Realizability and Kripke Forcing*

James Lipton**

TR 90-1163
October 1990

Department of Computer Science
Cornell University
Ithaca, NY 14853-7501

*An abbreviated version is to appear in *The Annals of Mathematics and Artificial Intelligence, Vol. 4, North-Holland.*
**This work was partially supported by a Sloan dissertation grant, a fellowship from the Mathematical Sciences Institute, Cornell University, and a postdoctoral fellowship from the Dept. of Computer Science, Cornell University.

Realizability and Kripke Forcing*

James Lipton†
Dept. of Computer Science
Cornell University ‡

Abstract

Realizability, developed by Stephen Kleene, is a type-free device for extracting computations from logical specifications. Realizability analyzes the computational content of reasoning: it models the *universe of recursive mathematics*. Kripke and associated Categorical interpretations give a broader, topological/algebraic semantics for constructive reasoning which is complete for intuitionistic logic. They are, therefore, a powerful and indispensable tool for modelling computationally meaningful formal systems. How are the two semantical paradigms related ? In this paper, we construct several Kripke and Categorical Models which are elementarily equivalent to Syntactic Realizability. By merging the two approaches we provide a new class of models and a framework for reasoning about computational evidence and the process of term extraction itself.

1 Introduction

To what extent can we compute with logical tools? The question has become of central importance in dealing with at least two major challenges in computer science: the problem of producing correct software, and that of producing programs that capture some kind of reasoning process in a given domain. Work in realizability ([39, 38]), automated deduction ([78, 41, 54]), and type theory ([12, 11, 35, 59, 9, 10]) in logic and computer science has led to major advances in this area. Some of these advances include *logic programming*([54]), which is a form of programming directly with *executable specifications*, and various paradigms for *term extraction* or automatically synthesizing code from *proofs of functionality of specifications.* What this means is that from a proof in certain constructive formal systems, of some assertion of the form

$$\forall x \exists y A(x, y) \tag{1}$$

*an abbreviated version is to appear in *the Annals of Mathematics and Artificial Intelligence, Vol.4, North-Holland*

†This work was partially supported by a Sloan dissertation grant, a fellowship from the Math Sciences Institute, Cornell, and a postdoctoral fellowship from the Department of Computer Science, Cornell

‡Dept. of Mathematics, University of Pennsylvania after Sept. 1990

one is able to automatically extract a program for a function f which meets the specification A in the sense that

$$\forall x A(x, f(x)) \tag{2}$$

is provable. Most approaches to term extraction are based on the use of constructive reasoning, from which it is at least plausible to extract explicit witnesses and algorithms. For background in constructive logic we refer the reader to [89, 92, 1]. A large body of work going back over half a century ([70, 1, 60]) shows that one cannot hope in general to extract such information from classical reasoning [1] An important program of research has been undertaken to reformulate the traditional mathematical disciplines in a computationally oriented manner, for which the reader is referred to [4, 6, 1] and the bibliographies in these sources.

The question of which constructive formalism to choose, and how to make precise and *explicit* the machinery for extracting computations has given rise to almost as many solutions as there are programming languages. Broadly speaking, we may consider two paradigms, which are active areas of research in logic and computer science. [2] The *propositions-as-types* paradigm ([9, 58, 59, 11, 2]) is based on the identification of logical formulae with types or sets of proofs. Proofs are *explicitly* brought into the reasoning process via rules of inference of the form, e.g.,

$$\frac{x : A \qquad t : A \to B}{(tx) : B} \tag{3}$$

which is a reformulation of the law of *modus ponens*: **from** A and $A \to B$ **infer** B in type-theoretic terms. The rule (3) asserts that a proof t of $A \to B$ *applied* to a proof x of A gives a proof tx of B. In fact it can even be viewed as a type-inference rule of the form: if x is a term of type A, and t a term (function) from A to B then (tx) is a term of type B. This straightforward identification of (constructive) proofs and computations has remarkable consequences for computer science. It underlies the so-called *Curry-Howard isomorphism*, ([34, 11, 9, 58, 59]), and it is the basis for explicit association of computational evidence with constructive proofs that is built in to such systems as Nuprl [9] and the Theory of Constructions (Coquand and Huet [12]). This correspondence has led to the development of new programming languages (ML [31], Miranda[91]), a syntactic proof of the consistency of second order arithmetic [26], as well as fundamental insights into the algebraic structure of proof theory ([87, 46]).

A similar approach, however, is implicit in much older work, introduced nearly half a century ago by Stephen Kleene [39, 38], called *recursive realizability*. It is type-free, and does not explicitly attach computations to the logic itself, but rather introduces them "after the fact" as *computational evidence* for what is being asserted. These computations, or "realizers" are not proofs. They represent an even stronger commitment to computational

[1] although for the special case of A *quantifier-free* in 1, H. Friedman ([21]) has shown that computations can be extracted from classical proofs. C. Murthy's dissertation [64] works out the computational significance and the details of automating this extraction.

[2] They are not the only approaches, however. Other have been considered by e.g. Gödel [27], and G. Kreisel [43]

evidence than is to be found in "pure" intuitionistic logic. They define a *semantics* for intuitionistic reasoning in which such ultra-constructive principles as "all functions from $\mathbf{N}$ to $\mathbf{N}$ are recursive" are satisfied (and, hence, shown consistent).

The realizability "Universe" Realizability can be thought of as an interpretation of intuitionistic reasoning in which "constructive" is identified with "computable". The basic idea is to associate, to each formula φ in a formal system (e.g. first-order arithmetic), a code e for an algorithm which "ratifies" or *realizes* the assertion φ, in symbols:

$$e \mathrel{\underset{\sim}{\mathbf{r}}} \varphi.$$

There are various ways to formalize such a relationship between codes and logical assertions. We will be studying the so-called syntactic version of Kleene's 1952 realizability, partly because it is one of the most constructive. First, a word about notation. We adopt the convention

$$\begin{aligned} e_0 &:= \textit{left projection of the number } e \\ e_1 &:= \textit{right projection of the number } e \end{aligned}$$

under the standard pairing and unpairing function (written out in section 2). The notation $\{e\}(x)$ is standard Kleene-bracket notation for "the e^{th} program applied to input x", while $\{e\}(x)\downarrow$ just asserts that the e^{th} program halts on input x. Now the definition of relizability. $e \mathrel{\underset{\sim}{\mathbf{r}}} \varphi$ is defined to be shorthand for a formula in first order arithmetic, according to the inductive definition below.

Notice that in the atomic case, $e \mathrel{\underset{\sim}{\mathbf{r}}} \varphi$ is just the statement φ itself, which attempts to capture the idea that "all computations realize statements like $2+2=4$, and nothing realizes statements like $2+2=5$." [3]

$$e \mathrel{\underset{\sim}{\mathbf{r}}} \varphi \quad \stackrel{\text{def}}{\equiv} \quad \varphi \qquad \text{if } \varphi \text{ is atomic} \tag{4}$$

$$e \mathrel{\underset{\sim}{\mathbf{r}}} (\varphi \,\&\, \psi) \quad \stackrel{\text{def}}{\equiv} \quad e_0 \mathrel{\underset{\sim}{\mathbf{r}}} \varphi \,\&\, e_1 \mathrel{\underset{\sim}{\mathbf{r}}} \psi \tag{5}$$

$$e \mathrel{\underset{\sim}{\mathbf{r}}} (\varphi \vee \psi) \quad \stackrel{\text{def}}{\equiv} \quad (e_0 = 0 \rightarrow e_1 \mathrel{\underset{\sim}{\mathbf{r}}} \varphi) \,\&\, (e_0 \neq 0 \rightarrow e_1 \mathrel{\underset{\sim}{\mathbf{r}}} \psi) \tag{6}$$

$$e \mathrel{\underset{\sim}{\mathbf{r}}} (\varphi \rightarrow \psi) \quad \stackrel{\text{def}}{\equiv} \quad \forall f (f \mathrel{\underset{\sim}{\mathbf{r}}} \varphi \rightarrow \{e\}(f)\downarrow \,\&\, \{e\}(f) \mathrel{\underset{\sim}{\mathbf{r}}} \psi) \tag{7}$$

$$e \mathrel{\underset{\sim}{\mathbf{r}}} \exists x \varphi(x) \quad \stackrel{\text{def}}{\equiv} \quad e_0 \mathrel{\underset{\sim}{\mathbf{r}}} \varphi(e_1) \tag{8}$$

$$e \mathrel{\underset{\sim}{\mathbf{r}}} \forall x \varphi(x) \quad \stackrel{\text{def}}{\equiv} \quad \forall n \in \mathbf{N}(\{e\}(n)\downarrow \,\&\, \{e\}(n) \mathrel{\underset{\sim}{\mathbf{r}}} \varphi(n)) \tag{9}$$

Definitions (4) and (6) implement in a computational manner Heyting's view ([33]) of the constructive meaning of implication as "a construction that builds evidence/proofs of the

[3] One could take an even more computational point of view (e.g., Martin-Löf [59]): namely $2+2=4$ is realized by the algorithm that carries out the addition on the left hand side and compares the answer to the right-hand side.

consequent from evidence for the antecedent", and of $\forall x\varphi(x)$ as a construction that builds for every individual n evidence for $\varphi(n)$. Definitions (3) and (4) underscore the constructive nature of the reasoning: evidence for a disjunction includes explicit mention of which disjunct is being established, evidence for an existential statement must include an explicit witness.

The "Realizability interpretation" of constructive arithmetic is taken as the set of properties which are realized by some computation. It is not hard to show ([89]) that such constructive principles as the intuitionistic *Church's Thesis*:

all total functions from $\mathbf{N}$ to $\mathbf{N}$ are recursive

are true in the Realizability interpretation. The interpretation sketched above admits a variety of extensions to anaysis and set theory, as well as a number of variants, many proposed by Kleene himself in subsequent work. Many are of great interest in Computer Science. *Kreisel-Troelstra realizability* for analysis [42] and *McCarty realizability* for constructive set theory [56] define realizability interpretations strong enough to model all data types in computer science as well as computational algebra and analysis. A variant of these realizabilities, known as **q**-*realizability* provides a uniform means of extracting computations from a constructive proof of a specification in analysis or set theory. For example, a realization e of

$$\forall x \exists y (x \in \mathbf{N} \,\&\, x \geq 0 \rightarrow x^2 = y)$$

would be a (necessarily correct) algorithm for computing square roots. A realization in set theory, e of

$$(\forall f)(\forall M)(\exists g) \quad (M \subseteq \mathbf{N} \quad \& \, finite(M) \,\&\, (f : M \rightarrow \mathbf{N}) \Rightarrow \\ (g : M \rightarrow \mathbf{N}) \,\&\, range(f) = range(g) \,\&\, (\forall n \in M)(g(n) \leq g(n+1)))$$

would be an algorithm that sorts any finite list of integers. The idea here is to develop correct programs by proving functionality of specifications in a constructive theory. Several major systems for algorithm development now exist based on typed or untyped variants of these ideas, notably Hayashi's **PX** [32], Constable's NUPRL, and an implementation at INRIA, in France, based on Coquand and Huet's *Theory of Constructions* [12]. Practical applications of these ideas now include automated hardware verification, and even extraction of automated deduction algorithms from constructive proofs [8].

Kripke's semantics for intuitionistic logic A general model theory for intuitionistic logic was first proposed by Tarski and McKinsey in the 1940's (see, e.g. [92, 88]), based on interpreting atomic formulas A as *open sets* $[\![A]\!]$ *in a topological space.* Using the operations of $\bigcup$, $\cap$ and $()^o$ (Interior), one is able to give a complete semantics for pure intuitionistic logic. The intuitionistic nature of this semantics arises from the fact that, in a topological space X, where $\neg A$ is modelled with the *interior* of the complement of the open set $[\![A]\!]$, the interpretation of $A \vee \neg A$, namely, $[\![A]\!] \cup [\![\neg A]\!]$ is not necessarily "truth" (= the whole of X). In fact, *any topological space* equipped with an atomic assignment function gives rise to a "logic" extending intuitionistic logic (see [75]). Topologies play the same role here that Boolean Algebras do for classical logic. This approach to semantics was streamlined and extended by Beth, Rasiowa-Sikorski [74], and Saul Kripke in the early 1960s, and generalized independently by Boileau and Joyal in the 1970s based on earlier, independent work by the Category Theorists Lawvere, Tierney and others. Kripke models, which will be defined in the next section, have proven to be a powerful and simple formalism for building counterexamples and studying the metamathematical properties of intuitionistic formal systems. They have come to play an important role in theoretical computer science ([61, 72]). Categorical models, especially Toposes [46, 57], provide an extremely rich generalization of the Kripke, Beth and Tarski paradigms. They have also become increasingly important in computer applications, and were used to provide the first viable semantical analysis of the programming language ML [35].

Relating the two semantics The work in this paper arose from an interest in studying the connections between realizability and Kripke semantics. The aim is to bring together the different strengths of the two approaches, by seeing them in a common framework. Some of the payoff sought here is to explore model theory tailored to particularly computational constructive formal systems, such as the Martin-Löf theory [59], where term extraction has been built-in, as well as to develop new ways of building counterexamples.

Work to relate the two paradigms has been undertaken by a number of researchers since the early 1970s. Martin Hyland, in 1982, gave a Topos model for *semantic* Kreisel-Troelstra realizability [42], extended to Constructive Set Theory, based on work by Hyland, Johnstone and Pitts in 1980. This model, known as the *Effective Topos* has since been connected to models developed by Mulry [63] and Ershov, (see [79]). It also played a crucial role in uncovering the so-called *Category of Modest Sets* ([55, 36]) and its relation to polymorphic programming languages. In addition to this work, a series of models combining realizability and Kripke forcing were developed by Goodman (e.g. in [28]). The realizability for Constructive Set Theory developed by McCarty in [56] can be viewed as a bit of a fusion between the two notions in that realizability is developed in the style of unramified Cohen forcing.

In the other direction, (the direction of capturing Kripke style semantics through a generalization of the idea of realizability) H. Läuchli defined an *abstract realizability* capable of subsuming Kripke semantics [47], using inherently classical methods. His work was studied and refined by workers in the field of *Logical Relations*, especially Statman, Plotkin, Mitchell, Moggi, Byerly, O'Donnell, as well as by Harvey Friedman (see [61] for references). A constructive version of Läuchli's result was developed in the author's dissertation (see [50, 51]).

In this paper, we obtain a Kripke-style characterization of the syntactic realizability defined above, and a number of variants (e.g. Beeson's EON-realizabilities [1]), and category-theoretic generalizations. We also obtain a new class of models of **HA** and of the partial applicative theories proposed by Beeson, Troelstra and Feferman (see [89, 1, 90]). Our work can be seen as a provability analogue of Hyland's results. Our models are constructive:

recursively presented in the sense of Gabbay [23]. Although we have not needed to spell out a particular framework for our metatheory, the reader should note that our arguments are fully constructive.

2 A model for formalized 1952 realizability

We begin now with a simple construction which illustrates the translation mediating between realizability, and Kripke models. The same idea will be used in slightly different guise in section three.

We will describe a Beth style model which is elementarily equivalent to Kleene's realizability for arithmetic, formalized in intuitionistic, or *Heyting* arithmetic, HA.

Let $\mathcal{L}$ be the language of arithmetic, together with a denumerable set $\underline{C} = \{c_i \mid i \in \omega\}$ of fresh constants. We will need to define a certain category $\mathcal{C}_0$, sometimes denoted $\mathcal{C}_0(\underline{C})$ to stress dependency on the set of constants added to the language.

Definition 2.1 *Let $\mathcal{C}_0$ be the category with*

objects: formulas $A(x)$ in one free variable over the language $\mathcal{L}$.

morphisms: triples $(A, e, B) \in Hom_{\mathcal{C}_0}(A, B)$, where e is a natural number (and not a member of $\underline{C}$) such that

$$\mathbf{HA} \vdash (\forall x)[A(x) \to (\exists u, v)(T(e, x, u) \,\&\, U(u) = v \,\&\, B(v)] \tag{10}$$

i.e., e is the index of a partial recursive function with $A \subseteq dom(\{e\})$ and $\{e\} : A \to B$, provably in HA.

We will often simply write

$$e : A \to B$$

to indicate that(10) holds. We will also express this informally by:

$$\mathbf{HA} \vdash \forall x[A(x) \to \{e\}(x) \downarrow \,\&\, B(\{e\}(x))]. \tag{11}$$

Note that although we allow parameters from $\underline{C}$ in objects, we do not allow them in morphisms (unlike the *polynomial categories* of Lambek-Scott [46]).

Observe that $Hom_{\mathcal{C}_0}(A, B)$ is recursively enumerable, as is the full set

$$\{(\ulcorner A \urcorner, e, \ulcorner B \urcorner) \mid \ e : A \ \to \ B\}$$

of morphisms of $\mathcal{C}_0$, or the set of morphisms out of, or into a particular object, since, in all cases, we generate the theorems of HA, enumerating the relevant e's or $\ulcorner B \urcorner s$ as they turn up.

We define *realizability formulas* in one free variable $|\varphi|(e)$ associated to each arithmetic *sentence* φ over the language $\mathcal{L}$:

$$
\begin{array}{lll}
|\varphi|(e) & \text{is} & \varphi \quad \textit{for atomic } \varphi \\
|\neg\varphi|(e) & \text{is} & \forall f \neg |\varphi|(f) \\
|\varphi \,\&\, \psi|(e) & \text{is} & |\varphi|(e_0) \,\&\, |\psi|(e_1) \\
|\varphi \vee \psi|(e) & \text{is} & [e_0 = 0 \,\&\, |\varphi|(e_1)] \vee [e_0 = 1 \,\&\, |\psi|(e_1)] \\
|\varphi \rightarrow \psi|(e) & \text{is} & (\forall f)[|\varphi|(f) \rightarrow \exists uv T(e, f, u) \,\&\, U(u) = v \,\&\, |\psi(v)|] \\
|\exists x\ \varphi(x)|(e) & \text{is} & |\varphi(e_1)|(e_0) \\
|\forall x\ \varphi(x)|(e) & \text{is} & \forall n[\exists uv(T(e, n, u) \,\&\, U(u) = v \,\&\, |\varphi(n)|(v)]
\end{array}
$$

where $e = \langle e_0, e_1 \rangle$ via the standard primitive recursive pairing and unpairing.
Formally, any assertion of the form e.g. $A(e_0)$ is given by $(\exists uv)$ $(\mathrm{Pair}(u, v, e) \,\&\, A(u))$, where $\mathrm{Pair}(u, v, e)$ is $(u+v)(u+v+1) = 2(e-v)$, the standard diagonal coding predicate. Alternatively, we can (conservatively) make a definitional extension of **HA**, introducing the function symbols $(\ )_0$, $(\ )_1$ and $\langle , \rangle$ for unpairing and pairing. We follow the latter convention informally and leave the formalization to the reader's taste. We will adopt the following notation, for formulas A, B in one free variable :

$$
\begin{array}{rcl}
(A \times B)(x) & \equiv & A(x_0) \,\&\, B(x_1) \\
(A + B)(x) & \equiv & (x_0 = 0 \,\&\, A(x_1)) \vee (x_0 = 1 \,\&\, B(x_1)) \\
(A \Rightarrow B)(x) & \equiv & \forall n[A(n) \rightarrow \exists uv(T(x, n, u) \,\&\, U(u) = v \,\&\, B(v)]
\end{array}
$$

For formulas $S(x, y)$ in two free variables (over $\mathcal{L}$) we define

$$
\begin{array}{rcl}
(\sum_x S)(z) & \equiv & S(z_0, z_1) \\
(\prod_x S)(z) & \equiv & \forall x[\exists uv(T(z, x, u) \,\&\, U(u) = v \,\&\, S(x, v)]
\end{array}
$$

which formalizes "z is a choice function on x in the collection $\{S(x, y)\}$".
With this notation, we can abbreviate the definitions of realizability formulas:

$$
\begin{array}{rcl}
|\varphi \,\&\, \psi| & \equiv & |\varphi| \times |\psi| \\
|\varphi \vee \psi| & \equiv & |\varphi| + |\psi| \\
|\varphi \rightarrow \psi| & \equiv & |\varphi| \Rightarrow |\psi| \\
|\exists x \varphi| & \equiv & (\sum_x |\varphi(x)|(y)) \\
|\forall x \varphi| & \equiv & (\prod_x |\varphi(x)|(y)).
\end{array}
$$

We adopt Kleene's convention of letting $\Lambda x.u$ be the code for the function $\lambda x.u$, when that function has been shown to be partial recursive. Define two morphisms (A, e, B) and (A, f, B) in $Hom_{\mathcal{C}_0}(A, B)$ to be equal iff

HA $\vdash \forall u, v, s, t, x[A(x) \,\&\, T(e, x, u) \,\&\, U(u) = v \,\&\, T(f, x, s) \,\&\, U(s) = t \rightarrow v = t]$.

Lemma 2.2 *$\mathcal{C}_0$ is weakly bi-Cartesian closed, i.e.*
i) $A \times B$ is the product of A and B
ii) $A + B$ is the coproduct of A and B
iii) $(A \Rightarrow B)$ is the weak exponential determined by A, B

that is to say, given objects A, B and C and a morphism $f \in Hom_{\mathcal{C}_0}(C \times A, B)$, there is a not necessarily unique map $\Lambda f \in Hom_{\mathcal{C}_0}(C, A \Rightarrow B)$ making the diagram

$$\begin{tikzcd}
(A \Rightarrow B) \times A \arrow[dr, "app"] & \\
& B \\
C \times A \arrow[uu, "\Lambda^* f \times 1_A"] \arrow[ur, "f"'] &
\end{tikzcd}$$

commute, i.e., $app \cdot (\Lambda f \times 1_A) = f$ provably in HA.

In $\mathcal{C}_0$, *app* will always be the universal code $\Lambda e.\Lambda x.\{e\}(x)$, whose existence is guaranteed by the enumeration theorem. In the proof below, we make use of the fact that the formalized version of the enumeration and *s-m-n* theorems are provable in HA, which is shown in, e.g. [89].

Proof: Let π_0, π_1, i_0, i_1, be (codes for) the canonical maps

$$\pi_0 : A \times B \to A, \qquad \pi_1 : A \times B \to B,$$

$$i_0 (= \Lambda x.\langle 0, x\rangle) : A \to A + B, \qquad i_1 : B \to \ A + B$$

and now suppose $e : C \to A$ and $f : C \to B$ are morphisms in $\mathcal{C}_0$. Define $\ll e, f \gg$ to be the code for the canonical map $\lambda x.\langle \{e\}(x), \{f\}(x)\rangle$. Then $\ll e, f \gg : C \to A \times B$ and $\pi_0 \cdot \ll e, f \gg = e$, $\pi_1 \cdot \ll e, f \gg = f$, provably in HA. Uniqueness *of* $\ll e, f \gg$ is easily verified. Thus, $A \times B$, with the associated maps $\pi_a = (A \times B, \pi_0, A)$, and $\pi_b = (A \times B, \pi_1, B)$, is a product in $\mathcal{C}_0$.

Now let $e : A \to C$ and $f : B \to C$ be given in $\mathcal{C}_0$. Letting $[\![e, f]\!]$ be the code for the map $\lambda x.\textbf{if } \pi_0 x = 0 \textbf{ then } \{e\}(\pi_1 x) \textbf{ else } \{f\}(\pi_1 x)$ in $Hom_{\mathcal{C}_0}(A + B, C)$, we have $[\![e, f]\!] \cdot i_0 = e$ and $[\![e, f]\!] \cdot i_1 = f$, provably in HA. Uniqueness of $[\![e, f]\!]$ is easily checked.

Finally, assume $f : C \times A \to B$ is given in $\mathcal{C}_0$. By the $s - m - n$ theorem, there is a code $\Lambda^* f$ for the mapping $c \mapsto \Lambda x.\{f\}(\langle x, c\rangle)$ in $Hom_{\mathcal{C}_0}(C,\ A \Rightarrow B)$. The map $\Lambda^* f \times 1_A$, given by $\langle c, a\rangle \mapsto \langle (\Lambda^* f)c, a\rangle$ clearly makes the "cartesian" diagram above commute, provably in HA. It is not, however, provably unique: in fact there are infinitely many codes which satisfy this extensional requirement. Hence, $\mathcal{C}_0$ is only "weakly bi-Cartesian closed". ∎

Making the category "strongly" Cartesian A standard construction for "correcting" this [4] is to work in the category $\mathcal{C}_0^{ER}$ of *pairs* $\langle A, E\rangle$ where A is an object of $\mathcal{C}_0$ and E is a *formal equivalence relation* on A, i.e., in our case, a formula in two free variables $E(x, y)$ satisfying:

$$\begin{aligned}\mathbf{HA} \ \vdash \ & \forall x, y, z(A(x) \,\&\, A(y) \,\&\, A(z) \to E(x,x) \,\&\\ & (E(x,y) \to E(y,x)) \,\&\, (E(x,y) \,\&\, E(y,z) \to E(x,z))).\end{aligned}$$

Morphisms in this category, are triples $< \langle A, E\rangle, e, \langle B, F\rangle >$, where, provably in **HA**, e is a code for a recursive function from A to B, as in $\mathcal{C}_0$, but satisfying the additional requirement of preservation of equivalence relations. This means that, formally in **HA**, if $A(x) \,\&\, A(y) \,\&\, E(x,y)$ then the images of x and y in B under $\{e\}$ are F-equivalent. One has to show, of course, that a natural product, coproduct and exponent can be defined in $\mathcal{C}_0^{ER}$. For this purpose, let E,F be equivalence relations on A and B respectively. Define $E \otimes F(\langle x, y\rangle, \langle x\prime, y\prime\rangle)$ to be the equivalence relation on $A \times B$ given by $E(x, x\prime) \,\&\, F(y, y\prime)$. Similarly, we now define $E \oplus F$ on $A + B$ and $(E \leadsto F)$ on $A \Rightarrow B$ by:

$$\begin{aligned} E \oplus F(x,y) & \stackrel{\text{def}}{\equiv} & x_0 = y_0 \,\&\, (x_0 = 0 \to E(x_1, y_1)) \\ & & \&\, (x_0 = 1 \to F(x_1, y_1)) \\ (E \leadsto F)(f, g) & \stackrel{\text{def}}{\equiv} & \forall x, x\prime E(x, x\prime) \to F(\{f\}(x), \{g\}(x\prime))\end{aligned}$$

We now define *products, coproducts* and *exponents* in $\mathcal{C}_0^{ER}$ by:

$$\begin{aligned}\langle A, E\rangle \hat{\times} \langle B, F\rangle & \stackrel{\text{def}}{\equiv} & \langle A \times B, E \otimes F\rangle \\ \langle A, E\rangle \hat{+} \langle B, F\rangle & \stackrel{\text{def}}{\equiv} & \langle A + B, E \oplus F\rangle \\ \langle A, E\rangle \hat{\Rightarrow} \langle B, F\rangle & \stackrel{\text{def}}{\equiv} & \langle A \Rightarrow B, E \leadsto F\rangle\end{aligned}$$

Let *equality* of morphisms f, g in $Hom_{\mathcal{C}_0^{ER}}(\langle A, E\rangle, \langle B, F\rangle)$ be given by

$$(E \leadsto F)(f, g)$$

provably in **HA**. These operations make $\mathcal{C}_0^{ER}$ into a bi-Cartesian Closed Category, that is to say, the operations just defined really are a product, coproduct and exponent, and the arrow $\Lambda^* f \times 1_A$ in the cartesian diagram above is now unique. The proof is tedious but straightforward. We sketch the proof of uniqueness of the exponential map and leave the rest to the reader.(See [53] for the details).

We establish, then, that if both maps H and K:$\langle C, G\rangle \to \langle (A \Rightarrow B), E \leadsto F\rangle$, make the following diagram commute when substituted for "$(*)$":

[4]pointed out to the author by Andre Scedrov

$$\begin{tikzcd}
\langle A \Rightarrow B, E \rightsquigarrow F\rangle \times \langle A,E\rangle \arrow[dr, "app"] & \\
& B \times F \\
\langle C,G\rangle \times \langle A,E\rangle \arrow[uu, "(*) \times 1_A"] \arrow[ur, "f"'] &
\end{tikzcd}$$

then they are equal in the category $\mathcal{C}_0^{ER}$. Writing out the definitions, we have that from

$$\mathbf{HA} \vdash ((G \otimes E) \rightsquigarrow F)(app \circ (H \times 1_A), app \circ (K \times 1_A)) \tag{12}$$

we want to prove

$$\mathbf{HA} \vdash (G \rightsquigarrow (E \rightsquigarrow F))(H, K). \tag{13}$$

Writing out 12 in full gives:

$$\mathbf{HA} \vdash (\forall x, y, x', y')G(x, x') \,\&\, E(y, y') \rightarrow F(\{Hx\}(y), \{Kx'\}(y')) \tag{14}$$

which is equivalent to

$$\mathbf{HA} \vdash (\forall x, x')G(x, x') \rightarrow (\forall y, y')[E(y, y') \rightarrow F(\{Hx\}(y), \{Kx'\}(y')] \tag{15}$$

in other words

$$\mathbf{HA} \vdash (\forall x, x')G(x, x') \rightarrow (E \rightsquigarrow F)(Hx, Kx') \tag{16}$$

which is equivalent to 13 ∎.

Kripke Semantics The reader is advised to consult any standard text in intuitionistic logic, such as [92], or [89] for a more complete discussion of Kripke and Beth semantics. Because of some differences between our formulation and the standard one, however, we give the definition of truth in a Kripke model. Our notion of *forcing with covers* is similar in some respects to that given in Grayson's [29]. It is, in fact, a special case of forcing over a *site*, or a *Grothendieck topology*, as set forth in, e.g. [89].

Definition 2.3 *A Kripke/Beth model $\mathcal{K} = \langle P, \leq, D, \Vdash\rangle$ over the language $\mathcal{L}$ is a partial order $\langle P, \leq\rangle$, whose members are often called* nodes, *with a set-valued map D on P, associating a domain $D(p)$ to each $p \in P$, satisfying*

$$\forall p, q \in P(p \leq q \rightarrow \quad D(p) \subset D(q))$$

and a binary forcing relation $\Vdash$ on $P \times F_0$, where F_0 is the set of atomic sentences over the language $\mathcal{L}$.

The forcing relation satisfies the monotonicity condition

$$\forall p, q \in P(p \Vdash \varphi \text{ and } \quad p \leq q \rightarrow \quad q \Vdash \varphi).$$

$\mathcal{K}$ is also equipped with a *notion of cover* : a binary relation $Cov(p, \mathbf{S})$, between members $p \in P$ and subsets $\mathbf{S}$ of P. A quite general set of axioms for the cover relation guaranteeing soundness (for HA, IZF and other intuitionistic theories) is given in Grayson's paper.
This data fully specifies the model $\mathcal{K}$.

The forcing relation is extended to all sentences over the language $\mathcal{L}$ by the following inductive definition:

$p \Vdash \varphi \,\&\, \psi$ iff $p \Vdash \varphi$ and $p \Vdash \psi$.
$p \Vdash \varphi \rightarrow \psi$ iff $\forall q \geq p(q \Vdash \varphi \Rightarrow q \Vdash \psi)$.
$p \Vdash (\forall x)\varphi(x)$ iff $(\forall q \geq p)(\forall a \in D(q))(q \Vdash \varphi(a))$
$p \Vdash \varphi \vee \psi$ iff for some $\mathbf{S} \subset P$, $Cov(p, \mathbf{S})$ and $\forall q \in \mathbf{S}(q \Vdash \varphi$ or $q \Vdash \psi)$
$p \Vdash (\exists x)\varphi(x)$ iff for some $\mathbf{S} \subset P$, $Cov(p, \mathbf{S})$ and
$\forall q \in \mathbf{S}$ (there is a $c \in D(q)$ with $q \Vdash \varphi(c)$) An $\mathcal{L}$ — sentence φ is *true in* $\mathcal{K}$ if it is forced at every $p \in P$. This state of affairs is denoted

$$\mathcal{K} \models \varphi.$$

It will be useful here to recast the $\exists-$ and $\forall$ - clauses of the above definition in terms of the existence predicate (as used in, e.g., the logic IQCE [89]). For our purposes, it will be sufficient to consider E exclusively as a semantical device. We just need to order formulas in such a way that, for example, $\varphi(c)$ and Ec are treated as "subformulas" of $\exists x \varphi(x)$, $\forall x \varphi(x)$ in induction arguments.

The Kripke, and realizability interpretations of this predicate are as follows:
Kripke: $(p \Vdash Ec) \equiv c \in D(p)$.
Realizability: $|Ec|(x) \equiv x = c$. Let $D \equiv \cup\{D(p) : p \in P\}$. In terms of these definitions, we can write:

$p \Vdash (\exists x)\varphi(x)$ iff for some $\mathbf{S} \subset P$, $Cov(p, \mathbf{S})$ and
$\forall q \in \mathbf{S}$ (there is a $c \in D$ with $q \Vdash Ec \,\&\, \varphi(c)$).
$p \Vdash (\forall x)\varphi(x)$ iff for every $c \in D \quad p \Vdash Ec \rightarrow \varphi(c)$

The reader can check that this gives precisely the same definition as before.

The model $\mathcal{R}$

We now define the Kripke/Beth structure $\mathcal{R} = \langle P, \leq, D, \Vdash \rangle$ we will use to model realizability as follows:

Nodes: the objects of $\mathcal{C}_0$.

Partial order: $A \geq B$ iff

$$(\exists e \in \omega)(\mathbf{HA} \vdash (\forall x)(A(x) \rightarrow \exists uv\ T(e,x,u) \,\&\, U(u) = v \,\&\, B(v))$$

i.e., if $Hom_{\mathcal{C}_0}(A,B)$ is nonempty. Strictly speaking, $\leq$ is a pre-partial order, but ignoring the difference will not cause us any trouble here.

Domain function: constant. $D(A) = \omega \cup \underline{C}$ for every A. Following the notation just introduced above we denote this set by D.

Recall $\underline{C} = \{c_n : n \in \omega\}$ is the set of constants we introduced into our language.

Now, for atomic closed sentences φ over the language $\mathcal{L}$, define:

$A \Vdash \varphi$ iff $A \geq |\varphi|$

i.e., for atomic φ we have, informally,

$A \Vdash \varphi$ iff for some $e \in \omega$, $\mathbf{HA} \vdash \forall x[A(x) \rightarrow \{e\}(x) \downarrow \,\&\, |\varphi|\ (\mathrm{ex})]$, which is equivalent to : $\mathbf{HA} \vdash \forall x[A(x) \rightarrow \varphi]$. Since atomic sentences are decidable in HA, for φ an atomic sentence in the language of HA, this is equivalent to:

A $\Vdash \varphi$ iff $\mathbf{HA} \vdash \varphi$ or $\mathbf{HA} \vdash \forall x(\neg A(x))$.

So true atomic $\varphi \in \mathcal{L}$ (HA) are always forced, false atomic arithmetic φ are forced only by those A which are provably uninhabited.

This will fully specify a *Kripke/Beth* model $\mathcal{R} = \langle P, \leq, D, \Vdash \rangle$ once we have given our notion of covers for P.

Definition 2.4 *If A is an object of $\mathcal{C}_0$ and $\mathbf{S}$ is a set of objects of $\mathcal{C}_0$, then we define $Cov(A,\mathbf{S})$ to be true whenever the following conditions hold:*

1) For every B in $\mathbf{S}$, $B \geq A$

2) If there is a C in $\mathcal{C}_0$ such that for every B in $\mathbf{S}$, $B \geq C$,
then $A \geq C$.

Conditions 1) and 2) simply state that A is the infimum of $\mathbf{S}$ in P.

Define $\mathbf{S}$ to be a *uniform cover of* A if $S(y,z)$ is a formula over $\mathcal{L}$ in two free variables such that the collection $\mathbf{S} = \{S(a,x) : a \in D\}$ is a cover of A and the following holds:

There is an $e \in \omega$ with

$\mathbf{HA} \vdash \forall x[A(x) \rightarrow \exists uv\ T(e,x,u) \,\&\, U(u) = v \,\&\, S(v_0,v_1)]$. In other words, $A \geq (\sum_x S)$.

Theorem 2.5 *For every sentence φ in arithmetic and every node A*
$A \Vdash \varphi$ iff $\exists e : A \rightarrow |\varphi|$, i.e iff $A \geq |\varphi|$

Proof: atomic case: by definition.

__and:__ $A \geq |\varphi \,\&\, \psi|$ means $(\exists e) e : A \rightarrow |\varphi| \times |\psi|$.

But then $\pi_0 \cdot e : A \rightarrow |\varphi|$ and $\pi_1 \cdot e : A \rightarrow |\psi|$, where $\pi_0 \cdot e$ is $\Lambda x.(ex)_0$, and $\pi_1 \cdot e$ is $\Lambda x.(\{e\}(x))_1$. Thus $A \geq |\varphi|$ and $A \geq |\psi|$ and, by inductive hypothesis, A $\Vdash \varphi$ and $A \Vdash \psi$, hence $A \Vdash \varphi \,\&\, \psi$.

Conversely, $A \Vdash \varphi \,\&\, \psi \Rightarrow A \Vdash \varphi$ and $A \Vdash \psi$, so by the induction hypothesis, $A \geq |\varphi|$ and $A \geq |\psi|$, say by f, g. Then $\langle f, g\rangle : A \to |\varphi| \times |\psi|$, so $A \geq |\varphi \,\&\, \psi|$.
or: Suppose $A \geq |\varphi \vee \psi|$. Then, for some e, $e : A \to |\varphi| + |\psi|$, i.e.,

$$\mathbf{HA} \vdash \forall x A(x) \to [(\{e\}(x))_0 = 0 \,\&\, |\varphi|((\{e\}(x))_1)] \vee [(\{e\}(x))_0 = 1 \,\&\, |\psi|((\{e\}(x))_1)]$$

Define:

$$A_0(x) \equiv A(x) \,\&\, (\{e\}(x))_0 = 0$$
$$A_1(x) \equiv A(x) \,\&\, (\{e\}(x))_0 = 1$$

Then the map e witnesses both $A_0 \geq |\varphi|$ and $A_1 \geq |\psi|$. Let $\mathbf{S} = \{A_0, A_1\}$. We claim $\mathbf{S}$ is a cover of A. $\Lambda x.x : A_i \to A$ for $i = 0$ or 1, so condition 1) is trivially met. Now suppose f and g are codes witnessing $A_0 \geq B$ and $A_1 \geq B$ respectively, for some object B. Then $[\![f, g]\!] : A_0 + A_1 \to B$, and if h is the code $\Lambda x.\langle(\{e\}(x))_0, x\rangle$, the code for the composition $h \cdot [\![f, g]\!]$ witnesses $A \geq B$. Now, by the inductive hypothesis, $(\forall B \in \mathbf{S}) B \Vdash \varphi$ or $B \Vdash \psi$. Thus, A $\Vdash \varphi \vee \psi$.

The other direction: say for some cover $\mathbf{S}$ of A $\forall B \in \mathbf{S}(B \Vdash \varphi$ or $B \Vdash \psi)$. Then, for every B in $\mathbf{S}$ there is an e such that $e : B \to |\varphi|$ or $e : B \to |\psi|$. But then we have a map $i_0 \cdot e$, or, equivalently $\Lambda x.\langle 0, \{e\}(x)\rangle : B \to |\varphi| + |\psi|$, or a map $i_1 \cdot e \equiv \Lambda x.\langle 1, \{e\}(x)\rangle : B \to |\varphi| + |\psi|$. Thus, $\forall B \in \mathbf{S}(B \geq |\varphi| + |\psi|)$. By the definition of covers: $A \geq |\varphi \vee \psi|$.
negation: see the argument in Theorem 3.9.
implies : Suppose $A \geq |\varphi \to \psi|$, that is to say, for some $f \in \omega$, HA proves

$$f : A \to (|\varphi| \Rightarrow |\psi|).$$

Further suppose $B \geq A$ and $B \Vdash \varphi$. Then, by the induction hypothesis we have $g : B \to |\varphi|$, and also, by definition, $h : B \to A$ in $\mathcal{C}_0$. Then $\mathbf{HA} \vdash \forall x[B(x) \to |\psi|(f(h(x))(g(x)))]$. [5]
So $\Lambda x.\ f(h(x))(g(x))$ witnesses $B \geq |\psi|$. By the induction hypothesis $B \Vdash \psi$. Therefore, $A \Vdash \varphi \to \psi$.

Conversely, suppose $(\forall B \geq A) B \Vdash \varphi \Rightarrow B \Vdash \psi$. Then, in particular, since $A \times |\varphi| \geq A$ via π_0, and $A \times |\varphi| \geq |\varphi|$ via π_1, hence $A \times |\varphi| \Vdash \varphi$ by the induction hypothesis, we have that $A \times |\varphi| \Vdash \psi$. By the inductive hypothesis again there is an f and a proof in HA that $f : A \times |\varphi| \to |\psi|$, hence by weak bi-Cartesian closure, $\Lambda f : A \to (|\varphi| \Rightarrow |\psi|)$, and therefore $A \geq |\varphi \to \psi|$.
exists : Suppose $A \geq |\exists x \varphi(x)|$ via a code e. Then,

$$\mathbf{HA} \vdash \forall x[A(x) \to \exists uv(T(e, x, u) \,\&\, U(u) = v \,\&\, |\varphi(v_0)|(v_1))].$$

Let $S(y, x) \equiv A(x) \,\&\, (\{e\}(x))_0 = y$. Let $\mathbf{S} = \{S(c, x) : C \in D\}$. We claim that

1) for each $c \in D$, $S(c, x) \geq |Ec \,\&\, \varphi(c)|$, and

2) $Cov(A, \mathbf{S})$.

By the hypothesis and elementary logic, HA proves

$$A(x) \,\&\, (ex)_0 = c \to \ (ex)_0 = c \,\&\, |\varphi(c)|((ex)_1).$$

[5]For the sake of legibility, we will write, *e.g.* $f(h(x))(g(x))$ or even $fhx(gx)$ instead of $\{\{f\}(\{h\}(x))\}(\{g\}(x))$.

This is assertion 1).
Trivially, for any c in D, the code $\Lambda x.x$ witnesses $S(c,x) \geq A$. Now suppose for some object B and every c in D , there is a code $e_c \in \omega$ such that

$$\mathbf{HA} \vdash \forall x[S(c,x) \rightarrow \exists uv(T(e_c,x,u) \,\&\, U(u) = v \,\&\, B(v))].$$

Pick c not in the language of HA or in A or B. Then for a fixed index e' corresponding to this c we obtain, by generalizing on the constant c:

$$\mathbf{A} \vdash (\forall z)(\forall x)[S(z,x) \rightarrow \exists uv(T(e',x,u) \,\&\, U(u) = v \,\&\, B(v))].$$

Now observe that $A \geq (\sum_z S(z,x))$. It follows easily from the hypothesis that $A(x) \rightarrow (ex)_0 \downarrow \,\&\, A(x) \,\&\, (ex)_0 = (ex)_0$ provably in HA. But this is precisely the assertion $A(x) \rightarrow S((ex)_0, x)$, i.e. $\Lambda x.\langle (ex)_0, x\rangle : A \rightarrow (\sum_z S(z,x))$. Composing e' with this map gives a witness to 2). Now apply the inductive hypothesis to 1) to obtain $(\forall B \in \mathbf{S})(\exists c \in D) B \Vdash Ec \,\&\, \varphi(c)$, and therefore, $A \Vdash \exists x \varphi(x)$.

One might point out here that $\mathbf{S}$ satisfies a rather strong uniformity condition, in addition to being a uniform cover of A, namely

$$(\forall c \in D) S(c,x) \Vdash \varphi(c).$$

Our arguments work if we restrict our definition of cover to uniform ones and define the forcing of existential sentences in this more stringent way.

Conversely, suppose A $\Vdash \exists x \varphi(x)$. Then for some cover $\mathbf{S}$ of A we have $(\forall B \in \mathbf{S})(\exists a \in D) B \Vdash \varphi(a)$, hence by the inductive hypothesis, $B \geq |\varphi(a)|$. But if, say, $f : B \rightarrow |\varphi(a)|$ then $\Lambda x.\langle a, f(x), n\rangle : B \rightarrow |\exists x \varphi(x)|$. So, in fact, for every B in $\mathbf{S}$, we have $B \geq |\exists x \varphi(x)|$, hence, by the definition of a cover, $A \geq |\exists x \varphi(x)|$.
<u>forall:</u> Suppose $A \geq |\forall x \varphi(x)|$ via a $\mathcal{C}_0$ - morphism e. In other words

$$\mathbf{HA} \vdash \forall x[A(x) \rightarrow \exists uv(T(e,x,u) \,\&\, U(u) = v \,\&\, (\textstyle\prod_x |\varphi(x)|)(v))].$$

In terms of the E- predicate (recall that $|Ez|(x) \equiv x = z$) :

$$\mathbf{HA} \vdash \forall x[A(x) \rightarrow \exists uv(T(e,x,u) \,\&\, U(u) = v \,\&\, \forall z(|Ez \rightarrow \varphi(z)|(e)))]$$

Thus, for every c in D we have $A \geq |Ec \rightarrow \varphi(c)|$, and hence, by the inductive hypothesis, $A \Vdash Ec \rightarrow \varphi(c)$. This establishes A $\Vdash \forall x \varphi(x)$.

Conversely, suppose $A \Vdash \forall x \varphi(x)$, that is to say, for every c in D $A \Vdash Ec \rightarrow \varphi(c)$. Then, by inductive hypothesis, for every c in D there is some code $e_c \in \omega$ such that

$$\mathbf{HA} \vdash \forall x[A(x) \rightarrow \exists uv(T(e_c,x,u) \,\&\, U(u) = v \,\&\, |Ec \rightarrow \varphi(c)|(v))].$$

As in the cover argument above, pick c not in $\mathcal{L}(HA + \{A, \varphi(x)\})$. Then for the fixed code $e' = e_c$ corresponding to this c, we have, generalizing on c,

$$\mathbf{HA} \vdash \forall x[A(x) \rightarrow \exists uv(T(e',x,u) \,\&\, U(u) = v \,\&\, (\forall z)|Ez \rightarrow \varphi(z)|(v))],$$

hence, $A \geq (\prod_z |\varphi(z)|)$, and thus, $A \geq |\forall x \varphi(x)|$. ∎

Corollary 2.6 $\mathcal{K} \models \varphi$ *iff* $\mathbf{HA} \vdash \exists x(x \mathrel{\underset{\sim}{\mathbf{r}}} \varphi)$.

Proof: Suppose $\mathcal{K} \models \varphi$. Then if $\{0\}$ is the predicate $\{0\}(x) \equiv x = 0$, we have $\{0\} \Vdash \varphi$. Hence, there is an f such that $\mathbf{HA} \vdash |\varphi(f(0))|$. Therefore $\mathbf{HA} \vdash f(0) \mathrel{\underset{\sim}{\mathbf{r}}} \varphi$, so $\mathbf{HA} \vdash \exists x(x \mathrel{\underset{\sim}{\mathbf{r}}} \varphi$).

Conversely, suppose $\mathbf{HA} \vdash \exists x(x \mathrel{\underset{\sim}{\mathbf{r}}} \varphi)$. By the existence property for HA, there is a numeral n such that $\mathbf{HA} \vdash n \mathrel{\underset{\sim}{\mathbf{r}}} \varphi$. Hence, $\mathbf{HA} \vdash \forall x(x = 0 \to \quad |\varphi|((\Lambda x.n)0)$, in other words, $\Lambda x.n : \{0\} \to |\varphi|$. $\{0\}$ is terminal in $\mathcal{C}_0$, as is any provably inhabited predicate, hence the least member of $\mathcal{K}$ ($\Lambda x.0$ witnesses $A \geq \{0\}$ for any A).
Therefore, for any A in the pre-order of $\mathcal{K}$, $A \geq |\varphi|$, hence A $\Vdash \varphi$. So $\mathcal{K} \models \varphi$. ▮.
Let us review some of the main features of the model just described. Using the definition:

$$A \geq B$$

$$\text{iff}$$

$$\exists x \in Hom_{\mathcal{C}_0}(A, B)$$

we established a correspondence between the resulting Kripke model and the weakly bi – Cartesian Closed structure induced by the realizing maps on first-order arithmetic formulas with parameters. The correspondence is an elementary equivalence with respect to the different semantics of the two structures.

The construction in fact gives us a whole class of Kripke models. For each formula A in one free variable, the collection of all $B \geq A$ constitutes a Kripke model, of all sentences "realized relative to A". We will not explore these possibilities here, but rather concentrate on exploiting the correspondences just described in varying contexts.

3 A Kripke Model for abstract realizability

The construction in the preceding section is easily generalized to realizability interpretations induced by an abstract theory of partial application. There are various versions of such a theory, APP introduced by Feferman in the mid 1970's (see [89]), the system of the same name formalized in set theory in McCarty's [56] and Beeson's EON [1]. (Except for McCarty's presentation, in these theories the notions of convergence and partial application are usually taken as primitive. We adopt Beeson's EON here.

To avoid introducing two sets of variables, formal variables and metavariables (which range over terms built up using the former) EON adopts the convention that every variable converges, but does not allow substitution of terms for variables in a universally quantified statement unless said terms converge. Thus $(\forall x)x \downarrow$ is a theorem, but $t \downarrow$ is not, for an arbitrary term t since the axiom $\forall x A \,\&\, t \downarrow \to \ A[t/x]$ cannot be applied unless $t \downarrow$ has already been shown.

We now briefly describe the theory EON, as does Beeson, as an extension of a more basic theory PCA. We refer the reader to Beeson's book for the details.

The logic of partial terms (LPT): This logic includes the usual rules for propositional logic plus the following rules of inference:

$$R\forall\ \frac{B \to A}{B \to \forall x A} \qquad R\exists\ \frac{A \to B}{\exists x A \to B} \quad (x \text{ not free in } B)$$

and the following axioms (note that A1, A2, A4, A5, A6 are axiom schemas using metavariables t, s, t_i for arbitrary terms and A7, A8 schemas for special terms.)
(A1) $\forall x A \,\&\, t \downarrow \to A[t/x]$
(A2) $A[t/x] \,\&\, t \downarrow \to \exists x A$
$\downarrow$ is a unary (post-fix) relation symbol in the language
$\simeq$ is a defined binary relation symbol

$$t \simeq s \equiv t \downarrow \vee s \downarrow \to \quad t = s.$$

We have the following axioms governing $\downarrow$, $\simeq$ *and* $=$

(A3) $x = x \,\&\, (x = y \to y = x)$
(A4) $t \simeq s \,\&\, \varphi(t) \to \varphi(s)$
(A5) $t = s \to t \downarrow \,\&\, s \downarrow$
(A6) $R(t_1, ..., t_n) \to t_1 \downarrow \,\&\, ... \,\&\, tn \downarrow$ for any atomic formula $R(t_1, ..., t_n)$ and any terms $t_1, ..., t_n$.
(A7) (i) For each constant symbol $c : c \downarrow$
(A8) (ii) For each variable $x : x \downarrow$
We note that (A5) is a special case of (A6). Another important special case of (A6) is
(A6') $f(t_1, t_2, ..., t_n) \downarrow \to t_1 \downarrow \,\&\, t_2 \downarrow \,\&\, ... \,\&\, t_n \downarrow$
N.B. (A6) does NOT imply that for any formula $\varphi, \varphi(t_1, ..., t_n) \to \ t_1 \downarrow \,\&\, ... \,\&\, t_n \downarrow$. Consider,e.g. $\neg t \downarrow \to t \downarrow$.

PCA : We now introduce the theory PCA over the logic of partial terms.
Language: Two constants, **k** and **s**.
A binary function symbol **Ap**
We will never explicitly write **Ap**. We use juxtaposition, (st) , or just st , to denote $Ap(s, t)$.
Axioms of PCA:
Those of LPT together with
(PCA1) $\mathbf{k}xy = x$
(PCA2) $\mathbf{s}xyz \simeq xz(yz) \,\&\, \mathbf{s}xy \downarrow$
(PCA3) $\mathbf{k} \neq \mathbf{s}$
We now define **EON**, Beeson's *Elementary Theory of Operations and Numbers* ([1]). It is **PCA** together with
constants $\pi_0, \pi_1, p, d, S_N, P_N$, 0,and a predicate letter N, with axioms

EON1 pxy $\downarrow \,\&\, \pi_0(pxy) = x \,\&\, \pi_1(pxy) = y$

EON2 $N(0) \mathbin{\&} \forall x(N(x) \to [N(S_N(x)) \mathbin{\&} P_N(S_N x) = x \mathbin{\&} S_N x \neq 0])$

EON3 $\forall x(N(x) \mathbin{\&} x \neq 0 \to N(P_N x) \mathbin{\&} S_N(P_N x) = x)$

EON4 Definition by integer cases

$$N(a) \mathbin{\&} N(b) \mathbin{\&} a = b \to \quad d(a,b,x,y) = x$$

$$N(a) \mathbin{\&} N(b) \mathbin{\&} a \neq b \to \quad d(a,b,x,y) = y$$

EON5 Induction schema: for each formula φ

$$\varphi(0) \mathbin{\&} \forall x[N(x) \mathbin{\&} \varphi(x) \to \quad \varphi(S_N x)] \to \forall x(N(x) \to \quad \varphi(x)).$$

We will often write
if $a = b$ **then** x **else** y for $d(a,b,x,y)$
$\langle x, y\rangle$ *for* pxy.

Proofs of the following results about **EON** (and related systems) can be found in Beeson's book [1] or [89].

Theorem 3.1 (The Recursion Theorem) *There is a term R such that PCA proves*

$$Rf\downarrow \mathbin{\&} [g = Rf \to \forall x(gx \simeq fgx)]$$

Theorem 3.2 *Let M be a model of EON. Then every partial recursive function is numerically representable in M.*

Theorem 3.3 (Numerical and Term Existence properties) *If* $EON \vdash \exists x A$ *then there is a term t such that*

$$EON \vdash t\downarrow \mathbin{\&} A(t).$$

If $EON \vdash \exists n(N(n) \mathbin{\&} A(n))$ *there is a numeral* $\bar{m} = \underbrace{s(s...s(}_{m}0)...)$ *such that* $EON \vdash A(\bar{m})$.

Abstract realizability and the auxiliary theories EONb In EON we can define an abstract notion of realizability, which gives rise to a realizability interpretation for each model of EON. Even in the syntax, however, we can easily generalize our treatment to a broad class of auxiliary theories EON**b** introduced in Beeson's [1]. These theories are obtained by adding a formal predicate for membership in Baire space $x \in \mathbf{N}^{\mathbf{N}}$, a new constant **b**, the axiom $\mathbf{b} \in \mathbf{N}^{\mathbf{N}}$ and a set of axioms $\mathbf{b}(\bar{n}) = \overline{b(n)}$ (termed the "diagram" of **b** by Beeson). One must also extend all axiom schemas of EON to the new language. Each **b**-parametrized theory induces its own realizability notion. The same soundness properties hold, and the proofs of the numerical and term existence properties cited in Theorem 3.3 (upon which our Corollary 3.10 depends) also follow, as is shown in [1]. Although we will not make explicit mention of this "Baire-space parameter" **b** in what follows, the reader should think of EON as being enriched by an arbitrary equational theory [6].

Alternatively, we may take the approach described in [51] to incorporate other theories into our formal system: interpret an arbitrary first order theory **S** into EON by adding a universe or domain predicate U whose extension is the target of the interpretation. This is essentially the approach defined by Shoenfield in [83]. Then we leave "up to the reader" how to define the atomic realizability $|A|(x)$ of formulas A from the object language $\mathcal{L}(\mathbf{S})$ by terms x from the *realizing metalanguage* EON. We will only require that the atomic realizability be faithfully copied by the atomic forcing assignments. The essential point is that we wish to consider almost any syntactic realizability over any theory. The role of EON is only to supply abstract realizers. We will not explicitly develop this approach here, however, since the notation is cumbersome and the details not especially enlightening.

The following definitions hold for EON (and EON**b**) as well as the variants described above, and the enrichment by new constants EON$\underline{C}$ (and EON**b**$\underline{C}$) we will be considering below.

Definition 3.4 *Let A, B be formulas in one free variable over the language of EON (with possibly a denumerable set of new constants added). Then*

$(A \times B)(x) \equiv A(\pi_0 x) \,\&\, B(\pi_1 x)$

$(A + B)(x) \equiv N(\pi_0 x) \,\&\, (\pi_0 x = 0 \to |A|(\pi_1 x))$

$\&\, (\pi_0 x \neq 0 \to |B|(\pi_1 x))$

$(A \Rightarrow B)(x) \equiv \forall y[A(y) \to xy \downarrow \,\&\, B(xy)]$

Let $A(x, y)$ be a formula in two free variables. Then $(\Sigma_x A(x, y)$ and $(\Pi_x A(x, y))$ are formulas in one free variable given by

$$(\textstyle\sum_x A(x, y))(z) \equiv A(\pi_0 z, \pi_1 z)$$
$$(\textstyle\prod_x A(x, y))(z) \equiv \forall y[(zy) \downarrow \,\&\, A(y, zy)]$$

Definition 3.5 *Let A, B be sentences over the language of EON (EON$\underline{C}$). Then we define inductively the realizability formulas $|A|$ in one free variable as follows:*

If A is prime $|A|(x)$ is $A \,\&\, x \downarrow$

$|A \,\&\, B| \equiv |A| \times |B|$

$|A \vee B| \equiv |A| + |B|$

$|A \to B| \equiv |A| \Rightarrow |B|$

$|\exists y A(y)| \equiv (\Sigma_y |A(y)|(x))$, *i.e.* $|\exists y A(y)|(z) \equiv |A(\pi_0 z)|(\pi_1 z)$

$|\forall y A(y)| \equiv (\Pi_y |A(y))(x)$,*i.e.* $|\forall y A(y)|(z) \equiv \forall y[(zy) \downarrow \,\&\, |A(y)|(zy)]$.

$|\neg A| \equiv |A \to \perp| \equiv \forall y \neg |A|(y)$

$|A|(x)$ is usually written $x \,\underset{\sim}{\mathbf{r}}\, A$. Note that if A is a formula in n variables over EON (or EON$\underline{C}$) then the above clauses defined an associated realizability formula in $n + 1$ variables. Note

[6] as Beeson points out, the enrichment of EON need not be equational: the only requirement is that the extra axioms $\phi(\mathbf{b})$ about $\mathbf{b}$ should be *self-realizing*. See Beeson,*op.cit.* for details.

that if A is *prime*, A is logically equivalent to $|A|(x)$ for any variable x, since by the LPT conventions, $A \to A \,\&\, (x \downarrow)$. However $|A|[t/x]$ is not logically equivalent to A since $|A|[t/x]$ is $A \,\&\, (t \downarrow)$, which requires additional proof. The point of this definition is to guarantee the base case of the following lemma, which is easily proved by induction.

Lemma 3.6 *$EON \vdash |A|(t) \to t \downarrow$ for every sentence A, term t.*

Let $\underline{C} = \{c_i | i \in \omega\}$ be a denumerable set of fresh constants. EON$\underline{C}$ is the theory EON, together with the constants in $\underline{C}$, the axioms $c \downarrow$ for each $c \in \underline{C}$ and all schemas extended to the new language. This means, e.g. we have the following new instances of the axioms:

$$\forall x A \,\&\, c \downarrow \to \quad A[c/x]$$

$$t \simeq c \,\&\, \varphi(t) \to \quad \varphi(c)$$

etc. However, no $c \in \underline{C}$ occurs in a non-logical axion of EON$\underline{C}$.

Let $\mathcal{L}$ be the language of EON$\underline{C}$. Let Δ be the set of all formal closed (variable-free) terms of $\mathcal{L}$. We now define a category $\mathcal{C}(\underline{C})$ as follows:

objects: formulas in one free variable over $\mathcal{L}$.

morphisms: triples (A, t, B) where A and B are objects, $t \in \Delta$
and EON$\underline{C}$ $\vdash \forall x[A(x) \to tx \downarrow \,\&\, B(tx)]$.

Equality of morphisms is provable extensional equality.

The following lemma is essentially a restatement of the so-called *combinatory completeness* property of the lambda calculus:

Lemma 3.7 (The a-lemma) *Let a be a constant in $\underline{C}$ and define Δ/a to be the set of all terms of Δ in which a does not occur.*
Then $e \in \Delta \Rightarrow \exists h \in \Delta/a$ such that

$$EON\underline{C} \vdash ha \simeq e.$$

Proof: By induction on the structure of terms in Δ. First we distinguish a special case:
if a does not occur in e, let $h = \mathbf{k}e$
if a does occur in e :
case 1: e is a. Then h is $\lambda x \cdot x \equiv \mathbf{skk}$.
case 2: e is (tu) and by the induction hypothesis, there are terms $t', u' \in \Delta/a$ such that $t \simeq t'a$ and $u \simeq u'a$ are provable in EON$\underline{C}$. Hence $(tu) \simeq (t'a)(u'a)$ so $h \simeq (\mathbf{s}t'u')$ ▮

Note: The notation $\Delta/\{c_1, ..., c_n\}$ will mean all terms $t \in \Delta$ in which none of $c_1, ..., c_n$ occur.

Lemma 3.8 *$\mathcal{C}(\underline{C})$ is weakly bi-Cartesian closed.*

Proof: The proof is similar to that of lemma 2.2. We sketch the arguments symbolically by expressing the relevant steps as derived rules of inference.

$$\pi_0 : A \times B \to A$$

$$\pi_1 : A \times B \to B$$

$$\frac{f : C \to A \quad g : C \to B}{\ll f, g \gg : C \to A \times B} \qquad \text{where } \ll f,g \gg \; \equiv \lambda z \cdot \langle fz, gz \rangle$$

Define $\iota_0 \equiv \lambda x \cdot \langle 0, x \rangle, \iota_1 \equiv \lambda x \cdot \langle 1, x \rangle$

Then $\iota_0 : A \to A + B$, $\iota_1 : B \to A + B$
and

$$\frac{f : A \to C \quad g : B \to C}{[\![f, g]\!] : A + B \to C}$$

where $[\![f, g]\!] \equiv \lambda x \cdot$ **if** $\pi_0 x = 0$ **then** fx **else** gx

$$\frac{f : A \times B \to C}{\lambda^* f : A \to (B \Rightarrow C)} \qquad \text{where } \Lambda^* f \equiv \lambda x \cdot \lambda y \cdot f(\langle x, y \rangle)$$

$$\frac{g : A \to (B \Rightarrow C)}{\Lambda_* g : A \times B \to C} \qquad \text{where } \Lambda_* g \equiv \lambda x \cdot (g(\pi_0 x))(\pi, x).$$

▮

We are now in a position to define the Kripke model we want. Nodes are objects of $\mathcal{C}(\underline{C})$ with ordering given by

$$A \geq B \text{ iff } (\exists t \in \Delta) \text{ EON}\underline{\text{C}} \vdash \forall x[A(x) \to \quad tx \downarrow \,\&\, B(tx)]$$

i.e.

$$\text{EON}\underline{\text{C}} \vdash (A \Rightarrow B)(t).$$

We will sometimes denote this state of affairs by A $\overset{t}{\geq}$ B, or $t : A \to B$. The domain of the Kripke Model is the constant domain Δ. Our notion of covers (see section I) will be as follows:

For A an object of $\mathcal{C}(\underline{C})$, and **S** a set of objects of $\mathcal{C}(\underline{C})$, then $Cov(A, \mathbf{S})$ iff

1. $\forall B \in \mathbf{S} \exists t \in \Delta \quad t : B \to A$

2. if whenever, for some object D

$$\forall B \in \mathbf{S} \exists t_B \in \Delta \; t_B : B \rightarrow \quad D$$

then $(\exists t \in \Delta)\, t : A \rightarrow D$, i.e., $A = \inf(\mathbf{S})$ in $\langle \mathbf{P}, \leq \rangle$ (see section 2 for notation). The definition of forcing is, for nodes A and $\mathcal{L}$ -sentences φ, similar to that of the preceding section, except that we will not require the existence predicate, E.

(1) A $\Vdash \varphi$ iff $(\exists t \in \Delta) t : A \rightarrow |\varphi|$ for atomic φ

(2) A $\Vdash \varphi \,\&\, \psi$ iff A $\Vdash \varphi$ and $A \Vdash \psi$

(3) A $\Vdash \varphi \vee \psi$ iff $(\exists \mathbf{S}) Cov(A, \mathbf{S})$ and $(\forall B \in \mathbf{S}) B \Vdash \varphi$ or $B \Vdash \psi$

(4) A $\Vdash \varphi \rightarrow \psi$ iff $(\forall B \geq A) B \Vdash \varphi \Rightarrow B \Vdash \psi$

(5) A $\Vdash \exists x \varphi(x)$ iff $(\exists \mathbf{S}) Cov(A, \mathbf{S})$ and $(\forall B \in \mathbf{S})(\exists a \in \Delta) B \Vdash \varphi(a)$

(6) A $\Vdash \forall x \varphi(x)$ iff $(\forall a \in \Delta) A \Vdash \varphi(a)$.

Theorem 3.9 *For every node A and sentence* φ

$$A \Vdash \varphi \text{ iff } (\exists t \in \Delta) t : A \rightarrow |\varphi|$$

Proof: By induction on the structure of φ, i.e. the inductive hypothesis is that $\forall A$ and every θ of simpler structure than φ

$$\text{A} \Vdash \theta \text{ iff } (\exists t \in \Delta) t : A \rightarrow |\theta| \;.$$

The atomic case is by definition. The proofs for conjunction, disjunction and implication are similar to those in Theorem 2.5. We will carry out the argument for the negative, universal and existential cases.

<u>exists:</u> Suppose A $\Vdash \exists x \theta(x)$.

Then, for some $\mathbf{S}$, $Cov(A, \mathbf{S})$ and $(\forall B \in \mathbf{S})(\exists a \in \Delta) B \Vdash \theta(a)$. Therefore, for some $f \in \Delta \;\; f : B \rightarrow \theta(a)$. But then $\lambda x \cdot \langle a, fx \rangle : B \rightarrow |\exists x \theta(x)|$, and the covering property gives $A \geq |\exists x \theta(x)|$.

Conversely, suppose $f : A \rightarrow |\exists x \theta(x)|$. Then

$$\text{EON}\underline{\text{C}} \;\vdash \forall x [A(x) \rightarrow fx \downarrow \;\&\; |\theta(\pi_0(fx))|(\pi_1(fx)) \;\&\; \pi_0(fx) \downarrow].$$

Let $S(t, x)$ be the formula $A(x) \,\&\, \pi_0(fx) = z$. Then $\lambda x \cdot \pi_1(fx) : S(z, x) \rightarrow |\theta(z)|$ hence, for each $a \in \Delta$, $S(a, x) \geq |\theta(a)|$. (The hypothesis $a \downarrow$ is built into $S(a, x)$, i.e., $\pi_0(fx) = a \rightarrow a \downarrow$). By the inductive hypothesis $S(a, x) \Vdash \theta(a)$.

We claim $\{S(a, x) |\; a \in \Delta\}$ is a (uniform) cover of A. $S(a, x) \geq A(x)$ via the identity (inclusion). Suppose, for each a, $e_a : S(a, x) \rightarrow B$, i.e.

$$\text{EON}\underline{\text{C}} \;\vdash \forall x [S(a, x) \rightarrow e_a x \downarrow \;\&\; B(e_a x)]$$

Then, pick $a \notin \mathcal{L}(EON + \{B(x), A(x)\})$, i.e., a not in the non-logical part of EON<u>C</u>
Then

$$\text{EON}\underline{\text{C}} \;\vdash \forall x [S(a, x) \rightarrow e_a x \downarrow \;\&\; B(e_a x)]$$

(once again, by the definition of S, the hypothesis $a \downarrow$ is not required). By the a-lemma 3.7 there is an $h \in \Delta / a$ such that

$$\text{EON}\underline{\text{C}} \;\vdash h \downarrow \;\&\; ha \simeq e_a.$$

Thus EON$\underline{C}$ $\vdash \forall x[S(a,x) \rightarrow (ha)x\downarrow \ \& \ B(ha)x)]$. By generalization on constants

$$EON\underline{C}/a \vdash \forall x \forall z[S(z,x) \rightarrow (hz)x\downarrow \ \& \ B((hz)x)].$$

By the definition of S

$$\text{EON}\underline{\text{C}} \ \vdash \forall x[A(x) \rightarrow (h\pi_0(fx))x\downarrow \ \& \ B((h\pi_0(fx))x)]$$

so $A \geq B$. This establishes $Cov(A,S)$. (Uniformity, *i.e.* $A \geq (\Sigma_z S(z,y))$ is easily checked). Therefore

$$A \Vdash \exists x \theta(x).$$

<u>for all:</u> Suppose $A \Vdash \forall x \theta(x)$. Then, for every $a \in \Delta, A \Vdash \theta(a)$. Hence, by the inductive hypothesis

$$(\forall a \in \Delta)(\exists e_a \in \Delta) \ \text{EON}\underline{\text{C}} \ \vdash \forall x[A(x) \rightarrow \quad e_a x\downarrow \ \& \ |\theta(a)|(e_a x)]$$

Now we make the same moves as in the existential case: Pick a constant

$$a \notin \mathcal{L}\,(EON \ \cup \ \{A(x), \ \theta(x)\})$$

and, by the a-lemma, find an $h \in \Delta/a$ with $e_a \simeq ha$ provably in EON$\underline{C}$. Then

$$\text{EON}\underline{\text{C}} \ \vdash \forall x[A(x) \rightarrow hax\downarrow \ \& \ |\theta(a)|(hax)].$$

Generalizing on fresh constants, we have, for $g = \lambda x \cdot \lambda z \cdot hzx$, that

$$\text{EON}\underline{\text{C}} \ \vdash \forall x[A(x) \rightarrow gx\downarrow \ \& \ \forall z(gxz\downarrow \ \& \ |\theta(z)|(gxz))].$$

But this is $A \geq |\forall x \theta(x)|$.

<u>negation:</u> Say A $\Vdash \neg\varphi$.

Then $(\forall B) B \geq A$ and $B \Vdash \varphi \Rightarrow B \Vdash \perp$ (i.e. $B \Vdash$ everything). Then $A \times |\varphi| \Vdash \perp$.
By the inductive hypothesis $A \times |\varphi| \geq |\perp|$, and by weak cartesian closure $A \geq |\varphi| \Rightarrow |\perp|$ i.e. $A \geq |\varphi \rightarrow \perp|$, so for some g, EON$\underline{C}$ $\vdash \forall x[A(x) \rightarrow gx\downarrow \ \& \ (|\varphi| \Rightarrow |\perp|)(gx)]$, hence

EON$\underline{C}$ $\vdash \forall x[A(x) \rightarrow gx\downarrow \ \& \ \forall z(|\varphi|(z) \rightarrow gxz\downarrow \ \& \ |0=1|(gxz))$, so in particular,
EON$\underline{C}$ $\vdash \forall x[A(x) \rightarrow gx\downarrow \ \& \ \forall z(|\varphi|(z) \rightarrow 0=1)]$, equivalently,
EON$\underline{C}$ $\vdash \forall x[A(x) \rightarrow gx\downarrow \ \& \ \forall z \neg |\varphi|(z)]$, hence:
EON$\underline{C}$ $\vdash \forall x[A(x) \rightarrow gx\downarrow \ \& \ |\neg\varphi|(gx)]$, i.e.,
$A \geq |\neg\varphi|$.

Conversely, suppose $f : A \rightarrow |\neg\varphi|$. Then
EON$\underline{C}$ $\vdash \forall x[A(x) \rightarrow fx\downarrow \ \& \ \forall z \neg |\varphi|(z)]$, equivalently,
EON$\underline{C}$ $\vdash \forall x[A(x) \rightarrow fx\downarrow \ \& \ \forall z(|\varphi|(z) \rightarrow 0=1)]$, from which it is easily shown that:
EON$\underline{C}$ $\vdash \forall x[(A \times |\varphi|)(x) \rightarrow f(\pi_0 x)\downarrow \ \& \ 0=1]$, in particular,

EON$\underline{C}$ $\vdash \forall x[(A \times |\varphi|)(x) \to |0 = 1|(x)]$, i.e.,
$A \times |\varphi| \geq | \perp |$.

By the inductive hypothesis $A \times |\varphi| \Vdash \perp$, so if $B \Vdash \varphi$ and $B \geq A$ then $B \geq |\varphi|$ so $B \geq A \times |\varphi|$ and $B \Vdash \perp$. But then A $\Vdash \varphi \to \perp$ i.e., $A \Vdash \neg\varphi$. ∎

Now, we immediately have the desired result, to wit, that the model just constructed is elementarily equivalent to abstract realizability over EON.

Corollary 3.10 *Let φ be a sentence in the language of EON, $\mathcal{K}$ the Kripke model described above. Then*

$$\mathcal{K} \models \varphi \text{ iff } EON \vdash \exists x(x \underset{\sim}{\mathrm{r}} \varphi)$$

(where $x \underset{\sim}{\mathrm{r}} \varphi$ is the traditional notation for $|\varphi|(x)$).

Proof: Suppose $\mathcal{K} \models \varphi$. Then $\{0\} \models \varphi$ ($\{0\}$ being the formula $x = 0$). By the theorem, there is an f in Δ such that

$$\text{EON}\underline{C} \vdash (\forall x)[x = 0 \to fx \downarrow \ \& \ |\varphi|(fx)$$

By the a-lemma, 3.7, applied repeatedly, if $\{c_1, ..., c_n\}$ is the set of all constants from $\underline{C}$ occurring in f, we can find a term f' in the language of EON such that

$$\text{EON}\underline{C} \vdash \forall x[(x = 0) \to (f'c_1...c_m)x \downarrow \ \& \ |\varphi|((f'c_1...c_m)x)]$$

Letting $\Gamma(\vec{c})$ be the conjunction of the finitely many axioms of EON$\underline{C}$ involving the constants $c_1, ..., c_m$ used in this proof

$$\text{EON} \vdash \Gamma(\vec{c}) \to \forall x[(x = 0) \to (f'c_1...c_m)x \downarrow \ \& \ |\varphi|((f'c_1...c_m)x)]$$

by generalization on new constants:

$$\text{EON} \vdash \forall z_1...z_m\Gamma(\vec{z}) \to \forall x[(x = 0) \to (f'z_1...z_m)x \downarrow \ \& \ |\varphi|((f'z_1...z_m)x)]$$

Observe that every axiom involving some c is trivially satisfied by a variable, hence $\Gamma(\vec{z})$ is a conjunction of axioms of LPT. Therefore

$$\text{EON} \vdash (\forall z_1...z_m)[f'(z_1...z_m)0 \downarrow \ \& \ |\varphi|((f'(z_1...z_m))0)]$$

Now substituting any provably convergent term, *e.g.* 0 for all of the z_i, and denoting $(f'000...0)0$ by e, a term in $\mathcal{L}$ (EON) we have

$$\text{EON} \vdash e \downarrow \ \& \ |\varphi|(e)$$

hence

$$\text{EON} \vdash \exists x(x \underset{\sim}{\mathrm{r}} \varphi).$$

Conversely, suppose EON $\vdash \exists x(x \underset{\sim}{\mathbf{r}} \varphi)$, i.e. EON $\vdash \exists x(|\varphi|(x))$ with φ a sentence in the language of EON. By the term existence property of EON (Theorem 3.3), there is a provably convergent term t of EON with

$$\mathrm{EON} \vdash |\varphi|(t)$$

hence

$$\mathrm{EON} \vdash \forall x[x = 0 \rightarrow (\lambda x \cdot t)x \downarrow \ \& \ |\varphi|((\lambda x \cdot t)x)]$$

so $\{0\} \geq |\varphi|$. Thus $\{0\} \Vdash \varphi$ and, since $\{0\}$ is below every node A in the Kripke model, with $\Lambda x \cdot 0 : A \rightarrow \{0\}$, we have $A \Vdash \varphi$ for every A in $\mathbf{P}$. Therefore $\mathcal{K} \models \varphi$. ▮

On the "Degrees of Inhabitation"

In sections 2 and 3 of this paper, we constructed Kripke models corresponding to various realizability interpretations. In particular, the interpretations were found to satisfy Church's thesis and the strong computational properties found in realizability semantics. But what is the structure of such models, and what light do they shed on the truth-value structure implicit in realizability? In this section we will briefly address this issue. To begin with, we note that the models have unusual properties. They are *closed under finite suprema and infima:* The cartesian product is the supremum, and + the infimum. (This is the content of lemma 2.2 and the comments immediately following it).

Also, in these models, every set of nodes has an upper bound, since we have "fallible" or inconsistent nodes at the top. Such fallible nodes were first discussed in the papers of Läuchli [47], Veldman [93] and de Swart [86], and seem to play a fundamental role in Kripke models associated with realizability, as well as in the intuitionistic completeness theorem of Friedman, Veldman, de Swart and Troelstra (see the discussion in [89]). This suggests that one should think of Kripke models of the type studied here as being developed in an intuitionistic metatheory, where consistency of nodes is not necessarily decidable. Our proofs are fully constructive and can be seen as a kind of syntactic counterpart to a constructive completeness theorem for Kripke semantics (this is brought out in more detail in [51]). These models are perhaps best conceived as *internal Kripke models*, that is to say, as models developed within a Topos or Kripke model. Another way of looking at the existence of inconsistent nodes is in terms of tableaux proofs in the style of Nerode, Fitting or Odifreddi (see e.g., [68, 65, 18]) in which we are not always constructively able to recognize when branches are infinite and consistent or finite and closed off.

We briefly enumerate some other "strong properties" of the models in this paper: *formulas are points: they are principally generated*, that is to say, the set of nodes p forcing a formula φ is precisely the set of nodes above the single node $|\varphi|$. Furthermore this is critical: the arguments for implication do not go through unless formulas φ on the right of a forcing relation $A \Vdash \varphi$ can be moved to the other side of the $\Vdash$ symbol, in the form $A \times |\varphi|$. In addition, it is not hard to see that these models have infinitely many nonequivalent nodes. Pick any node A, and let B be a *sentence* over the language of EON independent of $EON \cup \{\exists x A(x)\}$ We cannot have $A \geq B$, since this means that for some term f

$$EON \vdash \forall x(A(x) \to fx \downarrow \ \& \ B).$$

But then, by existential elimination and arrow introduction, we have

$$EON \vdash (\exists x A(x)) \to B,$$

contradicting independence. (Similarly we cannot have $A \geq \neg B$). We can, of course iterate this (tacitly using Gödel's incompleteness theorem to supply new sentences), obtaining a B_1 not below A or B, and then a B_2, etc. One may regard the lattice-theoretic structure of these models as a kind of syntactic analogue of reducibility orderings in recursion theory. Nodes ordered by

$$A \geq B \iff \exists t \ EON \vdash \forall x(A(x) \to tx \downarrow \ \& \ B(tx))$$

constitute a structure we might call syntactic *degrees of inhabitation*, not unlike provable M-degrees (which are a stronger ordering: functions must preserve complements). A slight modification of our Kripke structures gives us a sharper picture, however. We will need a few definitions to make this precise.

Definition 3.11 *A formula over a language L extending HA or EON is called* **negative** *if it contains no disjunctions or existential quantifiers, and* **almost negative** *if it contains no disjunctions and existential quantifiers are present only next to atomic subformulas.*

Negative and formulas play a role in realizability interpretations similar to absoluteness in set theory: they are equivalent to their own realizability in a uniform way.

Definition 3.12 *A formula A is called* **self-realizing** *if there is a term $\mathcal{J}_A$ of EON such that, provably in EON,*

(i) $A(x) \to \mathcal{J}_A(x) \,\underset{\sim}{\mathrm{r}}\, A$

(ii) $(q \,\underset{\sim}{\mathrm{r}}\, A) \to A$.

The following properties of negative and almost-negative formulas are well-known. See Beeson *op.cit.* and [89] for proofs.

Lemma 3.13 *Every negative formula is self-realizing.*

Lemma 3.14 *If A is almost negative then A is equivalent to some negative formula B, provably in EON.*

Lemma 3.15 *Every realizability formula $|A|(x)$, or $x \,\underset{\sim}{\mathrm{r}}\, A$ is almost negative.*

An immediate consequence is Nelson's 1947 theorem.

Theorem 3.16 (The idempotence of realizability) *The following two formulas are equivalent in EON (or HA) :*

$$\exists x(x \underset{\sim}{\mathbf{r}} A)$$

and

$$\exists x(x \underset{\sim}{\mathbf{r}} (\exists y(y \underset{\sim}{\mathbf{r}} A))).$$

Theorem 3.17 *If B is negative and C is any formula in two free variables, then*

$$EON \vdash \forall x(B(x) \rightarrow \exists z C(x,z))$$

implies that for some closed term f

$$EON \vdash \forall x(B(x) \rightarrow fx \downarrow \ \& \ C(x, fx)).$$

Inspection of the proofs of theorems 3.9 and 2.5 shows that the Kripke models $\mathcal{K}_{EON}$, $\mathcal{K}_{HA}$ of the preceding sections, are elementarily equivalent to their *almost-negative* **reducts** $\mathcal{K}^{an}_{EON}$, $\mathcal{K}^{an}_{HA}$ given by restricting the corresponding partial orders $\langle K, \leq \rangle$ to

$$\{A : A \text{ -equivalent to an almost negative formula of } \mathcal{L}(EON) \text{ (or HA)} \}.$$

The only important points to observe in the proof are that the nodes of $\mathcal{K}$ must continue to contain the realizability formulas, which is guaranteed by lemma 3.15 above, and that the covers produced in the proofs of the existential and disjunctive cases must preserve almost-negativity, which is immediate, since all such covers were obtained by forming "fibers" $A(x) \ \& \ tx = c$ of almost negative formulas A. Since $tx = c$ is atomic, these fibers are almost negative. We leave the routine verification to the reader.

What is the significance of cutting the models down to the **a-n** reducts? Theorem 3.17 provides the key. For almost negative formulas, the partial order in $\mathcal{K}$ can be easily characterized: $A \geq B$ is equivalent to the existence of a proof

$$EON \vdash (\exists x A(x)) \rightarrow (\exists y B(y)).$$

In short, our model is simply the *Lindenbaum algebra of existential closures of almost negative formulas* with the order reversed, a structure we have dubbed the *provable degrees of inhabitation.* In fact, the result is not entirely surprising. Realizability semantics means *inhabiting* statements with computational *evidence*. Therefore a natural algebraic interpretation of realizability is obtained by taking as truth-values the different degrees of consistency of provable inhabitation. To be realized is to be forced by the provably inhabited formulas. Other models are obtained by relativizing to any degree that is consistent and independent over HA or EON.[7] This also points the way towards the sort of converse studied in Läuchli's [47] and the author's [51]. If we are to construe arbitrary Kripke models as abstract realizability interpretations we must arrange to imbed the Heyting algebra of truth-values generated by the Kripke model into an algebra of degrees of inhabitation.

Conclusion

We have seen that a there is a uniform *constructive* translation of formalized realizability semantics into Kripke semantics. We obtain new Kripke models for **HA** satisfying Church's Thesis, with a R.E. partial order and forcing relation. We are thus able to incorporate into Kripke models, (by iterating or gluing to other models) the features of realizability interpretations, such as term extraction, Church's Thesis, etc.

This theme is further taken up in [51, 53], where we show that realizability interpretations can be translated into *uniform Beth models* via a straightforward adaptation of the Constructive Completeness theorem due to de Swart, Veldman, Friedman and Van Dalen (see e.g. [86, 93, 89]), and where we prove a constructive analogue of the Läuchli result discussed in the introduction. A tableau proof procedure for intuitionistic logic based on realizability is studied in [53]. A straightforward extension of the models in this paper give rise to Kripke models for Martin-Löf Type theory with one universe (see [52]).

Many open questions remain in this field, in particular, the application of these methods to subrecursive computations and fragments of arithmetic, as well as to the many existing variants of realizability for higher-order logic and type theory.

References

[1] Beeson, M. J. [1985a], *Foundations of Constructive Mathematics*, Springer-Verlag, Berlin.

[2] Beeson, M. J. [1985b], "Proving programs and programming proofs", in *Logic, Methodology, and Philosophy of Science* VII, North-Holland, Amsterdam.

[3] Beth, E. W. [1947], "Semantical considerations on intuitionistic mathematics", Koninklijke Nederlandse Akademie van Wetenschappen, Proceedings of the Section of Sciences.

[4] Bishop, E. [1967], *Foundations of Constructive Analysis*, (1985 edition with D. Bridges, co-author), McGraw-Hill, New York.

[5] Boileau, A. and A. Joyal [1981], "La logique des topos", Journal of Symbolic Logic 46, 6-16.

[6] Bridges, D., & Richman, F., [1987], *Varieties of Constructive Mathematics*, London Mathematical Society Lecture Note series 97, Cambridge University Press, Cambridge.

[7] This is a syntactic or "definable" counterpart to the object of truth-values in the effective topos,which is taken to be the full power-set of N in the "real world". In this case the degrees of inhabitation are considerably richer. The higher type structure is then the relativization of these inhabitation degrees to all sets: i.e. the collection of all realizable partial equivalence relations

[7] Buss, S. [1986b], "The polynomial hierarchy and intuitionistic bounded arithmetic", 1st IEEE Symposium on Structures and Complexity, Lecture Notes in Computer Science #223, Springer-Verlag, 77-103.

[8] Constable, R. L. and Howe, D. J., [1990], "Implementing Metamathematics as an Approach to Automatic Theorem Proving", in *Formal Techniques in Artificial Intelligence*, R. Banerji, ed., North-Holland.

[9] Constable, R. L., et al [1986], *Implementing Mathematics with the NUPRL Development System*, Prentice-Hall, N.J.

[10] Constable, R. L. [1983], "Programs as proofs", Information Processing Letters, 16(3), 105-112.

[11] Coquand, T. [1990], "On the analogy between propositions and types", in: *Logic Foundations of Functional Programming*, Addison-Wesley, Reading, MA.

[12] Coquand, T. and G. Huet [1985a], "Constructions: A higher order proof system for mechanizing mathematics", EUROCAL 85, Linz, Austria.

[13] Curry, H. B.and R. Feys [1958], *Combinatory Logic*, North-Holland, Amsterdam.

[14] Dragalin, A. G. [1987], *Mathematical Intuitionism: Introduction to Proof Theory*, Translations of Mathematical Monographs 67, AMS, Providence, R. I.

[15] Feferman, S. [1979], "Constructive theories of functions and classes", in Boffa, M., D. van Dalen and K. McAloon (eds.), *Logic Colloquium '78: Proceedings of the Logic Colloquium at Mons, 1978*, pp.159-224, North-Holland, Amsterdam.

[16] Feferman, S. [1977], "Theories of Finite Type", in: Barwise, J. (ed.), *Handbook of Mathematical Logic*, pp. 913-972, North-Holland, Amsterdam.

[17] Feferman, S. [1975], "A language and axioms for explicit mathematics", in: *Algebra and Logic*, Lecture Notes in Mathematics No. 450, pp. 87-139, Springer, Berlin.

[18] Fitting, M. [1983], *Proof Methods for Modal and Intuitionistic Logics*, D. Reidel, Dordrecht, The Netherlands.

[19] Fourman, M. P. and D. S. Scott [1979], "Sheaves and logic", in: Fourman, Mulvey and Scott, (eds.), *Applications of Sheaves*, Mathematical Lecture Notes 753, pp.302-401, Springer-Verlag, Berlin.

[20] Freyd, P. and A. Scedrov [1987], "Some semantic aspects of polymorphic lambda calculus", in: *Proceedings Symposium on Logic in Computer Science*, Cornell University, June 1987, IEEE, Washington, D.C.

[21] Friedman, H. [1978], "Classically and intuitionistically provably recursive functions", in: Müller, G.H. and D. S. Scott (eds.), *Higher Set Theory, Proceedings, Oberwolfach, 1977*, pp. 21-27, Lecture Notes in Mathematics No. 669, Springer, Berlin.

[22] Friedman, H. [1973], "Some applications of Kleene's methods for intuitionistic systems", in: Mathias, A. and H. Rogers (eds.), *Cambridge Summer School in Mathematical Logic*, pp. 113-170, Lecture Notes in Mathematics No. 337, Springer, Berlin.

[23] Gabbay, D. G. [1981], *Semantical Investigations in Heyting's Intuitionistic Logic*, D. Reidel, Dordrecht, The Netherlands.

[24] Girard, J-Y., Y. Lafont and P. Taylor [1989], *Proofs and Types*, Cambridge Tracts in the Theoretical Computer Science 7, Cambridge University Press, Cambridge.

[25] Girard, J-Y. [1977], "Functional interpretation and Kripke models", in: Butts and Hintikka (eds.), *Logic, foundations of Mathematics and Computability Theory*, pp. 33-57, Reidel, Dordrecht-Holland.

[26] Girard, J-Y. [1972], "Interprétation fonctionelle et élimination de coupures de l'arithmétique d'ordre supérieur," Thèse de Doctorat d'état, Université Paris VII.

[27] Gödel, K. [1958], "Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes", Dialectica 12, 280-287.

[28] Goodman, N. [1978], "Relativized realizability in intuitionistic arithmetic of all finite types", Journal of Symbolic Logic 43, 23-44.

[29] Grayson, R. J. [1983], "Forcing in intuitionistic systems without power set", Journal of Symbolic Logic 48, 670-682.

[30] Grayson, R. J. [1984], "Heyting-valued semantics", in: F. Lolli, G. Longo and G. Marcja (eds.), *Logic Colloquium* '82, North-Holland, Amsterdam.

[31] Harper, R., MacQueen, D., Milner, R., [1986], *Standard ML*, Technical Report, LFCS-86-2, Laboratory for the Foundations of Computer Science, University of Edinburgh.

[32] Hayashi, S.and H. Nakano [1989], *PX: A Computational Logic*, The MIT Press, Cambridge.

[33] Heyting, A. [1956], *Intuitionism: an Introduction*, North-Holland, Amsterdam.

[34] Howard, W. A. [1980], "The Formulae-as-types notion of construction", in: Seldin, J.P. and J. R. Hindley (eds.), *To H. B. Curry: Essays in Combinatory Logic, Lambda Calculus and Formalism*, Academic Press, New York, 479-490.

[35] Huet, G., [1990], ed. *Logic Foundations of Functional Programming Languages*, proceedings of a symposium from the Year of Programming at the University of Texas at Austin, 1986, Addison-Wesley, Reading, MA.

[36] Hyland, J. M. E. [1987a], "A small complete category", Lecture delivered at the Conference "Church's Thesis after 50 years", to appear in Annals of Pure and Applied Logic.

[37] Hyland, J. M. E. [1982], "The effective topos". in: Troelstra, A. S. and D. S. van Dalen (eds.), *L.E.J. Brouwer Centenary Symposium*, North-Holland, Amsterdam.

[38] Kleene, S. C. [1973], "Realizability: A retrospective survey", in: Mathias, A. R. D. (ed.), Cambridge Summer School in Mathematical Logic, Cambridge, Aug. 1971, Lecture Notes in Mathematics 337, Springer-Verlag, Berlin.

[39] Kleene, S. C. [1952], *Introduction to Metamathematics*, North-Holland (1971 edition), Amsterdam.

[40] Kleene, S. C. [1945], "On the interpretation of intuitionistic number theory", Journal of Symbolic Logic 10, 109-124.

[41] Kowalski, R. A. [1974], "Predicate logic as a programming language", Information Processing 74, North-Holland, Stockholm, 569-574.

[42] Kreisel, G. and A. S. Troelstra [1970], "Formal systems for some branches of intuitionistic analysis", Annals of Mathematical Logic 1, 229-387.

[43] Kreisel, G. [1951], "On the interpretation of non-finitist proofs I", Journal of Symbolic Logic.

[44] Kreisel, G. [1959], "Interpretation of analysis by means of constructive functionals of finite type", in: Heyting, A. (ed.), *Constructivity in Mathematics*, Proceedings of the Colloquium at Amsterdam, 1957, North-Holland, Amsterdam.

[45] Kripke, S. [1965], "Semantical analysis of intuitionistic logic I", in: Crossley, J. N. and M. Dummett (eds.), *Formal Systems and Recursive Functions*, Proceedings of the Eighth Logic Colloquium, Oxford, 1963, North-Holland, Amsterdam, 92-130.

[46] Lambek, J. and P. J. Scott [1986], *Introduction to higher order categorical logic*, Cambridge Studies in Advanced Mathematics 7, Cambridge.

[47] Läuchli, H. [1970], "An abstract notion of realizability for which predicate calculus is complete", in: Myhill, J., A. Kino, and R. E. Vesley (eds.), *Intuitionism and Proof Theory*, North-Holland, Amsterdam, 227-234.

[48] Lawvere, W. [1971], "Quantifiers and sheaves", in: *Actes du Congres International des Mathematiciens, 1-10*, Septembre 1970, Nice, France, Vol. I., pp. 329-334, Gauthier-Villars, Paris.

[49] Leivant, D. [1983b], "Reasoning about functional programs and complexity classes associated with type disciplines," in *24th Annual Symposium on Foundations of Computer Science*, 460-496.

[50] Lipton, J., [1990], "Relating Kripke Models and Realizability", Ph. D. dissertation, Cornell University.

[51] Lipton, J., [1990], "Constructive Kripke Semantics and Realizability", to appear in the proceedings of the *Logic for Computer Science* conference held at the Math. Sci. Research Institute, Berkeley, Nov. 1989.

[52] Lipton, J. [1990], "Some Kripke Models for 'one-universe' Martin Löf Type Theory", Technical Report, Department of Computer Science, Cornell University.

[53] Lipton, J.,[1990], *Constructive Kripke Models, Realizability and Deduction*, Technical report, Cornell University, to appear.

[54] Lloyd, J. W., [1987], *Foundations of Logic Programming*, second ed., Springer-Verlag.

[55] Longo, G. and E. Moggi [1988], "Constructive natural deduction and its modest interpretation", CMU Report CS-88-131, Lecture delivered at the workshop on "The semantics of natural and programming languages", Stanford, March 1987, M.I.T. Press.

[56] McCarty, D. C. [1986], "Realizability and recursive set theory", Annals of Pure and Applied Logic 32, 11-194.

[57] Makkai, M. and G. Reyes [1977], "First order categorical logic", Lecture Notes in Mathematics 611, Springer-Verlag, Berlin.

[58] Martin-Löf, P. [1982], "Constructive Mathematics and Computer Programming", in Logic, Methodology and Philosophy of Science IV, North Holland, Amsterdam.

[59] Martin-Löf, P. [1984], *Intuitionistic Type Theory*, Studies in Proof Theory Lecture Notes, BIBLIOPOLIS, Napoli, Italy.

[60] Metakides, G. and A. Nerode [1977], "Recursively enumerable vector spaces", Annals of Mathematical Logic 11, 147-171.

[61] Mitchell, J. and E. Moggi [1987], "Kripke-style models for typed lambda calculus", *Proceedings from Symposium on Logic in Computer Science*, Cornell University, June 1987, IEEE, Washington, D.C..

[62] Moggi, E. [1986], "On the PERS model", types E-mail communication, manuscript.

[63] Mulry, P. S. [1980], "The topos of recursive sets", Doctoral Dissertation, SUNY-Buffalo.

[64] Murthy, C., [1990], "Extracting Constructive Content from Classical Proofs: Compilation and the Foundations of Mathematics", Ph. D. dissertation, Cornell University.

[65] Nerode, A., and Odifreddi, Piergiorgio, [1990]

[66] Nerode, A., J. B. Remmel and A. Scedrov [1989], "Polynomially graded logic I: A graded version of System T", MSI Technical Report 89-21, Cornell University, Ithaca, N.Y.

[67] Nerode, A. [1989a], "Lectures on Intuitionistic Logic II", MSI Technical Report 89-56, Cornell University, Ithaca, N.Y.

[68] Nerode, A. [1989b], "Some lecures on Intuitionistic Logic I", Proceedings on Summer School on Logical Foundations of Computer Science, CIME, Montecatini, 1988, Lecture Notes in Mathematics, Springer-Verlag, Berlin.

[69] Nerode, A. [1960], "General topology and partial recursive functionals", Summaries of the talks presented at the Summer Institute for Symbolic Logic, Cornell University, 1957, Second Edition, Institute for Defence Analysis, Princeton, N.J.

[70] Odifreddi, G. [1989], *Classical Recursion Theory*, North-Holland, Amsterdam.

[71] Pitts, A. [1987], "Polymorphism is set theoretic, constructively", Symposium on Category Theory and Computer Science, SLNCS 283, Edinburgh.

[72] Plotkin, G., [1980], "Lambda definability in the full type hierarchy", in : Seldin, J.P. and J. R. Hindley (eds.), *To H.B. Curry: Essays in Combinatory Logic, Lambda Calculus and Formalism*, Academic Press, New York.

[73] Prawitz D. [1965], *Natural Deduction*, Almquist and Wiskell, Stockholm.

[74] Rasiowa, H. and R. Sikorski [1963], *Mathematics of Metamathematics*, Monographie Matematyczne, T. 41, Warsawa.

[75] Reyes, G.E., [1974], "From Sheaves to Logic", in: A. Daigneault, ed. *Studies in Algebraic Logic*, MAA.

[76] Reynolds, J. C. [1990], "An introduction to the polymorphic lambda calculus", in: *Logical Foundations of Functional Programming*, Addison-Wesley, Reading, MA.

[77] Robinson, E. [1988], "How complete is PER?", Technical Report 88-229, Queen's University, Department of Computing and Information Science.

[78] Robinson, J. A. [1965], "A machine-oriented logic based on the resolution principle", Journal of the ACM, V.12, 23-41.

[79] Rosolini, G., [1986], "Continuity and Effectiveness in Topoi", Ph. D. Dissertation, Oxford Univ.; also, report CMU-CS-86-123, department of Computer Science, Carnegie-Mellon University.

[80] Scedrov, A. [1990], "Recursive realizability semantics for the calculus of constructions", in: *Logical Foundations of Functional Programming*, Addison-Wesley, Reading, MA.

[81] Scedrov, A. [1984], *Forcing and Classifying Topoi*, Memoirs of the American Mathematical Society 295, Providence , R.I.

[82] Scott, D. S. [1975], "Data types as lattices", in: G. H. Muller, A. Oberschelp and K. Potthof, eds., *ISILC Logic Conference*, Springer-Verlag, Berlin, 579-651, also in: SIAM Journal, Comput. 5 (1976), 522-587.

[83] Shoenfield, [1967], *Mathematical Logic*, Addison-Wesley.

[84] Smorynski, C. A. [1973], "Applications of Kripke models", in: A. S. Troelstra (ed.), *Metamathematical Investigations of Intuitionistic Arithmetic and Analysis*, Lecture Notes in Math. 344, Springer-Verlag, Amsterdam, 324-391.

[85] Stenlund, S. [1972], *Combinators, λ-terms and proof theory*, D. Reidel, Dordrecht, The Netherlands.

[86] Swart, H. C. M. de [1976], "Another intuitionistic completeness proof", Journal of Symbolic Logic 41, 644-662.

[87] Szabo, M. E. [1978], "Algebra of proofs", *Studies in Logic and the Foundations of Mathematics Vol. 88*, North-Holland, Amsterdam.

[88] Tarski, A. [1956], *Sentenial Calculus and Topology in Logic, Semantics, and Metamathematics*, Clarendon Press, Oxford.

[89] Troelstra, A. S. and D. van Dalen [1988], *Constructivism in Mathematics: An Introduction*, Vol. II, Studies in Logic and the Foundations of Mathematics, Vol. 123, North-Holland, Amsterdam.

[90] Troelstra, A. S. [1973], *Metamathematical Investigation of Intuitionistic Arithmetic and Analysis*, Mathematical Lecture Notes 344, Springer-Verlag, Berlin.

[91] Turner, D. A., [1985], *Miranda: a non-strict functional language with polymorphic types*, LNCS, vol. 201, Springer-Verlag.

[92] van Dalen, D. [1986], "Intuitionistic logic", in *The Handbook of Philosophical Logic*, Vol. III, 225-339, D. Reidel, Dordrecht.

[93] Veldman, W. [1976], "An intuitionistic completeness theorem for intuitionistic predicate logic", Journal of Symbolic Logic 41, 159-166.