Big Proof 2019 Workshop

A Formal Classical Proof of Hahn-Banach in Coq

Marie Kerjean & Assia Mahboubi

Inria Nantes , LS2N

Based Mathcomp *and* MathComp Analysis *libraries,*

developed by Reynald Affeldt, Cyril Cohen, Assia Mahboubi, Damien Rouhling,

Pierre-Yves Strub

Big Proof 2019 Workshop

A user experience of Mathematical Components Analysis

Marie Kerjean & Assia Mahboubi

Inria Nantes , LS2N

Based Mathcomp *and* MathComp Analysis *libraries,*

developed by Reynald Affeldt, Cyril Cohen, Assia Mahboubi, Damien Rouhling,

Pierre-Yves Strub

Disclaimer

- I am relatively new to Coq, and completely new to ssreflect and Mathcomp Libraries.

```
case: z {zmax} gP => [c [_ _ bp _]] /= gP; apply/bp/gP .
```

- This proof is a test for the Mathematical Components Analysis libraries.

  https://github.com/math-comp/analysis/blob/hb/hahn_banach.v

- This talk is an excuse to speak about the Mathcomp Analysis project.

Lemma 001 of functional analysis

```
Theorem HB_geom_normed (V : normedtype R) (F : submod V) (f : {scalar F}) :
 (forall x , F x -> continuous_at x f)
 -> exists g : {scalar V} , ( continuous g ) /\ ( forall x, F x -> (f x = g x) )
```

Hahn-Banach before normed spaces

```
Variables (R : realFieldType) (V : lmodType R)
 (p : convex R) (F : submod V).

Theorem HahnBanach (f : scalar V) :
 ( forall x, F x -> ( f x <= p x )) ->
 exists g : {scalar V},
  (forall x, g x <= p x) /\ (forall x, F x -> g x = f x).
```

Textbook Proof:

- Extending f to a linear function $F \oplus \mathbb{R}v$ bounded by $p$is follows from the convexity of p and the linearity required for the extension.
- Extending f to the whole space V is done through Zorn's lemma.

Hahn-Banach before normed spaces

```
Variables (R : realFieldType) (V : lmodType R)
 (p : convex R) (F : submod V).

Theorem HahnBanach (f : scalar V) :
 ( forall x, F x -> ( f x <= p x )) ->
 exists g : {scalar V},
  (forall x, g x <= p x) /\ (forall x, F x -> g x = f x).
```

Textbook Proof: [Linear Algebra]

- Extending f to a linear function $F \oplus \mathbb{R}v$ bounded by $p$is follows from the convexity of p and the linearity required for the extension.
- Extending f to the whole space V is done through Zorn's lemma.

Hahn-Banach before normed spaces

```
Variables (R : realFieldType) (V : lmodType R)
 (p : convex R) (F : submod V).

Theorem HahnBanach (f : scalar V) :
 ( forall x, F x -> ( f x <= p x )) ->
 exists g : {scalar V},
  (forall x, g x <= p x) /\ (forall x, F x -> g x = f x).
```

Textbook Proof: [Linear Algebra]

- Extending f to a linear function $F \oplus \mathbb{R}v$ bounded by $p$is follows from the convexity of p and the linearity required for the extension.

  [real analysis and classical reasoning]

- Extending f to the whole space V is done through Zorn's lemma.

  [Axiome of Choice]

Hahn-Banach before normed spaces

```
Variables (R : realFieldType) (V : lmodType R)
 (p : convex R) (F : submod V).

Theorem HahnBanach (f : scalar V) :
 ( forall x, F x -> ( f x <= p x )) ->
 exists g : {scalar V},
  (forall x, g x <= p x) /\ (forall x, F x -> g x = f x).
```

This is my favorite **existence theorem**, with countless applications.

Existing Formalisations

- Existing Formalisations in Mizar [1993] and HoL/Isabelle [2000]
- Investigation on a constructive version in point-free topology by Coquand, Negri and Cederquist.

Mathematical-Components

A library in Coq constructed for the formalization of Feit-Thompson theorem [Gonthier and al., 2012].

Libraries for algebra with a strong taste for finite dimension:

- Finite Group Theory.
- Ring and modules.
- Finites dimensional vector spaces.
- Matrixes and Polynomials

Mathematical-Components : a peak into ssralg

[Graphs of scalar functions]

```
Variable (R : ringType) ( V : lmodType R).

Definition linear_rel (f : V -> R -> Prop) :=
  forall v1 v2 l r1 r2, f v1 r1 -> f v2 r2 -> f (v1 + l *: v2) (r1 + l * r2).

 Variable (f : V -> R -> Prop).
 Hypothesis lrf : linear_rel f.

 Lemma linrel_00 x r : f x r -> f 0 0.
 Proof.
 suff -> : f 0 0 = f (x + (-1) *: x) (r + (-1) * r) by move=> h; apply: lrf.
 by rewrite scaleNr mulNr mul1r scale1r !subrr.
 Qed.
```

Ssreflect : un peu, beaucoup, à la folie

- Ssreflect is a set of tactincts and notations, used extensively in the Mathcomp libraries.
- MathComp Proofs are often written in an **imperative minimal style** : easier to maintain.
-

Ssreflect : un peu, beaucoup, à la folie

- Ssreflect is a set of tactícts and notations, used extensively in the Mathcomp libraries.
- MathComp Proofs are often written in an **imperative minimal style** : easier to maintain.
- The user can choose to use it **as much as she likes**.

```
Lemma linrel_00 x r : f x r -> f 0 0.
Proof.
suff -> : f 0 0 = f (x + (-1) *: x) (r + (-1) * r) by move=> h; apply: lrf.
by rewrite scaleNr mulNr mul1r scale1r !subrr.
Qed.

Lemma long_linrel_00 x r : f x r -> f 0 0.
Proof.
have H : f 0 0 = f (x + (-1) *: x) (r + (-1) * r).
  rewrite scaleNr
  rewrite mulNr
  by rewrite mul1r scale1r subrr subrr. (* unfold if you want *)
intro h. (* move => h*)
apply: lrf.
by [].
Qed.
```

Mathematical-Components- Analysis

Enough of Algebra.

Analysis !

Why ?

- Because it's fun.
- Because it is needed for verification.

  [P.-Y. Strub - EasyCrypt - probabilistic computation].
- Because it is needed for verifying robotics .

  [R. Affeldt, C. Cohen, D. Rouhling - CoqRobot - Lassalle Invariance].

Mathematical-Components- Analysis

Fact

- Formalisation in Coq has been influenced a lot by the constructive point of vue on mathematics - because it can.

Mathematical-Components- Analysis

Opinion

- Formalisation in Coq has been **too much** influenced by the constructive point of vue on mathematics - because it can.

Mathematical Components Analysis : CIC + + Axiome of Choice + Excluded middle + Functional Extensionality + Propositional Equality

This library reinterprets and extends the **Coquelicot** project.

[Boldo and al, 2015]

Libraries

- Reals.
- Topology, Derivation.
- Norms and Complete spaces.
- Landau Notations and tactics,
- Soon* : Complex analysis and Lebesgue integration

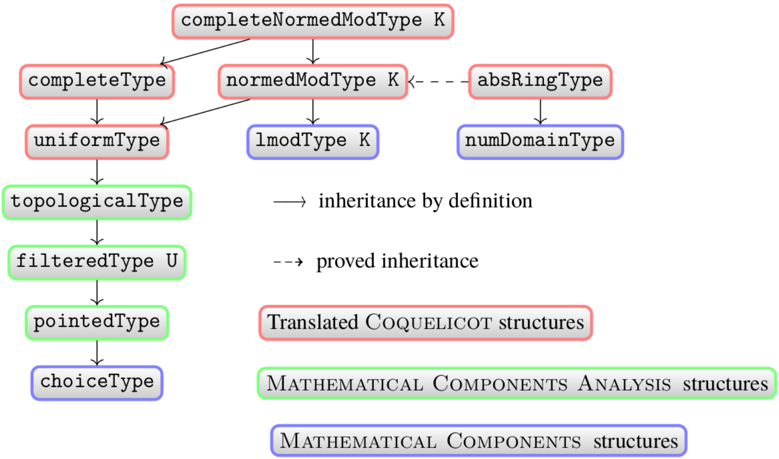

Figure: MATHEMATICAL COMPONENTS ANALYSIS hierarchy

[Cohen 2018]

Mathematical-Components- Analysis

```
Lemma tychonoff (I : eqType) (T : I -> topologicalType)
  (A : forall i, set (T i)) :
  (forall i, compact (A i)) ->
  @compact (product_topologicalType T)
[set f : forall i, T i | forall i, A i (f i)].

Variable ( M : uniformType).

Lemma flim_ballP {F} {FF : Filter F} (y : M) :
  F --> y <-> forall eps : R, 0 < eps -> \forall y' \near F, ball y eps y'.
Proof. by rewrite -filter_fromP !locally_simpl /=. Qed.

Variable ( U V : normedspace ) .

Lemma linear_for_continuous (f: {linear U -> V }) :
  (f : _ -> _) =O_ (0 : U) (cst (1 : R^o)) -> continuous f.
```

All about $\mathbb{R}$

- $\mathbb{R}$ in coq/reals.v : an axiomatic definition used by Coquelicot.

  ```
  Variable ( x : R). Check `|x|.
  ```

- $\mathbb{R}$ in analysis/reals.v : a realArchiType of mathcomp with a least upper bound operator.

  ```
  Variable (R : realType) ( x : R). Check `|x|.
  ```

- $\mathbb{R}$ in analysis/normedtype.v : a normed type when seen as R^0.

  ```
  Variable ( x : R^o). Check `|[x]|.
  ```

These features will be corrected soon but meanwhile some transports lemmas are needed.

```
Lemma absRE : forall x : R, abs x = normrr x
```

Hahn-Banach before normed spaces

```
Variables (R : realFieldType) (V : lmodType R)
 (p : convex R) (F : submod V).

Theorem HahnBanach (f : scalar V) :
 ( forall x, F x -> ( f x <= p x )) ->
 exists g : {scalar V},
  (forall x, g x <= p x) /\ (forall x, F x -> g x = f x).
```

Textbook Proof:

- Extending f to a linear function $F \oplus \mathbb{R}v$ bounded by $p$is follows from the convexity of p and the linearity required for the extension.
- Extending f to the whole space V is done through Zorn's lemma.

Hahn-Banach before normed spaces

```
Variables (R : realFieldType) (V : lmodType R)
 (p : convex R) (F : submod V).

Theorem HahnBanach (f : scalar V) :
 ( forall x, F x -> ( f x <= p x )) ->
 exists g : {scalar V},
  (forall x, g x <= p x) /\ (forall x, F x -> g x = f x).
```

Textbook Proof: [Linear Algebra]

- Extending f to a linear function $F \oplus \mathbb{R}v$ bounded by $p$is follows from the convexity of p and the linearity required for the extension.
- Extending f to the whole space V is done through Zorn's lemma.

Hahn-Banach before normed spaces

```
Variables (R : realFieldType) (V : lmodType R)
 (p : convex R) (F : submod V).

Theorem HahnBanach (f : scalar V) :
 ( forall x, F x -> ( f x <= p x )) ->
 exists g : {scalar V},
  (forall x, g x <= p x) /\ (forall x, F x -> g x = f x).
```

Textbook Proof: [Linear Algebra]

- Extending f to a linear function $F \oplus \mathbb{R}v$ bounded by $p$is follows from the convexity of p and the linearity required for the extension.

  [real analysis and classical reasoning]

- Extending f to the whole space V is done through Zorn's lemma.

  [Axiome of Choice]

Hahn-Banach Finally

Reasoning on the graphs of linear function which are bounded by a convex function and which extends f.

```
Definition spec (g : V -> R -> Prop) :=
  [/\ functional g, linear_rel g, maj_by p g & forall v, F v -> g v (f v) ].

Record zorn_type : Type := ZornType
  {carrier : V -> R -> Prop; specP : spec carrier}.

Lemma domain_extend (z : zorn_type) v :
    exists2 ze : zorn_type, (zorn_rel z ze) & (exists r, (carrier ze) v r).

Theorem HahnBanach : exists g : {scalar V},
 (forall x, g x <= p x) /\ (forall x, F x -> g x = f x).
```

chosing scalar V was maybe a bad choice.

Looking for Lemmas

```
Search (exists _ , _) "Hahn".
```

- Searching via patterns.

  ```
  Search _ (exists _ , _) (continuous _) in topology.
  ```

- Searchin via names (next slide).

  ```
  Search "HB".
  Search "my_favorite_thm".
  Search "why_on_earth_isnt_this_automated".
  ```

Looking for Lemmas

```
Search (exists _ , _) "Hahn".
```

- Searching via patterns.

  ```
  Search _ (exists _ , _) (continuous _) in topology.
  ```

- Searchin via names (next slide).

  ```
  Search "HB".
  Search "my_favorite_thm".
  Search "why_on_earth_isnt_this_automated".
  ```

- Combine the two.

Looking for Lemmas

```
Search (exists _ , _) "Hahn".
```

- Searching via patterns.

  ```
  Search _ (exists _ , _) (continuous _) in topology.
  ```

- Searchin via names (next slide).

  ```
  Search "HB".
  Search "my_favorite_thm".
  Search "why_on_earth_isnt_this_automated".
  ```

- Combine the two.

- Ask by mail / gitter.

Naming Convention

You should expect the name of hte main statement in the lemma.

```
normedModType_hausdorff : forall (K : absRingType) (V : normedModType K),
    hausdorff V
```

A list of suffix abbreviation :

`A`: associativity, `C` : commutativity, `D`: addition, `E` : definition elimination, characteristic properties (often reflection lemmas), `Z` : module/vector space scaling.

```
Lemma normmZ : forall (K : absRingType) (V : normedModType K) (l : K) (x : V),
      `|[l *: x]| = `|l|%real * `|[x]| .

Lemma normr_ge0 : forall (R : numDomainType) (x : R), 0 <= `|x| .

Lemma flim_normP : forall (K : absRingType) (V : normedModType K) (F :
    classical_sets.set (classical_sets.set V)),
        Filter F -> forall y : V, F --> y <-> (forall eps : R, 0 < eps -> \near
              F, `|[y - F]| < eps).


Lemma locally_normE : forall (K : absRingType) (V : normedModType K) (x : V) (P :
    classical_sets.set V), locally_ (ball_ norm) x P = (\near x, P x).
```

Hahn-Banach, Finally

The theorem is formalized, but questionable until it is not used somewhere:

https://github.com/math-comp/analysis/blob/hb/hahn_banach_applications.v

```
Variable ( V : normedModType R)

Lemma continuousR_bounded0 (f : {scalar V}) :
  (continuousR_at 0 f) -> ( exists r , (r > 0 ) /\ (forall x : V, ( `|f x| ) <=
        (`|[x]| ) * r ) ) .

Theorem HB_geom_normed ( F : pred V ) (H : submod_closed F) (f : {scalar V}) :
 continuousR_on F f ->
  exists g : {scalar V} , ( continuous g ) /\ ( forall x, F x -> (g x = f x)).
```

The Maths should be in Prop

Coq involves a sort Prop, allowing for propositional extensionality and transparent to extraction.

```
Variable Choice : forall T U (P : T -> U -> Prop),
    (forall t : T, exists u : U, P t u) -> { e, forall t, P t (e t) }.

Theorem HahnBanach : exists g : {scalar V},
  (forall x, g x <= p x) /\ (forall x, F x -> g x = f x).
```

However:

- Proving a result in Prop should be done using only axioms in Prop.
- The proof of Zorn in boolp.v used extensively the Choice in Type.

```
Definition xget {T : choiceType} x0 (P : set T) : T :=
  if pselect (exists x : T, `[<P x>]) isn't left exP then x0
  else projT1 (sigW exP).
```

Fixpoint theorem and Zorn in Prop

Following Lang's Algebra book :

```
Lemma fixpoint_T ( R : {strict_inductive_order T}) (f : T -> T) :
  (forall t, R t (f t)) -> exists t, t = f t.

Lemma Zorn T (R : {order T}) :
  (forall A : set T, total_on A R -> exists t, forall s, A s -> R s t) ->
  exists t, forall s, R t s -> s = t.
```

By (`Diaconescu : Choice -> EM`) Hahn-Banach Theorem depends of the following axioms ;

```
Axiom prop_irrelevance : forall (P : Prop) (x y : P), x = y.
Axiom funext : forall (T U : Type) (f g : T -> U), (f =1 g) -> f = g.
Axiom propext : forall (P Q : Prop), (P <-> Q) -> (P = Q).

Axiom choice_prop := ((forall T U (Q : T -> U -> Prop),
 (forall t : T, exists u : U, Q t u) -> (exists e, forall t, Q t (e t))) .
```

Difficulties

- Mathcomp has its focus on the interaction between `bool` and `Prop`, while Mathematical Components Analysis does everything with Prop.

  [classical_sets.v], properties on reals soon to be corrected

```
Check ub. (*forall R : archiFieldType, pred R -> pred R*)

Notation set R := R -> Prop

Definition ubd (A : set R) (a : R) := forall x, A x -> x <= a.
```

- Mathcomp Libraries have a discrete flavour. This is misleading for a new library user.

  [vector.v = finite dimensional vector spaces]

Conclusion

Better documentation is needed.

Meanwhile:

- Slides by Cyril Cohen :

  https://perso.crans.org/cohen/CoqWS2018.pdf
- Lessons and exercices on Coq, Ssreflect and Mathcomp libraries:

  https://team.inria.fr/marelle/en/coq-winter-school-2018-2019-ssreflect-mathcomp/
- A Book by Assia Mahboubi and Enrico Tassi :

  https://math-comp.github.io/mcb/
- **Gitter** forums : tell us

  https://gitter.im/math-comp/analysis

Installation: Via git or opam, or soon via Nix.