On Spector's bar-recursion.

Philipp Gerhardy
BRICS
Department of Computer Science
University of Aarhus

MAP '06, Castro Urdiales, 9.1.2006-13.1.2006.

Introduction

Informally, bar-recursion is recursion over well-founded trees and can be considered the computational equivalent to Brouwer's bar-induction.

Spector introduces bar-recursion to extend Gödel's consistency proof (via functional interpretation) for (classical) arithmetic to full classical analysis.

Interpretation of Peano arithmetic + countable/dependent choice allows program extraction from proofs in classical analysis.

Introduction

Spector uses bar-recursion(BR) to give a functional interpretation of the negative translation of countable choice(CC^N) (extended by Howard to dependent choice(DC^N)).

First, it is shown that CC^N (resp. DC^N) is intuitionistically derivable from the double-negation shift, DNS:

$$\forall x^0 \neg\neg P(x) \rightarrow \neg\neg\forall x^0 P(x), \text{ where } P \text{ is arbitrary.}$$

Then BR is used to give a functional interpretation of DNS.

Introduction

Aim of this talk: Present investigations of finite cases of the double-negation shift and their functional interpretation in order to gain better understanding of Spector's bar-recursive solution of the functional interpretation of the full double-negation shift.

What is the intuition behind Spector's solution ?

Introduction

Also, (this is very much work in progress !) compare functional interpretation and modified realizability interpretation of finite cases in order to shed light on differences between

- Spector's bar-recursion, used for functional interpretation of DNS, and
- modified bar-recursion, which is used for modified realizabilty interpretation of DNS.

Overview

- Introduction
- **Spector's solution**
- Finite cases of the double-negation shift
- Modified realizability and double-negation shift

Spector's solution

For the functional interpretation of the double-negation shift we have to consider:

$$\forall x^0 \neg\neg\exists a \forall b P(x,a,b) \to \neg\neg\forall y^0 \exists c \forall d P(y,c,d).$$

which using Gödel's $()_D$-translation is transformed further to

$$\exists x, B, C \ \forall Y, A, D$$
$$(P(x, A(x,B), B(A(x,B))) \to P(Y(C), C(Y(C)), D(C))).$$

Spector's solution

This gives rise to the following functional equations:

$$x = Y(C), \quad A(x, B) = C(Y(C)), \quad B(A(x, b)) = D(C),$$

where x is of type 0, C is of type $0 \to \rho$ and Y is of type $(0 \to \rho) \to 0$ for some type ρ.

To solve the functional interpretation, we have to define x, B and C (most importantly the sequence C) in terms of Y, A and D.

Spector's solution

Spector uses the following special form of bar-recursion

$$\varphi(x, C, n) = \begin{cases} Cn & \text{if } n < x, \\ \mathbf{0} & \text{if } n > x \wedge Y(\langle C0, \ldots C(x-1)\rangle) < x, \\ \varphi(x+1, \langle C0, \ldots C(x-1), a_0\rangle) & \text{otherwise .} \end{cases}$$

where $a_0 = G_0(x, \lambda a.\varphi(x+1, \langle C0, \ldots C(x-1), a\rangle))$ and $\varphi(x, C) = \lambda n.\varphi(x, C, n)$.

Spector's solution

This form of bar-recursion is then used to define the following functionals:

- $G_0 = \lambda m, E.A(m, \lambda a.D(E(a))$,
- $C_0 = \varphi(0, \mathcal{O})$, where $\mathcal{O}$ denotes the empty sequence,
- $E_m = \lambda a.\varphi(m+1, \langle C0, \ldots C(x-1), a \rangle)$,
- $B_m = \lambda a.D(E_m(a))$,

yielding the final solutions $C = C_0$, $x = Y(C)$ and $B = B_x$ for the functional interpretation of DNS.

Spector's solution

It is easy to verify that the given solution satisfies the functional equations for the double-negation shift.

Question: How did Spector think of this solution? What is the intuition behind the solution, especially behind the bar-recursive definition of the sequence C?

Overview

- Introduction
- Spector's solution
- **Finite cases of the double-negation shift**
- Modified realizability and double-negation shift

Finite cases of the double-negation shift

Spector remarks that if the range of the functional Y is finite, i.e. if $\forall C(Y(C) < n)$ for some n, then computing C is easy:

One computes $C(n)$ in terms of $C(0), \ldots, C(n-1)$, then $C(n-1)$ in terms of $C(0), \ldots, C(n-2)$, etc.

Intuition: if the range of the functional Y is finite one can unfold the bar-recursive definition of C to yield a primitive recursive one.

Finite cases of the double-negation shift

While the full double-negation shift does not have a functional interpretation by primitive recursive functionals, the following restricted form (k-DNS) does:

$$\forall k^0 (\forall x^0 \leq k \neg\neg\exists a \forall b P(x,a,b) \rightarrow \neg\neg\forall y \leq k \exists c \forall d P(y,c,d)).$$

For bar-recursion this corresponds to the case, where one is given a bound k on the range of Y independent of its argument C.

Finite cases of the double-negation shift

We give an informal intuitionistic proof of k-DNS by induction on k. The base case $k = 0$ this is trivial. For $k + 1$ we argue:

$$\begin{array}{ll} \forall x \leq k+1 \neg\neg P(x) & \Rightarrow \\ \forall x \leq k \neg\neg P(x) \wedge \neg\neg P(k+1) & \Rightarrow (IH) \\ \neg\neg\forall x \leq k P(x) \wedge \neg\neg P(k+1) & \Rightarrow (*) \\ \neg\neg(\forall x \leq k P(x) \wedge P(k+1)) & \Rightarrow \\ \neg\neg\forall x \leq k+1 P(x), & \end{array}$$

where (*) uses that $\neg\neg A \wedge \neg\neg B \rightarrow \neg\neg(A \wedge B)$ is provable in intuitionistic logic.

Finite cases of the double-negation shift

The crucial step is the application of $\neg\neg A \wedge \neg\neg B \to \neg\neg(A \wedge B)$.

As this is provable in intuitionistic logic, this has a functional interpretation in primitive recursive functionals.

Question: What is the relationship between (the functional interpretations of) k-DNS and full DNS?

Finite cases of the double-negation shift

As a first step, we compare the two approaches of obtaining a functional interpretation of k-DNS, via

- Spector bar-recursion, using $Y(C) \leq k$ for all C,
- induction on k and $\neg\neg A \wedge \neg\neg B \rightarrow \neg\neg(A \wedge B)$.

We carry out the comparison for the most simple, non-trivial case $k = 1$, focusing on the construction of the sequence C.

Finite cases of the double-negation shift

For bar-recursion, we must evaluate $\varphi(0, \mathcal{O})$ using the definition of the bar-recursor φ and the fact that $Y(C) \leq 1$ for all C.

Carrying out the evaluation step-by-step one arrives at the following sequence C

$$\langle A(0, \lambda a.D(\langle a, A(1, \lambda b.D(\langle a, b\rangle))\rangle)),$$
$$A(1, \lambda c.D(\langle [C(0)], c\rangle))\rangle.$$

Finite cases of the double-negation shift

Next, consider the functional interpretation of

$$(\neg\neg\exists a_0 \forall b_0 P_0(a_0, b_0) \wedge \neg\neg\exists a_1 \forall b_1 P_1(a_1, b_1)) \\ \rightarrow \neg\neg(\exists c_0 \forall d_0 P_0(c_0, d_0) \wedge \exists c_1 \forall d_1 P_1(c_1, d_1)).$$

where realizers for c_0, c_1 will depend on functionals A_0, A_1, D_0, D_1. Using functional interpretation one obtains the following functional realizers:

$$\begin{array}{rcl} C_0 & := & A_0(\lambda a.D_0(a, A_1(\lambda b.D_1(a, b)))) \\ C_1 & := & A_1(\lambda c.D_1([C_0], c)) \end{array}$$

Finite cases of the double-negation shift

Observation: The solutions are exactly the same!

First tentative conclusion: The functional interpretation of the full double-negation shift is actually the extension of the functional interpretation of $\neg\neg A \wedge \neg\neg B \rightarrow \neg\neg(A \wedge B)$ to the infinitary (but still well-founded) case.

Overview

- Introduction
- Spector's solution
- Finite cases of the double-negation shift
- **Modified realizability and double-negation shift**

Modified realizability and double-negation shift

Berardi, Bezem and Coquand(1998) and, inspired by that, Berger and Oliva(2002) give a modified realizability interpretation of the double-negation shift using so-called modified bar-recursion.

Whereas Spector's solution only requires Y to be well-founded, i.e.

$$\forall Y \; \forall C^{0\to\rho} \; \exists n^0 \; \forall m \geq n (Y(\overline{C, m}) < m),$$

verifying the mr-interpretation of DNS by modified bar-recursion requires Y to be *continuous*.

Hence, the mr-interpretation of DNS does not hold in the type structure of majorizable functionals (used for bound extraction).

Modified realizability and double-negation shift

For the mr-interpretation of DNS we need to realize

$$\forall x^0(P(n) \to \bot) \to \bot) \to (\forall x^0 P(x) \to \bot) \to \bot.$$

There is no realizers for $\bot$, so we need the additional step of A-translation. Let $*$ be the type of a realizer for the chosen A, then from realizers

$$Y^{(0\to\rho)\to *} \quad \text{mr } (\forall x^0 P(x) \to \bot)$$

$$G^{0\to((\rho\to *)\to *)} \quad \text{mr } \forall x^0((P(n) \to \bot) \to \bot)$$

we must produce a suitable realizer of type $*$.

Modified realizability and double-negation shift

Let us again look at the finite case, $k\text{-}DNS$, starting with the mr-interpretation of $\neg\neg P_0 \wedge \neg\neg P_1 \rightarrow \neg\neg(P_0 \wedge P_1)$.

Assume we are given the following realizers:

$$
\begin{array}{ll}
x_0 & \mathsf{mr}\ P_0 \\
x_1 & \mathsf{mr}\ P_1 \\
G_0 & \mathsf{mr}\ (P_0 \rightarrow \bot) \rightarrow \bot \\
G_1 & \mathsf{mr}\ (P_1 \rightarrow \bot) \rightarrow \bot \\
Y & \mathsf{mr}\ (P_0 \wedge P_1) \rightarrow \bot
\end{array}
$$

Modified realizability and double-negation shift

$Y(x_0, x_1)$	mr $\bot$
$\lambda x_1.Y(x_0, x_1)$	mr $P_1 \to \bot$
$G_1(\lambda x_1.Y(x_0, x_1))$	mr $\bot$
$G_0(\lambda x_0.G_1(\lambda x_1.Y(x_0, x_1))$	mr $\bot$

So in conclusion, $\lambda G_0, G_1, Y.G_0(\lambda x_0.G_1(\lambda x_1.Y(x_0, x_1))$ mr $\neg\neg P_0 \wedge \neg\neg P_1 \to \neg\neg(P_0 \wedge P_1)$.

Modified realizability and double-negation shift

However, attacking the same problem via modified bar-recursion – the restricted problem here corresponds to Y being uniformly continuous, sdepending only on the first two elements of the sequence – we get a very different solution:

$$Y(\langle \lambda n.H(G_0(\lambda x_0.Y(\langle x_0\rangle @ \lambda n.H(G_1(\lambda x_1.Y(\langle x_0, x_1\rangle)))))) \rangle)$$

where $H^{*\to\rho}$ is an auxilliary function satisfying $\forall n(H \text{ mr } B(n))$ and H is independent of n.

Modified realizability and double-negation shift

Questions: Why? What is the difference between extending the functional interpretation and the modified realizability interpretation to the infinitary case?

Is there a different way of treating the mr-interpretation of DNS?

Answers: I do not have the answers . . . yet!

Future work(in progress): Continue the investigations begun and discussed in this presentation.

References:

- Spector, "Provably recursive functionals of analysis : a consistency proof of analysis by an extension of principles formulated in current intuitionistic mathematics". In: Dekker(ed.), "Proc. of Symposia in Pure Mathematics", AMS, vol. 5, pp. 1-27, 1962.
- Berardi, Bezem and Coquand, "On the computational content of the axiom of choice". JSL, vol. 63, pp. 600-622, 1998.
- Berger, Oliva, "Modified bar recursion and classical dependent choice". In: Baaz, Friedman, and Krajcek(eds.), "Proc. of the Annual European Summer Meeting of the ASL", ASL and A K Peters, Lecture Notes in Logic, vol. 20, pp. 89-107, 2001.