A Coq Formalization of Lebesgue Integration of Nonnegative Functions

Sylvie Boldo

Inria, Université Paris-Saclay

January 7th, 2020
Disclaimers

Disclaimer 1: this is joint work with

- François Clément,
- Florian Faissole,
- Vincent Martin,
- Micaela Mayero.
Disclaimers

Disclaimer 1: this is joint work with
- François Clément,
- Florian Faissole,
- Vincent Martin,
- Micaela Mayero.

Disclaimer 2:
This formalization of mathematics is done in Coq.
Disclaimers

**Disclaimer 1:** this is joint work with

- François Clément,
- Florian Faissole,
- Vincent Martin,
- Micaela Mayero.

**Disclaimer 2:**
This formalization of mathematics is done in Coq.

**Disclaimer 3:**
There is (nearly) no computer arithmetic!
1 Introduction

2 Towards the Finite Element Method

3 Lebesgue Integration
   - Measurability
   - Measure
   - Simple Functions and their Integral
   - Lebesgue Integral of Nonnegative Functions

4 Conclusion and Perspectives
Introduction

Mathematics

\( \mathbb{R}, \int, \frac{\partial^2 u}{\partial t^2} \)

theorems
Mathematics

\[ \mathbb{R}, \int, \frac{\partial^2 u}{\partial t^2} \]

theorems

Applied Mathematics

numerical scheme, convergence algorithms + theorems
Mathematics

\[ \mathbb{R}, \int, \frac{\partial^2 u}{\partial t^2} \]
theorems

Applied Mathematics

numerical scheme, convergence
algorithms + theorems

Computer

floating-point numbers, implementation
programs + ?
Mathematics, applied mathematics, numerical schemes, convergence, algorithms, theorems.
Motivations

PDE (Partial Differential Equation) ⇒ weather forecast
⇒ nuclear simulation
⇒ optimal control
⇒ . . .
Motivations

**PDE (Partial Differential Equation)**  $\Rightarrow$  weather forecast  
$\Rightarrow$  nuclear simulation  
$\Rightarrow$  optimal control  
$\Rightarrow$  …

Usually too complex to solve by an exact mathematical formula  
$\Rightarrow$  approximated by **numerical scheme over discrete grids/volumes**

$\Rightarrow$  mathematical proofs of the convergence of the numerical scheme  
(we compute something close to the PDE solution if the size decreases)
Motivations

PDE (Partial Differential Equation) $\Rightarrow$ weather forecast
$\Rightarrow$ nuclear simulation
$\Rightarrow$ optimal control
$\Rightarrow$ ...

Usually too complex to solve by an exact mathematical formula
$\Rightarrow$ approximated by numerical scheme over discrete grids/volumes

$\Rightarrow$ mathematical proofs of the convergence of the numerical scheme
(we compute something close to the PDE solution if the size decreases)

$\Rightarrow$ real program implementing the scheme/method
Motivations

PDE (Partial Differential Equation) ⇒ weather forecast
⇒ nuclear simulation
⇒ optimal control
⇒ . . .

Usually too complex to solve by an exact mathematical formula
⇒ approximated by numerical scheme over discrete grids/volumes
⇒ mathematical proofs of the convergence of the numerical scheme
   (we compute something close to the PDE solution if the size decreases)
⇒ real program implementing the scheme/method

Let us machine-check this kind of programs!
Outline

1 Introduction

2 Towards the Finite Element Method

3 Lebesgue Integration
   - Measurability
   - Measure
   - Simple Functions and their Integral
   - Lebesgue Integral of Nonnegative Functions

4 Conclusion and Perspectives
The sinking of the Sleipner A offshore platform

Excerpted from a report of SINTEF, Civil and Environmental Engineering:

The Sleipner A platform produces oil and gas in the North Sea and is supported on the seabed at a water depth of 82 m. It is a Condeep type platform with a concrete gravity base structure consisting of 24 cells and with a total base area of 16 000 m². Four cells are elongated to shafts supporting the platform deck. The first concrete base structure for Sleipner A sprang a leak and sank under a controlled ballasting operation during preparation for deck mating in Gandsfjorden outside Stavanger, Norway on 23 August 1991.

Immediately after the accident, the owner of the platform, Statoil, a Norwegian oil company appointed an investigation group, and SINTEF was contracted to be the technical advisor for this group.

The investigation into the accident is described in 16 reports...

The conclusion of the investigation was that the loss was caused by a failure in a cell wall, resulting in a serious crack and a leakage that the pumps were not able to cope with. The wall failed as a result of a combination of a serious error in the finite element analysis and insufficient anchorage of the reinforcement in a critical zone.

A better idea of what was involved can be obtained from this photo and sketch of the platform. The top deck weighs 57,000 tons, and provides accommodation for about 200 people and support for drilling equipment weighing about 40,000 tons. When the first model sank in August 1991, the crash
Motivations

Real life applications need solving PDE (Partial Differential Equation) on complex 3D geometries.
Motivations

Real life applications need solving PDE (Partial Differential Equation) on complex 3D geometries.
Motivations

Real life applications need solving **PDE** (Partial Differential Equation) on complex 3D geometries.

Instead of regular 2D/3D grids, we consider meshes made of triangles/tetrahedra.
The Finite Element Method (FEM) is the most used method to solve PDEs over meshes.

\[
FEM \text{ encompasses methods for connecting many simple element equations over many small subdomains, named finite elements, to approximate a more complex equation over a larger domain.}
\]

Motivations

The Finite Element Method (FEM) is the most used method to solve PDEs over meshes.

\[ FEM \text{ encompasses methods for connecting many simple element equations over many small subdomains, named finite elements, to approximate a more complex equation over a larger domain.} \]


⇒ mathematical proofs of the FEM
⇒ C++ library (Felisce) implementing the FEM
Motivations

The Finite Element Method (FEM) is the most used method to solve PDEs over meshes.

\[ FEM \text{ encompasses methods for connecting many simple element equations over many small subdomains, named finite elements, to approximate a more complex equation over a larger domain.} \]


⇒ mathematical proofs of the FEM
⇒ C++ library (Felisce) implementing the FEM

Let us machine-check this program!
Motivations

The Finite Element Method (FEM) is the most used method to solve PDEs over meshes.

*FEM encompasses methods for connecting many simple element equations over many small subdomains, named finite elements, to approximate a more complex equation over a larger domain.*


⇒ mathematical proofs of the FEM
⇒ C++ library (Felisce) implementing the FEM

Let us machine-check this program!

First, let us understand/formally prove the mathematics.
Mathematicians at work for Lax-Milgram theorem

- more 50 pages of mathematical proofs
Mathematicians at work for Lax-Milgram theorem

- more 50 pages of mathematical proofs
- very detailed!
Mathematicians at work for Lax-Milgram theorem

- more 50 pages of mathematical proofs
- very detailed!
- more than 7,000 lines and 220,000 characters
Mathematicians at work for Lax-Milgram theorem

- more 50 pages of mathematical proofs
- very detailed!
- more than 7,000 lines and 220,000 characters
- with dependencies!
Mathematicians at work for Lax-Milgram theorem

- more 50 pages of mathematical proofs
- very detailed!
- more than 7,000 lines and 220,000 characters
- with dependencies!
Mathematicians at work for Lax-Milgram theorem

- more 50 pages of mathematical proofs
- very detailed!
- more than 7,000 lines and 220,000 characters
- with dependencies!
Proof engineering

Let us build upon the Coquelicot library (Boldo, Lelay, Melquiond)

+ general spaces
Proof engineering

Let us build upon the Coquelicot library (Boldo, Lelay, Melquiond)

+ general spaces
+ many existing theorems
Proof engineering

Let us build upon the Coquelicot library (Boldo, Lelay, Melquiond)

+ general spaces
+ many existing theorems
  - not always the space we need
Enriched Hierarchy

AbelianGroup (0, +, −)

sum \_n\_m

Ring (1, ×)
pow \_n
\(M_n(\mathbb{C})\)

ModuleSpace (·)
\(M_{n,m}(\mathbb{C})\)

AbsRing (| |)
\(\mathbb{R}, \mathbb{C}\)

NormedModule (|| ||)
\(\sum, \int, f'\)

→: “used to define”
→→: “parameter of”
→→: “is proved to be a”
Enriched Hierarchy

←→: “used to define”
<→: “parameter of”
<→: “is proved to be a”

AbelianGroup \((0, +, -)\)

\[
\text{sum}_{n,m}
\]

\[
\text{Ring } (1, \times)
\]

\[
pow_n
\]

\[
M_n(\mathbb{C})
\]

→

ModuleSpace (\(\cdot\))

\[
M_{n,m}(\mathbb{C})
\]

→

AbsRing (\(|\ |\))

\[
\mathbb{R}, \mathbb{C}
\]

→

NormedModule (\(||\ |||\))

\[
\sum, \int, f'
\]

→

UniformSpace (\(\text{ball}\))

locally
Enriched Hierarchy

$\text{AbelianGroup (0, +, -)}$

$\text{Ring (1, \times)}$

$\text{ModuleSpace (\cdot)}$

$\text{AbsRing (\mid \mid)}$

$\text{NormedModule (\mid \mid \mid)}$

$\text{UniformSpace (ball) locally}$

$\text{CompleteSpace (lim) Cauchy}$

$\sum, \int, f'$

---$\rightarrow$: “used to define”

---$\rightarrow$: “parameter of”

---$\rightarrow$: “is proved to be a”
Enriched Hierarchy

AbelianGroup (0, +, −)

Ring (1, ×)

AbsRing (| |)

CompleteNormedModule

CompleteSpace (lim)

UniformSpace (ball) locally

Summation

Integration

Derivative

NormedModule (|| ||)

ModuleSpace (·)

Mn,m(C)

Mn(C)

Powers

Subtraction

Addition

Sum

—→: “used to define”

-->: “parameter of”

—→: “is proved to be a”
Enriched Hierarchy

AbelianGroup \((0, +, -)\)

\[ \text{sum}_n \text{m} \]

Ring \((1, \times)\)

\[ \text{pow}_n \text{M}_n(\mathbb{C}) \]

PreHilbert (inner)

norm

AbsRing \((| |)\)

\[ \mathbb{R}, \mathbb{C} \]

NormedModule \((|| ||)\)

\[ \sum, \int, f' \]

CompleteNormedModule

\[ \int \lim = \lim \int \]

\[ \mathbb{R}, \mathbb{R}^2, \mathbb{C} \]

CompleteSpace (lim)

\[ \mathbb{R} \rightarrow \mathbb{C} \]

UniformSpace (ball)

locally

\[ \rightarrow: \text{“used to define”} \]

\[ \rightarrow: \text{“parameter of”} \]

\[ \rightarrow: \text{“is proved to be a”} \]
Enriched Hierarchy

- **AbelianGroup** $(0, +, -)$
  - $\text{sum}_n\ m$

- **Ring** $(1, \times)$
  - $\text{pow}_n\ M_n(\mathbb{C})$

- **ModuleSpace** $(\cdot)$
  - $M_{n,m}(\mathbb{C})$

- **PreHilbert** (inner)
  - $\text{norm}$

- **AbsRing** $(| |)$
  - $\mathbb{R}, \mathbb{C}$

- **NormedModule** $(\| \|)$
  - $\sum, \int, f'$

- **CompleteNormedModule**
  - $\int \lim = \lim \int$
  - $\mathbb{R}, \mathbb{R}^2, \mathbb{C}$

- **UniformSpace** (ball) locally

- **CompleteSpace** (lim) Cauchy
  - $\mathbb{R} \rightarrow \mathbb{C}$

- **Hilbert** (lim) Cauchy

The arrows represent:
- $\rightarrow$: “used to define”
- $\dashrightarrow$: “parameter of”
- $\rightarrow$: “is proved to be a”
Summary of the work done

- results about **functional spaces**, linear and bilinear mappings
Summary of the work done

- results about **functional spaces**, linear and bilinear mappings
- **fixed-point theorem** in a sub-complete normed module
Summary of the work done

- results about **functional spaces**, linear and bilinear mappings
- **fixed-point theorem** in a sub-complete normed module
- decide if a space \((\geq \text{NormedModule})\) is only zero
Summary of the work done

- results about **functional spaces**, linear and bilinear mappings
- **fixed-point theorem** in a sub-complete normed module
- decide if a space ($\geq$ NormedModule) is only zero
- norm on functions (operator norm) on $\mathbb{R} \cup \{+\infty\}$
Summary of the work done

- results about **functional spaces**, linear and bilinear mappings
- **fixed-point theorem** in a sub-complete normed module
- decide if a space ($\geq$ NormedModule) is only zero
- norm on functions (operator norm) on $\mathbb{R} \cup \{+\infty\}$
- 8 equivalences of continuity of linear mappings
Summary of the work done

- results about **functional spaces**, linear and bilinear mappings
- **fixed-point theorem** in a sub-complete normed module
- decide if a space ($\geq$ NormedModule) is only zero
- norm on functions (operator norm) on $\mathbb{R} \cup \{+\infty\}$
- 8 equivalences of continuity of linear mappings
- definitions of **pre-Hilbert and Hilbert spaces** in Coquelicot hierarchy
Summary of the work done

- results about functional spaces, linear and bilinear mappings
- fixed-point theorem in a sub-complete normed module
- decide if a space ($\geq$ NormedModule) is only zero
- norm on functions (operator norm) on $\mathbb{R} \cup \{+\infty\}$
- 8 equivalences of continuity of linear mappings
- definitions of pre-Hilbert and Hilbert spaces in Coquelicot hierarchy
- define $\text{clm}$: the set of the continuous linear mappings
Summary of the work done

- results about **functional spaces**, linear and bilinear mappings
- **fixed-point theorem** in a sub-complete normed module
- decide if a space \( \geq \text{NormedModule} \) is only zero
- norm on functions (operator norm) on \( \mathbb{R} \cup \{+\infty\} \)
- 8 equivalences of continuity of linear mappings
- definitions of **pre-Hilbert and Hilbert spaces** in Coquelicicot hierarchy
- define \( \text{clm} \): the set of the **continuous linear mappings**
- prove it is a \text{NormedModule}, to consider \( \text{clm} \ E (\text{clm} \ E \mathbb{R}) \)
Summary of the work done

- results about functional spaces, linear and bilinear mappings
- fixed-point theorem in a sub-complete normed module
- decide if a space ($\geq$ NormedModule) is only zero
- norm on functions (operator norm) on $\mathbb{R} \cup \{+\infty\}$
- 8 equivalences of continuity of linear mappings
- definitions of pre-Hilbert and Hilbert spaces in Coquelicot hierarchy
- define $\text{clm}$: the set of the continuous linear mappings
- prove it is a NormedModule, to consider $\text{clm} \ E$ ($\text{clm} \ E$ $\mathbb{R}$)
- prove Lax-Milgram theorem and Céa’s lemma
Summary of the work done

- results about functional spaces, linear and bilinear mappings
- fixed-point theorem in a sub-complete normed module
- decide if a space \((\geq \text{NormedModule})\) is only zero
- norm on functions (operator norm) on \(\mathbb{R} \cup \{+\infty\}\)
- 8 equivalences of continuity of linear mappings
- definitions of pre-Hilbert and Hilbert spaces in Coquelicicot hierarchy
- define \(\text{clm}\): the set of the continuous linear mappings
- prove it is a NormedModule, to consider \(\text{clm} \ E (\text{clm} \ E \ \mathbb{R})\)
- prove Lax-Milgram theorem and Céa’s lemma
- for a total of about 10k lines of Coq and 430 lemmas/definitions
Theorem (Lax-Milgram)

Let $E : \text{Hilbert}$, $f \in E'$, $C, \alpha \in \mathbb{R}_+^*$. Let $\varphi : E \to \text{Prop}$, $\varphi$ $\text{ModuleSpace}$-compatible and complete. Let $a$ be a bilinear form of $E$ bounded by $C$ and $\alpha$-coercive. Then:

$$\exists! u \in E, \varphi(u) \land \forall v \in E, \varphi(v) \implies f(v) = a(u, v) \land \|u\|_E \leq \frac{1}{\alpha} \|f\|_{\varphi}.$$ 

Lemma (Céa)

Let $E : \text{Hilbert}$, $f \in E'$, $0 < \alpha$. Let $\varphi : E \to \text{Prop}$, $\varphi$ $\text{ModuleSpace}$-compatible and complete. Let $a$ be a bilinear form of $E$, bounded by $C > 0$ and $\alpha$-coercive. Let $u$ and $u_{\varphi}$ be the solutions given by Lax–Milgram Theorem respectively on $E$ and on the subspace $\varphi$. Then:

$$\forall v_{\varphi} \in E, \varphi(v_{\varphi}) \implies \|u - u_{\varphi}\|_E \leq \frac{C}{\alpha} \|u - v_{\varphi}\|_E.$$
Where are we?

Towards the Coq formalization of the finite element method:

- Lax-Milgram theorem (√)

where \( E \) is the Sobolev space \( L^2 \) and we will prove it is an Hilbert space using Lebesgue integration!
Where are we?

Towards the Coq formalization of the finite element method:

- Lax-Milgram theorem (√)
- requires the subspace to be complete
  (√ for finite-dimensional subspaces)
Where are we?

Towards the Coq formalization of the finite element method:

- Lax-Milgram theorem (√)
- requires the subspace to be complete (√ for finite-dimensional subspaces)
- requires $E$ to be a Hilbert space

$E$ will be instantiated as the Sobolev space $L^2$. We need to prove that $L^2$ is an Hilbert space using Lebesgue integration!
Where are we?

Towards the Coq formalization of the finite element method:

- Lax-Milgram theorem (√)
- requires the subspace to be complete (√ for finite-dimensional subspaces)
- requires $E$ to be a Hilbert space
- $E$ will be instantiated as the Sobolev space $L_2$
Towards the Coq formalization of the finite element method:

- Lax-Milgram theorem (√)
- requires the subspace to be complete
  (√ for finite-dimensional subspaces)
- requires $E$ to be a Hilbert space
- $E$ will be instantiated as the Sobolev space $L_2$

⇒ prove that $L_2$ is an Hilbert space
Towards the Coq formalization of the finite element method:

- Lax-Milgram theorem (✓)
- requires the subspace to be complete (✓ for finite-dimensional subspaces)
- requires $E$ to be a Hilbert space
- $E$ will be instantiated as the Sobolev space $L_2$

$\Rightarrow$ prove that $L_2$ is an Hilbert space
$\Rightarrow$ Lebesgue integration!
Towards the Coq formalization of the finite element method:

- Lax-Milgram theorem (√)
- requires the subspace to be complete (√ for finite-dimensional subspaces)
- requires $E$ to be a Hilbert space
- $E$ will be instantiated as the Sobolev space $L_2$
  ⇒ prove that $L_2$ is an Hilbert space
  ⇒ Lebesgue integration!
- ...
Outline

1. Introduction

2. Towards the Finite Element Method

3. Lebesgue Integration
   - Measurability
   - Measure
   - Simple Functions and their Integral
   - Lebesgue Integral of Nonnegative Functions

4. Conclusion and Perspectives
Outline

1 Introduction

2 Towards the Finite Element Method

3 Lebesgue Integration
   - Measurability
   - Measure
   - Simple Functions and their Integral
   - Lebesgue Integral of Nonnegative Functions

4 Conclusion and Perspectives
Measurability

Given a set $E \rightarrow \text{Prop}$, is it measurable?
Given a set $E \rightarrow \text{Prop}$, is it measurable?

We chose the definition from the generator sets:

Context $\{ E : \text{Type} \}$.

(* initialization sets *)

Variable $\text{gen} : (E \rightarrow \text{Prop}) \rightarrow \text{Prop}$.

Inductive measurable : $(E \rightarrow \text{Prop}) \rightarrow \text{Prop} :=$

| measurable_gen : forall omega, gen omega \rightarrow measurable omega |
| measurable_empty : measurable (fun _ \Rightarrow False) |
| measurable_compl : forall omega, |
| measurable (fun x \Rightarrow not (omega x)) \rightarrow measurable omega |
| measurable_union_countable : forall omega:nat \rightarrow (E \rightarrow \text{Prop}), |
| (forall n, measurable (omega n)) \rightarrow |
| measurable (fun x \Rightarrow exists n, omega n x). |

The measurable sets are aka $\sigma$-algebras.
Measurability

Advantages of **Inductive**:
\[\Rightarrow\] induction is possible
\[\Rightarrow\] easy theorems
Measurability

Advantages of **Inductive**:

⇒ induction is possible

⇒ easy theorems

We defined generators on \( \mathbb{R} \) and \( \overline{\mathbb{R}} \):

**Definition** gen\_R\_cc := (fun om ⇒ exists a b, (forall x, om x ⇔ a ≤ x ≤ b)).

**Definition** gen\_Rbar\_mc := (fun om ⇒ exists a, (forall x, om x ⇔ Rbar\_le a x)).
Measurability

Advantages of **Inductive**: 
⇒ induction is possible 
⇒ easy theorems

We defined generators on $\mathbb{R}$ and $\mathbb{R}$:

**Definition** gen\_R\_cc := (fun om ⇒ exists a b, (forall x, om x ↔ a ≤ x ≤ b)).
**Definition** gen\_Rbar\_mc := (fun om ⇒ exists a, (forall x, om x ↔ Rbar_le a x)).

But we may use other generators and prove the measurable sets are the same. For instance $a < x < b$ or with $a$ and $b$ in $\mathbb{Q}$.
Measurability

Advantages of **Inductive**:
⇒ induction is possible
⇒ easy theorems

We defined generators on $\mathbb{R}$ and $\overline{\mathbb{R}}$:

**Definition** gen$_{\mathbb{R}}$$_{\_cc}$ := (fun om ⇒ exists a b, (forall x, om x ↔ a ≤ x ≤ b)).
**Definition** gen$_{\mathbb{R}bar}$$_{\_mc}$ := (fun om ⇒ exists a, (forall x, om x ↔ Rbar_le a x)).

But we may use other generators and prove the measurable sets are the same. For instance $a < x < b$ or with $a$ and $b$ in $\mathbb{Q}$.

And we proved that it is equivalent to the usual Borel $\sigma$-algebras:

**Lemma** measurable$_{\mathbb{R}}$$_{\_open}$ : forall om,
measurable gen$_{\mathbb{R}}$$_{\_cc}$ om ↔ measurable open om.
Measurable functions

A function $f : E \to F$ is measurable if the set $A(f(x))$ is measurable in $F$ for all measurable sets $A$ in $E$:

**Definition** measurable_fun : $(E \to F) \to Prop :=$

\[
\text{fun } f \Rightarrow (\forall \ A : F \to Prop), \text{measurable genF } A \Rightarrow \text{measurable genE } (\text{fun } x \Rightarrow A (f x)).
\]
A function $f : E \to F$ is measurable if the set $A(f(x))$ is measurable in $F$ for all measurable sets $A$ in $E$:

**Definition measurable_fun** : $(E \to F) \to \text{Prop} :=$

\[
\text{fun } f \Rightarrow (\forall (A : F \to \text{Prop}), \text{measurable genF } A \Rightarrow \text{measurable genE } (\text{fun } x \Rightarrow A(f(x)))).
\]

The sum and multiplication by a scalar of measurable functions on $\mathbb{R}$ and $\overline{\mathbb{R}}$ are measurable functions.
Measurable functions

A function \( f : E \rightarrow F \) is measurable if the set \( A(f(x)) \) is measurable in \( F \) for all measurable sets \( A \) in \( E \):

**Definition measurable_fun :** \((E \rightarrow F) \rightarrow \text{Prop} \) :=

\[
\text{fun } f \Rightarrow \left( \forall A : F \rightarrow \text{Prop}, \text{measurable genF } A \rightarrow \text{measurable genE } \left( \text{fun } x \Rightarrow A(f(x)) \right) \right).
\]

The sum and multiplication by a scalar of measurable functions on \( \mathbb{R} \) and \( \overline{\mathbb{R}} \) are measurable functions.

When considering the **restriction** of \( f \) on the subset \( A \).

**Lemma measurable_fun_when_charac :**

\[
\forall f f' : E \rightarrow \mathbb{R}, A : E \rightarrow \text{Prop},
\text{measurable gen } A \rightarrow
\left( \forall x, A x \rightarrow f x = f' x \right) \rightarrow
\text{measurable_fun_Rbar } f' \rightarrow
\text{measurable_fun_Rbar } \left( \text{fun } x \Rightarrow \text{Rbar_mult } (f x) (\text{charac } A x) \right).
\]

with \( \text{charac } A \) the characteristic function \( \mathbf{1}_A \).
Outline

1. Introduction

2. Towards the Finite Element Method

3. Lebesgue Integration
   - Measurability
   - Measure
     - Simple Functions and their Integral
     - Lebesgue Integral of Nonnegative Functions

4. Conclusion and Perspectives
Measure definition

We choose to not (yet) define the Lebesgue measure, but define what a measure is supposed to satisfy:

Context \{ E : Type \}.
Variable gen : (E → Prop) → Prop.

Record measure := mk_measure {
    meas :> (E→ Prop) → Rbar ;
    meas_False : meas (fun _ ⇒ False) = 0 ;
    meas_ge_0: forall om, Rbar_le 0 (meas om) ;
    meas_sigma_additivity : forall omega :nat → (E→ Prop),
        (forall n, measurable gen (omega n)) →
        (forall n m x, omega n x → omega m x → n = m)
        → meas (fun x ⇒ exists n, omega n x)
        = Sup_seq (fun n ⇒ sum_Rbar n (fun m ⇒ meas (omega m)))
}.

Note that we have at least a measure: the Dirac measure.
Many properties hold for all measures such as:

**Lemma measure_Boole_ineq** :\(\forall (\mu : \text{measure}) (A : \text{nat} \to E \to \text{Prop}) (N : \text{nat}), \)
\(\forall n, \ n \leq N \rightarrow \text{measurable gen } (A n) \rightarrow \)
\(\text{Rbar_le } (\mu (\text{fun } x \Rightarrow \exists n, \ n \leq N \land A n x)) \)
\(\text{(sum_Rbar } N (\text{fun } m \Rightarrow \mu (A m)))).\)

\[
\mu \left( \bigcup_{i \in \left[0..N\right]} A_i \right) \leq \sum_{i \in \left[0..N\right]} \mu(A_i)
\]
1. Introduction

2. Towards the Finite Element Method

3. Lebesgue Integration
   - Measurability
   - Measure
   - Simple Functions and their Integral
   - Lebesgue Integral of Nonnegative Functions

4. Conclusion and Perspectives
We have tried various definitions of simple functions, especially as we prefer to sum over a finite set of values.

Examples of simple functions @ mathonline

\[ f = \sum_{y \in f(E)} \mathbb{1}_{f^{-1}\{y\}} \]
Simple functions?

Examples of simple functions @ mathonline

\[ f \ = \ \sum_{y \in f(E)} 1_{f^{-1}(\{y\})} \]

We have tried *various definitions* of simple functions, especially as we prefer to sum over a finite set of values.
Definition finite_vals : (E → R) → (list R) → Prop :=
   fun f l ⇒ ∀ y, In (f y) l.

⇒ OK, but not unique.
Simple functions definition

**Definition** finite_vals : \((E \rightarrow \mathbb{R}) \rightarrow (\text{list R}) \rightarrow \text{Prop}\) :=
  \(\text{fun } f \ l \Rightarrow \forall y, \text{In}(f \ y) \ l.\)

⇒ OK, but not unique.

**Definition** finite_vals_canonic : \((E \rightarrow \mathbb{R}) \rightarrow (\text{list R}) \rightarrow \text{Prop}\) :=
  \(\text{fun } f \ l \Rightarrow (\text{LocallySorted Rlt} \ l) \land
    (\forall x, \text{In} x \ l \rightarrow \exists y, f \ y = x) \land
    (\forall y, \text{In} (f \ y) \ l).\)

⇒ unique!

We were able to construct the second list from the first.
Simple functions integration

\[
\int f \, d\mu \overset{\text{def.}}{=} \sum_{a \in f(X)} a \mu(f^{-1}(a)) \in \overline{\mathbb{R}}
\]
Simple functions integration

\[ \int f \, d\mu \overset{\text{def.}}{=} \sum_{a \in f(X)} a \mu(f^{-1}(a)) \in \overline{\mathbb{R}} \]

**Definition** \(SF_{\text{aux}} : (E \to \mathbb{R}) \to (\text{list}\ \mathbb{R}) \to \text{Prop} :=\)

\[ \text{fun } f \ l \Rightarrow \text{finite_vals_canonic } f \ l \land \]
\[ (\forall a, \text{measurable } \text{gen } (\text{fun } x \Rightarrow f \ x = a)).\]

**Definition** \(SF : (E \to \mathbb{R}) \to \text{Set} := \text{fun } f \Rightarrow \{ \ l \mid SF_{\text{aux}} f \ l\}.\)

**Definition** \(af1 (f:E \to \mathbb{R}) :=\)

\[ (\text{fun } a : \mathbb{R}_\text{bar} \Rightarrow \text{Rbar_mult } a \ (\mu (\text{fun } (x:E) \Rightarrow f \ x = a))).\]

**Definition** \(L\text{Int}_\text{simple}_\text{fun}_p :=\)

\[ \text{fun } (f:E \to \mathbb{R}) (H:SF \ \text{gen } f) \Rightarrow \text{let } l := (\text{proj1_sig } H) \text{ in } \]
\[ \text{sum_Rbar_map } l (af1 f).\]

We proved the value does not depends on the proof \(H\).
Simple functions integration

\[ \int f \, d\mu \overset{\text{def.}}{=} \sum_{a \in f(X)} a \mu(f^{-1}(a)) \in \mathbb{R} \]

**Definition** \( \text{SF\_aux} : (E \rightarrow R) \rightarrow (\text{list } R) \rightarrow \text{Prop} := \)

\[
\text{fun } f \ l \Rightarrow \text{finite\_vals\_canonic } f \ l \land
(f\text{\_forall } a, \text{\_measurable\_gen } (\text{fun } x \Rightarrow f \ x = a)).
\]

**Definition** \( \text{SF} : (E \rightarrow R) \rightarrow \text{Set} := \text{fun } f \Rightarrow \{ l \mid \text{SF\_aux } f \ l \}. \)

**Definition** \( \text{af1 } (f:E \rightarrow R) := \)

\[
(\text{fun } a : \text{Rbar } \Rightarrow \text{Rbar\_mult } a (\mu (\text{fun } (x:E) \Rightarrow f \ x = a))).
\]

**Definition** \( \text{LInt\_simple\_fun\_p} := \)

\[
\text{fun } (f:E \rightarrow R)(H:\text{SF gen } f) \Rightarrow \text{let } l := (\text{proj1\_sig } H) \in
\text{sum\_Rbar\_map } l (\text{af1 } f).
\]

We proved the value does not depends on the proof \( H. \)

\( \Rightarrow \) **theorems** about sum, multiplication by a scalar and change of variable
Outline

1. Introduction

2. Towards the Finite Element Method

3. Lebesgue Integration
   - Measurability
   - Measure
   - Simple Functions and their Integral
   - Lebesgue Integral of Nonnegative Functions

4. Conclusion and Perspectives
Lebesgue integration

\[ \int f \, d\mu \overset{\text{def.}}{=} \sup_{\varphi \in \mathcal{F}_+, \varphi \leq f} \int \varphi \, d\mu \in \overline{\mathbb{R}} \]
Lebesgue integral

\[ \int f \, d\mu \overset{\text{def.}}{=} \sup_{\varphi \in \mathcal{S} \mathcal{F}_+, \varphi \leq f} \int \varphi \, d\mu \in \overline{\mathbb{R}} \]
Lebesgue integral

\[ \int f \, d\mu \overset{\text{def.}}{=} \sup_{\varphi \in \mathcal{F}_+, \varphi \leq f} \int \varphi \, d\mu \in \overline{\mathbb{R}} \]

Riemann integral vs Lebesgue integral
Lebesgue integral definition

\[ \int f \, d\mu \overset{\text{def.}}{=} \sup_{\varphi \in SF_+, \varphi \leq f} \int \varphi \, d\mu \in \overline{\mathbb{R}} \]

**Definition** \( L\text{Int}_p : (E \to \mathbb{R}_{\text{bar}}) \to \mathbb{R}_{\text{bar}} := \text{fun } f \Rightarrow \)

\( R\text{bar}_{\text{lub}} (\text{fun } x \Rightarrow \text{exists } (g : E \to \mathbb{R}), \text{exists } (Hg : SF \text{ gen } g),
\text{non_neg } g \land
(\text{forall } (z : E), \text{Rbar_{le} } (g z) (f z)) \land
L\text{Int}_{\text{simple functor}} p \mu g Hg = x). \)
Theorem (Beppo Levi, monotone convergence)

Let \((f_n)_{n \in \mathbb{N}}\) be a sequence of nonnegative measurable functions, that is pointwise nondecreasing. Then, the pointwise limit of \((f_n)_{n \in \mathbb{N}}\) is nonnegative and measurable, and we have in \(\mathbb{R}\)

\[
\int \lim_{n \to \infty} f_n \, d\mu = \lim_{n \to \infty} \int f_n \, d\mu.
\]
Let \((f_n)_{n \in \mathbb{N}}\) be a sequence of nonnegative measurable functions. Then, we have in \(\mathbb{R}\):

\[
\int \liminf_{n \to \infty} f_n \, d\mu \leq \liminf_{n \to \infty} \int f_n \, d\mu.
\]

**Theorem Fatou-Lebesgue**: For all \(f : \text{nat} \to \text{E} \to \mathbb{R}\bar{\text{r}}\),

\[
(\forall n, \text{non_neg}(f_n)) \implies \\
(\forall n, \text{measurable_fun_Rbar gen}(f_n)) \implies \\
\text{Rbar_le}(\text{LInt_p mu (fun x \Rightarrow LimInf_seq'(fun n \Rightarrow f n x)))} \\
(\text{LimInf_seq'}(\text{fun n \Rightarrow LInt_p mu (f n)))).
\]
Theorems (3/3): focus on a hard one

\[ \int (f + g) = \int f + \int g \]

Lemma LInt_p_plus:
\[ \forall f \ g, \text{non} \neg \text{g} \rightarrow \text{measurable} \_ \text{fun} \_ \text{Rbar} \_ \text{gen} f \rightarrow \text{measurable} \_ \text{fun} \_ \text{Rbar} \_ \text{gen} g \rightarrow \\
\LInt_p \mu \left( \text{fun} x \Rightarrow \text{Rbar} \_ \text{plus} \( f \_ x \) \_ (g \_ x) \right) = \text{Rbar} \_ \text{plus} \left( \LInt_p \mu f \right) \_ (\LInt_p \mu g) \).
∫ (f + g) = ∫ f + ∫ g

**Lemma LInt_p_plus :** \(\forall f, g,\)

\(\text{non\_neg } f \rightarrow \text{non\_neg } g \rightarrow\)

\(\text{measurable\_fun\_Rbar gen } f \rightarrow \text{measurable\_fun\_Rbar gen } g \rightarrow\)

\(\text{LInt\_p } \mu (\text{fun } x \Rightarrow \text{Rbar\_plus } (f x) (g x))\)

\(= \text{Rbar\_plus } (\text{LInt\_p } \mu (f) (\text{LInt\_p } \mu (g)).\)
Proof of $\int (f + g) = \int f + \int g$ (1/2)

It needs adapted sequences:

**Definition** \(\text{is\_adapted\_seq}\ (f:E \to \mathbb{R})\ (\phi:nat \to E \to \mathbb{R}) :=\)
\[
\begin{align*}
\forall n, \nonneg (\phi n) \land \\
\forall (x:E)\ n, \phi n x \leq \phi (S n) x \land \\
\forall n, \exists l, \text{SF\_aux\ gen}\ (\phi n) l \land \\
\forall (x:E), \text{is\_sup\_seq}\ (\fun n \Rightarrow \phi n x) (f x)).
\end{align*}
\]

as their limit gives the integral:

**Lemma** \(\text{LInt\_p\_with\_adapted\_seq} :\)
\[
\begin{align*}
\forall f \phi, \text{is\_adapted\_seq}\ f\ \phi \Rightarrow \\
\text{is\_sup\_seq}\ (\fun n \Rightarrow \text{LInt\_p\ mu}\ (\phi n)) (\text{LInt\_p\ mu}\ f).
\end{align*}
\]
Proof of $\int (f + g) = \int f + \int g \ (2/2)$

Adapted sequences may be defined like that:

$$\forall x, \quad f_n(x) \overset{\text{def.}}{=} \begin{cases} \frac{\lfloor 2^n f(x) \rfloor}{2^n} & \text{when } f(x) < n, \\ n & \text{otherwise.} \end{cases}$$
Proof of $\int (f + g) = \int f + \int g$ (2/2)

Adapted sequences may be defined like that:

$$\forall x, \quad f_n(x) \overset{\text{def.}}{=} \begin{cases} \left\lfloor \frac{2^n f(x)}{2^n} \right\rfloor & \text{when } f(x) < n, \\ n & \text{otherwise.} \end{cases}$$

that may be written in Coq as:

Definition mk_adapted_seq (n:nat) (x:E) :=
    match Rbar_le_lt_dec (INR n) (f x) with
    | left _  ⇒ INR n
    | right _ ⇒ round radix2 (FIX_exp (-n)) Zfloor (f x)
    end.

relying on fixed-point arithmetic defined by the Flocq library!!

And then:

Lemma mk_adapted_seq_is_adapted_seq :
    is_adapted_seq f mk_adapted_seq.
1. Introduction

2. Towards the Finite Element Method

3. Lebesgue Integration
   - Measurability
   - Measure
   - Simple Functions and their Integral
   - Lebesgue Integral of Nonnegative Functions

4. Conclusion and Perspectives
Conclusion

For the Lebesgue integration:

- mathematicians at work: 184 pages and 600 lemmas/definitions
- formal proofs at work: 11 k lines lines and 635 lemmas/definitions
Conclusion

For the Lebesgue integration:

- mathematicians at work: 184 pages and 600 lemmas/definitions
- formal proofs at work: 11 k lines lines and 635 lemmas/definitions

Difficult parts:

- handling subspaces (mainly with $\mathbb{1}$ here)
- having usable simple functions
extend to functions of \textit{varying} sign

\[
\int f = \int \max(0, f) - \int \max(0, -f)
\]
extend to functions of \textit{varying} sign

\[
\int f = \int \max(0, f) - \int \max(0, -f)
\]

define the \textbf{Lebesgue measure}
 Perspectives

- extend to functions of varying sign
  \[ \int f = \int \max(0, f) - \int \max(0, -f) \]
- define the Lebesgue measure
- merge/take advantage of mathcomp-analysis

Sylvie Boldo (Inria)  Lebesgue Integration  January 7th, 2020  38 / 39
extend to functions of \textit{varying} sign
\[
\int f = \int \max(0, f) - \int \max(0, -f)
\]
define the \textbf{Lebesgue measure}
merge/take advantage of \texttt{mathcomp-analysis}
write the corresponding \texttt{article}
Perspectives

- extend to functions of varying sign

\[ \int f = \int \max(0, f) - \int \max(0, -f) \]

- define the Lebesgue measure
- merge/take advantage of mathcomp-analysis
- write the corresponding article
- define \( L_2 \) and prove it is a Hilbert
Perspectives

- extend to functions of varying sign
  \[ \int f = \int \max(0, f) - \int \max(0, -f) \]
- define the Lebesgue measure
- merge/take advantage of mathcomp-analysis
- write the corresponding article
- define \( L_2 \) and prove it is a Hilbert
- define the FEM algorithm and prove it
extend to functions of *varying* sign

\[ \int f = \int \max(0, f) - \int \max(0, -f) \]

- define the *Lebesgue measure*
- merge/take advantage of *mathcomp-analysis*
- write the corresponding *article*
- define \( L_2 \) and prove it is a Hilbert
- define the *FEM algorithm* and prove it
- prove a real *implementation* (in floating-point arithmetic)
Thank you for your attention