Automating Proof in Non-standard Analysis

Ewen Maclean, Jacques Fleuriot, Alan Smaill

February 20, 2001

1 Introduction

We propose to apply the existing techniques of proof-planning (Bundy 88) and an area of mathematics known as non-standard analysis (Robinson 66). Proof-planning is a technique which helps to automate proof by working at an abstracted level. Non-standard analysis allows intuitive calculus proofs to be performed with the assurance of correctness thanks to a solid logical construction of the number system that underlies it– namely the hyperreals. The aim of this work is to develop proof-plans which capture the common patterns of reasoning that take place when proving theorems in non-standard analysis, and thereby to automate their proof. The work outlined here has been completed in the $\lambda Clam$ higher order proof-planner (Richardson *et al.* 00). We start by giving a brief overview of some concepts in both non-standard analysis and proof-planning; we then highlight possible future directions for our development.

2 Non-standard analysis

Non-standard analysis is a tool which provides an intuitive yet rigorous alternative to Weierstraß's tricky $\epsilon - \delta$ proofs. Abraham Robinson (Robinson 66) brought together a number of ideas from mathematical logic to come up with a theory of analysis in the hyperreal domain. By formally introducing an "infinitely close" relation, definitions and proofs become simpler and more intuitive.

As an example of how a definition in non-standard analysis can be simpler and more intuitive than its standard counterpart, consider the standard and the non-standard definitions of a limit.

Definition 1 *The limit l, of a function f at a point a is defined as:*

$$\lim_{x\to a} f(x) = l \quad \equiv \quad (\forall \epsilon > 0)(\exists \delta > 0)\forall x \quad 0 < |x-a| < \delta \to |f(x) - l| < \epsilon$$

Proofs which make use of this definition are usually very difficult to automate, on account of the alternating quantifiers, which mean that instantiations for δ have to guessed early on in the proof. Theorem 1 gives us an alternative characterisation of a limit making use of the infinitely close relation $\approx$.

Theorem 1 *The limit l, of a function f at a point a can equivalently be defined in non-standard analysis by:*

$$\lim_{x\to a} f(x) = l \quad \Longleftrightarrow \quad \forall x \in {}^*\mathbb{R}.\ x \approx {}^*a \wedge x \neq {}^*a \to f^*(x) \approx {}^*l.$$

As this theorem expresses the notion of limit over the hyperreals, a few points are worth mentioning: the function f^* corresponds to the original function f extended to accept hyperreal arguments whereas ${}^*a$ corresponds to the embedding of real value a in the hyperreals. These notions are introduced for correctness and will not be examined further here. Only an intuitive understanding of their behaviour is needed for the rest of this abstract although the interest reader may find out more by examining (Robinson 66).

This theorem gives us a simple non-standard characterisation for a limit which formalises the intuition that underlies the $\epsilon - \delta$ formulation of a limit. For a proof of theorem 1 the reader is urged to consult (Fleuriot & Paulson 00), for example.

3 Proof-planning

Proof planning (Bundy 88) is a technique for devising an overall plan for a proof, which can then be used to guide the proof search itself. A proof-plan consists of methods and critics. The methods embody common patterns within proofs, such as the use of induction, and the associated tactics carry out these methods explicitly in the object level prover. For each conjecture a precise proof-plan is built, but a general form of a proof-plan can be developed for certain classes of conjecture. A method in proof planning has a number of "slots" assigned to it which correspond to a name by which it is recognised, an input which is matched against the current goal, and a set of preconditions which determine whether the method is applicable to the current goal. The method also has a set of effects which define what happens to the goal after it has been applied, and a tactic which controls the object level prover, in which the plan can be executed. Finally, it has an output which is the result of applying the effects to the current goal.

3.1 Critics

The use of inductive theorem provers such as the Boyer-Moore Theorem Prover shows that failed proofs can provide useful information about failure, and hence can help to suggest ways to render proofs successful. One piece of useful information that could come out of a failed proof attempt is a suggestion of how to pre-process the conjecture in some way to help the proof succeed. In the context of proof-planning, a critic is a component which analyses this failure. Considerable work has been done in this area (Ireland 92), which the interested reader is urged to consult.

4 Proposed work

The work we propose in this project involves generating plans for proofs in non-standard analysis automatically, and thence executing them in an object level theorem prover. The methodology is one of genericity and automation. We have compiled a set of minimal axioms for non-standard analysis from which to derive all of the rules we require, and we hope to build a generic proof-plan whose initial state will be constant, but whose eventual shape will be transformed by the addition of intelligent critics. In addition to this, we have introduced a method for unfolding axioms by introducing such notions as *field* and *field homomorphism*. For example, the real numbers form a field, as do the hyperreals, so it is sensible to express field axioms in such a way, that the axioms specific to each type can be generated in a generic manner.

Some non-trivial theorems already planned include the chain rule for differentiation, the product rule and a number of theorems about limits. It is clear from the completed plans that the structure of proofs in non-standard analysis is simpler than that of standard proofs, and that the problems reduce in the main to equational reasoning problems, thanks to the lack of quantifier complexity. Proof-planning lends itself naturally to these problems, both in its ability to embody common patterns of reasoning and to analyse failed proof attempts.

5 Conclusion

The main aim of this project is to gain an understanding of the structure of proofs in non-standard analysis. Proof-planning allows this to be done in a declarative way by formally specifying plans. We believe that this will show that non-standard analysis is a more natural way of proving theorems about analysis in general, and also, perhaps, that for teaching purposes it is a sensible alternative to the infamous $\epsilon - \delta$ proofs of standard analysis.

References

A. Bundy. The use of explicit plans to guide inductive proofs. In R. Lusk and R. Overbeek, editors, *9th International Conference on Automated Deduction*, pages 111–120. Springer-Verlag, 1988. Longer version available from Edinburgh as DAI Research Paper No. 349.

J. D. Fleuriot and L. C. Paulson. Mechanizing nonstandard real analysis. *LMS Journal of Computation and Mathematics*, 3:140–190, 2000.

A. Ireland. The Use of Planning Critics in Mechanizing Inductive Proofs. In A. Voronkov, editor, *International Conference on Logic Programming and Automated Reasoning – LPAR 92, St. Petersburg*, Lecture Notes in Artificial Intelligence No. 624, pages 178–189. Springer-Verlag, 1992. Also available from Edinburgh as DAI Research Paper 592.

J. Richardson, L. Dennis, J. Gow, and M. Jackson. User/programmer manual for the $\lambda Clam$ proof planner. 2000.

A. Robinson. *Non-standard Analysis.* North-Holland Publishing Company, Amsterdam, 1966. Studies in Logic and the Foundations of Mathematics.