A Coq Formalization of Lebesgue Induction Principle and Tonelli's Theorem

Sylvie Boldo, François Clément, Vincent Martin, Micaela Mayero, Houda Mouhcine

A Coq Formalization of Lebesgue Induction Principle and Tonelli's Theorem*

Sylvie Boldo[1][0000−0002−1970−3019], François Clément[2], Vincent Martin[3], Micaela Mayero[4], and Houda Mouhcine[1,2,4]

[1] Université Paris-Saclay, CNRS, ENS Paris-Saclay, Inria, Laboratoire Méthodes Formelles, 91190, Gif-sur-Yvette, France.
[2] a. Inria, 2 rue Simone Iff, 75589 Paris, France.
b. CERMICS, École des Ponts, 77455 Marne-la-Vallée, France.
[3] Université de technologie de Compiègne, LMAC, 60203 Compiègne, France.
[4] LIPN, Université Paris 13 - USPN, CNRS UMR 7030, Villetaneuse, F-93430, France.

Abstract. Lebesgue integration is a well-known mathematical tool, used for instance in probability theory, real analysis, and numerical mathematics. Thus, its formalization in a proof assistant is to be designed to fit different goals and projects. Once the Lebesgue integral is formally defined and the first lemmas are proved, the question of the convenience of the formalization naturally arises. To check it, a useful extension is Tonelli's theorem, stating that the (double) integral of a nonnegative measurable function of two variables can be computed by iterated integrals, and allowing to switch the order of integration. This article describes the formal definition and proof in Coq of product σ-algebras, product measures and their uniqueness, the construction of iterated integrals, up to Tonelli's theorem. We also advertise the *Lebesgue induction principle* provided by an inductive type for nonnegative measurable functions.

Keywords: Formal proof, Coq, Measure theory, Lebesgue integration, Tonelli's theorem

1 Introduction

This work deals with the Coq[5] formalization of the Lebesgue induction principle and Tonelli's theorem as a direct continuation of previous work [2]. Our long-term objective is to formally prove in Coq scientific computing programs and the correctness of parts of a C++ library, such as FreeFEM++[6] or XLiFE++,[7] that implements the Finite Element Method (FEM), a widely used method for numerically solving Partial Differential Equations (PDEs) arising in different domains like engineering and mathematical modeling. With this work, we carry on

* This work was partly supported by the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme – Grant Agreement n°810367.

[5] https://coq.inria.fr/
[6] https://freefem.org/
[7] https://uma.ensta-paris.fr/soft/XLiFE++/

with our goal of providing a Coq library usable by scientific computing people. It started with the first development of a real numbers library [18], and then with the first complete formalization and proof of a numerical program [3] (a C program for the approximated resolution of the wave equation). More recently, the Lax–Milgram theorem [1] (for the resolution of a class of PDEs), then Lebesgue integration of nonnegative measurable functions [2], and Bochner integration [4] (a generalization for functions taking their values in a Banach space).

The proof of Tonelli's theorem is the natural next step. And, as a side result, it also allows us to validate our previous developments and in particular our formalization choices for the definitions and results about the Lebesgue integral. For example, as we work in Coq, the question arises of whether to use classical or intuitionistic real analysis. Following [2], we decided to be completely classical.

The Lebesgue induction principle is a proof technique for properties about nonnegative measurable functions, and usually involves the integral. It reflects the three construction steps followed by Henri Lebesgue to build his integral [15]. The property is first established for indicator functions, then for nonnegative simple functions by checking that the property is compatible with positive linear operations, and finally for all nonnegative measurable functions by checking that it is compatible with the supremum. This is an important asset for the proof of Tonelli's theorem, and we provide it as a byproduct of an inductive type.

Tonelli's theorem provides a convenient way to ease the computation of multiple integrals by stating their equality with iterated integrals, each in a single dimension. Tonelli's theorem applies to nonnegative measurable functions. A similar result, Fubini's theorem, applies to integrable functions with an arbitrary sign, or even taking their values in a Banach space when using the Bochner integral. This article focuses on the case of nonnegative functions, and we only address the case of functions of two variables, as it is common in mathematics.

We aim to the construction of the full formal proof in Coq of Tonelli's theorem, stating that the (double) integral of a nonnegative measurable function of two variables can be computed by iterated integrals, and allowing to switch the order of integration. It can be expressed in a mathematical setting as follows.

Theorem 1: Tonelli

Let (X_1, Σ_1, μ_1) and (X_2, Σ_2, μ_2) be measure spaces. Assume that μ_1 and μ_2 are σ-finite. Let $f \in \mathcal{M}_+(X_1 \times X_2, \Sigma_1 \otimes \Sigma_2)$. Then, we have

$$\left(\forall x_1 \in X_1,\ f_{x_1} \in \mathcal{M}_+(X_2, \Sigma_2)\right) \quad \wedge \quad \int_{X_2} f_{x_1}\, d\mu_2 \in \mathcal{M}_+(X_1, \Sigma_1), \quad (1)$$

$$\left(\forall x_2 \in X_2,\ f^{x_2} \in \mathcal{M}_+(X_1, \Sigma_1)\right) \quad \wedge \quad \int_{X_1} f^{x_2}\, d\mu_1 \in \mathcal{M}_+(X_2, \Sigma_2), \quad (2)$$

$$\int_{X_1 \times X_2} f\, d(\mu_1 \otimes \mu_2) = \int_{X_1} \left(\int_{X_2} f_{x_1}\, d\mu_2 \right) d\mu_1 = \int_{X_2} \left(\int_{X_1} f^{x_2}\, d\mu_1 \right) d\mu_2.$$

The notations are specified in the remainder of this paper. Just note that many measures, including the Lebesgue measure, are σ-finite (defined in Section 4), $\mathcal{M}_+$ denotes the set of nonnegative measurable functions (see Section 2.2), and f_{x_1} and f^{x_2} are partial applications of f (see Section 5.1). Notice also that the properties (1) and (2) ensure the existence of all simple integrals, while the existence of the double integral is granted by the assumption on the function f.

The mathematical definitions and proofs are taken from textbooks [17,12,8]. The Coq code is available at (mainly in files `Tonelli.v`, `LInt_p.v` and `Mp.v`)

https://lipn.univ-paris13.fr/coq-num-analysis/tree/Tonelli.1.0/Lebesgue

where the tag `Tonelli.1.0` corresponds to the code of this article. An Opam package, coq-num-analysis, is also available.[8]

Tonelli's theorem is known enough and useful enough to have been formalized before our work in several proof assistants. It has been done in PVS in the PVS-NASA library[9] by Lester, probably as a follow-up of [16]. Some Fubini-like results are available in HOL Light [13]. More recently, Tonelli's theorem was formalized in Mizar by Endou [11]. The formalizations nearest to ours are in Isabelle/HOL and Lean. Hölzl and Heller defined binary and iterated product measure before Fubini's theorem [14]. It relies on Isabelle type classes and locales. A more recent work[10] extends it to the Bochner integral. In Lean, van Doorn defines the product of measures and properties of the product space towards Tonelli and Fubini's theorems in a way similar to ours, but for the Bochner integral [20]. He also provides a similar Lebesgue induction principle, but to our knowledge, our approach of getting it from an inductive type is new. A recent work in Coq has been developed for probability theory.[11] Many definitions are similar to ours, but in a simpler setting where measures are finite. Fubini's theorem also appeared in math-comp/analysis[12] after the submission of this article. This formalization relies on the math-comp/analysis hierarchy of classes. First, this hierarchy is not compatible with the canonical structures of Coquelicot we used to prove the Lax–Milgram theorem [1]. Second, the depth of this hierarchy involves many abstractions for the unfamiliar user to process.

For a comparison of the Lebesgue integral in various proof assistants, we refer the reader to [2,20], and we refer to [6] for a wider comparison of real analysis in proof assistants.

We think Coq is the most suitable tool for our goal: to prove properties on the FEM algorithm and program, including floating point errors. Coq indeed provides both libraries and results for the mathematical part [1,2] and the Flocq library for floating-point arithmetic [7]. We are not aware of another proof assistant able to address these two issues together.

This paper is organized as follows. Section 2 summarizes the prerequisites and the main concepts of measure and integration theories developed in previ-

[8] https://coq.inria.fr/opam/www/
[9] https://github.com/nasa/pvslib/blob/master/measure_integration/fubini_tonelli.pvs
[10] https://isabelle.in.tum.de/library/HOL/HOL-Analysis/Bochner_Integration.html
[11] https://github.com/jtassarotti/coq-proba/
[12] https://github.com/math-comp/analysis/blob/master/theories/lebesgue_integral.v

ous works. The formalization of the Lebesgue induction principle is detailed in Section 3. Section 4 describes the building of the product measure, and Section 5 is devoted to the building of the iterated integrals and the full proof of Tonelli's theorem. Finally, Section 6 concludes and provides directions for future work.

2 Prerequisites

Our formalizations and proofs are conducted in Coq. In this section, we present the necessary prerequisites and libraries for our developments.

2.1 The Coquelicot Library, $\overline{\mathbb{R}}$ and Logic

The Coquelicot[13] library [5] is a conservative extension of the standard Coq library of real numbers [10,18] supplying basic results in real analysis. It is a classical library, and a salient feature is that it provides total functions, e.g. for limit, derivative, and (Riemann) integral. This is consistent with classical logic, and it means a natural way to write mathematical formulas and theorem statements. The library also provides a formalization of the extended real numbers $\overline{\mathbb{R}} := \mathbb{R} \cup \{-\infty, +\infty\}$ equipped, among other operations, with `Rbar_lub` for the least-upper bound of subsets, and `Sup_seq` for the supremum of sequences.

As in the Coquelicot library, we use the full classical logic: total order on real numbers, propositional and functional extensionality axioms, excluded middle and choice axioms. We rely on the same axioms detailed in [2, Section 2].

2.2 Lebesgue Integration Theory

The theory of integration is commonly built upon measure theory (e.g. see [9]): first, the measurability of subsets is defined, and then a measure associates a nonnegative number in $\overline{\mathbb{R}}_+$ to each measurable subset; second, the measurability of functions is defined, and then the integral associates a nonnegative number in $\overline{\mathbb{R}}_+$ to each nonnegative measurable function. The following summarizes what we need from [2].

Measurable Subsets. A measurable space (X, Σ) consists of a set X, and a *σ-algebra* Σ collecting all measurable subsets. A σ-algebra is closed under most set operations, such as complement, (countable) union and intersection. It can be *generated* as the closure of a collection of subsets with respect to some of the set operations. In our Coq developments, the generators on `X : Type` are typically denoted `genX : (X → Prop) → Prop`, and a subset `A : X → Prop` belongs to the σ-algebra generated by `genX` when the inductive property `measurable genX A` holds.

When the set X has a topological structure, it is convenient to use its *Borel σ-algebra* that is generated by all the open subsets. The Borel σ-algebra of $\overline{\mathbb{R}}$ can also be generated by the right closed rays ($[a, \infty]$), denoted in Coq by `gen_Rbar`.

[13] https://gitlab.inria.fr/coquelicot/coquelicot/

Given two measurable spaces (X_1, Σ_1) and (X_2, Σ_2), the *product σ-algebra on $X_1 \times X_2$* is the one generated by the products of measurable subsets of X_1 and X_2. Some details are provided in Section 4 where it is a major ingredient.

Measure. A measure space (X, Σ, μ) contains an additional *measure* μ: a function $\Sigma \to \overline{\mathbb{R}}$ that is nonnegative, homogeneous ($\mu(\varnothing) = 0$), and σ-additive. In Coq, a measure is a record collecting the function μ and its three properties. In Section 4, we rely on the properties of *continuity from below* and *from above*. For all sequences $(A_n)_{n\in\mathbb{N}} \in \Sigma$, they respectively state for any measure μ that when the sequence is nondecreasing, $\mu\left(\bigcup_{n\in\mathbb{N}} A_n\right) = \lim_{n\to\infty} \mu(A_n) = \sup_{n\in\mathbb{N}} \mu(A_n)$, and when it is nonincreasing and one of the subsets is of finite measure, then we have $\mu\left(\bigcap_{n\in\mathbb{N}} A_n\right) = \inf_{n\in\mathbb{N}} \mu(A_n)$. Note also that the monotonicity of measures allows to replace the limit of a nondecreasing sequence by its supremum.

Measurable Functions. Given two measurable spaces (X, Σ) and $(Y, \mathcal{T})$, a function $f : X \to Y$ is said *measurable* when the preimage of every measurable subset is measurable:[14]

```
Definition measurable_fun (f : X → Y) : Prop :=
  ∀ B, measurable genY B → measurable genX (fun x ⇒ B (f x)).
```

When $Y := \overline{\mathbb{R}}$, and usually $\mathcal{T}$ is its Borel σ-algebra, we simply say that the function is *Σ-measurable*, and we use the predicate `measurable_fun_Rbar` corresponding to `genY` := `gen_Rbar`. We denote the *set of nonnegative measurable functions* $\mathcal{M}_+(X, \Sigma)$. The "(X, Σ)" annotation may be dropped when there is no possible confusion. Among other operations, $\mathcal{M}_+$ is closed under nonnegative scalar multiplication, addition, and supremum. In Coq, we use the predicate `Mplus genX` : (`X` → $\overline{\mathbb{R}}$) → `Prop` that gathers nonnegativity and measurability, and `Mplus_seq genX` : ($\mathbb{N}$ → `X` → $\overline{\mathbb{R}}$) → `Prop` for sequences of functions in $\mathcal{M}_+$.

Simple functions are functions whose image has finite cardinality. The *set of nonnegative measurable simple functions* is denoted $\mathcal{SF}_+(X, \Sigma)$. In Coq, simple functions are *canonically* represented by their strictly sorted list of values, and we use the predicate `SFplus genX` : (`X` → $\overline{\mathbb{R}}$) → `Prop`. Given $f \in \mathcal{M}_+$, `mk_adapted_seq` provides an *adapted sequence for f*, i.e. a nondecreasing sequence $(\varphi_n)_{n\in\mathbb{N}}$ in $\mathcal{SF}_+$ such that $f = \lim_{n\to\infty} \varphi_n = \sup_{n\in\mathbb{N}} \varphi_n$.

The *set of measurable indicator functions* is denoted $\mathcal{IF}(X, \Sigma)$. Note that an indicator function $\mathbb{1}_A$ is measurable whenever its support subset A belongs to Σ. Simple functions in $\mathcal{SF}_+$ are nonnegative linear combinations of indicator functions.

Lebesgue Integral. The construction of the Lebesgue integral in $\mathcal{M}_+$ operates in three steps. The first stage is to integrate indicator functions in $\mathcal{IF}$ by taking the measure of their support. Then, the second stage extends the integral to simple functions in $\mathcal{SF}_+$ by positive linearity. And finally, the third stage extends it again to measurable functions in $\mathcal{M}_+$ by taking the supremum. In the end,

[14] Note that we often rely on the Section mechanism of Coq for "hiding" some arguments, here `genX` and `genY` (see https://coq.inria.fr/refman/language/core/sections.html).

the *integral of a function* $f \in \mathcal{M}_+$ is defined as the supremum of the integrals of all simple functions in $\mathcal{SF}_+$ smaller than f and formalized in [2] by

```
Definition LInt_p (f : X → R̄ ) : R̄ :=
  Rbar_lub (fun z ⇒ ∃(phi : X → ℝ) (Hphi : SF genX phi),
    nonneg phi ∧ (∀ x, phi x ≤_R̄ f x) ∧ LInt_SFp mu phi Hphi = z).
```

The proof of Tonelli's theorem relies on several properties of the integral in $\mathcal{M}_+$, such as monotonicity, positive linearity, σ-additivity, and the Beppo Levi (monotone convergence) theorem. The latter states the compatibility with the supremum: for all nondecreasing sequences $(f_n)_{n\in\mathbb{N}} \in \mathcal{M}_+$, the limit $\lim_{n\to\infty} f_n$ (which actually equals $\sup_{n\in\mathbb{N}} f_n$) is also in $\mathcal{M}_+$, and the integral-limit exchange formula holds, $\int \sup_{n\in\mathbb{N}} f_n \, d\mu = \sup_{n\in\mathbb{N}} \int f_n \, d\mu$ (see [2, Section 7.2]).

3 Lebesgue Induction Principle

Let (X, Σ) be a measurable space. The properties of the function spaces $\mathcal{M}_+$, $\mathcal{SF}_+$ and $\mathcal{IF}$ recalled in Section 2.2 suggest to represent nonnegative measurable functions by an inductive type. Indeed, functions in $\mathcal{M}_+$ are the supremum of simple functions in $\mathcal{SF}_+$, which are themselves positive linear combinations of indicator functions in $\mathcal{IF}$. Moreover, the associated structural induction principle is a common proof technique for several results in Lebesgue integration theory, among which is Tonelli's theorem as noted in [20].

In addition to `Mplus` recalled in Section 2.2, we now define the inductive type

```
Inductive Mp : (X → R̄ ) → Prop :=
  | Mp_charac : ∀ A, measurable genX A → Mp (charac A)
  | Mp_scal : ∀ a f, 0 ⩽ a → Mp f → Mp (fun x ⇒ a *_R̄ f x)
  | Mp_plus : ∀ f g, Mp f → Mp g → Mp (fun x ⇒ f x +_R̄ g x)
  | Mp_sup : ∀ f, incr_fun_seq f → (∀ n, Mp (f n)) →
              Mp (fun x ⇒ Sup_seq (fun n ⇒ f n x)).
```

where `charac A` stands for the characteristic function of A (denoted $\mathbb{1}_A$), and `incr_fun_seq f` stands for the property `∀ x n, Rbar_le (f n x) (f (S n) x)`. In other words, `Mp` is the closure of measurable characteristic functions under positive linear combination and increasing supremum.

We also have an inductive type for $\mathcal{SF}_+$ denoted by `SFp`, whose constructors are essentially the same as the first three of `Mp`. Several inductive types equivalent to `Mp` are defined in order to split the proof steps, for instance one is built over `SFp`. They are not given here for the sake of simplicity and brevity.

The important point is then the correctness of this definition, compared to the existing one. The only delicate part is to obtain the correctness result for simple functions, stated as `Lemma SFp_correct : ∀f, SFp f ↔ SFplus gen f.`

For that, from a simple function represented by a list of values of size $n+1$, we need to construct a smaller simple function associated to a sublist of size n. The tricky needed result is the following:

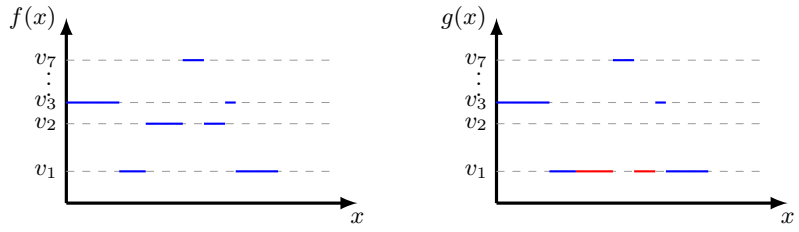

Fig. 1. Illustration of Lemma `SF_aux_cons`. The value v_2 taken by the simple function f (on the left) is replaced in g (on the right) by the value v_1 (in red).

```
Lemma SF_aux_cons :
  ∀ (f : X → ℝ) v1 v2 l, nonneg f → SF_aux genX f (v1 :: v2 :: l) →
    let g x := f x + (v1 − v2) ∗ charac (fun t ⇒ f t = v2) x in
    nonneg g ∧ SF_aux genX g (v1 :: l).
```

Given $f \in \mathcal{SF}_+$ and its associated canonical list ℓ, the lemma builds a new g in $\mathcal{SF}_+$ canonically associated with the list ℓ deprived of some item v_2. This means that on the nonempty subset $f^{-1}(\{v_2\})$, g must take one of the remaining values, v_1 as shown in Figure 1, which also provides the property $g \leqslant f$.

More precisely, let us assume that f is of the form $\sum_{v\in\{v_1,v_2\}\cup\ell} v \times \mathbb{1}_{f^{-1}(\{v\})}$ with $v_1 < v_2$ and $v_1, v_2 \notin \ell$. Then, by setting $g := f + (v_1 - v_2) \times \mathbb{1}_{f^{-1}(\{v_2\})}$, one has $g = \sum_{v\in\{v_1\}\cup\ell} v \times \mathbb{1}_{f^{-1}(\{v\})}$. Thus, g belongs to $\mathcal{SF}_+$ with a smaller list of values, and $f = g + (v_2 - v_1) \times \mathbb{1}_{f^{-1}(\{v_2\})}$ with $v_2 - v_1 \geq 0$. This is tricky for two reasons. First, we cannot set g to zero on $f^{-1}(\{v_2\})$ (as zero may be a new value, defeating the point of reducing the size of the value list); thus, the initial list must contain at least two values. Second, by proceeding the other way around and setting g to v_2 on $f^{-1}(\{v_1\})$, we cannot write f as the sum of g and a *nonnegative* value times an indicator function, as needed by the constructor `SFp_scal`, similar to `Mp_scal`.

Now, we have all the ingredients to check that the definition of `Mp` is satisfactory, that is to say that `Mp` represents $\mathcal{M}_+$ as `Mplus` already does:

```
Lemma Mp_correct : ∀f, Mp f ↔ Mplus genX f.
```

The proof is mainly based on inductions, the construction of adapted sequences `mk_adapted_seq` (see Section 2.2), and the previous lemma.

This gives us for free the following induction lemma corresponding to `Mp`:

```
Mp_ind : ∀ P : (E → ℝ̄) → Prop,
  (∀ A, measurable gen A → P (charac A)) →
  (∀ a f, 0 ⩽ a → Mp f → P f → P (fun x ⇒ a *_ℝ̄ f x)) →
  (∀ f g, Mp f → P f → Mp g → P g → P (fun x ⇒ f x +_ℝ̄ g x)) →
  (∀ f, incr_fun_seq f → (∀ n, Mp (f n)) →
    (∀ n, P (f n)) → P (fun x ⇒ Sup_seq (fun n ⇒ f n x))) →
  ∀ f, Mp f → P f.
```

This lemma can be stated informally as

Lemma 2: Lebesgue induction principle

Let P be a predicate on functions $X \to \overline{\mathbb{R}}$. Assume that P holds on $\mathcal{IF}$, and that it is compatible on $\mathcal{M}_+$ with positive linear operations and with the supremum of nondecreasing sequences:

$$\forall A, \quad A \in \Sigma \Rightarrow P(\mathbb{1}_A), \tag{3}$$

$$\forall a \in \mathbb{R}_+,\ \forall f \in \mathcal{M}_+, \quad P(f) \Rightarrow P(af), \tag{4}$$

$$\forall f, g \in \mathcal{M}_+, \quad P(f) \wedge P(g) \Rightarrow P(f+g), \tag{5}$$

$$\forall (f_n)_{n\in\mathbb{N}} \in \mathcal{M}_+, \quad (\forall n \in \mathbb{N},\ f_n \leqslant f_{n+1} \wedge P(f_n)) \Rightarrow P\left(\sup_{n\in\mathbb{N}} f_n\right). \tag{6}$$

Then, P holds on $\mathcal{M}_+$.

There are a few alternative statements of the Lebesgue induction principle. For instance, we choose to have a in $\mathbb{R}$ and not in $\overline{\mathbb{R}}$ in (4), as it makes an equivalent, but simpler to use lemma. Moreover, as noted in the Lean source code,[15] it is possible to sharpen the premises of the constructors. For instance, it may be sufficient to have in (5) simple functions that do not share the same image value, except 0, or with disjoint supports.

4 Product Measure on a Product Space

In this section, we build the product measure for the measurable subsets of a product space. This allows us to integrate on such a product space in Section 5.

Given two measure spaces (X_1, Σ_1, μ_1) and (X_2, Σ_2, μ_2), a *product measure on* $(X_1 \times X_2, \Sigma_1 \otimes \Sigma_2)$ *induced by* μ_1 *and* μ_2 is a measure μ defined on the product σ-algebra $\Sigma_1 \otimes \Sigma_2$ (defined in Section 4.1) satisfying the *box property*:

$$\forall A_1 \in \Sigma_1,\ \forall A_2 \in \Sigma_2, \quad \mu(A_1 \times A_2) = \mu_1(A_1)\,\mu_2(A_2). \tag{7}$$

To ensure the existence and uniqueness of such a product measure, we assume that μ_1 and μ_2 are σ-*finite*, i.e. that the full sets X_1 and X_2 are nondecreasing unions of subsets of finite measure (see a detailed definition in Section 4.3).

A candidate product measure is first built in three steps, see Figure 2. Firstly, X_1-*sections* (or *"vertical" cuttings*) of subsets are proved to be Σ_2-measurable. Then, the measure of sections is proved to be Σ_1-measurable. The candidate is the integral of the measure of sections. Then, this candidate is proved to be a product measure, and the product measure is guaranteed to be unique. The main argument for this construction is the monotone class theorem, whose intricate proof is not detailed here (e.g. see [9, Sec 1.6], and Section 4.3 for a

[15] https://leanprover-community.github.io/mathlib_docs/measure_theory/integral/lebesgue.html#measurable.ennreal_induction

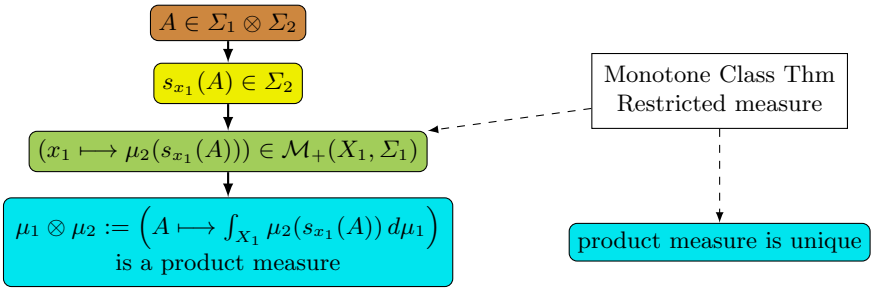

Fig. 2. Flowchart for the construction of the product measure. The fill colors refer to sections: 4.1 in brown, 4.2 in yellow, 4.3 in green, and 4.4 in blue. Dashed lines denote the use of the listed proof arguments, that were developed for the present work.

quick presentation). It is used twice: for the measurability of the measure of sections, and for the uniqueness of the product measure.

The definition of the product σ-algebra is first reviewed in Section 4.1. Then, Section 4.2 is dedicated to sections, and Section 4.3 to the measure of sections. Finally, the existence and uniqueness of the product measure is in Section 4.4.

4.1 Product σ-algebra

Let us detail the notion of product σ-algebra that was introduced in [2]. Given two measurable spaces (X_1, Σ_1) and (X_2, Σ_2), the *product σ-algebra on* $X_1 \times X_2$ is the σ-algebra $\Sigma_1 \otimes \Sigma_2$ generated by the products of measurable subsets:

$$\Sigma_1 \otimes \Sigma_2 := \ \sigma\text{-algebra generated by } \Sigma_1 \overline{\times} \Sigma_2 := \{A_1 \times A_2 \mid A_1 \in \Sigma_1 \wedge A_2 \in \Sigma_2\}.$$

Given generators `genX1` and `genX2` for Σ_1 and Σ_2, the generator $\Sigma_1 \overline{\times} \Sigma_2$ is denoted in Coq by `Product_Sigma_algebra genX1 genX2`. It is proved in [2, Sec. 4.3] that $\Sigma_1 \otimes \Sigma_2$ is also the σ-algebra generated by $\mathrm{gen}(\Sigma_1) \cup \{X_1\} \overline{\times} \mathrm{gen}(\Sigma_2) \cup \{X_2\}$. This generator is denoted in Coq by `Gen_Product genX1 genX2`, and simply by `genX1xX2` in the sequel. Symmetrically, `genX2xX1` represents `Gen_Product genX2 genX1`.

4.2 Section of Subset

The notion of *section* consists in keeping one of the variables fixed (see Figure 3). Given a subset A of $X_1 \times X_2$ and a point $x_1 \in X_1$, the *X_1-section of A at x_1* is the subset of X_2 defined by $s_{x_1}(A) := \{x_2 \in X_2 \mid (x_1, x_2) \in A\}$.

```
Definition section (x1 : X1) (A : X1 * X2 → Prop) (x2 : X2) : Prop := A (x1, x2).
```

Sections commute with most set operations. For example, they are compatible with the empty set ($s_{x_1}(\varnothing) = \varnothing$), the complement ($s_{x_1}(A^c) = s_{x_1}(A)^c$),

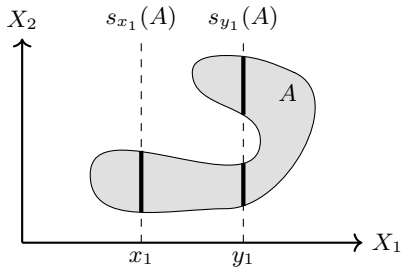

Fig. 3. X_1-sections of a subset A of $X_1 \times X_2$ at points x_1 and y_1.

countable union and intersection, and are monotone. Sections also satisfy the following box property: for all subsets $A_1 \subseteq X_1$, $A_2 \subseteq X_2$, and point $x_1 \in X_1$,

$$x_1 \in A_1 \Rightarrow s_{x_1}(A_1 \times A_2) = A_2 \quad \text{and} \quad x_1 \notin A_1 \Rightarrow s_{x_1}(A_1 \times A_2) = \varnothing. \tag{8}$$

Then, we prove that, if a subset A is $\Sigma_1 \otimes \Sigma_2$-measurable, then its X_1-sections at any point in X_1 are Σ_2-measurable. As measurability is an inductive type, the proof is a simple induction on the hypothesis.

```
Lemma section_measurable :
  ∀ A x1, measurable genX1xX2 A → measurable genX2 (section x1 A).
```

4.3 Measurability of Measure of Section

As sections are measurable (see Section 4.2), one can take their measure. In Section 4.4, the product measure is defined as the integral of the measure of sections, but before that, we have to prove nonnegativity and measurability of these functions. More precisely, that for all $\Sigma_1 \otimes \Sigma_2$-measurable subsets A, the function $(x_1 \mapsto \mu_2(s_{x_1}(A)))$ belongs to $\mathcal{M}_+(X_1, \Sigma_1)$.

The nonnegativity property directly follows from that of measures. The proof of measurability goes in two stages: firstly when the measure μ_2 is assumed to be *finite* (i.e. when $\mu_2(X_2)$ is finite), and then in the more general σ-finite case. The first stage is quite high-level; it relies on the monotone class theorem. The second stage extends the first one by means of restricted measures.

The measure of sections is represented in Coq by the total function

```
Definition meas_section (A : X1 * X2 → Prop) (x1 : X1) : ℝ̄ := muX2 (section x1 A).
```

Then, the first stage of the proof is stated in Coq as

```
Lemma meas_section_Mplus_finite : ∀A, is_finite_measure muX2 →
  measurable genX1xX2 A → Mplus genX1 (meas_section A).
```

Let $\mathcal{S}$ be the set of measurable subsets satisfying the property to prove,

$$\mathcal{S} := \left\{ A \in \Sigma_1 \otimes \Sigma_2 \,|\, \left(x_1 \longmapsto \mu_2(s_{x_1}(A))\right) \in \mathcal{M}_+(X_1, \Sigma_1) \right\}.$$

It suffices to show that $\Sigma_1 \otimes \Sigma_2 \subseteq \mathcal{S}$. Firstly, $\mathcal{S}$ is proved to contain the generator $\overline{\Sigma} := \Sigma_1 \overline{\times} \Sigma_2$ of $\Sigma_1 \otimes \Sigma_2$ (see Section 4.1). Then, it is proved to contain the algebra of sets generated by $\overline{\Sigma}$ (i.e. the closure of $\overline{\Sigma}$ under complement and finite union). Then, $\mathcal{S}$ is also proved to be a monotone class, i.e. closed under monotone countable union and intersection. This step uses the finiteness assumption on μ_2, and continuity from below and from above (see Section 2.2). And finally, we conclude by applying the following corollary of the monotone class theorem (with `X` := `X1 * X2`, `P` := $\mathcal{S}$, and `genX` := $\overline{\Sigma}$) which states that if a monotone class contains the smallest algebra of sets containing `genX`, then it also contains the smallest σ-algebra containing `genX`.

```
Theorem monotone_class_Prop :
  ∀ P : (X → Prop) → Prop, is_Monotone_class P →
    Incl (Algebra genX) P → Incl (Sigma_algebra genX) P.
```

Note that `Incl` denotes the inclusion of subsets of the power set of X.

In the second stage, the measure μ_2 is supposed to be σ-finite. Thus, there exists a nondecreasing sequence $(B_n)_{n\in\mathbb{N}} \in \Sigma_2$ such that $X_2 = \bigcup_{n\in\mathbb{N}} B_n$, and $\mu_2(B_n)$ is finite for all $n \in \mathbb{N}$. Then, for each $n \in \mathbb{N}$, the *restricted measure* $\mu_2^n := (A_2 \in \Sigma_2 \longmapsto \mu_2(A_2 \cap B_n) \in \overline{\mathbb{R}}_+)$ is proved to be a finite measure. Thus, the previous result applies,

$$\forall A \in \Sigma_1 \otimes \Sigma_2, \quad (x_1 \longmapsto \mu_2^n(s_{x_1}(A))) \in \mathcal{M}_+(X_1, \Sigma_1).$$

Moreover, from the properties of sections (see Section 4.2) and from the continuity from below of μ_2, for all $A \in \Sigma_1 \otimes \Sigma_2$ and $x_1 \in X_1$, we have

$$\mu_2(s_{x_1}(A)) = \mu_2\left(\bigcup_{n\in\mathbb{N}} s_{x_1}(A) \cap B_n\right) = \sup_{n\in\mathbb{N}} \mu_2^n(s_{x_1}(A)).$$

Finally, the closedness of $\mathcal{M}_+(X_1, \Sigma_1)$ under supremum (see Section 2.2) concludes the proof. Thus, the lemma in the σ-finite case holds,

```
Lemma meas_section_Mplus_sigma_finite :
  ∀ A, is_sigma_finite_measure muX2 →
    measurable genX1xX2 A → Mplus genX1 (meas_section A).
```

Note that from (8), the measure of the section of a box reads

$$\forall A_1 \in \Sigma_1,\ \forall A_2 \in \Sigma_2, \quad (x_1 \longmapsto \mu_2(s_{x_1}(A_1 \times A_2))) = \mu_2(A_2)\, \mathbb{1}_{A_1}. \tag{9}$$

4.4 Existence and Uniqueness of the Product Measure

As the measures of sections belong to $\mathcal{M}_+$ (see Section 4.3), one can take their integral. The candidate product measure is the function defined on the product σ-algebra $\Sigma_1 \otimes \Sigma_2$ (see Section 4.1) by $(\mu_1 \otimes \mu_2)(A) := \int_{X_1} \mu_2(s_{x_1}(A))\, d\mu_1$,

```
Definition meas_prod_meas (A : X1 * X2 → Prop) : R :=
  LInt_p muX1 (meas_section muX2 A).
```

We easily deduce that this candidate function is both nonnegative and equal to zero on the empty set. The σ-additivity property is obtained by means of the σ-additivity of the integral (see Section 2.2), and of the measure μ_2. This proves that the candidate is a measure, and that we can instantiate the record defining the product measure `meas_prod` as an object of type measure (see Section 2.2), so all the proved results on measures are available.

Moreover, Equation (9), and the positive linearity of the integral ensure the box property (7), thus making `meas_prod` a product measure.

Product measures are proved to keep the finiteness, or σ-finiteness, property of the initial measures μ_1 and μ_2. Then, the proof of the uniqueness of the product measure follows exactly the same path as for the measurability of the measure of sections (see Section 4.3). Firstly, when the measures μ_1 and μ_2 are finite, we introduce two (finite) product measures m and $\tilde{m}$ induced by μ_1 and μ_2, i.e. both satisfying (7). The set $\mathcal{S} \stackrel{\text{def.}}{=} \{A \in \Sigma_1 \otimes \Sigma_2 \,|\, m(A) = \tilde{m}(A)\}$ is proved to contain $\Sigma_1 \otimes \Sigma_2$ using `monotone_class_Prop`, which shows uniqueness. Then, the result is extended to σ-finite measures by means of restricted measures.

5 Tonelli's Theorem

With the product measure built in Section 4, we can now consider integration on a product space. As in Section 4, we assume that the measures are σ-finite, which ensures the existence and uniqueness of the product measure.

This section addresses the proof of Tonelli's theorem that allows to compute a double integral on a product space by integrating successively with respect to each variable, either way. Besides the following formulas, the theorem also states measurability properties that ensure the legitimacy of all integrals (see Theorem 1):

$$\int_{X_1 \times X_2} f \, d(\mu_1 \otimes \mu_2) = \int_{X_1} \left(\int_{X_2} f \, d\mu_2 \right) d\mu_1 \tag{10}$$

$$= \int_{X_2} \left(\int_{X_1} f \, d\mu_1 \right) d\mu_2. \tag{11}$$

Similarly to the process used in Section 4, the iterated integral (right-hand side of (10)) is built in three steps, see Figure 4. Firstly, X_1-sections of functions are proved to be Σ_2-measurable. Then, the integral (in X_2) of sections of functions is proved to be Σ_1-measurable. And the iterated integral is the integral (in X_1) of the integral (in X_2) of the sections of functions. Finally, Formula (10) is proved, and then (11) is deduced from the latter by a swap of variables relying both on a change of measure and on the uniqueness of the product measure. The main argument for this proof is the Lebesgue induction principle (see Section 3). It is used twice: for the measurability of the integral of sections of functions together with the first Tonelli formula, and for the change-of-measure formula.

Section 5.1 is dedicated to sections of functions, and Section 5.2 to the iterated integral and the proof of the first formula of Tonelli's theorem. Finally, the full proof of Tonelli's theorem is obtained in Section 5.3.

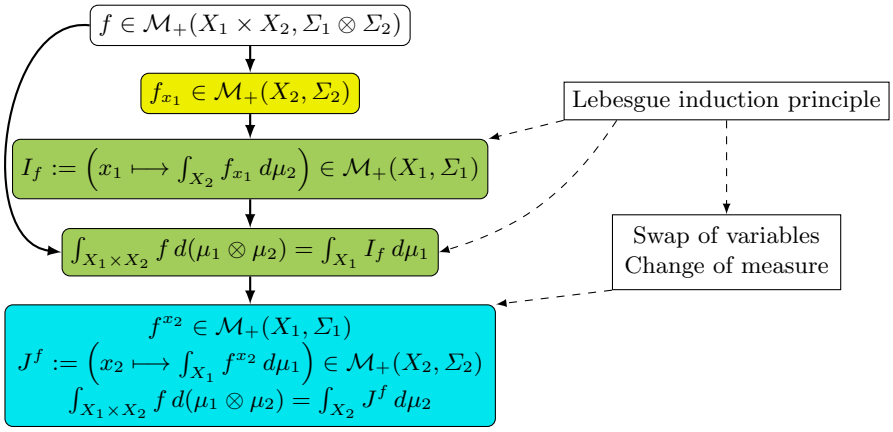

Fig. 4. Flowchart for the construction of the iterated integrals on a product space. The fill colors refer to sections: 5.1 in yellow, 5.2 in green, and 5.3 in blue. Dashed lines denote the use of the listed proof arguments, that were developed for the present work.

5.1 Section of Function

Similarly to Section 4.2, given a numeric function $f : X_1 \times X_2 \to \overline{\mathbb{R}}$ and $x_1 \in X_1$, the X_1-*section of* f *at* x_1 is the partial application $f_{x_1} := (x_2 \mapsto f(x_1, x_2))$.

```
Definition section_fun (x1 : X1) (f : X1 * X2 → R̄ ) (x2 : X2) : R̄ := f (x1, x2).
```

From the measurability of sections of subsets, we deduce that, if f belongs to $\mathcal{M}_+(X_1 \times X_2, \Sigma_1 \otimes \Sigma_2)$, then its X_1-sections are in $\mathcal{M}_+(X_2, \Sigma_2)$.

```
Lemma section_fun_Mplus :
  ∀ f x1, Mplus genX1xX2 f → Mplus genX2 (section_fun x1 f).
```

Symmetrically, for all $x_2 \in X_2$, we introduce the X_2-*section of* f *at* x_2, the partial application with respect to the second variable, $f^{x_2} := (x_1 \mapsto f(x_1, x_2))$.

5.2 Iterated Integral and the First Formula of Tonelli's Theorem

As sections of functions are nonnegative and Σ_2-measurable (see Section 5.1), one can take their integral (in X_2). For all functions $f \in \mathcal{M}_+(X_1 \times X_2, \Sigma_1 \otimes \Sigma_2)$, we define $I_f := \left(x_1 \longmapsto \int_{X_2} f_{x_1}\, d\mu_2\right)$,

```
Definition LInt_p_section_fun (f : X1 * X2 → R̄ ) x1 : R̄ :=
  LInt_p muX2 (section_fun x1 f).
```

The iterated integral corresponds to integrating once more (in X_1), but one must first establish that $I_f \in \mathcal{M}_+(X_1, \Sigma_1)$. The nonnegativity result directly follows from the monotonicity of the integral (see Section 2.2). The general measurability result and the first Tonelli formula (10), are proved by means of the Lebesgue induction principle of Section 3.

The function $I := (f \mapsto I_f)$ is shown monotone and positive linear. For all $x_1 \in X_1$, we have $I_{\mathbb{1}_A}(x_1) = \mu_2(s_{x_1}(A))$. And from the Beppo Levi (monotone convergence) theorem (see Section 2.2), the function I commutes with the supremum: for all nondecreasing sequence $(f_n)_{n\in\mathbb{N}}$ in $\mathcal{M}_+(X_1 \times X_2, \Sigma_1 \otimes \Sigma_2)$, $I_{\sup_{n\in\mathbb{N}} f_n} = \sup_{n\in\mathbb{N}} I_{f_n}$.

Let `P0 f := Mplus genX1 (LInt_p_section_fun f)` be the predicate of the nonnegativity and measurability of I_f, of type `(X1 * X2 → R̄) → Prop`. Then, previous formulas and closedness properties of $\mathcal{M}_+$ (see Section 2.2) provide the compatibility of `P0` with indicator functions, positive linearity, and the supremum of nondecreasing sequences. For instance, we have

```
Lemma LInt_p_section_fun_measurable_plus :
  ∀ f g, Mplus genX1xX2 f → Mplus genX1xX2 g →
    P0 f → P0 g → P0 (fun x ⇒ f x +_R̄ g x).
```

Let us now define the predicate `P` of the existence of the iterated integral (granted by `P0`) and the validity of the first Tonelli formula of (10):

```
Let P (f : X1 * X2 → R̄) : Prop :=
  P0 f ∧ LInt_p meas_prod f = LInt_p muX1 (LInt_p_section_fun f).
```

where `meas_prod` is the product measure defined in Section 4.4. Again, the compatibility of `P` with indicator functions, positive linearity, and the supremum is easily obtained from the previous results. For instance, we have

```
Lemma LInt_p_section_fun_meas_prod_Sup_seq :
  ∀ f, incr_fun_seq f → Mplus_seq genX1xX2 f →
    (∀ n, P (f n)) → P (fun x ⇒ Sup_seq (fun n ⇒ f n x)).
```

Now, the first part of Tonelli's theorem (10) can be stated in Coq as

```
Lemma Tonelli_aux1 : ∀f, Mplus genX1xX2 f →
  Mplus genX1 (LInt_p_section_fun f) ∧
  LInt_p meas_prod f = LInt_p muX1 (LInt_p_section_fun f).
```

Its proof is a direct application of the Lebesgue induction principle (see Section 3) with the predicate `P`, as all the premises are already shown.

5.3 Change of Measure, Second Formula, and Tonelli's Theorem

There is no doubt that the second formula (11) can be proved using the same path as the first claim: use sections with respect to the second variable, define J^f (see Figure 4), prove $J^f \in \mathcal{M}_+$ and the equality by the Lebesgue induction principle. This would be easy, but pretty long and redundant. Instead, we have exploited the "symmetry" between the right-hand sides of both formulas. The first idea is a simple exchange of the roles of the two variables that expresses the previous result for functions of type `X2 * X1 → R̄`. And then, the difficult part is a change of measure that brings back to the target type `X1 * X2 → R̄`.

The change of measure is an application of the concept of *image measure* (e.g. see [9, Sec 2.6]), also called *pushforward measure* as the measure is transported between σ-algebras, here from $\Sigma_2 \otimes \Sigma_1$ to $\Sigma_1 \otimes \Sigma_2$.

Change of Measure. Let (X, Σ) and $(Y, \mathcal{T})$ be measurable spaces. Let $h : X \to Y$ be a function and `Mh` be a proof of its measurability. Let μ be a measure on (X, Σ). The *image measure of* μ *by* h is the measure on $(Y, \mathcal{T})$ defined by $h_{\#}\mu := \mu \circ h^{-1}$, and denoted in Coq by `meas_image h Mh mu`. The proof that it is indeed a measure directly follows from the measure properties of μ, and `Mh`.

Now, given $g \in \mathcal{M}_+(Y, \mathcal{T})$, the compatibility of measurability with the composition of functions provides $g \circ h \in \mathcal{M}_+(X, \Sigma)$, and one has the change-of-measure formula: $\int_Y g\, d(h_{\#}\mu) = \int_X g \circ h\, d\mu$.

```
Lemma LInt_p_change_meas : ∀g, Mplus genY g →
  LInt_p (meas_image h Mh mu) g = LInt_p mu (fun x ⇒ g (h x)).
```

The proof follows the Lebesgue induction principle with the predicate `P` corresponding to the formula. Again, the compatibility of `P` with indicator functions, positive linearity, and the supremum follows from properties of the integral, such as positive linearity and the Beppo Levi (monotone convergence) theorem.

Swap and Second Formula. Using Section 4.4, let $\mu_{12} := \mu_1 \otimes \mu_2$ be the product measure on the product space $(X_1 \times X_2, \Sigma_1 \otimes \Sigma_2)$ induced by μ_1 and μ_2. In Coq, `muX1xX2` := `meas_prod muX1 muX2`. Symmetrically, let $\mu_{21} := \mu_2 \otimes \mu_1$ be the product measure on $(X_2 \times X_1, \Sigma_2 \otimes \Sigma_1)$. In Coq, `muX2xX1` := `meas_prod muX2 muX1`. Let $h := (x_2, x_1) \mapsto (x_1, x_2)$ be the swap of variables. The image measure $h_{\#}\mu_{21}$ is also proved to be a product measure on $(X_1 \times X_2, \Sigma_1 \otimes \Sigma_2)$ induced by μ_1 and μ_2. In Coq, `meas_prod_swap` := `meas_image h Mh muX2xX1`.

Now, let $f \in \mathcal{M}_+(X_1 \times X_2, \Sigma_1 \otimes \Sigma_2)$. One has $f \circ h \in \mathcal{M}_+(X_2 \times X_1, \Sigma_2 \otimes \Sigma_1)$, and using the section with respect to the second variable (see Section 5.1),

$$\forall x_2 \in X_2, \quad f^{x_2} := (x_1 \longmapsto f(x_1, x_2)) = (x_1 \longmapsto f \circ h(x_2, x_1)) = (f \circ h)_{x_2}. \quad (12)$$

We then deduce

$$\int_{X_1 \times X_2} f\, d\mu_{12} \overset{(a)}{=} \int_{X_1 \times X_2} f\, d(h_{\#}\mu_{21}) \overset{(b)}{=} \int_{X_2 \times X_1} f \circ h\, d\mu_{21}$$
$$\overset{(c)}{=} \int_{X_2} \left(\int_{X_1} (f \circ h)_{x_2}\, d\mu_1 \right) d\mu_2 \overset{(d)}{=} \int_{X_2} \left(\int_{X_1} f^{x_2}\, d\mu_1 \right) d\mu_2.$$

The uniqueness of the product measure of Section 4.4 yields $h_{\#}\mu_{21} = \mu_{12}$, thus gives (a). The above change-of-measure formula gives (b). The first formula of Tonelli's theorem (10) applied to $X_2 \times X_1$ gives (c), and Equation (12) gives (d). With `swap f` denoting $f \circ h$, the second part of Tonelli's theorem (11) is

```
Lemma Tonelli_aux2 : ∀f, Mplus genX1xX2 f →
  Mplus genX2 (LInt_p_section_fun muX1 (swap f)) ∧
  LInt_p meas_prod_swap f = LInt_p muX2 (LInt_p_section_fun muX1 (swap f)).
```

Statement of Tonelli's Theorem. Finally, assuming that X_1 and X_2 are non-empty and that μ_1 and μ_2 are σ-finite measures, we have (a more comprehensive theorem legitimating of all integrals is also provided as `Theorem Tonelli`):

```
Lemma Tonelli_formulas : ∀f, Mplus genX1xX2 f →
  LInt_p muX1xX2 f = LInt_p muX1 (LInt_p_section_fun muX2 f) ∧
  LInt_p muX1xX2 f = LInt_p muX2 (LInt_p_section_fun muX1 (swap f)).
```

6 Conclusion and perspectives

This paper is devoted to the full formal proof of Tonelli's theorem. An original point is the definition of nonnegative measurable functions as an inductive type. It is proved equivalent to the usual mathematical definition, and leads to a useful induction scheme. Although the Lebesgue induction principle is present in other works such as [20], we have not seen its construction from an inductive type in the literature.

To achieve this proof, we have also formalized in Coq generic results and constructions such as the monotone class theorem, restricted measures, image measures, and a change-of-measure formula for the integral. The latter, combined with a swap of variables, has prevented redundancies in our proofs.

This work confirms that the library we are developing, in line with the choices of the Coquelicot library, is rather comprehensive and usable. First, this work has resulted in few additions in the core of the library, except for the inductive definition for $\mathcal{M}_+$ (related to the needed Lebesgue induction principle). Second, both Coq and the library seem easy to learn, as one author was a Coq novice at the beginning of this work.

After Tonelli's theorem on nonnegative measurable functions, the natural extension is to prove Fubini's theorem. It provides the same formulas for integrable functions with an arbitrary sign, or taking their values in a Banach space when using the Bochner integral [4]. We can also take inspiration from [20], in particular for the "marginal integral" to handle finitary Cartesian products.

Our long-term purpose is to formally prove the correctness of parts of a library implementing the Finite Element Method, which is used to compute approximated solutions of Partial Differential Equations (PDEs). We already formalized the Lax–Milgram theorem [1], one of the key ingredients to numerically solve PDEs, and we need to build suitable Hilbert functional spaces on which to apply it. The target candidates are the Sobolev spaces, such as H^1, which represents square-integrable functions with square-integrable first derivatives. Of course, this will involve the formalization of the L^p Lebesgue spaces as complete normed vector spaces, and parts of the distribution theory [19].

References

1. Sylvie Boldo, François Clément, Florian Faissole, Vincent Martin, and Micaela Mayero. A Coq formal proof of the Lax–Milgram theorem. In *Proc. of the 6th ACM SIGPLAN Internat. Conf. on Certified Programs and Proofs (CPP 2017)*, pages 79–89. Association for Computing Machinery, New York, 2017.

2. Sylvie Boldo, François Clément, Florian Faissole, Vincent Martin, and Micaela Mayero. A Coq formalization of Lebesgue integration of nonnegative functions. *J. Autom. Reason.*, 2021. URL: https://hal.inria.fr/hal-03471095/.
3. Sylvie Boldo, François Clément, Jean-Christophe Filliâtre, Micaela Mayero, Guillaume Melquiond, and P. Weis. Wave equation numerical resolution: a comprehensive mechanized proof of a C program. *J. Autom. Reason.*, 50(4):423–456, 2013.
4. Sylvie Boldo, François Clément, and Louise Leclerc. A Coq formalization of the Bochner integral, January 2022. URL: https://hal.inria.fr/hal-03516749/.
5. Sylvie Boldo, Catherine Lelay, and Guillaume Melquiond. Coquelicot: A user-friendly library of real analysis for Coq. *Math. Comput. Sci.*, 9(1):41–62, 2015.
6. Sylvie Boldo, Catherine Lelay, and Guillaume Melquiond. Formalization of real analysis: A survey of proof assistants and libraries. *Math. Struct. Comput. Sci.*, 26(7):1196–1233, 2016. URL: https://hal.inria.fr/hal-00806920/.
7. Sylvie Boldo and Guillaume Melquiond. Flocq: A unified library for proving floating-point algorithms in Coq. In *Proc. of the IEEE 20th Symposium on Computer Arithmetic (ARITH-20)*, pages 243–252. IEEE, 2011.
8. François Clément and Vincent Martin. Lebesgue integration. Detailed proofs to be formalized in Coq. Research Report RR-9386, Inria, Paris, 2021. Version 2. URL: https://hal.inria.fr/hal-03105815v2.
9. Donald L. Cohn. *Measure Theory*. Birkhäuser, New York, 2nd edition, 2013.
10. The Coq reference manual. URL: https://coq.inria.fr/refman/.
11. Noboru Endou. Fubini's theorem. *Formaliz. Math.*, 27(1):67–74, 2019.
12. Thierry Gallouët and Raphaèle Herbin. *Mesure, intégration, probabilités*. Ellipses Edition Marketing, 2013. In French.
13. John Harrison. The HOL Light theory of Euclidean space. *J. Autom. Reason.*, 50(2):173–190, 2013. URL: https://doi.org/10.1007/s10817-012-9250-9.
14. Johannes Hölzl and Armin Heller. Three chapters of measure theory in Isabelle/HOL. In Marko van Eekelen, Herman Geuvers, Julien Schmaltz, and Freek Wiedijk, editors, *Proc. of the 2nd Internat. Conf. on Interactive Theorem Proving*, volume 6898 of *LNCS*, pages 135–151. Springer, Berlin - Heidelberg, 2011.
15. Henri Léon Lebesgue. *Leçons sur l'intégration et la recherche des fonctions primitives professées au Collège de France*. Cambridge University Press, Cambridge, 2009. Reprint of the 1904 original [Gauthier-Villars, Paris]. In French.
16. David R Lester. Topology in PVS: continuous mathematics with applications. In *Proc. of the 2nd Workshop on Automated Formal Methods (AFM 2007)*, pages 11–20, 2007. URL: https://doi.org/10.1145/1345169.1345171.
17. Francis Maisonneuve. *Mathématiques 2 : Intégration, transformations, intégrales et applications - Cours et exercices*. Presses de l'École des Mines, 2014. In French.
18. Micaela Mayero. *Formalisation et automatisation de preuves en analyses réelle et numérique*. Thèse de doctorat, Université Paris VI, 2001. In French.
19. Laurent Schwartz. *Théorie des distributions*. Hermann, Paris, 2nd edition, 1966. 1st edition in 1950–1951. In French.
20. Floris van Doorn. Formalized Haar measure. In L. Cohen and C. Kaliszyk, editors, *Proc. of the 12th Internat. Conf. on Interactive Theorem Proving*, volume 193 of *LIPIcs*, pages 18:1–18:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.