $d_0 \in \tau^{\mathtt{t}}$ and $e_0 \in [\![\iota]\!] \to [B^*]$ such that $e_0 \in |B[[\overline{0}]/x, d_0/y]|$ and

$$\boldsymbol{\lambda}k.(\boldsymbol{\lambda}\langle x,k'\rangle.\widetilde{\Psi}^{B^*}_{\langle x\rangle}\ (bk)\ (ck)\ k')\langle e_0k, (\boldsymbol{\lambda}k''.k'')k\rangle \quad \notin \quad \perp\!\!\!\perp$$

that is $\boldsymbol{\lambda}k.\widetilde{\Psi}^{B^*}_{\langle e_0k\rangle}(bk)(ck)k \notin \perp\!\!\!\perp$.

By iterating the argument (using classical choice), we obtain a sequence $(d_n, e_n)_{n\in\mathbb{N}}$ such that for all $n \in \mathbb{N}$,

(i) $d_n \in \tau^{\mathtt{t}}$,
(ii) $e_n \in |B[[\overline{n}]/x, d_n/y]|$,
(iii) $\boldsymbol{\lambda}k.\widetilde{\Psi}^{B^*}_{\langle e_0k,\dots,e_nk\rangle}(bk)(ck)k \notin \perp\!\!\!\perp$.

Let now $f \in [\iota \to \tau]$ be an innocent strategy such that $f[\overline{n}] = d_n$ for all $n \in \mathbb{N}$. Note that $f \in (\iota \to \tau)^{\mathtt{t}}$. Let moreover g be such that $g[\overline{n}] = e_n$ for all $n \in \mathbb{N}$. It follows that

$$h \quad := \quad \boldsymbol{\lambda}k.\boldsymbol{\lambda}\langle x,k'\rangle.g^\circ\langle x,k'\rangle k \quad \in \quad |\forall x^\iota B[fx/y]|$$

By assumption on c, we have $\boldsymbol{\lambda}k.\boldsymbol{\lambda}k'.ck\langle hk,k'\rangle \in |\bot|$. By continuity of c, this implies that there is $n \in \mathbb{N}$ such that

$$\begin{aligned}&\boldsymbol{\lambda}k.(ck)^\bullet(\langle e_0k,\dots,e_nk\rangle @ \\ &\qquad \boldsymbol{\lambda}\_.[\mathrm{ex}_{B^*}]^\bullet((bk)^\bullet\,[\overline{n+1}]\,(\boldsymbol{\lambda}x.\widetilde{\Psi}^{B^*}_{\langle e_0k,\dots,e_nk,x\rangle}\ (bk)\ (ck))^\circ))k \quad \in \quad \perp\!\!\!\perp\end{aligned}$$

It follows that

$$\boldsymbol{\lambda}k.\widetilde{\Psi}^{B^*}_{\langle e_0k,\dots,e_nk\rangle}\ (bk)\ (ck) \quad \in \quad \perp\!\!\!\perp$$

a contradiction. □

We can now prove Proposition C.1.

Proof (of Prop. C.1). Let A be a formula. We have to show that $\boldsymbol{\lambda} k.[t^{\tau,A}_{\mathsf{CAC}}]$ realizes the formula

$$|\forall x^\iota(\forall y^\tau(A \Rightarrow \bot) \Rightarrow \bot) \;\Rightarrow\; \forall f^{\iota\to\tau}(\forall x^\iota A[fx/y] \Rightarrow \bot) \;\Rightarrow\; \bot|$$

We apply Lemma C.2 with $B := (\mathsf{r}_\tau(y) \times A)$, and obtain that $\boldsymbol{\lambda} k.\widetilde{\Psi}^{\tau\times A^*}_{\langle\rangle}(bk)(ck) \in |\bot|$ provided

$$\begin{array}{rl} & b \;\in\; |\forall x^\iota(\tilde{\forall} y^\tau((\mathsf{r}_\tau(y) \times A) \Rightarrow \bot) \Rightarrow \bot)| \\ \text{and} & c \;\in\; |\tilde{\forall} f^{\iota\to\tau}(\forall x^\iota(\mathsf{r}_\tau(fx) \times A[fx/y]) \Rightarrow \bot)| \end{array}$$

In order to conclude, it remains to show the two following points:

$$\begin{array}{l} \boldsymbol{\lambda} k.[\lambda a.\lambda x.\lambda b.ax(\lambda y.\lambda z.b\langle y,z\rangle)] \;\in \\ \qquad |\forall x^\iota(\forall y^\tau(A \Rightarrow \bot) \Rightarrow \bot) \;\Rightarrow\; \forall x^\iota(\tilde{\forall} y^\tau((\mathsf{r}_\tau(y) \times A) \Rightarrow \bot) \Rightarrow \bot)| \end{array} \tag{1}$$

and

$$\begin{array}{l} \boldsymbol{\lambda} k.[\lambda c.\lambda a.c(\lambda x.\mathsf{p}_1(ax))(\lambda x.\mathsf{p}_2(ax))] \;\in \\ \qquad |\forall f^{\iota\to\tau}(\forall x^\iota A[fx/y] \Rightarrow \bot) \;\Rightarrow\; \tilde{\forall} f^{\iota\to\tau}(\forall x^\iota(\mathsf{r}_\tau(fx) \times A[fx/y]) \Rightarrow \bot)| \end{array} \tag{2}$$

1. Let $a \in |\forall x^\iota(\forall y^\tau(A \Rightarrow \bot) \Rightarrow \bot)|$, $n \in \mathbb{N}$ and $b \in |\tilde{\forall} y^\tau((\mathsf{r}_\tau(y) \times A) \Rightarrow \bot)|$. We have to show that

   $$\boldsymbol{\lambda} k.[\lambda a.\lambda x.\lambda b.ax(\lambda y.\lambda z.b\langle y,z\rangle)]\langle ak, [\overline{n}], bk, k\rangle \;\in\; \perp\!\!\!\perp$$

   i.e. that

   $$\boldsymbol{\lambda} k.ak\langle[\overline{n}], \boldsymbol{\lambda}\langle y,z,k'\rangle.bk\langle \boldsymbol{\lambda} k''.\textsf{\textbf{case}}\, k''\{y,z\}, k'\rangle, k\rangle \;\in\; \perp\!\!\!\perp$$

   Hence we are done if

   $$\boldsymbol{\lambda} k.\boldsymbol{\lambda}\langle y,z,k'\rangle.bk\langle \boldsymbol{\lambda} k''.\textsf{\textbf{case}}\, k''\{y,z\}, k'\rangle \;\in\; |\forall y^\tau(A \Rightarrow \bot)|$$

   But if $c \in \tau^{\mathtt{t}}$ and $d \in |A|$, we have

   $$\boldsymbol{\lambda} k.\langle \boldsymbol{\lambda} k'.\textsf{\textbf{case}}\, k'\{c, dk\}, k\rangle \in ||(\mathsf{r}_\tau(c) \times A) \Rightarrow \bot||$$

   and we are done by assumption on b.
2. The proof is similar to that of (1). $\square$

Table of Contents