

UniMath

Benedikt Ahrens and Catherine Lelay

Outline

- 1 What is UniMath?
- 2 How to use UniMath?
- 3 Contents of the packages
 - Foundations
 - Combinatorics
 - Category Theory
 - Algebra
 - Real Numbers
 - Topology
 - More specialized packages

Outline

- 1 What is UniMath?
- 2 How to use UniMath?
- 3 Contents of the packages
 - Foundations
 - Combinatorics
 - Category Theory
 - Algebra
 - Real Numbers
 - Topology
 - More specialized packages

UniMath

- Short for “Univalent Mathematics”
- Language for mathematics
- Library of mathematics formalized and computer-checked in that language
- Team of people contributing to the UniMath library

Goals of the UniMath project

Language: core type theory

- small enough for allowing semantic justification
- rich enough for formalizing mathematics

Library: towards formalization of research-level mathematics

- accommodate any field of mathematics
- accessible to and usable by mathematicians

The UniMath language in theory

Type former	Notation	(special case)
Sigma type	$\sum_{x:A} B(x)$	$A \times B$
Product type	$\prod_{x:A} B(x)$	$A \rightarrow B$
Coproduct type	$A + B$	
Identity type	$a =_A b$	
Universes	$U_0 : U_1$	
Nat, Bool, 1, 0		

- Definitional η -rules for \sum and \prod
- Axioms: function extensionality, univalence
- Resizing: any proposition lives in U_0

The UniMath language in practice

A subset of the Coq language:

- no general records
- no general inductive types
- no `match` construct

Coq features used to simulate the UniMath language:

- Resizing rule enabled by `-type-in-type` flag
- Avoid Coq's Prop for identity type by `-indices-matter` flag
- η for sums through primitive projections

Outside the kernel

Coq features used in UniMath:

- Coercions
- Implicit arguments
- (Unicode) notations
- (some) Tactics

Coq features not used in UniMath:

- Canonical structures
- Type classes
- Proof automation

The UniMath library

In theory Library of mathematics written in the UniMath language

- In practice**
- Collection of Coq files using the outlined subset of the Coq language
 - Infrastructure for convenient usage

Born by combining three libraries:

- Foundations (Voevodsky)
- RezkCompletion (Ahrens, Kapulkin, Shulman)
- Ktheory (Grayson)

Available on <https://github.com/UniMath/UniMath>

The UniMath library

Organized in ‘packages’:

- Foundations
- More Foundations
- Combinatorics
- Algebra
- Number Systems
- Real Numbers
- Category Theory
- Homological Algebra
- K-theory
- Topology
- ...

Some information on the UniMath library

- ca. 107,000 loc
- Compile time: too long
- 13 contributors
- Distributed under free software license

Code based on UniMath/UniMath: UniMath/TypeTheory

Repository [UniMath/TypeTheory](#)

- Adheres to the same guiding principles and language restrictions
- Hosts material on categorical structures for type theory
- Works around sometimes lengthy review process for UniMath/UniMath
- About 16,000 loc

Outline

- 1 What is UniMath?
- 2 **How to use UniMath?**
- 3 Contents of the packages
 - Foundations
 - Combinatorics
 - Category Theory
 - Algebra
 - Real Numbers
 - Topology
 - More specialized packages

Compilation

- Coq as a submodule: first compilation will download and compile a “good” version of Coq
- Package-wise compilation possible, e.g.,
\$ make CategoryTheory

Detailed installation instructions [here](#).

Editing the files

Workflow optimized for usage with Emacs + ProofGeneral:

- PG is pointed to the version of Coq compiled for UniMath
- Agda unicode input mode loaded automatically upon visiting a UniMath file
- Emacs TAGS mechanism available by
`$ make TAGS`

CoqIDE can be built using

```
$ make BUILD_COQIDE=yes
```

Browsing the files

Two versions of HTML documentation

- standard coqdoc HTML documentation
\$ make html
- HTML documentation with possibility to hide proofs
\$ make doc

How to get help

- Documentation:

- [README](#)
- [INSTALL](#)
- [Wiki](#)

- Mailing list:

univalent-mathematics@googlegroups.com

Archived on

[https://groups.google.com/forum/#!forum/
univalent-mathematics](https://groups.google.com/forum/#!forum/univalent-mathematics)

How to help

- Package PAdics needs update for compilation with latest Coq and UniMath
- Documentation, installation instructions
- contribute improvements, new code:
Issues on Github with a 'code change' tag

Outline

- 1 What is UniMath?
- 2 How to use UniMath?
- 3 Contents of the packages
 - Foundations
 - Combinatorics
 - Category Theory
 - Algebra
 - Real Numbers
 - Topology
 - More specialized packages

Outline

- 1 What is UniMath?
- 2 How to use UniMath?
- 3 Contents of the packages**
 - Foundations**
 - Combinatorics
 - Category Theory
 - Algebra
 - Real Numbers
 - Topology
 - More specialized packages

Package Foundations

Provides a specification of “univalent foundations”:

- Type and term constructors
- Definition of ‘weak equivalence’
- Definition of ‘h-level’ for types and functions
- Facts on propositions and sets
- Univalence axiom and consequences
- Arithmetic on natural numbers
- ...

Package MoreFoundations

Several variants of the same construction, e.g.,

$$(a, b) = (a', b') \simeq \sum_{p:a=a'} p_*(b) = b'$$
$$t = t' \simeq \sum_{p:t.1=t'.1} p_*(t.2) = t'.2$$

useful, but not interesting—dilutes Foundations.

Package MoreFoundations mirrors structure of Foundations:

- one variant of each construction in Foundations
- others in MoreFoundations

↪ Foundations as a textbook on univalent foundations

Outline

- 1 What is UniMath?
- 2 How to use UniMath?
- 3 Contents of the packages**
 - Foundations
 - Combinatorics**
 - Category Theory
 - Algebra
 - Real Numbers
 - Topology
 - More specialized packages

Package Combinatorics

- Standard finite sets
- Finite sets
- Finite sequences
- Ordered sets
- Wellordered sets
- Zermelo's wellordering theorem:
AC \Rightarrow every set can be well-ordered

Outline

- 1 What is UniMath?
- 2 How to use UniMath?
- 3 Contents of the packages**
 - Foundations
 - Combinatorics
 - Category Theory**
 - Algebra
 - Real Numbers
 - Topology
 - More specialized packages

Package CategoryTheory

- (Univalent) categories, functors, natural transformations
- Category of sets and Yoneda lemma
- Adjunctions, equivalences
- Rezk completion
- (Co)Limits in general, direct definition of some special (co)limits
- Abelian categories
- ...

Outline

- 1 What is UniMath?
- 2 How to use UniMath?
- 3 Contents of the packages**
 - Foundations
 - Combinatorics
 - Category Theory
 - Algebra**
 - Real Numbers
 - Topology
 - More specialized packages

Algebraic structures

Defined structures:

- monoid, abmonoid, gr, abgr
- rig, commrig, rng, commrng
- DivRig, CommDivRig, fld
- ConstructiveDivisionRig,
ConstructiveCommutativeDivisionRig,
ConstructiveField

Useful tools:

- abgrdiff : abmonoid \rightarrow abgr
 $X \times X / \approx$ with $(x,y) \approx (x',y') \Leftrightarrow \exists c \in Y, x+y'+c = x'+y+c$
Application: rigtorng, commrightocommrng, hz
- abmonoidfrac : $\forall X : \text{abmonoid}, \text{submonoid} \rightarrow$
abmonoid
 $X \times Y / \approx$ with $(x,y) \approx (x',y') \Leftrightarrow \exists c \in Y, x \cdot y' \cdot c = x' \cdot y \cdot c$
Application: commrngfrac, fldfrac, hq

Miscellaneous

Defined:

- Partial order, strong order, ...
- Lattice
- Truncated subtraction in monoids with a lattice

$$\forall x,y \quad \text{truncminus}(x,y) + y = \max(x,y)$$

- Archimedean property for monoids, groups, ring, field, ...

Proved: conservation of some properties by `abgrdiff`,
`abmonoidfrac`, ...

Outline

- 1 What is UniMath?
- 2 How to use UniMath?
- 3 Contents of the packages**
 - Foundations
 - Combinatorics
 - Category Theory
 - Algebra
 - Real Numbers**
 - Topology
 - More specialized packages

Non-negative Real Numbers

Dcuts := a subset L of $\mathbb{Q}_{\geq 0}$ such that:

bottom $\forall x \in L, \forall x', x' \leq x \Rightarrow x' \in L$

open $\forall x \in L, \exists x' \in L, x < x'$

correction $\forall c > 0, c \notin L \vee (\exists x \in L, (x + c) \notin L)$

Non-negative Real Numbers

Dcuts := a subset L of $\mathbb{Q}_{\geq 0}$ such that:

bottom $\forall x \in L, \forall x', x' \leq x \Rightarrow x' \in L$

open $\forall x \in L, \exists x' \in L, x < x'$

correction $\forall c > 0, c \notin L \vee (\exists x \in L, (x + c) \notin L)$

Defined:

- Strict and large order
- Apartness relation
- Injection from non-negative rationals to Dcuts
- Addition, multiplication, maximum, minimum, truncated minus, multiplicative inverse

Non-negative Real Numbers

Dcuts := a subset L of $\mathbb{Q}_{\geq 0}$ such that:

bottom $\forall x \in L, \forall x', x' \leq x \Rightarrow x' \in L$

open $\forall x \in L, \exists x' \in L, x < x'$

correction $\forall c > 0, c \notin L \vee (\exists x \in L, (x + c) \notin L)$

Defined:

- Strict and large order
- Apartness relation
- Injection from non-negative rationals to Dcuts
- Addition, multiplication, maximum, minimum, truncated minus, multiplicative inverse

Proved: Dcuts is

- a lattice
- an ordered division rig
- Archimedean
- complete (Cauchy, Dedekind)

Real numbers (1)

Reals := commrigtocommrng Dcuts
:= Dcuts \times Dcuts / \approx

with $(x^P, x^N) \approx (y^P, y^N) := \exists c, x^P + y^N + c = y^P + x^N + c.$

Inherited:

- Ring properties
- Orders and Apartness
- Archimedean property
- Lattice structure

Real numbers (2)

Defined:

$$\begin{aligned} \text{RtoNRNR} : \text{Reals} &\rightarrow \text{Dcuts} \times \text{Dcuts} \\ x &\mapsto (x^P - x^N, x^N - x^P) \text{ for any } (x^P, x^N) \text{ in } x \end{aligned}$$

Proved:

- $\forall x \neq 0$,
$$\begin{cases} x^{-1} := \text{class of } ((\text{RtoNRNR}(x))_1^{-1}, 0) & \text{if } (\text{RtoNRNR}(x))_1 \neq 0 \\ x^{-1} := \text{class of } (0, (\text{RtoNRNR}(x))_2^{-1}) & \text{if } (\text{RtoNRNR}(x))_2 \neq 0 \end{cases}$$
 is the inverse function.
- Reals is Cauchy complete

Outline

- 1 What is UniMath?
- 2 How to use UniMath?
- 3 Contents of the packages**
 - Foundations
 - Combinatorics
 - Category Theory
 - Algebra
 - Real Numbers
 - Topology**
 - More specialized packages

Filters

$\mathcal{F} \subseteq \mathcal{P}(X)$ is a $\text{Filter}(X)$ if:

- $\forall A, B \subseteq X, A \in \mathcal{F} \text{ and } A \subseteq B \Rightarrow B \in \mathcal{F}$
- $X \in \mathcal{F}$
- $\forall A, B \subseteq X, A \in \mathcal{F} \text{ and } B \in \mathcal{F} \Rightarrow A \cap B \in \mathcal{F}$
- $\forall A \subseteq X, A \in \mathcal{F} \Rightarrow \exists x \in X, x \in A$

Filters

$\mathcal{F} \subseteq \mathcal{P}(X)$ is a $\text{Filter}(X)$ if:

- $\forall A, B \subseteq X, A \in \mathcal{F} \text{ and } A \subseteq B \Rightarrow B \in \mathcal{F}$
- $X \in \mathcal{F}$
- $\forall A, B \subseteq X, A \in \mathcal{F} \text{ and } B \in \mathcal{F} \Rightarrow A \cap B \in \mathcal{F}$
- $\forall A \subseteq X, A \in \mathcal{F} \Rightarrow \exists x \in X, x \in A$

Various filters:

locally x the collection of neighborhoods of x is a filter,

FilterIm let \mathcal{F} a filter on X and $f : X \rightarrow Y$,

then $\{A \subseteq Y \mid f^{-1}(A) \in \mathcal{F}\}$ is a filter on Y, \dots

Filters

$\mathcal{F} \subseteq \mathcal{P}(X)$ is a $\text{Filter}(X)$ if:

- $\forall A, B \subseteq X, A \in \mathcal{F} \text{ and } A \subseteq B \Rightarrow B \in \mathcal{F}$
- $X \in \mathcal{F}$
- $\forall A, B \subseteq X, A \in \mathcal{F} \text{ and } B \in \mathcal{F} \Rightarrow A \cap B \in \mathcal{F}$
- $\forall A \subseteq X, A \in \mathcal{F} \Rightarrow \exists x \in X, x \in A$

Various filters:

locally x the collection of neighborhoods of x is a filter,

FilterIm let \mathcal{F} a filter on X and $f : X \rightarrow Y$,
then $\{A \subseteq Y \mid f^{-1}(A) \in \mathcal{F}\}$ is a filter on Y, \dots

Limit:

$\text{filterlim } (f : X \rightarrow Y) (\mathcal{F} : \text{Filter } X) (\mathcal{G} : \text{Filter } Y)$
:=

$$\forall A \subseteq Y, A \in \mathcal{G} \Rightarrow A \in \text{FilterIm}(\mathcal{F}, f)$$

Filters

$\mathcal{F} \subseteq \mathcal{P}(X)$ is a $\text{Filter}(X)$ if:

- $\forall A, B \subseteq X, A \in \mathcal{F} \text{ and } A \subseteq B \Rightarrow B \in \mathcal{F}$
- $X \in \mathcal{F}$
- $\forall A, B \subseteq X, A \in \mathcal{F} \text{ and } B \in \mathcal{F} \Rightarrow A \cap B \in \mathcal{F}$
- $\forall A \subseteq X, A \in \mathcal{F} \Rightarrow \exists x \in X, x \in A$

Various filters:

locally x the collection of **neighborhoods** of x is a filter,

FilterIm let \mathcal{F} a filter on X and $f : X \rightarrow Y$,
then $\{A \subseteq Y \mid f^{-1}(A) \in \mathcal{F}\}$ is a filter on Y, \dots

Limit:

$\text{filterlim} (f: X \rightarrow Y) (\mathcal{F}: \text{Filter } X) (\mathcal{G}: \text{Filter } Y)$
:=

$$\forall A \subseteq Y, A \in \mathcal{G} \Rightarrow A \in \text{FilterIm}(\mathcal{F}, f)$$

Topology

A topology on X is a collection $\mathcal{O} \subseteq \mathcal{P}(X)$ of open sets such that:

- $\forall \mathcal{A} \subseteq \mathcal{P}(X), (\forall A \in \mathcal{A} \Rightarrow A \in \mathcal{O}) \Rightarrow \bigcup_{A \in \mathcal{A}} A \in \mathcal{O}$
- $X \in \mathcal{O}$
- $\forall A, B \subseteq X, A \in \mathcal{O} \text{ and } B \in \mathcal{O} \Rightarrow A \cap B \in \mathcal{O}$

Definitions of topology:

- using a base of topology
- as product of two topologies
- using the collection of all neighborhood of all points
- ...

Uniform Structure

A uniform structure on X is a collection $\mathcal{U} \subseteq \mathcal{P}(X \times X)$ such that:

- $\forall A, B \subseteq X \times X, A \in \mathcal{U}$ and $A \subseteq B \Rightarrow B \in \mathcal{U}$
- $X \in \mathcal{F}$
- $\forall A, B \subseteq X \times X, A \in \mathcal{U}$ and $B \in \mathcal{U} \Rightarrow A \cap B \in \mathcal{U}$
- $\forall A \subseteq X \times X, \forall x \in X, (x, x) \in A$
- $\forall A \subseteq X \times X, \{(y, x) \in X \times X ; (x, y) \in A\} \in \mathcal{U}$
- $\forall A \subseteq X \times X, A \in \mathcal{U} \Rightarrow \exists B \subseteq X \times X, B \in \mathcal{U}$ and $\{(x, y) \in X \times X \mid \exists z \in X, (x, z) \in B \wedge (y, z) \in B\} \subset A$

Uniform Completeness

Cauchy filter: Let \mathcal{U} a uniform structure on X and \mathcal{F} a filter on X .
 \mathcal{F} is a Cauchy filter if:

$$\forall A \subseteq X, A \in \mathcal{F} \Rightarrow \exists B \in \mathcal{U}, A \times A \subseteq B$$

Uniformly complete: Let \mathcal{U} a uniform structure on X .
 X is complete if all Cauchy filter have a limit:

$$\exists x \in X, \text{ filterlim}(id, \mathcal{F}, \text{locally}(x))$$

Outline

- 1 What is UniMath?
- 2 How to use UniMath?
- 3 Contents of the packages**
 - Foundations
 - Combinatorics
 - Category Theory
 - Algebra
 - Real Numbers
 - Topology
 - More specialized packages**

More specialized packages

- Folds: categories via univalent FOLDS
- Homological Algebra: 5-lemma for triangulated categories, naive homotopy category $K(A)$ is triangulated
- Ktheory: Torsors, the circle as $B(\mathbb{Z})$
- Number Systems: Integers and rationals via constructions in Algebra package
- SubstitutionSystems: theory of syntax with variable binding
- Tactics: tactics for proving results on monoids, groups, etc.

More specialized packages

- Folds: categories via univalent FOLDS
- Homological Algebra: 5-lemma for triangulated categories, naive homotopy category $K(A)$ is triangulated
- Ktheory: Torsors, the circle as $B(\mathbb{Z})$
- Number Systems: Integers and rationals via constructions in Algebra package
- SubstitutionSystems: theory of syntax with variable binding
- Tactics: tactics for proving results on monoids, groups, etc.

Thanks for your attention!

Archimedean Property in UniMath

One definition by algebraic structure:

`isarchrig X :=`

- $\forall y_1, y_2 \in X, y_1 < y_2 \Rightarrow \exists n \in \mathbb{N}, 1 + n \cdot y_1 < n \cdot y_2,$
- $\forall x \in X, \exists n \in \mathbb{N}, x < n,$ and
- $\forall x : X, \exists n \in \mathbb{N}, 0 < n + x.$

`isarchrng X :=`

- $\forall x \in X, 0 < x \Rightarrow \exists n \in \mathbb{N}, 1 < n \cdot x$ and
- $\forall x \in X, \exists n \in \mathbb{N}, x < n.$

`isarchfld X :=`

- $\forall x \in X, \exists n \in \mathbb{N}, x < n.$

Proved: localization theorems, e.g.

`isarchrightorng ... : isarchrig X → isarchrng
(rightorng X)`

Lattice

Let X a set, $\min, \max : X \rightarrow X \rightarrow X$.

X is a **lattice** if:

- \min and \max are associative and commutative,
- $\forall x, y \in X, \min(x, \max(x, y)) = x$,
- $\forall x, y \in X, \max(x, \min(x, y)) = x$.

Proved: $\text{lt} : (x, y) \mapsto \min(x, y) = x$ is a partial order.

Lattices with other properties:

`bounded_lattice` lattice with a greater and a lower element

`latticewithgt` lattice with a strict order `gt` such that:

$$\forall x, y \in X, \neg \text{gt}(x, y) \Leftrightarrow \text{le}(x, y)$$

`latticedec` lattice where `le` is total and decidable

Lattice in algebraic structures

Truncated minus: Let X a monoid with a lattice structure, and minus a binary operator on X .
 minus is a truncated minus if:

$$\forall x, y \in X, \quad \text{minus}(x, y) + y = \max(x, y)$$

Proved: localization theorems, *e.g.*
if X is a monoid with a lattice structure lat , then
 $\text{abgrdiff_lattice } \text{lat}$ is a lattice structure on $\text{abgrdiff } X$.

Apartness and Order

Let $X, Y \in \text{Dcuts}$:

$$X < Y := \exists q \in Y, q \notin X$$

$$X \leq Y := \forall q \in X, q \in Y$$

$$X \neq Y := X < Y \amalg Y < X$$

Proved:

- order properties of \leq and $<$
- $x \leq y \Leftrightarrow \neg(y < x)$
- \neq is a tight apartness relation

Arithmetic Operations

Let $r \in \mathbb{Q}_{\geq 0}$, $X, Y \in \text{Dcuts}$:

$$[r]_{\text{Dcuts}} := \{q \in \mathbb{Q}_{\geq 0} \mid q < r\}$$

$$X + Y := X \cup Y \cup \{x + y \mid x \in X \text{ and } y \in Y\}$$

$$X \times Y := \{x \times y \mid x \in X \text{ and } y \in Y\}$$

$$(X, \neq : X \neq 0)^{-1} := \{q \mid \exists l \in (0, 1), \exists x \in X, q \times x \leq l\}$$

Proved:

- $([0]_{\text{Dcuts}}, [1]_{\text{Dcuts}}, +, \times, \square^{-1}, \neq)$ is a constructive division rig
- $+$ and \times are compatible with order

Lattice structure

Let $X, Y \in \text{Dcuts}$:

$$\max(X, Y) := X \cup Y$$

$$\min(X, Y) := X \cap Y$$

$$X - Y := \{q \mid \exists x \in X, \forall y \in Y \cup \{0\}, q + y < x\}$$

Proved:

$$\forall X, Y \in \text{Dcuts}, (X - Y) + Y = \max(X, Y)$$

Dcuts is a Complete Set

Proved:

- Cauchy completeness
- Dedekind completeness

Dcuts is a Complete Set

Proved:

- Cauchy completeness
- Dedekind completeness
- *Uniform completeness*