Type Classes and Filters for Mathematical Analysis in Isabelle/HOL

Johannes Hölzl[1⋆], Fabian Immler[1⋆⋆], and Brian Huffman[2]

[1] Institut für Informatik, Technische Universität München
hoelzl@in.tum.de, immler@in.tum.de
[2] Galois, Inc.
huffman@galois.com

Abstract. The theory of analysis in Isabelle/HOL derives from earlier formalizations that were limited to specific concrete types: $\mathbb{R}$, $\mathbb{C}$ and $\mathbb{R}^n$. Isabelle's new analysis theory unifies and generalizes these earlier efforts. The improvements are centered on two primary contributions: a generic theory of limits based on filters, and a new hierarchy of type classes that includes various topological, metric, vector, and algebraic spaces. These let us apply many results in multivariate analysis to types which are not Euclidean spaces, such as the extended real numbers, bounded continuous functions, or finite maps.

Keywords: Type classes, Filters, Mathematical analysis, Topology, Limits, Euclidean vector spaces, Isabelle/HOL

1 Introduction

Mathematical analysis studies a hierarchy of abstract objects, including various topological, metric, and vector spaces. However, previous formalizations of mathematical analysis have not captured this hierarchical structure. For example, in HOL Light's multivariate analysis library [4] most theorems are proved only for the fixed type $\mathbb{R}^n$ of finite Cartesian products. Similarly, Isabelle's original library of analysis by Fleuriot and Paulson [1] supported most concepts only on $\mathbb{R}$ and $\mathbb{C}$.

Isabelle/HOL's new library for mathematical analysis derives from these two earlier libraries, but brings them closer to the mathematical ideal: Isabelle/HOL provides the concept of type classes, which allows us to state lemmas generically for all types that provide the necessary operations and satisfy the corresponding assumptions. This approach is therefore perfectly suited to exhibit the hierarchical structure of spaces within mathematical analysis.

In the following text we present the new hierarchy of type classes for mathematical analysis in Isabelle/HOL and preview some example class instances:

– Finite Cartesian products $\mathbb{R}^n$, $\mathbb{R}$, and $\mathbb{C}$ are all Euclidean spaces.

⋆ Supported by the DFG Projekt NI 491/15-1
⋆⋆ Supported by the DFG Graduiertenkolleg 1480 (PUMA)

– The extended reals $\overline{\mathbb{R}} = \mathbb{R} \cup \{\infty, -\infty\}$ are a non-metric topological space.
– Finite maps (maps with finite domain) $\mathbb{N} \to_f \mathbb{R}$, are a complete metric space but not a vector space. They are used to construct stochastic processes [6].
– Bounded continuous functions $\mathbb{R} \to_{bc} \mathbb{R}$ form a Banach space but not a Euclidean space; their dimension is infinite. They are used to prove that ordinary differential equations have a unique solution [7].

Figure 1 shows the type class hierarchy we present in this paper. Full lines are inheritance relations and dashed lines are proved subclass relations. We group the type classes into topological, metric, vector and algebraic type classes. For completeness we show some of the algebraic type classes, but they are not the main focus of this paper. All type classes described in this paper are available in Isabelle 2013 and carry the same names in the formalization. An exception is the order topology, available in Isabelle's development repository[3].

Our formalization of filters and limits is another primary contribution of our work. While filters have long been used to express limits in topology, our generic limit operator parameterized by two filters is novel (see Section 4.2). Filters are also useful for more than just limits—e.g. a filter can express the *almost everywhere* quantifier, which recognizes predicates that hold with probability 1 on a probability space.

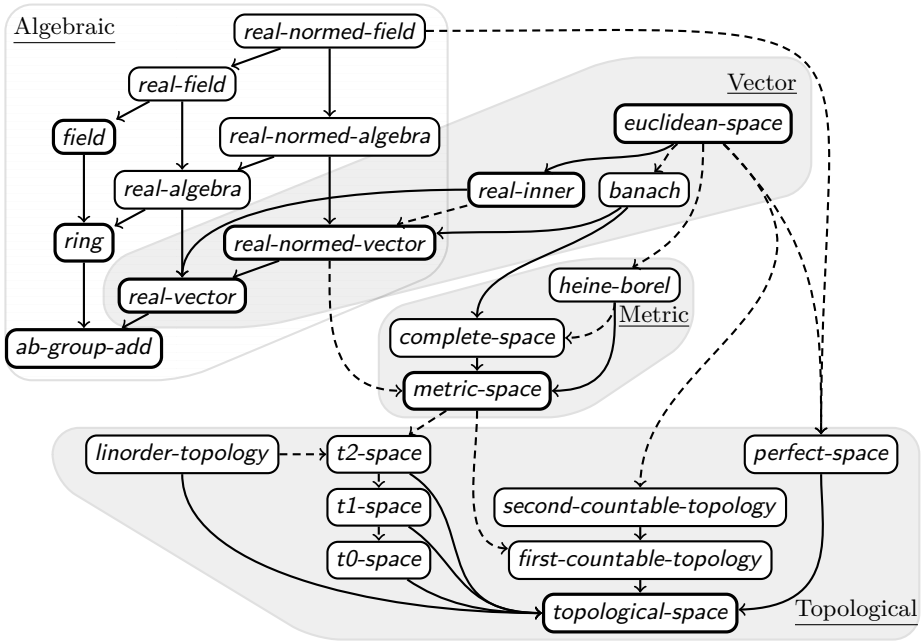

Fig. 1. Type class hierarchy

[3] http://isabelle.in.tum.de/repos/isabelle/rev/4392eb046a97

2 Preliminaries

The term syntax used in this paper follows Isabelle/HOL, i.e. as usual in λ-calculus function application is juxtaposition as in $f\ t$. The notation $t :: \tau$ means that term t has type τ. Types are built from the base types $\mathbb{B}$ (booleans), $\mathbb{N}$ (natural numbers), $\mathbb{R}$ (reals), type variables (α, β, etc), via the function type constructor $\alpha \to \beta$, and via the set type constructor $\alpha\ \mathsf{set}$. The universe of a type α, i.e. the set of all elements of type α, is written $\mathcal{U}_\alpha$. We use $\Longrightarrow$ for logical implication; it binds – in contrast to Isabelle/HOL notation – stronger than universal quantification, i.e. $\forall x.\ P\ x \Longrightarrow Q\ x$ equals $\forall x.\ (P\ x \Longrightarrow Q\ x)$.

Isabelle/HOL provides (axiomatic) type classes [2], which allow to organize polymorphic specifications. A type class C specifies assumptions $P_1, \ldots, P_k$ for constants $c_1, \ldots, c_m$ (that are to be overloaded) and may be based on other type classes $B_1, \ldots, B_n$. The command **class** declares type classes in Isabelle/HOL:

class $C = B_1 + B_2 + \ldots + B_n +$
  fixes $c_1 :: \alpha\ \kappa_1$ **and** $c_2 :: \alpha\ \kappa_2$ **and** $\cdots$ **and** $c_m :: \alpha\ \kappa_m$
  assumes P_1 **and** P_2 **and** $\cdots$ **and** P_k

In the type class specification only one type variable, α, is allowed to occur. Variables in $P_1, \ldots, P_k$ are implicitly universally quantified. A type α is said to be an instance of the type class C if it provides definitions for the respective constants and respects the required assumptions. In this case we write $\alpha :: C$.

With the command **instance** we can add subclass relations in addition to the declared base classes. We have for example the type class *finite* for types α where $\mathcal{U}_\alpha$ is finite and a type class *countable* for types α where $\mathcal{U}_\alpha$ is countable. Then we can use **instance** *finite* $\subseteq$ *countable* to add a subclass relation stating that all finite types are also countable types.

3 Related Work

Isabelle's original theory of real analysis was due to Fleuriot and Paulson [1]. It covered sequences, series, limits, continuity, transcendental functions, nth roots, and derivatives. These notions were all specific to $\mathbb{R}$, although much was also duplicated at type $\mathbb{C}$. This material has since been adapted to the new type class hierarchy. The non-standard analysis part with ${}^*\mathbb{R}$ and ${}^*\mathbb{C}$ is not adapted.

Much of the work presented in this paper comes from the Isabelle/HOL port of Harrison's multivariate analysis library for HOL Light [4]. In addition to limits, convergence, continuity, and derivatives, the library also covers topology and linear algebra. The Heinstock-Kurzweil integral is not yet described in [4], but it is now available in HOL Light and also ported to Isabelle/HOL. Compared to the work presented in this paper the HOL Light library is mostly specific to $\mathbb{R}^n$.

Instead of formalizing limits with filters, Harrison invented a variant of nets which also bore some similarities to filter bases. His library provided a tends-to

relation parameterized by a single net, but did not have an equivalent of our more general limit operator which is parameterized by two filters (see Section 4.2).

Lester [9] uses PVS to formalize topology. He formalizes topological spaces, T_2-spaces, second countable space, and metric spaces. He does not provide vector spaces *above* metric spaces and he does not use filters or nets to express limits.

Spitters and van der Weegen [10] formalize a type class hierarchy for algebraic types in Coq. Their goal is efficient computation, hence they support different implementations for isomorphic types. In contrast, our goal is to share definitions and proofs for types which share the same mathematical structure. They also introduce type classes in category theory which is not possible in Isabelle as type classes are restricted to one type variable. However, for mathematical analysis and also for the algebraic type class hierarchy in Isabelle/HOL they suffice.

Hölzl and Heller [5], Immler and Hölzl [7], and Immler [6] provide instances of the type classes presented in this paper: they formalize extended real numbers, bounded continuous functions, and finite maps.

4 Topology

Topology is concerned with expressing *nearness* of elements in a space. An open set contains for each element also all elements which are in some sense *near* it. This structure is sufficient to express limits and continuity of functions on topological spaces. This generality is actually needed to formulate a notion of limits and convergence that is also suitable for extended real numbers. More specific formulations (e.g. in terms of metric spaces) do not work for them. For a introduction into topology the reader may look into standard textbooks like [8].

4.1 Topological Spaces

A *topological space* is defined by its predicate of open sets. In mathematics the support space X, the union of all open sets, is usually explicitly given. In Isabelle/HOL a topological space is a type in the following type class:

class *topological-space* =
  fixes *open* :: α *set* $\to \mathbb{B}$
  assumes *open* $\mathcal{U}_\alpha$ **and** *open* $U \implies$ *open* $V \implies$ *open* $(U \cap V)$
    and $(\forall U \in S.$ *open* $U) \implies$ *open* $\bigcup S$
closed :: α *set* $\to \mathbb{B}$
closed $U \longleftrightarrow$ *open* $(\mathcal{U}_\alpha \setminus U)$

On a topological space, we define the limit points, interior, closure and frontier of a set in the usual way.

On the real numbers, the canonical topology contains all half-bounded open intervals: $]a, \infty[$ and $]\infty, a[$. It is also generated by them, i.e. it is the smallest topology containing all half-bounded open intervals. This is called an *order topology* on a linear order:

class *linorder-topology* = *linorder* + *topological-space* +
  assumes *open* = *generated-topology* $\bigcup_x \{]x, \infty[,]\infty, x[\}$

Here *generated-topology* A is the smallest topology where the sets in A are open. A is a subbase, i.e. A need not be closed under intersection. Instances for order topologies are the real numbers and the extended real numbers.

Separation spaces. As the open sets of a topology describe only *nearness*, it is still possible that two distinct elements are always near, i.e. they are *topologically indistinguishable*. This is not desirable when formulating unique limits in terms of open sets. To prevent this, different classes of *separation spaces* are specified, called T_0-, T_1-, and T_2-*spaces*:

class *t0-space* = *topological-space* +
  assumes $x \neq y \implies \exists U.\ \mathit{open}\ U \wedge \big(x \in U \iff y \notin U\big)$

class *t1-space* = *topological-space* +
  assumes $x \neq y \implies \exists U.\ \mathit{open}\ U \wedge \big(x \in U \wedge y \notin U\big)$

In T_1-spaces singleton sets are closed, i.e. *closed* $\{x\}$. A T_2-space (also called a Hausdorff space) is the strongest separation space we provide. A T_2-space provides for any distinct elements x and y two disjoint open sets around them:

class *t2-space* = *topological-space* +
  assumes $x \neq y \implies$
    $\exists U, V.\ \mathit{open}\ U \wedge \mathit{open}\ V \wedge x \in U \wedge y \in V \wedge U \cap V = \emptyset$

We provide type class inclusion for these spaces according to their numbering; i.e. a T_2-space is also a T_1-space is also a T_0-space. In Section 5.1 we also prove that each metric space and each linearly ordered topology is a T_2-space.

instance *t1-space* $\subseteq$ *t0-space*
instance *t2-space* $\subseteq$ *t1-space*
instance *linorder-topology* $\subseteq$ *t2-space*

While the T_1-spaces tell us that two elements can always be separated, we also need its dual: in a *perfect space* each open set containing an element always contains elements around it; the singleton set is never open. This is the dual to *closed* $\{a\}$. Only in perfect spaces is $\lim_{x \to a}$ meaningful for each point a.

class *perfect-space* = *topological-space* + **assumes** $\neg \mathit{open}\ \{a\}$

Instances of perfect spaces are Euclidean spaces and the extended real numbers.

Topologies with countable basis. A *first countable topology* assumes a countable basis for the neighborhoods of every point; i.e. it allows us to construct a sequence of open sets that converges towards a point x. Together with a T_1-space this allows us to construct a sequence of points that converges to a point x.

class *first-countable-topology* = *topological-space* +
  assumes $\exists A.\ \mathit{countable}\ A \wedge (\forall a \in A.\ \mathit{open}\ a \wedge x \in a) \wedge$
    $(\forall U.\ \mathit{open}\ U \wedge x \in U \implies \exists a \in A.\ a \subseteq U)$

Examples of first countable topologies are metric spaces.

Second countability is an extension of first countability; it provides a countable basis for the whole topology, not just for the neighborhoods of every point. This implies that compactness is equivalent to sequential compactness (which will be introduced in Section 4.4).

class *second-countable-topology* = *topological-space* +
  assumes $\exists B.$ *countable* $B \wedge$ *open* = *generated-topology* B
instance *second-countable-topology* $\subseteq$ *first-countable-topology*

Instances for second countable spaces are Euclidean spaces, the extended real numbers, and finite maps ($\alpha ::$ *countable*) $\rightharpoonup_f$ ($\beta ::$ *second-countable-topology*).

4.2 Filters and Limits

A *filter* is a set of sets (or equivalently a predicate on predicates) with a certain order structure. As we will see shortly, filters are useful in topology because they let us unify various kinds of limits and convergence, including limits of sequences, limits of functions at a point, one-sided and asymptotic limits.

Many varieties of logical quantification are filters, such as "for all x in set A"; "for sufficiently large n"; "for all but finitely many x"; "for x sufficiently close to y". These quantifiers are similar to the ordinary universal quantifier ($\forall$) in many ways. In particular, each holds for the always-true predicate, preserves conjunction, and is monotonic:

$(\square x.\ \mathit{True})$
$(\square x.\ P\ x) \implies (\square x.\ Q\ x) \implies (\square x.\ P\ x \wedge Q\ x)$
$(\forall x.\ P\ x \implies Q\ x) \implies (\square x.\ P\ x) \implies (\square x.\ Q\ x)$

We define a filter $\mathcal{F}$ as a predicate on predicates that satisfies all three of the above rules. (Note that we do not require filters to be *proper*; that is, we admit the trivial filter "for all x in {}" which holds for all predicates, including $\lambda x.\ \mathit{False}$.)

is-filter $::((\alpha \to \mathbb{B}) \to \mathbb{B}) \to \mathbb{B}$
is-filter $\mathcal{F} =$
  $\mathcal{F}\ (\lambda x.\mathit{True}) \wedge$
  $(\forall P, Q.\ \mathcal{F}\ (\lambda x.\ P\ x) \implies \mathcal{F}\ (\lambda x.\ Q\ x) \implies \mathcal{F}\ (\lambda x.\ P\ x \wedge Q\ x)) \wedge$
  $(\forall P, Q.\ (\forall x.\ P\ x \implies Q\ x) \implies \mathcal{F}\ (\lambda x.\ P\ x) \implies \mathcal{F}\ (\lambda x.\ Q\ x))$

We define the type α *filter* comprising all filters over the type α. The command **typedef** provides functions $\mathit{Rep}_{\mathit{filter}}$ and $\mathit{Abs}_{\mathit{filter}}$ to convert between α *filter* and $(\alpha \to \mathbb{B}) \to \mathbb{B}$; we use *eventually* $::(\alpha \to \mathbb{B}) \to \alpha$ *filter* $\to \mathbb{B}$ (defined as $\mathit{Rep}_{\mathit{filter}}$ with swapped argument order) to apply a filter to a predicate.

typedef α *filter* $= \{\mathcal{F} \mid$ *is-filter* $\mathcal{F}\}$

Note: For each filter $F :: \alpha$ *filter*, we will usually show only its characteristic equation *eventually* $P\ F \iff \mathcal{F}\ P$, leaving the raw definition $F = \mathit{Abs}_{\mathit{filter}}\ \mathcal{F}$ and the proof obligation *is-filter* $\mathcal{F}$ implicit.

Finer-than ordering. We define the ordering $F_1 \leq F_2$ to mean that filter F_1 is *finer than* F_2, i.e., $\forall P.\ \mathit{eventually}\ P\ F_2 \implies \mathit{eventually}\ P\ F_1$. For filters that represent bounded quantifiers, $\leq$ agrees with the subset order: "for all x in A" $\leq$ "for all x in B" iff $A \subseteq B$. This ordering also makes $\alpha\ \mathit{filter}$ into a complete lattice, with the trivial filter as the bottom element and $\forall$ as the top element.

$\square \leq \square \quad :: \quad \alpha\ \mathit{filter} \to \alpha\ \mathit{filter} \to \mathbb{B}$
$F_1 \leq F_2 \longleftrightarrow (\forall P.\ \mathit{eventually}\ P\ F_2 \implies \mathit{eventually}\ P\ F_1)$

Basic filters. On any linearly ordered type, we define filters *at-top* to mean "for sufficiently large y" or "as $y \longrightarrow +\infty$", and *at-bot* as "for sufficiently small y" or "as $y \longrightarrow -\infty$". We use *sequentially* as an abbreviation for *at-top* as a filter on the naturals.

lemma
$\mathit{eventually}\ P\ (\mathit{at\text{-}top} :: (\alpha :: \mathit{linorder})\ \mathit{filter}) \longleftrightarrow (\exists x.\ \forall y \geq x.\ P\ y)$
$\mathit{eventually}\ P\ (\mathit{at\text{-}bot} :: (\alpha :: \mathit{linorder})\ \mathit{filter}) \longleftrightarrow (\exists x.\ \forall y \leq x.\ P\ y)$

In the context of a topological space, we define *nhds* x as the *neighborhood filter*, which means "for all y in some open neighborhood of x".

lemma
$\mathit{eventually}\ P\ (\mathit{nhds}\ x) \longleftrightarrow (\exists U.\ \mathit{open}\ x \wedge x \in U \wedge (\forall y \in U.\ P\ y))$

The *principal filter* of a set B represent a bounded quantifier, i.e. "for all x in B". It is useful for constructing refinements of the neighborhood filter. We define *at* x *within* U as the *punctured* neighborhood filter, "for all $y \in U$ and $y \neq x$ in some neighborhood of x". We also define one-sided filters *at-left* and *at-right*. *at* x is an abbreviation for *at* x *within* $\mathcal{U}_\alpha$. $F_1 \sqcap F_2$ is the infimum of the filters F_1 and F_2.

lemma
$\mathit{eventually}\ P\ (\mathit{principal}\ S) \longleftrightarrow (\forall x \in S.\ P\ x)$

$\mathit{at}\ \square\ \mathit{within}\ \square \ :: \ (\alpha :: \mathit{topological\text{-}space}) \to \alpha\ \mathit{set} \to \alpha\ \mathit{filter}$
$\mathit{at\text{-}left}, \mathit{at\text{-}right} :: (\alpha :: \mathit{linorder\text{-}topology}) \to \alpha\ \mathit{filter}$

$\mathit{at}\ x\ \mathit{within}\ U = \mathit{nhds}\ x \sqcap \mathit{principal}\ (U \setminus \{x\})$
$\mathit{at\text{-}left}\ x = \mathit{at}\ x\ \mathit{within}\ ]\infty, x[$
$\mathit{at\text{-}right}\ x = \mathit{at}\ x\ \mathit{within}\ ]x, \infty[$

When we apply a function to the argument of each predicate in a filter we get a filter again. With *filtermap* f F we transform the filter F by a function f. We will shortly use it for expressing general limits.

lemma
$\mathit{eventually}\ P\ (\mathit{filtermap}\ f\ F) \longleftrightarrow \mathit{eventually}\ (\lambda x.\ P\ (f\ x))\ F$

lemma
$\mathit{filtermap}\ (\lambda x :: \mathbb{R}.\ -x)\ \mathit{at\text{-}top} = \mathit{at\text{-}bot}$

$\mathit{filtermap}\ (\lambda x :: \mathbb{R}.\ 1/x)\ \mathit{at\text{-}top} = \mathit{at\text{-}right}\ 0$

Limits. Filters can be used to express a general notion of limits. To illustrate this, we start with the usual epsilon-delta definitions of limits of functions and sequences on reals, and then incrementally generalize the definitions. Finally we end up with a single definition, parameterized over two filters, that can express diverse kinds of limits in arbitrary topological spaces. Here are the usual epsilon-delta definitions of limits for sequences and for functions at a point.

$$(y_n \longrightarrow L) = (\forall \epsilon > 0.\ \exists n_0.\ \forall n \geq n_0.\ |y_n - L| < \epsilon)$$
$$(\lim_{x \to a} f(x) = L) = (\forall \epsilon > 0.\ \exists \delta > 0.\ \forall x.\ 0 < |x - a| < \delta \implies |f(x) - L| < \epsilon)$$

The reader may recognize "$\exists n_0.\ \forall n \geq n_0$" as the filter *sequentially*. Also note that "$\exists \delta > 0.\ \forall x.\ 0 < |x - a| < \delta$" is equivalent to the punctured neighborhood filter $(\mathit{at}\ a)$. Therefore we can rewrite the above definitions as follows.

$$(y_n \longrightarrow L) = (\forall \epsilon > 0.\ \mathit{eventually}\ (\lambda n.\ |y_n - L| < \epsilon)\ \mathit{sequentially})$$
$$(\lim_{x \to a} f(x) = L) = (\forall \epsilon > 0.\ \mathit{eventually}\ (\lambda x.\ |f(x) - L| < \epsilon)\ (\mathit{at}\ a))$$

Already we can unify these two definitions by parameterizing over the filter. (This yields the same definition as the *tendsto* relation from HOL Light.)

$$(f \longrightarrow L)\ F = (\forall \epsilon > 0.\ \mathit{eventually}\ (\lambda x.\ |f(x) - L| < \epsilon)\ F) \tag{1}$$

We express many kinds of limits with $(f \longrightarrow x)\ F$ by instantiating F with various filters: *sequentially* for sequences, *at* a for a function at a point, *at-top* or *at-bot* for a function at $\pm\infty$, *at-left* a or *at-right* a for one-sided limits.

$$(x_n \longrightarrow L) = (x \longrightarrow L)\ \mathit{sequentially}$$
$$(\lim_{x \to a} f(x) = L) = (f \longrightarrow L)\ (\mathit{at}\ a)$$
$$(\lim_{x \to a^+} f(x) = L) = (f \longrightarrow L)\ (\mathit{at\text{-}right}\ a)$$
$$(\lim_{x \to -\infty} f(x) = L) = (f \longrightarrow L)\ \mathit{at\text{-}bot}$$

Up to now, we generalized how the limit is approached, but we can also generalize the right-hand side L. First we rewrite (1) using *filtermap*:

$$(f \longrightarrow L)\ F = (\forall \epsilon > 0.\ \mathit{eventually}\ (\lambda y.\ |y - L| < \epsilon)\ (\mathit{filtermap}\ f\ F))$$

This says that *filtermap* f F is eventually in every open neighborhood of L, which is equivalent to the following:

$$(f \longrightarrow L)\ F = (\mathit{filtermap}\ f\ F \leq \mathit{nhds}\ L)$$

Finally, we can generalize *nhds* L to an arbitrary filter G and obtain the generalized limit *LIM* x *in* F. $f\ x :> G$ (in Isabelle/HOL also written *filterlim* $f\ F\ G$).

$$\begin{aligned}
&\mathit{LIM}\ \square\ \mathit{in}\ \square.\ \square :> \square && :: (\alpha \to \beta) \to \alpha\ \mathit{filter} \to \beta\ \mathit{filter} \to \mathbb{B} \\
&\mathit{LIM}\ x\ \mathit{in}\ F.\ f\ x :> G && \iff \mathit{filtermap}\ f\ F \leq G \\
&(\square \longrightarrow \square)\ \square && :: (\alpha \to \beta) \to (\beta :: \mathit{topological\text{-}space}) \to \alpha\ \mathit{filter} \to \mathbb{B} \\
&(f \longrightarrow L)\ F && \iff \mathit{LIM}\ x\ \mathit{in}\ F.\ f\ x :> \mathit{nhds}\ L \\
&\square \longrightarrow \square && :: (\mathbb{N} \to \alpha) \to (\alpha :: \mathit{topological\text{-}space}) \to \mathbb{B} \\
&X \longrightarrow L && \iff (X \longrightarrow L)\ \mathit{sequentially}
\end{aligned}$$

This abstract notion of limit is only based on filters and does not even require topologies. Now we can express new limits (that are not expressible in HOL Light's library), e.g., *LIM x in at-bot.* $-x$:> *at-top* says that $-x$ goes to infinity as x approaches negative infinity, $\lim_{x \to -\infty} -x = \infty$.

For *filterlim* we can provide a composition rule for convergence. Further rules about e.g. elementary functions are available for normed vector spaces.

lemma
$(LIM\ x\ in\ F_1.\ f\ x :> F_2) \implies (LIM\ x\ in\ F_2.\ g\ x :> F_3) \implies$
$(LIM\ x\ in\ F_1.\ g\ (f\ x) :> F_3)$

We can prove e.g. $((\lambda x.\ exp\ (-1/x)) \longrightarrow 0)\ (at\text{-}right\ 0)$ from $(exp \longrightarrow 0)\ at\text{-}bot$, *LIM x in at-top.* $-x$:> *at-bot*, and *LIM x in at-right* $0.\ 1/x$:> *at-top*.

On the order topology, a function converges to x iff for all upper and lower bounds of x the function is eventually in these bounds.

lemma fixes $f :: \alpha \to (\beta :: linorder\text{-}topology)$
shows $(f \longrightarrow x)\ F \longleftrightarrow (\forall b > x.\ eventually\ (\lambda x.\ f\ x < b)\ F) \wedge$
$(\forall b < x.\ eventually\ (\lambda x.\ b < f\ x)\ F)$

Filters vs nets. As an alternative to filters, limits may also be defined using *nets*, which generalize sequences. While sequences are indexed by natural numbers, a net may be indexed by any directed set. Like filters, nets support an "eventually" operator: N eventually satisfies P iff $\exists x.\ \forall y \geq x.\ P(N(y))$.

In terms of formalizing limits and convergence, filters and nets are equally expressive. However, nets are not as convenient to formalize in HOL. A type α *net* of all nets over α does not work; nets require a second parameter type to allow arbitrary index sets.

4.3 Continuity

Continuity of a function f at a filter F says that the function converges on F towards its value $f\ x$ where F converges to x. We use filters to unify continuity at a point, continuity from left, continuity from right etc. With *Lim* $F\ (\lambda x.\ x)$ we select the convergence point of the filter F with definite choice. To have a unique value for x, the domain of the function needs to be a T_2-space.

Lim :: $\alpha\ filter \to (\alpha \to \beta) \to (\beta :: t2\text{-}space)$
Lim $F\ f$ = *THE* $L.\ (f \longrightarrow L)\ F$

continuous :: $\alpha\ filter \to (\alpha :: t2\text{-}space \to \beta :: topological\text{-}space) \to \mathbb{B}$
continuous $F\ f \longleftrightarrow (f \longrightarrow f\ (Lim\ F\ (\lambda x.\ x)))\ F$

This is similar to the definition in HOL Light, but generalized to topological spaces instead of Euclidean spaces.

Often a function needs to be continuous not only at a point, but on a set. For this we introduce *continuous-on*. Its domain is not restricted to a T_2-space.

continuous-on :: $\alpha\ set \to (\alpha :: topological\text{-}space \to \beta :: topological\text{-}space) \to \mathbb{B}$
continuous-on $S\ f \longleftrightarrow \forall x \in S.\ (f \longrightarrow f\ x)\ (at\ x\ within\ S)$

4.4 Compactness

An important topological concept is *compactness* of sets. There are different characterizations of compactness: sequential compactness, cover compactness and countable cover compactness. Unfortunately these characterizations are not equal on each topological space, but we will show in which type classes they are.

First we introduce *cover compactness*; it does not require any other topological concepts besides open sets. A *cover* of a set U is a set of open sets whose union is a superset of U. A set U is compact iff for each cover C there exists a finite subset of C which is also a cover:

compact :: $(\alpha :: \textit{topological-space})\ \textit{set} \to \mathbb{B}$
compact $U \longleftrightarrow$
$(\forall C.\ (\forall c \in C.\ \textit{open}\ c) \wedge U \subseteq \bigcup C \implies \exists D \subseteq C.\ \textit{finite}\ D \wedge U \subseteq \bigcup D)$

Topology usually talks about compact spaces U, where the open sets are restricted to the topological space U, which would be $\mathcal{U}_\alpha$ in our case. This would not be very helpful, we would need to define a type for each compact space. Luckily, cover compactness works also with covers which are proper supersets, which will be the case when we use it.

Cover compactness can be expressed using filters. A space U is compact iff for each proper filter on U exists an $x \in U$, s.t. a neighborhood of x is contained in the filter.

lemma
compact $U \longleftrightarrow$
$(\forall F > \bot.\ \textit{eventually}\ (\lambda x.\ x \in U)\ F \implies (\exists x \in U.\ \textit{nhds}\ x \sqcap F > \bot))$

Similarly to cover compactness we define *countably-compact*, where a set is compact iff for each *countable* cover exists a finite subcover. Then *compact* obviously implies *countably-compact*, the other direction holds at least for a *second-countable-topology* space.

With limits and filters, characterizations of compactness apart from cover or countable compactness are possible. One often used characterization of compactness is *sequential compactness*, where for each sequence on the compact space U, there exists a subsequence converging in U (a subsequence of X is defined by selecting increasing indices into X, *subseq* r states that r is strictly increasing).

seq-compact :: $\alpha\ \textit{set} \to \mathbb{B}$
seq-compact $U \longleftrightarrow$
$(\forall X.\ (\forall n.\ X\ n \in U) \implies \exists r.\ \textit{subseq}\ r \wedge \exists x \in U.\ (X \circ r) \longrightarrow x)$

On a first countable topology sequential equals countable cover compactness. On a second countable topology sequential, countable cover, and cover compactness are equal.

lemma fixes $U :: (\alpha :: \textit{first-countable-topology})\ \textit{set}$
shows *countably-compact* $U \longleftrightarrow$ *seq-compact* U

lemma fixes $U :: (\alpha :: \textit{second-countable-topology})\ \textit{set}$
shows *compact* $U \longleftrightarrow$ *seq-compact* U
shows *compact* $U \longleftrightarrow$ *countably-compact* U

5 Mathematical Analysis

Analysis works with infinite sequences and limits and develops concepts like differentiation and integration. As seen in the previous section, limits have been formalized generically for topological spaces. The formalization leading to differentiation and integration has largely been ported from Harrison's formalization in HOL Light [4] for the type $\mathbb{R}^n$. In this section, we present the generalization to our hierarchy of type classes. Following Fig. 1, we start with the type classes for metric spaces and then present the type classes for vector spaces, which culminate in Euclidean spaces.

5.1 Metric Spaces

Metric spaces are specializations of topological spaces: while topological spaces talk about *nearness*, metric spaces require to explicitly give a *distance* between elements. This distance then induces a notion of *nearness*: a set is open iff for every element in that set, one can give a distance within which every element is *near*, i.e. in the open set. The following type class formalizes open sets induced by a distance:

class *open-dist* = **fixes** *open* :: $\alpha\ \mathit{set} \to \mathbb{B}$ **and** *dist* :: $\alpha \to \alpha \to \mathbb{R}$
  assumes *open* $U \longleftrightarrow (\forall x \in U.\ \exists e > 0.\ \forall y.\ \mathit{dist}\ x\ y < e \implies y \in U)$

If the distance is a metric, it induces a particular topological space, namely a metric space. It is a first countable space and satisfies the Hausdorff separation property, i.e. it is actually a T_2-space.

class *metric-space* = *open-dist* +
  assumes *dist* $x\ y = 0 \longleftrightarrow x = y$ **and** *dist* $x\ y \leq$ *dist* $x\ z +$ *dist* $y\ z$
instance *metric-space* $\subseteq$ *t2-space*, *first-countable-topology*

One aspect that makes real numbers an interesting metric space is the fact that they are *complete*, which means that every sequence where the elements get arbitrarily close converges. Such a sequence is called *Cauchy sequence*, and a metric space is complete iff every Cauchy sequence converges.

Cauchy :: $(\mathbb{N} \to \alpha ::$ *metric-space*$) \to \mathbb{B}$
Cauchy $X \longleftrightarrow (\forall e > 0.\ \exists M.\ \forall m, n \geq M.$ *dist* $(X\ m)\ (X\ n) < e)$

complete :: $(\alpha ::$ *metric-space*$)\ \mathit{set} \to \mathbb{B}$
complete $U \longleftrightarrow (\forall X.\ (\forall i.\ X\ i \in U) \wedge$ *Cauchy* $X \implies \exists x \in U.\ X \longrightarrow x)$

class *complete-space* = *metric-space* + **assumes** *complete* $\mathcal{U}_\alpha$

We have generalized Harrison's formalization of the Banach fixed point theorem to metric spaces and we completed a characterization of compactness on metric spaces with total boundedness: compact sets are the complete ones that can, for every $e > 0$, be covered by a finite number of balls with radius e.

lemma $\forall U ::$ ($\alpha ::$ *metric-space*) *set*. *compact* $U \longleftrightarrow$
  complete $U \wedge (\forall e > 0.\ \exists T.$ *finite* $T \wedge U \subseteq \bigcup_{t \in T} \{s \mid$ *dist* $s\ t < e\})$

One instance of complete metric spaces is the type of finite maps $\alpha \rightharpoonup_f (\beta :: \textit{complete-space})$: the distance of two finite maps f, g with domains F, G is given by $\max_{i \in F \cup G} (\textit{dist}\ (f\ i)\ (g\ i)) + (\textbf{if}\ F = G\ \textbf{then}\ 0\ \textbf{else}\ 1)$. Then every Cauchy sequence eventually stabilizes at one particular finite domain and then converges uniformly. Another example is the type of bounded continuous functions $(\alpha :: \textit{topological-space}) \to_{bc} (\beta :: \textit{complete-space})$. Equipped with the supremum distance, they form a complete metric space.

Heine-Borel spaces. One can provide the convenient characterization that compact sets are exactly the bounded and closed sets on a metric space if the additional assumption that bounded sequences possess a convergent subsequence holds. We summarize this assumption in a type class, which allows for convenient access to the characterization and the theorems it implies. Euclidean spaces like $\mathbb{R}$, $\mathbb{C}$ and $\mathbb{R}^n$ are examples of instances.

class *heine-borel* = *metric-space* +
  assumes *bounded* $(\bigcup_x \{X\ x\}) \implies \exists x, r.$ *subseq* $r \wedge (X \circ r) \longrightarrow x$
instance *heine-borel* $\subseteq$ *complete-space*

lemma $\forall U ::$ ($\alpha ::$ *heine-borel*) *set. compact* $U \iff$ *bounded* $U \wedge$ *closed* U

5.2 Vector Spaces

One aspect that is often abstracted away from products of real numbers is their property of being a vector space, i.e. a space where addition and scaling can be performed. Let us present in this section the definition of vector spaces, normed vector spaces, and how derivatives are generalized for normed vector spaces.

Definition. Usually, a vector space is defined on an Abelian group of vectors V, which can be scaled with elements of a field F, and where distributive and compatibility laws need to be satisfied by scaling and addition. The type class based approach restricts the number of type variables to one; we therefore use locales (Isabelle's module system for dealing with parametric theories [3]) to abstractly reason about vector spaces with arbitrary combinations of F and V (which may be of different types). We define the type class *real-vector* for the common usage of $\mathbb{R}$ for the field F: the type class *ab-group-add*, which formalizes an Abelian group, provides the operations for addition and additive inverse for the type of vectors α (subtraction is defined in terms of these operations).

class *real-vector* = *ab-group-add* + **fixes** $\cdot_R :: \mathbb{R} \to \alpha \to \alpha$
  assumes $r \cdot_R (a + b) = r \cdot_R a + r \cdot_R b$ **and** $(r + q) \cdot_R a = r \cdot_R a + q \cdot_R a$
    and $r \cdot_R (q \cdot_R a) = (r \cdot q) \cdot_R a$ **and** $1 \cdot_R a = a$

A generalization of the length of a vector of real numbers is given by the norm in a vector space. The norm induces a distance in a vector space. Similar

to *open-dist*, which describes how *dist* induces *open* sets, we describe here how the norm induces the distance.

class *dist-norm* = **fixes** *norm* :: $\alpha \to \mathbb{R}$ **and** $-$ :: $\alpha \to \alpha \to \alpha$
  assumes *dist* x y = *norm* $(x - y)$

A *normed vector space* is then defined as a vector space *real-vector* together with the usual assumptions of a separating and positively scalable norm, for which the triangle equality holds. The distance for the instantiation as metric space and open sets for the topology are induced by *dist-norm* and *open-dist*, respectively. Then every normed vector space is a metric space.

class *real-normed-vector* = *real-vector* + *dist-norm* + *open-dist* +
  assumes *norm* $x = 0 \Longleftrightarrow x = 0$ **and** *norm* $(r \cdot_R x) = |r| \cdot$ *norm* x
    and *norm* $(x + y) \leq$ *norm* $x +$ *norm* y
instance *real-normed-vector* $\subseteq$ *metric-space*

We define a filter to describe that the norm tends to infinity (*at-infinity* = *filtermap norm at-top*). We have lemmas about limits of vector space operations – for example *LIM* x *in* F. $f\ x + g\ x :> G$ for $G =$ *nhds* L (if f and g converge) or $G =$ *at-infinity* (if f or g tend to infinity) – and hence continuity.

Complete normed vector spaces are called Banach spaces; we provide an extra type class for them. For example bounded continuous functions (α :: *topological-space*) $\to_{bc}$ (β :: *real-normed-vector*) equipped with pointwise addition and scaling form a Banach space.

class *banach* = *complete-space* + *real-normed-vector*

Derivatives. The HOL Light formalization includes derivatives of functions from $\mathbb{R}^n$ to $\mathbb{R}^m$. This derivative is a linear mapping, it is called Fréchet derivative or total derivative, and its matrix is called the Jacobian matrix. Our type class based formalization allows us to generalize (in accordance with textbook mathematics) the notion of Fréchet derivative to arbitrary normed vector spaces *real-normed-vector*, where the derivative is a *bounded* linear approximation. The limit may be approached from within an arbitrary set s:

bounded-linear :: (α :: *real-normed-vector* $\to$ β :: *real-normed-vector*) $\to \mathbb{B}$
bounded-linear $f' \Longleftrightarrow (f'\ (x + y) = f'\ x + f'\ y \wedge f'\ (a \cdot_R x) = a \cdot_R (f'\ x) \wedge$
  $(\exists K.\ \forall x.$ *norm* $(f'\ x) \leq K \cdot$ *norm* $x))$

FDERIV $\square\ \square : \square :> \square$ :: $(\alpha \to \beta) \to \alpha \to \alpha$ *set* $\to (\alpha \to \beta)$
FDERIV $f\ x : s :> f' \Longleftrightarrow$ (*bounded-linear* $f' \wedge$
  $((\lambda y.$ *norm* $(f\ y - f\ x - f'\ (y - x))/$*norm* $(y - x)) \longrightarrow 0)$ (*at* x *within* s))

We have generalized Harrison's results about derivatives of arithmetic operations, and the chain rule for differentiation to *real-normed-vector* spaces.

We provide a set of rules *FDERIV-eq-intros* that allows to compute derivatives: each of the rules assumes composition of a differentiable function with an

additional function and matches a variable to the derivative, which has to be solved by Isabelle's rewrite engine. Consider e.g., the following rule where the first assumption has to be solved by a repeated application of *FDERIV-eq-intros* and the second assumption needs to be solved by the simplifier:

lemma
  assumes *FDERIV* $f\ x : s :> f'$ **and** $(\lambda x.\ r \cdot_R (f'\ x)) = D$
  shows *FDERIV* $(\lambda x.\ r \cdot_R (f\ x))\ x : s :> D$

Algebraic vector spaces. Further specializations of (normed) vector spaces are available by including multiplication of vectors for a *real-normed-algebra* or *real-normed-field*. The only instances currently used are real and complex numbers $\mathbb{R}$ and $\mathbb{C}$ so we will not go into more detail here.

5.3 Euclidean Spaces

Another abstraction with geometric intuition is given by an *inner product* on normed vector spaces: while the norm can be interpreted as the length of a vector, the inner product can be used to describe the angle between two vectors together with their lengths (the cosine of the angle is the inner product divided by the product of the lengths). *dist-norm* and *open-dist* specify the induced metric and topology. The inner product is used to induce a norm. An inner product is a commutative bilinear operation $\bullet$ on vectors, for which $0 \leq x \bullet x$ holds with equality iff $x = 0$.

class *real-inner* = *real-vector* + *dist-norm* + *open-dist* +
  fixes $\bullet :: \alpha \to \alpha \to \mathbb{R}$
  assumes *norm* $x = \sqrt{x \bullet x}$ **and** $x \bullet y = y \bullet x$
    and $(x + y) \bullet z = x \bullet z + y \bullet z$ **and** $(r \cdot_R x) \bullet y = r \cdot_R (x \bullet y)$
    and $0 \leq x \bullet x$ **and** $x \bullet x = 0 \Longleftrightarrow x = 0$
instance *real-inner* $\subseteq$ *real-normed-vector*

For vector spaces with inner products, there is for example orthogonality of vectors formalized, i.e. vectors with inner product zero.

Finally, we introduce *Euclidean spaces* as spaces with inner product and a finite coordinate basis, that means a finite set of orthogonal vectors of length 1. In addition, the zero vector is characterized by zero "coordinates" with respect to the basis. Any Euclidean space is a Banach space with a perfect second countable topology and satisfies the Heine-Borel property:

class *euclidean-space* = *real-inner* +
  fixes *Basis* :: α *set*
  assumes *finite Basis* **and** *Basis* $\neq \emptyset$ **and** $(\forall u \in \mathit{Basis}.\ x \bullet u = 0) \Longleftrightarrow x = 0$
    and $u \in \mathit{Basis} \implies v \in \mathit{Basis} \implies u \bullet v =$ **if** $u = v$ **then** 1 **else** 0
instance *euclidean-space* $\subseteq$ *perfect-space*, *second-countable-topology*, *banach*, *heine-borel*

Linear algebra has been ported from Harrison's basic formalization, which includes notions of independence and span of a set of vectors. We prove for example independence of the basis and that the basis spans the whole Euclidean space.

For functions between *euclidean-space*s, we have ported from HOL Light that the Fréchet derivative can be described as the Jacobian matrix, the mean value theorem, and Brouwer's fixed point theorem, which allows to prove that the derivative of an inverse function is the inverse of the derivative.

Moreover we have ported Harrison's formalization of the gauge (or Heinstock-Kurzweil) integral and related properties (linearity, monotone and dominated convergence, and the fundamental theorem of calculus).

Instances for the type class *euclidean-space* are real numbers $\mathbb{R}$, complex numbers $\mathbb{C}$ and the Cartesian types $\mathbb{R}^\alpha$ where $\alpha :: \mathit{finite}$ (which are isomorphic to $\alpha \to \mathbb{R}$). One advantage of our type class based approach is that we can use the same formalizations of Euclidean space (e.g. of the integral) for the different types, whereas in HOL Light, one needs to project e.g. from $\mathbb{R}^1$ to $\mathbb{R}$.

5.4 Real Numbers

The type of real numbers $\mathbb{R}$ is a special instance of Euclidean spaces; some parts of our formalization are only available for this case. For a function on real numbers, one usually thinks of the "derivative" as the slope of the function (or of the linear approximation), we therefore use the constant *DERIV*:

$$\mathit{DERIV}\ \square\ \square :> \square \quad :: \quad (\mathbb{R} \to \mathbb{R}) \to \mathbb{R} \to \mathbb{R}$$
$$\mathit{DERIV}\ f\ x :> f' \iff \mathit{FDERIV}\ f\ x : \mathcal{U}_{\mathbb{R}} :> (\lambda x.\ f' \cdot x)$$

It turns out that the general formalization of limits with filters allows to conveniently express e.g. l'Hôpital's rules in Isabelle/HOL: if the denominator of a quotient tends to infinity, then the quotient tends to the quotient of the derivatives of nominator and denominator (if they exist).

lemma
  fixes $f\ g :: \mathbb{R} \to \mathbb{R}$
  assumes $\mathit{LIM}\ x\ \mathit{in}\ \mathit{at\text{-}top}.\ g\ x :> \mathit{at\text{-}top}$
    and $\mathit{eventually}\ (\lambda x.\ g'\ x \neq 0)\ \mathit{at\text{-}top}$
    and $\mathit{eventually}\ (\lambda x.\ \mathit{DERIV}\ f\ x :> f'\ x \wedge \mathit{DERIV}\ g\ x :> g'\ x)\ \mathit{at\text{-}top}$
    and $((\lambda x.\ f'\ x / g'\ x) \longrightarrow L)\ \mathit{at\text{-}top}$
  shows $((\lambda x.\ f\ x / g\ x) \longrightarrow L)\ \mathit{at\text{-}top}$

6 Summary

We used the type class mechanism in Isabelle/HOL to formalize a hierarchy of spaces often used in mathematical analysis: starting with topological spaces, over metric spaces to Euclidean spaces. As in mathematics, the intention of using a hierarchical structure is to share definitions and proofs.

The reuse occurs for the introduction of extended reals $\overline{\mathbb{R}}$, the spaces of bounded continuous functions $\alpha \to_{bc} \beta$, and finite maps $\alpha \to_f \beta$. The extended reals $\overline{\mathbb{R}}$ need to exploit the topological type classes, as they do not form a metric space. The bounded continuous function space $\alpha \to_{bc} \beta$ is a Banach space. Immler and Hölzl [7] apply them to the Banach fixed point theorem to prove the existence of unique solutions of ordinary differential equations. Immler [6] uses finite maps $\alpha \to_f \beta$ to construct stochastic processes via a projective limit.

Our approach still has the problem that all operations are defined on $\mathcal{U}_\alpha$. The usage of finite maps $\alpha \to_f \beta$ in [6] illustrates this. We need a metric space whose dimensionality depends on a variable inside of a proof. Luckily, the disjoint union of metric spaces can be extended to a metric space. But such a trick is not always applicable, i.e. this is not possible for normed vector spaces. This can only be avoided by adding a carrier set to each operation or by extending HOL.

Despite the last point, our work shows that Isabelle's type class system suffices to describe many abstract structures occurring in mathematical analysis.

Acknowledgements

We want to thank John Harrison and his colleagues for the development of HOL Light's multivariate analysis. Further we want to thank Amine Chaieb and Robert Himmelmann for porting it to Isabelle/HOL.

References

1. Fleuriot, J.D., Paulson, L.C.: Mechanizing nonstandard real analysis. LMS Journal of Computation and Mathematics 3, 140–190 (2000)
2. Haftmann, F., Wenzel, M.: Constructive Type Classes in Isabelle. In: Altenkirch, T., McBride, C. (eds.) TYPES 2006, LNCS, vol. 4502, pp. 160–174 (2007)
3. Haftmann, F., Wenzel, M.: Local theory specifications in Isabelle/Isar. In: Berardi, S., Damiani, F., De'Liguoro, U. (eds.) TYPES 2008, LNCS, vol. 5497 (2009)
4. Harrison, J.: A HOL theory of Euclidean space. In: Hurd, J., Melham, T. (eds.) TPHOLs 2005. LNCS, vol. 3603, pp. 114–129 (2005)
5. Hölzl, J., Heller, A.: Three chapters of measure theory in Isabelle/HOL. In: van Eekelen, M.C.J.D., Geuvers, H., Schmaltz, J., Wiedijk, F. (eds.) Interactive Theorem Proving (ITP 2011). LNCS, vol. 6898, pp. 135–151 (2011)
6. Immler, F.: Generic construction of probability spaces for paths of stochastic processes in Isabelle/HOL. Master's thesis, TU München (Oct 2012)
7. Immler, F., Hölzl, J.: Numerical analysis of ordinary differential equations in Isabelle/HOL. In: Beringer, L., Felty, A. (eds.) Interactive Theorem Proving (ITP 2012), LNCS, vol. 7406, pp. 377–392 (2012)
8. Joshi, K.D.: Introduction to General Topology. John Wiley and Sons (1983)
9. Lester, D.R.: Topology in PVS: continuous mathematics with applications. In: Second workshop on Automated formal methods. pp. 11–20. AFM '07 (2007)
10. Spitters, B., van der Weegen, E.: Type classes for mathematics in type theory. MSCS, 'Interactive theorem proving and the form. of math.' 21, 1–31 (2011)