Volume IV: Mathematics

Introduction to Formalization of Mathematics

Namdak Tonpa

2024 · Groupoid Infinity

IV

Зміст

Issue XXIV: Category Theory

Максим Сохацький [1]

[1] Національний технічний університет України
Київський політехнічний інститут імені Ігоря Сікорського
5 травня 2025 р.

Анотація

Formal definition of Category.
Keywords: Category Theory

1 Category Theory

Category Theory provides a rigorous framework for abstracting and unifying mathematical structures. Developed in the 1940s by Samuel Eilenberg and Saunders Mac Lane to address coherence problems in algebraic topology, it generalizes relationships between mathematical objects across diverse fields like algebra, geometry, and computer science. Category Theory captures objects and their morphisms—functions preserving structure—as a universal systems theory, akin to a universal algebra of functions, emphasizing composition and transformation. Interpreted as a foundational language, a tool for structural analysis, or a bridge to computer-aided formalization, it solves problems of abstraction and generalization. Categories serve as a stepping stone to topos theory, which enriches logical and geometric insights, and higher cohesive topos theory, extending to infinity-categories for advanced applications.

1.1 Category

First of all very simple category theory up to pullbacks is provided. We give here all definitions only to keep the context valid.

A **category** $\mathcal{C}$ consists of:

- A class of **objects**, $\mathrm{Ob}(\mathcal{C})$,
- A class of **morphisms**, $\mathrm{Hom}_{\mathcal{C}}(X, Y)$, for each pair $X, Y \in \mathrm{Ob}(\mathcal{C})$,
- Composition maps $\circ : \mathrm{Hom}(Y, Z) \times \mathrm{Hom}(X, Y) \to \mathrm{Hom}(X, Z)$,
- Identity morphisms $\mathrm{id}_X \in \mathrm{Hom}(X, X)$ for each X,

satisfying associativity and identity laws.

Definition 1. (Category Signature). The signature of category is a $\sum_{A:U} A \to A \to U$ where U could be any universe. The pr_1 projection is called Ob and pr_2 projection is called $\mathrm{Hom}(a, b)$, where $a, b : \mathrm{Ob}$.

```
cat: U = (A: U) * (A -> A -> U)
```

Definition 2. (Precategory). More formal, precategory C consists of the following. (i) A type Ob_C, whose elements are called objects; (ii) for each $a, b : \mathrm{Ob}_C$, a set $\mathrm{Hom}_C(a, b)$, whose elements are called arrows or morphisms. (iii) For each $a : \mathrm{Ob}_C$, a morphism $1_a : \mathrm{Hom}_C(a, a)$, called the identity morphism. (iv) For each $a, b, c : \mathrm{Ob}_C$, a function $\mathrm{Hom}_C(b, c) \to \mathrm{Hom}_C(a, b) \to \mathrm{Hom}_C(a, c)$ called composition, and denoted $g \circ f$. (v) For each $a, b : \mathrm{Ob}_C$ and $f : \mathrm{Hom}_C(a, b)$, $f = 1_b \circ f$ and $f = f \circ 1_a$. (vi) For each $a, b, c, d : A$ and $f : \mathrm{Hom}_C(a, b)$, $g : \mathrm{Hom}_C(b, c)$, $h : \mathrm{Hom}_C(c, d)$, $h \circ (g \circ f) = (h \circ g) \circ f$.

```
def cat : U₁
 := Σ (ob: U) (hom: ob -> ob -> U), unit

def isPrecategory (C: cat) : U := Σ
    (id:       Π (x: C.ob), C.hom x x)
    (∘:        Π (x y z: C.ob),
                 C.hom x y -> C.hom y z -> C.hom x z)
    (homSet:   Π (x y: C.ob), isSet (C.hom x y))
    (∘-left:   Π (x y: C.ob) (f: C.hom x y),
               = (C.hom x y) (∘ x x y (id x) f) f)
    (∘-right:  Π (x y: C.ob) (f: C.hom x y),
               = (C.hom x y) (∘ x y y f (id y)) f)
    (∘-assoc:  Π (x y z w: C.ob) (f: C.hom x y)
                 (g: C.hom y z) (h: C.hom z w),
               = (C.hom x w) (∘ x z w (∘ x y z f g) h)
                             (∘ x y w f (∘ y z w g h))), 1
def precategory: U₁ := Σ (C: cat) (P: isPrecategory C), unit
```

Univalent Categories:

```
def isoCat (P: precategory) (A B: P.C.ob) : U := Σ
    (f: P.C.hom A B)
    (g: P.C.hom B A)
    (retract: Path (P.C.hom A A) (P.P.∘ A B A f g) (P.P.id A))
    (section: Path (P.C.hom B B) (P.P.∘ B A B g f) (P.P.id B)), 1

def isCategory (P: precategory): U
 := Σ (A: P.C.ob), isContr (Π (B: P.C.ob), isoCat P A B)

def category: U₁
 := Σ (P: precategory), isCategory P
```

1.2 Pullback

Definition 3. (Categorical Pullback). The pullback of the cospan $A \xrightarrow{f} C \xleftarrow{g} B$ is a object $A \times_C B$ with morphisms $pb_1 : \times_C \to A$, $pb_2 : \times_C \to B$, such that diagram commutes:

$$\begin{tikzcd}
A \times_C B \arrow[r, "pb_2"] \arrow[d, "f"'] \arrow[dr] & B \arrow[d, "pb_1"] \\
A \arrow[r, "g"] & C
\end{tikzcd}$$

Pullback $(\times_C, pb_1, pb_2)$ must be universal, means for any (D, q_1, q_2) for which diagram also commutes there must exists a unique $u : D \to \times_C$, such that $pb_1 \circ u = q_1$ and $pb_2 \circ q_2$.

```
def homTo  (P: precategory) (X: P.C.ob): U
 := Σ (Y: P.C.ob), P.C.hom Y X

def cospan (P: precategory): U
 := Σ (X: P.C.ob) (_: homTo P X), homTo P X

def hasCospanCone (P: precategory) (D: cospan P) (w: P.C.ob) : U
 := Σ (f: P.C.hom w D.2.1.1) (g: P.C.hom w D.2.2.1),
    = (P.C.hom w D.1) (P.P.∘ w D.2.1.1 D.1 f D.2.1.2)
                      (P.P.∘ w D.2.2.1 D.1 g D.2.2.2)
def cospanCone (P: precategory) (D: cospan P): U
 := Σ (w: P.C.ob), hasCospanCone P D w

def isCospanConeHom (P: precategory) (D: cospan P)
    (E1 E2: cospanCone P D) (h: P.C.hom E1.1 E2.1) : U
 := Σ (_ : = (P.C.hom E1.1 D.2.1.1)
             (P.P.∘ E1.1 E2.1 D.2.1.1 h E2.2.1) E1.2.1),
           = (P.C.hom E1.1 D.2.2.1)
             (P.P.∘ E1.1 E2.1 D.2.2.1 h E2.2.2.1) E1.2.2.1

def cospanConeHom (P: precategory) (D: cospan P) (E1 E2: cospanCone P D) : U
 := Σ (h: P.C.hom E1.1 E2.1), isCospanConeHom P D E1 E2 h

def isPullback (P: precategory) (D: cospan P) (E: cospanCone P D) : U
 := Σ (h: cospanCone P D), isContr (cospanConeHom P D h E)

def hasPullback (P: precategory) (D: cospan P) : U
 := Σ (E: cospanCone P D), isPullback P D E
```

1.3 Functor

A **functor** $F : \mathcal{C} \to \mathcal{D}$ assigns to each:

- Object $X \in \mathcal{C}$ an object $F(X) \in \mathcal{D}$,
- Morphism $f : X \to Y$ a morphism $F(f) : F(X) \to F(Y)$,

such that $F(\mathrm{id}_X) = \mathrm{id}_{F(X)}$ and $F(g \circ f) = F(g) \circ F(f)$.

Definition 4. (Category Functor). Let A and B be precategories. A functor $F : A \to B$ consists of: (i) A function $F_{Ob} : Ob_h A \to Ob_B$; (ii) for each $a, b : Ob_A$, a function $F_{Hom} : Hom_A(a, b) \to Hom_B(F_{Ob}(a), F_{Ob}(b))$; (iii) for each $a : Ob_A$, $F_{Ob}(1_a) = 1_{F_{Ob}}(a)$; (iv) for $a, b, c : Ob_A$ and $f : Hom_A(a, b)$ and $g : Hom_A(b, c)$, $F(g \circ f) = F_{Hom}(g) \circ F_{Hom}(f)$.

```
def catfunctor (A B: precategory): U
 := Σ (ob: A.C.ob -> B.C.ob)
       (mor:    Π (x y: A.C.ob),
                  A.C.hom x y -> B.C.hom (ob x) (ob y))
       (id:     Π (x: A.C.ob),
                = (B.C.hom (ob x) (ob x))
                  (mor x x (A.P.id x)) (B.P.id (ob x)))
       (fcomp: Π (x y z: A.C.ob) (f: A.C.hom x y) (g: A.C.hom y z),
                = (B.C.hom (ob x) (ob z))
                  (mor x z (A.P.∘ x y z f g))
                  (B.P.∘ (ob x) (ob y) (ob z) (mor x y f) (mor y z g))), 1
```

1.4 Terminals

Definition 5. (Terminal Object). Is such object Ob_C, that

$$\prod_{x,y:\mathrm{Ob}_C} \mathrm{isContr}(\mathrm{Hom}_C(y,x)).$$

```
def isInitial (P: precategory) (bot: P.C.ob): U
 := Π (x: P.C.ob), isContr (P.C.hom bot x)

def isTerminal (P: precategory) (top: P.C.ob): U
 := Π (x: P.C.ob), isContr (P.C.hom x top)

def initial (P: precategory): U
 := Σ (bot: P.C.ob), isInitial P bot

def terminal (P: precategory): U
 := Σ (top: P.C.ob), isTerminal P top
```

1.5 Natural Transformation

A **natural transformation** $\eta : \mathsf{F} \Rightarrow \mathsf{G}$ between functors $\mathsf{F}, \mathsf{G} : \mathcal{C} \to \mathcal{D}$ consists of morphisms $\eta_{\mathsf{X}} : \mathsf{F}(\mathsf{X}) \to \mathsf{G}(\mathsf{X})$ such that for every $\mathsf{f} : \mathsf{X} \to \mathsf{Y}$ in $\mathcal{C}$,

$$\begin{tikzcd} \mathsf{F}(\mathsf{X}) \arrow[r, "\eta_{\mathsf{X}}"] \arrow[d, "\mathsf{F}(\mathsf{f})"'] & \mathsf{G}(\mathsf{X}) \arrow[d, "\mathsf{G}(\mathsf{f})"] \\ \mathsf{F}(\mathsf{Y}) \arrow[r, "\eta_{\mathsf{Y}}"'] & \mathsf{G}(\mathsf{Y}) \end{tikzcd}$$

commutes.

```
def isNaturalTransformation
     (C D: precategory)
     (F G: catfunctor C D)
     (eta: Π (x: C.C.ob), D.C.hom (F.ob x) (G.ob x)) : U
 := Π (x y: C.C.ob) (h: C.C.hom x y),
    = (D.C.hom (F.ob x) (G.ob y))
      (D.P.∘ (F.ob x) (F.ob y) (G.ob y) (F.mor x y h) (eta y))
      (D.P.∘ (F.ob x) (G.ob x) (G.ob y) (eta x) (G.mor x y h))

def nattrans (C D: precategory) (F G: catfunctor C D): U
 := Σ (η: Π (x: C.C.ob), D.C.hom (F.ob x) (G.ob x))
      (commute: isNaturalTransformation C D F G η), unit

def natiso (C D: precategory) (F G: catfunctor C D) : U
 := Σ (left: nattrans C D F G)
      (right: nattrans C D G F), 1
```

1.6 Adjunction

An **adjunction** between categories $\mathcal{C}$ and $\mathcal{D}$ consists of functors

$$\mathsf{F} : \mathcal{C} \leftrightarrows \mathcal{D} : \mathsf{G}$$

and natural transformations (unit η and counit ε)

$$\eta : \mathrm{Id}_{\mathcal{C}} \Rightarrow \mathsf{G} \circ \mathsf{F}, \quad \varepsilon : \mathsf{F} \circ \mathsf{G} \Rightarrow \mathrm{Id}_{\mathcal{D}}$$

satisfying the triangle identities.

```
ntransL (C D: precategory) (F G: catfunctor C D)
         (f: ntrans C D F G) (B: precategory) (H: catfunctor B C)
       : ntrans B D (compFunctor B C D H F) (compFunctor B C D H G)
       = (eta, p) where
         F': catfunctor B D = compFunctor B C D H F
         G': catfunctor B D = compFunctor B C D H G
         eta (x: carrier B): hom D (F'.1 x) (G'.1 x) = f.1 (H.1 x)
         p (x y: carrier B) (h: hom B x y): Path (hom D (F'.1 x) (G'.1 y))
              (compose D (F'.1 x) (F'.1 y) (G'.1 y) (F'.2.1 x y h) (eta y))
              (compose D (F'.1 x) (G'.1 x) (G'.1 y) (eta x) (G'.2.1 x y h))
           = f.2 (H.1 x) (H.1 y) (H.2.1 x y h)

ntransR (C D: precategory) (F G: catfunctor C D)
     (f: ntrans C D F G) (E: precategory) (H: catfunctor D E)
   : ntrans C E (compFunctor C D E F H) (compFunctor C D E G H)
   = (eta, p) where
     F': catfunctor C E = compFunctor C D E F H
     G': catfunctor C E = compFunctor C D E G H
     eta (x: carrier C): hom E (F'.1 x) (G'.1 x)
       = H.2.1 (F.1 x) (G.1 x) (f.1 x)
     p (x y: carrier C) (h: hom C x y): Path (hom E (F'.1 x) (G'.1 y))
         (compose E (F'.1 x) (F'.1 y) (G'.1 y) (F'.2.1 x y h) (eta y))
         (compose E (F'.1 x) (G'.1 x) (G'.1 y) (eta x) (G'.2.1 x y h))
       = <i> comp (<_> hom E (F'.1 x) (G'.1 y))
                  (H.2.1 (F.1 x) (G.1 y) (f.2 x y h @ i))
         [ (i =
0) -> H.2.2.2 (F.1 x) (F.1 y) (G.1 y) (F.2.1 x y h) (f.1 y),
           (i =
1) -> H.2.2.2 (F.1 x) (G.1 x) (G.1 y) (f.1 x) (G.2.1 x y h) ]
```

```
areAdjoint (C D: precategory)
           (F: catfunctor D C)
           (G: catfunctor C D)
           (unit: ntrans D D (idFunctor D) (compFunctor D C D F G))
           (counit: ntrans C C (compFunctor C D C G F) (idFunctor C)): U
  = prod  ((x: carrier C) -> = (hom D (G.1 x) (G.1 x))
                               (path D (G.1 x)) (h0 x))
          ((x: carrier D) -> = (hom C (F.1 x) (F.1 x))
                               (path C (F.1 x)) (h1 x)) where
    h0 (x:  carrier C) : hom D (G.1 x) (G.1 x)
                 = compose D (G.1 x) (G.1 (F.1 (G.1 x))) (G.1 x)
          ((ntransL D D (idFunctor D)
                        (compFunctor D C D F G) unit C G).1 x)
          ((ntransR C C (compFunctor C D C G F)
                        (idFunctor C) counit D G).1 x)
    h1 (x: carrier D) : hom C (F.1 x) (F.1 x)
                = compose C (F.1 x) (F.1 (G.1 (F.1 x))) (F.1 x)
          ((ntransR D D (idFunctor D)
                        (compFunctor D C D F G) unit C F).1 x)
          ((ntransL C C (compFunctor C D C G F)
                        (idFunctor C) counit D F).1 x)

adjoint (C D: precategory) (F: catfunctor D C) (G: catfunctor C D): U
  = (unit: ntrans D D (idFunctor D) (compFunctor D C D F G))
  * (counit: ntrans C C (compFunctor C D C G F) (idFunctor C))
  * areAdjoint C D F G unit counit
```

Issue XXV: Locally Cartesian Closed Categories

Namdak Tonpa

May 5, 2025

Анотація

We introduce locally cartesian closed categories (LCCCs), a class of categories where each slice category is cartesian closed. Definitions of categories, slice categories, and cartesian closed categories are provided, followed by the formal definition of LCCCs. We discuss their significance in categorical logic and dependent type theory, including a theorem on their correspondence to type theories with dependent products

2 Locally Cartesian Closed Categories

Locally cartesian closed categories (LCCCs) are categories where each slice category $\mathcal{C}/x$ is cartesian closed, meaning it has products, exponentials, and a terminal object. LCCCs are fundamental in categorical logic, providing models for dependent type theories with dependent products. This article defines the necessary structures, presents key properties, and highlights their role in type theory, with references from the nLab.

Definition 6 (Cartesian Closed Category)**.** A *cartesian closed category* (CCC) is a category $\mathcal{C}$ equipped with:

- A *terminal object* $1 \in \mathrm{ob}(\mathcal{C})$, such that for every $x \in \mathrm{ob}(\mathcal{C})$, there exists a unique morphism $!_x : x \to 1$.
- For each pair $A, B \in \mathrm{ob}(\mathcal{C})$, a *product* $A \times B \in \mathrm{ob}(\mathcal{C})$ with projections $p_1 : A \times B \to A$, $p_2 : A \times B \to B$, and a universal property: for any $X \in \mathrm{ob}(\mathcal{C})$ with morphisms $f : X \to A$, $g : X \to B$, there exists a unique $\langle f, g\rangle : X \to A \times B$ such that $p_1 \circ \langle f, g\rangle = f$ and $p_2 \circ \langle f, g\rangle = g$.

$$\begin{tikzcd}
X \arrow[rr, dashed, "\langle f, g\rangle"] \arrow[dr, "f"'] \arrow[drrr, "g"] & & A \times B \arrow[dl, "p_1"'] \arrow[dr, "p_2"] & \\
& A & & B
\end{tikzcd}$$

- For each pair $A, B \in \mathrm{ob}(\mathcal{C})$, an *exponential object* $B^A \in \mathrm{ob}(\mathcal{C})$ with an evaluation morphism $\mathrm{ev} : B^A \times A \to B$, and a universal property: for any $X \in \mathrm{ob}(\mathcal{C})$ with $f : X \times A \to B$, there exists a unique $\lambda f : X \to B^A$ such that $\mathrm{ev} \circ (\lambda f \times \mathrm{id}_A) = f$.

$$\begin{tikzcd} X \times A \arrow[rr, dashed, "\lambda f \times \mathrm{id}_A"] \arrow[dr, "f"'] & & B^A \times A \arrow[dl, "\mathrm{ev}"] \\ & B & \end{tikzcd}$$

Remark 1. A CCC has finite products (via the terminal object and binary products) and internal homs (via exponentials), making it a model for simply typed lambda calculus.

Definition 7 (Locally Cartesian Closed Category)**.** A category $\mathcal{C}$ is *locally cartesian closed* if, for every object $x \in \mathrm{ob}(\mathcal{C})$, the slice category $\mathcal{C}/x$ is cartesian closed, i.e., $\mathcal{C}/x$ has a terminal object, binary products, and exponential objects.

Theorem 1 (LCCCs and Dependent Type Theory)**.** (Seely, [3]) A locally cartesian closed category $\mathcal{C}$ provides a categorical model for a dependent type theory with dependent products. Conversely, any dependent type theory with dependent sums and products can be interpreted in an LCCC.

Sketch. In an LCCC $\mathcal{C}$, the slice category $\mathcal{C}/x$ models the context of types over a base type x. The terminal object in $\mathcal{C}/x$ corresponds to the trivial type, products in $\mathcal{C}/x$ correspond to dependent pairs, and exponentials model dependent function types. The pullback functor along morphisms $f : y \to x$ in $\mathcal{C}$ corresponds to substitution in type theory. The universal properties of products and exponentials in each $\mathcal{C}/x$ ensure the rules of dependent products are satisfied. Conversely, a type theory with dependent sums and products constructs an LCCC via its syntactic category, where contexts are objects and terms are morphisms. $\square$

2.1 Examples

1. The category $\mathbf{Set}$ of sets is locally cartesian closed. For any set X, the slice category $\mathbf{Set}/X$ is equivalent to the category of X-indexed families of sets, which has products, exponentials, and a terminal object (the identity family).

2. The category $\mathbf{Top}$ of topological spaces is not locally cartesian closed, as not all slice categories $\mathbf{Top}/X$ are cartesian closed (e.g., exponentials may not exist for arbitrary spaces).

3. The category of presheaves $\mathbf{Set}^{\mathcal{C}^{op}}$ on a small category $\mathcal{C}$ is locally cartesian closed, as each slice $\mathbf{Set}^{\mathcal{C}^{op}}/F$ is equivalent to a presheaf category over a comma category, which is cartesian closed.

2.2 Conclusion

Locally cartesian closed categories bridge category theory and dependent type theory, providing a semantic framework for modeling complex type systems. Their slice categories' cartesian closed structure supports dependent products, making them a powerful tool in categorical logic. Theorem 1 underscores their significance, and examples like **Set** illustrate their applicability.

Література

[1] S. Awodey, *Category Theory*, Oxford Logic Guides, Vol. 49, Oxford University Press, 2010.

[2] P. T. Johnstone, *Sketches of an Elephant: A Topos Theory Compendium*, Oxford Logic Guides, Vol. 43–44, Oxford University Press, 2002.

[3] R. A. G. Seely, *Locally cartesian closed categories and type theory*, Mathematical Proceedings of the Cambridge Philosophical Society, Vol. 95, No. 1, 1987, pp. 33–48.

Issue XXVI: Symmetric Monoidal Categories

Namdak Tonpa

May 5, 2025

Анотація

We present the formal definitions of monoidal, braided, and symmetric monoidal categories, emphasizing their coherence conditions. Key theorems, including Mac Lane's coherence theorem for monoidal categories and the coherence theorem for symmetric monoidal categories, are discussed. The exposition is grounded in category theory, with diagrams illustrating the triangle, pentagon, and hexagon identities.

3 Symmetric Monoidal Categories

Monoidal categories provide a framework for studying algebraic structures with a tensor product, such as vector spaces or abelian groups. Braided and symmetric monoidal categories introduce commutativity via a braiding or symmetry, with applications in topology, quantum algebra, and theoretical physics. This article defines these structures and their coherence conditions, culminating in coherence theorems that ensure the consistency of associativity, unit, and braiding operations. We follow the categorical formalism pioneered by Saunders Mac Lane and Max Kelly.

Definition 8 (Monoidal Category)**.** A *monoidal category* is a category $\mathcal{C}$ equipped with:

- A functor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$, called the tensor product.
- An object $I \in \mathrm{ob}(\mathcal{C})$, called the unit object.
- Natural isomorphisms:

$$\lambda_x : I \otimes x \to x \quad \text{(left unitor)},$$

$$\rho_x : x \otimes I \to x \quad \text{(right unitor)},$$

$$\alpha_{x,y,z} : (x \otimes y) \otimes z \to x \otimes (y \otimes z) \quad \text{(associator)},$$

satisfying the following coherence conditions:

- *Triangle identity*: For all $x, y \in \mathrm{ob}(\mathcal{C})$,

$$\alpha_{x,I,y} \circ \rho_x \otimes \mathrm{id}_y = \mathrm{id}_x \otimes \lambda_y : (x \otimes I) \otimes y \to x \otimes y.$$

$$\begin{tikzcd}
 & (x \otimes I) \otimes y \arrow[dl, "\rho_x \otimes \mathrm{id}_y"'] \arrow[dr, "\alpha_{x,I,y}"] & \\
x \otimes y & & x \otimes (I \otimes y) \arrow[ll, "\mathrm{id}_x \otimes \lambda_y"]
\end{tikzcd}$$

- *Pentagon identity*: For all $x, y, z, w \in \mathrm{ob}(\mathcal{C})$,

$$\alpha_{x,y,z\otimes w} \circ \alpha_{x\otimes y,z,w} = (\mathrm{id}_x \otimes \alpha_{y,z,w}) \circ \alpha_{x,y\otimes z,w} \circ \alpha_{x,y,z} \otimes \mathrm{id}_w : ((x\otimes y)\otimes z)\otimes w \to x\otimes(y\otimes(z\otimes w)).$$

$$\begin{tikzcd}
(x \otimes y) \otimes (z \otimes w) \arrow[rr, "\alpha_{x,y,z\otimes w}"] & & x \otimes (y \otimes (z \otimes w)) \\
 & ((x \otimes y) \otimes z) \otimes w \arrow[ul, "\alpha_{x\otimes y,z,w}"'] \arrow[dl, "\alpha_{x,y,z} \otimes \mathrm{id}_w"'] & \\
(x \otimes (y \otimes z)) \otimes w \arrow[rr, "\alpha_{x,y\otimes z,w}"'] & & x \otimes ((y \otimes z) \otimes w) \arrow[uu, "\mathrm{id}_x \otimes \alpha_{y,z,w}"']
\end{tikzcd}$$

Theorem 2 (Coherence for Monoidal Categories)**.** (Mac Lane, [1]) In a monoidal category, every diagram composed of instances of α, λ, ρ, their inverses, identities, and tensor products, that has the same source and target, commutes.

Remark 2. The triangle and pentagon identities ensure that all ways of rebracketing tensor products or removing units are consistent. Theorem 2 implies that no additional coherence conditions are needed beyond those specified.

3.1 Definitions

Definition 9 (Braided Monoidal Category)**.** A *braided monoidal category* is a monoidal category $(\mathcal{C}, \otimes, I, \alpha, \lambda, \rho)$ equipped with a natural isomorphism

$$\beta_{x,y} : x \otimes y \to y \otimes x \quad \text{(braiding)},$$

satisfying the following hexagon identities:

- *Hexagon 1*: For all $x, y, z \in \mathrm{ob}(\mathcal{C})$,

  $\alpha_{x,z,y} \circ \beta_{x\otimes y,z} \circ \alpha_{x,y,z} = (\beta_{x,z} \otimes \mathrm{id}_y) \circ \alpha_{x,z,y} \circ (\mathrm{id}_x \otimes \beta_{y,z}) : (x \otimes y) \otimes z \to x \otimes (z \otimes y).$

$$\begin{tikzcd}
(x \otimes z) \otimes y \arrow[rr, "\alpha_{x,z,y}"] & & x \otimes (z \otimes y) \\
& (x \otimes y) \otimes z \arrow[ul, "\beta_{x\otimes y,z}"'] \arrow[d, "\alpha_{x,y,z}"'] & \\
(y \otimes z) \otimes x \arrow[uu, "\alpha_{y,z,x}"] & x \otimes (y \otimes z) \arrow[l, "\beta_{x,y\otimes z}"] \arrow[uur, "\mathrm{id}_x \otimes \beta_{y,z}"'] &
\end{tikzcd}$$

- *Hexagon 2*: For all $x, y, z \in \mathrm{ob}(\mathcal{C})$,

  $\alpha^{-1}_{x,z,y} \circ \beta_{x,y\otimes z} \circ \alpha^{-1}_{x,y,z} = (\mathrm{id}_z \otimes \beta_{x,y}) \circ \alpha^{-1}_{z,x,y} \circ (\beta_{x,z} \otimes \mathrm{id}_y) : x \otimes (y \otimes z) \to (z \otimes x) \otimes y.$

$$\begin{tikzcd}
(z \otimes x) \otimes y & & (x \otimes z) \otimes y \arrow[ll, "\alpha^{-1}_{x,z,y}"'] \\
& x \otimes (y \otimes z) \arrow[ur, "\beta_{x,y\otimes z}"] \arrow[d, "\alpha^{-1}_{x,y,z}"'] & \\
z \otimes (x \otimes y) \arrow[uu, "\alpha^{-1}_{z,x,y}"] & (x \otimes y) \otimes z \arrow[l, "\mathrm{id}_z \otimes \beta_{x,y}"] \arrow[uur, "\beta_{x\otimes y,z}"'] &
\end{tikzcd}$$

Definition 10 (Symmetric Monoidal Category)**.** A *symmetric monoidal category* is a braided monoidal category $(\mathcal{C}, \otimes, I, \alpha, \lambda, \rho, \beta)$ where the braiding satisfies the symmetry condition:

$$\beta_{y,x} \circ \beta_{x,y} = \mathrm{id}_{x\otimes y} : x \otimes y \to x \otimes y,$$

for all $x, y \in \mathrm{ob}(\mathcal{C})$.

3.2 Theorems

Theorem 3 (Coherence for Symmetric Monoidal Categories)**.** (Joyal and Street, [3]) In a symmetric monoidal category, every diagram composed of instances of α, λ, ρ, β, their inverses, identities, and tensor products, that has the same source and target, commutes.

Remark 3. The symmetry condition $\beta_{y,x} \circ \beta_{x,y} = \mathrm{id}_{x \otimes y}$ ensures that the braiding is its own inverse up to isomorphism, distinguishing symmetric monoidal categories from braided ones. Theorem 3 guarantees that all braiding and associativity operations are coherent, extending Theorem 2.

3.3 Examples

1. The category $\mathbf{Set}$ of sets, with cartesian product as the tensor product and a singleton set as the unit, is a symmetric monoidal category. The braiding $\beta_{X,Y} : X \times Y \to Y \times X$ is given by $(x, y) \mapsto (y, x)$.

2. The category $\mathbf{Vect}_k$ of vector spaces over a field k, with the tensor product of vector spaces and k as the unit, is symmetric monoidal. The braiding swaps tensor factors: $v \otimes w \mapsto w \otimes v$.

3. The category $\mathbf{Ab}$ of abelian groups, with tensor product $\otimes_{\mathbb{Z}}$ and $\mathbb{Z}$ as the unit, is symmetric monoidal.

3.4 Conclusion

Symmetric monoidal categories generalize algebraic structures with associative, unital, and commutative operations, with coherence theorems ensuring consistency. These structures are foundational in category theory and have applications in quantum mechanics, knot theory, and computer science. The coherence theorems of Mac Lane and Joyal-Street provide a rigorous foundation for reasoning about such categories.

Література

[1] S. Mac Lane, *Natural associativity and commutativity*, Rice University Studies, Vol. 49, No. 4, 1963.

[2] S. Mac Lane, *Categories for the Working Mathematician*, Springer, Graduate Texts in Mathematics, Vol. 5, 1971 (2nd ed., 1998).

[3] A. Joyal and R. Street, *The geometry of tensor calculus I*, Advances in Mathematics, Vol. 88, No. 1, 1991, pp. 55–112.

[4] P. Etingof, S. Gelaki, D. Nikshych, and V. Ostrik, *Tensor Categories*, Mathematical Surveys and Monographs, Vol. 205, American Mathematical Society, 2015.

Issue XXVII: Categories with Families

Namdak Tonpa

Анотація

We present a categorical model of Martin-Löf Type Theory (MLTT-75) with dependent products (Π-types), dependent sums (Σ-types), identity types (Id-types), and additional types (⊤, U, Bool), formalized in Agda using Categories with Families (CwF) as described in [1]. The model captures type dependency and context extension via a presheaf of families and context comprehension, supporting all specified type formers. Formal definitions and Agda code are provided, with pullback diagrams resembling Awodey's natural models, illustrating the type formers with constructors (e.g., λ, pair, refl) on upper arrows and type formers on lower arrows.

4 Categories with Families

4.1 Визначення

Definition 11 (Fam)**.** Категорія Fam — це категорія сімей множин, де об'єкти є залежними функціональними просторами $(x : A) \to B(x)$, а морфізми з доменом $\Pi(A, B)$ і кодоменом $\Pi(A', B')$ — це пари функцій $\langle f : A \to A', g(x : A) : B(x) \to B'(f(x))\rangle$.

Definition 12 (Π-похідність)**.** Для контексту Γ і типу A позначимо $\Gamma \vdash A = (\gamma : \Gamma) \to A(\gamma)$.

Definition 13 (Σ-охоплення)**.** Для контексту Γ і типу A маємо $\Gamma; A = (\gamma : \Gamma) * A(\gamma)$. Охоплення не є асоціативним:

$$\Gamma; A; B \neq \Gamma; B; A$$

Definition 14 (Контекст)**.** Категорія контекстів C — це категорія, де об'єкти є контекстами, а морфізми — підстановками. Термінальний об'єкт $\Gamma = 0$ у C називається порожнім контекстом. Операція охоплення контексту $\Gamma; A = (x : \Gamma) * A(x)$ має елімінатори: $p : \Gamma; A \vdash \Gamma$, $q : \Gamma; A \vdash A(p)$, що задовольняють універсальну властивість: для будь-якого $\Delta : ob(C)$, морфізму $\gamma : \Delta \to \Gamma$ і терму $a : \Delta \to A$ існує єдиний морфізм $\theta = \langle \gamma, a \rangle : \Delta \to \Gamma; A$, такий що $p \circ \theta = \gamma$ і $q(\theta) = a$. Твердження: підстановка є асоціативною:

$$\gamma(\gamma(\Gamma, x, a), y, b) = \gamma(\gamma(\Gamma, y, b), x, a)$$

Definition 15 (CwF-об'єкт)**.** CwF-об'єкт — це пара $\Sigma(C, C \to \mathsf{Fam})$, де C — категорія контекстів з об'єктами-контекстами та морфізмами-підстановками, а $T : C \to \mathsf{Fam}$ — функтор, який відображає контекст Γ у C на сім'ю множин термів $\Gamma \vdash A$, а підстановку $\gamma : \Delta \to \Gamma$ — на пару функцій, що виконують підстановку γ у термах і типах відповідно.

Definition 16 (CwF-морфізм)**.** Нехай $(C, T) : \mathsf{ob}(C)$, де $T : C \to \mathsf{Fam}$. CwF-морфізм $m : (C, T) \to (C', T')$ — це пара $\langle F : C \to C', \sigma : T \to T'(F) \rangle$, де F — функтор, а σ — натуральна трансформація.

Definition 17 (Категорія типів)**.** Для CwF з об'єктами (C, T) і морфізмами $(C, T) \to (C', T')$, для заданого контексту $\Gamma \in \mathsf{Ob}(C)$ можна побудувати категорію $\mathsf{Type}(\Gamma)$ — категорію типів у контексті Γ, де об'єкти — множина типів у контексті, а морфізми — функції $f : \Gamma; A \to B(p)$.

4.2 Семантика залежної теорії типів

Definition 18 (Терми та типи)**.** У CwF для контексту Γ терми $\Gamma \vdash a : A$ є елементами множини $A(\gamma)$, де $\gamma : \Gamma$. Типи $\Gamma \vdash A$ є об'єктами в $\mathsf{Type}(\Gamma)$, а підстановка $\gamma : \Delta \to \Gamma$ діє на типи та терми через функтор T.

Theorem 4 (Композиція підстановок)**.** Підстановки в категорії контекстів C є асоціативними та мають одиницю (ідентичну підстановку). Формально, для $\gamma : \Delta \to \Gamma$, $\delta : \Theta \to \Delta$ і $\epsilon : \Gamma \to \Lambda$ виконується:

$$(\gamma \circ \delta) \circ \epsilon = \gamma \circ (\delta \circ \epsilon), \quad \mathsf{id}_\Gamma \circ \gamma = \gamma, \quad \gamma \circ \mathsf{id}_\Delta = \gamma.$$

Доведення. Асоціативність випливає з універсальної властивості охоплення контексту (Визначення 1.4). Для будь-яких γ, δ, ϵ композиція морфізмів у C відповідає послідовному застосуванню підстановок, що зберігає структуру контекстів. Ідентична підстановка id_Γ діє як нейтральний елемент, оскільки $p \circ \mathsf{id}_\Gamma = \mathsf{id}_\Gamma$ і $q(\mathsf{id}_\Gamma) = q$. □

Definition 19 (Залежні типи)**.** Залежний тип у контексті Γ — це відображення $\Gamma \to \mathsf{Fam}$, де для кожного $\gamma : \Gamma$ задається множина $A(\gamma)$. У категорії $\mathsf{Type}(\Gamma)$ залежні типи є об'єктами, а морфізми між A і B — це функції $f : \Gamma; A \to B(p)$, що зберігають структуру підстановок.

Theorem 5 (Універсальна властивість залежних типів)**.** Для будь-якого контексту Γ, типу A і терму $a : \Gamma \vdash A$ існує унікальний морфізм $\theta : \Gamma \to \Gamma; A$, який задовольняє $p \circ \theta = \mathsf{id}_\Gamma$ і $q(\theta) = a$. Це забезпечує коректність залежної типізації в CwF.

Доведення. За Визначенням 1.4, універсальна властивість охоплення контексту гарантує існування $\theta = \langle \mathsf{id}_\Gamma, a \rangle$. Унікальність випливає з того, що будь-який інший морфізм θ' з тими ж властивостями ($p \circ \theta' = \mathsf{id}_\Gamma$, $q(\theta') = a$) збігається з θ через єдиність композиції в C. □

Martin-Löf Type Theory (MLTT-75) is a dependent type theory with Π-types, Σ-types, Id-types, and additional type formers like $\top$, universe types (U), and Bool. Its categorical semantics can be modeled using Categories with Families (CwF), a framework designed to capture contexts, types, terms, and context extension in a unified way [1, ?]. Unlike Grothendieck fibrations or comprehension categories, CwFs use a presheaf of families to represent types and terms, with context comprehension for type dependency. We formalize a CwF model for MLTT-75 in Agda, supporting all specified type formers, based on [1]. Pullback diagrams, styled after Awodey's natural models [5], illustrate the type formers, with constructors on upper arrows and type formers on lower arrows.

A Category with Families (CwF) models dependent type theory by assigning types and terms to contexts, with context comprehension for type dependency.

Definition 20 (Category with Families)**.** A *Category with Families* (CwF) consists of:

- A category $\mathcal{C}$ with a terminal object $1 \in \mathcal{C}.\mathrm{Ob}$.
- A presheaf $\mathrm{Ty} : \mathcal{C}^{\mathrm{op}} \to \mathrm{Set}$, assigning to each $\Gamma \in \mathcal{C}.\mathrm{Ob}$ a set $\mathrm{Ty}(\Gamma)$ of types, and to each $\sigma : \Delta \to \Gamma$ a function $\sigma^* : \mathrm{Ty}(\Gamma) \to \mathrm{Ty}(\Delta)$, preserving identities and composition.
- For each $\Gamma \in \mathcal{C}.\mathrm{Ob}$ and $A \in \mathrm{Ty}(\Gamma)$, a set $\mathrm{Tm}(\Gamma, A)$ of terms, with reindexing: for $\sigma : \Delta \to \Gamma$, a function $\mathrm{Tm}(\Gamma, A) \to \mathrm{Tm}(\Delta, \sigma^* A)$, preserving identities and composition.
- For each $\Gamma \in \mathcal{C}.\mathrm{Ob}$ and $A \in \mathrm{Ty}(\Gamma)$, a context comprehension consisting of:
  - An object $\Gamma.A \in \mathcal{C}.\mathrm{Ob}$.
  - A projection morphism $p_A : \Gamma.A \to \Gamma$.
  - A universal term $q_A \in \mathrm{Tm}(\Gamma.A, p_A^* A)$.
  - For any $\Delta \in \mathcal{C}.\mathrm{Ob}$, $\sigma : \Delta \to \Gamma$, and $t \in \mathrm{Tm}(\Delta, \sigma^* A)$, there exists a unique $\langle \sigma, t \rangle : \Delta \to \Gamma.A$ such that $p_A \circ \langle \sigma, t \rangle = \sigma$ and $\langle \sigma, t \rangle^* q_A = t$.

4.3 Type Formers

To model MLTT-75, the CwF is equipped with structure for Π-types, Σ-types, Id-types, $\top$, U, and Bool.

Definition 21 (CwF with MLTT-75 Type Formers)**.** A *CwF with MLTT-75 type formers* extends a CwF with:

- *Empty context*: An object $\bullet \in \mathcal{C}.\mathrm{Ob}$ with a unique morphism $\epsilon : \Gamma \to \bullet$ for any Γ.
- *Pullbacks*: For any $f : A \to B$, $g : C \to B$ in $\mathcal{C}$, there exists a pullback P with morphisms $h_1 : P \to A$, $h_2 : P \to C$ such that $f \circ h_1 = g \circ h_2$, and P is universal.
- Π-*types*: For $\Gamma \in \mathcal{C}.\mathrm{Ob}$, $A \in \mathrm{Ty}(\Gamma)$, $B \in \mathrm{Ty}(\Gamma.A)$, there exists $\Pi_A B \in \mathrm{Ty}(\Gamma)$, a term $\mathrm{lam}(t) \in \mathrm{Tm}(\Gamma, \Pi_A B)$ for $t \in \mathrm{Tm}(\Gamma.A, B)$, and $\mathrm{app} : \mathrm{Tm}(\Gamma, \Pi_A B) \to \mathrm{Tm}(\Gamma.A, B)$, satisfying β- and η-rules.
- Σ-*types*: For $\Gamma \in \mathcal{C}.\mathrm{Ob}$, $A \in \mathrm{Ty}(\Gamma)$, $B \in \mathrm{Ty}(\Gamma.A)$, there exists $\Sigma_A B \in \mathrm{Ty}(\Gamma)$, a term $(u, v) \in \mathrm{Tm}(\Gamma, \Sigma_A B)$ for $u \in \mathrm{Tm}(\Gamma, A)$, $v \in \mathrm{Tm}(\Gamma, B[\mathrm{id}, u])$, and projections fst, snd, satisfying β- and η-rules.
- *Id-types*: For $\Gamma \in \mathcal{C}.\mathrm{Ob}$, $A \in \mathrm{Ty}(\Gamma)$, $a, b \in \mathrm{Tm}(\Gamma, A)$, there exists $\mathrm{Id}_A(a, b) \in \mathrm{Ty}(\Gamma)$, a term $\mathrm{refl}(u) \in \mathrm{Tm}(\Gamma, \mathrm{Id}_A(u, u))$, and an eliminator J, satisfying β-rule.

```
def algebra : U₁ := Σ
    -- a semicategory of contexts and substitutions:
    (Con: U)
    (Sub: Con → Con → U)
    (◊: Π (Γ Θ Δ : Con), Sub Θ Δ → Sub Γ Θ → Sub Γ Δ)
    (◊-assoc: Π (Γ Θ Δ Φ : Con) (σ: Sub Γ Θ) (δ: Sub Θ Δ)
        (ν: Sub Δ Φ), PathP (<_>Sub Γ Φ) (◊ Γ Δ Φ ν (◊ Γ Θ Δ δ σ))
                                          (◊ Γ Θ Φ (◊ Θ Δ Φ ν δ) σ))
    -- identity morphisms as identity substitutions:
    (id: Π (Γ : Con), Sub Γ Γ)
    (id-left: Π (Θ Δ : Con) (δ : Sub Θ Δ),
              Path (Sub Θ Δ) δ (◊ Θ Δ Δ (id Δ) δ))
    (id-right: Π (Θ Δ : Con) (δ : Sub Θ Δ),
               Path (Sub Θ Δ) δ (◊ Θ Θ Δ δ (id Θ)))
    -- a terminal oject as empty context:
    (•: Con)
    (ε: Π (Γ : Con), Sub Γ •)
    (•-η: Π (Γ: Con) (δ: Sub Γ •), Path (Sub Γ •) (ε Γ) δ)
    (Ty: Con → U)
    (_|_|ᵀ: Π (Γ Δ : Con), Ty Δ → Sub Γ Δ → Ty Γ)
    (|id|ᵀ: Π (Δ : Con) (A : Ty Δ), Path (Ty Δ) (_|_|ᵀ Δ Δ
A (id Δ)) A)
    (|◊|ᵀ: Π (Γ Δ Φ: Con) (A : Ty Φ) (σ : Sub Γ Δ) (δ : Sub Δ Φ),
        PathP (<_>Ty Γ) (_|_|ᵀ Γ Φ A (◊ Γ Δ Φ δ σ))
                        (_|_|ᵀ Γ Δ (_|_|ᵀ Δ Φ A δ) σ))
    -- a (covariant) presheaf on the category of elements as terms:
    (Tm: Π (Γ : Con), Ty Γ → U)
    (_|_|ᵗ: Π (Γ Δ : Con) (A : Ty Δ) (B : Tm Δ A)
             (σ: Sub Γ Δ), Tm Γ (_|_|ᵀ Γ Δ A σ))
    (|id|ᵗ: Π (Δ : Con) (A : Ty Δ) (t: Tm Δ A),
            PathP (<i> Tm Δ (|id|ᵀ Δ A @ i))
                  (_|_|ᵗ Δ Δ A t (id Δ)) t)
    (|◊|ᵗ: Π (Γ Δ Φ: Con) (A : Ty Φ) (t: Tm Φ A)
             (σ : Sub Γ Δ) (δ : Sub Δ Φ),
             PathP (<i> Tm Γ (|◊|ᵀ Γ Δ Φ A σ δ @ i))
                   (_|_|ᵗ Γ Φ A t (◊ Γ Δ Φ δ σ))
          (_|_|ᵗ Γ Δ (_|_|ᵀ Δ Φ A δ) (_|_|ᵗ Δ Φ A t δ) σ))
```

4.4 Example MLTT-75 Model

We model MLTT-75 using a CwF with type formers, interpreting contexts, types, and terms via the presheaf of families and context comprehension.

Definition 22 (MLTT-75 CwF Model)**.** Given a CwF with type formers, the model of MLTT-75 is defined as:

- *Contexts*: Objects $\Gamma \in \mathcal{C}.\mathrm{Ob}$, indexed by levels $i \in \mathbb{N}$.
- *Types*: Elements $A \in \mathrm{Ty}_i(\Gamma)$, indexed by levels.
- *Terms*: Elements $t \in \mathrm{Tm}(\Gamma, A)$.
- *Context extension*: For $\Gamma \vdash A$, the context $\Gamma, x : A$ is $\Gamma.A$, with projection $p_A : \Gamma.A \to \Gamma$.
- *Type formers*: Π-types via fiber exponentials, Σ-types via dependent sums, Id-types via diagonals, $\top$ as the unit type, U as a universe, Bool with case analysis.

4.5 Π-Types

For $\Gamma \vdash A : \mathrm{Type}_i$ and $\Gamma, x : A \vdash B : \mathrm{Type}_j$, the Π-type $\Pi_{x:A} B : \mathrm{Type}_{i \sqcup j}$ is formed using exponentials in the presheaf category.

The constructor λ forms terms of $\Pi_{x:A} B$. The pullback diagram is:

$$\begin{tikzcd} \Gamma \times A \arrow[r, "\lambda"] \arrow[d] & B \arrow[d] \\ \Gamma \arrow[r, "\Pi"'] & \Pi_A B \end{tikzcd}$$

4.6 Σ-Types

For $\Gamma \vdash A : \mathrm{Type}_i$ and $\Gamma, x : A \vdash B : \mathrm{Type}_j$, the Σ-type $\Sigma_{x:A} B : \mathrm{Type}_{i \sqcup j}$ is formed via dependent sums in the CwF.

The constructor pair forms terms of $\Sigma_{x:A} B$. The pullback diagram is:

$$\begin{tikzcd} \Sigma_A B \arrow[r, "\mathrm{pair}"] \arrow[d] & B \arrow[d] \\ \Gamma \arrow[r, "\Sigma"'] & \Gamma \times A \end{tikzcd}$$

4.7 Id-Types

For $\Gamma \vdash A : \mathrm{Type}_i$ and $a, b : A$, the identity type $\mathrm{Id}_A(a, b) : \mathrm{Type}_i$ is formed using the diagonal map in the CwF.

The constructor refl forms terms of $\mathrm{Id}_A(a, a)$. The pullback diagram is:

$$\begin{tikzcd}
\mathrm{Id}_A(a, b) \arrow[r, "\mathrm{refl}"] \arrow[d] & A \arrow[d, "\Delta_A"] \\
\Gamma \arrow[r, "\mathrm{Id}"'] & A \times_\Gamma A
\end{tikzcd}$$

4.8 Conclusion

The Agda formalization provides a categorical semantics for MLTT-75 using a Category with Families, capturing type dependency and context extension via a presheaf of families and context comprehension. Supporting Π-types, Σ-types, Id-types, $\top$, U, and Bool, the model, based on [1], contrasts with fibration-based approaches by directly encoding types and terms. The pullback diagrams, styled after Awodey, clarify the categorical constructions. Future work includes exploring univalent extensions and concrete instantiations.

Література

[1] Awodey, S., Frey, J., Speight, S., "Categorical Structures for Type Theory in Univalent Foundations," arXiv:1907.07562, 2019.

[2] Hofmann, M., "On the interpretation of type theory in locally cartesian closed categories," LFCS, 1997.

[3] Jacobs, B., "Comprehension categories and the semantics of type dependency," TCS, 1993.

[4] Awodey, S., "Natural models of homotopy type theory," MSCS, 2018.

Issue XXVIII: Categories with Representable Maps

Максим Сохацький [1]

[1] Національний технічний університет України
Київський політехнічний інститут імені Ігоря Сікорського
5 травня 2025 р.

Анотація

This article presents a modern categorical framework, termed Categories with Representable Maps (CwR), designed to model structures for dependent type theories. Inspired by Uemura's work, the framework unifies related models such as categories with families, categories with attributes, comprehension categories, and natural models. We provide a comprehensive set of classical mathematical definitions and theorems, focusing on specialized categorical structures like fibrations, indexed categories, and representable maps, while establishing their properties and equivalences.

As example we present a categorical model of Martin-Löf Type Theory (MLTT-75) with dependent products (Π-types), dependent sums (Σ-types), and identity types (Id-types). The model is based on Grothendieck fibrations and Uemura's categories with representable maps, generalizing Awodey's natural models. Formal definitions are provided, with pullback diagrams resembling Awodey's style.

5 Categories with Representable Maps

The Categories with Representable Maps (CwR) framework offers a robust foundation for categorical semantics, generalizing prior models used in type theory. Assuming a base category $\mathcal{C}$ with all pullbacks, this framework builds on specialized structures to define representable maps and their properties, ensuring flexibility and unification across related categorical models. This article delineates the core definitions and theorems of the CwR framework, providing a concise yet complete theory.

Martin-Löf Type Theory (MLTT-75) is a dependent type theory with Π-types, Σ-types, and Id-types. We model its categorical semantics using a *category with representable maps* (CwR), starting from Grothendieck fibrations, as described in [1].

5.1 Definitions

Definition 23 (Fiber Category)**.** For a functor $p : \mathcal{E} \to \mathcal{C}$ and an object $c \in \mathcal{C}$, the *fiber category* $\mathcal{E}_c$ has:

- Objects: $e \in \mathcal{E}$ such that $p(e) = c$.
- Morphisms: $f : e' \to e$ in $\mathcal{E}$ such that $p(f) = \mathrm{id}_c$.

Definition 24 (Cartesian Morphism)**.** For a functor $p : \mathcal{E} \to \mathcal{C}$, a morphism $\phi : e' \to e$ in $\mathcal{E}$ is *Cartesian* if, for any $g : e'' \to e$ in $\mathcal{E}$ and $h : p(e'') \to p(e')$ in $\mathcal{C}$ with $p(g) = p(\phi) \circ h$, there exists a unique $k : e'' \to e'$ in $\mathcal{E}$ such that $p(k) = h$ and $g = \phi \circ k$.

Definition 25 (Grothendieck Fibration)**.** A functor $p : \mathcal{E} \to \mathcal{C}$ is a *Grothendieck fibration* if, for every $e \in \mathcal{E}$ and $f : c' \to p(e)$ in $\mathcal{C}$, there exists a Cartesian morphism $\phi : e' \to e$ in $\mathcal{E}$ such that $p(\phi) = f$.

Definition 26 (Grothendieck Construction)**.** For an indexed category $\Phi : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$, the *Grothendieck construction* produces a category $\int \Phi$ with:

- Objects: Pairs (c, x), where $c \in \mathcal{C}$, $x \in \Phi(c)$.
- Morphisms: From $(c', x') \to (c, x)$, pairs (f, α), where $f : c' \to c$ in $\mathcal{C}$, $\alpha : x' \to \Phi(f)(x)$ in $\Phi(c')$.
- Composition: For $(g, \beta) : (c'', x'') \to (c', x')$ and $(f, \alpha) : (c', x') \to (c, x)$, the composite is $(f \circ g, \Phi(g)(\alpha) \circ \beta)$.

The functor $p : \int \Phi \to \mathcal{C}$, mapping $(c, x) \mapsto c$, $(f, \alpha) \mapsto f$, is a Grothendieck fibration.

Definition 27 (Discrete Fibration)**.** A functor $p : \mathcal{E} \to \mathcal{C}$ is a *discrete fibration* if, for every $e \in \mathcal{E}$ and $f : c' \to p(e)$ in $\mathcal{C}$, there exists a unique $\tilde{f} : e' \to e$ in $\mathcal{E}$ such that $p(\tilde{f}) = f$.

Definition 28 (Indexed Category)**.** An *indexed category* over $\mathcal{C}$ is a functor $\Phi : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$. For each $c \in \mathcal{C}$, $\Phi(c)$ is a category, and for each $f : c' \to c$, $\Phi(f) : \Phi(c) \to \Phi(c')$ is a functor.

Definition 29 (Representable Functor)**.** A functor $F : \mathcal{C}^{\mathrm{op}} \to \mathbf{Set}$ is *representable* if there exists $c \in \mathcal{C}$ such that $F \cong \mathrm{Hom}_{\mathcal{C}}(-, c)$.

Definition 30 (Representable Map)**.** In a category $\mathcal{C}$ with pullbacks, a morphism $f : A \to B$ is *representable* if it belongs to a class $\mathrm{Rep}(f)$ satisfying:

- *Pullback stability*: For every $g : C \to B$, the pullback $P = C \times_B A$ exists with projections $h_1 : P \to A$, $h_2 : P \to C$, and $\mathrm{Rep}(h_2)$.
- *Universality*: For any Q with $q_1 : Q \to A$, $q_2 : Q \to C$ such that $f \circ q_1 = g \circ q_2$, there exists a unique $u : Q \to P$ such that $h_1 \circ u = q_1$, $h_2 \circ u = q_2$.

Definition 31 (CwR)**.** A category with representable maps (CwR) is a category with a class of morphisms (representable maps) that are pullback-stable and exponentiable, generalizing Awodey's natural models. A *category with representable maps* (CwR) is a structure with:

- A category $\mathcal{C}$.
- A predicate $\mathrm{Rep} : \mathcal{C}.\mathrm{Hom}(A, B) \to \mathrm{Prop}$ for representable maps.
- *Pullback stability*: For every $f : A \to B$ with $\mathrm{Rep}(f)$ and $g : C \to B$, there exists a pullback P with morphisms $h_1 : P \to A$, $h_2 : P \to C$ such that $f \circ h_1 = g \circ h_2$, $\mathrm{Rep}(h_2)$, and P is universal.
- *Exponentiability*: For every $f : A \to B$ with $\mathrm{Rep}(f)$, there exists $\Pi_f : \mathrm{Ob}$ and $\pi : \Pi_f \to B$ with $\mathrm{Rep}(\pi)$, such that for any $g : C \to A$, there exists $h : C \to \Pi_f$ with $\pi \circ h = f \circ g$.

```
structure CwR where
  cat : Category
  Rep : ∀ {A B : cat.Ob}, cat.Hom A B → Prop
  pullback : ∀ {A B C : cat.Ob} {f : cat.Hom A B},
    Rep f → (g : cat.Hom C B) →
    ∃ (P : cat.Ob) (∃ (h1 : cat.Hom P A) (∃ (h2 : cat.Hom P C)
      (cat.comp f h1 = cat.comp g h2 ∧
       Rep h2 ∧
       ∀ (Q : cat.Ob) (q1 : cat.Hom Q A) (q2 : cat.Hom Q C),
       cat.comp f q1 = cat.comp g q2 →
       ∃ (u : cat.Hom Q P)
         (cat.comp h1 u = q1 ∧ cat.comp h2 u =q2))))
  exponentiable : ∀ {A B : cat.Ob} {f : cat.Hom A B},
    Rep f →
    ∃ (Pi_f : cat.Ob) (∃ (pi : cat.Hom Pi_f B)
      (Rep pi ∧
       ∀ (C : cat.Ob) (g : cat.Hom C A),
       ∃ (h : cat.Hom C Pi_f) (cat.comp pi h =cat.comp f g)))
```

5.2 Theorems

The CwR framework is supported by five theorems that establish its properties and connections to related categorical structures.

Theorem 6 (Fibration-Indexed Category Equivalence)**.** For any indexed category $\Phi : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$, the Grothendieck construction produces a Grothendieck fibration $p : \int \Phi \to \mathcal{C}$, and every Grothendieck fibration arises as the Grothendieck construction of some indexed category.

Theorem 7 (Representable Map Stability)**.** In a CwR $(\mathcal{C}, \mathrm{Rep}, \Pi)$, the class of representable maps is closed under pullback stability, and every representable map $f : A \to B$ induces a representable morphism $\pi_f : \Pi_f \to B$.

Theorem 8 (Discrete Fibration Representation)**.** Every discrete fibration $p : \mathcal{E} \to \mathcal{C}$ corresponds to a representable map in the slice category $\mathcal{C}/c$ for some $c \in \mathcal{C}$, and every representable map induces a discrete fibration in a suitable slice category.

Theorem 9 (Framework Equivalence)**.** Every CwR $(\mathcal{C}, \mathrm{Rep}, \Pi)$ can be equipped with a structure equivalent to a category with families, or natural model under the existence of terminal objects.

5.3 Example MLTT-75 Model

We model MLTT-75 in a CwR, interpreting contexts, types, terms, and type formers.

Definition 32 (MLTT-75 Model)**.** Given a CwR $\mathcal{C}$, the model of MLTT-75 is defined as:

- *Contexts*: Objects $\Gamma \in \mathcal{C}$.Ob.
- *Types*: Pairs $(A, f : A \to \Gamma)$ with Rep(f), representing A in context Γ.
- *Terms*: Morphisms $t : \Gamma \to A$ such that $f \circ t = \mathrm{id}_\Gamma$, i.e., sections of f.
- *Context extension*: For $\Gamma \vdash A$, the context $\Gamma, x : A$ is the pullback of $f : A \to \Gamma$ along id_Γ.
- *Type formers*: Π-types, Σ-types, and Id-types, defined via exponentials, pullbacks, and diagonals.

```
structure MLTT75 (cwr : CwR) where
  Context : Type
  Context := cwr.cat.Ob

  Type : Context → Type
  Type Γ := ∃ (A : cwr.cat.Ob)
    (∃ (f : cwr.cat.Hom A Γ) (cwr.Rep f))

  Term : ∀ (Γ : Context), Type Γ → Type
  Term Γ (∃ A (∃ f _))
    := ∃ (t : cwr.cat.Hom Γ A)
         (cwr.cat.comp f t = cwr.cat.id)

  ContextExt : ∀ (Γ : Context), Type Γ → Context
  ContextExt Γ (∃ A (∃ f rf)) := (cwr.pullback rf cwr.cat.id).fst
```

5.4 Π-Types

For $\Gamma \vdash A : \text{Type}$ and $\Gamma, x : A \vdash B : \text{Type}$, the Π-type $\Pi_{x:A} B$ is formed using the exponential in the slice category.

```
PiType : ∀ (Γ : Context) (A : Type Γ), Type (ContextExt Γ A) → Type Γ
PiType Γ (∃ A (∃ f rf)) (∃ B (∃ g rg)) :=
  let exp := cwr.exponentiable rf
  ∃ exp.fst (∃ exp.snd.fst exp.snd.snd.fst)
```

The constructor λ forms terms of $\Pi_{x:A} B$. The pullback diagram is:

$$\begin{tikzcd} \Gamma \times A \arrow[r, "\lambda"] \arrow[d] & B \arrow[d] \\ \Gamma \arrow[r, "\Pi"'] & \Pi_A B \end{tikzcd}$$

5.5 Σ-Types

For $\Gamma \vdash A : \text{Type}$ and $\Gamma, x : A \vdash B : \text{Type}$, the Σ-type $\Sigma_{x:A} B$ is the composition via pullback.

```
SigmaType : ∀ (Γ : Context) (A : Type Γ), Type (ContextExt Γ A) → Type Γ
SigmaType Γ (∃ A (∃ f rf)) (∃ B (∃ g rg)) :=
  let pull := cwr.pullback rg (cwr.cat.id)
  ∃ pull.fst (∃ pull.snd.fst pull.snd.snd.snd.fst)
```

The constructor pair forms terms of $\Sigma_{x:A} B$. The pullback diagram is:

$$\begin{tikzcd} \Sigma_A B \arrow[r, "\text{pair}"] \arrow[d] & B \arrow[d] \\ \Gamma \arrow[r, "\Sigma"'] & \Gamma \times A \end{tikzcd}$$

5.6 Id-Types

For $\Gamma \vdash A$: Type and $a, b : A$, the identity type $\mathrm{Id}_A(a, b)$ is formed using the diagonal map.

```
Diagonal : ∀ (Γ : Context) (A : Type Γ),
  cwr.cat.Hom (A.fst) (cwr.pullback A.snd.fst cwr.cat.id).fst
Diagonal Γ (∃ A (∃ f _))
  := (cwr.cat.id, cwr.cat.id, rfl)

IdType : ∀ (Γ : Context) (A : Type Γ) (a b : Term Γ A), Type Γ
IdType Γ (∃ A (∃ f rf)) (∃ a _) (∃ b _) :=
  let pull := cwr.pullback rf (Diagonal Γ (∃ A (∃ f rf)))
  ∃ pull.fst (∃ pull.snd.fst pull.snd.snd.snd.fst)
```

The constructor refl forms terms of $\mathrm{Id}_A(a, a)$. The pullback diagram is:

$$\begin{tikzcd} \mathrm{Id}_A(a,b) \arrow[r, "\mathrm{refl}"] \arrow[d] & A \arrow[d, "\Delta_A"] \\ \Gamma \arrow[r, "\mathrm{Id}"'] & A \times_\Gamma A \end{tikzcd}$$

5.7 Conclusion

The CwR framework provides a unified and flexible foundation for categorical semantics, integrating fibrations, indexed categories, and representable maps. Its definitions and theorems ensure robustness and connectivity to related categorical models, making it a powerful tool for theoretical and applied category theory.

Література

[1] Uemura, T., "A general framework for the semantics of type theory," arXiv:1904.04097.

[2] Awodey, S., "Natural models of homotopy type theory," Mathematical Structures in Computer Science, 2018.

Issue XXIX: Comprehension Categories

Namdak Tonpa

Анотація

Comprehension categories provide a powerful categorical framework for modeling dependent type theories, bridging the gap between categorical logic, topos theory, and type-theoretic semantics. This paper presents a unified theoretical framework for comprehension categories, offering precise definitions, key theorems, and novel applications.

We define a comprehension category as a category $\mathcal{C}$ equipped with a fibration $p : \mathcal{E} \to \mathcal{C}$ and a comprehension map that assigns to each type $A \in \mathcal{E}A \in \mathcal{E}$ over a context $\Gamma \in \mathcal{C}$ an extended context $\Gamma.A \in \mathcal{C}$, satisfying pullback stability. We introduce variants, including split and non-split comprehension categories, and contextual categories, to accommodate strict and non-strict type theories. Key theorems include the equivalence theorem, establishing that every comprehension category induces a model of dependent type theory, and the splitting theorem, demonstrating that any comprehension category can be replaced by an equivalent split comprehension category. We further explore the relationship between comprehension categories and related structures, such as Categories with Representations (CwR) and Categories with Families (CwF), highlighting their functorial and computational interpretations. Applications are presented in categorical semantics, homotopy type theory, and topos theory, including the interpretation of univalence axioms and the construction of syntactic categories. This framework unifies existing approaches, clarifies the categorical underpinnings of dependent types, and paves the way for future developments in type-theoretic and geometric foundations of mathematics.

As instantiation example we present a categorical model of Martin-Löf Type Theory (MLTT-75) with dependent products (Π-types), dependent sums (Σ-types), and identity types (Id-types) using Comprehension Categories. The model uses a comprehension category, a Grothendieck fibration with a comprehension functor, to capture type dependency and context extension. Formal definitions are provided, with pullback diagrams resembling Awodey's natural models.

6 Comprehension Categories

Martin-Löf Type Theory (MLTT-75) is a dependent type theory with Π-types, Σ-types, and Id-types. Its categorical semantics is often modeled using

Grothendieck fibrations, with comprehension categories providing a structured framework for type dependency and context extension [**?**, 1]. We formalize a model using a comprehension category, based on a split Grothendieck fibration with a comprehension functor, inspired by the codomain fibration. The model is implemented in Lean 4 without dependencies, ensuring a minimal presentation. Pullback diagrams, styled after Awodey's natural models [5], illustrate the type formers, with constructors (e.g., λ, pair, refl) on upper arrows and type formers on lower arrows.

6.1 Definitions

A split Grothendieck fibration $p : \mathcal{E} \to \mathcal{B}$ models dependent types, with functorial Cartesian lifts for strict substitution.

Definition 33 (Cleavage)**.** A *cleavage* for a Grothendieck fibration $p : \mathcal{E} \to \mathcal{C}$ assigns to each $e \in \mathcal{E}$ and $f : c' \to p(e)$ in $\mathcal{C}$ a Cartesian morphism $\phi_f : f^*e \to e$ in $\mathcal{E}$ such that $p(\phi_f) = f$, where $f^*e \in \mathcal{E}_{c'}$.

Definition 34 (Split Fibration 1)**.** A Grothendieck fibration $p : \mathcal{E} \to \mathcal{C}$ is a *split fibration* if it has a cleavage such that the assignment $f \mapsto f^*e$ defines a functor $f^* : \mathcal{E}_{p(e)} \to \mathcal{E}_{c'}$ for each fiber category $\mathcal{E}_c$, and $(g \circ f)^* = f^* \circ g^*$.

Definition 35 (Split Fibration 2)**.** A *split fibration* $p : \mathcal{E} \to \mathcal{B}$ is a functor p with:

- For every $e \in \mathcal{E}$.Ob and $f : b' \to p(e)$ in $\mathcal{B}$, a chosen lift $(e', \phi : e' \to e)$ with $p(\phi) = f$.
- Uniqueness: For any two lifts (e_1, ϕ_1), (e_2, ϕ_2) with $p(\phi_1) = p(\phi_2) = f$, there exists $\chi : e_2 \to e_1$ with $p(\chi) = \text{id}$ and $\phi_1 \circ \chi = \phi_2$.

```
structure SplitFibration (E B : Category) where
  functor : Functor E B
  lift : ∀ {e : E.Ob} {b' : B.Ob} (f : B.Hom b' (functor.obj e)),
    (e' : E.Ob) × (phi : E.Hom e' e) × (functor.map phi =f)
  lift_unique : ∀ {e : E.Ob} {b' : B.Ob} (f : B.Hom b' (functor.obj e))
    (e1 e2 : E.Ob) (phi1 : E.Hom e1 e) (phi2 : E.Hom e2 e),
    functor.map phi1 = f → functor.map phi2 =f →
    ∃ (chi : E.Hom e2 e1), functor.map chi =
B.id ∧ E.comp phi1 chi =phi2
```

Definition 36 (Arrow Category)**.** The *arrow category* $\mathcal{C}^{\to}$ of a category $\mathcal{C}$ has:

- Objects: Morphisms $f : A \to B$ in $\mathcal{C}$.
- Morphisms: From $f : A \to B$ to $g : C \to D$, a pair $(h_1 : A \to C, h_2 : B \to D)$ such that $g \circ h_1 = h_2 \circ f$.
- Composition: For $(h_1, h_2) : f \to g$ and $(k_1, k_2) : g \to l$, the composite is $(k_1 \circ h_1, k_2 \circ h_2)$.

Definition 37 (Comprehension Functor)**.** For a split fibration $p : \mathcal{E} \to \mathcal{C}$, a *comprehension functor* is a functor $\{-\} : \mathcal{E} \to \mathcal{C}^{\to}$ that maps each object $A \in \mathcal{E}$ to a morphism $\pi : \Gamma' \to p(A)$ in $\mathcal{C}$, and each morphism $f : A \to B$ in $\mathcal{E}$ to a morphism $(h_1, h_2) : \{A\} \to \{B\}$ in $\mathcal{C}^{\to}$.

Definition 38 (Comprehension Category)**.** A *comprehension category* consists of:

- A split fibration $p : \mathcal{E} \to \mathcal{C}$.

- A terminal object $\top \in \mathcal{C}$.
- A comprehension functor $\{-\} : \mathcal{E} \to \mathcal{C}^{\to}$, mapping $A \in \mathcal{E}$ to $(\Gamma', \pi : \Gamma' \to p(A))$.
- An adjunction: For $\sigma : \Delta \to \Gamma$ in $\mathcal{C}$ and $A \in \mathcal{E}_\Gamma$, there exists $A' \in \mathcal{E}_\Delta$ with $p(A') = \Delta$ and a morphism $f : A' \to A$ such that $p(f) = \sigma$.

Definition 39 (Comprehension Category). A comprehension category models MLTT-75 with a fibration and a comprehension functor for context extension. A *comprehension category* consists of:

- A split fibration $p : \mathcal{E} \to \mathcal{B}$.
- A terminal object $\top \in \mathcal{B}$.Ob.
- A comprehension functor $\{-\} : \mathcal{E} \to \mathcal{B}^{\to}$, mapping $A \in \mathcal{E}$ to $(\Gamma', \pi : \Gamma' \to p(A))$.
- An adjunction: For $\sigma : \Delta \to \Gamma$ and $A \in \mathcal{E}_\Gamma$, there exists $A' \in \mathcal{E}_\Delta$ with $p(A') = \Delta$ and a morphism $f : A' \to A$ such that $p(f) = \sigma$.
- Pullbacks in $\mathcal{B}$ for context extension.
- Structure for Π-types (fiber exponentials), Σ-types (composition), and Id-types (diagonals).

Definition 40 (Beck-Chevalley Condition). Let $p : \mathcal{E} \to \mathcal{C}$ be a fibration, and consider a pullback square in $\mathcal{C}$:

$$\begin{tikzcd} \Delta \arrow[r, "q"] \arrow[d, "h"'] & \Gamma' \arrow[d, "g"] \\ \Gamma \arrow[r, "f"'] & \Theta \end{tikzcd}$$

where $f \circ h = g \circ q$. For a functor $F : \mathcal{E}_{\Gamma'} \to \mathcal{E}_\Gamma$ with a left or right adjoint $G : \mathcal{E}_\Gamma \to \mathcal{E}_{\Gamma'}$, the *Beck-Chevalley condition* holds if the canonical natural transformation induced by the pullback, $h^* \circ G \to q^* \circ F$ (for right adjoints) or $q^* \circ F \to h^* \circ G$ (for left adjoints), is an isomorphism.

Definition 41 (Dependent Sum)**.** In a comprehension category with fibration $p : \mathcal{E} \to \mathcal{C}$, a *dependent sum* for a type $\sigma \in \mathcal{E}_\Gamma$ is a functor $\Sigma_\sigma : \mathcal{E}_{\Gamma.\sigma} \to \mathcal{E}_\Gamma$, left adjoint to the substitution functor $p_\sigma^* : \mathcal{E}_\Gamma \to \mathcal{E}_{\Gamma.\sigma}$, such that for all morphisms $f : \Delta \to \Gamma$ in $\mathcal{C}$, the Beck-Chevalley condition holds, i.e., the canonical natural transformation $\Sigma_{f^*\sigma} \circ q(f, \sigma)^* \cong f^* \circ \Sigma_\sigma$ is an isomorphism.

Definition 42 (Dependent Product)**.** In a comprehension category with fibration $p : \mathcal{E} \to \mathcal{C}$, a *dependent product* for a type $\sigma \in \mathcal{E}_\Gamma$ is a functor $\Pi_\sigma : \mathcal{E}_{\Gamma.\sigma} \to \mathcal{E}_\Gamma$, right adjoint to the substitution functor $p_\sigma^* : \mathcal{E}_\Gamma \to \mathcal{E}_{\Gamma.\sigma}$, such that for all morphisms $f : \Delta \to \Gamma$ in $\mathcal{C}$, the Beck-Chevalley condition holds, i.e., the canonical natural transformation $f^* \circ \Pi_\sigma \cong \Pi_{f^*\sigma} \circ q(f, \sigma)^*$ is an isomorphism.

Definition 43 (Identity Type)**.** In a split comprehension category with fibration $p : \mathcal{E} \to \mathcal{C}$, an *identity type* for a type $\sigma \in \mathcal{E}_\Gamma$ consists of:

- A type $\mathrm{Id}_\sigma \in \mathcal{E}_{\Gamma.\sigma.\sigma}$, where $\Gamma.\sigma.\sigma = p_\sigma^* \sigma$.
- A morphism $r_\sigma : \Gamma.\sigma \to I_\sigma$, where $I_\sigma = \Gamma.\sigma.\sigma.\mathrm{Id}_\sigma$, such that $p_{\mathrm{Id}_\sigma} \circ r_\sigma = \mathrm{id}$.
- For any commutative square $\langle f, M \rangle : \Delta \to \Gamma.\sigma$, $\langle g, N \rangle : \Delta.\tau \to \Gamma.\sigma.\sigma$, a diagonal lifting $h : I_\sigma \to \Delta.\tau$ making both triangles commute.

All data must be stable under substitutions.

Definition 44 (Category with Attributes)**.** A *category with attributes* is a full split comprehension category, where the comprehension functor $\{-\} : \mathcal{E} \to \mathcal{C}^\to$ is fully faithful, and types over $\Gamma \in \mathcal{C}$ are determined by a functor $\mathrm{Ty} : \mathcal{C}^{\mathrm{op}} \to \mathbf{Set}$.

Definition 45 (Display Map Category)**.** A *display map category* is a comprehension category where the comprehension functor $\{-\} : \mathcal{E} \to \mathcal{C}^\to$ is the inclusion of a full subcategory of $\mathcal{C}^\to$, and all morphisms in the image are display maps.

Definition 46 (Contextual Category)**.** A *contextual category* is a category with attributes equipped with:

- A terminal object $\bullet \in \mathcal{C}$.
- A length function $\ell : \mathrm{obj}(\mathcal{C}) \to \mathbb{N}$ such that $\ell(\bullet) = 0$, and for any type $\sigma \in \mathcal{E}_\Gamma$, $\ell(\Gamma.\sigma) = \ell(\Gamma) + 1$.
- For any non-empty context Γ, a unique context Δ (the father) and type $\sigma \in \mathcal{E}_\Delta$ such that $\Gamma = \Delta.\sigma$.

Definition 47 (Weakening Morphism)**.** In a comprehension category, a *weakening morphism* is defined inductively:

- A display map $p_\sigma : \Gamma.\sigma \to \Gamma$ is a weakening morphism.
- If $f : \Delta \to \Gamma$ is a weakening morphism and $\sigma \in \mathcal{E}_\Gamma$, then $q(f, \sigma) : \Delta.f^*\sigma \to \Gamma.\sigma$ is a weakening morphism.

Definition 48 (Variable)**.** In a comprehension category, for a type $\sigma \in \mathcal{E}_\Gamma$, the *variable* of type σ is the unique term $\nu_\sigma : \Gamma.\sigma \to \mathsf{p}_\sigma^*\sigma$ such that $\mathsf{p}_{\mathsf{p}_\sigma^*\sigma} \circ \nu_\sigma = \mathrm{id}$.

Definition 49 (Universe)**.** In a split comprehension category with terminal object $\bullet \in \mathcal{C}$, a *universe* consists of:

- A type $\mathcal{U} \in \mathcal{E}_\bullet$, the context $\bullet.\mathcal{U}$ also denoted $\mathcal{U}$.
- A type $\mathrm{El} \in \mathcal{E}_\mathcal{U}$, with context $\mathcal{U}.\mathrm{El}$ denoted $\widetilde{\mathcal{U}}$.

For a morphism $\mathsf{f} : \Gamma \to \mathcal{U}$, the type $\sigma_\mathsf{f} \in \mathcal{E}_\Gamma$ is the substitution of El along f.

6.2 Theorems

Theorem 10 (Split Fibration Cleavage)**.** Every split fibration $\mathsf{p} : \mathcal{E} \to \mathcal{C}$ has a cleavage such that the reindexing functors $\mathsf{f}^* : \mathcal{E}_{\mathsf{p}(\mathsf{e})} \to \mathcal{E}_{\mathsf{c}'}$ satisfy $(\mathsf{g} \circ \mathsf{f})^* = \mathsf{f}^* \circ \mathsf{g}^*$, and every Grothendieck fibration with such a cleavage is a split fibration.

Theorem 11 (Framework Equivalence)**.** Every comprehension category can be equipped with a structure equivalent to a category with families (CwF), category with representable maps (CwR), or Awodey's natural model under the existence of terminal objects.

6.3 Example MLTT-75 Model

We model MLTT-75 using a comprehension category, interpreting contexts, types, and terms via the fibration and comprehension functor.

Definition 50 (MLTT-75 Comprehension Model)**.** Given a comprehension category with categories $\mathcal{E}$, $\mathcal{B}$, a split fibration $p : \mathcal{E} \to \mathcal{B}$, and a comprehension functor $\{-\}$, the model of MLTT-75 is defined as:

- *Contexts*: Objects $\Gamma \in \mathcal{B}.\mathrm{Ob}$.
- *Types*: Pairs $(A, p_A : p(A) = \Gamma)$, representing a type A in context Γ.
- *Terms*: Morphisms $t : \Gamma \to A$ in $\mathcal{E}$ such that $p(t) = \mathrm{id}_\Gamma$, i.e., sections.
- *Context extension*: For $\Gamma \vdash A$, the context $\Gamma, x : A$ is $\{A\}$, the domain of the comprehension.
- *Type formers*: Π-types via fiber exponentials, Σ-types via composition, Id-types via diagonals.

6.4 Π-Types

For $\Gamma \vdash A$: Type and $\Gamma, x : A \vdash B$: Type, the Π-type $\Pi_{x:A} B$ is formed using exponentials in the fiber category $\mathcal{E}_\Gamma$.

The constructor λ forms terms of $\Pi_{x:A} B$. The pullback diagram is:

$$\begin{tikzcd}
\Gamma \times A \arrow[r, "\lambda"] \arrow[d] & B \arrow[d] \\
\Gamma \arrow[r, "\Pi"'] & \Pi_A B
\end{tikzcd}$$

6.5 Σ-Types

For $\Gamma \vdash A$: Type and $\Gamma, x : A \vdash B$: Type, the Σ-type $\Sigma_{x:A} B$ is formed via composition in the fibration.

The constructor pair forms terms of $\Sigma_{x:A} B$. The pullback diagram is:

$$\begin{tikzcd}
\Sigma_A B \arrow[r, "\text{pair}"] \arrow[d] & B \arrow[d] \\
\Gamma \arrow[r, "\Sigma"'] & \Gamma \times A
\end{tikzcd}$$

6.6 Id-Types

For $\Gamma \vdash A$: Type and $a, b : A$, the identity type $\mathrm{Id}_A(a, b)$ is formed using the diagonal map in the fibration.

The constructor refl forms terms of $\mathrm{Id}_A(a, a)$. The pullback diagram is:

$$\begin{tikzcd}
\mathrm{Id}_A(a,b) \arrow[r, "\mathrm{refl}"] \arrow[d] & A \arrow[d, "\Delta_A"] \\
\Gamma \arrow[r, "\mathrm{Id}"'] & A \times_\Gamma A
\end{tikzcd}$$

```
structure ComprehensionCategory (E B : Category) where
  fib : SplitFibration E B
  terminal : ∃ (T : B.Ob), ∀ (A : B.Ob), ∃! (t : B.Hom A T), True
  comp_functor : ∀ (A : E.Ob), Σ (Γ' : B.Ob) (π : B.Hom Γ' (fib.functor.obj A))
  comp_adj : ∀ (Γ : B.Ob) (A : E.Ob) (pA : fib.functor.obj A =Γ
) (σ : B.Hom Δ Γ),
    ∃ (A' : E.Ob) (pA' : fib.functor.obj A' =Δ) (f : E.Hom A' A),
    fib.functor.map f = σ
  pullback : ∀ {A B C : B.Ob} (f : B.Hom A B) (g : B.Hom C B),
    ∃ (P : B.Ob) (h1 : B.Hom P A) (h2 : B.Hom P C),
    B.comp f h1 = B.comp g h2 ∧
    ∀ (Q : B.Ob) (q1 : B.Hom Q A) (q2 : B.Hom Q C),
    B.comp f q1 = B.comp g q2 → ∃ (u : B.Hom Q P), B.comp h1 u =
q1 ∧ B.comp h2 u =q2
  pi : ∀ (Γ : B.Ob) (A e : E.Ob) (f : E.Hom A e) (pA pe : fib.functor.obj A =Γ
 ∧ fib.functor.obj e =Γ),
    ∃ (Pi : E.Ob) (pi : E.Hom Pi Γ), fib.functor.obj Pi =Γ ∧
    ∀ (C : E.Ob) (g : E.Hom C A) (pC : fib.functor.obj C =Γ),
    ∃ (h : E.Hom C Pi), E.comp pi h =E.comp f g
  sigma : ∀ (Γ : B.Ob) (A e : E.Ob) (f : E.Hom A e) (pA pe : fib.functor.obj A =Γ
 ∧ fib.functor.obj e =Γ),
    ∃ (Sigma : E.Ob) (sigma : E.Hom Sigma Γ), fib.functor.obj Sigma =Γ
  id : ∀ (Γ : B.Ob) (A : E.Ob) (pA : fib.functor.obj A =Γ),
    ∃ (Id : E.Ob) (id : E.Hom Id A), fib.functor.obj Id =Γ

  Context : Type
  Context := B.Ob

  Type : Context → Type
  Type Γ := Σ (A : E.Ob), fib.functor.obj A =Γ

  Term : ∀ (Γ : Context), Type Γ → Type
  Term Γ (A, pA) := Σ (t : E.Hom Γ A), fib.functor.map t =B.id

  ContextExt : ∀ (Γ : Context), Type Γ → Context
  ContextExt Γ (A, pA) := (comp_functor A).1

  PiType : ∀ (Γ : Context) (A : Type Γ), Type (ContextExt Γ A) → Type Γ
  PiType Γ (A, pA) (e, pe) := let res := pi Γ A e E.id (pA, pe) in (res.1, res.2.1)

  SigmaType : ∀ (Γ : Context) (A : Type Γ), Type (ContextExt Γ A) → Type Γ
  SigmaType Γ (A, pA) (e, pe) := let res := sigma Γ A e E.id (pA, pe) in (res.1, res.2.1)

  IdType : ∀ (Γ : Context) (A : Type Γ) (a b : Term Γ A), Type Γ
  IdType Γ (A, pA) (a, pa) (b, pb) := let res := id Γ A pA in (res.1, res.2.1)
```

6.7 Conclusion

The Lean 4 formalization provides a minimal, dependency-free model of MLTT-75 using a comprehension category, explicitly capturing type dependency and context extension via a Grothendieck fibration and comprehension functor. This contrasts with the representable maps approach, aligning more closely with traditional fibration-based models. The pullback diagrams, styled after Awodey, clarify the categorical constructions. Future work includes verifying the model with concrete examples and extending it to homotopy type theory.

Література

[1] Jacobs, B., "Comprehension categories and the semantics of type dependency," TCS, 1993.

[2] Hofmann, M., "On the interpretation of type theory in locally cartesian closed categories," LFCS, 1997.

[3] Cartmell, J., "Generalised algebraic theories and contextual categories," APAL, 1986.

[4] Voevodsky, V., "Notes on type systems," 2014.

[5] Awodey, S., "Natural models of homotopy type theory," MSCS, 2018.

Issue XXX: Quillen Model Categories

Namdak Tonpa

5 травня 2025 р.

Анотація

Ця стаття є оглядом теорії модельних категорій, започаткованої Деніелом Квілленом у його новаторській праці 1967 року “Гомотопічна алгебра”. Ми розглядаємо історичний контекст, основні аксіоми та застосування модельних категорій у топології та суміжних галузях, зокрема у доведенні кон'єктур Мілнора та Блоха-Като Воєводським. Також обговорюються сучасні узагальнення, такі як інфініті-категорії та модельні структури на симпліційних і кубічних множинах, з акцентом на їхню релевантність у математиці та теоретичній інформатиці.

7 Model Categories

PhD Деніела Квілена була присвячена диференціальним рівнянням, але відразу після цього він перевівся в MIT і почав працювати в алгебраїній топології, під впливом Дена Кана. Через три роки він видає Шпрінгеровські лекції з математики "Гомотопічна алгебра"[1], яка назавжди трансформувала алгебраїчну топологію від вивчення топологічних просторів з точністю до гомотопій до загального інструменту, що застосовується в інших галузях математики.

Модельні категорії вперше були успішно застосовані Воєводським на підтвердження кон'юнктури Мілнора [2] (для 2) і потім мотивної кон'юнктури Блоха-Като [3] (для n). Для доказу для 2 була побудована зручна гомотопічна стабільна категорія узагальнених схем. Інфініті категорії Джояля, досить добре досліджені Лур'є [4], є прямим узагальненням модельних категорій.

7.1 Означення модельних категорій

До часу, коли Квіллен написав "Гомотопічну алгебру вже було деяке уявлення про те, як має виглядати теорія гомотопій. Починаємо ми з категорії $\mathcal{C}$ та колекції морфізмів W – слабкими еквівалентностями. Завдання вправи інвертувати W морфізму щоб отримати гомотопічну категорію. Хотілося б мати спосіб, щоб можна було конструтувати похідні функтори. Для топологічного простору X, його апроксимації LX і слабкої еквівалентності $\mathsf{LX} \rightarrow \mathsf{X}$ це означає, що ми повинні замінити X на LX. Це аналогічно до заміни модуля або ланцюгового комплексу на проективну резольвенту. Подвійним чином, для симпліційної множини K, Кан комплексу RK, і слабкої еквівалентності $\mathsf{K} \rightarrow \mathsf{RK}$ ми повинні замінити K на RK. У цьому випадку це аналогічно до заміни ланцюгового комплексу ін'єктивною резольвентою.

```
modelStructure (C: category): U
  = (fibrations: fib C)
  * (cofibrations: cofib C)
  * (weakEqivalences: weak C)
  * unit
```

Таким чином Квілену потрібно було окрім поняття слабкої еквівалентності ще й поняття розшарованого (RK) та корозшарованого (LX) об'єктів. Ключовий інстайт з топології тут наступний, в неабелевих ситуаціях об'єкти не надають достатньої структури поняття точної послідовності. Тому стало зрозуміло, що для відновлення структури необхідно ще два класи морфізмів: розшарування та корозшарування на додаток до слабких еквівалентностей, яким ми повинні інвертувати для розбудови гомотопічної категорії. Природно ці три колекції морфізом повинні задовольняти набору умов, званих аксіомами модельних категорій: 1) наявність малих лімітів і колимітів; 2) правило 3-для-2; 3) правило ректрактів; 4) правило підйому; 5) правило факторизації.

Definition 51. Модельна категорія — це категорія $\mathcal{C}$, оснащена трьома класами морфізмів: 1) $\mathtt{fib}(\mathcal{C})$ — розшарування; 2) $\mathtt{cof}(\mathcal{C})$ — корозшарування; 3) $\mathsf{W}(\mathcal{C})$ — слабкі еквівалентності, які задовольняють аксіоми, наведені вище.

Цікавою властивістю модельних категорій є те, що дуальні до них категорії
перевертають розшарування та корозшарування, таким чином реалізуючи дуальність Екманна-Хілтона. Розшарування та корозшарування пов'язані, тому взаємовизначені. Корозшарування є морфізми, що мають властивість лівого гомотопічного підйому по відношенню до ациклічних розшарування і розшарування є морфизми, що мають властивість правого гомотопічного підйому по відношенню до ациклічних кофібрацій.

7.2 Застосування в топології

Основним застосуванням модельних категорій у роботі Квілена було присвячено категоріям топологічних просторів. Для топологічних просторів існує дві модельні категорії: Квілена (1967) та Строма (1972). Перша як розшарований використовує розшарування Серра, а як корозшаровування морфізму які мають лівий гомотопічний підйом по відношенню до ациклічних розшарування Серра, еквівалентно це ретракти відповідних CW-комплексів, а як слабка еквівалентність виступає слабка гомотопічна.

Друга модель Строма як розшарування використовуються розшарування Гуревича, як корозшарування стандартні корозшаровування, і як слабка еквівалентність — сильна гомотопічна еквівалентність.

```
quillen67
    : modelStructure Top
    = ( serreFibrations ,
        retractsCW ,
        weakHomotopyEquivalence )

strom1972
    : modelStructure Top
    = ( hurewiczFibrations ,
        cofibrations ,
        strongHomotopyEquivalence )
```

7.3 Модельні категорії для множин

Найпростіші модельні категорії можна побудувати для категорії множин, де кількість ізоморфних моделей зростає до дев'яти. Наведемо деякі конфігурації модельних категорій для категорії множин:

```
set0: modelStructure Set = (all,all,bijections)
set1: modelStructure Set = (bijections,all,all)
set2: modelStructure Set = (all,bijections,all)
set3: modelStructure Set = (surjections,injections,all)
set4: modelStructure Set = (injections,surjections,all)
```

7.4 Застосування в алгебраїчній геометрії

Модельні категорії вперше були успішно застосовані Воєводським на підтвердження кон'юнктури Мілнора [2] (для 2) і потім мотивної кон'юнктури Блоха-Като [3] (для n). Для доказу для 2 була побудована зручна гомотопічна стабільна категорія узагальнених схем.

7.5 Інфініті-категорії та сучасні узагальнення

Для переходу від модельних категорій до (∞,1)-категорій необхідно перейти до категорій де морфізми утворюють не множини, а симпліційні множини. Потім можна переходити до локалізації.

```
simplicial
    : modelStructure sSet
    = ( kanComplexes ,
        monos ,
        simplicialBijections )
```

Але для нас, для програмістів найцікавішими є модельні категорії симпліціальних множин та модельні категорії кубічних множин, саме в цьому сеттингу написано CCHM пейпер 2016 року, де показано модельну структуру категорії кубічних множин [5].

```
cubical
    : modelStructure cSet
    = ( kanComplexes ,
        monos ,
        geometricRealisation )
```

де $\mathsf{cSet} = [\square^{\mathrm{op}}, \mathsf{Set}]$, а $\square$ — категорія збагачена структурою алгебри де Моргана.

7.6 Висновки

Модельні категорії, запроваджені Квілленом, стали фундаментальним інструментом у сучасній математиці, забезпечуючи гнучкий фреймворк для роботи з гомотопіями в різних категоріях. Їхні застосування варіюються від топології до алгебраїчної геометрії та теоретичної інформатики, а узагальнення, такі як інфініті-категорії, відкривають нові горизонти для досліджень. Подальший розвиток теорії, ймовірно, буде пов'язаний із застосуванням модельних структур у комп'ютерних науках, зокрема в семантиці мов програмування та гомотопічній теорії типів.

Література

[1] Д. Квіллен, *Гомотопічна алгебра*, Springer Lecture Notes in Mathematics, 1967.

[2] В. Воєводський, *The Milnor conjecture*, 1996.

[3] В. Воєводський, *Bloch-Kato conjecture for* $\mathbb{Z}/2$ *coefficients and algebraic Morava K-theories*, 2003.

[4] Дж. Лур'є, *Higher Topos Theory*, Princeton University Press, 2009.

[5] Е. Кавалло, А. Мьортберг, А. Сван, *Model structure on cubical sets*, 2019.

[6] А. Стром, *Note on cofibrations II*, Mathematische Zeitschrift, 1972.

[7] Дж. Джардін, *Model structure on cubical sets*, 2002.

[8] К. Капулкін, П. Ламсдейн, В. Воєводський, *Univalence in Simplicial Sets*, 2012.

[9] Н. Гамбіно, К. Саттлер, К. Шуміло, *The constructive Kan-Quillen model structure: two new proofs*, 2019.

[10] Д. Кан, А. Бусфілд, *Homotopy Limits, Completions and Localizations*, 1972.

[11] Ф. Морель, В. Воєводський, *A1-homotopy theory of schemes*, 1999.

Issue XXXI: Abelian Categories

Namdak Tonpa

5 травня 2025 р.

Анотація

Ця стаття є оглядом абелевих категорій, введених Александром Гротендіком у 1957 році, як фундаментального інструменту гомологічної алгебри, алгебраїчної геометрії, теорії представлень, топологічної квантової теорії поля та теорії категорій. Ми розглядаємо формальне означення абелевих категорій, їхню роль у побудові похідних категорій і функторів, а також ключові застосування в різних галузях математики та фізики.

8 Abelian Categories

Абелеві категорії, вперше введені Александром Гротендіком у його статті 1957 року «Sur quelques points d'algèbre homologique» [1], стали основою для уніфікації гомологічної алгебри в різних математичних дисциплінах, таких як алгебраїчна геометрія, алгебраїчна топологія та теорія представлень. Вони забезпечують природне середовище для вивчення гомологій, когомологій, похідних категорій і функторів, що мають широке застосування в математиці та математичній фізиці.

8.1 Означення абелевих категорій

Абелеві категорії — це збагачене поняття категорії Сандерса-Маклейна поняттями нульового об'єкту, що одночасно ініціальний та термінальний, властивостями існування всіх добутків та кодобутків, ядер та коядер, а також, що всі мономорфізми і епіморфізми є ядрами і коядрами відповідно (тобто нормальними).

Формально, абелева категорія визначається наступним чином:

```
def isAbelian (C: precategory): U₁
 := Σ (zero:     hasZeroObject C)
      (prod:     hasAllProducts C)
      (coprod:   hasAllCoproducts C)
      (ker:      hasAllKernels C zero)
      (coker:    hasAllCokernels C zero)
      (monicsAreKernels:
         Π (A S: C.C.ob) (k: C.C.hom S A),
```

```
    Σ (B: C.C.ob) (f: C.C.hom A B),
    isKernel C zero A B S f k)
  (epicsAreCoKernels:
    Π (B S: C.C.ob) (k: C.C.hom B S),
    Σ (A: C.C.ob) (f: C.C.hom A B),
    isCokernel C zero A B S f k), U
```

Ця сигнатура включає: 1) існування нульового об'єкта; 2) існування всіх добутків; 3) існування всіх кодобутків; 4) існування всіх ядер; 5) існування всіх коядер; 6) властивість, що кожен мономорфізм є ядром; 7) властивість, що кожен епіморфізм є коядром.

8.2 Деталізоване формальне означення

Для чіткості наведемо ключові компоненти абелевої категорії в сучасному формалізмі, наприклад, у кубічній Агді, як описано в магістерській роботі Девіда Еліндера 2021 року [2]:

```
module abelian where
import lib/mathematics/categories/category
import lib/mathematics/homotopy/truncation

def zeroObject(C: precategory) (X: C.C.ob): U₁
 := Σ (bot: isInitial C X) (top: isTerminal C X), U

def hasZeroObject (C: precategory) : U₁
 := Σ (ob: C.C.ob) (zero: zeroObject C ob), unit

def hasAllProducts (C: precategory) : U₁
 := Σ (product: C.C.ob -> C.C.ob -> C.C.ob)
      (π₁: Π (A B : C.C.ob), C.C.hom (product A B) A)
      (π₂: Π (A B : C.C.ob), C.C.hom (product A B) B), U

def hasAllCoproducts (C: precategory) : U₁
 := Σ (coproduct: C.C.ob -> C.C.ob -> C.C.ob)
      (σ₁: Π (A B : C.C.ob), C.C.hom A (coproduct A B))
      (σ₂: Π (A B : C.C.ob), C.C.hom B (coproduct A B)), U

def isMonic (P: precategory) (Y Z : P.C.ob) (f : P.C.hom Y Z) : U
 := Π (X : P.C.ob) (g1 g2 : P.C.hom X Y),
    Path (P.C.hom X Z) (P.P.∘ X Y Z g1 f) (P.P.∘ X Y Z g2 f)
 -> Path (P.C.hom X Y) g1 g2

def isEpic (P : precategory) (X Y : P.C.ob) (f : P.C.hom X Y) : U
 := Π (Z : P.C.ob) (g1 g2 : P.C.hom Y Z),
    Path (P.C.hom X Z) (P.P.∘ X Y Z f g1) (P.P.∘ X Y Z f g2)
 -> Path (P.C.hom Y Z) g1 g2

def kernel (C: precategory) (zero: hasZeroObject C)
    (A B S: C.C.ob) (f: C.C.hom A B) : U₁
 := Σ (k: C.C.hom S A) (monic: isMonic C S A k), unit

def cokernel (C: precategory) (zero: hasZeroObject C)
    (A B S: C.C.ob) (f: C.C.hom A B)  : U₁
 := Σ (k: C.C.hom B S) (epic: isEpic C B S k), unit
```

```
def isKernel (C: precategory) (zero: hasZeroObject C)
    (A B S: C.C.ob) (f: C.C.hom A B) (k: C.C.hom S A) : U₁
 := Σ (ker: kernel C zero A B S f), Path (C.C.hom S A) ker.k k

def isCokernel (C: precategory) (zero: hasZeroObject C)
    (A B S: C.C.ob) (f: C.C.hom A B) (k: C.C.hom B S) : U₁
 := Σ
(coker: cokernel C zero A B S f), Path (C.C.hom B S) coker.k k

def hasKernel (C: precategory) (zero: hasZeroObject C)
    (A B: C.C.ob) (f: C.C.hom A B) : U₁
 := ‖_‖₋₁ (Σ (monic: isMonic C A B f), unit)

def hasCokernel (C: precategory) (zero: hasZeroObject C)
    (A B: C.C.ob) (f: C.C.hom A B) : U₁
 := ‖_‖₋₁ (Σ (epic: isEpic C A B f), unit)

def hasAllKernels (C : precategory) (zero: hasZeroObject C) : U₁
 := Σ (A B : C.C.ob) (f : C.C.hom A B), hasKernel C zero A B f

def hasAllCokernels (C : precategory) (zero: hasZeroObject C) : U₁
 := Σ (A B : C.C.ob) (f : C.C.hom A B), hasCokernel C zero A B f
```

Ці означення уточнюють поняття нульового об'єкта, добутків, кодобутків, мономорфізмів, епіморфізмів, ядер і коядер, необхідних для абелевих категорій.

8.3 Мотивація та застосування

Абелеві категорії мають численні застосування в різних галузях математики та фізики. Ось п'ять ключових напрямів:

1) Гомологічна алгебра: абелеві категорії забезпечують основу для гомологічної алгебри, яка вивчає властивості груп гомології та когомології. Теорія похідних функторів, фундаментальний інструмент гомологічної алгебри, базується на понятті абелевої категорії.

2) Алгебраїчна геометрія: абелеві категорії використовуються для вивчення когомологій пучка, що є потужним інструментом для розуміння геометричних властивостей алгебраїчних многовидів. Зокрема, категорія пучків абелевих груп на топологічному просторі є абелевою категорією.

3) Теорія представлень: абелеві категорії виникають у теорії представлень, яка досліджує алгебраїчні структури, пов'язані з симетріями. Наприклад, категорія модулів над кільцем є абелевою категорією.

4) Топологічна квантова теорія поля: абелеві категорії відіграють центральну роль у топологічній квантовій теорії поля, де вони виникають як категорії граничних умов для певних типів теорій топологічного поля.

5) Теорія категорій: абелеві категорії є важливим об'єктом дослідження в теорії категорій, зокрема для вивчення адитивних функторів. Рекомендується робота Бакура і Деляну «Вступ в теорію категорій та функторів» [3] для поглибленого ознайомлення.

8.4 Похідні категорії та функтори

Абелеві категорії забезпечують природну основу для гомологічної алгебри, яка є розділом алгебри, що має справу з алгебраїчними властивостями груп гомологій та когомологій. Зокрема, абелеві категорії створюють сеттінг, де можна визначити поняття похідних категорій і похідних функторів.

Основна ідея похідних категорій полягає в тому, щоб ввести нову категорію, яка побудована з абелевої категорії шляхом «інвертування» певних морфізмів, майже так само, як будується поле часток на області цілісності. Похідна категорія абелевої категорії фіксує «правильне» поняття гомологічних і когомологічних груп і забезпечує потужний інструмент для вивчення алгебраїчних властивостей цих груп.

Похідні функтори є фундаментальним інструментом гомологічної алгебри, і їх можна визначити за допомогою концепції похідної категорії. Основна ідея похідних функторів полягає в тому, щоб взяти функтор, який визначено в абелевій категорії, і «підняти» його до функтора, який визначений у похідній категорії. Похідний функтор потім використовується для обчислення вищих груп гомології та когомології об'єктів в абелевій категорії.

Використання похідних категорій і функторів зробило революцію у вивченні гомологічної алгебри, і це призвело до багатьох важливих застосувань в алгебраїчній геометрії, топології та математичній фізиці. Наприклад, похідні категорії використовувалися для доведення фундаментальних результатів алгебраїчної геометрії, таких як знаменита теорема Гротендіка-Рімана-Роха. Вони також використовувалися для вивчення дзеркальної симетрії в теорії суперструн.

8.5 Висновки

Абелеві категорії, введені Гротендіком, є фундаментальним інструментом сучасної математики, що забезпечує уніфікований підхід до гомологічної алгебри, алгебраїчної геометрії, теорії представлень, топологічної квантової теорії поля та теорії категорій. Їхня роль у побудові похідних категорій і функторів відкрила нові можливості для вивчення гомологій і когомологій, а також їхніх застосувань у математиці та фізиці. Подальший розвиток теорії абелевих категорій, зокрема в контексті унівалентної теорії типів, як показано в роботі Еліндера [2], обіцяє нові перспективи для формальної математики та комп'ютерних наук.

Література

[1] А. Гротендік, *Sur quelques points d'algèbre homologique*, 1957.

[2] Д. Еліндер, *Дослідження абелевих категорій і унівалентної теорії типів*, магістерська робота, 2021.

[3] І. Бакур, А. Деляну, *Вступ в теорію категорій та функторів*.

Issue XXXII: Grothendieck Yogas

Namdak Tonpa

5 травня 2025 р.

Анотація

Ця стаття присвячена огляду функторіальних йог Гротендіка, зокрема шести функторів, когезивних топосів та їхньої ролі в теорії похідних категорій. Ми розглядаємо основні концепції, такі як когомології та їх узагальнення, а також зв'язок із сучасною алгебраїчною геометрією та мотивною гомотопічною теорією. Стаття базується на сучасних джерелах, зокрема на лекціях Мартіна Галлауера про шестифункторний формалізм.

9 Grothendieck Yogas

Шестифункторний формалізм Гротендіка є одним із ключових інструментів сучасної алгебраїчної геометрії, що дозволяє узагальнити класичні когомологічні теорії та застосовувати їх у різних контекстах, від топології до мотивної гомотопічної теорії. Цей формалізм, розроблений Александром Гротендіком, включає шість основних операцій (функторів), які діють на категорії пучків або їх узагальнень, забезпечуючи багатий набір інструментів для вивчення геометричних об'єктів.

У цій статті ми зосередимося на трьох основних аспектах:

1. **Чому шестифункторний формалізм важливий?** Він узагальнює когомології, дозволяючи працювати з відносною точкою зору та застосовувати їх у складних геометричних контекстах.

2. **Що таке шестифункторний формалізм?** Ми розглянемо основні функтори та їх властивості, такі як локалізація, дуальність та відносна чистота.

3. **Як його конструюють?** Ми обговоримо методи побудови формалізму, зокрема через системи коефіцієнтів та когезивні топоси.

9.1 Узагальнення когомологій

Когомології є фундаментальним інструментом у топології та алгебраїчній геометрії. Наприклад, для топологічного простору X числа Бетті $b_n(X)$ вимірюють кількість n-вимірних дірок, але гомології $H_n(X)$ є багатшим інваріантом, оскільки містять інформацію про цикли та границі.

Example 1. Для різноманіття X над скінченним полем $k = \mathbb{F}_q$, ζ-функція $\zeta_X(T)$ кодує кількість раціональних точок. Гротендік показав, що властивості цієї функції випливають із ℓ-адичних когомологій $H^*(X_{\bar{k}}; \mathbb{Q}_\ell)$, які, у свою чергу, походять із похідної категорії $D^b_c(X_{\bar{k}}; \mathbb{Q}_\ell)$.

Шестифункторний формалізм узагальнює ці ідеї, дозволяючи працювати з категоріями рівня, які керують поведінкою когомологій.

9.2 Відносна точка зору

Гротендік наголошував на важливості відносної точки зору, де замість окремих об'єктів (наприклад, схем) розглядаються морфізми між ними. Це дозволяє вивчати когомології не ізольовано, а разом із дією морфізмів:

$$f^* : H^*(Y) \to H^*(X).$$

Remark 4. Навіть для однієї схеми X часто необхідно розглядати когомології пов'язаних об'єктів, наприклад, при індукції за розмірністю або розбитті на простіші частини.

9.3 Основні функтори

Шестифункторний формалізм складається з шести основних функторів, які діють на категорії пучків (або їх узагальнень, таких як похідні категорії):

- f^*: обернений образ (pull-back),
- f_*: прямий образ (push-forward),
- $f_!$: прямий образ із компактною підтримкою,
- $f^!$: винятковий обернений образ,
- $\otimes$: тензорний добуток,
- $\underline{\mathrm{Hom}}$: внутрішній гом.

Ці функтори пов'язані між собою ад'юнкціями:

$$f^* \dashv f_*, \quad f_! \dashv f^!.$$

Definition 52. Для простору X (наприклад, топологічного простору або схеми) категорія $C(X)$ є замкненою тензорною триангулятивною категорією, оснащеною операціями $\otimes$ та $\underline{\mathrm{Hom}}$. Для морфізму $f : X \to Y$ визначено ад'юнкції $f^* \dashv f_*$, $f_! \dashv f^!$, а також природну трансформацію $f_! \to f_*$.

9.4 Когезивні топоси

Когезивні топоси є природним контекстом для шестифункторного формалізму, оскільки вони забезпечують категоріальну структуру, яка підтримує геометричні та когомологічні операції. Топос $\mathcal{E}$ називається когезивним, якщо він має набір ад'юнктних функторів, що моделюють геометричні трансформації.

Example 2. Категорія пучків $\mathrm{Sh}(X)$ на топологічному просторі X є когезивним топосом, де f^* та f_* відповідають оберненим і прямим образам.

У контексті алгебраїчної геометрії когезивні топоси часто виникають як категорії пучків на схемах або стеках, оснащені додатковими структурами, такими як стабільні ∞-категорії.

9.5 Роль абелевих категорій

Абелеві категорії відіграють фундаментальну роль у шестифункторному формалізмі, оскільки вони є основою для побудови похідних категорій, які використовуються для опису пучків та їх когомологій. Абелева категорія — це категорія, в якій морфізми мають ядра та кокернали, а кожна монада та епіморфізм є нормальними. Типовим прикладом є категорія абелевих пучків $\mathrm{Ab}(X)$ на топологічному просторі X або категорія когерентних пучків на схемі.

У шестифункторному формалізмі абелеві категорії, такі як $\mathrm{Sh}(X)$, слугують вихідним пунктом для визначення функторів f^* та f_*. Наприклад, для неперервного відображення $f : X \to Y$, функтор прямого образу $f_*\mathcal{F}$ визначається через секції $\Gamma(f^{-1}(U), \mathcal{F})$, де $\mathcal{F} \in \mathrm{Sh}(X)$, а f^* є його лівою ад'юнктою. Однак, щоб врахувати гомотопічні властивості та виняткову функторіальність $(f_!, f^!)$, необхідно перейти до похідних категорій $D(\mathrm{Sh}(X))$, які будуються з абелевих категорій шляхом локалізації за квазіізоморфізмами.

Remark 5. Абелеві категорії забезпечують строгу алгебраїчну структуру, але їх обмеження (наприклад, відсутність природної триангулятивної структури) роблять похідні категорії більш придатними для шестифункторного формалізму, особливо в контексті ℓ-адичних або мотивних пучків.

9.6 Похідні категорії

Похідна категорія $D(\mathrm{Sh}(X))$ пучків на просторі X є природним узагальненням категорії пучків, що враховує гомотопічні властивості. Вона дозволяє працювати з похідними функторами, такими як:

$$R^n f_*(\mathcal{F}) \simeq H^n(X; \mathcal{F}), \quad R^n f_!(\mathcal{F}) \simeq H^n_c(X; \mathcal{F}).$$

Example 3. Для ℓ-адичних пучків на схемі X похідна категорія $D^b_c(X; \mathbb{Q}_\ell)$ є основою для ℓ-адичних когомологій, які використовувалися для доведення гіпотез Вейля.

9.7 Конструкція шестифункторного формалізму

Конструкція шестифункторного формалізму є складним завданням, яке часто потребує значних зусиль. Одним із ключових викликів є побудова виняткової функторіальності $(f_!, f^!)$.

Remark 6. За Делінем, для морфізму $f : X \to Y$ можна використати компактифікацію Нагати, щоб розкласти f на відкрите вкладення j та власний морфізм p:

$$f = p \circ j, \quad f_! := p_* j_\sharp.$$

Ця
конструкція вимагає доведення незалежності від вибору факторизації та існування правої ад'юнкти $f^!$.

9.8 Застосування в мотивній гомотопічній теорії

Мотивна гомотопічна теорія, розроблена Морелем і Воєводським, використовує шестифункторний формалізм для узагальнення класичних гомотопічних теорій на алгебраїчні схеми. Категорія SH(X) стабільних мотивних гомотопічних пучків є прикладом системи коефіцієнтів, яка підтримує всі шість функторів.

Example 4. Для поля k категорія DM(k; $\mathbb{Q}$) геометричних мотивів Воєводського еквівалентна компактній частині $\mathrm{DM}_\mathrm{B}(k)$, що є основою для раціональних мотивних когомологій.

9.9 Висновки

Шестифункторний формалізм Гротендіка є потужним інструментом, який узагальнює когомології та дозволяє працювати з відносними інваріантами в алгебраїчній геометрії. Його зв'язок із когезивними топосами та похідними категоріями відкриває нові можливості для дослідження складних геометричних структур. У майбутньому цей формалізм, ймовірно, залишатиметься ключовим у розвитку мотивної гомотопічної теорії та інших областей математики.

Література

[1] Gallauer, M. (2021). An introduction to six-functor formalisms. *arXiv:2112.10456v1.*

[2] Ayoub, J. (2007). Les six opérations de Grothendieck et le formalisme des cycles évanescents dans le monde motivique. *Astérisque*, 314-315.

[3] Cisinski, D.-C., & Déglise, F. (2019). Triangulated categories of mixed motives. *Springer Monographs in Mathematics*.

Issue XXXIII: Structure Preserving Theorems

Namdak Tonpa

5 травня 2025 р.

Анотація

This article unifies algebra and geometry by characterizing algebra as the domain of homomorphisms preserving structure and geometry as the domain of inverse images of homomorphisms preserving structure. We introduce two new theorems: the Homomorphism Preservation Theorem (HPT) for Algebraic Categories and the Inverse Image Preservation Theorem (IIPT) for Geometric Categories. These build on foundational results like the First Isomorphism Theorem, Continuity Theorem, Pullback Theorem, Stone Duality, Gelfand Duality, and Adjoint Functor Theorem. Aimed at advanced graduate students, this exposition uses category theory to illuminate the algebraic-geometric duality.

10 Algebra and Geometry

Algebra and geometry, foundational to pure mathematics, differ in focus: algebra on abstract structures and their transformations, geometry on spatial properties and invariants. We propose a unifying perspective: algebra is defined by homomorphisms preserving structure, and geometry by the inverse images of homomorphisms preserving structure. This article formalizes this view through two explicit theorems—the Homomorphism Preservation Theorem (HPT) for Algebraic Categories and the Inverse Image Preservation Theorem (IIPT) for Geometric Categories—building on established results. Assuming familiarity with category theory, algebraic topology, and commutative algebra, we provide a framework for graduate students to explore these fields' interplay.

10.1 Homomorphisms in Algebra

Definition 53. Let $\mathcal{C}$ be a category, and let A, B be objects in $\mathcal{C}$. A *homomorphism* $\phi : A \to B$ is a morphism in $\mathcal{C}$ that preserves the structure defined by the category's operations and relations.

In algebraic categories (e.g., $\mathbf{Grp}$, $\mathbf{Ring}$, $\mathbf{Mod}_R$), homomorphisms preserve operations like group multiplication or module scalar multiplication.

Example 5. In $\mathbf{Grp}$, a group homomorphism $\phi : G \to H$ satisfies $\phi(g_1 g_2) = \phi(g_1)\phi(g_2)$ for all $g_1, g_2 \in G$, preserving the group operation.

Theorem 12 (First Isomorphism Theorem)**.** Let $\phi : G \to H$ be a group homomorphism with kernel $K = \ker(\phi)$. Then $G/K \cong \mathrm{im}(\phi)$.

Theorem 13 (Universal Property of Free Objects)**.** In an algebraic category (e.g., $\mathbf{Grp}$, $\mathbf{Ring}$), for a free object $F(X)$ on a set X, any map $f : X \to A$ (where A is an object) extends uniquely to a homomorphism $\phi : F(X) \to A$.

We now introduce a theorem encapsulating the algebraic perspective.

Theorem 14 (Homomorphism Preservation Theorem for Algebraic Categories)**.** Let $\mathcal{C}$ be an algebraic category (e.g., $\mathbf{Grp}$, $\mathbf{Ring}$, $\mathbf{Mod}_R$) with a forgetful functor $U : \mathcal{C} \to \mathbf{Set}$. For any surjective homomorphism $\phi : A \to B$ in $\mathcal{C}$ with kernel K (a normal subobject), there exists an isomorphism $\psi : A/K \to B$ such that $\psi \circ \pi = \phi$, where $\pi : A \to A/K$ is the canonical projection. Moreover, any object A can be generated by a free object $F(X)$ via a surjective homomorphism whose structure is preserved by ϕ.

Доведення. The first part follows from the First Isomorphism Theorem [1]: for a surjective homomorphism $\phi : A \to B$ with kernel K, the quotient $A/K \cong B$ via the isomorphism $\psi : aK \mapsto \phi(a)$. The second part follows from the Universal Property of Free Objects [2]: for any object A, there exists a set X and a free object $F(X)$ with a surjective homomorphism $\eta : F(X) \to A$, and any homomorphism $\phi : A \to B$ extends the structure-preserving maps from $F(X)$. □

Remark 7. The HPT formalizes that homomorphisms in algebraic categories preserve structure forward, inducing isomorphisms on quotients and respecting generators, unifying the First Isomorphism Theorem and Universal Property. The name avoids confusion with the Structure-Identity Principle in category theory [2].

10.2 Homomorphisms in Geometry

Geometry emphasizes spaces where structure is preserved under inverse images of homomorphisms, as in $\mathbf{Top}$ or $\mathbf{Sch}$.

Definition 54. Let $\phi : X \to Y$ be a morphism in a category $\mathcal{C}$. The *inverse image* of a subobject $S \subseteq Y$ (if it exists) is the subobject $\phi^{-1}(S) \subseteq X$ defined via the pullback of $S \hookrightarrow Y$ along ϕ.

Example 6. In $\mathbf{Top}$, a continuous map $\phi : X \to Y$ ensures that $\phi^{-1}(V) \subseteq X$ is open for every open set $V \subseteq Y$.

Theorem 15 (Continuity in Topology)**.** A function $\phi : X \to Y$ between topological spaces is continuous if and only if for every open set $V \subseteq Y$, the inverse image $\phi^{-1}(V)$ is open in X.

Theorem 16 (Pullback Theorem in Sheaf Theory)**.** For a morphism $\phi : X \to Y$ in a category with sheaves (e.g., $\mathbf{Top}$, $\mathbf{Sch}$), the inverse image functor ϕ^{-1} : $\mathrm{Sh}(Y) \to \mathrm{Sh}(X)$ is exact, preserving the structure of sheaves.

We now define a theorem for geometric categories.

Theorem 17 (Inverse Image Preservation Theorem for Geometric Categories)**.** Let $\mathcal{C}$ be a geometric category (e.g., $\mathbf{Top}$, $\mathbf{Sch}$) with pullbacks. For any morphism $\phi : X \to Y$ in $\mathcal{C}$, the inverse image functor $\phi^{-1} : \mathrm{Sub}(Y) \to \mathrm{Sub}(X)$ preserves the lattice structure of subobjects. If $\mathcal{C}$ admits sheaves, $\phi^{-1} : \mathrm{Sh}(Y) \to \mathrm{Sh}(X)$ is exact and preserves sheaf isomorphisms, ensuring that the geometric structure of Y is reflected in X.

Доведення. In $\mathbf{Top}$, the Continuity Theorem [4] ensures that $\phi : X \to Y$ is continuous if and only if $\phi^{-1}(V)$ is open for every open set $V \subseteq Y$, so ϕ^{-1} preserves the lattice of open sets. In categories with sheaves (e.g., $\mathbf{Top}$, $\mathbf{Sch}$), the Pullback Theorem [5] guarantees that $\phi^{-1} : \mathrm{Sh}(Y) \to \mathrm{Sh}(X)$ is exact, preserving sheaf structures. For schemes, ϕ^{-1} maps prime ideals to prime ideals [3], preserving geometric properties. Since ϕ^{-1} is functorial and preserves monomorphisms, it maintains isomorphisms of subobjects or sheaves. □

Remark 8. The IIPT captures the geometric essence of inverse images preserving structure, unifying the Continuity Theorem and Pullback Theorem. The name distinguishes it from the Structure-Identity Principle [2].

Example 7. For a morphism of schemes $\phi : X \to Y$, the inverse image of a prime ideal under the induced map on stalks is prime, preserving geometric structure [3].

10.3 Categorical Unification

Category theory bridges algebra and geometry through dualities, where the HPT and IIPT interplay.

Theorem 18 (Stone Duality)**.** The category of Boolean algebras, $\mathbf{BoolAlg}$, is dually equivalent to the category of Stone spaces, $\mathbf{Stone}$, via the spectrum functor.

Theorem 19 (Gelfand Duality)**.** The category of commutative C*-algebras is dually equivalent to the category of compact Hausdorff spaces via the spectrum functor.

Theorem 20 (Adjoint Functor Theorem)**.** In a complete category, a functor has a left adjoint if it preserves limits, and a right adjoint if it preserves colimits.

Remark 9. Stone and Gelfand Dualities [6, 7] connect algebraic homomorphisms (HPT) to geometric inverse images (IIPT). The Adjoint Functor Theorem [2] underpins dualities like Spec, where algebraic and geometric structures are preserved [3].

Example 8. The Spec functor maps a ring homomorphism $\phi : R \to S$ to a morphism $\mathbf{Spec}S \to \mathbf{Spec}R$, with inverse images of prime ideals preserving geometric structure.

10.4 Applications and Implications

The HPT and IIPT, supported by prior results, impact advanced research:

- **Algebraic Topology**: The HPT governs homology maps, while the IIPT defines covering spaces.
- **Algebraic Geometry**: The IIPT underpins étale cohomology via inverse images, while the HPT applies to ring homomorphisms.
- **Category Theory**: Stone, Gelfand, and Adjoint Functor Theorems reveal algebra-geometry correspondences.

Corollary 1. In any category with pullbacks, $\phi^{-1} : \mathrm{Sub}(Y) \to \mathrm{Sub}(X)$ preserves subobject lattices, as per the IIPT.

10.5 Conclusion

The Homomorphism Preservation Theorem and Inverse Image Preservation Theorem formalize that algebra preserves structure via homomorphisms and geometry via inverse images. Building on the First Isomorphism Theorem, Continuity Theorem, Pullback Theorem, and dualities, these theorems unify pure mathematics. Graduate students are encouraged to apply this framework to algebraic topology, algebraic geometry, and category theory, deepening their research.

Література

[1] Lang, S., *Algebra*, Graduate Texts in Mathematics, Springer, 2002.

[2] Mac Lane, S., *Categories for the Working Mathematician*, Graduate Texts in Mathematics, Springer, 1998.

[3] Hartshorne, R., *Algebraic Geometry*, Graduate Texts in Mathematics, Springer, 1977.

[4] Munkres, J. R., *Topology*, 2nd ed., Prentice Hall, 2000.

[5] Kashiwara, M., Schapira, P., *Sheaves on Manifolds*, Springer, 1990.

[6] Johnstone, P. T., *Stone Spaces*, Cambridge University Press, 1982.

[7] Takesaki, M., *Theory of Operator Algebras I*, Springer, 2002.

Issue XXXIV: Grothendieck Schemes

Namdak Tonpa

5 травня 2025 р.

Анотація

We present Grothendieck's functorial definition of schemes as sheaves on the category of affine schemes, structured according to the functor of points perspective. We also outline a path toward formalizing these objects within Homotopy Type Theory (HoTT).

11 Grothendieck Schemes

We view schemes as **sheaves on the category of affine schemes**, satisfying a gluing condition analogous to the usual descent condition in topology.

11.1 Affine Schemes

Let:

$$\mathsf{Aff} := (\mathsf{CRing})^{\mathrm{op}}$$

denote the category of affine schemes, i.e., the opposite of the category of commutative rings.

An affine scheme is of the form $\mathrm{Spec}(\mathsf{A})$, for a commutative ring A.

11.2 Zariski Covers

A **presheaf of sets** on Aff is a functor:

$$\mathsf{F} : \mathsf{Aff}^{\mathrm{op}} \to \mathit{Set}.$$

This is the *functor of points* perspective: each affine scheme $\mathrm{Spec}(\mathsf{A})$ represents the "test ring"A, and $\mathsf{F}(\mathrm{Spec}(\mathsf{A}))$ can be thought of as the A-points of F.

A **Zariski sheaf** is a presheaf that satisfies descent for Zariski covers: if $\{\mathrm{Spec}(\mathsf{A}_{f_i}) \to \mathrm{Spec}(\mathsf{A})\}$ is a Zariski open affine cover, then the diagram

$$\mathsf{F}(\mathrm{Spec}(\mathsf{A})) \to \mathrm{Eq}\left(\prod_i \mathsf{F}(\mathrm{Spec}(\mathsf{A}_{f_i})) \rightrightarrows \prod_{i,j} \mathsf{F}(\mathrm{Spec}(\mathsf{A}_{f_i f_j}))\right)$$

is an equalizer diagram.

11.3 Grothendieck Scheme

A **scheme** is a Zariski sheaf

$$\mathsf{F} : \mathbf{Aff}^{\mathrm{op}} \to \boldsymbol{Set}$$

such that:

- There exists a Zariski cover $\{U_i \to F\}$ where each U_i is **representable**, i.e., $U_i \cong \mathrm{Spec}(A_i)$ for some ring A_i.
- Each morphism $U_i \to F$ is an **open immersion** (in the sheaf-theoretic sense).

This means F is **locally isomorphic to affine schemes** and satisfies Zariski descent.

Equivalently: Schemes are Zariski sheaves on $\mathbf{Aff}$ that are **locally representable by affine schemes**.

11.4 Formalization in HoTT

Categories and Presheaves in HoTT

In HoTT, a category can be defined as a type of objects together with types of morphisms and operations satisfying associativity and identity laws up to higher homotopies. A presheaf is then a functor:

$$F : \mathcal{C}^{\mathrm{op}} \to \mathcal{U}_0$$

where $\mathcal{U}_0$ is the universe of 0-types (sets). For $\mathcal{C} = \mathbf{Aff}$, this gives us the functor-of-points view.

Sheaf Conditions in HoTT

A sheaf in HoTT is a presheaf that satisfies a descent condition with respect to a Grothendieck topology, formalized via homotopy limits or truncations, depending on the level of the types involved.

Defining Schemes in HoTT

Within HoTT, a scheme is a sheaf $F : \mathbf{Aff}^{\mathrm{op}} \to \mathcal{U}_0$ satisfying:

- A Zariski descent condition.
- Local representability: there exists a family of open immersions $\{\mathrm{Spec}(A_i) \to F\}$ covering F.

This mirrors the classical definition but is grounded in type-theoretic and higher-categorical constructions.

11.5 Conclusion

Grothendieck's functorial approach to schemes provides a clean and general definition that is well-suited for formalization in Homotopy Type Theory. This opens the way for a synthetic and structured foundation for algebraic geometry in type-theoretic settings.

Issue XXXV: Categories of Spectra

Maksym Sokhatskyi [1]

[1] National Technical University of Ukraine
Igor Sikorsky Kyiv Polytechnical Institute
5 травня 2025 р.

Анотація

Keywords: Stable Homotopy Theory

12 Categories of Spectra

Issue XXXVI: Spectral Categories

Maksym Sokhatskyi [1]

[1] National Technical University of Ukraine
Igor Sikorsky Kyiv Polytechnical Institute
5 травня 2025 р.

Анотація

Keywords: Stable Homotopy Theory

13 Spectral Categories

Issue XXXVII: Spectral Categories

Maksym Sokhatskyi [1]

[1] National Technical University of Ukraine
Igor Sikorsky Kyiv Polytechnical Institute
5 травня 2025 р.

Анотація

Keywords: Stable Homotopy Theory

14 Simplicial Categories

Issue XXXIX: Topos on Category of Sets

Maxim Sokhatsky

[1] National Technical University of Ukraine
Igor Sikorsky Kyiv Polytechnical Institute

5 травня 2025 р.

Анотація

The purpose of this work is to clarify all topos definitions using type theory. Not much efforts was done to give all the examples, but one example, a topos on category of sets, is constructively presented at the finale.

As this cricial example definition is used in presheaf definition, the construction of category of sets is a mandatory excercise for any topos library. We propose here cubicaltt[1] version of elementary topos on category of sets for demonstration of categorical semantics (from logic perspective) of the fundamental notion of set theory in mathematics.

Other disputed foundations for set theory could be taken as: ZFC, NBG, ETCS. We will disctinct syntetically: i) category theory; ii) set theory in univalent foundations; iii) topos theory, grothendieck topos, elementary topos. For formulation of definitions and theorems only Martin-Löf Type Theory is requested. The proofs involve cubical type checker primitives.

Keywords: Homotopy Type Theory, Topos Theory

[1]Cubical Type Theory, http://github.com/mortberg/cubicaltt

15 Topos Theory

One can admit two topos theory lineages. One lineage takes its roots from published by Jean Leray in 1945 initial work on sheaves and spectral sequences. Later this lineage was developed by Henri Paul Cartan, André Weil. The peak of lineage was settled with works by Jean-Pierre Serre, Alexander Grothendieck, and Roger Godement.

Second remarkable lineage take its root from William Lawvere and Myles Tierney. The main contribution is the reformulation of Grothendieck topology by using subobject classifier.

15.1 Set Theory

Here is given the ∞-groupoid model of sets.

Definition 55. (Mere proposition, PROP). A type P is a mere proposition if for all $x, y : P$ we have $x = y$:

$$\text{isProp}(P) = \prod_{x,y:P} (x = y).$$

Definition 56. (0-type). A type A is a 0-type is for all $x, y : A$ and $p, q : x =_A y$ we have $p = q$.

Definition 57. (1-type). A type A is a 1-type if for all $x, y : A$ and $p, q : x =_A y$ and $r, s : p =_{=_A} q$, we have $r = s$.

Definition 58. (A set of elements, SET). A type A is a SET if for all $x, y : A$ and $p, q : x = y$, we have $p = q$:

$$\text{isSet}(A) = \prod_{x,y:A} \prod_{p,q:x=y} (p = q).$$

Definition 59. `data N = Z | S (n: N)`

```
n_grpd (A: U) (n: N): U = (a b: A) -> rec A a b n where
  rec (A: U) (a b: A) : (k: N) -> U
    = split { Z -> Path A a b ; S n -> n_grpd (Path A a b) n }

isContr (A: U): U = (x: A) * ((y: A) -> Path A x y)
isProp  (A: U): U = n_grpd A Z
isSet   (A: U): U = n_grpd A (S Z)
PROP  : U = (X:U) * isProp X
SET   : U = (X:U) * isSet X
```

Definition 60. (Π-Contractability). If fiber is set thene path space between any sections is contractible.

```
setPi (A: U) (B: A -> U) (h: (x: A) -> isSet (B x)) (f g: Pi A B)
      (p q: Path (Pi A B) f g)
    : Path (Path (Pi A B) f g) p q
```

Definition 61. (Σ-Contractability). If fiber is set then Σ is set.

```
setSig (A:U) (B: A -> U) (base: isSet A)
       (fiber: (x:A) -> isSet (B x)) : isSet (Sigma A B)
```

Definition 62. (Unit type, $\mathbf{1}$). The unit $\mathbf{1}$ is a type with one element.

```
data unit = tt
unitRec (C: U) (x: C): unit -> C = split tt -> x
unitInd (C: unit -> U) (x: C tt): (z:unit) -> C z
  = split tt -> x
```

Theorem 21. (Category of Sets, $\boldsymbol{Set}$). Sets forms a Category. All compositional theorems proved by using reflection rule of internal language. The proof that Hom forms a set is taken through Π-contractability.

```
Set: precategory = ((Ob,Hom),id,c,HomSet,L,R,Q) where
  Ob: U = SET
  Hom (A B: Ob): U = A.1 -> B.1
  id (A: Ob): Hom A A = idfun A.1
  c (A B C: Ob) (f: Hom A B) (g: Hom B C): Hom A C
    = o A.1 B.1 C.1 g f
  HomSet (A B: Ob): isSet (Hom A B) = setFun A.1 B.1 B.2
  L (A B:Ob) (f:Hom A B): Path (Hom A B)(c A A B (id A)f)f
    = refl (Hom A B) f
  R (A B:Ob) (f:Hom A B): Path (Hom A B)(c A B B f(id B))f
    = refl (Hom A B) f
  Q (A B C D: Ob) (f:Hom A B) (g:Hom B C) (h:Hom C D)
    : Path (Hom A D) (c A C D (c A B C f g) h)
                     (c A B D f (c B C D g h))
    = refl (Hom A D) (c A B D f (c B C D g h))
```

15.2 Topological Structure

Topos theory extends category theory with notion of topological structure but reformulated in a categorical way as a category of sheaves on a site or as one that has cartesian closure and subobject classifier. We give here two definitions.

Definition 63. (Topology). The topological structure on A (or topology) is a subset $S \in A$ with following properties: i) any finite union of subsets of S is belong to S; ii) any finite intersection of subsets of S is belong to S. Subets of S are called open sets of family S.

```
def =₁ (A : U₁) (x y : A) := PathP (<_> A) x y
def isProp₁ (A : U₁) := Π (a b : A), =₁ A a b
def isSet₁ (A : U₁) := Π (a b : A) (x y : =₁ A a b), =₁
(=₁ A a b) x y
def Prop := U → 2
def ℙ (X: U₁) := X → Prop

def ∅ (X: U₁) : ℙ X
 := λ (_: X) (_: U), false

def total (X: U₁) : ℙ X
 := λ (_: X) (_: U), true

def ∈ (X: U₁) (el: X) (set: ℙ X) : U₁
 := =₁ (U → 2) (set el) (\(_: U), true)

def ∉ (X: U₁) (el: X) (set: ℙ X) : U₁
 := =₁ (U → 2) (set el) (\(_: U), false)

def ⊆ (X: U₁) (A B: ℙ X)
 := Π (x: X), (∈ X x A) × (∈ X x B)

def ⊆ (X: U₁) : ℙ X → ℙ X
 := λ (h  : ℙ X), λ (x: X) (Y: U), not (h x Y)

def ∪ (X: U₁) : ℙ X → ℙ X → ℙ X
 := λ (h1 : ℙ X) (h2: ℙ X), λ (x: X) (Y: U), or (h1 x Y) (h2 x Y)

def ∩ (X: U₁) : ℙ X → ℙ X → ℙ X
 := λ (h1 : ℙ X) (h2: ℙ X), λ (x: X) (Y: U), and (h1 x Y) (h2 x Y)
```

For fully functional general topology theorems and Zorn lemma you can refer to the Coq library [2]topology by Daniel Schepler.

[2]https://github.com/verimath/topology

15.3 Grothendieck Topos

Grothendieck Topology is a calculus of coverings which generalizes the algebra of open covers of a topological space, and can exist on much more general categories. There are three variants of Grothendieck topology definition: i) sieves; ii) coverage; iii) covering families. A category have one of these three is called a Grothendieck site.

Examples: Zariski, flat, étale, Nisnevich topologies.

A sheaf is a presheaf (functor from opposite category to category of sets) which satisties patching conditions arising from Grothendieck topology, and applying the associated sheaf functor to preashef forces compliance with these conditions.

The notion of Grothendieck topos is a geometric flavour of topos theory, where topos is defined as category of sheaves on a Grothendieck site with geometric moriphisms as adjoint pairs of functors between topoi, that satisfy exactness properties. [?]

As this flavour of topos theory uses category of sets as a prerequisite, the formal construction of set topos is cricual in doing sheaf topos theory.

Definition 64. (Sieves). Sieves are a family of subfunctors

$$\mathsf{R} \subset \mathsf{Hom}_C(\_, \mathsf{U}), \mathsf{U} \in C,$$

such that following axioms hold: i) (base change) If $\mathsf{R} \subset \mathsf{Hom}_C(\_, \mathsf{U})$ is covering and $\phi : \mathsf{V} \to \mathsf{U}$ is a morphism of C, then the subfuntor

$$\phi^{-1}(\mathsf{R}) = \{\gamma : W \to \mathsf{V} \| \phi \cdot \gamma \in \mathsf{R}\}$$

is covering for V; ii) (local character) Suppose that $\mathsf{R}, \mathsf{R}' \subset \mathsf{Hom}_C(\_, \mathsf{U})$ are subfunctors and R is covering. If $\phi^{-1}(\mathsf{R}')$ is covering for all $\phi : \mathsf{V} \to \mathsf{U}$ in R, then R' is covering; iii) $\mathsf{Hom}_C(\_, \mathsf{U})$ is covering for all $\mathsf{U} \in C$.

Definition 65. (Coverage). A coverage is a function assigning to each Ob_C the family of morphisms $\{f_i : U_i \to U\}_{i \in I}$ called covering families, such that for any $g : V \to U$ exist a covering family $\{h : V_j \to V\}_{j \in J}$ such that each composite $h_j \circ g$ factors some f_i:

$$\begin{tikzcd} V_j \arrow[r, "k"] \arrow[d, "h"] & U_i \arrow[d, "f_i"] \\ V \arrow[r, "g"] & U \end{tikzcd}$$

```
def Co (C: precategory) (cod: C.C.ob) : U
 := Σ (dom: C.C.ob), C.C.hom dom cod

def Delta (C: precategory) (d: C.C.ob) : U₁
 := Σ (index: U), index -> Co C d

def Coverage (C: precategory): U₁
 := Σ (cod: C.C.ob) (fam: Delta C cod)
      (coverings: C.C.ob -> Delta C cod -> U),
      coverings cod fam

def site  (C: precategory): U₁
 := Σ (C: precategory), Coverage C
```

Definition 66. (Grothendieck Topology). Suppose category C has all pullbacks. Since C is small, a pretopology on C consists of families of sets of morphisms

$$\{\phi_\alpha : U_\alpha \to U\}, U \in \mathrm{C},$$

called covering families, such that following axioms hold: i) suppose that $\phi_\alpha : U_\alpha \to U$ is a covering family and that $\psi : V \to U$ is a morphism of C. Then the collection $V \times_U U_\alpha \to V$ is a cvering family for V. ii) If $\{\phi_\alpha : U_\alpha \to U\}$ is covering, and $\{\gamma_{\alpha,\beta} : W_{\alpha,\beta} \to U_\alpha\}$ is covering for all α, then the family of composites

$$W_{\alpha,\beta} \xrightarrow{\gamma_{\alpha,\beta}} U_\alpha \xrightarrow{\phi_\alpha} U$$

is covering; iii) The family $\{1 : U \to U\}$ is covering for all $U \in \mathrm{C}$.

Definition 67. (Site). Site is a category having either a coverage, grothendieck topology, or sieves.

```
site (C: precategory): U
  = (C: precategory) * Coverage C
```

Definition 68. (Presheaf). Presheaf of a category C is a functor from opposite category to category of sets: $\mathrm{C}^{op} \to \mathrm{Set}$.

```
presheaf (C: precategory): U
  = catfunctor (opCat C) Set
```

Definition 69. (Presheaf Category, **PSh**). Presheaf category **PSh** for a site C is category were objects are presheaves and morphisms are natural transformations of presheaf functors.

Definition 70. (Sheaf). Sheaf is a presheaf on a site. In other words a presheaf $F : C^{op} \to \boldsymbol{Set}$ such that the cannonical map of inverse limit

$$F(U) \to \overleftarrow{\lim_{V \to U \in R}} F(V)$$

is an isomorphism for each covering sieve $R \subset \mathrm{Hom}_C(\_, U)$. Equivalently, all induced functions

$$\mathrm{Hom}_C(\mathrm{Hom}_C(\_, U), F) \to \mathrm{Hom}_C(R, F)$$

should be bejections.

```
sheaf (C: precategory): U
  = (S: site C)
  * presheaf S.1
```

Definition 71. (Sheaf Category, **Sh**). Sheaf category **Sh** is a category where objects are sheaves and morphisms are natural transformation of sheves. Sheaf category is a full subcategory of category of presheaves **PSh**.

Definition 72. (Grothendieck Topos). Topos is the category of sheaves $\mathbf{Sh}(C, J)$ on a site C with topology J.

Theorem 22. (Giraud). A category C is a Grothiendieck topos iff it has following properties: i) has all finite limits; ii) has small disjoint coproducts stable under pullbacks; iii) any epimorphism is coequalizer; iv) any equivalence relation $R \to E$ is a kernel pair and has a quotient; v) any coequalizer $R \to E \to Q$ is stably exact; vi) there is a set of objects that generates C.

Definition 73. (Geometric Morphism). Suppose that C and D are Grothendieck sites. A geometric morphism

$$f : \mathbf{Sh}(C) \to \mathbf{Sh}(D)$$

consist of functors $f_* : \mathbf{Sh}(C) \to \mathbf{Sh}(D)$ and $f^* : \mathbf{Sh}(D) \to \mathbf{Sh}(C)$ such that f^* is left adjoint to f_* and f^* preserves finite limits. The left adjoint f^* is called the inverse image functor, while f_* is called the direct image. The inverse image functor f^* is left and right exact in the sense that it preserves all finite colimits and limits, respectively.

Definition 74. (Cohesive Topos). A topos E is a cohesive topos over a base topos S, if there is a geometric morphism $(p^*, p_*) : E \to S$, such that: i) exists adjunction $p^! \vdash p_*$ and $p^! \dashv p_*$; ii) p^* and $p^!$ are full faithful; iii) $p_!$ preserves finite products.

This quadruple defines adjoint triple:

$$\int \dashv \flat \dashv \sharp$$

15.4 Elementary Topos

Giraud theorem was a synonymical topos definition involved only topos properties but not a site properties. That was step forward on predicative definition. The other step was made by Lawvere and Tierney, by removing explicit dependance on categorical model of set theory (as category of set is used in definition of presheaf). This information was hidden into subobject classifier which was well defined through categorical pullback and property of being cartesian closed (having lambda calculus as internal language).

Elementary topos doesn't involve 2-categorical modeling, so we can construct set topos without using functors and natural transformations (what we need in geometrical topos theory flavour). This flavour of topos theory more suited for logic needs rather that geometry, as its set properties are hidden under the predicative predicative pullback definition of subobject classifier rather that functorial notation of presheaf functor. So we can simplify proofs at the homotopy levels, not to lift everything to 2-categorical model.

Definition 75. (Monomorphism). An morphism $f : Y \to Z$ is a monic or mono if for any object X and every pair of parralel morphisms $g_1, g_2 : X \to Y$ the

$$f \circ g_1 = f \circ g_2 \to g_1 = g_2.$$

More abstractly, f is mono if for any X the Hom(X, _) takes it to an injective function between hom sets Hom(X, Y) $\to$ Hom(X, Z).

```
mono (P: precategory) (Y Z: carrier P) (f: hom P Y Z): U
   = (X: carrier P) (g1 g2: hom P X Y)
  -> Path (hom P X Z) (compose P X Y Z g1 f)
                      (compose P X Y Z g2 f)
  -> Path (hom P X Y) g1 g2
```

Definition 76. (Subobject Classifier[?]). In category C with finite limits, a subobject classifier is a monomorphism $\mathsf{true} : 1 \to \Omega$ out of terminal object 1, such that for any mono $U \to X$ there is a unique morphism $\chi_U : X \to \Omega$ and pullback diagram:

$$\begin{array}{ccc} U & \xrightarrow{k} & 1 \\ \downarrow & & \downarrow{\scriptstyle \mathsf{true}} \\ X\Omega & \xrightarrow{\chi_u} & \Omega \end{array}$$

```
subobjectClassifier (C: precategory): U
  = (omega: carrier C)
  * (end: terminal C)
  * (trueHom: hom C end.1 omega)
  * (chi: (V X: carrier C) (j: hom C V X) -> hom C X omega)
  * (square: (V X: carrier C) (j: hom C V X) -> mono C V X j
    -> hasPullback C (omega,(end.1,trueHom),(X,chi V X j)))
  * ((V X: carrier C) (j: hom C V X) (k: hom C X omega)
    -> mono C V X j
    -> hasPullback C (omega,(end.1,trueHom),(X,k))
    -> Path (hom C X omega) (chi V X j) k)
```

Theorem 23. (Category of Sets has Subobject Classifier).

Definition 77. (Cartesian Closed Categories). The category C is called cartesian closed if exists all: i) terminals; ii) products; iii) exponentials. Note that this definition lacks beta and eta rules which could be found in embedding **MLTT**.

```
isCCC (C: precategory): U
  = (Exp:    (A B: carrier C) -> carrier C)
  * (Prod:   (A B: carrier C) -> carrier C)
  * (Apply:  (A B: carrier C) -> hom C (Prod (Exp A B) A) B)
  * (P1:     (A B: carrier C) -> hom C (Prod A B) A)
  * (P2:     (A B: carrier C) -> hom C (Prod A B) B)
  * (Term:   terminal C)
  * unit
```

Theorem 24. (Category of Sets is cartesian closed). As you can see from exp and pro we internalize Π and Σ types as SET instances, the isSet predicates are provided with contractability. Exitense of terminals is proved by propPi. The same technique you can find in **MLTT** embedding.

```
cartesianClosure : isCCC Set
  = (expo,prod,appli,proj1,proj2,term,tt) where
    exp (A B: SET): SET = (A.1    -> B.1, setFun A.1 B.1 B.2)
    pro (A B: SET): SET = (prod A.1 B.1, setSig A.1 (\(_ : A.1)
                          -> B.1) A.2 (\(_ : A.1) -> B.2))
    expo:  (A B: SET) -> SET = \(A B: SET) -> exp A B
    prod:  (A B: SET) -> SET = \(A B: SET) -> pro A B
    appli: (A B: SET) -> hom Set (pro (exp A B) A) B
       = \(A B: SET) -> \(x:(pro(exp A B)A).1)-> x.1 x.2
    proj1: (A B: SET) -> hom Set (pro A B) A
       = \(A B: SET) (x: (pro A B).1) -> x.1
    proj2: (A B: SET) -> hom Set (pro A B) B
       = \(A B: SET) (x: (pro A B).1) -> x.2
    unitContr (x: SET) (f: x.1 -> unit) : isContr (x.1 -> unit)
      = (f, \(z: x.1 -> unit) -> propPi x.1 (\(_:x.1)->unit)
          (\(x:x.1) -> propUnit) f z)
    term: terminal Set = ((unit,setUnit),
          \(x: SET) -> unitContr x (\(z: x.1) -> tt))
```

Note that rules of cartesian closure forms a type theoretical langage called lambda calculus.

Definition 78. (Elementary Topos). Topos is a precategory which is cartesian closed and has subobject classifier.

```
Topos (cat: precategory) : U
  = (cartesianClosure: isCCC cat)
  * subobjectClassifier cat
```

Theorem 25. (Topos Definitions). Any Grothendieck topos is an elementary topos too. The proof is sligthly based on results of Giraud theorem.

Theorem 26. (Category of Sets forms a Topos). There is a cartesian closure and subobject classifier for a categoty of sets.

```
internal : Topos Set
         = (cartesianClosure, hasSubobject)
```

Theorem 27. (Freyd). Main theorem of topos theory[**?**]. For any topos C and any $\mathsf{b} : \mathrm{Ob}_{\mathsf{C}}$ relative category $\mathsf{C} \downarrow \mathsf{b}$ is also a topos. And for any arrow $\mathsf{f} : \mathsf{a} \to \mathsf{b}$ inverse image functor $\mathsf{f}^* : \mathsf{C} \downarrow \mathsf{b} \to \mathsf{c} \downarrow \mathsf{a}$ has left adjoint $\sum_{\mathsf{f}}$ and right adjoin $\prod_{\mathsf{f}}$.

Conclusion

We gave here constructive definition of topology as finite unions and intersections of open subsets. Then make this definition categorically compatible by introducing Grothendieck topology in three different forms: sieves, coverage, and covering families. Then we defined an elementary topos and introduce category of sets, and proved that **Set** is cartesian closed, has object classifier and thus a topos.

This intro could be considered as a formal introduction to topos theory (at least of the level of first chapter) and you may evolve this library to your needs or ask to help porting or developing your application of topos theory to a particular formal construction.

Issue XL: Cohesive Topos

Максим Сохацький [1]

[1] Національний технічний університет України
Київський політехнічний інститут імені Ігоря Сікорського
5 травня 2025 р.

Анотація

Formal definition of Cohesive Topos.
Keywords: Topos Theory

16 Cohesive Topos Theory

16.1 Preliminaries

A **category** $\mathcal{C}$ consists of:

- A class of **objects**, $\mathrm{Ob}(\mathcal{C})$,
- A class of **morphisms**, $\mathrm{Hom}_{\mathcal{C}}(X, Y)$, for each pair $X, Y \in \mathrm{Ob}(\mathcal{C})$,
- Composition maps $\circ : \mathrm{Hom}(Y, Z) \times \mathrm{Hom}(X, Y) \to \mathrm{Hom}(X, Z)$,
- Identity morphisms $\mathrm{id}_X \in \mathrm{Hom}(X, X)$ for each X,

satisfying associativity and identity laws.

A **functor** $F : \mathcal{C} \to \mathcal{D}$ assigns to each:

- Object $X \in \mathcal{C}$ an object $F(X) \in \mathcal{D}$,
- Morphism $f : X \to Y$ a morphism $F(f) : F(X) \to F(Y)$,

such that $F(\mathrm{id}_X) = \mathrm{id}_{F(X)}$ and $F(g \circ f) = F(g) \circ F(f)$.

A **natural transformation** $\eta : F \Rightarrow G$ between functors $F, G : \mathcal{C} \to \mathcal{D}$ consists of morphisms $\eta_X : F(X) \to G(X)$ such that for every $f : X \to Y$ in $\mathcal{C}$,

$$\begin{tikzcd} F(X) \arrow[r, "\eta_X"] \arrow[d, "F(f)"'] & G(X) \arrow[d, "G(f)"] \\ F(Y) \arrow[r, "\eta_Y"'] & G(Y) \end{tikzcd}$$

commutes.

An **adjunction** between categories $\mathcal{C}$ and $\mathcal{D}$ consists of functors

$$\mathsf{F} : \mathcal{C} \leftrightarrows \mathcal{D} : \mathsf{G}$$

and natural transformations (unit η and counit ε)

$$\eta : \mathrm{Id}_{\mathcal{C}} \Rightarrow \mathsf{G} \circ \mathsf{F}, \quad \varepsilon : \mathsf{F} \circ \mathsf{G} \Rightarrow \mathrm{Id}_{\mathcal{D}}$$

satisfying the triangle identities.

16.2 Topos

A **topos** $\mathcal{E}$ is a category that:

- Has all finite limits and colimits,
- Is Cartesian closed: has exponential objects $[\mathsf{X}, \mathsf{Y}]$,
- Has a subobject classifier Ω.

16.3 Geometric Morphism

A **geometric morphism** $\mathsf{f} : \mathcal{E} \to \mathcal{F}$ between topoi consists of an adjoint pair

$$\mathsf{f}^* : \mathcal{F} \leftrightarrows \mathcal{E} : \mathsf{f}_*$$

with $\mathsf{f}^* \dashv \mathsf{f}_*$, where f^* preserves finite limits (i.e., is left exact).

16.4 Cohesive Topos

A **cohesive topos** is a topos $\mathcal{E}$ equipped with a quadruple of adjoint functors:

$$\Pi \dashv \Delta \dashv \Gamma \dashv \nabla : \mathcal{E} \leftrightarrows \boldsymbol{Set}$$

such that:

- Γ is the global sections functor,
- Δ is the constant sheaf functor,
- ∇ sends a set to a codiscrete object,
- Π is the shape or fundamental groupoid functor,
- Δ and ∇ are fully faithful,
- Δ preserves finite limits,
- Π preserves finite products (in some variants).

16.5 Cohesive Adjunction Diagram and Modalities

$$\begin{tikzcd} \mathcal{E} \arrow[r, shift left=3, "\Pi"] \arrow[r, shift right=3, "\Gamma"'] & \boldsymbol{Set} \arrow[l, shift left=1, "\Delta"'] \arrow[l, shift right=1, "\nabla"] \arrow[l, phantom, "\dashv"] \end{tikzcd}$$

$$\begin{tikzcd} \mathcal{E} \arrow[rr, bend left, "\int"] \arrow[rr, bend right, "\sharp"'] \arrow[rr, phantom, "\Downarrow \flat"] & & \mathcal{E} \end{tikzcd}$$

16.6 Cohesive Modalities

The above adjoint quadruple canonically induces a triple of endofunctors on $\mathcal{E}$:

$$(\int \dashv \flat \dashv \sharp) : \mathcal{E} \to \mathcal{E}$$

defined as follows:

$$\int \coloneqq \Delta \circ \Pi$$
$$\flat \coloneqq \Delta \circ \Gamma$$
$$\sharp \coloneqq \nabla \circ \Gamma$$

This yields an **adjoint triple** of endofunctors on $\mathcal{E}$:

$$\int \dashv \flat \dashv \sharp$$

These are:

- $\int$ — the **shape modality**: captures the fundamental shape or homotopy type,
- $\flat$ — the **flat modality**: forgets cohesive structure while remembering discrete shape,
- $\sharp$ — the **sharp modality**: codiscretizes the structure, reflecting the full cohesion.

Each of these is an **idempotent** (co)monad, hence a *modality* in the internal language (type theory) of $\mathcal{E}$.

16.7 Differential Cohesion

A **differential cohesive topos** is a cohesive topos $\mathcal{E}$ equipped with an additional adjoint triple of endofunctors:

$$(\mathfrak{R} \dashv \mathfrak{I} \dashv \&) : \mathcal{E} \to \mathcal{E}$$

These are:

- $\mathfrak{R}$: the **reduction modality** — forgets nilpotents,
- $\mathfrak{I}$: the **infinitesimal shape modality** — retains infinitesimal data,
- $\&$: the **infinitesimal flat modality** — reflects formally smooth structure.

Important object classes:

- An object X is **reduced** if $\mathfrak{R}(\mathsf{X}) \cong \mathsf{X}$.
- It is **coreduced** if $\&(\mathsf{X}) \cong \mathsf{X}$.
- It is **formally smooth** if the unit map $\mathsf{X} \to \&\mathsf{X}$ is an effective epimorphism.

Formally étale maps are those morphisms $\mathsf{f} : \mathsf{X} \to \mathsf{Y}$ such that the square

$$\begin{tikzcd} \mathsf{X} \arrow[r] \arrow[d, "\mathsf{f}"'] & \mathfrak{I}\mathsf{X} \arrow[d, "\mathfrak{I}(\mathsf{f})"] \\ \mathsf{Y} \arrow[r] & \mathfrak{I}\mathsf{Y} \end{tikzcd}$$

is a pullback.

16.8 Graded Differential Cohesion

In **graded differential cohesion**, such as used in synthetic supergeometry, one introduces an adjoint triple:

10) $\rightrightarrows \dashv \rightsquigarrow \dashv \mathrm{Rh}$

$$(\rightrightarrows \dashv \rightsquigarrow \dashv \mathrm{Rh}) : \mathcal{E} \to \mathcal{E}$$

These are:

- $\rightrightarrows$: the **fermionic modality** — captures anti-commuting directions,
- $\rightsquigarrow$: the **bosonic modality** — filters out fermionic directions,
- Rh: the **rheonomic modality** — encodes constraint structures.

These modal operators form part of the internal logic of supergeometric or supersymmetric type theories.

Issue XLI: T-Spectra

Maksym Sokhatskyi [1]

[1] National Technical University of Ukraine
Igor Sikorsky Kyiv Polytechnical Institute
5 травня 2025 р.

Анотація

Keywords: Stable Homotopy Theory

17 Categories of T-Spectra

Issue XLII: ∞-Categories

Maksym Sokhatskyi [1]

[1] National Technical University of Ukraine
Igor Sikorsky Kyiv Polytechnical Institute
5 травня 2025 р.

Анотація

Keywords: Stable Homotopy Theory

18 ∞-Categories