

Understanding and Using Spector's Bar Recursive Interpretation of Classical Analysis

Paulo Oliva*

Department of Computer Science
Queen Mary, University of London
Mile End Road
London E1 4NS, UK
pbo@dcs.qmul.ac.uk

Abstract. This note reexamines Spector's remarkable computational interpretation of full classical analysis. Spector's interpretation makes use of a rather abstruse recursion schema, so-called bar recursion, used to interpret the double negation shift DNS. In this note bar recursion is presented as a generalisation of a simpler primitive recursive functional needed for the interpretation of a finite (intuitionistic) version of DNS. I will also present two concrete applications of bar recursion in the extraction of programs from proofs of $\forall\exists$ -theorems in classical analysis.

1 Introduction

In [3], Gödel presents an interpretation of first-order intuitionistic arithmetic \mathbf{HA} into a quantifier-free calculus of higher-order primitive recursive functionals. Gödel's interpretation, nowadays called *Dialectica interpretation*, can be naturally extended to an interpretation of \mathbf{HA}^ω , intuitionistic arithmetic in the language of finite types (see [7]). Moreover, via the negative translation (e.g. Kuroda's [4]) of classical into intuitionistic logic, Dialectica interpretation is also applicable to \mathbf{PA}^ω , classical arithmetic in the language of finite types.

One of the nicest features of Gödel's interpretation is that (in the intuitionistic context) it trivialises both Markov principle

$$\text{MP} \quad : \quad \neg\forall n^{\mathbb{N}} A_{\text{qf}}(n) \rightarrow \exists n \neg A_{\text{qf}}(n),$$

$A_{\text{qf}}(n)$ a decidable formula, and the axiom of choice

$$\text{AC} \quad : \quad \forall x^\rho \exists y^\tau A(x, y) \rightarrow \exists f^{\rho \rightarrow \tau} \forall x A(x, fx).$$

Given that the interpretation strengthens $\forall\exists$ -formulas¹, this gives a simple proof that $\mathbf{HA}^\omega + \text{MP} + \text{AC}$ is $\forall\exists$ -conservative over \mathbf{HA}^ω . The same cannot be said about $\mathbf{PA}^\omega + \text{AC}$, however, since the Dialectica interpretation of $\mathbf{PA}^\omega + \text{AC}$ will have to interpret the negative translation of AC, which is not necessarily weaker than AC itself. In fact, \mathbf{PA}^ω extended with a weaker form of choice

* This research has been supported by the UK EPSRC grant GR/S31242/01.

¹ If A is a $\forall\exists$ -formula then the interpretation of A implies A , provably in \mathbf{HA}^ω .

$$\text{cAC} \quad : \quad \forall n^{\mathbb{N}} \exists y^{\sigma} A(n, y) \rightarrow \exists f^{\mathbb{N} \rightarrow \sigma} \forall n A(n, fn)$$

so-called *countable choice*, is already strong enough to prove the *comprehension schema*

$$\text{CA} \quad : \quad \exists f^{\mathbb{N} \rightarrow \sigma} \forall n^{\mathbb{N}} (fn = 0 \leftrightarrow A(n)).$$

Simply apply cAC to the classically valid statement $\forall n \exists k (k = 0 \leftrightarrow A(n))$. This shows that AC as stated above is innocuous in an intuitionistic context, but not in a classical one.

Therefore, we choose $\mathbf{PA}^{\omega} + \text{cAC}$ as the formal system of *classical analysis*. Note that $\mathbf{PA}^{\omega} + \text{cAC}$ (and even $\mathbf{PA}^{\omega} + \text{CA}$) clearly contains second-order arithmetic \mathbf{PA}^2 , if we represent sets via their characteristic functions. In [6], Spector observes that classical analysis has a negative translation into $\mathbf{HA}^{\omega} + \text{cAC} + \text{DNS}$, where

$$\text{DNS} \quad : \quad \forall n^{\mathbb{N}} \neg \neg A(n) \rightarrow \neg \neg \forall n A(n)$$

is called the *double negation shift*. This is the case since, after classical logic is eliminated via the negative translation, one is left with the system $\mathbf{HA}^{\omega} + \text{cAC}^N$ to be interpreted, where

$$\text{cAC}^N \quad : \quad \forall n \neg \neg \exists y A(n, y) \rightarrow \neg \neg \exists f \forall n A(n, fn)$$

is the schema sufficient for proving the negative translation of each cAC instance. On the other hand, cAC^N follows (in \mathbf{HA}^{ω}) from $\text{cAC} + \text{DNS}$.

It is worth noting that the double negation shift is intuitionistically equivalent to the double negation of the *generalised Markov principle*

$$\text{GMP} \quad : \quad \neg \forall n^{\mathbb{N}} A(n) \rightarrow \exists n \neg A(n)$$

for arbitrary formulas $A(n)$, i.e. $\mathbf{HA}^{\omega} \vdash \text{DNS} \leftrightarrow \neg \neg \text{GMP}$.

This note concerns Spector's [6] Dialectica interpretation of DNS (and hence, full classical analysis). Spector's interpretation makes use of a new recursion schema, called *bar recursion*. Our aim is to present bar recursion as a generalisation of a primitive recursive functional needed for the interpretation of $\bigwedge_{i=0}^n \neg \neg A_i \rightarrow \neg \neg \bigwedge_{i=0}^n A_i$, an intuitionistically valid special case of DNS . We will also present two concrete applications of bar recursion in the analysis of two $\forall \exists$ -theorems in classical analysis.

1.1 Dialectica interpretation

In this section we will shortly recall Gödel's Dialectica interpretation, using the unifying notation of [5]. Sequences of variables x_0, \dots, x_n will be abbreviated as \mathbf{x} , and a functional application of two sequences of variables $\mathbf{x}\mathbf{y}$ is a shorthand for $x_0\mathbf{y}, \dots, x_n\mathbf{y}$. In the Dialectica interpretation, each formula A of \mathbf{HA}^{ω} is associate with a new (quantifier-free) formula $|A|_{\mathbf{y}}^{\mathbf{x}}$, with two distinguished tuples of free-variables \mathbf{x} (so-called *witness variables*) and \mathbf{y} (so-called *counter-example*

variables), other than the variables already free in A . Intuitively, the formula A is interpreted by the formula $\exists \mathbf{x} \forall \mathbf{y} |A|_{\mathbf{y}}^{\mathbf{x}}$.

For an atomic formula P we set $|P| := P$, with empty tuples of both witness and counter-example variables. Assume we have already defined $|A|_{\mathbf{y}}^{\mathbf{x}}$ and $|B|_{\mathbf{w}}^{\mathbf{v}}$, we define

$$\begin{aligned} |A \wedge B|_{\mathbf{y}, \mathbf{w}}^{\mathbf{x}, \mathbf{v}} &::= |A|_{\mathbf{y}}^{\mathbf{x}} \wedge |B|_{\mathbf{w}}^{\mathbf{v}} \\ |A \vee B|_{\mathbf{y}, \mathbf{w}}^{\mathbf{x}, \mathbf{v}, n} &::= (n = 0 \wedge |A|_{\mathbf{y}}^{\mathbf{x}}) \vee (n \neq 0 \wedge |B|_{\mathbf{w}}^{\mathbf{v}}) \\ |A \rightarrow B|_{\mathbf{x}, \mathbf{w}}^{\mathbf{f}, \mathbf{g}} &::= |A|_{\mathbf{g} \mathbf{x} \mathbf{w}}^{\mathbf{x}} \rightarrow |B|_{\mathbf{w}}^{\mathbf{f} \mathbf{x}} \\ |\forall z A(z)|_{\mathbf{y}, z}^{\mathbf{f}} &::= |A(z)|_{\mathbf{y}}^{\mathbf{f} z} \\ |\exists z A(z)|_{\mathbf{y}}^{\mathbf{x}, z} &::= |A(z)|_{\mathbf{y}}^{\mathbf{x}} \end{aligned}$$

For instance, the interpretation of the formula $A \equiv \forall n \exists m (m \geq n \wedge P(m))$ has Dialectica interpretation $f n \geq n \wedge P(f n)$, where f is the witness variable and n is the counter-example variable. In our notation $|A|_n^f \equiv f n \geq n \wedge P(f n)$.

The soundness theorem for the Dialectica interpretation guarantees that if \mathbf{HA}^ω proves a formula A , then there exists a sequence of terms \mathbf{t} , over the free-variables of A , such that (the quantifier-free fragment of) \mathbf{HA}^ω proves $|A|_{\mathbf{y}}^{\mathbf{t}}$. In order to extend Gödel's Dialectica interpretation of Heyting arithmetic with a new principle B , we must be able to produce a witnessing term \mathbf{s} for the interpretation of B , i.e. we must be able to show $\forall \mathbf{w} |B|_{\mathbf{w}}^{\mathbf{s}}$.

In this paper I will be using Kuroda's [4] negative translation of classical into intuitionistic logic, which simply places double negation after each universal quantifiers and in front of the whole formula.

Notation. We use \mathbb{N} for the basic finite type, and $\rho \rightarrow \tau$ for functional types. For convenience we will also use sequence types, i.e. ρ^* denotes the type of sequences of elements of type ρ . The length of a finite sequence is represented as $|s|$, while $\langle x_0, \dots, x_n \rangle$ denotes the finite sequence of length $n + 1$ with elements x_0, \dots, x_n . The concatenation of two finite sequences, a finite sequence and an element, or a finite sequence with an infinite sequence will all be denote via the $*$ construction, i.e. $s * t$, $s * x$ and $s * f$, respectively. Whenever possible we omit parenthesis in functional application, i.e. fxy stands for $f(x, y)$. The following primitive recursive constructions will be used. If $s : \rho^*$ then

- $\hat{s}(i)$ is s_i if $i < |s|$ and 0^ρ otherwise
- if $i \leq |s|$ then $(\overline{s, i})(n)$ is s_n when $n < i$ and 0^ρ otherwise
- if $i \leq |s|$ then $\bar{s}i$ is $\langle s_0, \dots, s_{i-1} \rangle$.

2 Double Negation Shift

The principle DNS can be viewed as the infinite counterpart of the following intuitionistic schema

$$\text{fDNS} \quad : \quad \bigwedge_{i=0}^n \neg\neg A_i \rightarrow \neg\neg \bigwedge_{i=0}^n A_i$$

which we will refer to as the *finite double negation shift*. In order to understand Spector's interpretation of DNS, we will first look at the Dialectica interpretation of fDNS. A possible proof of fDNS in minimal logic is shown in Table 1.

$$\begin{array}{c}
[A_0]_{\alpha_0} \quad \dots \quad [A_n]_{\alpha_n} \\
\hline
\bigwedge_{i=0}^n A_i \quad \quad \quad [\neg \bigwedge_{i=0}^n A_i]_{\beta} \\
\hline
\frac{\perp}{\neg A_0} (\alpha_0) \quad \quad \quad \frac{[\bigwedge_{i=0}^n \neg\neg A_i]_{\gamma}}{\neg\neg A_0} \\
\hline
\frac{\perp}{\neg A_1} (\alpha_1) \quad \quad \quad \frac{[\bigwedge_{i=0}^n \neg\neg A_i]_{\gamma}}{\neg\neg A_1} \\
\hline
\frac{\perp}{\neg A_n} (\alpha_n) \quad \quad \quad \frac{[\bigwedge_{i=0}^n \neg\neg A_i]_{\gamma}}{\neg\neg A_n} \\
\hline
\frac{\perp}{\neg\neg \bigwedge_{i=0}^n A_i} (\beta) \\
\hline
\frac{\neg\neg \bigwedge_{i=0}^n A_i}{\bigwedge_{i=0}^n \neg\neg A_i \rightarrow \neg\neg \bigwedge_{i=0}^n A_i} (\gamma)
\end{array}$$

Table 1. Proof of fDNS

Assume that each A_i has Dialectica interpretation $|A_i|_{y_i}^{x_i}$ (although x_i and y_i are potentially tuples of functionals, I will write them as single functionals for simplicity). Given that $\neg A$ is a shorthand for $A \rightarrow \perp$, the Dialectica interpretation of each $\neg\neg A_i$ is

$$|\neg\neg A_i|_{g_i}^{\Phi_i} \equiv \neg\neg |A_i|_{g_i(\Phi_i g_i)}^{\Phi_i g_i}$$

while $\neg\neg \bigwedge_{i=0}^n A_i$ has Dialectica interpretation

$$|\neg\neg \bigwedge_{i=0}^n A_i|_{\mathbf{\Delta}}^{\mathbf{x}} \equiv \neg\neg \bigwedge_{i=0}^n |A_i|_{\Delta_i(\mathbf{x} \mathbf{\Delta})}^{x_i \mathbf{\Delta}}.$$

By the interpretation of implication, the interpretation of fDNS asks for sequences of functionals \mathbf{g} and \mathbf{x} satisfying

$$|\text{fDNS}|_{\mathbf{v}}^{\mathbf{g}, \mathbf{x}} \equiv \bigwedge_{i=0}^n \neg\neg |A_i|_{g_i(\mathbf{v}, \Phi_i(g_i \mathbf{v}))}^{\Phi_i(g_i \mathbf{v})} \rightarrow \neg\neg \bigwedge_{i=0}^n |A_i|_{\Delta_i(\mathbf{x} \mathbf{v})}^{x_i \mathbf{v}}$$

where $v \equiv \Delta, \Phi$. Such functionals can be produced if we can solve (on the parameters Δ, Φ) the following set of equations

$$\begin{aligned} x_i &\stackrel{\rho_i}{=} \Phi_i g_i \\ g_i(x_i) &\stackrel{\tau_i}{=} \Delta_i(\mathbf{x}) \end{aligned} \quad (1)$$

for $i \in \{0, \dots, n\}$, where the parameters Δ, Φ have been omitted for clarity. There is an apparent circularity since each x_i needs the definition of g_i , and each g_i seems to need the previous definition of all x_0, \dots, x_n . Consider the case where $n = 2$, i.e.

$$\begin{aligned} x_0 &= \Phi_0 g_0 \\ x_1 &= \Phi_1 g_1 \\ x_2 &= \Phi_2 g_2 \\ g_0(x_0) &= \Delta_0(x_0, x_1, x_2) \\ g_1(x_1) &= \Delta_1(x_0, x_1, x_2) \\ g_2(x_2) &= \Delta_2(x_0, x_1, x_2) \end{aligned} \quad (2)$$

The analysis of the proof gives us the following solution. First define x_2 and g_2 assuming that x_0, x_1 have already been defined:

$$\begin{aligned} G_2[x_0, x_1] &:= \lambda x. \Delta_2(x_0, x_1, x) \\ X_2[x_0, x_1] &:= \Phi_2(G_2[x_0, x_1]) \end{aligned}$$

Next, we give a parametrised definition of x_1 and g_1 , assuming only that x_0 has already been defined, as

$$\begin{aligned} G_1[x_0] &:= \lambda x. \Delta_1(x_0, x, X_2[x_0, x]) \\ X_1[x_0] &:= \Phi_1(G_1[x_0]) \end{aligned}$$

We then define x_0 and g_0 , using the parametrised definitions of x_1 and x_2 , as

$$\begin{aligned} g_0 &:= \lambda x. \Delta_0(x, X_1[x], X_2[x, X_1[x]]) \\ x_0 &:= \Phi_0 g_0 \end{aligned}$$

Finally, x_1, x_2 and g_1, g_2 can be defined, using the definition of x_0 , as

$$\begin{aligned} g_1 &:= G_1[x_0] \\ x_1 &:= \Phi_1 g_1 \\ g_2 &:= G_2[x_0, x_1] \\ x_2 &:= \Phi_2 g_2 \end{aligned}$$

The cunning solution above breaks the apparent circularity of (2) by making use of the fact that the definition of g_i does not require x_i to be already defined, although g_i does require all other x_j , for $j \neq i$ to be defined.

The general algorithm we get from the proof shown in Table 1 (following Dialectica interpretation) can be succinctly presented as follows.

Let us define a master functional fB (*finite bar recursion*), which is supposed to return the suffix $\langle x_i, \dots, x_n \rangle$ assuming we already have an approximate solution $\langle x_0, \dots, x_{i-1} \rangle$, as follows:

$$\text{fB}(\langle x_0, \dots, x_{i-1} \rangle) := \begin{cases} \langle \rangle & n < i \\ X_{\langle x_0, \dots, x_{i-1} \rangle} * \text{fB}(\langle x_0, \dots, x_{i-1}, X_{\langle x_0, \dots, x_{i-1} \rangle} \rangle) & n \geq i \end{cases}$$

where

$$\begin{aligned} G_{\langle x_0, \dots, x_{i-1} \rangle} &:= \lambda x. \Delta_i(\langle x_0, \dots, x_{i-1}, x \rangle * \text{fB}(\langle x_0, \dots, x_{i-1}, x \rangle)) \\ X_{\langle x_0, \dots, x_{i-1} \rangle} &:= \Phi_i G_{\langle x_0, \dots, x_{i-1} \rangle}. \end{aligned}$$

We can then take $\langle x_0, \dots, x_n \rangle := \text{fB}(\langle \rangle)$ and $g_i := G_{\langle x_0, \dots, x_{i-1} \rangle}$. One can show by induction on i that if $\text{fB}(\langle \rangle) = \langle x_0, \dots, x_n \rangle$ then $\text{fB}(\langle x_0, \dots, x_{i-1} \rangle) = \langle x_i, \dots, x_n \rangle$. This implies that $x_i = X_{\langle x_0, \dots, x_{i-1} \rangle}$, i.e. $x_i = \Phi_i g_{\langle x_0, \dots, x_{i-1} \rangle} = \Phi_i g_i$. Moreover, $g_i(x_i) = \Delta_i(\mathbf{x})$ since

$$\langle x_0, \dots, x_i \rangle * \text{fB}(\langle x_0, \dots, x_i \rangle) = \langle x_0, \dots, x_n \rangle.$$

2.1 Example

We present here a concrete example of a proof whose Dialectica interpretation can be solved using the finite bar recursion fB . Let $m \in S$ be a shorthand for $S m = 0$, $\text{INF}(S)$ be the predicate stating that the set S is infinite, i.e. $\forall n \exists m \geq n (m \in S)$, and \bar{S} denote the set complement of S . We will also use $\text{INF}_0(S, g, n)$ as an abbreviation for the quantifier-free part of the Skolemised $\text{INF}(S)$, namely $(gn \geq n) \wedge (gn \in S)$. Consider the following simple \mathbf{PA}^ω -theorem.

Theorem 1. *For all $r \in \mathbb{N}$ and r -partition P of \mathbb{N} (namely, $P : \mathbb{N} \rightarrow \{0, \dots, r\}$) there exist a P -homogeneous infinite subset $H \subseteq \mathbb{N}$, i.e. $\exists k \forall n \in H (Pn = k)$.*

Let the P -homogeneous sets H_k be defined as $n \in H_k$ whenever $Pn = k$, for $k \leq r$. Therefore, Theorem 1 can be written formally as:

$$\forall r, P^{\mathbb{N} \rightarrow \{0, \dots, r\}} \exists k \leq r \text{INF}(H_k).$$

After the negative translation, the Dialectica interpretation asks for witnessing functionals for the following:

$$\forall r, P^{\mathbb{N} \rightarrow \{0, \dots, r\}}, \Phi \exists g, k \leq r \text{INF}_0(H_k, g, \Phi k g).$$

We will produce for each $k \leq r$ a function g_k (on parameters r, P and Φ) such that for some $0 \leq k \leq r$, we have $\text{INF}_0(H_k, g_k, \Phi k g_k)$. It is easy to see that if those functions satisfy the following set of equations then we are done:

$$\begin{aligned} x_k &= \Phi k g_k \\ g_k(x_k) &= \max\{x_0, \dots, x_r\} \end{aligned} \tag{3}$$

since, for all $0 \leq k \leq r$, all $g_k(\Phi k g_k)$ would have the same value, namely $\max\{x_0, \dots, x_r\}$. Moreover, $g_k(x_k) \geq x_k$. Therefore, once those have been computed, we can produce k and g as

$$k, g := \begin{cases} 0, g_0 & \text{if } \max\{x_0, \dots, x_r\} \in H_0 \\ \dots & \\ r, g_r & \text{if } \max\{x_0, \dots, x_r\} \in H_r \end{cases}$$

But (3) is exactly the kind of system of equations that can be solved via finite bar recursion, as shown above.

2.2 Spector's interpretation

In this section we present Spector's [6] bar recursive Dialectica interpretation of DNS. Let $|A(n)|_y^x$ be the interpretation of $A(n)$. The Dialectica interpretation of the double negation shift DNS is as follows

$$|\text{DNS}|_{\Phi, \Psi}^{n, g, f} \equiv \neg\neg |A(n)|_{g(\Phi n g)}^{\Phi n g} \rightarrow \neg\neg |A(\Psi f)|_{\Delta f}^{f(\Psi f)}$$

where, for clarity, we have omitted the parameters Φ, Ψ of the witnessing functionals n, g, f . Such functionals can be produced if we can solve the following generalisation of (1)

$$\begin{aligned} n &\stackrel{\mathbb{N}}{=} \Psi f \\ f n &\stackrel{\rho}{=} \Phi n g \\ g(f n) &\stackrel{\tau}{=} \Delta f \end{aligned} \tag{4}$$

Therefore, instead of a finite tuple $\langle x_1, \dots, x_n \rangle$, we must produce an infinite sequence f . Intuitively, if we were to assume the continuity of Ψ we would only need to produce a finite initial segment of f . Based on this intuition, Spector then solves the following (more general) problem

$$\begin{aligned} |s| &> \Psi \hat{s} \\ s_i &\stackrel{\rho}{=} \Phi i g_i \\ g_i s_i &\stackrel{\tau}{=} \Delta \hat{s} \end{aligned} \tag{5}$$

which asks for a finite sequence s and a sequence of g_i , for all $i \leq \Psi \hat{s}$. Given a sequence s and a family $(g_i)_{i \leq \Psi \hat{s}}$ satisfying (5), we can take $f := \hat{s}$, $n := \Psi f$ and $g := g_n$, in order to solve (4).

Problem (5) is almost the same as (1) of Section 2, except that we do not know the required length of the sequence s . The only thing we have is a lower bound $\Psi \hat{s}$ on that length which depends on s itself! We must somehow be able to produce as long a sequence s as it takes to satisfy $|s| > \Psi \hat{s}$. Let $\mathbf{B}(s)$ be defined recursively as follows

$$\mathbf{B}(s) := \begin{cases} \langle \rangle & \Psi \hat{s} < |s| \\ X_s * \mathbf{B}(s * X_s) & \text{otherwise} \end{cases}$$

where

$$g_s := \lambda x. \Delta(s * x * \mathbf{B}(s * x) * (\lambda n. 0^\rho))$$

$$X_s := \Phi(|s|, g_s)$$

Lemma 1. *Let $s := \mathbf{B}(\langle \rangle)$. The following holds for all $0 \leq i \leq |s|$*

- (a) $s = (\bar{s}i) * \mathbf{B}(\bar{s}i)$
- (b) $\Psi(\bar{s}, i) \geq i$, if $i < |s|$
- (c) $\Psi(\bar{s}, i) < i$, if $i = |s|$ (i.e. $\Psi \hat{s} < |s|$)
- (d) $s_i = X_{\bar{s}i}$, if $i < |s|$

Proof. Points (b) and (c) follow easily from (a), given that $\mathbf{B}(\bar{s}i) = \langle \rangle$ if and only if $\Psi(\bar{s}, i) < i$. Point (d) follows from (a) and (b), simply by the definition of \mathbf{B} . We prove (a) by induction on i . If $i = 0$ the result is trivial since $\bar{s}0 = \langle \rangle$. Assume $s = (\bar{s}i) * \mathbf{B}(\bar{s}i)$ and $i + 1 \leq |s|$. In this case $\mathbf{B}(\bar{s}i)$ cannot be the empty sequence, i.e. $\mathbf{B}(\bar{s}i) = X_{\bar{s}i} * \mathbf{B}(\bar{s}i * X_{\bar{s}i})$. By induction hypothesis, $s_i = X_{\bar{s}i}$, which implies $s = (\bar{s}(i + 1)) * \mathbf{B}(\bar{s}(i + 1))$. \square

Given the definition of $X_{(\cdot)}$ and $g_{(\cdot)}$ above, points (c) and (d) of Lemma 1 imply that $s := \mathbf{B}(\langle \rangle)$ and $g_i := g_{\bar{s}i}$ solve (5), i.e.

$$s_i = \Phi i g_{\bar{s}i} \wedge g_{\bar{s}i}(s_i) = \Delta \hat{s}$$

for all $i \leq \Psi \hat{s}$. Spector stated the following general recursion schema, so-called *bar recursion*

$$\text{BR}(\Psi, \Delta, \Phi, s^{\rho^*}) \stackrel{\tau}{=} \begin{cases} \Phi s & \text{if } \Psi \hat{s} < |s| \\ \Delta(s, \lambda x^\rho. \text{BR}(\Psi, \Delta, \Phi, s * x)) & \text{otherwise} \end{cases}$$

where ρ^* is a finite sequence of objects of type ρ and the types of Ψ, Δ and Φ can be inferred from the context.

3 Bar Recursion in Use

In this section we present two examples of how bar recursion can be used in the computational interpretation of $\forall\exists$ -theorem in classical analysis. In each case we will not go into details of how the bar recursive programs have been extracted from the proof. Our focus is on the characteristics of the final programs.

3.1 Example 1: no injection from $\mathbb{N} \rightarrow \mathbb{N}$ to \mathbb{N}

Our first example is a simple theorem which states that there can not be an injection from $\mathbb{N} \rightarrow \mathbb{N}$ to \mathbb{N} . We analyse its straightforward classical proof.

Theorem 2. $\forall \Psi^2 \exists \alpha^1, \beta^1 (\alpha \neq \beta \wedge \Psi \alpha = \Psi \beta)$.

Proof. From the axiom $\forall \Psi, k (\exists \beta (k = \Psi \beta) \rightarrow \exists \alpha (k = \Psi \alpha))$ we get by classical logic

$$\forall \Psi, k \exists \alpha (\exists \beta (k = \Psi \beta) \rightarrow k = \Psi \alpha).$$

By countable choice cAC^1 we can prove the existence of a functional f satisfying

$$\forall \Psi \exists f \forall k (\exists \beta (k = \Psi \beta) \rightarrow k = \Psi(fk)),$$

i.e. f is an enumeration of functions such that $\Psi(fk) = k$, whenever k is in the image of Ψ . Let $\delta_f := \lambda k. (f(k)(k) + 1)$ and $k := \Psi \delta_f$. We have

$$\forall \Psi \exists f (\exists \beta (\Psi \delta_f = \Psi \beta) \rightarrow \Psi \delta_f = \Psi(f(\Psi \delta_f))).$$

The premise being provable we get

$$\forall \Psi \exists f (\delta_f \neq f(\Psi \delta_f) \wedge \Psi \delta_f = \Psi(f(\Psi \delta_f))),$$

since $\delta_f \neq f(\Psi \delta_f)$ (they differ at point $\Psi \delta_f$). So we get the desired result for $\alpha := \delta_f$ and $\beta := f(\Psi \delta_f)$. \square

The reader is invited to reflect upon a *general* effective procedure for computing α and β given the functional Ψ . Obviously, if we assume that Ψ is continuous then we can simply consider the point of continuity n of Ψ on the constant zero function 0^1 . Any two distinct functions which coincide up to n will have the same value for Ψ . On the other hand, if Ψ is not continuous, but is assumed to be majorizable, then we can again consider the point of weak continuity (see Lemma 5 of [2]) n of Ψ on the constant zero function. By the pigeon hole principle, any set of $n + 1$ distinct functions which coincide with 0^1 up to n will have two with the same value. We present, however, a bar recursive solution which does not rely on the specific properties of the model for solving the problem.

The proof of Theorem 2 presented above can be formalised in the system $\text{PA}^\omega + \text{cAC}$. A bar recursive analysis of the negative translation of the proof (which we omit here for lack of space) will lead us to consider the finite sequence $t := \text{B}(\langle \rangle)$, where

$$\text{B}(s) \stackrel{1^*}{:=} \begin{cases} \langle \rangle & \Psi \delta_s < |s| \\ \delta_{\hat{r}} * \text{B}(s * \delta_{\hat{r}}) & \Psi \delta_{\hat{r}} = |s| \\ 0^1 * \text{B}(s * 0^1) & \text{otherwise,} \end{cases}$$

and $r := s * 0^1 * \text{B}(s * 0^1)$. Intuitively, we wish $t := \text{B}(\langle \rangle)$ to be an enumeration of functions t_k such that $(+)$ $\Psi t_k = k$, whenever k is in the image of Ψ , i.e.

$$\forall k < |t| (\exists \beta (\Psi \beta = k) \rightarrow \Psi t_k = k).$$

Obviously, we cannot do that effectively for all indices of t . The trick is to test at each stage k whether (+) holds for the diagonal function $\delta_{\hat{r}}$, i.e. $\Psi \delta_{\hat{r}} = k$? If that is the case then we add that function in current position k . If, however, that is not the case, we simply make sure that the whole sequence t equals r , so that $\Psi \delta_{\hat{r}} \neq k$. Therefore, the enumeration t will be such that for each $k \neq \Psi \delta_{\hat{r}}$ we have $\Psi t_k = k$. Let $n := \Psi \delta_{\hat{r}}$. By the stopping condition we have $n < |t|$. It follows that $\Psi t_n = n$, which gives us

$$\Psi \delta_f = \Psi (f(\Psi \delta_f))$$

taking $f := \hat{t}$.

3.2 Example 2: update procedures

In [1], Avigad shows that the 1-consistency of arithmetic is equivalent to the existence of solutions for a particular class of recursive equations, so-called *update equations*. In this second example we show how bar recursion can be used to compute the solution of one single update equation. The bar recursive program we present has been extracted from the proof of Lemma 2.1 in [1] (presented below).

In the following σ and τ will denote finite partial functions from \mathbb{N} to \mathbb{N} , i.e. partial functions which are defined on a finite domain. A partial function which is everywhere undefined is denoted by $\langle \ \rangle$, whereas a partial function defined only at position k (with value n) is denoted by $\langle k, n \rangle$. The finite partial functions can be viewed as finite sequences of pairs of natural numbers. For a given partial function σ , we define $\hat{\sigma}$ as the total function which is obtained from σ by defining the output to be 0 (zero) wherever σ is undefined. We say that τ extends σ , written as $\sigma \sqsubseteq \tau$, if τ is defined wherever σ is defined, and on those points they coincide in value. We denote the domain of σ as $\text{dom}(\sigma)$. For a finite partial function σ and $k, n \in \mathbb{N}$ we define the finite partial function $\sigma \oplus \langle k, n \rangle$ which maps k to n and agrees with σ otherwise, i.e.

$$(\sigma \oplus \langle k, n \rangle)(i) := \begin{cases} n & i = k \\ \sigma(i) & i \neq k \wedge i \in \text{dom}(\sigma) \\ \uparrow & \text{otherwise.} \end{cases}$$

Let $\Psi : (\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N}$ be a *continuous* functional with respect to the standard topology on the Baire space. Let Φ be also of type $(\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N}$. We say that the pair $\langle \Psi, \Phi \rangle$ forms a unary *update procedure* if whenever τ extends $\sigma \oplus \langle \Psi \hat{\sigma}, \Phi \hat{\sigma} \rangle$ and $\Psi \hat{\sigma} = \Psi \hat{\tau}$ then $\Phi \hat{\sigma} = \Phi \hat{\tau}$.

Theorem 3 (Lemma 2.1, [1]). *Every unary update procedure has a finite fixed point, i.e.*

$$\forall \Psi, \Phi (\text{Update}(\Psi, \Phi) \rightarrow \exists \sigma (\sigma = \sigma \oplus \langle \Psi \hat{\sigma}, \Phi \hat{\sigma} \rangle)).$$

Proof. Define the sequence of partial functions $\sigma_{(0)}, \sigma_{(1)}, \sigma_{(2)}, \dots$ as

$$\begin{aligned}\sigma_{(0)} &:= \langle \rangle \\ \sigma_{(i+1)} &:= \sigma_{(i)} \oplus \langle \Psi \hat{\sigma}_{(i)}, \Phi \hat{\sigma}_{(i)} \rangle.\end{aligned}$$

The fact that $\langle \Psi, \Phi \rangle$ is an update procedure implies that $\sigma_{(0)} \sqsubseteq \sigma_{(1)} \sqsubseteq \sigma_{(2)} \dots$. Let g be the partial function extending all the $\sigma_{(i)}$, that is $g := \bigcup_{i \in \mathbb{N}} \sigma_{(i)}$. The continuity of Ψ implies that for some i we have

$$\Psi \hat{g} = \Psi \hat{\sigma}_{(i)} = \Psi \hat{\sigma}_{(i+1)} = \dots$$

Since $\langle \Psi, \Phi \rangle$ forms an update pair we get

$$\sigma_{(i+1)} = \sigma_{(i)} \oplus \langle \Psi \hat{\sigma}_{(i)}, \Phi \hat{\sigma}_{(i)} \rangle = \sigma_{(i+1)} \oplus \langle \Psi \hat{\sigma}_{(i+1)}, \Phi \hat{\sigma}_{(i+1)} \rangle.$$

So, $\sigma_{(i+1)}$ is the desired fixed point. \square

Comprehension is used in the proof above in order to obtain the function \hat{g} as

$$\hat{g}(k) := \begin{cases} n & \exists i (\langle k, n \rangle \in \sigma_{(i)}) \\ 0 & \text{otherwise.} \end{cases}$$

If we could effectively build a functional f satisfying

$$\forall k (\exists i (k \in \text{dom}(\sigma_{(i)})) \rightarrow (k \in \text{dom}(\sigma_{(fk)})))$$

we could produce \hat{g} as

$$g_f(k) := \begin{cases} \sigma_{(fk)}(k) & k \in \text{dom}(\sigma_{(fk)}) \\ 0 & \text{otherwise} \end{cases}$$

since (the oracle) f tells us at which stages $i = fk$ each k is guaranteed to be defined, i.e. $k \in \text{dom}(\sigma_{(i)})$, if k does eventually become defined. The point of continuity n of Ψ on g guarantees that Ψ only looks at positions $k \leq n$. Therefore, the fixed point is surely obtained at point $\max\{fk : k \leq n\}$. Unfortunately, no such functional f can be built effectively, uniformly on all parameters. What we can build using bar recursion is an approximation for f , which as we will see, is sufficient for computing the position where the fixed point is attained. Define (uniformly on parameters ψ, ϕ) the following bar recursive functional:

$$\mathbf{B}_{\psi, \phi}(s) := \begin{cases} \langle \rangle & \psi \hat{s} < |s| \\ (\phi \hat{r}) * \mathbf{B}_{\psi, \phi}(s * (\phi \hat{r})) & |s| \in \text{dom}(\sigma_{(\phi \hat{r})}) \\ 0^1 * \mathbf{B}_{\psi, \phi}(s * 0^1) & \text{otherwise} \end{cases}$$

where $r := s * 0^1 * \mathbf{B}_{\psi, \phi}(s * 0^1)$. In the definition above, $\sigma_{(i)}$ denotes the i -th element of the inductive sequence used in the proof of Theorem 3. Intuitively, taking $t := \mathbf{B}_{\psi, \phi}(\langle \rangle)$, the procedure above makes sure that $k \in \text{dom}(\sigma_{(t_k)})$, whenever $k \in \text{dom}(\sigma_{(\phi \hat{t})})$, i.e.

$$(+) \quad \forall k < |t|(k \in \text{dom}(\sigma_{(\hat{\phi}t)}) \rightarrow k \in \text{dom}(\sigma_{(t_k)}))$$

Finally, define

$$\phi f := \max\{fk : k \leq \omega_\Psi(g_f)\} + 1$$

$$\psi f := \omega_\Psi(g_f)$$

where ω_Ψ is the modulus of pointwise continuity of Ψ and g_f has been defined above. Notice that ω_Ψ is part of the witnessing information for the assumption that Ψ is continuity (see definition of $\text{Update}(\Psi, \Phi)$), and therefore, it is one of the “inputs” for our effective procedure.

Since $\omega_\Psi(g_f)$ tells us what initial segment of g_f is necessary to compute Ψg_f , the functional ϕf computes (using f as an oracle) how far in the iteration $\{\sigma_{(i)}\}_{i \in \mathbb{N}}$ we need to go to get that much of initial segment. Let $t := \mathbf{B}_{\psi, \phi}(\langle \rangle)$ and $n := \hat{\phi}t$. Since $\psi \hat{t} < |t|$, by (+), we have

$$\forall k \leq \omega_\Psi(g_{\hat{t}})(k \in \text{dom}(\sigma_{(n)}) \rightarrow k \in \text{dom}(\sigma_{(t_k)})).$$

Moreover, by the definition of ϕ we have that $n = \hat{\phi}t > t_k$, for all $k \leq \omega_\Psi(g_{\hat{t}})$. This implies that all positions in the domain of $\sigma_{(n)}$ have been defined before, since $\langle \Psi, \Phi \rangle$ is assumed to form an update procedure. Therefore, by the continuity of Ψ , it must be the case that $\Psi(\hat{\sigma}_{(n)}) \in \text{dom}(\sigma_{(n)})$, and $\sigma_{(n-1)} = \sigma_{(n)}$.

Acknowledgements. I would like to thank the two anonymous referees for valuable comments and suggestions on the preliminary version of this paper. This research has been supported by the UK EPSRC grant GR/S31242/01.

References

1. J. Avigad. Update procedures and the 1-consistency of arithmetic. *Mathematical Logic Quarterly*, 48:3–13, 2002.
2. U. Berger and P. Oliva. Modified bar recursion and classical dependent choice. *Lecture Notes in Logic*, 20:89–107, 2005.
3. K. Gödel. Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes. *Dialectica*, 12:280–287, 1958.
4. S. Kuroda. Intuitionistische Untersuchungen der formalistischen Logik. *Nagoya Mathematical Journal*, 3:35–47, 1951.
5. P. Oliva. Unifying functional interpretations. To appear in: *Notre Dame Journal of Formal Logic*, 2006.
6. C. Spector. Provably recursive functionals of analysis: a consistency proof of analysis by an extension of principles in current intuitionistic mathematics. In F. D. E. Dekker, editor, *Recursive Function Theory: Proc. Symposia in Pure Mathematics*, volume 5, pages 1–27. American Mathematical Society, Providence, Rhode Island, 1962.
7. A. S. Troelstra (ed.). *Metamathematical Investigation of Intuitionistic Arithmetic and Analysis*, volume 344 of *Lecture Notes in Mathematics*. Springer, Berlin, 1973.