COMPUTATIONAL CONTENT OF THE INTERMEDIATE VALUE THEOREM

HELMUT SCHWICHTENBERG

We report on a case study for program extraction in constructive analysis in the style of Bishop [1], concerning the intermediate value theorem. This proof is first given "on paper", in the usual mathematical style. Then we shortly describe a proof system (Minlog) it has been formalized in, as close as possible to the paper proof, but with the later program extraction in mind. In particular we take care that witnesses have as low a type level as possible. This clearly is important for the complexity of the extracted programs. For example, a continuous function on the reals is determined by its values on the rationals, and hence can be represented by a type-one (rather than type-two) object.

In this setup a short and easy to understand program can be extracted from the formalized proof. In doing so some interesting aspects concerning choice principles come up. It turns out that what one needs is a choice rule, which can be derived by "reflecting" modified realizability, that is by formalizing in the system the proof of the soundness theorem.

Related work has been done by Cruz-Filipe in his recent thesis [2].

1. Informal Proof

Lemma (Approximate Splitting Lemma). *For reals x, y, z,*

$$\forall x, y, z, k.x <_k y \to z \leq y \vee x \leq z.$$

Proof. Let $(a_n)_n$, $(b_n)_n$, $(c_n)_n$ be the Cauchy sequences of x, y, z with moduli M, N, L. $x <_k y$ means $1/2^k < b_p - a_p$ with $p := \max(M(k+2), N(k+2))$. Let $q := \max(L(k+2), p)$. Then in case $c_q \leq (a_p + b_p)/2$ we have $z \leq y$, and otherwise $x \leq z$. □

Lemma (Auxiliary Lemma for the Intermediate Value Theorem). *Let f be continuous on $[a, b]$ with a lower bound $1/2^l$ of its slope. Let $c, d \in [a, b]$ be given with $1/2^n \leq d - c$ such that $f(c) \leq 0 \leq f(d)$. Then there are $c_1, d_1 \in [c, d]$ such that $d_1 - c_1 = (2/3)(d - c)$ and again $f(c_1) \leq 0 \leq f(d_1)$.*

Proof. Let $c_0 := c + (1/3)(d - c)$ and $d_0 := c + (2/3)(d - c)$. We need a k to compare 0 with $f(c_0) <_k f(d_0)$. By assumption,

$$\frac{1}{2^{l+n+2}} \leq \frac{1}{2^{l+n}} \cdot \frac{1}{3} \leq \frac{1}{2^l} \cdot \frac{1}{3}(d-c) = \frac{1}{2^l}(d_0 - c_0) \leq f(d_0) - f(c_0).$$

This gives us $k := l + n + 2$ such that $f(c_0) <_k f(d_0)$. The Approximate Splitting Lemma allows us to distinguish cases as follws.

Case $0 \leq f(d_0)$. Take $c_1 := c$ and $d_1 := d_0$.

Case $f(c_0) \leq 0$. Take $c_1 := c_0$ and $d_1 := d$. □

Date: February 13, 2004.

Theorem (Intermediate Value Theorem). *Let f be continuous on $[a,b]$ with a lower bound $1/2^l$ of its slope. Assume $f(a) \leq 0 \leq f(b)$. Then f has a zero in $[a,b]$.*

Proof. Using the lemma above we construct two sequences $(c_n)_n$ and $(d_n)_n$ of rationals such that for all n

$$\begin{aligned}
&a = c_0 \leq c_1 \leq \cdots \leq c_n < d_n \leq \cdots \leq d_1 \leq d_0 = b,\\
&f(c_n) \leq 0 \leq f(d_n),\\
&d_n - c_n = (2/3)^n(b-a).
\end{aligned}$$

Let x, y be given by the Cauchy sequences $(c_n)_n$ and $(d_n)_n$ with the obvious modulus. As f is continuous, $f(x) = 0 = f(y)$ for the real number $x = y$. □

2. The Minlog System

2.1. **Logic.** Propositions and types are kept separate, as in ordinary mathematics. This eliminates the need for dependent types, at least in applications like the present one. Type variables and predicate variables are allowed as parameters. They are seen as placeholers for types and formulas (more precisely, comprehension terms), but – to ensure predicativity – cannot be quantified. This logic system is (conservative over) first order minimal logic.

2.2. **Arithmetic.** Ground types are (finitary or infinitary) algebras. Definition by (Gödel) primitive recursion and proof by induction is allowed. Function constants come with computation rules. They guarantee that every closed term of ground type evaluates to a numeral. In addition, every derived equation can be declared to be a rewrite rule. Then terms, formulas and proofs which are equal modulo these computation and rewrite rules are identified. This arithmetical system is (conservative over) first order minimal arithmetic.

3. Formalization and Program Extraction

The proof above can be formalized in the Minlog system, and a (rather short and perspicius) program can be extracted.

We only emphasize one aspect: the auxiliary lemma for the intermediate value theorem has the form $\forall c, d \exists c_1, d_1 A(c, d, c_1, d_1)$. However, in the proof of the intermediate value theorem we need a *function* assigning c_1, d_1 to c, d. Of course, such a function could have been constructed in the (constructive) proof of the lemma, but it is common and often easier not to do this explicitely. One might think here that the axiom of choice is nedded here, but this is not so: One can make use is a derivable choice rule instead.

This can be seen as follows. From the proof of $\forall c, d \exists c_1, d_1 A(c, d, c_1, d_1)$ (i.e., of the lemma) we can extract a term t realizing it. Now the soundness theorem says that t realizes $\forall c, d \exists c_1, d_1 A(c, d, c_1, d_1)$, and the proof of the soundness theorem for (modified) realizability can be done internally. But this means that we do have the desired function available.

References

1. Errett Bishop, *Foundations of constructive analysis*, McGraw-Hill, New York, 1967.
2. Luis Cruz-Filipe, *Constructive real analysis: a type-theoretical formalization and applications*, Ph.D. thesis, Nijmegen University, 2004.