Transfer Principle Proof Tactic for Nonstandard Analysis

Brian Huffman
OGI School of Science and Engineering at OHSU

NETCA Workshop, Oxford, 26 August 2005

Introduction

- What is nonstandard analysis (NSA)?
  - NSA extends the reals $\mathbb{R}$ with new elements to form the "hyperreals" $\mathbb{R}^*$
  - Hyperreals include standard reals, plus infinite and infinitesimal values
  - Hyperreals inherit many algebraic properties from the reals
- What is it good for?
  - Provides formalization of intuitive notions in calculus
  - Can express concepts like limits more simply, with fewer quantifiers

Previous Work

- Fleuriot and Paulson, Mechanizing Nonstandard Real Analysis (2000)
  - Includes formalization of reals, hyperreals, and much real analysis in Isabelle/HOL
- My work presented here is based on theirs
  - Their formalization of reals can be reused
  - Their formalization of analysis may be adapted
- My contribution is a new way to define nonstandard types and operations on them, which provides more generality, more abstraction, and more automation

Part 1: Nonstandard Extensions of Types

Free Ultrafilters

- An ultrafilter $\mathcal{U}$ is a set of sets of naturals

- Basically, $\mathcal{U}$ specifies a relation that says when two countable sequences "mostly agree":

$$(X \sim Y) \equiv \{n.\ X_n = Y_n\} \in \mathcal{U}$$

- For any ultrafilter $\mathcal{U}$, this relation is reflexive, symmetric, and transitive

- For a *free* ultrafilter $\mathcal{U}$, if two sequences mostly agree, they must match in infinitely many places (i.e. $\mathcal{U}$ contains no finite sets)

Hyperreal Numbers

- The hyperreals are defined by sequences of reals, modulo ($\sim$) relation
- Standard reals are represented by constant sequences
- Algebraic operations ($+$, $-$, $\times$, $\div$, $\sin$, $\exp$, etc.) are defined pointwise
- Predicates hold for sequences if they hold for "most" terms:

$$P^*(X) \equiv \{n.\ P(X_n)\} \in \mathcal{U}$$

- The sequence $(1, 2, 3, 4, \ldots)$ represents an infinite hyperreal

Type Constructor `star`

- The previous quotient construction is not limited to reals
  - NSA traditionally makes use of hypercomplexes $\mathbb{C}^*$, hypernaturals $\mathbb{N}^*$
  - Why stop there? The construction works the same way over any type!

- It is natural to define a type constructor for nonstandard extensions of types
  - For every Isabelle type `'a`, we have a nonstandard extension `'a star`
  - Constructor functions:
    (surjective) `star_n::(nat => 'a) => 'a star`
    (injective) `star_of::'a => 'a star`

Defining Functions on Nonstandard Types

- Old method: Define functions pointwise over sequences
  - Drawback: Have to prove that this respects equivalence relation ($\sim$)
  - Different proofs needed for each function arity
- New method: Use nonstandard extension of function space!
  - Combinator `Ifun` returns so-called "internal" functions
    `Ifun::('a => 'b) star => ('a star => 'b star) (infixl ⋆)`
  - Definition designed to satisfy the following property:
    `star_n F ⋆ star_n X = star_n (λn.(F n) (X n))`
  - Reasoning about equivalence relation is only needed once!

Using `Ifun`

- We can use `Ifun` to lift functions of any arity

```
Ifun_of::('a => 'b) => 'a star => 'b star
Ifun_of f x == star_of f ⋆ x

Ifun2_of::('a => 'b => 'c) => 'a star => 'b star => 'c star
Ifun2_of f x y == star_of f ⋆ x ⋆ y

Ifun3_of::('a => 'b => 'c => 'd)
    => 'a star => 'b star => 'c star => 'd star
Ifun3_of f x y z == star_of f ⋆ x ⋆ y ⋆ z
```

Defining Predicates with `unstar`

- Function `unstar` is the inverse of `star_of` for booleans

```
unstar::bool star => bool
unstar b == (b = star_of True)
```

- Together with `Ifun`, we can lift predicates of any arity

```
Ipred_of::('a => bool) => 'a star => bool
Ipred_of p x == unstar (star_of p ⋆ x)

Ipred2_of::('a => 'b => bool) => 'a star => 'b star => bool
Ipred2_of p x y == unstar (star_of p ⋆ x ⋆ y)
```

Part 2: The Transfer Principle

What is the Transfer Principle?

- Transfer principle states that any first-order statement about reals is logically equivalent to a syntactically similar statement about hyperreals
- Transfer principle is a meta-mathematical theorem, since it quantifies over statements
- Compare with duality principle for statements about sets
- As a meta-mathematical theorem, we can't express the transfer principle as a theorem in Isabelle's logic
- We *can* write an algorithm for generating a proof for any instance of the transfer principle

Transfer Principle Proof Tactic

- In Isabelle, the transfer tactic replaces a subgoal about nonstandard types with an equivalent subgoal about ordinary types

- The tactic produces a proof of the equivalence, using a syntax-directed procedure

Conclusions

- Generalization
  - Can make nonstandard extensions of any type
  - Polymorphic functions over nonstandard types are now possible
- Abstraction
  - Combinators `star_of`, `Ifun`, and `unstar` are enough to define nearly all operations on nonstandard types
- Automation
  - The transfer tactic can quickly and effortlessly prove many theorems about nonstandard types