Realizability

Thomas Streicher

WS 17/18

Contents

1 Introduction

Realizability was invented in 1945 by S. C. Kleene as an attempt to make explicit the *algorithmic content* of *constructive proofs*.
From proofs of existence statements $\exists y R(\vec{x}, y)$ one would like to read off a so-called *Skolem function*, i.e. a function f such that $R(\vec{x}, f(\vec{x}))$ holds for all $\vec{x}$. Assuming (a mild form of) axiom of choice such an f always exists if $\exists y R(\vec{x}, y)$ holds. However, in general such an f will not be computable: if P is an undecidable property of natural numbers then $\exists y\ (y{=}0 \wedge P(x)) \vee (y{=}1 \wedge \neg P(x))$ although there cannot exist an algorithmic Skolem function $f : \mathbb{N} \to \{0, 1\}$ with $\forall x\ (f(x){=}0 \wedge P(x)) \vee (f(x){=}1 \wedge \neg P(x))$ as otherwise f would give rise to a decision procedure for the predicate P. But even if $\vec{x}$ is empty from provability of $\exists x A(x)$ it does not necessarily follow that there is a constant c for which $A(c)$ is provable. For example let P be a decidable predicate of natural numbers such that $\forall x \neg P(x)$ holds but is not provable[1] then $\exists x\, (\neg P(x) \to \forall y \neg P(y))$ is provable (already in classical predicate logic) but for no natural number n one can prove $\neg P(n) \to \forall y \neg P(y)$ as it is logically equivalent to the unprovable statement $\forall y \neg P(y)$ (because $\neg P(n)$ is trivially provable).
These examples show that classical proofs of $\exists x A(x)$ do not always give rise to *witnesses*, i.e. objects c for which $A(c)$ is provable. The very idea of *constructive* (or *intuitionistic*) logic (introduced by L. E. J. Brouwer at the beginning of the 20th century) was to restrict the rules and axioms of logic in such a way that

(E) whenever $\exists x A(x)$ is provable then $A(t)$ is provable for some term t

(D) if $A \vee B$ is provable then A is provable or B is provable (or both).

Actually these requirements form part of an *informal semantics* of constructive logic which has come to be widely known under the name of

Brouwer-Heyting-Kolmogorov (BHK) Interpretation

1. a witness of $A \wedge B$ is a pair $\langle p, q\rangle$ such that p is a witness of A and q is a witness of B
2. a witness of $A \to B$ is a function p mapping any witness q of A to a witness $p(q)$ of B
3. a witness of $A \vee B$ is a pair $\langle i, p\rangle$ such that either $i = 0$ and p is a witness of A or $i = 1$ and p is a witness of B
4. a witness of $\forall x A(x)$ is a function p mapping any object c to a witness $p(c)$ of $A(c)$
5. a witness of $\exists x A(x)$ is a pair $\langle c, p\rangle$ such that p is a witness of $A(c)$
6. there is no witness for $\perp$ (falsity).

[1] According to Gödel's 2nd Incompleteness Theorem one could take for $P(x)$ the predicate saying that x codes a derivation of 0=1 in the formal system under consideration.

For basic assertions A it is intentionally left unspecified what are their witnesses. Typically, e.g. in arithmetic, the witness for $n = m$ is either a basic unspecified object $*$ if $n = m$ or there is no witness at all if $n \neq m$.

Notice that "being a witness of a proposition" is a basic notion that cannot be further analyzed but this also applies to the notion of "truth of a proposition" as employed in the usual informal explanation of classical logic à la Tarski. Whereas the meaning explanation à la Tarski is usually called *truth value semantics* the meaning explanation à la Brouwer-Heyting-Kolmogorov may be called a *proof semantics* as it specifies for every proposition A what is a "proof" or – as we say – "witness" of A.[2] Notice, however, that such a "witness" shouldn't be thought of as a formal derivation as every true Π^0_1 sentence $\forall x.t{=}s$ is witnessed by the function $\lambda x.*$.[3]

The basic idea of *realizability* is to provide mathematically precise instantiations of the BHK interpretation where the informal notion of "witness" is replaced by a particular mathematical structure $\mathcal{A}$ which can be understood as a (universal) *untyped model of computation.* Having fixed such an $\mathcal{A}$ propositions are interpreted as subsets of $\mathcal{A}$, i.e. a proposition A is identified with the set of its witnesses (in $\mathcal{A}$).

We assume that $\mathcal{A}$ is a non-empty set of "algorithms" together with a partial binary operation on $\mathcal{A}$ where $a \cdot b$ is thought of as the result of applying algorithm a to b.[4] A (conservative) choice is taking $\mathbb{N}$ for $\mathcal{A}$ and defining $n \cdot m$ as Kleene application $\{n\}(m)$, i.e. n-th partial recursive function applied to m.[5] The only assumption about the structure $(\mathcal{A}, \cdot)$ is that for every polynomial $t[x_1, \ldots, x_n, x]$ there is a polynomial $\Lambda x.t[x_1, \ldots, x_n, x]$ in the variables $x_1, \ldots, x_n$ such that for all $a_1, \ldots, a_n, a \in \mathcal{A}$ it holds that $(\Lambda x.t[a_1, \ldots, a_n, x]) \cdot a = t[a_1, \ldots, a_n, a]$ whenever $t[a_1, \ldots, a_n, a]\downarrow$, i.e. whenever $t[a_1, \ldots, a_n, a]$ is defined. Notice, however, that we do *not* require that definedness of $(\Lambda x.t[a_1, \ldots, a_n, x]) \cdot a$ implies definedness of $t[a_1, \ldots, a_n, a]$ (although for the "first Kleene algebra", i.e. $\mathbb{N}$ with Kleene application, and most other $\mathcal{A}$ we will encounter such a choice will be possible!).

Now given such an untyped model $(\mathcal{A}, \cdot)$ of computation, usually called a *pca* (acronym for *p*artial *c*ombinatory *a*lgebra), one may build a category $\mathbf{Asm}(\mathcal{A})$ of so-called *assemblies over* $\mathcal{A}$ which has got enough structure to interpret most of *higher order intuitionistic logic* (HOIL). An assembly (over $\mathcal{A}$) is a pair $X = (|X|, ||\cdot||_X)$ where X is a set and $||\cdot||_X : |X| \to \mathcal{P}(\mathcal{A})$ such that $||x||_X \neq \emptyset$ for all $x \in |X|$. The non-empty subset $||x||_X$ of $\mathcal{A}$ is thought of as the set or *realizers* or codes for the element $x \in |X|$. We also write $a \Vdash_X x$ (speak "a realizes x") for $a \in ||x||_X$. If X and Y are assemblies over $\mathcal{A}$ then a morphism from X to Y in $\mathbf{Asm}(\mathcal{A})$ is a (set-theoretic) function $f : |X| \to |Y|$ which is *realized* or *tracked*

[2] We prefer to use the more neutral word "witness" rather than "proof" as the latter might be (mis)understood as "formal derivation" which is definitly *not* what we have in mind!

[3] Only when we *formalize* realizability one may reasonably ask whether it is *provable* (in the formal system under consideration) that $\lambda x.*$ is a witness for $\forall x.t{=}s$.

[4] The operation $\cdot$ is assumed as potentially partial because the evaluation of $a \cdot b$ may fail to terminate. Moreover, we do not distinguish between algorithms and data and, accordingly, everything is thrown into a single set $\mathcal{A}$.

[5] We employ Kleene's notation $\{n\}$ for the n-th partial recursive function.

by an element $e \in \mathcal{A}$ meaning that $\forall x \in |X|\ \forall a \in ||x||_X\ e \cdot a \downarrow \wedge e \cdot a \in ||f(x)||_Y$. We write also $e \Vdash f$ for "f is realized by e". Intuitively, the function f is realizable iff it can be implemented (in terms of codes) by an algorithm from $\mathcal{A}$. The set of realizable maps from X to Y can itself be organized into an assemply Y^X with $|Y^X| = \mathbf{Asm}(\mathcal{A})(X, Y)$ and $||f||_{Y^X} = \{e \in \mathcal{A} \mid e \Vdash f\}$.
An interesting and most useful full subcategory of $\mathbf{Asm}(\mathcal{A})$ is the category $\mathbf{Mod}(\mathcal{A})$ whose objects are those assemblies X where $x = x'$ whenever $e \in ||x||_X \cap ||x'||_X$. The objects of $\mathbf{Mod}(\mathcal{A})$ are called *modests sets* (over $\mathcal{A}$). The intuition behind this notion is that elements of modest sets are determined uniquely by their realizers. Accordingly, a modest set X can be understood as a *partially enumerated set* : let $C_X = \{a \in \mathcal{A} \mid \exists x \in |X|\ a \in ||x||_X\}$ and $\varepsilon_X : C_X \to |X|$ be the (surjective!) function sending $e \in C_X$ to the unique element $\varepsilon_X(e) \in |X|$ with $e \Vdash \varepsilon_X(e)$.

The main aim of these lectures is to demonstrate that

- $\mathbf{Asm}(\mathcal{A})$ has enough structure for interpreting constructive logic and mathematics and

- $\mathbf{Mod}(\mathcal{A})$ is a well-behaved full subcategory of $\mathbf{Asm}(\mathcal{A})$ containing all *data types* needed for (functional) computation.

2 Kleene's Number Realizability

Although the emphasis of this course is on realizability *models* in this introductory chapter we present Kleene's original account of *number realizability* which was motivated rather by *proof-theoretic* aims, namely the *extraction of algorithms from constructive proofs*.
Kleene's idea was to associate with every closed formula A of arithmetic a predicate on natural numbers telling which n realize A. He defined his notion of *number realizability* by recursion on the structure of A as follows

- n realizes $t = s$ iff $t = s$
- n realizes $A \wedge B$ iff $\mathrm{fst}(n)$ realizes A and $\mathrm{snd}(n)$ realizes B
- n realizes $A \to B$ iff for every m realizing A the computation $\{n\}(m)$ terminates and its result realizes B
- n realizes $A \vee B$ iff $\mathrm{fst}(n) = 0$ and $\mathrm{snd}(n)$ realizes A or $\mathrm{fst}(n) \neq 0$ and $\mathrm{snd}(n)$ realizes B
- n realizes $\forall x.A(x)$ iff for all numbers m the computation $\{n\}(m)$ terminates and its result realizes $A(m)$
- n realizes $\exists x.A(x)$ iff $\mathrm{snd}(n)$ realizes $A(\mathrm{fst}(n))$

where fst and snd are prim. rec. projections for some prim. rec. pairing function $\langle\cdot,\cdot\rangle : \mathbb{N} \times \mathbb{N} \xrightarrow{\cong} \mathbb{N}$ (i.e. $\langle \mathrm{fst}(n), \mathrm{snd}(n)\rangle = n$ for all $n \in \mathbb{N}$). Obviously, these clauses are quite similar to those of the BHK interpretation but more specific in the sense that a) witnesses are bound to be natural numbers and b) application of witnesses is given by Kleene application. Notice that a Π_2 sentence $\forall x \exists y\, R(x, y)$ (where $R(x, y) \equiv r(x, y) = 0$ for some prim. rec. function r) is realized by e iff for all $n \in \mathbb{N}$ the computation $\{e\}(n)$ terminates with a value m such that $R(n, \mathrm{fst}(m))$ holds (and is realized by $\mathrm{snd}(m)$). Thus e realizes $\forall x \exists y\, R(x, y)$ iff e is the Gödel number of an algorithm such that $\Lambda n.\, \mathrm{fst}(\{e\}(n))$ computes a Skolem function for this sentence. Notice that the sentence $0 = 1$ has no realizer at all and, therefore, can be taken as the *false proposition* also denoted as $\perp$. As usual in constructive logic negation is *defined* as $\neg A \equiv A \to \perp$. We have e realizes $\neg A$ iff from n realizes A it follows that $\{e\}(n)$ terminates and realizes $\perp$. As no number realizes $\perp$ we have that e realizes $\neg A$ iff there is no realizer for A. Accordingly, e realizes $\neg\neg A$ iff there is some realizer for A. Thus negated formulas have no realizer at all or are realized by all numbers. Accordingly, from realizers of negated formulas one cannot read off any computational content at all.
An example of a classically provable formula that is not realizable is

$$A \equiv \forall x\ \{x\}(x)\downarrow \vee\ \neg\{x\}(x)\downarrow$$

with $\{x\}(y)\downarrow$ standing for $\exists z\, T(x, y, z)$ where T is Kleene's T predicate (see [Ro]). Now if e were a realizer for A then $\Lambda n.\mathrm{fst}(\{e\}(n))$ would give rise to an algorithm deciding the halting problem which is clearly impossible. Thus A is not realizable and accordingly $\neg A$ is realized by all natural numbers. This illustrates how classically wrong propositions may well be realizable.

Actually, for every arithmetical formula A the predicate "n realizes A" on n can itself be expressed in the language of arithmetic. That's done in the next definition where we also drop the assumption that A is a closed formula.

Definition 2.1. (formalized number realizability)
*The realizability relation n **rn** A is defined by induction on the structure of A via the following clauses*

$$\begin{array}{lcl}
n\,\mathbf{rn}\,P & \equiv & P \qquad \textit{where } P \textit{ is atomic} \\
n\,\mathbf{rn}\,A \wedge B & \equiv & \mathrm{fst}(n)\,\mathbf{rn}\,A \;\wedge\; \mathrm{snd}(n)\,\mathbf{rn}\,B \\
n\,\mathbf{rn}\,A \to B & \equiv & \forall m.(m\,\mathbf{rn}\,A \to \{n\}(m)\!\downarrow \wedge \{n\}(m)\,\mathbf{rn}\,B) \\
n\,\mathbf{rn}\,A \vee B & \equiv & \big(\mathrm{fst}(n) = 0 \to \mathrm{snd}(n)\,\mathbf{rn}\,A\big) \wedge \big(\mathrm{fst}(n) \neq 0 \to \mathrm{snd}(n)\,\mathbf{rn}\,B\big) \\
n\,\mathbf{rn}\,\forall x.A(x) & \equiv & \forall m.\{n\}(m)\!\downarrow \wedge \{n\}(m)\,\mathbf{rn}\,A(m) \\
n\,\mathbf{rn}\,\exists x.A(x) & \equiv & \mathrm{snd}(n)\,\mathbf{rn}\,A(\mathrm{fst}(n))
\end{array}$$

where in $n\,\mathbf{rn}\,A$ the variable n is (tacitly) assumed not to be free in A. $\diamondsuit$

Notice that when expanding the defining clauses for implication and universal quantification according to the conventions introduced in Appendix A we get

$$\begin{array}{lcl}
n\,\mathbf{rn}\,A \to B & \equiv & \forall m.\, m\,\mathbf{rn}\,A \to \exists k.\, T(n,m,k) \wedge U(k)\,\mathbf{rn}\,B \\
n\,\mathbf{rn}\,\forall x.A(x) & \equiv & \forall m.\exists k.\, T(n,m,k) \wedge U(k)\,\mathbf{rn}\,A(m)
\end{array}$$

which are more explicit but also less readable.
It is desirable to show that *whenever A is provable then there exists a natural number e such that $e\,\mathbf{rn}\,A$ is provable as well.* Of course, such a statement depends on what is meant by "provable".
For the purpose of making "provable" precise one usually considers the formal system **HA** (Heyting Arithmetic) and extensions of it.[6] The underlying (first order) language of **HA** consists of symbols for every (definition of a) primitive recursive function (see Def. A.1). Thus, in particular, we have a constant 0 and a unary function symbol succ (for the successor operation). For every natural number n there is a term $\mathsf{succ}^n(0)$, the numeral for n, which for sake of readability[7] we also denote by n. Heyting arithmetic **HA** is based on constructive or intuitionistic logic for which formal systems can be found in Appendix B. The non-logical axioms of **HA** (besides the usual equality axioms[8]) consist of

[6] In the proof theoretic literature one often finds also subsystems of **HA** where the induction schema is restricted to formulas of a certain logical complexity, e.g. restriction to quantifier-free formulas gives rise to **PRA** (Primitive Recursive Arithmetic) whose provably total recursive functions are precisely the primitive recursive ones.

[7] Often in the literature (e.g. the papers by A. S. Troelstra cited in the references) one finds $\underline{n}$ as a notation for $\mathsf{succ}^n(0)$. This is certainly more precise but also more cumbersome.

[8] namely $x = x$ and $A[x] \wedge x = y \to A[y]$

(1) defining equations for primitive recursive function definitions

(2) *Induction Scheme* $\quad A(0) \wedge \forall x \left(A(x) \to A(\mathsf{succ}(x))\right) \to \forall x A(x)$

(3) $\neg\, 0 = \mathsf{succ}(x)$.

In the induction scheme A may be instantiated with an arbitrary predicate expressible in the language of **HA**. The third axiom is needed for ensuring that not all numbers are equal.[9]

For understanding the formulation of the following Soundness Theorem recall the notational conventions introduced in Appendix A.

Theorem 2.1. (Soundness of Number Realizability)
If a closed formula A can be derived in **HA** *then there is a term e built up from constants for primitive recursive functions, Kleene application and Λ-abstraction such that $e \,\mathbf{rn}\, A$ can be derived in* **HA**.

Proof. As we want to prove soundness by induction on the structure of derivations in **HA** we have to generalise our claim as follows: whenever $A_1, \ldots, A_n \vdash A$ is derivable in **HA** then there is a term e such that **HA** proves

$$u_1 \,\mathbf{rn}\, A_1 \wedge \ldots \wedge u_n \,\mathbf{rn}\, A_n \vdash e \,\mathbf{rn}\, A$$

where the variables u_i are fresh and e is a term built from constants for primitive recursive functions, Kleene application $\{\cdot\}(\cdot)$, Λ-abstraction and variables from $FV(A_1, \ldots, A_n, A) \cup \{u_1, \ldots, u_n\}$.
For sake of readability we often write $\vec{u} \,\mathbf{rn}\, \Gamma$ for $u_1 \,\mathbf{rn}\, A_1 \wedge \ldots \wedge u_n \,\mathbf{rn}\, A_n$ when $\Gamma \equiv A_1, \ldots, A_n$.
It is easy to show that the generalised claim holds for the structural rules (ax), (ex), (w) and (c) as primitive recursive functions contain all projections and are closed under permutation of arguments, addition of dummy arguments and identification of arguments.
$(\wedge I)$ If **HA** proves $\vec{u} \,\mathbf{rn}\, \Gamma \vdash e_1 \,\mathbf{rn}\, A$ and $\vec{u} \,\mathbf{rn}\, \Gamma \vdash e_2 \,\mathbf{rn}\, B$ then **HA** proves $\vec{u} \,\mathbf{rn}\, \Gamma \vdash \langle e_1, e_2\rangle \,\mathbf{rn}\, A \wedge B$.
$(\wedge E)$ If **HA** proves $\vec{u} \,\mathbf{rn}\, \Gamma \vdash e \,\mathbf{rn}\, A \wedge B$ then **HA** proves $\vec{u} \,\mathbf{rn}\, \Gamma \vdash \mathrm{fst}(e) \,\mathbf{rn}\, A$ and $\vec{u} \,\mathbf{rn}\, \Gamma \vdash \mathrm{snd}(e) \,\mathbf{rn}\, B$.
$(\to I)$ If **HA** proves $\vec{u}, v \,\mathbf{rn}\, \Gamma, A \vdash e \,\mathbf{rn}\, B$ then $\vec{u} \,\mathbf{rn}\, \Gamma \vdash \Lambda v.e \,\mathbf{rn}\, A \to B$ can be proved in **HA**.
$(\to E)$ If **HA** proves $\vec{u} \,\mathbf{rn}\, \Gamma \vdash e_1 \,\mathbf{rn}\, A \to B$ and $\vec{u} \,\mathbf{rn}\, \Gamma \vdash e_2 \,\mathbf{rn}\, A$ then **HA** proves $\vec{u} \,\mathbf{rn}\, \Gamma \vdash \{e_1\}(e_2) \,\mathbf{rn}\, B$.
$(\bot E)$ Suppose that **HA** proves $\vec{u} \,\mathbf{rn}\, \Gamma \vdash e \,\mathbf{rn}\, \bot$. Then **HA** proves $\vec{u} \,\mathbf{rn}\, \Gamma \vdash \bot$ because $e \,\mathbf{rn}\, \bot$ is provably equivalent to $\bot$. Thus $\vec{u} \,\mathbf{rn}\, \Gamma \vdash 0 \,\mathbf{rn}\, A$ can be proved in **HA**.
$(\forall I)$ Suppose that **HA** proves $\vec{u} \,\mathbf{rn}\, \Gamma \vdash e \,\mathbf{rn}\, A(x)$ where $x \notin FV(\Gamma)$. Then **HA** proves $\vec{u} \,\mathbf{rn}\, \Gamma \vdash \Lambda x.e \,\mathbf{rn}\, \forall x.A(x)$.

[9] That succ is injective can be proved in **HA** because due to the defining equations for the predecessor function pred from $\mathsf{succ}(x) = \mathsf{succ}(y)$ it follows that $x = \mathsf{pred}(\mathsf{succ}(x)) = \mathsf{pred}(\mathsf{succ}(y)) = y$.

($\forall E$) If **HA** proves $\vec{u}\,\mathbf{rn}\,\Gamma \vdash e\,\mathbf{rn}\,\forall x.A(x)$ then $\vec{u}\,\mathbf{rn}\,\Gamma \vdash \{e\}(t)\,\mathbf{rn}\,A(t)$ is provable in **HA**.
($\exists I$) If **HA** proves $\vec{u}\,\mathbf{rn}\,\Gamma \vdash e\,\mathbf{rn}\,A(t)$ then $\vec{u}\,\mathbf{rn}\,\Gamma \vdash \langle t, e\rangle\,\mathbf{rn}\,\exists x.A(x)$ can be proved in **HA**.
($\exists E$) Suppose that **HA** proves $\vec{u}\,\mathbf{rn}\,\Gamma \vdash e_1\,\mathbf{rn}\,\exists x.A(x)$ and $\vec{u}, u\,\mathbf{rn}\,\Gamma, A(x) \vdash e_2\,\mathbf{rn}\,B$ where $x \notin FV(B)$. Then $\vec{u}\,\mathbf{rn}\,\Gamma \vdash e_2[\mathrm{fst}(e_1), \mathrm{snd}(e_1)/x, u]\,\mathbf{rn}\,B$ can be proved in **HA**.
($\vee I$) and ($\vee E$) are left as exercises.
It remains to check that the axioms of **HA** are realized. This is trivial for the equations as these are realized by any number (e.g. 0). The axiom $\neg\,\mathsf{succ}(x) = 0$ is realized e.g. by $\Lambda n.0$.
Next we consider instances of the induction scheme. First of all notice that there exists[10] a number r such that

$$\{\{r\}(\langle e_0, e_1\rangle)\}(0) = e_0 \quad \{\{r\}(\langle e_0, e_1\rangle)\}(k{+}1) \simeq \{\{e_1\}(k)\}(\{\{r\}(\langle e_0, e_1\rangle)\}(k))$$

holds for all numbers e_0, e_1 and k and these properties can be verified in **HA**. Now, for a predicate $A(x)$ with free variables $\vec{z}$ besides x one can prove in **HA** that $r\,\mathbf{rn}\,A(0) \wedge (\forall x.(A(x) \to A(\mathsf{succ}(x)))) \to \forall x.A(x)$, i.e. that r realizes the induction scheme. □

Now one might hope that for every formula A one can prove in **HA** the equivalence $A \leftrightarrow \exists x.x\,\mathbf{rn}\,A$ or at least that[11] $\mathbf{HA} \vdash A$ iff $\mathbf{HA} \vdash \exists x.x\,\mathbf{rn}\,A$. Alas, this hope is in vain since for

$$\mathrm{CT}_0 \qquad (\forall x.\exists y.A(x,y)) \to \exists e.\forall x.A(x, \{e\}(x))$$

we have $\mathbf{HA} \vdash \exists x.\,x\,\mathbf{rn}\,\mathrm{CT}_0$, but CT_0 cannot be proved in **HA** as CT_0 cannot be proved in **PA** since for some instance of CT_0 its negation can be proved in **PA** (Exercise!). However, for an Extended Church's Thesis ECT_0 defined subsequently we can achieve our goal, namely prove that

Theorem 2.2. (Characterisation of Number Realizability)
For all formulas A of **HA** *it holds that*

(1) $\mathbf{HA} + \mathrm{ECT}_0 \vdash A \leftrightarrow \exists x.\,x\,\mathbf{rn}\,A$

(2) $\mathbf{HA} + \mathrm{ECT}_0 \vdash A$ *iff* $\mathbf{HA} \vdash \exists x.\,x\,\mathbf{rn}\,A$.

In order to formulate ECT_0 we have to introduce the following notion.

Definition 2.2. *The* almost negative *or* almost $\exists$-free *formulas are those which can be built from atomic formulas and formulas of the form $\exists x.\,t{=}s$ by $\wedge$, $\to$ and $\forall$.* ◇

[10] This is a typical argument by appeal to Church's Thesis. One can easily exhibit an algorithm for the primitive recursion operator R in any programming language whatsoever and, therefore, this algorithm has a Gödel number, say r.
[11] We employ the notation $\mathbf{HA} \vdash A$ for the meta-mathematical statement that **HA** proves the sequent $\vdash A$.

Now we can formulate the *Extended Church's Thesis*

$\mathrm{ECT}_0 \quad \forall x.\big(A(x){\to}\exists y.B(x,y)\big) \to \exists e.\forall x.\big(A(x) \to \{e\}(x){\downarrow} \wedge B(x,\{e\}(x))\big)$

where A is required to be *almost negative*. Using the notational conventions of Appendix A one can reformulate ECT_0 as

$$\forall x.\big(A(x){\to}\exists y.B(x,y)\big) \to \exists e.\forall x.\big(A(x) \to \exists z.T(e,x,z) \wedge \wedge B(x,U(z))\big)$$

for almost negative A.
Before proving Theorem 2.2 we have to establish some useful properties of almost negative formulas.
By inspection of the defining clauses for number realizability (Def. 2.1) it is evident that for all formulas A the formula $x\,\mathbf{rn}\,A$ is provably equivalent to an almost negative formula (by eliminating all occurrences of $\{n\}(m)$ as described in Appendix A).
Next we show that almost negative formulas A are equivalent to $\exists x.\, x\,\mathbf{rn}\,A$ and that this equivalence can be proved in **HA**.

Lemma 2.1. *For almost negative formulas A it holds that*

(1) $\mathbf{HA} \vdash (\exists x.\, x\,\mathbf{rn}\,A) \to A$ *and*

(2) *there is a term ψ_A with $\mathbf{HA} \vdash A \to \psi_A\,\mathbf{rn}\,A$*

and, therefore, that $\mathbf{HA} \vdash A \leftrightarrow \exists x.\, x\,\mathbf{rn}\,A$.

Proof. We prove (1) and (2) simultaneously by induction on the structure of almost negative formulas.
For primitive formulas $t{=}s$ we have that $\exists x.\, x\,\mathbf{rn}\,t{=}s$ equals $\exists x.\, t{=}s$ which is equivalent to $t{=}s$ as x is not free in $t{=}s$. Thus, (1) holds for $t{=}s$. Claim (2) holds for $t{=}s$ by putting $\psi_{t=s} \equiv 0$.
For formulas of the form $\exists x.\, t{=}s$ we have that

$$x\,\mathbf{rn}\,\exists x.\, t{=}s \;\equiv\; \mathrm{snd}(x)\,\mathbf{rn}\,t{=}s[\mathrm{fst}(x)/x]$$

and, therefore, one easily proves $x\,\mathbf{rn}\,\exists x.\, t{=}s \to \exists x.\, t{=}s$. For claim (2) one puts $\psi_{\exists x.t=s} \equiv \langle \mu x.t{=}s, 0\rangle$ where $\mu x.t{=}s$ is the (Gödel number of an) algorithm searching for the least x satisfying the decidable condition $t{=}s$. Obviously, $\mu x.t{=}s$ terminates if $\exists x.t{=}s$ and, therefore, **HA** proves that $\exists x.\, t{=}s \to 0\,\mathbf{rn}\,t{=}s[\mu x.t{=}s/x]$. But as $0\,\mathbf{rn}\,t{=}s[\mu x.t{=}s/x]$ is easily seen to be equivalent to $\langle \mu x.t{=}s, 0\rangle\,\mathbf{rn}\,\exists x.\, t{=}s$ it follows that $\mathbf{HA} \vdash \exists x.\, t{=}s \to \psi_{\exists x.t=s}\,\mathbf{rn}\,\exists x.\, t{=}s$.
Suppose as induction hypothesis that the almost negative formulas A and B satisfy the claims (1) and (2).
Then claim (1) holds for $A{\wedge}B$ as $y\,\mathbf{rn}\,A \to A$ and $z\,\mathbf{rn}\,B \to B$ hold by induction hypothesis and thus also $(\mathrm{fst}(x)\,\mathbf{rn}\,A \wedge \mathrm{snd}(x)\,\mathbf{rn}\,B) \to A{\wedge}B$, i.e. $x\,\mathbf{rn}\,A{\wedge}B \to A{\wedge}B$. Claim (2) for $A \wedge B$ follows readily by putting $\psi_{A\wedge B} \equiv \langle \psi_A, \psi_B\rangle$.
Now we show (1) for $A{\to}B$. Suppose $x\,\mathbf{rn}\,A{\to}B$, i.e. $\forall y.\, y\,\mathbf{rn}\,A \to \{x\}(y)\,\mathbf{rn}\,B$. As by induction hypothesis $A \to \psi_A\,\mathbf{rn}\,A$ we get that $A \to \{x\}(\psi_A)\,\mathbf{rn}\,B$ and

as $z\,\mathbf{rn}\,B \to B$ by induction hypothesis for B it follows that $A{\to}B$. As this argument can be formalised in **HA** it follows that $\mathbf{HA} \vdash x\,\mathbf{rn}\,A{\to}B \to A{\to}B$ and we have established claim (1) for $A{\to}B$. Claim (2) for $A{\to}B$ follows by putting $\psi_{A\to B} \equiv \Lambda x.\psi_B$ using that by induction hypothesis we have $x\,\mathbf{rn}\,A \to A$ and $B \to \psi_B\,\mathbf{rn}\,B$.
We leave the case of the universal quantifier as an exercise.
As (2) entails that $\mathbf{HA} \vdash A \to \exists x.\, x\,\mathbf{rn}\,A$ for almost negative A it follows from (1) and (2) that $\mathbf{HA} \vdash A \leftrightarrow \exists x.\, x\,\mathbf{rn}\,A$ for almost negative A. ☐

The following *idempotency* of formalized realizability appears as a corollary.

Corollary 2.1. *For every formula A in the language of* **HA** *it holds that* $\mathbf{HA} \vdash \exists x.\, x\,\mathbf{rn}\,A \leftrightarrow \exists x.\, x\,\mathbf{rn}\,(\exists x.\, x\,\mathbf{rn}\,A)$.

Proof. Straightforward exercise using Lemma 2.1 and that $x\,\mathbf{rn}\,A$ is provably equivalent to an almost negative formula. ☐

Using Lemma 2.1 one can now show that

Lemma 2.2. *For every instance A of* ECT_0 *we have* $\mathbf{HA} \vdash \exists e.\, e\,\mathbf{rn}\,A$.

Proof. Let A be almost negative. Suppose that $e\,\mathbf{rn}\,\forall x\big(A(x) \to \exists y.B(x,y)\big)$, i.e. that

$$\forall x,n.\big(n\,\mathbf{rn}\,A(x) \to \exists z.\, T(\{e\}(x),n,z) \wedge U(z)\,\mathbf{rn}\,\exists y.B(x,y)\big)$$

Substituting ψ_A for n we get

$$\forall x.\big(\psi_A\,\mathbf{rn}\,A(x) \to \exists z.\, T(\{e\}(x),\psi_A,z) \wedge U(z)\,\mathbf{rn}\,\exists y.B(x,y)\big)$$

As A is almost negative from Lemma 2.1 we get $n\,\mathbf{rn}\,A(x) \to \psi_A\,\mathbf{rn}\,A(x)$ and, therefore, we have

$$\forall x,n.\big(n\,\mathbf{rn}\,A(x) \to \exists z.\, T(\{e\}(x),\psi_A,z) \wedge U(z)\,\mathbf{rn}\,\exists y.B(x,y)\big)$$

i.e.

$$\forall x,n.\big(n\,\mathbf{rn}\,A(x) \to \exists z.\, T(\{e\}(x),\psi_A,z) \wedge \mathrm{snd}(U(z))\,\mathbf{rn}\,B(x,\mathrm{fst}(U(z)))\big)$$

Let $t_1[e] \equiv \Lambda x.\mathrm{fst}(\{\{e\}(x)\}(\psi_A))$. As

$$\forall x\big(A(x) \to \exists z.\ T(t_1[e],x,z) \wedge B(x,U(z))\big)$$

is realized by $t_2[e] \equiv \Lambda x.\Lambda n.\langle \mu z.T(t_1[e],x,z), \langle 0, \mathrm{snd}(\{\{e\}(x)\}(\psi_A))\rangle\rangle$ we finally get that $\Lambda e.\langle t_1[e], t_2[e]\rangle$ realizes

$$\forall x.\big(A(x){\to}\exists y.B(x,y)\big) \to \exists e.\forall x.\big(A(x){\to}\exists z.\, T(e,x,z){\wedge}B(x,U(z))\big)$$

as desired.
As the whole argument can be formalized within **HA** the claim follows. ☐

The assumption that A is almost negative has been used for making the choice of y with $B(x,y)$ independent from the realizer of the premiss A. Actually, adding the unrestricted[12] scheme

$$\mathrm{ECT}_0^* \qquad (\forall x.\, A \to \exists y. B(x,y)) \to \exists e. \forall x.\, A \to \exists z. T(e,x,z) \wedge B(x,U(z))$$

to **HA** is inconsistent as can be seen when instantiating A by $\exists z.\, T(x,x,z) \vee \neg\exists z.\, T(x,x,z)$ and $B(x,y)$ by $(y{=}0 \wedge \exists z. T(x,x,z)) \vee (y{=}1 \wedge \neg\exists z.\, T(x,x,z))$ (*cf.* the Remark on p.197 of [Tr73]).[13]

Now we are ready to give the

Proof of Theorem 2.2:
(1) We show that $\mathbf{HA} + \mathrm{ECT}_0 \vdash A \leftrightarrow \exists x.\, x\,\mathbf{rn}\,A$ by induction on the structure of formulas A in **HA**.
Condition (1) is obvious for atomic formulas.
($\wedge$) Obviously, $\exists x.\, x\,\mathbf{rn}\,A{\wedge}B \leftrightarrow \exists x. x\,\mathbf{rn}\,A \wedge \exists x. x\,\mathbf{rn}\,B$ is provable in **HA**. Thus, as by induction hpothesis $\mathbf{HA} + \mathrm{ECT}_0 \vdash A \leftrightarrow \exists x.\, x\,\mathbf{rn}\,A$ and $\mathbf{HA} + \mathrm{ECT}_0 \vdash B \leftrightarrow \exists x.\, x\,\mathbf{rn}\,B$ it follows that $\mathbf{HA} + \mathrm{ECT}_0 \vdash A{\wedge}B \leftrightarrow \exists x. x\,\mathbf{rn}\,A{\wedge}B$.
($\to$) By induction hypothesis A and B satisfy (1). Therefore, $A{\to}B$ is equivalent to $\forall x.\, x\,\mathbf{rn}\,A \to \exists y.\, y\,\mathbf{rn}\,B$ which by ECT_0 (as $x\,\mathbf{rn}\,A$ is almost negative) is equivalent to $\exists z. \forall x.\, x\,\mathbf{rn}\,A \to \{z\}(x){\downarrow} \wedge \{z\}(x)\,\mathbf{rn}\,B$, i.e. $\exists z.\, z\,\mathbf{rn}\,A{\to}B$.
($\forall$) By induction hypothesis $A(y)$ satisfies (1). Thus $\forall y. A(y)$ is equivalent to $\forall y. \exists x.\, x\,\mathbf{rn}\,A(y)$ which by ECT_0 is equivalent to $\exists z. \forall y.\, \{z\}(y){\downarrow} \wedge \{z\}(y)\,\mathbf{rn}\,A(y)$, i.e. $\exists z.\, z\,\mathbf{rn}\,\forall y. A(y)$.
($\exists$) Assume as induction hypothesis that $\mathbf{HA} + \mathrm{ECT}_0 \vdash A(x) \leftrightarrow \exists z.\, z\,\mathbf{rn}\,A(x)$. By definition $x\,\mathbf{rn}\,\exists x.\, A(x) \equiv \mathrm{snd}(x)\,\mathbf{rn}\,A(\mathrm{fst}(x))$. Thus, we have $\mathbf{HA}{+}\mathrm{ECT}_0 \vdash x\,\mathbf{rn}\,\exists x.\, A(x) \to A(\mathrm{fst}(x))$ as it follows from the induction hypothesis (by substituting $\mathrm{fst}(x)$ for x) that $\mathbf{HA} + \mathrm{ECT}_0 \vdash \mathrm{snd}(x)\,\mathbf{rn}\,A(\mathrm{fst}(x)) \to A(\mathrm{fst}(x))$. But from $\mathbf{HA} + \mathrm{ECT}_0 \vdash x\,\mathbf{rn}\,\exists x.\, A(x) \to A(\mathrm{fst}(x))$ it follows immediately that $\mathbf{HA}{+}\mathrm{ECT}_0 \vdash x\,\mathbf{rn}\,\exists x.\, A(x) \to \exists x.\, A(x)$ and, therefore, also that $\mathbf{HA}{+}\mathrm{ECT}_0 \vdash \exists x.\, x\,\mathbf{rn}\,\exists x.\, A(x) \to \exists x.\, A(x)$.
On the other hand by induction hypothesis we have $\mathbf{HA} + \mathrm{ECT}_0 \vdash A(x) \to \exists z.\, z\,\mathbf{rn}\,A(x)$. As $\mathbf{HA} \vdash z\,\mathbf{rn}\,A(x) \to \langle x,z\rangle\,\mathbf{rn}\,\exists x. A(x)$ and, therefore, also $\mathbf{HA} \vdash z\,\mathbf{rn}\,A(x) \to \exists x.\, x\,\mathbf{rn}\,\exists x. A(x)$ it follows that $\mathbf{HA} \vdash \exists z.\, z\,\mathbf{rn}\,A(x) \to$

[12] i.e. there are no restrictions on the syntactic form of A

[13] For this choice of A and B the premiss of ECT_0^* is obviously provable in **HA**. Thus, by ECT_0^* it follows that $\exists e. \forall x.\, A(x) \to \{e\}(x){\downarrow} \wedge B(x, \{e\}(x))$. As $\neg\neg A(x)$ is provable in **HA** it follows from ECT_0^* that $\exists e. \forall x. \neg\neg(\{e\}(x){\downarrow} \wedge B(x, \{e\}(x))$, i.e. more explicitly that

$$(1) \quad \forall x.\, \neg\neg\big(\{e\}(x){\downarrow} \wedge \big((\{e\}(x){=}0 \wedge \{x\}(x){\downarrow}) \vee (\{e\}(x){=}1 \wedge \neg\{x\}(x){\downarrow})\big)\big)$$

for some e. Let e_0 be a Gödel number of an algorithm such that $\{e_0\}(x){\downarrow}$ iff $\{e\}(x){=}1$. Now instantiating x in (1) by e_0 we get

$$(2) \quad \neg\neg\big(\{e\}(e_0){\downarrow} \wedge \big((\{e\}(e_0){=}0 \wedge \{e_0\}(e_0){\downarrow}) \vee (\{e\}(e_0){=}1 \wedge \neg\{e_0\}(e_0){\downarrow})\big)\big)$$

which, however, is contradictory as due to the nature of e_0 if $\{e\}(e_0){=}0$ then $\neg\{e_0\}(e_0){\downarrow}$ and if $\{e\}(e_0){=}1$ then $\{e_0\}(e_0){\downarrow}$.

$\exists x.\, x \,\mathbf{rn}\, \exists x.A(x)$. Thus, $\mathbf{HA} + \mathrm{ECT}_0 \vdash A(x) \to \exists x.\, x \,\mathbf{rn}\, \exists x.A(x)$ from which it readily follows that $\mathbf{HA} + \mathrm{ECT}_0 \vdash \exists x.A(x) \to \exists x.\, x \,\mathbf{rn}\, \exists x.A(x)$.
($\vee$) This case is redundant as disjunction can be expressed in terms of the other connectives and quantifiers.

(2) Suppose that $\mathbf{HA} \vdash \exists e.\, e \,\mathbf{rn}\, A$. Then also $\mathbf{HA} + \mathrm{ECT}_0 \vdash \exists e.\, e \,\mathbf{rn}\, A$ from which it follows by the already established claim (1) that $\mathbf{HA} + \mathrm{ECT}_0 \vdash A$.
Suppose that $\mathbf{HA}+\mathrm{ECT}_0 \vdash A$. Then $\mathbf{HA} \vdash B_1 \wedge \ldots \wedge B_n \to A$ for some instances B_i of ECT_0. By Theorem 2.1 we have $\mathbf{HA} \vdash \exists e.\, e \,\mathbf{rn}\, (B_1 \wedge \ldots \wedge B_n \to A)$ from which it follows that $\mathbf{HA} \vdash \exists e.\, e \,\mathbf{rn}\, A$ as for the B_i we have $\mathbf{HA} \vdash \exists e.\, e \,\mathbf{rn}\, B_i$ by Lemma 2.2. $\square$

Notice, however, that in general $\mathbf{HA}$ does *not* prove $\exists x.\, x \,\mathbf{rn}\, A \to A$ as can be seen when substituting for A an instance of CT_0 that is not derivable in $\mathbf{HA}$. This defect can be remedied by changing the notion of number realizability to *number realizability combined with truth*, i.e. one associates with every formula A a predicate $x \,\mathbf{rnt}\, A$ (with x fresh) where all clauses are as in Def. 2.1 with the single exception that the clause for implication is modified as follows

$$n \,\mathbf{rnt}\, A \to B \quad \equiv \quad \big(\forall m.\, m \,\mathbf{rnt}\, A \to \{n\}(m){\downarrow} \wedge \{n\}(m) \,\mathbf{rnt}\, B\big) \wedge (A \to B)$$

For this notion of realizability with truth one easily proves that

Theorem 2.3. *For all formulas A in the language of* $\mathbf{HA}$ *it holds that*

(1) $\mathbf{HA} \vdash (\exists x.\, x \,\mathbf{rnt}\, A) \to A$

(2) *If* $\mathbf{HA} \vdash A$ *then there is a number e with* $\mathbf{HA} \vdash \{e\}(\langle \vec{x} \rangle) \,\mathbf{rnt}\, A$ *where $\vec{x}$ contains all free variables of A.*

Thus, for a closed formula A we have $\mathbf{HA} \vdash A$ *iff* $\mathbf{HA} \vdash \exists x.\, x \,\mathbf{rnt}\, A$.[14]

Proof. Exercise! $\square$

Notice that in $\mathbf{HA}$ one cannot always prove the equivalence of A and $\exists x.\, x \,\mathbf{rnt}\, A$ since this equivalence may fail in the standard model $\mathbb{N}$ of $\mathbf{HA}$. But for negated formulas this equivalence holds.

Theorem 2.4. *For all formulas of* $\mathbf{HA}$ *we have* $\mathbf{HA} \vdash \neg A \leftrightarrow \exists n.\, n \,\mathbf{rnt}\, \neg A$.

Proof. Since $\mathbf{HA}$ proves $n \,\mathbf{rnt}\, A \to A$ it also proves $\neg A \to \neg(n \,\mathbf{rnt}\, A)$ and thus also $\neg A \to \forall n.\, n \,\mathbf{rnt}\, \neg A$ and thus in particular also $\neg A \to \exists n.\, n \,\mathbf{rnt}\, \neg A$. $\square$

Using Th. 2.3 one easily proves the following important metamathematical property of $\mathbf{HA}$.

Theorem 2.5. (Disjunction and Existence Property)

[14] It is an open problem (spotted by P. Lietz) to find an extension $\mathbf{HA}^*$ of $\mathbf{HA}$ such that for closed A, $\mathbf{HA}^* \vdash A$ iff $\mathbf{PA} \vdash \exists x.\, x \,\mathbf{rnt}\, A$.

(1) *If* $\mathbf{HA} \vdash A \vee B$ *with* A *and* B *closed then* $\mathbf{HA} \vdash A$ *or* $\mathbf{HA} \vdash B$

(2) *If* $\mathbf{HA} \vdash \exists x.A(x)$ *and* $\exists x.A(x)$ *is closed then there exists a number* n *such that* $\mathbf{HA} \vdash A(n)$.

One might dislike that the formulation of ECT_0 is somewhat complicated as it requires the syntactic notion "almost negative". Actually, one can avoid this if one postulates[15] the so-called *Markov's Principle*

MP $\qquad \neg\neg\exists x.A(x) \to \exists x.A(x) \qquad$ (A primitive recursive).

Using MP one easily shows that every almost negative formula is provably equivalent to a *negative* formula, i.e. one without any occurrences of $\vee$ or $\exists$.[16] Thus, in particular, for every formula A the formula $x\,\mathbf{rn}\,A$ is provably equivalent to a negative formula $R_A(x)$. Accordingly, in $\mathbf{HA} + \mathrm{MP} + \mathrm{ECT}_0$ one can prove the equivalences $\neg A \Leftrightarrow \neg\exists x.x\,\mathbf{rn}\,A \Leftrightarrow \forall x.\neg R_A(x)$. As the latter formula is negative in $\mathbf{HA}+\mathrm{MP}+\mathrm{ECT}_0$ every negated formula is provably equivalent to a negative one. Thus $\mathbf{HA} + \mathrm{MP} + \mathrm{ECT}_0$ proves

$\mathrm{ECT}_0' \qquad (\forall x.(\neg A(x) \to \exists y.B(x,y))) \to \exists e.\forall x.(\neg A(x) \to B(x,\{e\}(x)))$

for arbitrary formulas A and B. Notice that ECT_0' entails ECT_0 as under MP every almost negative formula is equivalent to a negated formula and thus to its double negation.
Now from Theorem 2.2 it follows immediately that

Theorem 2.6. *For all formulas* A *of* $\mathbf{HA}$ *it holds that*

(1) $\mathbf{HA} + \mathrm{MP} + \mathrm{ECT}_0' \vdash A \leftrightarrow \exists x.\,x\,\mathbf{rn}\,A$

(2) $\mathbf{HA} + \mathrm{MP} + \mathrm{ECT}_0' \vdash A$ *iff* $\mathbf{HA} + \mathrm{MP} \vdash \exists x.\,x\,\mathbf{rn}\,A$.

Using the fact that $\mathbf{PA}$ is conservative w.r.t. almost negative formulas over $\mathbf{HA}$ one can show that $\mathbf{PA} \vdash \exists x.\,x\,\mathbf{rn}\,A$ iff $\mathbf{HA} + \mathrm{MP} + \mathrm{ECT}_0' \vdash \neg\neg A$.
Theorems 2.2 and 2.6 have become known under the name "Trolestra's Axiomatization of Realizability" and date back to the early 1970ies, see [Tr73] which is encyclopedic also for axiomatizations of other notions of realizability (and related interpretations like e.g. Gödel's functional interpretation).

[15] Actually, one can show (exercise!) that MP is equivalent to

$$\neg\neg\exists z.T(x,y,z) \to \exists z.T(x,y,z)$$

saying that "a computation terminates if it is impossible that it diverges".

[16] The reason is that for primitive recursive predicates $P(x)$ Markov's Principle says that $\exists x.P(x) \leftrightarrow \neg\neg\exists x.P(x)$ and the right hand side of the latter equivalence is logically equivalent to $\neg\forall x.\neg P(x)$, i.e. a negative formula.

3 Partial Combinatory Algebras

In this chapter we introduce the basic notion of structure over which one can build realizability models, namely so-called *partial combinatory algebras* (pca's) which provide a notion of *untyped model of computation.* This notion has a lot of instances and we will present the most important examples that will be used later on again and again.

Definition 3.1. *A* weak partial combinatory algebra (wpca) *is a pair* $\mathcal{A} = (|\mathcal{A}|, \cdot)$ *where* $|\mathcal{A}|$ *is a non-empty set and* $\cdot : |\mathcal{A}| \times |\mathcal{A}| \rightharpoonup |\mathcal{A}|$ *is a partial binary operation on* $|\mathcal{A}|$ *such that there exist elements* $\mathsf{k}, \mathsf{s} \in |\mathcal{A}|$ *satisfying the conditions*

(1) $\mathsf{k} \cdot a \cdot b = a$

(2) $\mathsf{s} \cdot a \cdot b \downarrow$

(3) $\mathsf{s} \cdot a \cdot b \cdot c = a \cdot c \cdot (b \cdot c)$ *whenever* $a \cdot c \cdot (b \cdot c) \downarrow$

for all $a, b, c \in |\mathcal{A}|$.
A partial combinatory algebra (pca) $\mathcal{A}$ *is a weak pca* $\mathcal{A}$ *where* s *can be chosen in such a way that* $\mathsf{s} \cdot a \cdot b \cdot c \downarrow$ *implies* $a \cdot c \cdot (b \cdot c) \downarrow$ *for all* $a, b, c \in |\mathcal{A}|$. ◊

Notation Often, for sake of readability, we write simply ab instead of $a \cdot b$.

At first sight the notion of partial combinatory algebra may look a bit weird due to its existential quantification over k and s satisfying a couple of fancy properties. The next lemma gives an alternative characterization of pca's. For this purpose we have to introduce the notion of *polynomial over* $\mathcal{A} = (|\mathcal{A}|, \cdot)$, i.e. terms built from countably many variables and constants[17] for elements of $|\mathcal{A}|$ via the binary operation $\cdot : |\mathcal{A}| \times |\mathcal{A}| \rightharpoonup |\mathcal{A}|$. We write $T(\mathcal{A})$ for the set of polynomials over $\mathcal{A}$. Moreover, we write $t_1 \simeq t_2$ as an abbreviation for the statement that either t_1 and t_2 are both undefined or both sides are defined and equal (so-called *strong equality*).[18]

Lemma 3.1. *Let* $\mathcal{A}$ *be an applicative structure, i.e.* $\mathcal{A} = (|\mathcal{A}|, \cdot)$ *where* $|\mathcal{A}|$ *is a non-empty set and* $\cdot : |\mathcal{A}| \times |\mathcal{A}| \rightharpoonup |\mathcal{A}|$. *Then* $\mathcal{A}$ *is a weak partial combinatory algebra iff for every polynomial* $t \in T(\mathcal{A})$ *and variable* x *there exists a polynomial* $\Lambda x.t \in T(\mathcal{A})$ *with* $\mathrm{FV}(\Lambda x.t) \subseteq \mathrm{FV}(t) \setminus \{x\}$ *such that* $\Lambda x.t \downarrow$ *and* $(\Lambda x.t) \cdot a = t[a/x]$ *whenever* $t[a/x] \downarrow$ *.*
Moreover, $\mathcal{A}$ *is a pca iff for every polynomial* $t \in T(\mathcal{A})$ *and variable* x *there exists a polynomial* $\Lambda x.t \in T(\mathcal{A})$ *with* $\mathrm{FV}(\Lambda x.t) \subseteq \mathrm{FV}(t) \setminus \{x\}$ *such that* $\Lambda x.t \downarrow$ *and* $(\Lambda x.t) \cdot a \simeq t[a/x]$ *for all* $a \in |\mathcal{A}|$.

Proof. $\Leftarrow$: The elements k and s are given by $\Lambda x.\Lambda y.x$ and $\Lambda x.\Lambda y.\Lambda z.xz(yz)$, respectively. It is straightforward to check that the so defined k and s satisfy conditions (1)-(3) of Def. 3.1.
$\Rightarrow$: We define $\Lambda x.t$ by structural recursion on $t \in T(\mathcal{A})$ as follows: $\Lambda x.x \equiv \mathsf{skk}$, $\Lambda x.y \equiv \mathsf{k}y$ if y is different from x and $\Lambda x.t_1 t_2 \equiv \mathsf{s}(\Lambda x.t_1)(\Lambda x.t_2)$. □

[17] We use a itself as the constant denoting $a \in |\mathcal{A}|$.
[18] more constructively, we may formulate $t_1 \simeq t_2$ as $(t_1\downarrow \vee\ t_2\downarrow) \Rightarrow t_1 = t_2$

Thus, an applicative structure $\mathcal{A}$ is a pca iff there is some kind of functional abstraction available for polynomials over $\mathcal{A}$. In a weak pca we permit that $(\Lambda x.t)a$ may be defined even if $t[a/x]$ is not defined. This weaker form of functional abstraction is sometimes easier to establish and, more importantly, sufficient for building realizability models.

Partial combinatory algebras whose application operation $\cdot$ is total were originally introduced as models for *combinatory logic* and λ-calculus[19] (see [HS]). However, most models of computation are *inherently partial* (as e.g. classical recursion theory, see [Ro]) and the notion of pca is defined in a way that it subsumes these partial models as well.

Example 3.1. (the first Kleene algebra $\mathcal{K}_1$)
The underlying set of $\mathcal{K}_1$ is the set $\mathbb{N}$ of natural numbers and application is given by Kleene application, i.e. $n \cdot m \simeq \{n\}(m)$. Appropriate elements k and s are given by $\Lambda x.\Lambda y.x$ and $\Lambda x.\Lambda y.\Lambda z.xy(yz)$, respectively.
Notice that this choice of s exhibits $\mathcal{K}_1$ as a pca and not only a weak pca.

Example 3.2. (Scott's $\mathcal{P}\omega$)
The underlying set of $\mathcal{P}\omega$ is the powerset of $\omega = \mathbb{N}$. In order to define a (total) application on $\mathcal{P}\omega$ we have to introduce (besides a prim. rec. pairing function with prim. rec. projections) the following bijection between finite subsets of $\mathbb{N}$ and $\mathbb{N}$ itself: $e_n = A$ iff $n = \sum_{k \in A} 2^k$. Obviously, the predicates $m \in e_n$ and $m = |e_n|$ are primitive recursive. In $\mathcal{P}\omega$ application is defined as follows

$$a \cdot b = \{n \in \mathbb{N} \mid \exists m \in \mathbb{N}.\ e_m \subseteq b \wedge \langle m, n \rangle \in a\}$$

for $a, b \in \mathcal{P}\omega$. Notice that a map $f : \mathcal{P}\omega \to \mathcal{P}\omega$ is of the form $f(x) = a \cdot x$ for some $a \in \mathcal{P}\omega$ iff f is continuous w.r.t. the Scott topology on the cpo $\mathcal{P}\omega$.[20] Moreover, the map $\mathsf{ev} : \mathcal{P}\omega \to \mathcal{P}\omega^{\mathcal{P}\omega} : a \mapsto \big[b \mapsto a \cdot b\big]$ has a right inverse $\mathsf{fun} : \mathcal{P}\omega^{\mathcal{P}\omega} \to \mathcal{P}\omega : f \mapsto \{\langle n, m \rangle \mid m \in f(e_n)\}$, i.e. $\mathsf{ev} \circ \mathsf{fun} = id$.[21] Using ev and fun we can implement the combinators k and s by $\mathsf{fun}(\lambda x.\mathsf{fun}(\lambda y.x))$ and $\mathsf{fun}(\lambda x.\mathsf{fun}(\lambda y.\mathsf{fun}(\lambda z.\mathsf{ev}(\mathsf{ev}(x)(z))(\mathsf{ev}(y)(z)))))$, respectively. Using the facts that domains form a model of typed λ-calculus (see [St4]) and $\mathsf{ev} \circ \mathsf{fun} = id$ it is straightforward to verify that the so defined k and s actually satisfy the requirements (1)-(3) of Def. 3.1. Since the application operation is total it follows trivially that $(\mathcal{P}\omega, \cdot)$ is a pca and not only a weak pca.[22]

Obviously, with the same argument every domain U containing U^U as a retract gives rise to a total pca as it provides a model for the λ_β-calculus (see [Sc80]). Prominent examples of such U are Scott's D_∞ and $[\mathbb{N} \rightharpoonup \mathbb{N}]$, the domain of partial maps of natural numbers, see e.g. [St4] for more information.

[19] In order to model untyped λ-calculus pca's have to satify some additional properties as dicussed in [HS].
[20] For background information about elementary domain theory see e.g. [St4].
[21] It also holds that $\mathsf{fun}(\mathsf{ev}(a)) \supseteq a$ for all $a \in \mathcal{P}\omega$.
[22] For a direct account avoiding elementary domain theory see vol.2 of [TvD].

Example 3.3. ($\mathcal{P}\omega_{\mathit{eff}}$)
One easily observes that k and s as chosen in Example 3.2 are recursively enumerable (r.e.) sets and that r.e. sets are closed under the application defined in Example 3.2. We write $\mathcal{P}\omega_{\mathit{eff}}$ for the ensuing (sub-)pca (of $\mathcal{P}\omega$).

Example 3.4. (the second Kleene algebra $\mathcal{K}_2$)
The underlying set of $\mathcal{K}_2$ is the set $\mathbb{N}^\mathbb{N}$ of all total functions from $\mathbb{N}$ to $\mathbb{N}$. The set $\mathbb{N}^\mathbb{N}$ can be endowed with the topology whose basic opens are of the form $U_s = \{\alpha \in \mathbb{N}^\mathbb{N} \mid s \preceq \alpha\}$ for $s \in \mathbb{N}^*$. The ensuing space is known as *Baire space*, the countable product of $\mathbb{N}$ considered as a discrete space, and denoted as $\mathcal{B}$.
It is an old observation due to L. E. J. Brouwer (see vol.1 of [TvD]) that every (total) continuous map $\phi : \mathcal{B} \to \mathbb{N}$ is induced (or better *"realized"*) by an appropriately chosen $\alpha \in \mathcal{B}$ in the sense that

$$\phi(\beta) = n \quad \text{iff} \quad \exists k{\in}\mathbb{N}.\alpha(\overline{\beta}(k)) = n{+}1 \ \wedge\ \forall \ell{<}k.\alpha(\overline{\beta}(\ell)) = 0$$

for all $\beta \in \mathcal{B}$ and $n \in \mathbb{N}$. We write $\alpha \Vdash \phi$ as a shorthand for "α induces ϕ" or "α realizes ϕ". Obviously, an α realizes a total continuous ϕ iff for all $\beta \in \mathcal{B}$ there exists a $k \in \mathbb{N}$ with $\alpha(\overline{\beta}(k)) > 0$. Such α are called *neighbourhood functions* iff, moreover, from $\alpha(s) > 0$ and $s \preceq s'$ it follows that $\alpha(s) = \alpha(s')$.[23] Obviously, for every continuous ϕ one can find a neighbourhood function α with $\alpha \Vdash \phi$ and every neighbourhood function induces a continuous ϕ. Notice, however, that different neighbourhood functions may induce the same continuous functional.
We say that $\alpha \in \mathcal{B}$ *induces* or *realizes* a continuous operator $\Phi : \mathcal{B} \to \mathcal{B}$ (notation: $\alpha \Vdash \Phi$) iff $\lambda s.\alpha(\langle n\rangle{*}s) \Vdash \lambda\beta.\Phi(\beta)(n)$ for all $n \in \mathbb{N}$. Obviously, an α induces a continuous operator Φ iff for all $n \in \mathbb{N}$ the function $\lambda s.\alpha(n{*}s) \in \mathcal{B}$ realizes a continuous operation from $\mathcal{B}$ to $\mathbb{N}$.
Application in $\mathcal{K}_2$ is defined as

$$\alpha \cdot \beta \simeq \gamma \quad \text{iff} \quad \forall n.\exists k.\, \alpha(\langle n\rangle{*}\overline{\beta}(k)) = \gamma(n){+}1 \ \wedge\ \forall \ell{<}k.\, \alpha(\langle n\rangle{*}\overline{\beta}(\ell)) = 0$$

for $\alpha, \beta, \gamma \in \mathcal{B} = |\mathcal{K}_2|$. Notice that α realizes a continuous $\Phi : \mathcal{B} \to \mathcal{B}$ iff $\alpha \cdot \beta\downarrow$ for all $\beta \in \mathcal{B}$. But, of course, if $\alpha \cdot \beta\downarrow$ for some β it will not be the case in general that α realizes a continuous operator $\Phi : \mathcal{B} \to \mathcal{B}$.
Now we sketch an argument why $\mathcal{K}_2$ is a pca. First observe that there is a homeomorphism $(\cdot,\cdot) : \mathcal{B}{\times}\mathcal{B} \xrightarrow{\cong} \mathcal{B}$. It can be shown that for $\mathcal{K}_2$ there holds an analogue of Th.A.1(2).

Lemma 3.2. *There is an $v \in \mathcal{B}$ and a total continuous function $\sigma : \mathcal{B}{\times}\mathcal{B} \to \mathcal{B}$ such that*

[23] The set of neighbourhood functions can be defined inductively as the least subset K of $\mathcal{B}$ such that

(1) $\lambda s.n{+}1 \in K$ for all $n \in \mathbb{N}$ and

(2) $\alpha \in K$ whenever $\alpha(\langle\rangle) = 0$ and $\lambda s.\alpha(\langle n\rangle{*}s) \in K$ for all $n \in \mathbb{N}$.

This is a useful observation as it allows us to prove a statement of the form $\forall\phi.A(\phi)$ (where ϕ ranges over continuous functionals from $\mathcal{B}$ to $\mathbb{N}$) by induction over K: replace $\forall\phi.A(\phi)$ by an equivalent statement $\forall\alpha{\in}K.A^*(\alpha)$ where $A^*(\alpha)$ is equivalent to $A(\phi)$ whenever $\alpha \Vdash \phi$. Notice, moreover, that K corresponds to the countably branching well-founded trees whose leaves are labelled by natural numbers. For details see vol.1 of [TvD].

(1) $\upsilon \cdot (\alpha, \beta) \simeq \alpha \cdot \beta$

(2) $\sigma(\alpha, \beta) \cdot \gamma \simeq \alpha \cdot (\beta, \gamma)$

for all $\alpha, \beta, \gamma \in \mathcal{B}$.

Proof. A lengthy and tedious programming exercise which does not provide much insight.
For details see pp.74-75 of [Tr73] or [vOo] 1.4.3.
The idea is that one may define a primitive recursive predicate T^* and a primitive recursive function U^* such that $(\alpha \cdot \beta)(x) = y$ iff $\exists z. T^*(\overline{(\alpha, \beta)}(z), x, z) \wedge U^*(z) = y$.
From this one can read off an υ satisfying (1).
A σ satisfying (2) can be constructed as follows

$$\sigma(\alpha, \beta)(\langle\rangle) = 0$$

$$\begin{aligned} &\sigma(\alpha, \beta)(\langle x \rangle * n) = y + 1 \\ &\qquad \text{if } \exists z \leq \mathrm{lgth}(n).\, T^*(\overline{(\alpha, (\beta, f_n))}(z), x, z) \wedge U(z) = y \end{aligned}$$

$$\sigma(\alpha, \beta)(\langle x \rangle * n) = 0 \text{ otherwise.}$$

where $f_n(i) = n_i$. $\square$

From Lemma 3.2(1) it follows that every polynomial over $\mathcal{K}_2$ in n variables induces a continuous map from $\mathcal{B}^n \cong \mathcal{B}$ to $\mathcal{B}$. From Lemma 3.2(2) (and $\mathcal{B}^n \cong \mathcal{B}$) it follows that for every polynomial $t[x_1, \ldots, x_n, x]$ there exists a polynomial $\Lambda x.t[x_1, \ldots, x_n, x]$ such that $t[\alpha_1, \ldots, \alpha_n, \alpha] \simeq \Lambda x.t[\alpha_1, \ldots, \alpha_n, x] \cdot \alpha$.
Thus, by Lemma 3.1 it follows that $\mathcal{K}_2$ is a pca.

The pca $\mathcal{K}_2$ is an abstraction of Kleene's *function realizability* (see [KV]) introduced for the purpose of extracting computational contents from proofs in intuitionistic analysis. Like his number realizability he introduced his function realizability as a syntactic translation. Function realizability does not validate Church's Thesis but instead the following two principles, namely *Generalized Continuity*

$$\text{GC} \quad (\forall \alpha.(A(\alpha) \to \exists \beta.B(\alpha, \beta))) \to \exists \gamma.\forall \alpha.(A(\alpha) \to B(\alpha, \gamma \cdot \alpha))$$

for almost negative A and *Bar Induction*

$$\begin{aligned} \text{BI} \quad & \big(\forall \alpha.\exists n.P(\overline{\alpha}(n))\big) \to \big(\forall n.(P(n) \to \forall m.P(n*m))\big) \to \big(\forall n.P(n) \to Q(n)\big) \\ & \to \big(\forall n.\, (\forall m.Q(n*\langle m \rangle)) \to Q(n)\big) \to Q(\langle\rangle) \end{aligned}$$

an induction principle for well-founded trees. A remarkable consequence of GC is that all functions on the real numbers are continuous, called *Brouwer's Continuity Theorem* (as he considered GC as a "logical" (in the sense of "evident") principle).

Example 3.5. ($\mathcal{K}_{2,\mathit{eff}}$)
The underlying set of $\mathcal{K}_{2,\mathit{eff}}$ are the total recursive functions from $\mathbb{N}$ to $\mathbb{N}$ which are closed under the application operation defined on $\mathcal{B}$ in Example 3.4. Moreover, as the υ and σ of Lemma 3.2 can be chosen as computable it follows that $\mathcal{K}_{2,\mathit{eff}}$ is a (sub-)pca of $\mathcal{K}_2$.
Notice the analogy with Example 3.3 where $\mathcal{P}\omega$ contains a sub-pca $\mathcal{P}\omega_{\mathit{eff}}$ consisting of the computable elements of $\mathcal{P}\omega$.

Example 3.6. (syntactic pca's)
Last but not least there are pca's of fairly syntactic nature.[24]
The simplest (total) pca's in this vein are term models of *Combinatory Logic* (see e.g. [HS]). The terms of combinatory logic are built from constants K and S via a binary operation (denoted by juxtaposition). We write $\mathcal{C}$ for the ensuing inductively defined set of terms. A *congruence* on $\mathcal{C}$ is an equivalence relation $\sim$ on $\mathcal{C}$ such that

$$t_1 \sim t_2 \text{ implies } t_1 s \sim t_2 s \text{ and } st_1 \sim st_2$$

for all $t_1, t_2, s \in \mathcal{C}$. A congruence $\sim$ on $\mathcal{C}$ is called a CL-theory iff $Kt_1t_2 \sim t_1$ and $St_1t_2t_3 \sim t_1t_3(t_2t_3)$. One readily checks that for every CL-theory T the quotient $\mathcal{C}/T$ gets a total pca when endowed with the application operation $[t]_T \cdot [s]_T = [ts]_T$ choosing $\mathsf{k} = [K]_T$ and $\mathsf{s} = [S]_T$.
Instead of combinatory logic one may consider untyped λ-calculus (see e.g. [HS]). Let Λ be the set of λ-terms modulo α-conversion, i.e. capture-free renaming of bound variables. A λ-theory is an equivalence relation $\sim$ on Λ such that $t_1 \sim t_2$ implies $t_1 s \sim t_2 s$, $st_1 \sim st_2$ and $\lambda x.t_1 \sim \lambda x.t_2$ and $(\lambda x.t)s \sim t[s/x]$. Obviously, for every λ-theory T the set Λ/T gets a total pca when endowed with the application operation $t \cdot s = [ts]_T$ choosing $\mathsf{k} = [\lambda x.\lambda y.x]_T$ and $\mathsf{s} = [\lambda x.\lambda y.\lambda z.xz(yz)]_T$.
Let Λ^0 be the set of closed λ-terms. Then for every every λ-theory T the set Λ^0/T gives rise to a sub-pca of Λ/T.
The following λ-theories will be of interest later on: the least λ-theory $\sim_\beta$ and so-called *sensible* λ-theories, i.e. theories identifying all *unsolvable*[25] terms. The most important instance of a sensible λ-theory is $\mathcal{K}^*$, the maximal consistent sensible λ-theory, equating all those terms t_1 and t_2 such that for all terms t, tt_1 is unsolvable iff tt_2 is unsolvable.
A slightly more "realistic" (in the sense of closer to practice) syntactic pca are LISP programs[26] modulo observational equivalence, i.e. $P_1 \sim_{obs} P_2$ iff for all programs P it holds that $PP_1 \downarrow$ iff $PP_2 \downarrow$.

We conclude this chapter by establishing a couple of facts about the coding capabilities of partial combinatory algebras.

[24]Though $\mathcal{K}_1$ and $\mathcal{K}_2$ are also fairly "intensional" as their elements can be thought of as codes of algorithms for partial functions on $\mathbb{N}$ and $\mathcal{B}$, respectively.
[25]a term is *unsovable* iff it does not reduce to a head normal form, i.e. leftmost-outermost reduction does not terminate
[26]one could take any *untyped* functional programming language

For the rest of this chapter let $\mathcal{A}$ be an arbitrary, but fixed pca. We write A as a shorthand for $|\mathcal{A}|$ and k and s for some choice of elements satisfying the conditions of Def. 3.1.
In subsequent proofs we will often (implicitly) use the equality

$$(\beta_*) \qquad (\Lambda x.t)a \simeq t[a/x] \quad \text{for all } a \in A$$

which due to Lemma 3.1 holds in any pca. Notice that in general $s\downarrow$ does not imply $(\Lambda x.t)s \simeq t[s/x]$ unless every free occurrence of x in t is not within the scope of a Λ-abstraction.[27]

Lemma 3.3. (Pairing and Booleans)

(1) *There exist* $\mathsf{p}, \mathsf{p}_0, \mathsf{p}_1 \in A$ *such that*

$$\mathsf{p}ab\downarrow \qquad \mathsf{p}_0(\mathsf{p}ab) = a \qquad \mathsf{p}_1(\mathsf{p}ab) = b$$

for all $a, b \in A$.

(2) *There exist* $\mathsf{true}, \mathsf{false}, \mathsf{cond} \in A$ *such that*

$$\mathsf{cond}\, a\, b\downarrow \qquad \mathsf{cond}\, a\, b\, \mathsf{true} = a \qquad \mathsf{cond}\, a\, b\, \mathsf{false} = b$$

for all $a, b \in A$.

Proof. ad (1) : Put $\mathsf{p} = \Lambda xyz.zxy$, $\mathsf{p}_0 = \Lambda z.z(\Lambda xy.x)$ and $\mathsf{p}_1 = \Lambda z.z(\Lambda xy.y)$. The claim then follows from (β_*).
ad (2) : Put $\mathsf{true} = \lambda xy.x$, $\mathsf{false} = \Lambda xy.y$ and $\mathsf{cond} = \Lambda xyz.zxy$. The claim follows again from (β_*). □

In the following we will often write $\langle a, b\rangle$ for $\mathsf{p}ab$. We also write i as abbreviation for skk and notice that $\mathsf{i}\,a = a$ for all $a \in A$.
Now we will show how natural numbers can be implemented within pca's.

Definition 3.2. (Numerals)
With every natural number n we associate an element $\underline{n} \in A$ by recursion on n in the following way

$$\underline{0} = \langle \mathsf{true}, \mathsf{i}\rangle \qquad \textit{and} \qquad \underline{n{+}1} = \langle \mathsf{false}, \underline{n}\rangle$$

We call $\underline{n}$ the numeral for *n.* ◊

Lemma 3.4. *There exist* $\mathsf{succ}, \mathsf{pred}, \mathsf{isz} \in A$ *such that*

$$\mathsf{succ}\,\underline{n} = \underline{n{+}1} \qquad \mathsf{pred}\,\underline{0} = \underline{0} \quad \mathsf{pred}\,\underline{n{+}1} = \underline{n} \qquad \mathsf{isz}\,\underline{0} = \mathsf{true} \quad \mathsf{isz}\,\underline{n{+}1} = \mathsf{false}$$

for all $n \in \mathbb{N}$.

[27] This can be seen from the following counterexample (due to Longley, see [Lon]): let $t \equiv \Lambda y.x$ and $s = \mathsf{ss}$ then $(\Lambda x.t)s = \mathsf{k}(\mathsf{ss})$ whereas $t[s/x] \equiv \Lambda y.\mathsf{ss} = \mathsf{s}(\mathsf{ks})(\mathsf{ks})$. Obviously, the problem is that in $\Lambda y.\mathsf{ss}$ the term ss is treated as a term and not as the value it denotes.

Proof. Put $\mathsf{succ} = \Lambda x.\langle \mathsf{false}, x\rangle$, $\mathsf{isz} = \mathsf{p}_0$ and $\mathsf{pred} = \Lambda x.\,\mathsf{cond}\,\underline{0}\,(\mathsf{p}_1 x)\,(\mathsf{isz}\,x)$. Using Lemma 3.3 one immediately verifies that the so defined elements satisfy the required properties. □

Theorem 3.1. (Fixpoint Operator)
There exists a $\mathsf{fix} \in A$ *such that*

$$\mathsf{fix}\,f\downarrow \qquad \textit{and} \qquad \mathsf{fix}\,f\,a \simeq f\,(\mathsf{fix}\,f)\,a$$

for all $f, a \in A$.

Proof. Let $\mathsf{fix} = \Lambda x.(\Lambda yz.x(yy)z)(\Lambda yz.x(yy)z)$. Let $f \in A$. We write χ_f for the value of $\Lambda yz.f(yy)z$. As $\mathsf{fix}\,f \simeq \chi_f\chi_f \simeq \Lambda z.f(\chi_f\chi_f)z$ and $\Lambda z.f(\chi_f\chi_f)z\downarrow$ we have $\mathsf{fix}\,f\downarrow$. Moreover, we have

$$\mathsf{fix}\,f\,a \simeq (\Lambda z.f(\chi_f\chi_f)z)a = f(\chi_f\chi_f)a \simeq f(\mathsf{fix}\,f)a$$

for all $a \in A$. □

Corollary 3.1. (Primitive Recursion Operator)
There is a $\mathsf{rec} \in A$ *such that*

$$\mathsf{rec}\,a\,f\,\underline{0} = a \qquad \textit{and} \qquad \mathsf{rec}\,a\,f\,\underline{n{+}1} \simeq f\,\underline{n}\,(\mathsf{rec}\,a\,f\,\underline{n})$$

for all $a, f \in A$ *and* $n \in \mathbb{N}$.

Proof. Define $\mathsf{rec} \equiv \mathsf{fix}(\Lambda r.\Lambda xfn.\,\mathsf{cond}\,x\,(f\,(\mathsf{pred}\,n)\,(r\,x\,f\,(\mathsf{pred}\,n))\,(\mathsf{isz}\,n))$. It is a good exercise in using (β_*) to show that the so defined rec satifies the required two properties. □

These results show that a partial combinatory algebra actually gives rise to a (kind of) *untyped functional programming language* supporting general recursion, the basic data types of booleans and natural numbers and a conditional (namely cond of Lemma 3.3).

4 Assemblies and Modest Sets

In this section we will introduce for every (weak) pca $\mathcal{A}$ a category $\mathbf{Asm}(\mathcal{A})$ of *assemblies over* $\mathcal{A}$ which is a model of impredicative (intuitionistic) type theory containing as full reflective subcategories both the category $\mathbf{Set}$ of classical sets and the category $\mathbf{Mod}(\mathcal{A})$ of *modest sets over* $\mathcal{A}$ which can be considered as the category of data types w.r.t. the notion of computability as given by the (weak) pca $\mathcal{A}$.

Definition 4.1. (assemblies and modest sets)
Let $\mathcal{A}$ be a (weak) pca. The category $\mathbf{Asm}(\mathcal{A})$ of assemblies over $\mathcal{A}$ *has as objects pairs $X = (|X|, ||\cdot||_X)$ where $|X|$ is a set and $||\cdot||_X$ is a mapping associating with every $x \in |X|$ a non-empty subset $||x||_X$ of $\mathcal{A}$. We also write $a \Vdash_X x$ instead of $a \in ||x||_X$. The morphisms from X to Y in $\mathbf{Asm}(\mathcal{A})$ are those maps $f : |X| \to |Y|$ for which there exists $e \in \mathcal{A}$ such that for every $x \in |X|$ and $a \in ||x||_X$ it holds that $e \cdot a \downarrow$ and $e \cdot a \in ||f(x)||_Y$ in which case we say "e realizes f" or "e tracks f" and which we denote as $e \Vdash f$. Composition in and identities of $\mathbf{Asm}(\mathcal{A})$ are inherited*[28] *from* $\mathbf{Set}$.
Let $\nabla : \mathbf{Set} \hookrightarrow \mathbf{Asm}(\mathcal{A})$ be the full and faithful functor sending a set S to $\nabla(S)$ with $|\nabla(S)| = S$ and $||s||_{\nabla(S)} = \mathcal{A}$ for all $s \in S$ and $\nabla(f) = f : \nabla(T) \to \nabla(S)$ for $f : T \to S$ in $\mathbf{Set}$.
An assembly X over $\mathcal{A}$ is a modest set *(over $\mathcal{A}$) iff $x = y$ whenever $||x||_X \cap ||y||_X$ is non-empty. We write $\mathbf{Mod}(\mathcal{A})$ for the full subcategory of $\mathbf{Asm}(\mathcal{A})$ on modest sets over $\mathcal{A}$ and $\mathsf{J} : \mathbf{Mod}(\mathcal{A}) \hookrightarrow \mathbf{Asm}(\mathcal{A})$ for the obvious inclusion functor.* $\Diamond$

Intuitively, morphism between assemblies X and Y are those maps between the underlying sets $|X|$ and $|Y|$ which can be "implemented" or "tracked" or "realized" by an algorithm operating on realizers instead of elements. The intuition behind "modest sets" is that realizers determine uniquely the objects they realize. Thus we have the following

Lemma 4.1. *Let $f, g : X \to A$ be morphisms in $\mathbf{Asm}(\mathcal{A})$ with $A \in \mathbf{Mod}(\mathcal{A})$. If $e \Vdash f$ and $e \Vdash g$ then $f = g$. Thus, the collection $\mathbf{Asm}(X, A)$ together with the assignment $f \mapsto \{e \in \mathcal{A} \mid e \Vdash f\}$ gives rise to a modest set usually denoted as A^X (c.f. Lemma 4.3).*

Proof. Suppose $e \Vdash f$ and $e \Vdash g$. Suppose $x \in |X|$. Then there exists $a \in ||x||_X$. Thus $e \cdot a \downarrow$ with $e \cdot a \in ||f(x)||_A$ and $e \cdot a \in ||g(x)||_A$ from which it follows that $f(x) = g(x)$ since A is modest by assumption. $\square$

Next we will establish the many good properties that are satisfied by $\mathbf{Asm}(\mathcal{A})$ and $\mathbf{Mod}(\mathcal{A})$. For explanation of basic categorical notions see [St2] or some of the sources referred to in *loc. cit.*

Lemma 4.2. *For every (weak) pca $\mathcal{A}$ the category $\mathbf{Asm}(\mathcal{A})$ has all finite limits. Moreover $\mathbf{Mod}(\mathcal{A})$ is closed under finite limits taken in $\mathbf{Asm}(\mathcal{A})$.*

[28] If $a \Vdash f : X \to Y$ and $b \Vdash g : Y \to Z$ then $g \circ f$ is realized by $\Lambda x.\, b \cdot (a \cdot x)$. Identity morphisms in $\mathbf{Asm}(\mathcal{A})$ are realized by $\mathsf{i} = \Lambda x.x$.

Proof. A terminal object is given by the assembly 1 with $|1| = \{*\}$ and $||*||_1 = |\mathcal{A}|$. Obviously 1 is modest. Let X and Y be assemblies over $\mathcal{A}$. Their cartesian product is given by the assembly $X{\times}Y$ whose underlying set is given by $|X|{\times}|Y|$ and $||\langle x, y\rangle||_{X\times Y} = \{e \in |\mathcal{A}| \mid \mathsf{p}_0 e \in ||x||_X \wedge \mathsf{p}_1 e \in ||y||_Y\}$. The first and second projections are given by the maps $\pi_0 : X{\times}Y \to X : \langle x, y\rangle \mapsto x$ and $\pi_1 : X{\times}Y \to Y : \langle x, y\rangle \mapsto y$ which are realized by p_0 and p_1, respectively. That $X{\times}Y$ is modest if X and Y are modest can be seen as follows. Suppose $e \Vdash_{X\times Y} \langle x, y\rangle$ and $e \Vdash_{X\times Y} \langle x', y'\rangle$. Then $\mathsf{p}_0 e \Vdash_X x$ and $\mathsf{p}_0 e \Vdash_X x'$ from which it follows that $x = x'$ as X is assumed as modest. Similarly, one sees that $y = y'$. Thus $\langle x, y\rangle = \langle x', y'\rangle$ as desired.
For $f, g : X \to Y$ in $\mathbf{Asm}(\mathcal{A})$ their equalizer is given by the assembly E whose underlying set is given by $|E| = \{x \in X \mid f(x) = g(x)\}$ and $||x||_E = ||x||_X$ and the inclusion map $e : E \to X$ realized by $\mathsf{i} = \Lambda x.x$. From the construction of E it is obvious that E is modest whenever X is modest.
The verification of the desired universal properties of the above constructions is left to the reader. □

Lemma 4.3. *For every (weak) pca $\mathcal{A}$ the category* $\mathbf{Asm}(\mathcal{A})$ *is cartesian closed. Moreover, for every $X \in \mathbf{Asm}(\mathcal{A})$ and $A \in \mathbf{Mod}(\mathcal{A})$ we have $A^X \in \mathbf{Mod}(\mathcal{A})$.*

Proof. Let X and Y be assemblies over $\mathcal{A}$. Their exponential $Y^X = [X{\to}Y]$ is given by the assembly with underlying set $\mathbf{Asm}(\mathcal{A})(X, Y)$ and $||f||_{[X\to Y]} = \{e \in \mathcal{A} \mid e \Vdash f\}$. The evaluation map $\mathsf{ev}_{X,Y} : [X{\to}Y]{\times}X \to Y : (f, x) \mapsto f(x)$ is realized by the algorithm $\Lambda x.\mathsf{p}_0 x(\mathsf{p}_1 x) \in \mathcal{A}$.
For showing that $\mathsf{ev}_{X,Y}$ satisfies the universal property required for an exponential suppose $e \Vdash f : Z{\times}X \to Y$. We have to show that there exists a unique $g \in \mathbf{Asm}(\mathcal{A})(Z, [X{\to}Y])$ with $\mathsf{ev}_{X,Y} \circ (g{\times}id_X) = f$. Thus $g(z)(x) = f(z, x)$ determining g uniquely. For existence of g as morphism of assemblies we just have to check that the map g is tracked by some element of $\mathcal{A}$. Well, one easily checks that $\Lambda x.\Lambda y.e(\mathsf{p}xy) \Vdash g$ as if $c \Vdash z$ and $a \Vdash x$ then $\mathsf{p}ca \Vdash \langle z, x\rangle$ and thus $e(\mathsf{p}ca) \Vdash f(z, x) = g(z)(x)$ as desired. □

Notice that if $\mathcal{A}$ is only a weak pca then $(\Lambda x.\mathsf{p}_0 x(\mathsf{p}_1 x))(\mathsf{p}ea)$ may terminate even if e does not realize an $f : X \to Y$ or a does not realize an $x \in |X|$. This, however, is not a problem because for $(\Lambda x.\mathsf{p}_0 x(\mathsf{p}_1 x)) \Vdash \mathsf{ev}_{X,Y}$ it suffices that $\mathsf{p}_0 c(\mathsf{p}_1 c) \Vdash f(x)$ whenever $\mathsf{p}_0 c \Vdash_{[X\to Y]} f$ and $\mathsf{p}_1 c \Vdash_X x$ and *nothing is required for the case that this precondition is not satisfied.* Similarly, if $e \Vdash f$ then $(\Lambda x.\Lambda y.e(\mathsf{p}xy))ca$ may terminate even if c or a do not realize an element of $|Z|$ or $|X|$, respectively. These considerations demonstrate why it suffices to assume that $\mathcal{A}$ is only a *weak* pca.

Next we show that $\mathbf{Mod}(\mathcal{A})$ and $\mathbf{Set}$ are full reflective subcategories of $\mathbf{Asm}(\mathcal{A})$.

Theorem 4.1. *For a (weak) pca $\mathcal{A}$ the full and faithful functors $\nabla : \mathbf{Set} \hookrightarrow \mathbf{Asm}(\mathcal{A})$ and $\mathsf{J} : \mathbf{Mod}(\mathcal{A}) \hookrightarrow \mathbf{Asm}(\mathcal{A})$ have left adjoints. Thus* $\mathbf{Set}$ *and* $\mathbf{Mod}(\mathcal{A})$ *appear as full reflective subcategories of* $\mathbf{Asm}(\mathcal{A})$.

Moreover, a left adjoint of ∇ *is given by the global sections functor* $\Gamma = \mathbf{Asm}(\mathcal{A})(1,-) : \mathbf{Asm}(\mathcal{A}) \to \mathbf{Set}$ *which is isomorphic to the forgetful functor* $|-| : \mathbf{Asm}(\mathcal{A}) \to \mathbf{Set}$ *which is obviously faithful. Thus* $\mathbf{Asm}(\mathcal{A})$ *and* $\mathbf{Mod}(\mathcal{A})$ *are well-pointed.*

Proof. As $\mathbf{Asm}(\mathcal{A})(1,X) \cong |X|$ and

$$\begin{tikzcd} \mathbf{Asm}(\mathcal{A})(1,X) \arrow[r, "\cong"] \arrow[d, "\mathbf{Asm}(\mathcal{A})(1,f)"'] & |X| \arrow[d, "|f|"] \\ \mathbf{Asm}(\mathcal{A})(1,Y) \arrow[r, "\cong"] & |Y| \end{tikzcd}$$

commutes for all $f : X \to Y$ in $\mathbf{Asm}(\mathcal{A})$ it follows that the global sections functor $\Gamma = \mathbf{Asm}(\mathcal{A})(1,-)$ is faithful and accordingly $\mathbf{Asm}(\mathcal{A})$ is well-pointed. As 1 is modest $\mathbf{Mod}(\mathcal{A})$ is well-pointed, too.

From now on we treat Γ and $|-|$ as identical. For $X \in \mathbf{Asm}(\mathcal{A})$ the map $\eta_X : |X| \to |\nabla(\Gamma(X))| : x \mapsto x$ is realized e.g. by $\Lambda x.x$. Suppose $f : X \to \nabla(S)$. Let $g : \Gamma(X) \to S : x \mapsto f(x)$ in $\mathbf{Set}$. Obviously, we have $\nabla(g) \circ \eta_X = f$. As the underlying map of η_X is onto and ∇ is (full and) faithful it follows that g is actually the unique map with $\nabla(g) \circ \eta_X = f$. Thus $\Gamma \dashv \nabla$ as desired.

Let $X \in \mathbf{Asm}(\mathcal{A})$. Define $\sim$ as the least equivalence relation on $|X|$ such that $x \sim x'$ whenever $a \in ||x||_X \cap ||x'||_X$ for some $a \in \mathcal{A}$. Let $\mathsf{M}(X)$ be the assembly with $|\mathsf{M}(X)| = |X|_{/\sim}$ and $||\,[x]_\sim||_{\mathsf{M}(X)} = \bigcup_{x' \in [x]_\sim} ||x'||_X$. The map $\eta_X : |X| \to |\mathsf{M}(X)| : x \mapsto [x]_\sim$ is realized by $\Lambda x.x$ and thus $\eta_X : X \to \mathsf{M}(X)$ is a morphism of assemblies. Suppose $A \in \mathbf{Mod}(\mathcal{A})$ and $f : X \to \mathsf{J}(A)$. Let $e \Vdash f$. If $a \in ||x||_X \cap ||x'||_X$ then $ea \in ||f(x)||_A \cap ||f(x')||_A$ and thus $f(x) = f(x')$ as A is modest by assumption. Thus $f(x) = f(x')$ whenever $x \sim x'$. Accordingly, the map $g : |\mathsf{M}(X)| \to |A| : [x]_\sim \mapsto f(x)$ is well defined and realized by any realizer for f. We have $f = g \circ \eta_X$ and g is unique with this property since the underlying map of η_X is onto. Thus J has a left adjoint M whose unit at X is given by η_X. For $f : X \to Y$ the map $\mathsf{M}(f)$ is defined uniquely by the requirement $\mathsf{M}(f) \circ \eta_X = \eta_Y \circ f$. □

Next we characterize monomorphisms in $\mathbf{Asm}(\mathcal{A})$ and $\mathbf{Mod}(\mathcal{A})$.

Lemma 4.4. *Let* $\mathcal{A}$ *be a (weak) pca. Then a map* $f : X \to Y$ *in* $\mathbf{Asm}(\mathcal{A})$ *is monic in* $\mathbf{Asm}(\mathcal{A})$ *iff its underying map is one-to-one and a map* $f : A \to B$ *in* $\mathbf{Mod}(\mathcal{A})$ *is monic in* $\mathbf{Mod}(\mathcal{A})$ *iff its underlying map is one-to-one.*

Proof. If the underlying map of f is one-to-one then f is obviously monic in $\mathbf{Asm}(\mathcal{A})$. Suppose $f : X \to Y$ is monic in $\mathbf{Asm}(\mathcal{A})$ and $f(x) = f(x')$. Let g and g' be the maps from 1 to X with $g(*) = x$ and $g'(*) = x'$, respectively. Then $f \circ g = f \circ g'$ and thus $g = g'$ (as f is monic by assumption) from which it follows that $x = x'$.

This argument goes through for $\mathbf{Mod}(\mathcal{A})$ as well since 1 is modest. □

Next we consider and characterize the particularly nice class of *regular monos*, i.e. those monos which appear as equalizers.

Lemma 4.5. *Let $\mathcal{A}$ be a (weak) pca. Then a mono $m : X \to Y$ in $\mathbf{Asm}(\mathcal{A})$ is regular iff there exists $e \in \mathcal{A}$ such that $ea \in ||x||_X$ whenever $a \in ||m(x)||_Y$.*

Proof. First notice that the characterizing condition is stable under isomorphism.
The equalizers constructed in the proof of Lemma 4.2 obviously satisfy the characterizing property (take $\Lambda x.x$ for e).
Suppose $m : X \rightarrowtail Y$ and $e \in \mathcal{A}$ as required by the characterizing condition. W.l.o.g. suppose $|X| \subseteq |Y|$ and $m(x) = x$ for all $x \in |X|$. Let $f, g : Y \to \nabla(2)$ with f constantly 0 and $g(y) = 0$ iff $y \in X$. We show that m is an equalizer of f and g. Suppose $h : Z \to Y$ with $fh = gh$. Then $\Gamma(h) : |Z| \to |Y|$ factors through $|X|$. Let $k : Z \to X$ be defined as $k(z) = h(z)$ for all $z \in |Z|$. Let $e' \Vdash h$. If $a \in ||z||_Z$ then $e'a \in ||h(z)||_Y$ and also $e(e'a) \in ||k(z)||_X$ as $m(k(z)) = h(z)$. Thus, we have $\Lambda x.e(e'x) \Vdash k$, i.e. $k : Z \to X$ with $mk = h$. Uniqueness of k follows from m being monic. $\square$

It is obvious from this characterization that in $\mathbf{Asm}(\mathcal{A})$ regular monos are closed under composition. Moreover, one can show easily (exercise!) that regular monos are stable under pullbacks along arbitrary morphisms.

Lemma 4.6. *If $m : X \to A$ is a regular mono in $\mathbf{Asm}(\mathcal{A})$ and A is modest then m is a regular mono in $\mathbf{Mod}(\mathcal{A})$.*

Proof. It is easily shown (exercise!) that X is modest as well.
W.l.o.g. assume that $|X| \subseteq |A|$ and $m(x) = x$. From Lemma 4.5 we know that there is an $e \in \mathcal{A}$ such that $ea \in ||x||_X$ whenever $a \in ||m(x)||_A$. Let B be the modest set with $|B| = \{0,1\}\times(|A|\backslash|X|) \cup \{0\}\times|X|$ and $||\cdot||_B$ defined as follows: $||\langle 0, x\rangle||_B = \{a \in \mathcal{A} \mid \mathsf{p}_0 a \in \{\mathsf{true}, \mathsf{false}\} \wedge \mathsf{p}_1 a \in ||x||_A\}$ for $x \in |X|$ and $||\langle 0, y\rangle||_B = \{a \in \mathcal{A} \mid \mathsf{p}_0 a = \mathsf{true} \wedge \mathsf{p}_1 a \in ||y||_A\}$ and $||\langle 1, y\rangle||_B = \{a \in \mathcal{A} \mid \mathsf{p}_0 a = \mathsf{false} \wedge \mathsf{p}_1 a \in ||y||_A\}$ for $y \in |A|\backslash|X|$. Let f and g be the morphisms from A to B realized by $\Lambda x.\mathsf{p}\,\mathsf{true}\,x$ and $\Lambda x.\mathsf{p}\,\mathsf{false}\,x$, respectively. We show that m is an equalizer of f and g. Obviously, for $y \in |A|$ we have $f(y) = g(y)$ iff $y \in |X|$. Thus $fm = gm$. Suppose $h : C \to A$ in $\mathbf{Mod}(\mathcal{A})$ with $fh = gh$. Let $k : |C| \to |X| : z \mapsto h(x)$. As in the proof of Lemma 4.5 one shows that $\Lambda x.e(e'x) \Vdash k$ where $e' \Vdash h$. Thus k is a morphism in $\mathbf{Mod}(\mathcal{A})$ with $mk = h$ and k is unique with this property as m is monic.
Thus, we have exhibited m as equalizer of f and g in $\mathbf{Mod}(\mathcal{A})$ as desired. $\square$

Again the regular monos in $\mathbf{Mod}(\mathcal{A})$ are stable under composition and arbitrary pullbacks.

Now we can characterize epi(morphism)s in $\mathbf{Asm}(\mathcal{A})$ and $\mathbf{Mod}(\mathcal{A})$.

Lemma 4.7. *Let $\mathcal{A}$ be a (weak) pca. A morphism f in $\mathbf{Asm}(\mathcal{A})$ or $\mathbf{Mod}(\mathcal{A})$ is epic iff its underlying map $|f|$ is onto.*

Proof. Obviously, if $|f|$ is onto then f is epic as both $\mathbf{Asm}(\mathcal{A})$ and $\mathbf{Mod}(\mathcal{A})$ are well-pointed.

For the reverse direction suppose that $f : X \to Y$ is epic in $\mathbf{Asm}(\mathcal{A})$ or $\mathbf{Mod}(\mathcal{A})$. Let Z be the assembly with $|Z| = \{f(x) \mid x \in |X|\}$ and $||z||_Z = ||z||_Y$ for $z \in |Z|$. Let m be the inclusion of $|Z|$ into $|Y|$ giving rise to the regular monomorphism $m : Z \rightarrowtail Y$ realized by i. Obviously Z is modest whenever Y is modest. Let $e : X \twoheadrightarrow Z$ with $e(x) = f(x)$ for $x \in |X|$ (e is realized by any realizer for f). Obviously, we have $f = me$. As m is regular there are morphisms $g, h : Y \to W$ such that m is an equalizer of g and h. Due to Lemma 4.6 the maps g and h can be chosen from $\mathbf{Mod}(\mathcal{A})$ provided Y is in $\mathbf{Mod}(\mathcal{A})$. As $gf = gme = hme = hf$ and f is epic it follows that $g = h$ and thus m is an isomorphism. Then $|m|$ is an isomorphism from which it follows that $|Z| = |Y|$. Thus $|f|$ is onto as desired. □

Next we discuss colimits. For that purpose we introduce some notation. For sets I_0 and I_1 their *disjoint union* is given by $I_0 + I_1 = \{0\}\times I_0 \cup \{1\}\times I_1$. For $i{=}0,1$ we write $\iota_i : I_i \to I_0 + I_1$ for the map with $\iota_i(z) = \langle i, z\rangle$, i.e. ι_i is the inclusion of the i-th summand into the sum $I_0 + I_1$.

Lemma 4.8. *For every (weak) pca $\mathcal{A}$ the categories $\mathbf{Asm}(\mathcal{A})$ and $\mathbf{Mod}(\mathcal{A})$ have finite colimits which are preserved by $\mathsf{J} : \mathbf{Mod}(\mathcal{A}) \hookrightarrow \mathbf{Asm}(\mathcal{A})$.*

Proof. Let X and Y be assemblies over $\mathcal{A}$. Then their sum is given by the assembly $X{+}Y$ with $|X{+}Y| = |X| + |Y|$, $||\iota_0(x)||_{X+Y} = \{\mathsf{p}\,\mathsf{true}\,a \mid a \in ||x||_X\}$ for all $x \in |X|$ and $||\iota_1(y)||_{X+Y} = \{\mathsf{p}\,\mathsf{false}\,b \mid b \in ||y||_Y\}$ for all $y \in |Y|$. The maps $\iota_0 : X \to X{+}Y$ and $\iota_1 : X \to X{+}Y$ are realized by $\Lambda x.\mathsf{p}\,\mathsf{true}\,x$ and $\Lambda y.\mathsf{p}\,\mathsf{false}\,y$, respectively.
For showing that ι_0 and ι_1 satisfy the desired universal property suppose that $f : X \to Z$ and $g : Y \to Z$ are morphisms in $\mathbf{Asm}(\mathcal{A})$. That there exists a unique morphism $[f,g] : X{+}Y \to Z$ with $[f,g]\circ\iota_0 = f$ and $[f,g]\circ\iota_1 = g$ can be seen as follows. Put $[f,g](\iota_0(x)) = f(x)$ for $x \in |X|$ and $[f,g](\iota_1(y)) = g(y)$ for $y \in |Y|$. As ι_0 and ι_1 are jointly surjective as maps of their underlying sets it is immediate that the so define $[f,g]$ is the unique candidate. Suppose f and g are realized by e_0 and e_1, respectively. As $\mathsf{true} = \Lambda x.\Lambda y.x$ and $\mathsf{false} = \Lambda x.\Lambda y.y$ it is immediate that $[f,g]$ is realized by $\Lambda z.\mathsf{p}_0 z e_0 e_1(\mathsf{p}_1 z)$.
Obviously $X{+}Y$ is modest if X and Y are modest.
The empty sum, i.e. the initial object, is given by the assembly 0 whose underlying set is empty. Obviously, 0 is a modest set.
Suppose $f, g : X \to Y$ in $\mathbf{Asm}(\mathcal{A})$. Let $\sim$ be the least equivalence relation on $|Y|$ such that $f(x) \sim g(x)$ for all $x \in |X|$. We define Q as the assembly with $|Q| = |Y|_{/\sim}$ and $||\,[y]_\sim\,||_Q = \bigcup_{y' \in [y]_\sim} ||y'||_Y$. Obviously, Q is modest if Y is modest. Let $e : Y \to Q$ be the map sending $y \in |Y|$ to $e(y) = [y]_\sim$. It is a morphism in $\mathbf{Asm}(\mathcal{A})$ since it is realized by $\Lambda x.x$. Suppose $h : X \to Z$ with $hf = hg$. Then every $k : Q \to Z$ with $h = ke$ has to satisfy $k([y]_\sim) = h(y)$. As $hf = hg$ and the underlying map of e is onto the map k is well-defined and unique. Every realizer for h is also a realizer for k. Thus $\mathbf{Asm}(\mathcal{A})$ has coequalizers which stay within $\mathbf{Mod}(\mathcal{A})$ if Y is in $\mathbf{Mod}(\mathcal{A})$. □

Thus $\mathbf{Asm}(\mathcal{A})$ and $\mathbf{Mod}(\mathcal{A})$ have coequalizers of all kernel pairs. Moreover, as we shall show next they are so-called *regular categories*.

Recall that in a category $\mathcal{C}$ a morphism $e : X \to Q$ is a *regular epi(morphism)* iff it appears as coequalizer of some pair $f, g : Y \to X$ in $\mathcal{C}$. If $\mathcal{C}$ has finite limits then e is a regular epi iff e is a coequalizer of its kernel pair (exercise!).

Definition 4.2. (regular category)
A category $\mathcal{C}$ is called regular *iff $\mathcal{C}$ has finite limits and coequalizers of kernel pairs and regular epis are stable under pullbacks along arbitrary morphisms in $\mathcal{C}$.* $\Diamond$

Lemma 4.9. *Let $\mathcal{C}$ be a regular category and $f : X \to Y$ a morphism in $\mathcal{C}$. Let $k_0, k_1 : R \to X$ be a kernel pair of f and $e : X \twoheadrightarrow Q$ a coequalizer of k_0 and k_1. Then the unique morphism $m : Q \to Y$ with $m \circ e = f$ is a monomorphism. Thus k_0, k_1 is also a kernel pair of e.*
Moreover, whenever $f = m' \circ f'$ for some mono $m' : Z \rightarrowtail Y$ then there exists a unique mono n making the diagram

$$\begin{tikzcd} X \arrow[r, "f'"] \arrow[d, "e"'] & Z \arrow[d, tail, "m'"] \\ Q \arrow[r, tail, "m"'] \arrow[ur, dashed, "n"'] & Y \end{tikzcd}$$

commute. Thus m is the least subobject of Y through which f factors.

Proof. For showing that $m : Q \to Y$ is monic suppose $m \circ g = m \circ h$ for $g, h : V \to Q$. Consider the pullback

$$\begin{tikzcd} W \arrow[r, "a"] \arrow[d, "\langle p_0, p_1 \rangle"'] \arrow[dr, phantom, "\lrcorner", very near start] & V \arrow[d, "\langle g, h \rangle"] \\ X \times X \arrow[r, "e \times e"'] & Q \times Q \end{tikzcd}$$

As

$$fp_0 = mep_0 = mga = mha = mep_1 = fp_1$$

there is a unique $b : W \to R$ with $\langle k_0, k_1 \rangle \circ b = \langle p_0, p_1 \rangle$. Thus we have

$$ga = ep_0 = ek_0b = ek_1b = ep_1 = ha$$

from which it follows that $g = h$ if we can show that a is epic. As

$$\begin{tikzcd} X \times X \arrow[r, "e \times X"] \arrow[d, "\pi_0"'] \arrow[dr, phantom, "\lrcorner", very near start] & Q \times X \arrow[d, "\pi_0"] \\ X \arrow[r, "e"'] & Q \end{tikzcd} \qquad \begin{tikzcd} Q \times X \arrow[r, "Q \times e"] \arrow[d, "\pi_1"'] \arrow[dr, phantom, "\lrcorner", very near start] & Q \times Q \arrow[d, "\pi_1"] \\ X \arrow[r, "e"'] & Q \end{tikzcd}$$

are pullbacks and regular epis are stable under pullbacks it follows that $e{\times}X$ and $Q{\times}e$ are also regular epis. As $e{\times}e = (Q{\times}e) \circ (e{\times}X)$ it follows that a is a composite of pullbacks of regular epis. Thus a is a composite of regular epis and, therefore, epic itself as desired.
That f and e have the same kernel pair follows from the observation that for all $h_0, h_1 : U \to X$ we have $eh_0 = eh_1$ iff $meh_0 = meh_1$ iff $fh_0 = fh_1$ (as m is monic).
Now suppose $f = m'f'$ for some mono $m' : Z \rightarrowtail Y$. Then f' coequalizes the kernel pair of f as from $m'f'k_0 = fk_0 = fk_1 = m'f'k_1$ it follows that $f'k_0 = f'k_1$. Thus, there exists a unique n with $f' = ne$. Thus, we have also $m'ne = m'f' = f = me$ from which it follows that $m'n = m$ as e is epic. Thus n is monic as well. $\square$

The regular epimorphisms in $\mathbf{Asm}(\mathcal{A})$ and $\mathbf{Mod}(\mathcal{A})$ can be characterized as follows.

Lemma 4.10. *Let $\mathcal{A}$ be a (weak) pca. Then $f : X \to Y$ is a regular epi in $\mathbf{Asm}(\mathcal{A})$ iff there is an $e \in \mathcal{A}$ such that for all $y \in |Y|$ and $a \in ||y||_Y$ there is an $x \in |X|$ with $f(x) = y$ and $e \cdot a \in ||x||_X$. This condition characterizes also regular epis in $\mathbf{Mod}(\mathcal{A})$.*

Proof. Suppose $f : X \to Y$ is a regular epi in $\mathbf{Asm}(\mathcal{A})$. Let Z be the assembly with $|Z| = \{f(x) \mid x \in |X|\}$ and $||z||_Z = \bigcup_{x \in f^{-1}(\{z\})} ||x||_X$ for all $z \in |Z|$. Let $f' : X \to Z$ with $f'(x) = f(x)$ which is realized by $\Lambda x.x$. Then the inclusion $m : Z \hookrightarrow Y$ is realized by any realizer for f. Let k_0, k_1 be a kernel pair of f. Notice that f is a coeqalizer of k_0 and k_1 as f is a regular epi by assumption. As $m \circ f' \circ k_0 = f \circ k_0 = f \circ k_1 = m \circ f' \circ k_1$ and m is monic it follows that $f' \circ k_0 = f' \circ k_1$. Thus, there exists a unique morphism $g : Y \to Z$ with $f' = g \circ f$. Let $e \Vdash g$. Suppose $y \in |Y|$ and $a \in ||y||_Y$. Then $g(y) \in |Z|$ and $e \cdot a \in ||g(y)||_Z$. Thus, there exists $x \in |X|$ with $e \cdot a \in ||x||_X$ and $g(y) = f(x)$. As f is epic and $m \circ g \circ f = m \circ f' = f$ it follows that $m \circ g = id_Y$. Thus $y = m(g(y)) = g(y)$ and, accordingly, we have $y = g(y) = f(x)$ as desired.
Now assume that the right hand side of the claimed equivalence holds for f. First of all notice that this implies that $f : |X| \to |Y|$ is onto. We will show that f is actually a coequalizer of its kernel pair k_0, k_1, i.e. that f is a regular epi. Suppose $g : X \to Z$ with $g \circ k_0 = g \circ k_1$. Then $g(x) = g(x')$ whenever $f(x) = f(x')$. As f is epic we can define a map $h : |Y| \to |Z|$ by sending $y \in |Y|$ to $g(x)$ for some $x \in f^{-1}(\{y\})$. Thus $h(f(x)) = g(x)$ for all $x \in |X|$. As $f : |X| \to |Y|$ is onto h is the unique candidate for a morphism $h : Y \to Z$ with $g = h \circ f$. It remains to show that h is realizable. Let $e' \Vdash g$ then $\Lambda x.e'(ex) \Vdash h$ as if $a \Vdash_Y y$ then $ea \Vdash_X x$ for some $x \in |X|$ with $y = f(x)$ and thus $e'(ea) \Vdash g(x) = h(y)$.
By inspection of this proof since Z is modest if X is modest it follows that the above characterization applies also to $\mathbf{Mod}(\mathcal{A})$. $\square$

Furthermore, Lemma 4.10 gives rise to

Lemma 4.11. *In* $\mathbf{Asm}(\mathcal{A})$ *and* $\mathbf{Mod}(\mathcal{A})$ *regular epis are stable under composition and pullbacks along arbitrary morphisms.*

Proof. Straightforward exercise! $\square$

Now we are ready to prove that

Theorem 4.2. *For every (weak) pca* $\mathcal{A}$ *the categories* $\mathbf{Asm}(\mathcal{A})$ *and* $\mathbf{Mod}(\mathcal{A})$ *are regular.*

Proof. By Lemma 4.2 $\mathbf{Asm}(\mathcal{A})$ has finite limits. As by Lemma 4.8 $\mathbf{Asm}(\mathcal{A})$ has all finite colimits it has in particular coequalizers of kernel pairs. As by Lemma 4.11 regular epis are stable under arbitrary pullbacks it follows that $\mathbf{Asm}(\mathcal{A})$ is a regular category.
This argument restricts to $\mathbf{Mod}(\mathcal{A})$ and thus $\mathbf{Mod}(\mathcal{A})$ is regular as well. $\square$

Next we discuss how $\mathbf{Asm}(\mathcal{A})$ and $\mathbf{Mod}(\mathcal{A})$ give rise to models of first order intuitionistic logic.

Definition 4.3. (subobject fibration)
For every $X \in \mathbf{Asm}(\mathcal{A})$ *let* $\mathsf{Sub}(X)$ *be the preorder of subobjects of* X *where for* $m : P \rightarrowtail X$ *and* $m' : P' \rightarrowtail X$ *we have* $m \leq_X m'$ *iff there exists a unique* $n : P \to P'$ *with* $m'n = m$.
For $f : Y \to X$ *in* $\mathbf{Asm}(\mathcal{A})$ *let* $\mathsf{Sub}(f) : \mathsf{Sub}(X) \to \mathsf{Sub}(Y)$ *be the map sending* $m \in \mathsf{Sub}(X)$ *to* $f^*m \in \mathsf{Sub}(Y)$, *the pullback of* m *along* f

$$\begin{tikzcd} f^*P \arrow[r] \arrow[d, tail, "f^*m"'] \arrow[dr, phantom, "\lrcorner", very near start] & P \arrow[d, tail, "m"] \\ Y \arrow[r, "f"'] & X \end{tikzcd}$$

Obviously $\mathsf{Sub}(f) = f^*$ *is order preserving.*
Although for $g : Z \to Y$ *it need not be the case that* $g^*f^*m = (fg)^*m$ *it holds nevertheless that* $g^*f^*m \cong (fg)^*m$ *which suffices for our purposes.*
Thus, we may consider Sub *as a pseudo-functor*[29] *from* $\mathbf{Asm}(\mathcal{A})^{\mathsf{op}}$ *to* $\mathbf{PreOrd}$, *the category of preorders and monotone maps.* $\diamondsuit$

Theorem 4.3. (quantification for the subobject fibration)
For every $f : Y \to X$ *in* $\mathbf{Asm}(\mathcal{A})$ *the monotone map* $f^* : \mathsf{Sub}(X) \to \mathsf{Sub}(Y)$ *has a left adjoint* $\exists_f$ *and a right adjoint* $\forall_f$, *i.e.* $\exists_f \dashv f^* \dashv \forall_f$.
These quantifiers satisfy the so-called Beck-Chevalley condition (BC), *i.e.* $g^*\exists_f m \cong \exists_p q^*m$ *and* $g^*\forall_f m \cong \forall_p q^*m$ *for all pullbacks*

$$\begin{tikzcd} U \arrow[r, "q"] \arrow[d, "p"'] \arrow[dr, phantom, "\lrcorner", very near start] & X \arrow[d, "f"] \\ Z \arrow[r, "g"'] & Y \end{tikzcd}$$

[29]Here "pseudo" means that composition is preserved only up to isomorphism. For details see vol.1 of [Bor].

in $\mathbf{Asm}(\mathcal{A})$ *and* $m \in \mathsf{Sub}(X)$.

Proof. First we show the existence of $\exists_f \dashv f^*$. For a subobject $m : P \rightarrowtail X$ we construct $\exists_f m$ as follows. Let $e : P \to Q$ be the coequalizer of the kernel pair of $fm : P \to Y$ and $\exists_f m$ the unique map $n : Q \to Y$ with $fm = ne$. From Lemma 4.9 it follows that n is monic and, moreover, that $n \leq_Y n'$ whenever fm factors through n', i.e. $fm = n'f'$ for some f'. Obviously fm factors through n' iff $m \leq_X f^*n'$. On the other hand if $\exists_f m \leq_Y n'$, i.e. $n'n'' = n$ for some n'', then $fm = ne = n'n''e$, i.e. fm factors through n' (via $n''e$), and thus $m \leq_x f^*n'$. Thus, we have $\exists_f m \leq_Y n'$ iff $m \leq_X f^*n'$ for all $n' \in \mathsf{Sub}(Y)$, i.e. $\exists_f \dashv f^*$. The Beck-Chevalley condition holds for existential quantification as monos and regular epis are stable under pullbacks in $\mathbf{Asm}(\mathcal{A})$.
From the explicit construction of coequalizers in the proof of Lemma 4.8 it follows that $\exists_f m$ is (isomorphic to) the subobject $n : Q \rightarrowtail Y$ where $|Q| = \{f(x) \mid x \in |P|\}$ (assuming that $|m| : |P| \hookrightarrow |X|$), $n(y) = y$ and $||y||_Q = \bigcup_{x \in |P| \cap f^{-1}(y)} ||x||_P$.
Next we show that f^* has a right adjoint $\forall_f$. For $m \in \mathsf{Sub}(X)$ we define a map $q_m : |Y| \to \mathcal{P}(\mathcal{A})$ with $e \in q_m(y)$ iff for all $x \in f^{-1}(y)$ and for all $a \in ||x||_X$ there is a (unique) $z \in |P|$ with $m(z) = x$ and $ea \in ||z||_P$. Let Q be the assembly with $|Q| = \{y \in |Y| \mid q_m(y) \neq \emptyset\}$ and $||y||_Q = \{\mathsf{p}ab \mid a \in ||y||_Y \text{ and } b \in q_m(y)\}$ and $n : Q \rightarrowtail Y$ be the mono with $n(y) = y$ which is realized by p_0. It is tedious, but straightforward to check that $n' \leq_Y n$ iff $f^*n' \leq_X m$ for all $n' \in \mathsf{Sub}(Y)$. Thus we may take n for $\forall_f m$.
The Beck-Chevalley condition for universal quantification follows from that for existential quantification (exchanging the roles of f and g and p and q, respectively) because $f^*\exists_g \dashv g^*\forall_f$ and $\exists_q p^* \dashv \forall_p q^*$. $\square$

For morphisms $f : X \to Y$ in $\mathbf{Asm}(\mathcal{A})$ the functors $f^* : \mathsf{Sub}(Y) \to \mathsf{Sub}(X)$ appear as restriction of pullback functors $f^* : \mathbf{Asm}(\mathcal{A})/Y \to \mathbf{Asm}(\mathcal{A})/X$. Now Theorem 4.3 can be strengthened in the sense that these pullback functors f^* have left and right adjoints Σ_f and Π_f, respectively, satisfying a Beck-Chevalley condition.

Theorem 4.4. *For every morphism* $f : X \to Y$ *in* $\mathbf{Asm}(\mathcal{A})$ *the pullback functor* $f^* : \mathbf{Asm}(\mathcal{A})/Y \to \mathbf{Asm}(\mathcal{A})/X$ *has a left adjoint* Σ_f *and a right adjoint* Π_f. *Moreover, these adjunctions satisfy the Beck-Chevalley condition in the sense that for every pullback*

$$\begin{tikzcd} U \arrow[r, "q"] \arrow[d, "p"'] \arrow[dr, phantom, "\lrcorner", very near start] & X \arrow[d, "f"] \\ Z \arrow[r, "g"'] & Y \end{tikzcd}$$

the canonical natural transformations $\sigma : \Sigma_p q^* \to g^*\Sigma_f$ *and* $\tau : g^*\Pi_f \to \Pi_p q^*$

as given by

$$
\begin{array}{c}
q^* \xrightarrow{q^*\eta} q^*f^*\Sigma_f \\
\hline
q^* \longrightarrow p^*g^*\Sigma_f \\
\hline
\Sigma_p q^* \xrightarrow{\sigma} g^*\Sigma_f
\end{array}
\qquad\qquad
\begin{array}{c}
q^*f^*\Pi_f \xrightarrow{q^*\varepsilon} q^* \\
\hline
p^*g^*\Pi_f \longrightarrow q^* \\
\hline
g^*\Pi_f \xrightarrow{\tau} \Pi_p q^*
\end{array}
$$

are isomorphisms.

Proof. The left ajoints Σ_f send objects $h : V \to X$ of $\mathbf{Asm}(\mathcal{A})/X$ to $\Sigma_f h = fh$ and morphisms $k : h' \to h$ in $\mathbf{Asm}(\mathcal{A})$ to $\Sigma_f(k{:}h'{\to}h) = k : fh' \to fh$ in $\mathbf{Asm}(\mathcal{A})/Y$. That $\Sigma_f \vdash f^*$ can be seen from the natural correspondence between $k : \Sigma_f h \to h'$ and $\widetilde{k} : h \to f^*h'$ as depicted in the diagram

$$
\begin{tikzcd}
\cdot \arrow[dr, "\widetilde{k}"] \arrow[drr, "k"] \arrow[ddr, "h"'] & & \\
& \cdot \arrow[r] \arrow[d, "f^*h'"] \arrow[dr, phantom, "\lrcorner", very near start] & \cdot \arrow[d, "h'"] \\
& X \arrow[r, "f"'] & Y
\end{tikzcd}
$$

A straightforward diagram chasing shows that σ is even the identity.
The right adjoint Π_f to f^* is constructed as follows. Let $h : V \to X$. We construct $\Pi_f h : P \to Y$ as follows. Let P_0 be the set of all pairs $\langle y, s\rangle$ such that $y \in |Y|$ and $s : f^{-1}(y) \to |V|$ such that $h(s(x)) = x$ for all $x \in f^{-1}(y)$. We say that $e \Vdash \langle y, s\rangle$ iff $\mathsf{p}_0 e \Vdash_Y y$, $\mathsf{p}_1 e{\downarrow}$ and $\mathsf{p}_1 ea \Vdash_V s(x)$ whenever $a \Vdash_X x \in f^{-1}(y)$. Then we define P as the assembly where $|P|$ consists of those $\langle y, s\rangle \in P_0$ with $e \Vdash \langle y, s\rangle$ for some $e \in |\mathcal{A}|$ and $||\langle y, s\rangle||_P = \{e \in |\mathcal{A}| \mid e \Vdash \langle y, s\rangle\}$. Finally $\Pi_f h : P \to Y$ sends $\langle y, s\rangle$ to y and is thus realized by p_0. The counit $\varepsilon_h : f^*\Pi_f h \to h$ is given by evaluation, i.e. $\varepsilon_h(\langle x, \langle f(x), s\rangle\rangle) = s(x)$. It is realized by $\Lambda e.\mathsf{p}_1(\mathsf{p}_1 e)(\mathsf{p}_0 e)$.
Showing that Beck-Chevalley condition holds for Π we leave as an exercise to the inclined reader. $\square$

Theorem 4.4 provides the basis for showing how Martin-Löf's dependent type theory can be interpreted in categories of assemblies. Of course, dependent sum types are interpreted by Σ and dependent product types are interpreted by Π. For more details see [St, Jac]. Notice also that Theorem 4.4 restricts to $\mathbf{Mod}(\mathcal{A})$ and thus Martin-Löf type theory can be interpreted within the comparatively small model of modest sets (see [Bau] for details).

After having established quantification for $\mathbf{Asm}(\mathcal{A})$ in Theorem 4.3 we now show that we can interpret propositional connectives.

Theorem 4.5. *For every X in $\mathbf{Asm}(\mathcal{A})$ the preorder $\mathsf{Sub}(X)$ is a Heyting (pre)lattice (i.e. finitely complete and cocomplete and cartesian closed as a category) and for every morphism $f : Y \to X$ in $\mathbf{Asm}(\mathcal{A})$ the reindexing map $f^* : \mathsf{Sub}(X) \to \mathsf{Sub}(Y)$ preserves this structure.*

Proof. Empty meets and joins in $\mathsf{Sub}(X)$ are given by $id_X : X \to X$ and $0 \to X$, respectively (where 0 is the initial object). For constructing binary meets and joins suppose $m_0 : P_0 \rightarrowtail X$ and $m_1 : P_1 \rightarrowtail X$ are monos. Their meet is given by the pullback

$$\begin{tikzcd}
P_0 \wedge P_1 \arrow[r, tail] \arrow[d, tail] \arrow[dr, tail, "m_0 \wedge m_1" description] \arrow[dr, phantom, very near start, "\lrcorner"] & P_1 \arrow[d, tail, "m_1"] \\
P_0 \arrow[r, tail, "m_0"'] & X
\end{tikzcd}$$

Let $m \circ e = [m_0, m_1]$ where e is a regular epi and m is a mono. Then $m_i \leq_X m$ via $e \circ \iota_i$. If $n : Q \rightarrowtail X$ with $m_i \leq_X n$ for $i = 0, 1$. Let h_i be the unique map with $n \circ h_i = m_i$. Then $n \circ [h_0, h_1] = [m_0.m_1]$ from which it follows by Lemma 4.9 that $m \leq_X n$. Thus we have shown that m is a supremum of m_0 and m_1.
That the exponential $m_0{\to}m_1$ is given by $\forall_{m_0} m_0^* m_1$ can be seen as folllows. For $m \in \mathsf{Sub}(X)$ we have $m \leq_X m_0{\to}m_1$ iff $m_0^* m \leq_{P_0} m_0^* m_1$ iff $m_0 \circ m_0^* m \leq_X m_0 \circ m_0^* m_1$ iff $m_0 \wedge m \leq_X m_0 \wedge m_1$ iff $m_0 \wedge m \leq_X m_1$.
That $f^* : \mathsf{Sub}(Y) \to \mathsf{Sub}(X)$ preserves (finite) meets and joins follows from the fact that (by Theorem 4.3) the map f^* has a left and a right adjoint.
For showing that f^* preserves Heyting implication (i.e. exponentiation) instantiate the Beck-Chevalley condition for $\forall$ by the pullback

$$\begin{tikzcd}
\cdot \arrow[r, "q"] \arrow[d, "p"'] \arrow[dr, phantom, very near start, "\lrcorner"] & P_0 \arrow[d, "m_0"] \\
Y \arrow[r, "f"'] & X
\end{tikzcd}$$

from which it follows that

$$f^*(m_0{\to}m_1) = f^* \forall_{m_0} m_0^* m_1 \cong \forall_p q^* m_0^* m_1 \cong \forall_p p^* f^* m_1 = f^* m_0{\to}f^* m_1$$

since $p = f^* m_0$. $\square$

Theorems 4.3 and 4.5 guarantee that one may interpret *first order intuitionistic logic* in $\mathbf{Asm}(\mathcal{A})$ and also in $\mathbf{Mod}(\mathcal{A})$ because Theorems 4.3 and 4.5 restrict to $\mathbf{Mod}(\mathcal{A})$ (for details see [Bau]). Equality predicates on X are interpreted as $\delta_X = \langle id_X, id_X \rangle \in \mathsf{Sub}(X{\times}X)$.
In $\mathbf{Asm}(\mathcal{A})$ we can also interpret higher order intuitionistic logic (to some extent) because there is a generic mono in $\mathbf{Asm}(\mathcal{A})$.

Theorem 4.6. *Let* $\mathsf{Prop} = \nabla(\mathcal{P}(|\mathcal{A}|))$ *and* Tr *be the assembly with* $|\mathsf{Tr}| = \mathcal{P}(\mathcal{A}) \setminus \{\emptyset\}$ *and* $||p||_{\mathsf{Tr}} = p$ *for all* $p \in |\mathsf{Tr}|$*. Further let* $\mathsf{tr} : \mathsf{Tr} \rightarrowtail \mathsf{Prop}$ *be the inclusion of* $|\mathsf{Tr}|$ *into* $\mathcal{P}(\mathcal{A})$*. This monomorphism* $\mathsf{tr} : \mathsf{Tr} \rightarrowtail \mathsf{Prop}$ *is* generic *in the sense that for every mono* $m : P \rightarrowtail X$ *there exists a map* $p : X \to \mathsf{Prop}$ *with*

$$\begin{tikzcd} P \arrow[r] \arrow[d, tail, "m"'] \arrow[dr, phantom, "\lrcorner", very near start] & \mathsf{Tr} \arrow[d, tail, "\mathsf{tr}"] \\ X \arrow[r, "p"'] & \mathsf{Prop} \end{tikzcd}$$

which, however, in general is not unique with this property.

Proof. For a subobject $m : P \rightarrowtail X$ an appropriate $p : X \to \mathsf{Prop}$ is given by $p(x) = \{e \in |\mathcal{A}| \mid \exists z \in m^{-1}(x).\ e \in ||z||_P\}$. $\square$

The mono $m = id_1 : 1 \rightarrowtail 1$ is isomorphic to $p^*\mathsf{tr}$ for all $p : 1 \to \mathsf{Prop}$ with $p(*) \neq \emptyset$. Thus, in general there is not a unique p with $m \cong p^*\mathsf{tr}$. This argument just shows that the particular tr as defined above is not a subobject classifier.
That there cannot exist any subobject classifier in $\mathbf{Asm}(\mathcal{A})$ for nontrivial $\mathcal{A}$ can be seen quite easily as follows. If $\mathbf{Asm}(\mathcal{A})$ had a subobject classifier then $\mathbf{Asm}(\mathcal{A})$ were a topos (as it has finite limits and is cartesian closed). This, however, is impossible as $\mathbf{Asm}(\mathcal{A})$ is not balanced because the reflection map $\eta_2 : 2 \to \nabla(\Gamma(2))$ is monic and epic but not an isomorphism.
There cannot exist a generic mono in $\mathbf{Mod}(\mathcal{A})$ for nontrivial $\mathcal{A}$ as the assembly $\Delta(\mathcal{A})$ with $|\Delta(\mathcal{A})| = |\mathcal{A}|$ and $||a||_{\Delta(\mathcal{A})} = \{a\}$ has at least $2^{|\mathcal{A}|}$ subobjects whereas there are at most $|\mathcal{A}|$ morphisms from $\Delta(\mathcal{A})$ to Prop if Prop were modest.

Intuitionistic higher order logic can be interpreted in $\mathbf{Asm}(\mathcal{A})$ as follows. For every assembly X let Prop^X be the *type of predicates on* X. The *elementhood predicate* $\in_X \rightarrowtail X \times \mathsf{Prop}^X$ is obtained as pullback of the generic mono tr along $\mathsf{ev} \circ \langle \pi_2, \pi_1 \rangle$. Obviously, for every $r : R \rightarrowtail X \times Y$ there exists a map $\rho : Y \to \mathsf{Prop}^X$ such that

$$\begin{tikzcd} R \arrow[r] \arrow[d, tail, "r"'] \arrow[dr, phantom, "\lrcorner", very near start] & \in_X \arrow[d, tail] \\ X \times Y \arrow[r, "X \times \rho"'] & X \times \mathsf{Prop}^X \end{tikzcd}$$

which guarantees that the *comprehension axiom* of higher order logic is validated by its interpretation in $\mathbf{Asm}(\mathcal{A})$.

From Theorems 4.5 and 4.6 it follows that there are maps $\top, \bot : 1 \to \mathsf{Prop}$ and $\wedge, \vee, \to : \mathsf{Prop} \times \mathsf{Prop} \to \mathsf{Prop}$ such that

(1) $id_X \cong (\top \circ !_X)^*\mathsf{tr}$ and $0 \rightarrowtail X$ is isomorphic to $(\bot \circ !_X)^*\mathsf{tr}$

(2) $p^*\mathsf{tr} \,\Box\, q^*\mathsf{tr} \cong (\Box \circ \langle p, q \rangle)^*\mathsf{tr}$ for all $p, q : X \to \mathsf{Prop}$ and $\Box \in \{\wedge, \vee, \to\}$

i.e. all propositional connectives can be expressed as operations on Prop in $\mathbf{Asm}(\mathcal{A})$.
For sake of convenience we explicitate canonical choices of these operations, namely

$$\top = |\mathcal{A}| \quad \text{and} \quad \bot = \emptyset$$

$$p \wedge q = \{\langle a, b\rangle \mid a \in p \text{ and } b \in q\}$$

$$p \vee q = \{\langle \mathsf{true}, a\rangle \mid a \in p\} \cup \{\langle \mathsf{false}, b\rangle \mid b \in q\}$$

$$p \to q = \{e \in |\mathcal{A}| \mid \forall a \in p.\ a \in p \Rightarrow e \cdot a \in q\}$$

for $p, q \in \mathcal{P}(\mathcal{A})$, which make clear the connection to traditional realizability interpretations.
But quantifiers can also be "internalized" as follows. For every $X \in \mathbf{Asm}(\mathcal{A})$ there are morphisms $\exists_X, \forall_X : \mathsf{Prop}^X \to \mathsf{Prop}$ such that for $r : Y \times X \to \mathsf{Prop}$ we have $(Q_X \circ \lambda(r))^*\mathsf{Tr} \cong \forall_\pi(r^*\mathsf{Tr})$ for $Q \in \{\exists, \forall\}$. Explicitly these internal quantifiers are given by

$$\exists_X(p) = \{\langle a, b\rangle \mid a \Vdash_X x \text{ and } b \in p(x) \text{ for some } x \in |X|\}$$

and

$$\forall_X(p) = \{e \in \mathcal{A} \mid ea \in p(x) \text{ whenever } a \Vdash_X x\}$$

for $p : |X| \to \mathcal{P}(\mathcal{A})$.

One can show that the interpretation of higher order intuitionistic logic in $\mathbf{Asm}(\mathcal{A})$ validates the *Axiom of Unique Choice* (AUC)

$$\forall R \in \mathsf{Prop}^{X \times Y} \big(\forall x{:}X.\exists! y{:}Y.R(x, y) \to \exists f{:}Y^X.\forall x{:}X.R(x, f(x))\big)$$

for all $X, Y \in \mathbf{Asm}(\mathcal{A})$. However, in general the Axiom of Choice (AC) is not validated by interpretation in $\mathbf{Asm}(\mathcal{A})$. For example $\mathbf{Asm}(\mathcal{K}_1)$ validates

$$\forall f{:}N^N.\exists n{:}N.\{n\} = f$$

but not

$$\exists F{:}N^{N^N}.\forall f{:}N^N.\{F(f)\} = f$$

as otherwise equality of total recursive functions were decidable (see [Ro]).

Although the *extensionality principle for functions*, i.e.

$$\forall f, g{:}Y^X.\big(\forall x{:}X.f(x) = g(x)\big) \to f = g$$

holds in arbitrary realizability models the *extensionality principle for predicates*, i.e.

$$\forall P, Q \in \mathsf{Prop}^X.\big(\forall x{:}X.P(x) \leftrightarrow Q(x)\big) \to P = Q$$

fails for nontrivial $\mathcal{A}$ because it entails that $\mathsf{tr} : \mathsf{Tr} \rightarrowtail \mathsf{Prop}$ is a subobject classifier.

Thus it may appear as desirable to enlarge $\mathbf{Asm}(\mathcal{A})$ to a topos $\mathbf{RT}(\mathcal{A})$, the so-called *realizability topos* over $\mathcal{A}$. The traditional construction of realizability toposes will be presented in the next section. It is not based on $\mathbf{Asm}(\mathcal{A})$ and rather identifies $\mathbf{Asm}(\mathcal{A})$ as a certain full subcategory of $\mathbf{RT}(\mathcal{A})$, namely that of the so-called *$\neg\neg$-separated objects*.
An alternative construction of $\mathbf{RT}(\mathcal{A})$ from $\mathbf{Asm}(\mathcal{A})$ is by "adding quotients" (see [CFS]). The new objects are pairs (X, E_X) where X is an object of $\mathbf{Asm}(\mathcal{A})$ and $E_X \rightarrowtail X \times X$ is an equivalence relation on X. The morphisms from (X, E_X) to (Y, E_Y) will be those relations $F \rightarrowtail X \times Y$ validating the requirements

$$F(x, y) \wedge E_X(x, x') \wedge E_Y(y, y') \to F(x', y')$$

$$F(x, y) \wedge F(x, y') \to E_Y(y, y')$$

$$\forall x{:}X.\exists y{:}Y.F(x, y)$$

of *congruence* (w.r.t. E_X and E_Y), *single-valuedness* and *totality*, respectively. Composition of these morphism is given by ordinary relational composition, i.e. $(G \circ F)(x, z) \equiv \exists y{:}Y.F(x, y) \wedge G(y, z)$, and the identity on (X, E_X) is given by E_X itself. Then it is a tedious, but straightforward task to verify that the ensuing category obtained by "adding quotients" is actually a topos. The subobject classifier Ω will be provided by $(\mathsf{Prop}, \leftrightarrow)$.
Notice that this construction can be considered as a logical interpretation of higher order intuitionistic logic with extensionality principle for predicates in higher order intuitionistic logic without this principle.

We conclude this section with a remark on classical logic within $\mathbf{Asm}(\mathcal{A})$. It is an easy exercise(!) to show that the regular monos $P \rightarrowtail X$ are precisely those subobjects of X for which $\forall x{:}X.\neg\neg P(x) \to P(x)$ holds in $\mathbf{Asm}(\mathcal{A})$. Thus, the regular monos into X can be considered as the classical predicates from which it follows that they satisfy the usual closure properties as known from the $\neg\neg$-translation[30]. It is shown easily (exercise) that $\nabla(0 : 1 \to 2)$ classifies regular monos in $\mathbf{Asm}(\mathcal{A})$, i.e. that $\nabla(0)$ is a regular mono and that for every regular mono $m : P \rightarrowtail X$ there exists a *unique* map $\chi : X \to \nabla(2)$ with

$$\begin{tikzcd} P \arrow[r] \arrow[d, tail, "m"'] \arrow[dr, phantom, "\lrcorner", very near start] & \nabla(1) \arrow[d, tail, "\nabla(0)"] \\ X \arrow[r, "\chi"'] & \nabla(2) \end{tikzcd}$$

namely $\chi(x) = 0$ iff $x = m(z)$ for some $z \in |P|$.

[30]of classical into intuitionistic logic as devised by Gödel and Gentzen independently in the early 1930ies

5 Realizability Triposes and Toposes

In this section for every (weak) pca $\mathcal{A}$ we introduce the *realizability tripos* $\mathcal{H}(\mathcal{A})$ and the *realizability topos* $\mathbf{RT}(\mathcal{A})$ following the original approach as can be found in [HJP] (and implicitly in [Hyl]).

Definition 5.1. (realizability tripos)
Let $\mathcal{A}$ be a (weak) pca. Then the functor $\mathcal{H}(\mathcal{A}) : \mathbf{Set}^{\mathsf{op}} \to \mathbf{PreOrd}$ is defined as follows. For every $I \in \mathbf{Set}$ let $\mathcal{H}(\mathcal{A})(I)$ be the preorder $(\mathcal{P}(\mathcal{A})^I, \vdash_I)$ where $\phi \vdash_I \psi$ iff there exists $e \in \mathcal{A}$ such that $\forall i{\in}I. \forall a{\in}\phi(i).\, ea \in \psi(i)$ (where $ea \in \psi(i)$ means that ea is defined and an element of $\psi(i)$). For $f : J \to I$ in $\mathbf{Set}$ the map $\mathcal{H}(\mathcal{A})(f) : \mathcal{P}(\mathcal{A})^I \to \mathcal{P}(\mathcal{A})^J$ sends ϕ to $\mathcal{H}(\mathcal{A})(f)(\phi) = f^*\phi = \phi \circ f$. $\diamondsuit$

Using notation from the previous section we have $\phi \vdash_I \psi$ iff $\bigcap_{i \in I} \phi(i) {\to} \psi(i)$ is nonempty. Thus, obviously, from $\phi \vdash_I \psi$ it follows that $f^*\phi \vdash_J f^*\psi$. Moreover, we have $id^*\phi = \phi$ and $g^*f^*\phi = (fg)^*\phi$ from which it follows that $\mathcal{H}(\mathcal{A})$ is actually a functor.
Now we will show (in several steps) that $\mathcal{H}(\mathcal{A})$ provides a model for higher order intuitionistic logic. For the rest of this section let $\mathcal{A}$ be an arbitrary, but fixed (weak) pca.

Lemma 5.1. *All $\mathcal{H}(\mathcal{A})(I)$ are Heyting prelattices and all reindexing functions $\mathcal{H}(\mathcal{A})(f) : \mathcal{H}(\mathcal{A})(I) \to \mathcal{H}(\mathcal{A})(J)$ preserve this structure.*

Proof. A terminal object in $\mathcal{H}(\mathcal{A})(I)$ is given by any constant function from I to $\mathcal{P}(\mathcal{A})$ with nonempty value (e.g. $\mathcal{A}$). An infimum (or product) of ϕ and ψ is given by $(\phi \wedge \psi)(i) = \phi(i) \wedge \psi(i) = \{\langle a, b\rangle \mid a \in \phi(i) \text{ and } b \in \psi(i)\}$. Heyting implication in $\mathcal{H}(\mathcal{A})(I)$ is given by (exercise!)

$$(\phi \to \psi)(i) = \phi(i) \to \psi(i) = \{e \in \mathcal{A} \mid \forall a \in \phi(i).\, ea \in \psi(i)\}$$

An initial object of $\mathcal{H}(\mathcal{A})(I)$ is given by the constant function with value $\emptyset$. A join (or sum) of ϕ and ψ is given by

$$(\phi \vee \psi)(i) = \phi(i) \vee \psi(i) = \{\langle \mathsf{true}, a\rangle \mid a \in \phi(i)\} \cup \{\langle \mathsf{false}, b\rangle \mid b \in \psi(i)\}$$

From the pointwise construction of these logical operations it is obvious that they are preserved by reindexing. $\square$

Notice that reindexing preserves the logical operations as chosen in the proof of Lemma 5.1 "on the nose", i.e. up to equality.

Lemma 5.2. *For every $f : J \to I$ in $\mathbf{Set}$ the reindexing map f^* has a left adjoint $\exists_f$ and a right adjoint $\forall_f$. These adjoints satisfy the Beck-Chevalley condition, i.e. for every pullback*

$$\begin{tikzcd} L \arrow[r, "q"] \arrow[d, "p"'] \arrow[dr, phantom, "\lrcorner", very near start] & J \arrow[d, "f"] \\ K \arrow[r, "g"'] & I \end{tikzcd}$$

we have $g^*\exists_f \cong \exists_p q^*$ *and* $g^*\forall_f \cong \forall_p q^*$.

Proof. Let $eq(i,j) = \{a \in \mathcal{A} \mid i = j\}$.
For $f : J \to I$ in **Set** the left adjoint $\exists_f$ to f^* is given by

$$\exists_f(\phi)(i) = \bigcup_{j \in J} eq(f(j), i) \wedge \phi(j)$$

and the right adjoint $\forall_f$ to f^* is given by

$$\forall_f(\phi)(i) = \bigcap_{j \in J} eq(f(j), i) \to \phi(j)$$

We leave the proof that these are actually adjoints and that they satisfy the Beck-Chevally condition as an exercise(!) for the inclined reader. □

Lemma 5.3. *Let* $\Omega = \mathcal{P}(\mathcal{A})$ *and* $T = id_\Omega \in \mathcal{H}(\Omega)$. *Then* $T \in \mathcal{H}(\mathcal{A})(\Omega)$ *is a* generic predicate *in the sense that for all* $\phi \in \mathcal{H}(\mathcal{A})(I)$ *there exists a map* $f : I \to \Omega$ *with* $f^*T \cong \phi$.

Proof. Take ϕ for f. □

Notice that in general for $\phi \in \mathcal{H}(\mathcal{A})(I)$ there will be many different f with $\phi \cong f^*T$.

Corollary 5.1. *For every set* I *there is a predicate* $In_I \in \mathcal{H}(\mathcal{A})(I \times \Omega^I)$ *such that for every* $\rho \in \mathcal{H}(\mathcal{A})(I \times J)$ *there exists a map* $r : J \to \Omega^I$ *such that* $\rho \cong (id_I \times r)^* In_I$.

Proof. Define In_I as $In_I(i,p) = p(i)$ for $i \in I$ and $p \in \Omega^I$. For $\rho \in \mathcal{H}(\mathcal{A})(I \times J)$ take $r(j) = \lambda i{:}I.\rho(i,j)$. □

In [HJP] Hyland, Johnstone and Pitts have introduced the notion of *tripos* (for "*t*opos *r*epresenting *i*ndexed *pos*et"), namely (pseudo)functors $\mathcal{H} : \mathbf{Set}^{\mathsf{op}} \to$ **pHa** (where **pHa** is the category of pre-Heyting-algebras and morphism preserving the structure up to isomorphism) satisfying the requirements of Lemma 5.2 and Lemma 5.3. Triposes $\mathcal{H}$ provide a notion of model for higher order intuitionistic logic in the sense that $\mathcal{H}(I)$ is the pre-Heyting-algebra of *predicates on* I, left and right adjoints to reindexing provide existential and universal quantification, respectively, and the structure provided in Cor. 5.1 allows one to interpret types of predicates (as Ω^I), predication (via In_I) and comprehension (via the r assoiated with a ρ).
For every set I there is an equality predicate $eq_I = \exists_{\delta_I}(\top_I) \in \mathcal{H}(I \times I)$ which is isomorphic (exercise!) to the predicate $\forall P{\in}\Omega^I.In_I(i,P) \to In_I(j,P)$.[31]
We leave it as an exercise to explicitate the interpretation of higher order logic in (realizability) triposes (for details see [HJP]).

[31] Notice that $\exists_{\delta_I}(\top_I)$ is available even if one does not postulate a generic predicate T.

In [HJP] it has been shown[32] how to associate with every tripos $\mathcal{H}$ a topos $\mathbf{Set}[\mathcal{H}]$. In case of $\mathcal{H}(\mathcal{A})$ we get the so-called *realizability topos* $\mathbf{RT}(\mathcal{A}) = \mathbf{Set}[\mathcal{H}(\mathcal{A})]$ as it was introduced originally in [HJP, Hyl]. This tripos-to-topos construction essentially consists in "adding quotients of equivalence relations" and is spelled out in the following definition.

Definition 5.2. (realizability topos)
Let $\mathcal{H}(A)$ be a realizability tripos. The associated (realizability) topos $\mathbf{RT}(\mathcal{A})$ $= \mathbf{Set}[\mathcal{H}(\mathcal{A})]$ *is defined as follows. Its objects are pairs $X = (|X|, E_X)$ where $|X|$ is a set and $E_X \in \mathcal{H}(|X|\times|X|)$ such that*

(symm) $\quad E_X(x,y) \vdash E_X(y,x)$
(trans) $\quad E_X(x,y) \wedge E_X(y,z) \vdash E_X(x,z)$

We write $E_X(x)$ as an abbreviation for $E_X(x,x)$.[33] *Morphisms from X to Y in $\mathbf{Set}[\mathcal{H}(\mathcal{A})]$ are given by $F \in \mathcal{H}(|X|\times|Y|)$ satisfying*

(strict) $\quad F(x,y) \vdash E_X(x) \wedge E_Y(y)$
(cong) $\quad E_X(x,x') \wedge E_Y(y,y') \wedge F(x,y) \vdash F(x',y')$
(singval) $\quad F(x,y) \wedge F(x,y') \vdash E_Y(y,y')$
(tot) $\quad E_X(x) \vdash \exists y{:}|Y|.F(x,y)$

which are identified up to logical equivalence. We write $[F]$ for the morphism determined by F. Obviously $[F]$ and $[F']$ are equal iff $F(x,y) \vdash F'(x,y)$ and $F'(x,y) \vdash F(x,y)$. If $[F] : X \to Y$ and $[G] : Y \to Z$ then their composition in $\mathbf{Set}[\mathcal{H}(\mathcal{A})]$ is given by $[H]$ where $H(x,z) \equiv \exists y{:}|Y|.F(x,y) \wedge G(y,z)$. The identity morphism on X is the equivalence class $[E_X]$. $\Diamond$

One easily checks that composition and identity maps satisfy the required properties. Notice, moreover, that $[F] = [F']$ already if $F(x,y) \vdash F'(x,y)$.
The construction of Definition 5.2 applies also to general triposes $\mathcal{H} : \mathcal{C}^{\mathsf{op}} \to \mathbf{pHa}$ giving rise to $\mathcal{C}[\mathcal{H}]$. For example $\mathsf{Sub} : \mathbf{Asm}(\mathcal{A})^{\mathsf{op}} \to \mathbf{pHa}$ gives rise to $\mathbf{Asm}(\mathcal{A})[\mathsf{Sub}]$ which is equivalent to $\mathbf{RT}(\mathcal{A})$. This amounts to the construction of $\mathbf{RT}(\mathcal{A})$ from $\mathbf{Asm}(\mathcal{A})$ as in [CFS] (see also penultimate paragraph of section 4). Every topos $\mathcal{E}$ arises in this way because $\mathcal{E}$ is equivalent to $\mathcal{E}[\mathsf{Sub}_{\mathcal{E}}]$. Also *sheaf toposes* over a complete Heyting algebra A arise in this way as $\mathrm{Sh}(A) = \mathbf{Set}[\mathcal{H}(A)]$ where $\mathcal{H}(A)(I) = A^I$, $\phi \vdash_I \psi$ iff $\phi(i) \leq_A \psi(i)$ for all $i \in I$ and $\mathcal{H}(A)(f)(\phi) = \phi \circ f$.

We next establish step by step that $\mathbf{RT}(\mathcal{A})$ satisfies all the properties required for a topos.

Lemma 5.4. *The category $\mathbf{RT}(\mathcal{A})$ has finite limits.*

[32] In [HJP] they considered triposes $\mathcal{H} : \mathcal{C}^{\mathsf{op}} \to \mathbf{pHa}$ over arbitrary base categories $\mathcal{C}$ with finite limits and have shown how to construct a topos $\mathcal{C}[\mathcal{H}]$ from a tripos $\mathcal{H}$.

[33] We read $E_X(x,y)$ as the proposition "x and y are equal elements of X" and $E_X(x)$ as the proposition "x exists as an element of X".

Proof. A terminal object is given by $1 = (\{*\}, E_1)$ where $E_1(*, *) = \top$. For an object X in $\mathbf{RT}(\mathcal{A})$ the terminal projection $t_X : X \to 1$ is given by $[T_X]$ where $T_X(x, *) \equiv E_X(x)$.
Let $[F] : X \to Z$ and $[G] : Y \to Z$. Then their pullback is given by $[P] : W \to X$ and $[Q] : W \to Y$ where $|W| = |X| \times |Y|$,

$$E_W((x, y), (x', y')) \equiv E_X(x, x') \wedge E_Y(y, y') \wedge \exists z{:}|Z|.\, F(x, z) \wedge G(y, z)$$

and P and Q are defined as $P((x, y), x') \equiv E_W((x, y)) \wedge E_X(x, x')$ and $Q((x, y), y') \equiv E_W((x, y)) \wedge E_Y(y, y')$, respectively.
We leave the straightforward verification of the required universal properties to the inclined reader. $\square$

Notice that a product $X \times Y$ of X and Y in $\mathbf{RT}(\mathcal{A})$ is given by $|X \times Y| = |X| \times |Y|$ and $E_{X\times Y}((x, y), (x', y')) \equiv E_X(x, x') \wedge E_Y(y, y')$.

Lemma 5.5. *The category* $\mathbf{RT}(\mathcal{A})$ *has exponentials.*

Proof. For objects X and Y of $\mathbf{RT}(\mathcal{A})$ their exponential Y^X can be constructed as follows. We put $|Y^X| = \mathcal{H}(\mathcal{A})(|X|\times|Y|)$ and define the equality predicate E_{Y^X} as follows: for $F, G \in \mathcal{H}(\mathcal{A})(|X|\times|Y|)$ let $E_{Y^X}(F, G)$ be the conjunction

$$\text{(strict)} \wedge \text{(cong)} \wedge \text{(singval)} \wedge \text{(tot)} \wedge \forall (x, y){:}|X|\times|Y|. F(x, y) \leftrightarrow G(x, y)$$

where (strict), (cong), (singval) and (tot) are as in Def. 5.2 but with $\vdash$ replaced by $\to$ and all free variables universally quantified. The evaluation map is given by $[Ev] : Y^X \times X \to Y$ where $Ev((F, x), y) \equiv E_{Y^X}(F) \wedge F(x, y)$. Again the straightforward verification of the desired universal property is left to the inclined reader. $\square$

Before embarking on the construction of a subobject classifier in $\mathbf{RT}(\mathcal{A})$ we give a characterisation of monos in $\mathbf{RT}(\mathcal{A})$. Obviously, a map $[M] : Y \to X$ is monic iff $M(y, x) \wedge M(y', x) \vdash E_Y(y, y')$. For such a mono $[M]$ we can now construct a predicate $P \in \mathcal{H}(\mathcal{A})(|X|)$ putting $P(x) \equiv \exists y{:}|Y|.M(y, x)$ which satisfies the properties

$$\begin{array}{ll} \text{(strict)} & P(x) \to E_X(x) \\ \text{(cong)} & P(x) \wedge E_X(x, x') \to P(x') \end{array}$$

Now for every $P \in \mathcal{H}(\mathcal{A})(|X|)$ satisfying (strict) and (cong) one easily checks (exercise!) that $[M_P] : X_P \rightarrowtail X$ is monic where $|X_P| = |X|$, $E_{X_P}(x, x') \equiv E_X(x, x') \wedge P(x)$ and $M_P(x', x) \equiv P(x') \wedge E_X(x', x)$. One also checks easily that for every mono $[M] : Y \rightarrowtail X$ the subobject $[M_P]$ is isomorphic to $[M]$ where $P(x) \equiv \exists y{:}|Y|.M(y, x)$.

Lemma 5.6. *The category* $\mathbf{RT}(\mathcal{A})$ *has a subobject classifier* $t : 1 \to \Omega$*, i.e.* t *is monic and for every mono* $m : Y \rightarrowtail X$ *in* $\mathbf{RT}(\mathcal{A})$ *there exists a unique map*

$\chi_m : X \to \Omega$ *with*

$$\begin{tikzcd} Y \arrow[r] \arrow[d, tail, "m"'] \arrow[dr, phantom, "\lrcorner", very near start] & 1 \arrow[d, tail, "t"] \\ X \arrow[r, "\chi_m"'] & \Omega \end{tikzcd}$$

Proof. Let Ω be the object in $\mathbf{RT}(\mathcal{A})$ with $|\Omega| = \mathcal{P}(\mathcal{A})$ and $E_\Omega(p, q) \equiv p{\leftrightarrow}q$ which, obviously, is symmetric and transitive. Let $t : 1 \to \Omega$ be the map $[T]$ with $T(*, p) \equiv p$.
Obviously, the map t is monic (as 1 is terminal). Let $m = [M] : Y \rightarrowtail X$. Define P as in the remark after Lemma 5.5, namely as $P(x) \equiv \exists y{:}|Y|.M(y, x)$. Now we define χ_m as $[X_M]$ where $X_M(x, p) \equiv P(x){\leftrightarrow}p$. One easily checks that $\chi_m^* t$ is isomorphic to m because $\chi_m^* t$ is isomorphic to $[M_P]$ as in the remark after Lemma 5.5.
Uniqueness of classifying maps can be seen as follows. Let $\chi_1, \chi_2 : X \to \Omega$ and X_1, X_2 with $\chi_i = [X_i]$ for $i{=}1, 2$. Define $P_i \in \mathcal{H}(\mathcal{A})(|X|)$ as $P_i(x) \equiv X_i(x, \top)$. One easily sees that the P_i satisfy (strict) and (cong). Now if M_{P_1} and M_{P_2} are isomorphic as subobjects of X one can check that $P_1 \leftrightarrow P_2$ from which it follows that $X_1 \leftrightarrow X_2$ and thus $\chi_1 = \chi_2$ as desired. ☐

Obviously, the truth value object Ω of $\mathbf{RT}(\mathcal{A})$ has precisely two global elements, namely $t : 1 \to \Omega$ and $f : 1 \to \Omega$ given by $p \mapsto \top{\leftrightarrow}p$ and $p \mapsto \bot{\leftrightarrow}p$, respectively. Thus $\mathbf{RT}(\mathcal{A})$ is 2-valued. However, the topos $\mathbf{RT}(\mathcal{A})$ is not wellpointed as otherwise it were boolean (see e.g. [St2]) which is only the case iff $\mathcal{A}$ is trivial (as we shall see soon in Cor. 5.2).

Now we will identify $\mathbf{Asm}(\mathcal{A})$ as equivalent to a full subcategory of $\mathbf{RT}(\mathcal{A})$, namely the $\neg\neg$-separated objects of $\mathbf{RT}(\mathcal{A})$.

Definition 5.3. (separated objects of a topos)
An object X of a topos $\mathcal{E}$ is called $\neg\neg$-separated *(or simply* separated*) iff $\forall x, y{:}X.\, \neg\neg x{=}y \to x{=}y$ holds in $\mathcal{E}$. We write* $\mathsf{Sep}_{\neg\neg}(\mathcal{E})$ *(or simply* $\mathsf{Sep}(\mathcal{E})$*) for the ensuing full subcategory of $\mathcal{E}$.* ◊

It is a well-known fact from topos theory (see e.g. [Joh]) that $\mathsf{Sep}(\mathcal{E})$ is a full reflective subcategory of $\mathcal{E}$ where the reflection map preserves finite products (but not equalizers in general since otherwise $\mathsf{Sep}(\mathcal{E})$ were a topos itself!). Moreover, it is known that $\mathsf{Sep}(\mathcal{E})$ is a so-called *quasi-topos*, i.e. a finitely cocomplete regular locally cartesian closed category with a classifier for regular monos.[34]
Obviously, an object X of $\mathbf{RT}(\mathcal{A})$ is separated iff $E_X(x, x')$ is equivalent to $E_X(x) \wedge E_X(x') \wedge \neg\neg E_X(x, x')$. As $\neg\neg p = \bot$ if $p = \bot$ and $\neg\neg p = \top$ otherwise

[34] A category $\mathcal{C}$ is *locally cartesian closed* (lcc) iff $\mathcal{C}$ has finite limits and and for all $f : Y \to X$ the pullback functor $f^* : \mathcal{C}/X \to \mathcal{C}/Y$ has a right adjoint $\Pi_f : \mathcal{C}/Y \to \mathcal{C}/X$. As Π_f is a right adjoint it preserves regular subobjects and thus $\Pi_f m$ is a regular mono whenever m is a regular mono. Thus, regular monos are closed under universal quantification and thus also under implication.

it follows that X is separated iff

$$E_X(x,x') \leftrightarrow \big(E_X(x) \wedge E_X(x') \wedge eq_X(x,x')\big)$$

holds in $\mathcal{H}(\mathcal{A})$ where $eq_X = \neg\neg E_X$, i.e. $eq_X(x,x') = \{a \in \mathcal{A} \mid E_X(x,x') \neq \emptyset\}$. From this observation it follows that a separated object X is isomorphic to the *canonically separated* object X' which is defined as follows. Let $\sim_X$ be the relation on $|X|$ with $x \sim_X x'$ iff $E_X(x,x') \neq \emptyset$. The underlying set of X' is defined as $|X'| = |X|_{/\sim_X}$ and $E_{X'}([x],[x']) = \bigcup\{E_X(x'') \mid x'' \in [x] \cap [x']\}$. This suggests the following general definition of *canonically separated object*.

Definition 5.4. *An object X of $\mathbf{RT}(\mathcal{A})$ is* canonically separated *iff the following conditions hold for all $x, x' \in |X|$*

(1) $E_X(x,x) \neq \emptyset$

(2) $E_X(x,x') \neq \emptyset$ *implies* $x = x'$. $\Diamond$

Thus $\mathsf{Sep}(\mathbf{RT}(\mathcal{A}))$ is equivalent to the full subcategory of canonically separated objects of $\mathbf{RT}(\mathcal{A})$ which in turn is obviously equivalent to $\mathbf{Asm}(\mathcal{A})$.
At this place a short sketch of the history seems to be appropriate. In [HJP] realizability triposes and the ensuing realizability toposes were introduced the first time (following suggestions of D. Scott). Immediately afterwards J.M.E.Hyland provided a detailed investigation of the *effective topos* $\mathcal{E}\!f\!f = \mathbf{RT}(\mathcal{K}_1)$ in [Hyl]. In [Hyl] Hyland observed that $\mathcal{E}\!f\!f$ contains $\mathbf{Set}$ as the full reflective subcategory of $\neg\neg$-sheaves (see e.g. [Joh] for information about sheaves), i.e. that the global sections functor $\Gamma : \mathcal{E}\!f\!f \to \mathbf{Set}$ has a full and faithful right adjoint $\nabla : \mathbf{Set} \to \mathcal{E}\!f\!f$ sending a set S to $\nabla(S) = (S, eq_S)$ where $eq_S(x,y) = \top$ if $x = y$ and $eq_S(x,y) = \bot$ otherwise. From this point of view it appeared as natural to consider the $\neg\neg$-separated objects which – in general topos theoretic terms – are defined as those objects X for which the reflection map $\eta_X : X \to \nabla\Gamma X$ is monic. From this it follows rather immediately that the $\neg\neg$-separated objects are those which arise as subobjects of objects of the form $\nabla(S)$. It was observed already in [Hyl] that every separated object is equivalent to a canonically separated one in the sense of Def. 5.4. Later on (starting around 1985 with an observation by E. Moggi, see section 6) the category $\mathbf{Asm}(\mathcal{A}) \simeq \mathsf{Sep}(\mathbf{RT}(\mathcal{A}))$ was used for the purpose of constructing models of the polymorphic λ-calculus and other impredicative type theories like the *Calculus of Constructions* of Th. Coquand and G. Huet (for details see [St, Jac] and the references in there). As $\mathbf{Asm}(\mathcal{A})$ is wellpointed it is much easier to work in it than in $\mathbf{RT}(\mathcal{A})$. The only thing missing in $\mathbf{Asm}(\mathcal{A})$ are well-behaved quotients which we discuss next.
As $\mathbf{RT}(\mathcal{A})$ is a topos (see [Joh]) it has finite colimits and *exact* quotients in the sense that for every equivalence relation $r = \langle r_1, r_2\rangle : R \rightarrowtail X \times X$ the coeqalizer $q : X \twoheadrightarrow Q$ of r_1 and r_2 has the pleasant property that (r_1, r_2) is the kernel pair of q. To illustrate this consider the equivalence relation $R \rightarrowtail \mathsf{Prop} \times \mathsf{Prop}$ induced by the predicate $(p,q) \mapsto p \leftrightarrow q$ on $\mathsf{Prop} \times \mathsf{Prop}$. Then one can check easily (exercise!) that the ensuing quotient is given by the map $c_\Omega : \mathsf{Prop} \to \Omega$ induced by the predicate $C_\Omega \in \mathcal{H}(\mathcal{A})(\mathcal{P}(\mathcal{A}) \times \mathcal{P}(\mathcal{A}))$ with $C_\Omega(p,q) \equiv p \leftrightarrow q$. However,

taking the quotient of R in $\mathbf{Asm}(\mathcal{A})$ gives rise to the map $\widetilde{q}_\Omega : \mathsf{Prop} \to \nabla(2)$ with $\widetilde{q}_\Omega(p) = 0$ for $p \neq \emptyset$ and $\widetilde{q}_\Omega(\emptyset) = 1$. Thus, the reflection of Ω in $\mathbf{RT}(\mathcal{A})$ to $\mathbf{Asm}(\mathcal{A})$ is $\nabla(2)$. This observation is used for proving the following

Lemma 5.7. *A (weak) pca $\mathcal{A}$ is trivial whenever $\mathbf{RT}(\mathcal{A})$ is boolean.*

Proof. Suppose $\mathbf{RT}(\mathcal{A})$ is boolean, i.e. $\Omega \cong 1{+}1$. Then $1{+}1 \cong \nabla(2)$ because $\nabla(2)$ is the reflection of Ω to $\mathbf{Asm}(\mathcal{A})$ and $1{+}1$ is already in $\mathbf{Asm}(\mathcal{A})$. But if $\nabla(2) \cong 1{+}1$ then $\mathsf{true} = \mathsf{false}$. Thus, for arbitrary $a, b \in \mathcal{A}$ we have $a = \mathsf{true}\, a\, b = \mathsf{false}\, a\, b = b$, i.e. $\mathcal{A}$ is trivial. □

As a consequence we get that

Corollary 5.2. *A (weak) pca $\mathcal{A}$ is trivial whenever $\Omega_{\mathbf{RT}(\mathcal{A})}$ is separated.*

Proof. Suppose $\Omega = \Omega_{\mathbf{RT}(\mathcal{A})}$ is $\neg\neg$-separated, i.e. in $\mathbf{RT}(\mathcal{A})$ it holds that $\forall u, v{\in}\Omega. \neg\neg(u{=}v) {\to} u{=}v$. Then $\forall p{\in}\Omega. \neg\neg(p{=}\top) {\to} p{=}\top$. As $(p{=}\top) \leftrightarrow p$ it follows that $\forall p{:}\Omega. \neg\neg p {\to} p$. Thus, the topos $\mathbf{RT}(\mathcal{A})$ is boolean from which it follows by Lemma 5.7 that $\mathcal{A}$ is trivial. □

We will show now that every object X of $\mathbf{RT}(\mathcal{A})$ can be covered by an epi $c_X : C_X \twoheadrightarrow X$ with C_X canonically separated. Let C_X be the assembly with $|C_X| = \{x \in |X| \mid E_X(x) \neq \bot\}$ and $||x||_{C_X} = E_X(x)$. The map c_X is given by the predicate $R_X \in \mathcal{H}(|C_X| {\times} |X|)$ with $R_X(x', x) \equiv E_X(x', x)$ which gives rise to an epi as $E_X(x) \to \exists x'{:}|C_X|. E_X(x', x)$ holds in $\mathcal{H}(\mathcal{A})$.
This fact explains why one can construct $\mathbf{RT}(\mathcal{A})$ from $\mathbf{Asm}(\mathcal{A})$ by "adding quotients" as in [CFS].
We leave it as an exercise(!) for the inclined reader to verify the following characterisation of epis and isos in $\mathbf{RT}(\mathcal{A})$.

Lemma 5.8. *Let $[F] : X \to Y$ be a morphism in $\mathbf{RT}(\mathcal{A})$. Then $[F]$ is an epi iff $E_Y(y) \to \exists x{:}|X|. F(x, y)$ holds in $\mathcal{H}(\mathcal{A})$.*
Accordingly $[F]$ is an isomorphism iff both $E_Y(y) \to \exists x{:}|X|. F(x, y)$ and $F(x, y) \wedge F(x', y) \to E_X(x, x')$ hold in $\mathcal{H}(\mathcal{A})$ (besides the conditions (strict), (cong), (singval) and (tot)*).*

Notice that arithmetic is available in $\mathbf{Asm}(\mathcal{A})$ and thus in $\mathbf{RT}(\mathcal{A})$ via the assembly N with $|N| = \mathbb{N}$ and $||n||_N = \{\underline{n}\}$ (see Def. 3.2). The category $\mathbf{Asm}(\mathcal{A})$ models higher order intuitionistic arithmetic when interpreting $P(X)$ as Prop^X. The category $\mathbf{RT}(\mathcal{A})$ models higher order arithmetic with *extensionality principle for predicates* when interpreting $P(X)$ as Ω^X.
Thus, realizability toposes provide a framework sufficiently rich for interpreting higher order (i.e. impredicative) intuitionistic mathematics. Actually, one can show that realizability toposes do even host models for Intuitionistic Zermelo Fraenkel set theory **IZF** (see [JM] and the references in there).

6 Modest Models of Polymorphism

One of the main benefits of modest sets is that they allow one to interpret so-called “polymorphic” type theories (see [St, Jac]) as e.g. the polymorphic λ-calculus (originally called “system F by its inventor Jean-Yves Girard) in a nontrivial way. This is remarkable because all its models in **Set** are bound to be trivial in the sense that all terms (of the same type) get identified in such a model.

Before describing realizability models of polymorphic type theories we show that $\mathbf{Mod}(\mathcal{A})$ constitutes a “small complete category internal to $\mathbf{Asm}(\mathcal{A})$”. To make this precise we first define what is a *family of modest sets indexed by an assembly.*

Definition 6.1. *A* family of modest sets *in* $\mathbf{Asm}(\mathcal{A})$ *(indexed by an assembly* X*) is a morphism* $a : A \to X$ *in* $\mathbf{Asm}(\mathcal{A})$ *such that for all* $x : 1 \to X$ *the object* A_x *in*

$$\begin{tikzcd} A_x \arrow[r] \arrow[d, "x^*a"'] \arrow[dr, phantom, "\lrcorner", very near start] & A \arrow[d, "a"] \\ 1 \arrow[r, "x"'] & X \end{tikzcd}$$

is modest. For $X \in \mathbf{Asm}(\mathcal{A})$ *we write* $\mathbf{Mod}(\mathcal{A})(X)$ *for the full subcategory of the slice category* $\mathbf{Asm}(\mathcal{A})/X$ *whose objects are families of modest sets indexed by* X*.* $\Diamond$

Obviously, families of modest sets are stable under pullbacks along arbitrary morphisms in $\mathbf{Asm}(\mathcal{A})$.

The following characterisation will be used tacitly in the following.

Lemma 6.1. *A morphism* $a : A \to X$ *in* $\mathbf{Asm}(\mathcal{A})$ *is a family of modest sets iff* $y_1 = y_2$ *whenever* $a(y_1) = a(y_2)$ *and* $||y_1||_Y \cap ||y_2||_Y \neq \emptyset$*.*

Proof. Straightforward exercise! $\square$

Lemma 6.2. *For every* $X \in \mathbf{Asm}(\mathcal{A})$ *the category* $\mathbf{Mod}(\mathcal{A})(X)$ *has finite limits and colimits.*

Proof. Straightforward exrecise! $\square$

Lemma 6.3. *For every* $f : Y \to X$ *in* $\mathbf{Asm}(\mathcal{A})$ *the functor* Π_f *preserves families of modest sets, i.e. whenever* $a : A \to Y$ *is a family of modest sets then* $\Pi_f a$ *is a family of modest sets as well.*

Proof. Recall the construction of Π_f from Theorem 4.4. Suppose $e \Vdash \langle x, s_1\rangle, \langle x, s_2\rangle$. Then $\mathsf{p}_1 e \Vdash s_1, s_2$. We show that then $s_1 = s_2$ and thus $\langle x, s_1\rangle = \langle x, s_2\rangle$ as desired.

Suppose $y \in f^{-1}(x)$. Let $a \Vdash_Y y$. Then from $\mathsf{p}_1 e \Vdash s_1, s_2$ it follows that $\mathsf{p}_1 ea \Vdash s_1(y), s_2(y)$ because $a(s_1(y)) = a(s_2(y))$ and a is a family of modest sets. $\square$

Lemma 6.2 and 6.3 together say that "modest sets fibred over assemblies are internally complete".[35]
Notice, however, that $\Sigma_f a$ need not be a family of modest sets even if a is. For example if $f : Y \to X$ is not a family of modest sets then $\Sigma_f id_Y = f$ is not a family of modest sets although id_Y is.
However, there exists a left adjoint $\exists_f \dashv f^* : \mathbf{Mod}(\mathcal{A})(X) \to \mathbf{Mod}(\mathcal{A})(Y)$ given by $R_X \circ \Sigma_f$ where R_X is left adjoint to the inclusion $\mathbf{Mod}(\mathcal{A})(X) \hookrightarrow \mathbf{Asm}(\mathcal{A})/X$. The construction of R_X and the verification of the Beck-Chevalley condition we leave as a (slightly nontrivial) exercise to the inclined reader.

For proving that the category of modest sets is essentially small the following observation is crucial. Every modest set $X \in \mathbf{Mod}(\mathcal{A})$ is equivalent to the modest set X_c where $|X_c| = \{||x||_X \mid x \in |X|\}$ and $||A||_{X_c} = A$, i.e. X_c is obtained from X by replacing every element $x \in |X|$ by its set $||x||_X$ of realizers. Let us call modest sets of the form X_c *canonically modest*. There is an obvious 1-1-correspondence between canonically modest sets and so-called *partial equivalence relations* on $\mathcal{A}$, i.e. symmetric and transitive binary relations on $\mathcal{A}$ (that in general are not reflexive!). If X is canonically modest then the corresponding partial equivalence relation ("per") R_X is given by aR_Xb iff $\exists x{\in}|X|.a, b \Vdash_X x$, i.e. iff a and b realize the same element in $|X|$. On the other hand for every per R on $\mathcal{A}$ the corrsponding canonically modest set A_R is given by $|A_R| = \mathcal{A}_{/R} = \{[a]_R \mid aRa\}$ where $[a]_R = \{a' \in \mathcal{A} \mid aRa'\}$ and $||A||_{A_R} = A$, i.e. an equivalence class is realized by its elements.

Lemma 6.4. *There exists a generic family of modest sets, i.e. a family γ of modest sets such that for all families a of modest sets there is a map f with $a \cong f^*\gamma$.*

Proof. Let $\mathsf{PER}(\mathcal{A})$ be the set of all partial equivalence relations on $\mathcal{A}$. Let G be the assembly with $|G| = \{\langle R, A\rangle \mid R \in \mathsf{PER}(\mathcal{A}) \text{ and } A \in \mathcal{A}_{/R}\}$ and $||\langle R, A\rangle||_G = A$. Then a generic family of modest sets is given by

$$\gamma : G \to \nabla(\mathsf{PER}(\mathcal{A})) : \langle R, A\rangle \mapsto R$$

(realized e.g. by i) : if $a : A \to X$ is a family of modest sets then $a \cong f^*\gamma$ for the map $f : X \to \nabla(\mathsf{PER}(\mathcal{A}))$ with $f(x) = \{\langle a_1, a_2\rangle \mid \exists y \in a^{-1}(x).\, a_1, a_2 \in ||y||_A\}$. ☐

This lemma together with Lemma 6.2 and 6.3 says that "modest sets form a small full internal subcategory of $\mathbf{Asm}(\mathcal{A})$ which is internally complete".[36]

We will now describe in a slightly more concrete way how $\mathbf{Mod}(\mathcal{A})$ gives rise to models of polymorphic type theories.

Lemma 6.5. *Let $f : Y \to X$ and $A : Y \to \nabla(\mathsf{PER}(\mathcal{A}))$. Then we have $\forall_f(A)^*\gamma \cong \Pi_f A^*\gamma$ where $\forall_f(A) : X \to \nabla(\mathsf{PER}(\mathcal{A}))$ is defined as follows*

$$e\,\forall_f(A)(x)\,e' \quad \textit{iff} \quad ea\,A(y)\,e'a' \textit{ for all } y \in f^{-1}(x) \textit{ and } a, a' \in ||y||_Y\ .$$

[35] See vol. 2 of [Bor], [Jac] or [St3] for a precise account of internal completeness.
[36] Again see [Jac, St3] for an explanation of these notions.

Proof. Straightforward exercise! □

As a consequence we get that universal quantification over assemblies of the form $\nabla(I)$ is given by *intersection* of per's.

Lemma 6.6. *Let $f : Y \to X$, $A : Y \to \nabla(\mathsf{PER}(\mathcal{A})))$ and $x \in |X|$ such that $a \Vdash_Y y$ forall $y \in f^{-1}(x)$ and $a \in |\mathcal{A}|$. Then (the modest set induced by the per) $\forall_f(A)(x)$ is isomorphic to (the modest set induced by the per) $\bigcap_{y\in f^{-1}(x)} A(y)$.*

Proof. By Lemma 6.5 we have $e\ \forall_f(A)(x)\ e'$ iff $ea\ A(y)\ e'a'$ for all $y \in f^{-1}(x)$ and $a, a' \in |\mathcal{A}|$, i.e. iff $ea\ \bigcap_{y\in f^{-1}(x)} A(y)\ e'a'$ for all $a, a' \in |\mathcal{A}|$. Let A_1 and A_2 be the canonically modest sets induced by $\forall_f(A)(x)$ and $\bigcap_{y\in f^{-1}(x)} A(y)$, respectively, and $\iota : A_1 \to A_2$ be the map realized by $\Lambda x.x\mathsf{i}$. Then ι is an isomorphism with ι^{-1} realized by $\Lambda x.\Lambda y.x$. □

Thus, the isomorphism of Lemma 6.6 can be chosen *uniformly in* $x \in |X|$ because its realizer does not depend on x.

For a detailed description of the interpretation of polymorphic type theories based on Lemma 6.5 and 6.6 see [St, Jac]. We just sketch here how it works for polymorphic λ-calculus (Girard's system F) as it was originally suggested by E. Moggi in 1985 (when he was still a PhD student!).
The big type (also called "kind") Tp of small system F types is interpreted by the assembly $\nabla(\mathsf{PER}(\mathcal{A}))$. Type judgements $X_1, \ldots, X_n \vdash A$ will be interpreted as morphisms $[\![A]\!] : \mathsf{Tp}^n \to \mathsf{Tp}$ where $[\![\Theta \vdash \forall X.A]\!](\vec{R}) = \bigcap_{R\in\mathsf{PER}(\mathcal{A})}[\![\Theta, X \vdash A]\!](\vec{R}, R)$. Typing judgements $X_1, \ldots, X_n \mid x_1, \ldots, x_m \vdash t : B$ will be interpreted as equivalence classes of the per

$$\bigcap_{\vec{R}\in\mathsf{PER}(\mathcal{A})^n} \big[[\![A_1]\!](\vec{R}) \times \cdots \times [\![A_n]\!](\vec{R}) \to [\![B]\!](\vec{R})\big]$$

where the operations $\times$ and $\to$ on $\mathsf{PER}(\mathcal{A})$ mimick the corresponding ones on $\mathbf{Mod}(\mathcal{A})$.
For the part of the polymorphic λ-calculus coming from simply typed λ-calculus the interpretation is like the usual interpretation of simply typed λ-calculus in ccc's (here the $\mathbf{Mod}(\mathcal{A})(\mathsf{Tp}^n)$). For $\Theta, X \mid \Gamma \vdash t : A$ with $x \notin \mathrm{FV}(\Gamma)$ we put

$$[\![\Theta \mid \Gamma \vdash \Lambda X.t : \forall X.A]\!](\vec{R})(\vec{a}) = \bigcap_{R\in\mathsf{PER}(\mathcal{A})} [\![\Theta, X \mid \Gamma \vdash t : A]\!](\vec{R}, R)(\vec{a})$$

and for $\Theta \mid \Gamma \vdash t : \forall X.A$ and $\Theta \vdash B$ we put

$$[\![\Theta \mid \Gamma \vdash t\{B\}]\!](\vec{R})(\vec{a}) = [e]_{[\![\Theta, X\vdash A]\!](\vec{R}, [\![\Theta\vdash B]\!](\vec{R}))}$$

with $e \in [\![\Theta \vdash t : \forall X.A]\!](\vec{R})(\vec{a})$.

A Elementary Recursion Theory

For the convenience of the reader we recall here the basic definitions and facts from elementary recursion theory as far as they are needed for our development of realizability. For more detailed information it might be helpful to consult Chapter 3 of [TvD] or the comprehensive book of Rogers [Ro].

Definition A.1. (partial recursive functions) *The* partial recursive functions *are the subset $\mathcal{P}$ of $\bigcup_{k\in\mathbb{N}}\big[\mathbb{N}^k \rightharpoonup \mathbb{N}\big]$ (where $\big[A \rightharpoonup B\big]$ stands for the set of partial functions from A to B) defined inductively by the following clauses*

(1) $\mathsf{zero} : \mathbb{N}^0 \to \mathbb{N} : \langle\rangle \mapsto 0$ *is in* $\mathcal{P}$.

(2) *The successor function* $\mathsf{succ} : \mathbb{N} \to \mathbb{N} : n \mapsto n+1$ *is in* $\mathcal{P}$.

(3) *For every* $n > 0$ *and* i *with* $1 \leq i \leq n$ *the projection function*

$$\pi^n_i : \mathbb{N}^n \to \mathbb{N} : (x_1, \ldots, x_n) \mapsto x_i$$

is in $\mathcal{P}$.

(4) *If* $g : \mathbb{N}^n \rightharpoonup \mathbb{N}$ *and* $h_i : \mathbb{N}^m \rightharpoonup \mathbb{N}$ *for* $i = 1, \ldots, n$ *then the function*

$$f : \mathbb{N}^m \rightharpoonup \mathbb{N} : \vec{x} \mapsto g(h_1(\vec{x}), \ldots, h_n(\vec{x}))$$

is in $\mathcal{P}$ *whenever* g *and the* h_i *are all in* $\mathcal{P}$.

(5) *If* $g : \mathbb{N}^n \rightharpoonup \mathbb{N}$ *and* $h : \mathbb{N}^{n+2} \rightharpoonup \mathbb{N}$ *are in* $\mathcal{P}$ *then the function* $f : \mathbb{N}^{n+1} \rightharpoonup \mathbb{N}$ *with*

$$f(\vec{x}, 0) \simeq g(\vec{x}) \quad \textit{and} \quad f(\vec{x}, n+1) \simeq h(\vec{x}, n, f(\vec{x}, n))$$

is in $\mathcal{P}$.

(6) *If* $f : \mathbb{N}^{k+1} \rightharpoonup \mathbb{N}$ *is in* $\mathcal{P}$ *then the function* $\mu(f) : \mathbb{N}^k \rightharpoonup \mathbb{N}$ *defined as*

$$\mu(f)(\vec{x}) \simeq \begin{cases} n & \text{if } f(\vec{x}, n) = 0 \text{ and } \forall m < n.\, f(\vec{x}, m) > 0 \\ \uparrow & \textit{otherwise} \end{cases}$$

is in $\mathcal{P}$.

We write $\mathcal{R}$ *for the set of* total recursive *functions, i.e. functions in* $\mathcal{P}$ *which are total in the sense that they are defined for all arguments.*
The functions inductively generated by clauses (1)-(5) are called primitive recursive *and we write* $\mathcal{PR}$ *for the set of all primitive recursive functions.* $\lozenge$

The most important fact about the unary partial recursive functions is that they can be gödelized in the following most pleasant way.

Theorem A.1. *There is a surjective map* ϕ *from* $\mathbb{N}$ *to the unary partial recursive functions satisfying the following conditions.*

(1) *The function*

$$u : \mathbb{N}^2 \rightharpoonup \mathbb{N} : (e, n) \mapsto \phi_e(n)$$

is partial recursive.

(2) *For every* $k \in \mathbb{N}$ *and* $k+1$*–ary partial recursive function* f *there is a* k*–ary primitive recursive function* h *such that*

$$\phi_{h(\vec{n})}(m) \simeq f(\vec{n}, m)$$

for all $\vec{n} \in \mathbb{N}^k$ *and* $m \in \mathbb{N}$.

Moreover, there is a ternary primitive recursive function T *and a unary primitive recursive function* U *such that*

$$\phi_n(m) \simeq U(\mu k.\, T(n, m, k))$$

where T *is called* Kleene's T-predicate *and* U *is called the* result extraction function. *Moreover, the predicate* T *can be chosen in such a way that* $T(n, m, k) \wedge T(n, m, k') \rightarrow k = k'$.

Proof. For details see e.g. [TvD]. We just mention the idea behind T and U. The intuitive reading of $T(n, m, k)$ is that k is a code for a (successful) computation of the algorithm with number n applied to argument m and $U(k)$ is the result of this computation. For given n and m there exists at most one (code of a) successful computation from which "single-valuedness" of T is obvious. $\square$

For reasons of tradition we write $\{n\}$ instead of ϕ_n. Whether $\{n\}$ means the n-th partial recursive function or the singleton set containing n will always be clear from the context as e.g. in $\{n\}(m)$ where $\{n\}$ means the partial function as it is applied to an argument.

The partial operation $\{\cdot\}(\cdot)$ is called *Kleene application* and will be used freely for building terms. Let e be an expression describing a partial recursive function in the free variables of e. Then by Theorem A.1(2) there exists a primitive recursive term $\Lambda x.e$ with $\{\Lambda x.e\}(n) \simeq e[n/x]$ for all $n \in \mathbb{N}$. Terms which possibly contain Kleene application will be called *partial terms*.

For partial terms t and s we write $t = s$ as an abbreviation for $\exists x. t = x \wedge s = x$ expressing that both t and s are defined and equal. We usually write $t\downarrow$ for $t = t$ saying that t is defined, i.e. t terminates. This fixes what $P(t)$ means when t is a partial term and $P(x)$ is an atomic formula. The homomorphic extension to compound predicates $A(x)$ is also denoted by $A(t)$. But notice that $A(t)$ does not in general imply $t\downarrow$ e.g. for $A(x) \equiv \neg x = x$ we have $A(t) \rightarrow t\downarrow$ iff $\neg\neg t\downarrow$. Finally notice that $t\downarrow \wedge A(t)$ is equivalent to $\exists x. t = x \wedge A(x)$.

Definition A.2. *Let* $A \subseteq \mathbb{N}$. A *is called* recursively enumerable[37] (r.e.) *iff there is a unary partial recursive function* f *such that* $n \in A$ *iff* $f(n)\downarrow$ *and* A *is called* decidable *iff there is a unary total recursive function* f *such that* $n \in A$ *iff* $f(n) = 0$. $\diamondsuit$

[37] This terminology may be surprising at first sight but it isn't as one can show that a set A of natural numbers is r.e. iff A is empty or there exists a total recursive function f with $A = \{f(n) \mid n \in \mathbb{N}\}$.

Obviously, every decidable set is also recursively enumerable but the reverse inclusion does not hold.

Theorem A.2. *The set* $K := \{n \in \mathbb{N} \mid \{n\}(n)\!\downarrow\}$ *is recursively enumerable but not decidable.*

Proof. If K were decidable then $\mathbb{N} \setminus K = \{n \in \mathbb{N} \mid \{n\}(n)\!\uparrow\}$ were recursively enumerable, i.e. there were an $e \in \mathbb{N}$ with

$$\{e\}(n)\!\downarrow \quad \Leftrightarrow \quad \{n\}(n)\!\uparrow$$

but then (putting $n = e$) it would hold that

$$\{e\}(e)\!\downarrow \quad \Leftrightarrow \quad \{e\}(e)\!\uparrow$$

which clearly is impossible. □

Consequently, the *halting set* $H := \{\langle n, m\rangle \mid \{n\}(m)\!\downarrow\}$ is not decidable as otherwise K were decidable in contradiction to Theorem A.2.
Notice that $n \notin K$ can be expressed by the arithmetic formula $\forall k.\, \neg T(n, n, k)$. Thus, no formal system can prove all true formulas of the form $\forall k.\, \neg T(n, n, k)$ since otherwise K were decidable.

Theorem A.3. *Let* $A_i = \{n \in \mathbb{N} \mid \{n\}(n){=}i\}$ *for* $i = 0, 1$. *Then there is no total recursive function* f *with* $f(n) = i$ *whenever* $n \in A_i$ *for* $i = 0, 1$.

Proof. If there were such a recursive f then there would exist a total recursive g with $g[\mathbb{N}] \subseteq \{0, 1\}$ satisfying

$$n \in A_0 \Rightarrow g(n) = 1 \qquad \text{and} \qquad n \in A_1 \Rightarrow g(n) = 0$$

for all $n \in \mathbb{N}$. Let $g = \{e\}$. Then $\{e\}(e) \in \{0, 1\}$ and, therefore, $e \in A_0 \cup A_1$. But this is impossible as if $e \in A_0$ then $0 = \{e\}(e) = g(e) = 1$ and if $e \in A_1$ then $1 = \{e\}(e) = g(e) = 0$. □

One also says that A_0 and A_1 are *recursively inseparable* as there does not exist a recursive set P such that $A_0 \subseteq P$ and $A_1 \subseteq \mathbb{N} \setminus P$.

Finally we fix some notation concerning the primitive recursive coding of finite sequences of natural numbers by natural numbers. Such an encoding can be obtained via the coding of pairs $\langle\cdot,\cdot\rangle$ and its projections fst and snd in the following way: 0 codes the empty sequence, $\langle 0, n\rangle + 1$ codes the sequence of length 1 with n as its single element and $\langle k{+}1, n\rangle{+}1$ is the code of the sequence

$$\mathrm{fst}(n), \mathrm{fst}(\mathrm{snd}(n)), \ldots, \mathrm{fst}(\mathrm{snd}^{k-1}(n)), \mathrm{snd}^{k}(n)$$

We write $\langle n_0, \ldots, n_{k-1}\rangle$ for the unique code of the sequence $n_0, \ldots, n_{k-1}$. Moreover, there exists a primitive recursive concatenation function $*$ satisfying

$$\langle s\rangle{*}\langle t\rangle = \langle s, t\rangle$$

for all $s, t \in \mathbb{N}^*$. The function lgth defined as

$$\mathrm{lgth}(\langle n_0, \ldots, n_{k-1} \rangle) = k$$

is primitive recursive. For $n = \langle m_0, \ldots, m_{k-1} \rangle$ and $i \in \mathbb{N}$ we define

$$n_i = \begin{cases} m_i & \text{if } i < k \\ 0 & \text{otherwise} \end{cases}$$

which mapping is primitive recursive.
We write $\langle s \rangle \preceq \langle t \rangle$ iff s is a prefix of t and $\langle s \rangle \prec \langle t \rangle$ iff s is a proper prefix of t. Obviously, $\preceq$ and $\prec$ are primitive recursive predicates on codes of sequences. Furthermore, for a function α from $\mathbb{N}$ to $\mathbb{N}$ we write $\overline{\alpha}(n)$ for (the code of) the finite sequence $\langle \alpha(0), \ldots, \alpha(n{-}1) \rangle$. This operation is primitive recursive in α. We write $s \preceq \alpha$ for $s = \overline{\alpha}(\mathrm{lgth}(s))$, i.e. if α has prefix s. We also write $\langle s_0, \ldots, s_{n-1} \rangle * \alpha$ for the function β from $\mathbb{N}$ to $\mathbb{N}$ with

$$\beta(k) = \begin{cases} s_k & \text{if } k < n \\ \alpha(k{-}n) & \text{otherwise.} \end{cases}$$

B Formal Systems for Intuitionistic Logic

The syntax of predicate logic employed here deviates from the usual practice in one particular aspect: instead of having negation as a basic propositional connective we introduce a propositional constant $\bot$ ("falsity") for the false proposition and introduce negation via the "macro" $\neg A \equiv A \to \bot$.
We suggest it as an informative exercise to justify the validity of the proof rules of the following definition in terms of the BHK interpretation.

Definition B.1. (Natural Deduction)
Sequents are expressions of the form $A_1, \ldots, A_n \vdash B$ where the A_i and B are formulas of predicate logic. The intended meaning is that the assumptions $A_1, \ldots, A_n$ entail *conclusion B. The valid sequences of* Intuitionistic Predicate Logic *are defined inductively via the following proof rules*

Propositional Connectives

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\,(\wedge I) \qquad \frac{\Gamma \vdash A_1 \wedge A_2}{\Gamma \vdash A_i}\,(\wedge E_i)$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\,(\to I) \qquad \frac{\Gamma \vdash A \to B \quad \Gamma \vdash A}{\Gamma \vdash B}\,(\to E)$$

$$\frac{\Gamma \vdash A_i}{\Gamma \vdash A_1 \vee A_2}\,(\vee I_i) \qquad \frac{\Gamma \vdash A \vee B \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C}\,(\vee E)$$

$$\frac{\Gamma \vdash \bot}{\Gamma \vdash C}\,(\bot E)$$

Quantifiers

$$\frac{\Gamma \vdash A(x) \quad x \notin FV(\Gamma)}{\Gamma \vdash \forall x.A(x)}\,(\forall I) \qquad \frac{\Gamma \vdash \forall x.A(x)}{\Gamma \vdash A(t)}\,(\forall E)$$

$$\frac{\Gamma \vdash A(t)}{\Gamma \vdash \exists x.A(x)}\,(\exists I) \qquad \frac{\Gamma \vdash \exists x.A(x) \quad \Gamma, A(x) \vdash C \quad x \notin FV(\Gamma, C)}{\Gamma \vdash C}\,(\exists E)$$

Structural Rules

$$\frac{}{A \vdash A}\ (\text{ax}) \qquad \frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, B, A, \Delta \vdash C}\ (\text{ex})$$

$$\frac{\Gamma \vdash C}{\Gamma, A \vdash C}\ (\text{w}) \qquad \frac{\Gamma, A, A \vdash C}{\Gamma, A \vdash C}\ (\text{c})$$

where we write $FV(A_1, \ldots, A_n)$ for the finite set of variables having an unbound occurrence in any of the formulas A_i. $\diamondsuit$

Notice that there are two elimination rules $(\wedge E_1)$ and $(\wedge E_2)$ for conjunction and two introduction rules $(\vee I_1)$ and $(\vee I_2)$ for $\vee$.
It is absolutely necessary to take the *variable conditions* seriously in rules $(\forall I)$ and $(\exists E)$ as otherwise one could derive obviously wrong sequents (like e.g. $\exists x.A(x) \vdash \forall x.A(x)$).

Although Natural Deduction is very close to the actual practice of mathematical proofs it is sometimes useful to have available an inductive characterisation of the set of all formulas A for which $\vdash A$ is derivable in Natural Deduction. Such an inductive characterisation of valid formulas is usually called a *Hilbert Style* axiomatization of logic.

Theorem B.1. *The set of all formulas A of predicate logic for which the sequent $\vdash A$ is derivable in the calculus of Natural Deduction is defined inductively by the following rules*

(L1) $A \to A$

(L2) $A \;,\; A \to B \;\Rightarrow\; B$

(L3) $A \to B \;,\; B \to C \;\Rightarrow\; A \to C$

(L4) $A \wedge B \to A \;,\; A \wedge B \to B$

(L5) $C \to A \;,\; C \to B \Rightarrow C \to A \wedge B$

(L6) $A \to A \vee B \;,\; B \to A \vee B$

(L7) $A \to C \;,\; B \to C \Rightarrow A \vee B \to C$

(L8) $A \wedge B \to C \;\Rightarrow\; A \to B \to C$

(L9) $A \to B \to C \;\Rightarrow\; A \wedge B \to C$

(L10) $\perp \to A$

(L11) $B \to A(x) \;\Rightarrow\; B \to \forall x.A(x) \quad (x \notin FV(B))$

(L12) $\forall x.A \to A(t)$

(L13) $A(t) \to \exists x.A$

(L14) $A(x) \to B \Rightarrow \exists x.A(x) \to B \quad (x \notin FV(B))$.

Proof. One easily shows that if A can be derived via the rules (L1)–(L14) then $\vdash A$ can be proved by Natural Deduction.
For the reverse direction one shows that if $A_1, \ldots, A_n \vdash B$ can be derived in the calculus of natural deduction then the formula $A_1 \to \ldots \to A_n \to B$ is derivable via the rules (L1)-(L14). $\square$

C Alternative Proof that $\mathbf{Asm}(\mathcal{A})$ is Regular

As shown in [Bor](vol.2) in a regular category $\mathcal{E}$ a morphism e is a regular epi iff e is a *cover*, i.e. every mono through which e factors is an isomorphism. This suggests that a category $\mathcal{E}$ is regular iff it validates the following conditions

(1) $\mathcal{E}$ has finite limits

(2) every morphism of $\mathcal{E}$ factors as a cover followed by a mono

(3) covers are stable under pullbacks along arbitrary morphisms in $\mathcal{E}$

and actually as shown in A.1.3.4 of [Joh] a category validating these conditions is regular since all covers are equalizers of their kernel pairs. (It is an easy exercise to show that in regular categories regular epis are covers.)
Let $f : X \to Y$ be a morphism in $\mathbf{Asm}$)($\mathcal{A}$). Then it factors as $f = m_f \circ e_f$ where $e_f : X \to I_f$ is the coequalizer of the kernel pair of f. Recall that $|I_f| = f[|X|]$, $a \Vdash_{I_f} y$ iff $a \Vdash_X x$ for some $x \in |X|$ with $e(x) = y$. Moreover, we have $e_f(x) = f(x)$ and $\mathsf{i} \Vdash e_f$. The map m_f sends $y \in |I_f|$ to y and it is realized by any realizer of f.

Lemma C.1. *In* $\mathbf{Asm}(\mathcal{A})$ *for a morphism* $f : X \to Y$ *the following conditions are equivalent*

(1) f *is a cover*

(2) f *is the coequalizer of its kernel pair*

(3) *the map* $f : |X| \to |Y|$ *is onto and there is an* $e \in \mathcal{A}$ *such that for every* $y \in |Y|$ *and* $a \Vdash_Y y$ *there is an* $x \in f^{-1}(y)$ *with* $ea \Vdash_X x$.

Proof. Suppose f is a cover. Then m_f is an isomorphism and thus f is a coequalizer of its kernel pair.
Suppose f is the coequalizer of its kernel pair. Then m_f is an isomorphism. Then there exists $e \in \mathcal{A}$ such that for any $y \in |Y|$ and $a \Vdash_Y y$ we have $ea \Vdash m_f^{-1}(y)$, i.e. $ea \vdash_X x$ for some $x \in f^{-1}(y)$.
Suppose f validates condition (3). Suppose $f = m \circ g$ with $m : Z \to Y$ monic. Since $f : |X| \to |Y|$ is onto the map $m : |Z| \to |Y|$ is a bijection. Let $e \in \mathcal{A}$ such that for every $y \in |Y|$ and $a \Vdash_Y y$ there is an $x \in f^{-1}(y)$ with $ea \Vdash_X x$. Moreover, let $\tilde{e}$ be a realizer for g. Then the inverse of m is realized by $\Lambda a.\tilde{e}(ea)$. ☐

Theorem C.1. *The category* $\mathbf{Asm}(\mathcal{A})$ *is regular.*

Proof. It is already known that $\mathbf{Asm}(\mathcal{A})$ has finite limits.
Every morphism f in $\mathbf{Asm}(\mathcal{A})$ factors as $f = m_f e_f$ where m_f is monic and e_f is the coequalizer of the kernel pair of f. Obviously, the map e_f validates condition (3) of Lemma C.1 and thus is a cover.
One easily checks that morphisms validating condition (3) of Lemma C.1 are stable under pullbacks along arbitrary morphisms in $\mathbf{Asm}(\mathcal{A})$. For this reason covers are stable under pullbacks along arbitrary morphisms in $\mathbf{Asm}(\mathcal{E})$. ☐

One easily checks that for $f : X \to Y$ in $\mathbf{Asm}(\mathcal{A})$ the object I_f is modest whenever$X$ is modest. From this it follows that Lemma refreglm holds also for $\mathbf{Mod}(\mathcal{A})$ which allows us to prove

Theorem C.2. *The category* $\mathbf{Mod}(\mathcal{A})$ *is regular.*

in a way amost identical with the proof of Th. C.1.

References

[Bar] J. Barwise (ed.) *Handbook of Mathematical Logic* North Holland, 1977.

[Bau] A. Bauer *The Realizability Approach to Computable Analysis and Topology* PhD Thesis, Carnegie-Mellon Univ. 2000.

[BiBr] E. Bishop, D. Bridges *Constructive Analysis* Grundlehren der mathematischen Wissenschaften 279, Springer, 1985.

[Bor] F. Borceux *Handbook of Categorical Algebra* 3 vols., Cambridge Univ. Press (1994).

[Bu] S. Buss (ed.) *Handbook of Proof Theory* Elsevier 1998.

[CFS] A. Carboni, P. J. Freyd, A. Scedrov *A categorical approach to realizability and polymorphic types* in Springer Lecture Notes in Comput. Sci., 298, pp.23–42 (1987).

[Hyl] M. Hyland *The effective topos* in Proc. of *The L.E.J. Brouwer Centenary Symposium* pp.165–216, North-Holland, 1982.

[HJP] M. Hyland, P. Johnstone, A. Pitts *Tripos Theory* Math. Proc. Cambridge Philos. Soc. 88, no. 2, pp.205–231, 1980.

[HS] J. R. Hindley, J. P. Seldin *Introduction to combinatory logic and λ-calculus* Cambridge University Press, UK 1986.

[Jac] B. Jacobs *Categorical Logic and Type Theory* North Holland (1999).

[Joh] P. T. Johnstone *Sketches of an Elephant. A Topos Theory Compendium.* 2 vols. OUP (2002).

[JM] A. Joyal, I. Moerdijk *Algebraic Set Theory* CUP (1995).

[KV] S. C. Kleene, R. Vesley *The Foundations of Intuitionistic Mathematics* North Holland, 1965.

[Lon] J. Longley *Realizability Toposes and Language Semantics.* PhD Thesis, Univ. Edinburgh 1994.

[Ro] H. Rogers jr. *Theory of recursive functions and effective computability.* 2^{nd} edition, MIT Press, Cambridge, MA, 1987.

[Roh] A. Rohr *A Universal Realizability Model for Sequential Computation.* PhD Thesis, TU Darmstadt (2002) electronically available from `www.mathematik.tu-darmstadt.de/~streicher/THESES/rohr.ps.gz`

[Sc80] Dana S. Scott *Relating theories of the λ-calculus* in *To H. B. Curry: essays on combinatory logic, lambda calculus and formalism* pp. 403-450 Academic Press, London-New York (1980).

[St] T. Streicher *Semantics of Type Theory* Birkhäuser, 1991.

[St1] T. Streicher *Introduction to Constructive Logic and Mathematics.* Lecture notes, 2001. electronically available at `www.mathematik.tu-darmstadt.de/~streicher/CLM/clm.pdf`

[St2] T. Streicher *Introduction to Category Theory and Categorical Logic.* Lecture notes, 2003. electronically available at `www.mathematik.tu-darmstadt.de/~streicher/CTCL.pdf`

[St3] T. Streicher *Fibred Categories à la Bénabou.* Lecture notes, 2003-2017. electronically available at `www.mathematik.tu-darmstadt.de/~streicher/FIBR/FibLec.pdf`

[St4] T. Streicher *Domain-theoretic Foundations of Functional Programming.* Imperial College Press, 2007.

[Tr73] A. Troelstra (ed.) *Metamathematical Investigations of Intuitionistic Arithmetic and Analysis* SLNM 344, Springer Verlag, 1973.

[Tr77] A. Troelstra *Aspects of Constructive Mathematics* pp. 973-1052 of [Bar].

[TvD] A. Troelstra, D. vanDalen *Constructivism in Mathematics* 2 vol.'s, North Holland, 1988.

[Tr98] A. Troelstra *Realizability* pp. 407-473 of [Bu].

[vOo] J. van Oosten *Realizability. An Introduction to its Categorical Side.* Elsevier (2008).