

Formalization of the real interpolation theorem for the Carleson project

Jim Portegies Michael Rothgang

October 7, 2026

Abstract

The Marcinkiewicz real interpolation theorem is a cornerstone result in harmonic analysis. We formalize a generalized version in the Lean theorem prover, which enables proving weak type estimates for the Hardy–Littlewood maximal function. To our knowledge, this is the first formalization in any theorem prover. Our proof is a combination of different modern proofs, allowing for uniform treatment of different cases and yielding a uniform explicit constant. This work was motivated from and part of the formalization of Carleson’s theorem.

1 Introduction

The Marcinkiewicz real interpolation theorem is a standard foundational result in harmonic analysis, an area that has received relatively little attention in the formal mathematics community so far. We formalize its proof in the Lean proof assistant [MU21], making sure to be general enough for applications such as the Hardy–Littlewood maximal function. Our work has been used as part of the Carleson project [Bec+26].

1.1 Motivation and relevance

The real interpolation theorem is a result in harmonic analysis that allows to derive new bounds on subadditive operators based on two given bounds. The new bounds of the operator can be interpreted as lying in between the original bounds, and are obtained by interpolating between the original bounds. The theorem is similar to the Riesz–Thorin interpolation theorem, although it also works for nonlinear operators, and allows for weaker endpoint bounds. As the interpolated bounds get closer to the endpoints, they (naturally) deteriorate.

A famous application of the real interpolation theorem is in the proof that the Hardy–Littlewood maximal function is bounded from L^p to L^p , which can be derived from a direct bound from L^∞ to L^∞ , and the fact that the maximal function is weakly bounded from L^1 to L^1 . Another example application is in proving the boundedness of the Hilbert transform.

After these initial results, interpolation theory has found many more applications. This makes the interpolation theorem a foundational result, appropriate for foundational libraries such as Lean’s mathematical library `Mathlib`¹.

The formalization was carried out as part of the Carleson project [Bec+26], the first project formalizing current research results in analysis in near-real time. The Carleson project formalized a modern generalization [Bec+24] of Lennart Carleson’s classical result about pointwise a.e. convergence of Fourier series. As there had been little prior work formalizing harmonic analysis, completing the project required developing more background material in this field: one such result is the real interpolation theorem. It is a key prerequisite for formalizing the proof of the generalized Carleson theorem: for example, it is applied to the Hardy–Littlewood maximal function to deduce it is of strong type L^p , for $p \in (1, \infty)$.

1.2 Related work

Our work is the first formalization of the real interpolation theorem in any theorem prover. In general, harmonic analysis has not received much attention in the mathematical formalization community so far. There are formalizations of the basics of Fourier series in Lean’s `Mathlib` ², Isabelle/HOL² and HOL Light. Mizar’s mathematical library only knows about discrete Fourier transforms [TS06], but contains Minkowski’s inequality for infinite sums [Suz05]. `Mathlib` also contains some Fourier analysis thanks to Doll’s work on tempered distributions [Dol25].

Prior to the Carleson project, the Hardy–Littlewood maximal function had only been formalized in Rocq [AS24b; AS24a], for functions $\mathbb{R} \rightarrow \mathbb{R}$. In contrast, the Carleson project defines the maximal operator for any function $X \rightarrow E$ from a measurable pseudo-metric space into any space with an extended norm (see Subsection 4.5).³

When we started the formalization, many measure-theoretic results were already available in the `Mathlib` library, but definitions of weak norms and weak-type estimates were missing. The Minkowski integral inequality was also missing.

1.3 Main results

The main contribution of this work is formalizing the real interpolation theorem in Lean. This included a proof of the Minkowski integral inequality (though perhaps not yet in the ideal generality). To demonstrate the usability of this theorem, we emphasize that the theorem we proved applies to the general version of the Hardy–Littlewood maximal function studied in the Carleson project. Indeed, it was proven that the Hardy–Littlewood maximal function has strong

¹See [mat20] and <https://github.com/leanprover-community/mathlib4/>

²<https://isa-afp.org/entries/Fourier.html>

³Another difference is that the maximal function in the Carleson project takes values in $\mathbb{R}_{\geq 0}$ [Bec+26, § 4.2.3].

type (p, p) for any $1 < p$. (This proof involved significant contributions from the authors, but also Pietro Monticone, James Sundstrom, Jeremy Tan and Lua Viana Reis in the Carleson project.)

In the course of formalizing the result, we ran into various challenges. As we resolved these challenges, it turned out that we could get a more general version of the real interpolation theorem than we initially set out to prove. As an example, the current theorem statement works for functions taking values in additive monoids endowed with an extended norm (see Subsection 4.5), rather than normed vector spaces.

Even though we had consulted several references before embarking on the proof, in retrospect we should have conducted an even more careful literature search and strategic planning. When we started working on the proof, preliminary work on the formalization suggested to follow Folland’s proof of the real interpolation theorem. Although Folland’s proof has its advantages, some of the challenges in the formalization were due to the fact that we started from this proof. We managed to generalize it in various directions, but arguably the most natural generalization, to the interpolation theorem for Lorentz spaces, was made harder by this initial choice: as explained in Section 3, the real interpolation theorem can be proven truncations or decreasing rearrangements. While we ended up choosing a different kind of truncation, we stuck to Folland’s approach of using truncations. Both approaches work equally well in L^p spaces; for Lorentz spaces, the decreasing rearrangements strategy is advantageous. At the same time, this Lorentz space generalization would have required significant additional work (to formalise Lorentz spaces and their basic theory). We estimate this to have doubled or tripled the overall size of this project.⁴ Pursuing this extension in separate steps also has the advantage of formalising both proofs: both truncations and decreasing rearrangements are basic tools deserving to be formalised.

All our code is publicly available in the Carleson project; the proof of the real interpolation theorem specifically is in the `ToMathlib/RealInterpolation` directory.  Overall, our formalization comprises about 5000 lines of new Lean code, as well as extensive modifications (about 2500 lines) to the `Mathlib` library made for this project.

Throughout this article, we show code excerpts from our formalisation. We use the clickable symbol  to link to code in our project or in `Mathlib` (as appropriate). All code excerpts refer to a fixed, frozen version of this repository  and the corresponding version of `Mathlib` (on August 22, 2026). Some code excerpts have been slightly edited for readability.

Structure of this article. We will explain some of these challenges in Section 4. Before that, it will be useful to first recall the real interpolation theorem (Section 2), after which we give a close-reading of some of the proofs and explain

⁴In the Carleson, after completing the original project, the formalisation was extended to Hunt’s generalisation of Carleson’s theorem (from continuous to L^p functions, $p \geq 1$). This is complete modulo the proof of the Lorentz space real interpolation theorem; in the process, about 5000 lines of code on Lorentz spaces, Lorentz norms and refactoring a `mathlib` definition were added.

the proof we chose to formalize (Section 3).

Acknowledgements.

We would like to thank Floris van Doorn for initiating the Carleson project, which provided the impetus to this formalization project, and for many useful discussions. We would like to thank Lua Viana Reis and Joris Roos for general discussions around interpolation theory. We are also grateful to Yael Dillies for making mathlib’s lemma statements use `enorm` when easily possible⁵, and to Leo Diederich for some help with generalizing some lemmas to `enorms`⁶. M.R. was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy – EXC-2047/1 – 390685813 and by ERC Synergy Grant 101224275. Part of this work was conducted at a workshop at Institut Pascal at Université Paris-Saclay with the support of the program “Investissements d’avenir” ANR-11-IDEX-0003-01.

2 The real interpolation theorem

Let us begin by stating the real interpolation theorem along with its necessary prerequisites. Throughout this section, let $(X, \mathcal{M}, \mu)$ and $(X', \mathcal{N}, \nu)$ be measure spaces, let E be a topological space with the structure of an additive monoid with a compatible extended seminorm (see Subsection 4.5), and let E' be a topological space with a continuous extended seminorm.

Definition 2.1 (Weak p -norm [↗]). *Let $p \in (0, \infty]$. If $p \in (0, \infty)$, we define the weak p -norm of $f: X \rightarrow E$ as*

$$[f]_p := \left(\sup_{t \in (0, \infty)} t^p \mu(\{\|f(x)\|_E > t\}) \right)^{1/p}.$$

If $p = \infty$, the weak p -norm is defined to equal the strong ∞ -norm,

$$[f]_\infty := \|f\|_\infty.$$

Despite its name, the weak p -norm is not a norm, not even for $p \geq 1$, since the triangle inequality fails in general.

In Lean, the definition of the weak p -norm parallels the one given above.

```
def wnorm (f : α → ε) (p : ℝ ≥ 0 ∞) (μ : Measure α) : ℝ ≥ 0 ∞ :=
  if p = ∞ then eLpNormEssSup f μ else
  ⌊ t : ℝ ≥ 0, t * (μ {x | t < ‖f x‖_ε}) ⌋ ^ p.toReal-1
```

Here `eLpNormEssSup` is the strong L^∞ -norm (i.e., the ess sup of a function); the notation `⌊` denotes the supremum.

⁵<https://github.com/leanprover-community/mathlib4/pull/20806>

⁶pull request 381

In contrast to the proof given by Folland, we will eventually allow functions that take values in spaces that are more general than $\mathbb{C}$. In this context, L^p functions are not just assumed to be almost-everywhere measurable, but actually almost-everywhere strongly measurable [HVVW16, Definition 1.2.15]. If the codomain of the functions is a pseudometrizable Borel space with second countable topology, the two measurability concepts are equivalent $\square$, but in general, strong measurability is stronger than measurability.

We say that a function is *weak- L^p* if it is almost-everywhere strongly measurable and its weak p -norm is finite. Every function that is strong L^p is weak L^p , but not the other way around. When an operator is bounded from strong L^p to weak L^q , we call say it has *weak type* (p, q) :

Definition 2.2 (Weak type $\square$). *Let $p, q \in [0, \infty]$. We say that $T: (X \rightarrow E) \rightarrow (X' \rightarrow E')$ has weak type (p, q) if it is bounded as a map from $L^p(X)$ to weak $L^q(X')$. In other words, there exists a constant $c > 0$ such that for all $f \in L^p(X)$, the function Tf is weak- L^q with $[Tf]_q \leq c\|f\|_p$.*

In Lean, the definition reads as follows.

```
def HasWeakType
  [ENorm E] [ENorm E'] [TopologicalSpace E] [TopologicalSpace E']
  (T : (α → E) → (α' → E')) (p p' : ℝ≥0∞)
  (μ : Measure α) (ν : Measure α') (c : ℝ≥0∞) : Prop :=
  ∀ f : α → E, MemLp f p μ → AESTronglyMeasurable (T f) ν ∧ wnorm (T
    f) p' ν ≤ c * eLpNorm f p μ
```

Here, `eLpNorm` denotes the usual L^p norm and `MemLp f p` means that $f \in L^p(X)$. Note that this definition also explicitly encodes the constant appearing in the estimate. The class `ENorm` $\square$ simply postulates the existence of an extended norm (see Subsection 4.5).

```
class ENorm (E : Type*) where
  enorm : E → ℝ≥0∞
```

When an operator is bounded from strong L^p to strong L^q , we say it has strong type (p, q) .

Definition 2.3 (Strong type $\square$). *Let $p, q \in [0, \infty]$. We say that $T: (X \rightarrow E) \rightarrow (X' \rightarrow E')$ has strong type (p, q) if it is bounded as a map from $L^p(X)$ to $L^q(X')$. In other words, there exists a constant $c > 0$ such that for all $f \in L^p(X)$, we have $Tf \in L^q(X')$ with $\|Tf\|_q \leq c\|f\|_p$.*

Here is the definition of strong type in Lean.

```
def HasStrongType
  [ENorm E] [ENorm E'] [TopologicalSpace E] [TopologicalSpace E']
  {MeasurableSpace α} {MeasurableSpace α'} (T : (α → E) → (α' → E'))
  (p p' : ℝ≥0∞) (μ : Measure α) (ν : Measure α') (c : ℝ≥0∞) : Prop
  :=
  ∀ f : α → E, MemLp f p μ → AESTronglyMeasurable (T f) ν ∧ eLpNorm
    (T f) p' ν ≤ c * eLpNorm f p μ
```

The real interpolation theorem considers operators that are *subadditive*.

Definition 2.4 (Subadditive $\square$). *We say that $T: (X \rightarrow E) \rightarrow (X' \rightarrow E')$ is L -subadditive, if for μ -a.e. $x \in X$,*

$$\|(T(f_1 + f_2))(x)\|_e \leq L(\|(Tf_1)(x)\|_e + \|(Tf_2)(x)\|_e) \quad (1)$$

for all f_1 and f_2 . We say T is L -subadditive on $S \subseteq (X \rightarrow E)$ if (1) holds for all $f_1, f_2 \in S$.

```
def ASubadditiveOn [TopologicalSpace E] [ESeminormedAddMonoid E] [ENorm
  E']
  (T : (α → E) → α' → E') (P : (α → ε₁) → Prop) (A : ℝ ≥ 0 ∞) (ν :
    Measure α') : Prop :=
  ∀ (f g : α → E), P f → P g → ∀m x ∂ν,
    ||T (f + g) x||e ≤ A * (||T f x||e + ||T g x||e)
```

The notation $\forall^m x \partial\nu$ means a statement holds for ν -a.e. x .

The class `ESeminormedAddMonoid E` $\square$ demands that E is a monoid endowed with a topology, such that the extended norm is continuous and compatible with the addition. Both conditions are satisfied by normed groups. The full Lean definition is the following:

```
class ContinuousENorm (E : Type*) [TopologicalSpace E] extends ENorm E
  where
    continuous_enorm : Continuous enorm
```

```
class ESeminormedAddMonoid (E : Type*) [TopologicalSpace E]
  extends ContinuousENorm E, AddMonoid E where
  enorm_zero : ||(0 : E)||e = 0
  protected enorm_add_le : ∀ x y : E, ||x + y||e ≤ ||x||e + ||y||e
```

Many documented versions of the real interpolation theorem require sublinearity of the operator. As our eventual version does not require it, we omit the Lean definition.

Definition 2.5 (Sublinear $\square$). *We say that $T: (X \rightarrow E) \rightarrow (X' \rightarrow E')$ is L -sublinear if it is L -subadditive and for all $\lambda > 0$, and all f , and μ -a.e. $x \in X$,*

$$\|(T(\lambda f))(x)\|_e = \lambda \|(Tf)(x)\|_e. \quad (2)$$

For $S \subseteq (X \rightarrow E)$, we call T L -sublinear on S if it is L -subadditive on S and (2) holds for all $f \in S$.

The version of the real interpolation given by Folland is the following.

Theorem 2.6 (Real interpolation theorem, cf. [Fol99, Theorem 6.28]). *Let $(X, \mathcal{M}, \mu)$ and $(Y, \mathcal{N}, \nu)$ be measure spaces. Let $p_0, p_1, q_0, q_1 \in [1, \infty]$ such that*

$p_0 \leq q_0$, $p_1 \leq q_1$ and $q_0 \neq q_1$. Let $t \in (0, 1)$. Define $p \in [1, \infty]$ and $q \in [1, \infty]$ through the interpolations

$$\frac{1}{p} = \frac{t}{p_0} + \frac{1-t}{p_1} \quad \text{and} \quad \frac{1}{q} = \frac{t}{q_0} + \frac{1-t}{q_1}.$$

Let T be a 1-sublinear map from $L^{p_0}(X) + L^{p_1}(X)$ to the space of measurable functions on Y . Assume T is weak type (p_0, q_0) and weak type (p_1, q_1) . Then T is strong type (p, q) .

This is the theorem that we eventually proved.

Theorem 2.7 ($\square^{\mathfrak{A}}$). Let $(X, \mathcal{M}, \mu)$ and $(Y, \mathcal{N}, \nu)$ be measure spaces. Let E_1 be a topological space with the structure of an additive monoid with a continuous extended seminorm, and E_2 be a topological space with a continuous extended norm function. Let $p_0, p_1, q_0, q_1 \in (0, \infty]$ such that $p_0 \leq q_0$, $p_1 \leq q_1$ and $q_0 \neq q_1$. Let $t \in (0, 1)$. Define $p \in (0, \infty]$ and $q \in (0, \infty]$ through the interpolations

$$\frac{1}{p} = \frac{t}{p_0} + \frac{1-t}{p_1} \quad \text{and} \quad \frac{1}{q} = \frac{t}{q_0} + \frac{1-t}{q_1}.$$

Let T be an A -subadditive map from $L^{p_0}(X; E_1) + L^{p_1}(X; E_1)$ to $Y \rightarrow E_2$, that maps functions in $L^p(X; E_1)$ to almost-everywhere strongly measurable functions $Y \rightarrow E_2$. Assume T is weak type (p_0, q_0) with constant C_0 and weak type (p_1, q_1) with constant C_1 . Then T is strong type (p, q) with constant

$$2Aq^{q^{-1}} (|q - q_1|^{-1} + |q - q_0|^{-1})^{q^{-1}} C_0^{1-t} C_1^t.$$

With respect to the original theorem, we allowed for the exponents to be strictly smaller than 1, we allowed for target spaces to be additive commutative monoids with a norm that can take on the value ∞ , the operator T only needs to be subadditive, and we have an explicit expression for the constant.

Note that the assumption that T maps $L^p(X; E_1)$ to almost-everywhere strongly measurable functions is necessary, in that it does not follow automatically from T being both weak type (p_0, q_0) and (p_1, q_1) . To construct a counterexample, one can use that if an operator T is sublinear, any operator $\tilde{T}$ is as well, if $\tilde{T}$ satisfies that for all f and a.e. x , $\|\tilde{T}f(x)\|_e = \|Tf(x)\|_e$. One can then just define $\tilde{T}f$ to be a non-measurable function such that for all x , $\|\tilde{T}f(x)\|_e = \|Tf(x)\|_e$, for some $f \in L^p(X; E_1)$ that is not in $L^{p_0}(X; E_1)$ nor in $L^{p_1}(X; E_1)$.

In Lean, the statement of the real interpolation theorem that we proved reads as follows.

```

theorem exists_hasStrongType_real_interpolation {p₀ p₁ q₀ q₁ p q : ℝ ≥
  0 ∞}
  [TopologicalSpace E₁] [ESeminormedAddMonoid E₁]
  [TopologicalSpace E₂] [ContinuousENorm E₂]
  (hp₀ : p₀ ∈ Ioc 0 q₀) (hp₁ : p₁ ∈ Ioc 0 q₁) (hq₀q₁ : q₀ ≠ q₁)

```

```

{C0 C1 A : ℝ ≥ 0} (hA : 1 ≤ A) (ht : t ∈ Ioo 0 1) (hC0 : 0 < C0)
(hC1 : 0 < C1)
(hp : p-1 = (1 - t) / p0 + t / p1) (hq : q-1 = (1 - t) / q0 + t / q1)
(hmT : ∀ f, MemLp f p μ → AESTronglyMeasurable (T f) ν)
(hT : AESubadditiveOn T (fun f ↦ MemLp f p0 μ ∨ MemLp f p1 μ) A ν)
(h0T : HasWeakType T p0 q0 μ ν C0) (h1T : HasWeakType T p1 q1 μ ν
C1) :
HasStrongType T p q μ ν (C_realInterpolation p0 p1 q0 q1 q C0 C1 A
t) := sorry

```

In the formalization, we parametrize the exponents p and q such that $p^{-1} = (1-t)p_0^{-1} + tp_1^{-1}$ and $q^{-1} = (1-t)q_0^{-1} + tq_1^{-1}$. This allows us to give an explicit description of the constant in the theorem's conclusion (see Subsection 4.1).

In the Carleson project, we also formalized two key applications of this theorem, to the Hardy–Littlewood maximal function. This demonstrates the usability of our formalization.

Definition 2.8 ($\square$). *Suppose $f : \mathbb{R}^d \rightarrow \mathbb{C}$ is measurable. The (centered) Hardy–Littlewood maximal function $Mf : \mathbb{R}^d \rightarrow \mathbb{R}$ is defined as*

$$Mf(x) := \sup_{r>0} \frac{1}{\mu(B(r,x))} \int_{B(r,x)} |f(y)| dy.$$

Proposition 2.9 ($\square$, $\square$). *For all $p \in [1, \infty)$, the Hardy–Littlewood maximal operator $M : (\mathbb{R}^d \rightarrow \mathbb{C}) \rightarrow (\mathbb{R}^d \rightarrow \mathbb{R})$ has weak type (p, p) . If $p \neq 1$, it also has strong type (p, p) .*

Corollary 2.10 ($\square$). *If $f \in L^p(X)$ for $p \geq 1$, then Mf is finite almost everywhere.*

3 Proof strategy

3.1 Close-reading of some proofs

When we started working on formalization of the real interpolation theorem in the context of the Carleson project, it was already suggested in the source code to follow Folland's proof. Certainly, there are many other proofs available in the literature. Even though we had consulted several sources, and even though the theorem as stated by Folland was practically sufficient for the purposes of the Carleson project, in retrospect we should have done a more careful planning and study of the various proofs around. In our work, we were having a bit too much inertia and tunnel vision around Folland's approach. We spent time generalizing the proof in various directions, but only fully realized the benefit of a further generalization to Lorentz spaces when the formalization was practically finished and could no longer be easily adapted to include that generality.

In this section we highlight some of the advantages, disadvantages, alternative approaches and possible generalizations of Folland's proof, as a precursor to

the parts of the article that are dedicated to explaining how we have approached these generalizations, sometimes purely mathematically and sometimes with extra support from Lean.

An advantage of the proof by Folland was that it works in the general, i.e. non-diagonal, case in which the exponents in the domain are allowed to differ from each other, and the exponents in the target are allowed to be different as well. Another advantage of the proof is that it is quite direct. It does, for instance, not require approximations with simple functions. Instead it mostly relies on integral manipulations and estimates.

A disadvantage of the theorem statement given by Folland is that it does not give an explicit expression for the constant in the conclusion. Folland’s proof distinguishes various cases. For some of these, no constant is derived, while for other cases, a constant is derived, but it is arguably not the *right* constant. In the easier case where $p_0 = q_0$ and $p_1 = q_1$, many proofs give a constant

$$2 \left(\frac{q(q_1 - q_0)}{(q - q_0)(q_1 - q)} \right)^{1/q} C_0^{1-t} C_1^t \quad (3)$$

which shows a nice interpolation between the constants C_0 and C_1 (although note that the prefactor is still dependent on t , and blows up as q approaches q_0 or q_1 , as expected). This form is not present in the constant in Folland’s proof.

Another disadvantage of Folland’s proof and theorem statement is that it does not hold in full generality on the exponents: it does not work for exponents strictly less than 1. Moreover, it requires sublinearity of the operator, and requires access to a scalar multiplication in domain and target space.

Many other proofs that we found in literature generalized the statement in some respect, but not in others. For instance, a proof sketched by Tao [Tao09] uses a discrete decomposition of input functions. The proof allows for exponents less than one, but uses scalar multiplication and assumes sublinearity. The proof in Grafakos’ textbook [Gra14a, Theorem 1.3.2] is very similar to Folland’s proof and derives an explicit constant, but deals with the diagonal case and needs sublinear operators.

The proof presented by Tao requires approximation arguments to derive the final result. At least initially it seemed Folland’s proof would be easier to formalize. Eventually, we could adapt Folland’s proof to allow for the generality described above.

Nevertheless, arguably the right level of generality is given by the interpolation theorem for Lorentz spaces, of which the real interpolation theorem is a special case. A book by Stein and Weiss [SW71, Theorem V.2.4] states the non-diagonal case of the real interpolation theorem and only requires subadditive operators. It defers its proof to that of the Lorentz-space version of the interpolation theorem and does not explicitly state a constant in the theorem statement. We did not manage to easily adapt our formalization to this more general case, and will go further into this aspect in the discussion.

3.2 Our proof strategy

Let us sketch our chosen proof of the real interpolation theorem. We take an a.e. strongly measurable function f such that the strong L^p -norm of f is bounded. We would then like to estimate

$$\|Tf\|_q \leq C\|f\|_p.$$

For this we use the layer-cake formula

$$\|Tf\|_q^q = q \int_0^\infty t^{q-1} \mu(\{\|Tf\|_e > t\}) dt,$$

which gives us the task to find, for every $t > 0$, a good estimate of the measure $\mu(\{\|Tf\|_e > t\})$.

The key idea is to split f in a part where it is large, and a part where it is small. In case $p_0 < p_1$, for the part where f is small, the weak-type estimate $[Tg]_{q_1} \leq C_1\|g\|_{p_1}$ is relevant, whereas for the part where f is large, we can rely on the weak-type estimate $[Tg]_{q_0} \leq C_0\|g\|_{p_0}$.

Importantly, for every different t , we can make a different split. We set

$$f(t) = f_{>A(t)}(t) + f_{\leq A(t)}(t)$$

where the functions on the right-hand side are truncations, with A denoting the truncation level, but for the moment, we postpone the exact definitions of the functions on the right-hand side. By subadditivity,

$$\mu(\{\|Tf\|_e > 2t\}) \leq \mu(\{\|Tf_{>A(t)}\|_e > t\}) + \mu(\{\|Tf_{\leq A(t)}\|_e > t\}). \quad (4)$$

The weak-type estimates now help in bounding the terms on the right-hand side in (4). Indeed, from $[Tf]_{q_0} \leq C_0\|f\|_{p_0}$ we learn that

$$(t^{q_0} \mu(\{\|Tf_{>A(t)}\|_e > t\}))^{1/q_0} \leq C_0\|f_{>A(t)}\|_{p_0}$$

from which we derive

$$\mu(\{\|Tf_{>A(t)}\|_e > t\}) \leq (C_0\|f_{>A(t)}\|_{p_0}/t)^{q_0}.$$

Putting everything together, we obtain

$$\begin{aligned} \|Tf\|_q &= q \int_0^\infty t^{q-1} \mu(\{\|Tf\|_e > t\}) dt \\ &= 2^q q \int_0^\infty t^{q-1} \mu(\{\|Tf\|_e > 2t\}) dt \\ &\leq 2^q q \int_0^\infty t^{q-1} (\mu(\{\|Tf_{>A(t)}\|_e > t\}) + \mu(\{\|Tf_{\leq A(t)}\|_e > t\})) dt \\ &\leq 2^q q \int_0^\infty t^{q-1} \left(\left(\frac{C_0\|f_{>A(t)}\|_{p_0}}{t} \right)^{q_0} + \left(\frac{C_1\|f_{\leq A(t)}\|_{p_1}}{t} \right)^{q_1} \right) dt. \end{aligned} \quad (5)$$

From here, one can change the order of integration and, by an appropriate choice of the function A , derive an expression on the right-hand side that involves the p -norm of f . Let us illustrate this in the somewhat simpler, diagonal, case when $p_0 = q_0$ and $p_1 = q_1$, when the change in order of integration is just given by Tonelli's theorem, and A can be chosen as the identity function, and with $f_{>t}(x) := f(x)\mathbf{1}_{\|f(x)\|_e > t}$. Concentrating on one of the terms, we obtain

$$\begin{aligned}
& \int_0^\infty C_0^{q_0} t^{q-q_0-1} \int_X \|f_{>t}\|_e^{p_0} dt \\
&= C_0^{q_0} \int_X \int_0^\infty t^{q-q_0-1} \|f(x)\|_e^{p_0} \mathbf{1}_{\|f(x)\|_e > t} dt dx \\
&= C_0^{q_0} \frac{1}{q-q_0} \int_X \|f(x)\|_e^{p_0+q-q_0} dx \\
&= C_0^{q_0} \frac{1}{q-q_0} \|f\|_p^p.
\end{aligned} \tag{6}$$

The other term can be estimated in an analogous way, and this way, we can bound the strong q -norm of Tf in terms of the strong p -norm of f .

In the general, non-diagonal, case, Minkowski's integral inequality implies that one can still change the order of integration, at the cost that the change does not lead to an equality but an inequality, which is not a problem because the inequality is in the right direction.

3.3 Choice of truncations and allowing for exponents less than 1

Unlike what we did in the end of the proof above, the proof that Folland gives first rewrites the norms on the right-hand side in terms of an integral over the distribution function using the layer-cake formula. This is where Folland really uses that the involved exponents are larger than or equal to 1.

The truncations that Folland uses are somewhat more complicated than truncations that appear in other parts of the literature, and in particular, working with these other truncations would save quite some work in the formalization. Folland uses

$$f_{\leq M}(x) := \begin{cases} f(x) & \text{if } \|f(x)\|_e \leq M, \\ \frac{f(x)}{\|f(x)\|_e} & \text{if } \|f(x)\|_e > M \end{cases}$$

whereas it would be easier to work with

$$f_{\leq M}(x) := \begin{cases} f(x) & \text{if } \|f(x)\|_e \leq M, \\ 0 & \text{if } \|f(x)\|_e > M. \end{cases}$$

To make the second type of truncation work, we deviated from rewriting the norms on the right-hand side in terms of a distribution function, but directly

wrote out the norms in the last expression in (5) as illustrated in (6). This also helped us to allow for exponents smaller than 1. Note also that Folland’s truncations need a possibility for normalization, or in other words, scaling in the target space needs to be possible. By using the other type of truncation, we could remove another dependency on this assumption.

This did give us the extra complication that Minkowski’s integral theorem requires the involved measure spaces to be σ -finite, which is the case for $\mathbb{R}$, but not in general for the measure space $(X, \mathcal{M}, \mu)$. We fixed this by restricting to the support of f , whose σ -finiteness can be deduced from the assumption that $f \in L^p$.⁷

After the formalization had been completed, we also found an alternative argument, in which the norms in the last expression in (5) are indeed related back to distribution functions, and one uses a change-of-variables argument in the term that otherwise would have been difficult to estimate. This approach would probably simplify the argument a bit more (if it is easy to perform the change of variables in Lean).

4 Key points in the formalization

In this section we will mostly discuss the challenges encountered in the formalization. In some cases, resolving some of these challenges helped us generalize Folland’s proof.

4.1 Cases and extracting the constant

The proof presented by Folland practically combines many different cases, of which only one is worked out in detail. These cases arise from the choices $q_0 < q_1$ or $q_0 > q_1$, whether the highest of these exponents, q_j say, is ∞ or not, whether p_j is also equal to ∞ or not.

For many of these cases, Folland uses a *different* choice for the truncation function A . In our formalization, we have for this reason built a relatively general framework that allows for different choices of the function A , that we could possibly tune to improve the final constant, and that we could adapt to every case. While aiming to improve the constant, however, we found that it was possible have one single expression for the function A , that worked in all different cases. To do this, we firmly relied on properties of the extended non-negative real numbers in Lean.

The function A [↗] is given by

$$A(t) = \left(\frac{t}{d}\right)^\sigma$$

⁷Generalizing to `ESeminormedAddMonoids` required a further minor tweak, as we no longer assume the `enorm` on the codomain of f to be positive definite. Restricting instead to the support of $\|f\|_e$ enabled the proofs to generalize.

in which the exponent is given by

$$\sigma := \frac{p^{-1}(q_1^{-1} - q_0^{-1})}{q^{-1}(p_1^{-1} - p_0^{-1})}$$

and the denominator equals

$$d := \frac{C_0^{\frac{q_1^{-1}}{-1-q_0^{-1}}} \|f\|_p^{\frac{p_0^{-1}q_1^{-1}}{q_1^{-1}-q_0^{-1}}}}{C_1^{\frac{q_0^{-1}}{-1-q_0^{-1}}} \|f\|_p^{\frac{p_1^{-1}q_0^{-1}}{q_1^{-1}-q_0^{-1}}}} \quad (7)$$

with the convention that $\infty^{-1} = 0$. This general choice that worked across all of the cases was also useful because it lead to a more natural, and better, constant in the real interpolation theorem.

Eventually, we obtain a constant

$$2Aq^{q^{-1}} (|q - q_1|^{-1} + |q - q_0|^{-1})^{q^{-1}} C_0^{1-t} C_1^t,$$

where $|q - q_1|^{-1}$ or $|q - q_0|^{-1}$ are zero when q_1 or q_0 are infinite, respectively. (Recall that the constant A occurs in the theorem's hypothesis: the operator T is assumed to be A -subadditive.) This expression for the constant reduces to the same expression as given in (3) in the diagonal case.

Note that this particular argument does not work if $\|f\|_p = 0$, and this is a case that we had to handle separately (see Subsection 4.7).

The work felt a bit different from working on paper, because with the help of the interactive theorem prover, it is much easier to deal with computation rules for infinity and zero. In particular, $1/\infty = 0$, but even $0/0 = 0$ in $[0, \infty)$.

4.2 Removing the need for scaling

The expression for the constant d in (7) seems more complicated than necessary, especially considering that in many formulations of the real interpolation theorem the operator is assumed to be sublinear, which would allow one to assume without loss of generality that $\|f\|_p = 1$. At first, this was not an intentional choice. After switching the type of truncations used, we discovered that leaving this factor in allows further generalization of our statements and decided to keep it. This allowed us to only require that the operator is subadditive, remove the scaling structure in domain and target space, and the additive structure in the target space.

4.3 Coercions

Just as in the Carleson project on a whole, we were having difficulties in design decisions on which types to use for real numbers. Indeed, three types of real numbers played a role: $\mathbb{R}$, $\mathbb{R}_{\geq 0}$ and $\overline{\mathbb{R}_{\geq 0}}$.

Although in set-based mathematics there are obvious inclusions of sets, a term x of type $\mathbb{R}_{\geq 0}$ is not of type $\mathbb{R}$ nor of type $\overline{\mathbb{R}_{\geq 0}}$. One needs explicit conversions, casts, or coercions to go from one to the other. For the direction of the set-theoretical inclusions, they are often inserted automatically, and automation at times can help extract useful results, but conversions in the other directions are also necessary and often more cumbersome to work with.

One place where such a conversion is necessary was in our computations around exponents of L^p spaces. Because these exponents are positive and possibly infinite, encoding them as elements of $\overline{\mathbb{R}_{\geq 0}}$ seems very natural. But computing with them can be difficult, especially when subtraction is involved. Moreover, as a^p is not defined for p in $\mathbb{R}_{\geq 0}$, one would need an explicit conversion first to use such expressions.

All in all, this added layer of explicit conversions made many computations that look very simple on paper incredibly tedious in Lean, requiring long sequences of manual rewrites, often complicated by the need for resolving side conditions.

4.4 Measurability issues

In a textbook proof of the real interpolation theorem, measurability of the given functions would rarely be discussed. Colloquially, every function that you can think of is measurable, but this is not an argument that Lean will support. The way of arguing with measurability within Lean is also different from how one would often go about it in mathematics. In Lean, most integrals we dealt with were Lebesgue lower integrals, with the advantage that one can easily integrate *any* functions, not merely measurable ones. This makes integration a total function and allows one to work quite long without needing measurability assumptions. Nonetheless, one really is just postponing the difficulty. To prove statements such as that the integral of the sum of two functions is the sum of the integrals, one really needs the measurability of these functions.

For cleanliness, in the next to last line in (5), it would make sense to split the integral over the sum of two distribution functions into the sum of integrals of separate distribution functions. However, at that stage it was unclear to us why the individual functions were measurable. [TODO: doublecheck]

More generally, in working on the formalization quite some effort was put in settling measurability issues, whereas we would likely have glossed over them when giving a textbook-style proof.

4.5 Generalization to enormed spaces

Many results in the Carleson project were generalized to a form in which they would include enormed spaces, which are basically spaces in which the norm could possibly also take on the value ∞ .

The main motivation for doing so was to eventually ease the formalization, for instance around the use of the Hardy–Littlewood maximal function. In the Carleson project, the real interpolation is applied to get strong-type estimates

on this function. As a supremum over a countable set, it naturally takes values in $\overline{\mathbb{R}_{\geq 0}}$. But at the outset of the formalization, one could not even formulate strong-type estimates for such a function: Membership of L^p required functions to take values in normed additive groups. One could decide to redefine the function where it takes value ∞ , so that one would end up with a function in $\mathbb{R}$, but this would be more cumbersome for the formalization. The other approach, which is more natural, is to allow for norms taking possibly infinite values, and generalizing definitions of membership of L^p and definition of weak and strong-type estimates accordingly.

For this, we introduced the notation class `ENorm`, which just describes a type endowed with a function `enorm`: $X \rightarrow \overline{\mathbb{R}_{\geq 0}}$. Here, `enorm` is merely a name, and natural properties one might want from a norm, such as triangle inequality, are not encoded into `ENorm`. Such properties are provided by further type classes. The real interpolation theorem was eventually formulated for `ESeminormedAddMonoids`. These are additive monoids with a continuous positive semi-definite enorm that satisfies the triangle inequality. In fact, the target space only needs to be endowed with a continuous enorm (without any further properties). Note how the spaces themselves reflect that scalar multiplication or sublinearity are not necessary for the real interpolation theorem to hold.

The refactor to allow for enormed spaces in the real interpolation theorem required a considerable amount of work. To begin, all relevant definitions and basic results in `Mathlib` had to be generalised: overall, this touched about 2500 lines of `Mathlib` code. For the interpolation theorem itself, the generalisation modified about 2000 lines of code (out of 5000).

4.6 Improved tactic support

A downside of working with $\overline{\mathbb{R}_{\geq 0}}$ is that tactic support is less strong. Indeed, for real numbers, the `linarith` and `field_simp` tactics work relatively well, but they do not support terms in type $\overline{\mathbb{R}_{\geq 0}}$. Support of the `linarith` for terms in $\overline{\mathbb{R}_{\geq 0}}$ was added only after the project was completed⁸. The ring tactic works for $\overline{\mathbb{R}_{\geq 0}}$, but not as well.

Many rewriting steps working with $\overline{\mathbb{R}_{\geq 0}}$ required that some side conditions would be satisfied. It is convenient if those side conditions can be discharged automatically. For instance, certain algebraic rules for terms in $\overline{\mathbb{R}_{\geq 0}}$ require the terms to be finite. The `finiteness` tactic can sometimes automatically prove these. Similarly, to prove that some terms are positive, the `positivity` tactic helps. Often however, we needed a combination of those, as for the finiteness of one expression one would need the positivity of another.

Based on our experiences with the real interpolation theorem, we did make the `finiteness` tactic more powerful. Other places in the Carleson project also provided good example cases for where the `finiteness` tactic really should have worked, and led to further improvements of the tactic.

⁸<https://github.com/leanprover-community/mathlib4/pull/35155>

4.7 Handling a.e. vanishing functions

Formalization often brings special cases to attention that would be glossed over on paper. One such instance is handling almost every vanishing functions. If $f \in L^p(X)$ has vanishing L^p norm, the existing argument bounding $\|Tf\|_{L^q}$ in terms of $\|f\|_{L^p}$ breaks down: the truncations it uses require dividing by $\|f\|_{L^p}$. Fortunately, there is an independent argument for this case: if $\|f\|_{L^p} = 0$, then f is zero almost everywhere, in particular also $f \in L^{p_0}(X)$. As T has weak type (p_0, q_0) by hypothesis, in particular Tf has vanishing weak L^{q_0} -norm. This implies $Tf = 0$ a.e., hence $\|Tf\|_{L^q} = 0$ as desired.

5 Discussion and outlook

Arguably the right level of generalization would have been to formalize the interpolation theorem for Lorentz spaces [Hun64]. In fact, the original proof by Hunt of this fact is quite accessible and has many similarities with the proof found presented by Folland. However, Hunt uses an argument based on monotone rearrangements of functions, whereas Folland’s proof uses distribution functions. We tried to convert the proof to the one on distribution functions, but did not manage to find a simple conversion. For this reason, we did not see an easy and small extension of the current formalization to include the case for Lorentz spaces.

A wholly different approach would be given by resorting to general interpolation theory [BL12]. Its general setup could be very appealing for a library like `Mathlib`, since interpolation theorems for other theorems could be obtained in other settings and for other spaces as well. Yet on the face of it, this theory will only say something about operators that are *linear* and not *sublinear*. In certain cases, there are some ways around this [Mal89]. It seems to require a scaling operation on the target space, but that condition is almost vacuous because one could replace without loss of generality the target space $Y \rightarrow E_2$ by functions $Y \rightarrow \mathbb{R}_{\geq 0}$ by postcomposing with the norm. One would need to check carefully if in this particular case, the interpolated norms still correspond to the Lorentz space norms in the most general settings.

Another approach to a proof of the real interpolation theorem for Lorentz spaces would be to follow a more discrete approach. This is for instance illustrated in Tao’s blog [Tao09, Exercise 33], and a related approach is given in Grafakos’ proof [Gra14a, Theorem 1.4.19]. Tao mentions the condition that the secondary exponent of the Lorentz space needs to be larger than or equal to 1, while it is also implicit in the proof given by Grafakos. The version of the theorem as presented by Hunt does not have this condition, but the proof considers this case for simplicity.

In future work, we would therefore like to formalize the real interpolation theorem for Lorentz spaces following abstract interpolation theory. We do have the intention to keep the generality that we have achieved so far, avoiding a scaling operation, and allowing for all exponents, even the secondary ones, to

be less than one. A further generalization is towards a multi(sub)linear version of the real interpolation theorem, such as in [Gra14b, Theorem 7.2.2].

In a different direction, it would be worthwhile to formalize more applications of the real interpolation theorem, such as boundedness of the Hilbert transform.

6 Conclusion

We formalized the real interpolation theorem, starting from a proof presented by Folland. The formalization reflects a few generalizations, such as a widening of a range of possible exponents, and allowing for function spaces without a scaling operation. Individually, these generalizations were present in the literature, but it was difficult to find a formulation that combines all of them. In retrospect, we should have paid even more attention to planning and designing which version to formalize. Most likely, the best way forward is to formalize the interpolation theorem for Lorentz spaces using abstract interpolation theory.

References

- [AS24a] Reynald Affeldt and Zachary Stone. “A Comprehensive Overview of the Lebesgue Differentiation Theorem in Coq”. In: *15th International Conference on Interactive Theorem Proving (ITP 2024)*. Ed. by Yves Bertot, Temur Kutsia, and Michael Norrish. Vol. 309. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024, 5:1–5:19. ISBN: 978-3-95977-337-9. DOI: 10.4230/LIPIcs.ITP.2024.5. URL: <https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ITP.2024.5>.
- [AS24b] Reynald Affeldt and Zachary Stone. “Towards the Fundamental Theorem of Calculus for the Lebesgue Integral in Coq”. In: *35es Journées Francophones des Langages Applicatifs (JFLA 2024)*. Saint-Jacut-de-la-Mer, France, Jan. 2024. URL: <https://inria.hal.science/hal-04406350>.
- [Bec+24] Lars Becker, Floris van Doorn, Asgar Jamneshan, Rajula Srivastava, and Christoph Thiele. “Carleson operators on doubling metric measure spaces”. arXiv:2508.05563 [math.CA]. 2024.
- [Bec+26] Lars Becker et al. “Formalizing Carleson’s Theorem in Lean”. Manuscript, submitted. 2026. URL: <https://https://florisvandoorn.com/carleson/paper.pdf>.
- [BL12] Jöran Bergh and Jörgen Löfström. *Interpolation spaces: an introduction*. Vol. 223. Springer Science & Business Media, 2012.
- [Dol25] Moritz Doll. “Formalizing Schwartz functions and tempered distributions”. arXiv:2510.24060 [cs.LO]. Oct. 2025. URL: <https://arxiv.org/abs/2510.24060>.

- [Fol99] Gerald B Folland. *Real analysis: modern techniques and their applications*. John Wiley & Sons, 1999.
- [Gra14a] Loukas Grafakos. *Classical Fourier Analysis*. Graduate Texts in Mathematics. Springer New York, 2014. DOI: 10.1007/978-1-4939-1194-3.
- [Gra14b] Loukas Grafakos. *Modern Fourier Analysis*. Graduate Texts in Mathematics. Springer New York, 2014. DOI: 10.1007/978-1-4939-1230-8.
- [Hun64] Richard A Hunt. “An extension of the Marcinkiewicz interpolation theorem to Lorentz spaces”. In: *Bulletin of the American Mathematical Society* 70.6 (1964), pp. 803–807.
- [HVW16] Tuomas Hytönen, Jan Van Neerven, Mark Veraar, and Lutz Weis. *Analysis in Banach spaces*. Vol. 12. Springer, 2016.
- [Mal89] Lech Maligranda. “On interpolation of nonlinear operators”. In: *Commentationes Mathematicae* 28.2 (1989).
- [mat20] The mathlib Community. “The Lean Mathematical Library”. In: *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs*. CPP 2020. New York, NY, USA: Association for Computing Machinery, 2020, pp. 367–381. ISBN: 9781450370974. DOI: 10.1145/3372885.3373824.
- [MU21] Leonardo de Moura and Sebastian Ullrich. “The Lean 4 Theorem Prover and Programming Language”. In: *Automated Deduction – CADE 28*. Ed. by André Platzer and Geoff Sutcliffe. Cham: Springer International Publishing, 2021, pp. 625–635. ISBN: 978-3-030-79876-5.
- [Suz05] Yasumasa Suzuki. “Hölder’s Inequality and Minkowski’s Inequality”. In: *Formalized Mathematics* 13.1 (2005), pp. 59–61. URL: https://fm.mizar.org/2005-13/pdf13-1/holder_1.pdf.
- [SW71] Elias M Stein and Guido Weiss. *Introduction to Fourier analysis on Euclidean spaces*. Vol. 1. Princeton university press, 1971.
- [Tao09] Terence Tao. *Interpolation of L_p spaces*. 2009. URL: <https://terrytao.wordpress.com/2009/03/30/245c-notes-1-interpolation-of-lp-spaces/>.
- [TS06] Krzysztof Treyderowski and Christoph Schwarzweller. “Multiplication of Polynomials using Discrete Fourier Transformation”. In: *Formalized Mathematics* 14.4 (Dec. 2006), pp. 121–128. URL: <https://mizar.uwb.edu.pl/fm/2006-14/pdf14-4/polyonom8.pdf>.