Now f is continuous (exercise!)

Rob Arthan

Lemma 1 Ltd. / Queen Mary, University of London

ETAPS/AIPA 2012

1st April, 2012

Background

- Goal is ITP for doing day-to-day mathematics.
- Proof obligations like the following are very common:
  - f is continuous
  - f is a homomorphism
  - f is a linear transformation
  - $\ldots$

  where f is defined by some complex expression.
- Can be very tedious to do manually.
- Obvious candidate for proof automation.
- Want a unified framework to solve these problems.

Overview

1. Implementing the categories of day-to-day maths in type theory.
2. Comparison with **Set**, **ScottDom** etc. Products (and sums).
3. Proving morphismhood in finitely presented categories:
   (a) Algorithms;
   (b) Implementation issues;
   (c) Additional features.

- Prototyped using the ProofPower-HOL Mathematical Case Studies.
- Slides available on line. URL on the last slide.

1. Concrete categories (I)

- Recall that a *concrete category* is one in which:
  - each object X has an *underlying set* $U(X)$;
  - morphisms $X \to Y$ are functions $f : U(X) \to U(Y)$;
  - $g \circ f = \lambda x \bullet g(f(x)))$.
- Represents a common mathematical scenario dealing with:
  - sets equipped with some extra structure;
  - functions between the sets that "respect" the structure.

1. Concrete categories (II)

- Examples:

Name	Objects	Morphisms
Set	All sets	Arbitrary functions
Grp	Groups	Group homomorphisms
$\mathbb{R}-$**Vec**	Real vector spaces	Linear maps
Top	Topological spaces	Continuous functions

and many, many more.

- Non-examples:

Name	Objects	Morphisms
Rel	All sets	Arbitrary relations
Toph	Topological spaces	$\mathbf{Top}(X, Y)/\simeq$

where $f \simeq g$ means f and g are homotopy equivalent.

1. Representing a concrete category in type theory

- E.g. **Top**: an object of **Top** is given by a *topology*:

$$
\begin{aligned}
Topology = \{\tau : {}'a\ SET\ SET \mid \\
&(\forall\ V \bullet\ V \subseteq \tau \Rightarrow \bigcup V \in \tau) \\
\wedge\ &(\forall\ A\ B \bullet\ A \in \tau \wedge B \in \tau \Rightarrow A \cap B \in \tau)\}
\end{aligned}
$$

- We call the underlying set of an object its *space*: $Space_T\ \tau\ = \bigcup \tau$

- The morphisms are the *continuous* functions:

$$
\begin{aligned}
(\sigma, \tau)\ Continuous = \{f : {}'a \rightarrow {}'b \mid \\
&(\forall\ x \bullet\ x \in Space_T\ \sigma \Rightarrow f\ x \in Space_T\ \tau) \\
\wedge\ &(\forall\ A \bullet\ A \in \tau \Rightarrow \{x \mid x \in Space_T\ \sigma \wedge f\ x \in A\} \in \sigma)\}
\end{aligned}
$$

(Syntax: *Continuous* is a postfix operator on pairs of topologies.)

1. Proving morphismhood in $(\mathbb{R}; \circ, f_1, f_2, \ldots)$

- A specific topology: the interval topology on $\mathbb{R}$:

$$O_R = \{A : \mathbb{R}\ SET \mid \forall\ t \bullet\ t \in A \Rightarrow (\exists\ x\ y \bullet$$
$$t \in OpenInterval\ x\ y \wedge OpenInterval\ x\ y \subseteq A)\}$$

- Assume given the following facts:

$\vdash Exp \in Cts$; $\vdash Sin \in Cts$; $\vdash Cos \in Cts$; $\vdash Ic \in Cts$;

$\vdash \forall f\ g \bullet\ f \in Cts \wedge g \in Cts \Rightarrow g\ o\ f \in Cts$.

where $Cts = (O_R,\ O_R)\ Continuous$ and Ic is the **I** combinator.

- To prove, say: $(\lambda x \bullet Sin(Cos(Exp\ x))) \in Cts$
  - rewrite as $(Sin\ o\ Cos\ o\ Exp) \in Cts$
  - then backchain with the facts.

(We could have written $(g\ o\ f) = \lambda x \bullet g(f\ x)$ in the facts, but this is not a linear pattern, so higher-order matching is not immediately helpful here.)

2. Comparison with **Set** (and **ScottDom** and ...) (I)

- Concrete categories may have (finite) products, but need not.
- Say the product is *standard* if $U(X \times Y) = U(X) \times U(Y)$.
- Many useful examples do have standard products. E.g.,
  - **Top**;
  - Any concrete category axiomatised by first-order Horn clauses.
    - E.g., **Grp**, $\mathbb{R}-$**Vec**, **POGrp**, ...
    - Not fields.
- Similar situation for sums. E.g.,
  - **Top** has standard sums;
  - $\mathbb{R}-$**Vec** has (finite) products that are also sums: $X + Y = X \times Y$.
- Focus on products in this talk.

2. Comparison with **Set** (and **ScottDom** and ...) (II)

- Cartesian-closed concrete categories are rare.
- **Top** is not Cartesian-closed:
  - lots of ways of topologising $X \to Y$;
  - "pathological" cases defeat them all.
- **Grp**, $\mathbb{R}$−**Vec** and ... are not Cartesian-closed.
- Curry is off the menu!
  $\lambda x \bullet \lambda y \bullet t$ is:
  - at best a second-class citizen (e.g., in **Top**);
  - more often an outright outlaw (e.g., in **Grp**).

3(a). Proving morphismhood in $(\mathbb{R}, \times; \circ, \langle\rangle, f_1, f_2, \ldots) \subseteq \mathbf{Top}$ (I)

- Product of two topologies:

$$\sigma \times_T \tau = \{C : ('a \times 'b)\ SET \mid \forall\ x\ y\bullet\ (x,\ y) \in C \Rightarrow$$
$$(\exists\ A\ B\bullet\ A \in \sigma \wedge B \in \tau \wedge x \in A \wedge y \in B \wedge (A \times B) \subseteq C)\}$$

- Pairing functions on underlying sets:

$$Pair\ (f,\ g) = (\lambda\ x\bullet\ (f\ \ x,\ g\ \ x))$$

- New facts: for $\rho,\ \sigma,\ \tau\ \in\ \{O_R,\ O_R\ \times_T\ O_R,\ ...\}$:

$$\vdash \forall f\ \ g\bullet\ f\ \in (\rho,\ \sigma)\ Continuous \wedge g \in (\rho,\ \tau)\ Continuous$$
$$\Rightarrow Pair\ (f,\ g) \in (\rho,\ \sigma \times_T \tau)\ Continuous$$
$$\vdash \forall f\ \ g\bullet\ f\ \in (\rho,\ \sigma)\ Continuous \wedge g \in (\sigma,\ \tau)\ Continuous$$
$$\Rightarrow g\ \ o\ \ f\ \in (\rho,\ \tau)\ Continuous$$

3(a). Proving morphismhood in $(\mathbb{R}, \times; \circ, \langle\rangle, f_1, f_2, \ldots) \subseteq \mathbf{Top}$ (II)

- To prove, say:

  $$(\lambda x \bullet (Sin(Exp\ x),\ Cos\ (Exp\ x))) \in (O_R,\ O_R \times_T O_R)\ Continuous$$

  - rewrite LHS as $Pair\ (Sin\ o\ Exp\ ,\ Cos\ o\ Exp)$
  - then backchain with the facts.

- What about binary operations? E.g.,
  $(\lambda(x,\ y) \bullet\ Exp(x\ +\ y)) \in (O_R \times_T O_R,\ O_R)\ Continuous$
  - rewrite LHS as $Exp\ o\ Uncurry\ \$+\ o\ Pair\ (Fst,\ Snd)$
  - then backchain using a new fact:
    $\vdash\ Uncurry\ \$+\ \in (O_R \times_T O_R,\ O_R)\ Continuous$

  (Syntax: the \$ prevents + being treated as an infix operator.)

- Maybe defining $+ : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ rather than $+ : \mathbb{R} \to \mathbb{R} \to \mathbb{R}$ would have been better after all?

3(a). Proving morphismhood in $(\mathbb{R}, \times; \circ, \langle\rangle, f_1, f_2, \ldots) \subseteq \mathbf{Top}$ (III)

- What about constant operands? E.g.,
  $(\lambda x \bullet\ \mathit{2.0}\ *\ (x\ \widehat{}\ \mathit{4})) \in (O_R,\ O_R)\ \mathit{Continuous}$
  - rewrite LHS as $\mathit{Uncurry}\ \$*\ o\ \mathit{Pair}\ (\mathit{Kc}\ \mathit{2.0},\ (\lambda\ x \bullet\ x\ \widehat{}\ \mathit{4}))$ where Kc is the K combinator.
  - then backchain using new facts:
    $\vdash\ \forall c \bullet\ \mathit{Kc}\ c\ \in\ (\sigma,\ \tau)\ \mathit{Continuous}$
    $\vdash\ \forall n \bullet\ (\lambda\ x \bullet\ x\ \widehat{}\ n)\ \in\ (O_R,\ O_R)\ \mathit{Continuous}$
- We are treating $\lambda\ x \bullet\ x\ \widehat{}\ n$ as family of continuous functions parametrized by $n : \mathbb{N}$.

3(a). Continuity of $f : \mathbb{R} \rightarrow \mathbb{C}$ where $f(x) = e^{2\pi i x}$

- Let's try a famous example:

  $(\lambda x \bullet\ \mathit{Exp}(\mathbb{R}\mathbb{C}\ 2.\ *\ \mathbb{R}\mathbb{C}\ \pi\ *\ I_C\ *\ \mathbb{R}\mathbb{C}\ x)) \in (O_R,\ O_C)\ \mathit{Continuous}$

  - Expand definitions of the complex topology and complex operators:

    $(\lambda\ x\bullet\ (\mathit{Exp}\ 0.\ *\ \mathit{Cos}\ (2.\ *\ \pi\ *\ x),\ \mathit{Exp}\ 0.\ *\ \mathit{Sin}\ (2.\ *\ \pi\ *\ x)))$
    $\in (O_R,\ O_R \times_T O_R)\ \mathit{Continuous}$

  - rewrite LHS as

    Pair (Uncurry $\$$ o Pair (Kc (Exp 0.), Cos o Uncurry $\$*$ o*
    Pair (Kc 2., Uncurry $\$$ o Pair (Kc π, Ic))),*
    Uncurry $\$$ o Pair (Kc (Exp 0.), Sin o Uncurry $\$*$ o*
    Pair (Kc 2., Uncurry $\$$ o Pair (Kc π, Ic))))*
    $\in (O_R,\ O_R \times_T O_R)\ \mathit{Continuous}$

  - then backchain as usual.

- a one-liner for a user:

  a(basic_continuity_tac[ℂ_exp_def, ℝℂ_def, ℂ_i_def, ℂ_times_def, open_ℂ_def]);

3(a). The Rewrite System

$$
\begin{array}{rcll}
(\lambda V \bullet x) & \rightsquigarrow & \pi_x^V & x \in \mathsf{frees}(V) \\
(\lambda V \bullet y) & \rightsquigarrow & \mathsf{K}\, y & y \notin \mathsf{frees}(V) \\
(\lambda V \bullet c) & \rightsquigarrow & \mathsf{K}\, c & c \in \mathsf{Constant} \\
(\lambda V \bullet (t_1, t_2)) & \rightsquigarrow & \langle (\lambda V \bullet t_1), (\lambda V \bullet t_2) \rangle & \\
(\lambda V \bullet f\, t) & \rightsquigarrow & f \circ (\lambda V \bullet t) & f \in \mathsf{Unary} \\
(\lambda V \bullet g\, t_1\, t_2) & \rightsquigarrow & \mathsf{Uncurry}\, g \circ \langle (\lambda V \bullet t_1), (\lambda V \bullet t_2) \rangle & g \in \mathsf{Binary} \\
(\lambda V \bullet h\, t\, p) & \rightsquigarrow & (\lambda x \bullet h\, x\, p) \circ (\lambda V \bullet t) & h \in \mathsf{Parametrized}
\end{array}
$$

Where V is a pattern made up from (distinct) variables using $(\_,\_)$ and:

- We write $\langle f, g \rangle$ for $Pair(f,\ g)$;
- If V is a pattern with a free occurrence of the variable x, we write π_x^V for the combination of projections which extracts x.
  - E.g., writing π_1 and π_2 and for Fst and Snd, $\pi_x^{((z,x),y)}$ is $\pi_2 \circ \pi_1$.
  - As a special case, $\pi_x^x = \mathsf{I}$, and we may simplify $f \circ \mathsf{I}$ to f.

3(b). Implementation Notes (I)

- Miller-Nipkow higher-order matching is all we need.
- Don't need to handle non-linear patterns or paired abstraction:
  - A non-linear template theorem such as:
    $\vdash \ \forall f \ s \ t \bullet \ (\lambda x \bullet f \ (s \ x) \ (t \ x)) \ = \ \mathit{Uncurry} \ f \ \ o \ \mathit{Pair}(s, \ t)$
    instantiates to linear form:
    $\vdash \ \forall s \ t \bullet \ (\lambda x \bullet (s \ x) \ + \ (t \ x)) \ = \ \mathit{Uncurry} \ \$+ \ \ o \ \mathit{Pair}(s, \ t).$
  - A paired abstraction in the goal such as $(\lambda (x, \ y) \bullet x \ + \ y)$ can be preprocessed into $\lambda xy \bullet \ \mathit{Fst} \ xy \ + \ \mathit{Snd} \ xy$.
- Aside: I would still like an implementation of the Löchner-Fettig algorithm. Pointers appreciated!

3(b). Implementation Notes (II)

- **Unary**, **Binary** and **Parametrized** determine the basic homomorphisms of the category.
- For $(\mathbb{R}, \times; \circ, \langle\rangle, f_1, f_2, \ldots) \subseteq \mathbf{Top}$

Unary	Fst, Snd, $\sim$, Exp, Sin, Cos, ...
Binary	\$+, \$*
Parametrized	\$ ^

- For $(\mathbb{R}_+, \mathbb{C}_+, \times; \circ, \langle\rangle, f_1, f_2, \ldots) \subseteq \mathbf{Grp}$

Unary	Fst, Snd, $\sim$, \$ * $(c : \mathbb{R})$, \$ * $(c : \mathbb{C})$, \$⁻ : $\mathbb{C} \to \mathbb{C}$
Binary	\$+ : $\mathbb{R} \to \mathbb{R} \to \mathbb{R}$, \$+ : $\mathbb{C} \to \mathbb{C} \to \mathbb{C}$
Parametrized	\$* : $\mathbb{R} \to \mathbb{R} \to \mathbb{R}$, \$* : $\mathbb{C} \to \mathbb{C} \to \mathbb{C}$

  Because $\lambda(x,\ y) \bullet\ x\ *\ y$ is *not* an additive homomorphism while $\lambda x \bullet c\ *\ x$ and $\lambda x \bullet x\ *\ c$ are.

(Syntax: the postfix operator \$⁻ is complex conjugation.)

Defining $* : \mathbb{R} \to \mathbb{R} \to \mathbb{R}$ is convenient here!

3(b). Implementation Notes (III)

- The infinite schemas like:

$$\vdash \forall f \ g \bullet \ f \in (\rho,\ \sigma)\ Continuous \wedge g \in (\sigma,\ \tau)\ Continuous$$
$$\Rightarrow g \ o \ f \in (\rho,\ \tau)\ Continuous$$

  may be implemented using template theorems:

$$\vdash \forall \ \rho \ \sigma \ \tau \ f \ g \bullet \rho \in \ Topology \wedge \sigma \in \ Topology \wedge \tau \in \ Topology \wedge$$
$$f \in (\rho,\ \sigma)\ Continuous \wedge g \in (\sigma,\ \tau)\ Continuous$$
$$\Rightarrow g \ o \ f \in (\rho,\ \tau)\ Continuous$$

- But you need to find witnesses for intermediate objects like σ above.
- If we assume there is at most one object per type, can find witness using type. E.g., $(\mathbb{R} \ \times \ \mathbb{R})SET\ SET$, gives witness $O_R \ \times_T \ O_R$.
- Easy to implement by matching types with types of the constructors, O_R, $\$ \times_T$, ...

3(b). Proving morphismhood in $(\mathbb{R}, \mathbb{C}, \times; \circ, \langle\rangle, f_1, f_2, \ldots) \subseteq \mathbf{Grp}$ (I)

- Let's try proving that $f(x) = e^{2\pi i x}$ defines a group homomorphism:

  $(\lambda x \bullet\ \mathit{Exp}(\mathbb{RC}\ \mathit{2.} * \mathbb{RC}\ \pi * I_C * \mathbb{RC}\ x)) \in \mathit{Homomorphism}\ (\mathbb{R}_+, \mathbb{C}_*)$

- rewrite LHS as

  $\mathit{Exp}\ o\ \$*\ (\mathbb{RC}\ \mathit{2.})\ o\ \$*\ (\mathbb{RC}\ \pi)\ o\ \$*\ I_C\ o\ \mathbb{RC}$

- then backchain as usual using additional facts:

  $\vdash \mathit{Exp} \in \mathit{Homomorphism}\ (\mathbb{R}_+ \times_G \mathbb{R}_+, \mathbb{C}_*)$;
  $\vdash \mathbb{RC} \in \mathit{Homomorphism}\ (\mathbb{R}_+, \mathbb{C}_*)$;
  $\vdash \forall\ c : \mathbb{C} \bullet\ \$*\ c \in \mathit{Homomorphism}\ (\mathbb{R}_+ \times_G \mathbb{R}_+, \mathbb{R}_+ \times_G \mathbb{R}_+)$:

- But we were a little lucky ...

3(b). Proving morphismhood in $(\mathbb{R}, \mathbb{C}, \times; \circ, \langle\rangle, f_1, f_2, \ldots) \subseteq \mathbf{Grp}$ (II)

- Let's try another example of a group homomorphism:

  $(\lambda x \bullet \; Exp(x)^-) \in Homomorphism(\mathbb{R}_+ \times_G \mathbb{R}_+, \mathbb{C}_*)$

  - rewrite LHS as

  $\$^- \;\; o \;\; Exp$

  - then backchain as usual using additional fact:

  $\vdash \$^- \in Homomorphism\;(\mathbb{C}_*, \mathbb{C}_*)$

  - Fails with false subgoals:

  $?\vdash \$^- \in Homomorphism\;(\mathbb{R}_+ \times_G \mathbb{R}_+, \mathbb{C}_*)$

  $?\vdash Exp \in Homomorphism\;(\mathbb{R}_+ \times_G \mathbb{R}_+, \mathbb{R}_+ \times_G \mathbb{R}_+)$

  - The one-object-per-type approach has chosen the wrong intermediate group structure.

3(c). Improving the witnessing method

- The procedure found the wrong witness to the goal:

$$
\begin{aligned}
&?\vdash \exists G \bullet\ G \in \mathit{Group} \\
&\quad \wedge\ \$^{-} \in \mathit{Homomorphism}\ (G,\ \mathbb{C}_*) \\
&\quad \wedge\ \mathit{Exp} \in \mathit{Homomorphism}\ (\mathbb{R}_+ \times_G \mathbb{R}_+,\ G)
\end{aligned}
$$

- Can find the right witness by matching goal conjuncts with facts.
- With $G = \mathbb{C}_*$ all is well.
- May need a slightly deeper analysis, e.g., for chains of projections:

$$\mathit{Fst}\ o\ \mathit{Snd}\ o\ \mathit{Fst}$$

3(c). Other ways of making new morphisms from old

- Definition by cases is a common way of getting new functions from old.
- Here is a principle of definition by cases in **Top**:

$\vdash \forall\ c\ f\ g\ \sigma\ \tau \bullet \sigma \in \mathit{Topology} \land \tau \in \mathit{Topology}$
$\land\ c \in (\sigma,\ \mathit{Open}_R)\ \mathit{Continuous}$
$\land\ f \in (\sigma,\ \tau)\ \mathit{Continuous} \land g \in (\sigma,\ \tau)\ \mathit{Continuous}$
$\land\ (\forall\ x \bullet\ x \in \mathit{Space}_T\ \sigma \land c\ x = 0. \Rightarrow f\ x = g\ x)$
$\Rightarrow (\lambda\ x \bullet\ \mathit{if}\ c\ x \le 0.\ \mathit{then}\ f\ x\ \mathit{else}\ g\ x) \in (\sigma,\ \tau)\ \mathit{Continuous}: \mathit{THM}$

- The real-valued function c partitions $\mathit{Space}_T\ \sigma$ into two pieces.
- The new function agrees with f on one piece and with g on the other.
- f and g must agree where the pieces overlap.
- Fits into the framework as a new sort of fact . . .
- . . . provided users agree to make their definitions in the right style.
- Many other definitional principles worth investigating.

Final Remarks

- For the slides: `http://www.lemma-one.com/papers/`
- For ProofPower: `http://www.lemma-one.com/ProofPower/`
- Tools for proving morphismhood in the usual categories of day-to-day maths are both:
  - extremely useful &
  - relatively simple to implement.

Thank you!