Lawvere-Tierney Sheafification in Homotopy Type Theory

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

29 June 2015

Lawvere-Tierney Sheafification in Homotopy Type Theory

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

2015-06-29

Lawvere-Tierney Sheafification in Homotopy Type Theory

Lawvere-Tierney Sheafification in Homotopy Type Theory
Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France
29 June 2015

Lawvere-Tierney Sheafification in Homotopy Type Theory

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Introduction
The construction
Idea and context
Definitions
From types to separated types
From separated types to sheaves
Consequences
Future works
References

In set theory, one can change a model of ZFC into a new model of ZFC satisfying new principles, using the forcing construction [CD66].

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

In set theory, one can change a model of ZFC into a new model of ZFC satisfying new principles, using the forcing construction [CD66].

In set theory, one can change a model of ZFC into a new model of ZFC satisfying new principles, using the forcing construction [CD66].

Forcing has a topos-theoretic version: starting from a topos, one can construct a new topos satisfying some new principles, using the *sheafification* process [MM92].

In set theory, one can change a model of ZFC into a new model of ZFC satisfying new principles, using the forcing construction [CD66].

Forcing has a topos-theoretic version: starting from a topos, one can construct a new topos satisfying some new principles, using the *sheafification* process [MM92].

In set theory, one can change a model of ZFC into a new model of ZFC satisfying new principles, using the forcing construction [CD66].

Forcing has a topos-theoretic version: starting from a topos, one can construct a new topos satisfying some new principles, using the *sheafification* process [MM92].

Then (Grothendieck) sheafification has been extended to higher topos theory [Lur09].

In set theory, one can change a model of ZFC into a new model of ZFC satisfying new principles, using the forcing construction [CD66].

Forcing has a topos-theoretic version: starting from a topos, one can construct a new topos satisfying some new principles, using the *sheafification* process [MM92].

Then (Grothendieck) sheafification has been extended to higher topos theory [Lur09].

In set theory, one can change a model of ZFC into a new model of ZFC satisfying new principles, using the forcing construction [CD66].

Forcing has a topos-theoretic version: starting from a topos, one can construct a new topos satisfying some new principles, using the *sheafification* process [MM92].

Then (Grothendieck) sheafification has been extended to higher topos theory [Lur09].

We will present here a work-in-progress attempt to define an homotopy type theoretic version of this process.

In set theory, one can change a model of ZFC into a new model of ZFC satisfying new principles, using the forcing construction [CD66].

Forcing has a topos-theoretic version: starting from a topos, one can construct a new topos satisfying some new principles, using the *sheafification* process [MM92].

Then (Grothendieck) sheafification has been extended to higher topos theory [Lur09].

We will present here a work-in-progress attempt to define an homotopy type theoretic version of this process.

Introduction
The construction
Idea and context
Definitions
From types to separated types
From separated types to sheaves
Consequences
Future works
References

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Introduction
The construction
Idea and context
Definitions
From types to separated types
From separated types to sheaves
Consequences
Future works
References

Let's recall that in a topos, a Lawvere-Tierney topology is an idempotent map $\Omega \to \Omega$, preserving true and products. We notice that it corresponds to a left-exact modality on the subobject classifier Ω.

Then, the sheafification process extend this modality to the whole topos.

Let's recall that in a topos, a Lawvere-Tierney topology is an idempotent map $\Omega \to \Omega$, preserving true and products. We notice that it corresponds to a left-exact modality on the subobject classifier Ω.

Then, the sheafification process extend this modality to the whole topos.

1. Here, we call modality the same thing as in Type, but truncated to $n\text{-}\mathrm{Type}$
2. Sets in HoTT (Rijke-Spitters) tells us we can view HProp as an object classifier : Ω will HProp, and the topos HSet

Let's recall that in a topos, a Lawvere-Tierney topology is an idempotent map $\Omega \to \Omega$, preserving true and products. We notice that it corresponds to a left-exact modality on the subobject classifier Ω.

Then, the sheafification process extend this modality to the whole topos.

We want to follow this idea : from a left exact modality on HProp, we will define a left exact modality on all (finite) homotopy levels, by induction on this level.

Let's recall that in a topos, a Lawvere-Tierney topology is an idempotent map $\Omega \to \Omega$, preserving true and products. We notice that it corresponds to a left-exact modality on the subobject classifier Ω.

Then, the sheafification process extend this modality to the whole topos.

We want to follow this idea : from a left exact modality on HProp, we will define a left exact modality on all (finite) homotopy levels, by induction on this level.

1. Here, we call modality the same thing as in Type, but truncated to n-Type
2. Sets in HoTT (Rijke-Spitters) tells us we can view HProp as an object classifier : Ω will HProp, and the topos HSet

Recall : Modalities

We use the same notion of modalities as in [Uni13, Section 7.7], but restricted to be on n-truncated types.

Definition

Let $n \geq -1$ be a truncation index. A left exact modality at level n is the data of

(i) *A predicate $P : \mathrm{Type}_n \to \mathrm{HProp}$*

(ii) *For every n-truncated type A, a n-truncated type $\bigcirc A$ such that $P(\bigcirc A)$*

(iii) *For every n-truncated type A, a map $\eta_A : A \to \bigcirc A$*

such that

(iv) *For every n-truncated types A and B, if $P(B)$ then*

$$\begin{cases} (\bigcirc A \to B) & \to & (A \to B) \\ f & \mapsto & f \circ \eta_A \end{cases}$$

is an equivalence.

Recall : Modalities
We use the same notion of modalities as in [Uni13, Section 7.7], but restricted to be on n-truncated types.
Definition
Let $n \geq -1$ be a truncation index. A left exact modality at level n is the data of
(i) A predicate $P : \mathrm{Type}_n \to \mathrm{HProp}$
(ii) For every n-truncated type A, a n-truncated type $\bigcirc A$ such that $P(\bigcirc A)$
(iii) For every n-truncated type A, a map $\eta_A : A \to \bigcirc A$
such that
(iv) For every n-truncated types A and B, if $P(B)$ then
$\begin{cases} (\bigcirc A \to B) & \to & (A \to B) \\ f & \mapsto & f \circ \eta_A \end{cases}$
is an equivalence.

(v) for any $A : \text{Type}_n$ and $B : A \to \text{Type}_n$ such that $P(A)$ and $\prod_{x:A} P(Bx)$, then $P\left(\sum_{x:A} B(x)\right)$

(vi) for any $A : \text{Type}_n$ and $x, y : A$, if $\bigcirc A$ is contractible, then $\bigcirc(x = y)$ is contractible.

Conditions (i) to (iv) define a *reflective subuniverse*, (i) to (v) a *modality*.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

(v) for any $A : \text{Type}_n$ and $B : A \to \text{Type}_n$ such that $P(A)$ and $\prod_{x:A} P(Bx)$, then $P\left(\sum_{x:A} B(x)\right)$
(vi) for any $A : \text{Type}_n$ and $x, y : A$, if $\bigcirc A$ is contractible, then $\bigcirc(x = y)$ is contractible.
Conditions (i) to (iv) define a *reflective subuniverse*, (i) to (v) a *modality*.

Recall: Sheafification in topos

Let j be a Lawvere-Tierney topology on a topos $\mathcal{T}$, with subobject classifier Ω.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Recall: Sheafification in topos

Let j be a Lawvere-Tierney topology on a topos $\mathcal{T}$, with subobject classifier Ω.

Recall: Sheafification in topos

Let j be a Lawvere-Tierney topology on a topos $\mathcal{T}$, with subobject classifier Ω.

$$\begin{array}{ccc} T & \xrightarrow{\{\cdot\}_T} & \Omega^T \\ & & \big\downarrow{\scriptstyle j^T} \\ & & (\Omega_j)^T \end{array}$$

Send T to Ω^T via the singleton map, then postcompose with $j : \Omega \to \Omega_j$

Recall: Sheafification in topos

Let j be a Lawvere-Tierney topology on a topos $\mathcal{T}$, with subobject classifier Ω.

$$\begin{array}{ccc} T & \xrightarrow{\{\cdot\}_T} & \Omega^T \\ & & \big\downarrow{\scriptstyle j^T} \\ & & (\Omega_j)^T \end{array}$$

Send T to Ω^T via the singleton map, then postcompose with $j : \Omega \to \Omega_j$

Recall: Sheafification in topos

Let j be a Lawvere-Tierney topology on a topos $\mathcal{T}$, with subobject classifier Ω.

$$\begin{array}{ccc} T & \xrightarrow{\{\cdot\}_T} & \Omega^T \\ {\scriptstyle \mu_T}\downarrow & & \downarrow{\scriptstyle j^T} \\ \mathrm{Im}\left(j^T \circ \{\cdot\}_T\right) & \xhookrightarrow{\text{mono}} & (\Omega_j)^T \end{array}$$

Compute the image of this map: it is a subobject of $(\Omega_j)^T$

Recall: Sheafification in topos

Let j be a Lawvere-Tierney topology on a topos $\mathcal{T}$, with subobject classifier Ω.

$$\begin{array}{ccc} T & \xrightarrow{\{\cdot\}_T} & \Omega^T \\ {\scriptstyle \mu_T}\downarrow & & \downarrow{\scriptstyle j^T} \\ \mathrm{Im}\left(j^T \circ \{\cdot\}_T\right) & \xhookrightarrow{\text{mono}} & (\Omega_j)^T \end{array}$$

Compute the image of this map: it is a subobject of $(\Omega_j)^T$

Recall: Sheafification in topos

Let j be a Lawvere-Tierney topology on a topos $\mathcal{T}$, with subobject classifier Ω.

$$\begin{array}{ccc}
T & \xrightarrow{\{\cdot\}_T} & \Omega^T \\
\downarrow{\scriptstyle \mu_T} & & \downarrow{\scriptstyle j^T} \\
\mathrm{Im}\left(j^T \circ \{\cdot\}_T\right) & \xhookrightarrow{\text{mono}} & (\Omega_j)^T \\
& \searrow \quad \nearrow & \\
& \mathfrak{a}(T) \stackrel{def}{=} \overline{\mathrm{Im}\left(j^T \circ \{\cdot\}_T\right)} &
\end{array}$$

Close this subobject

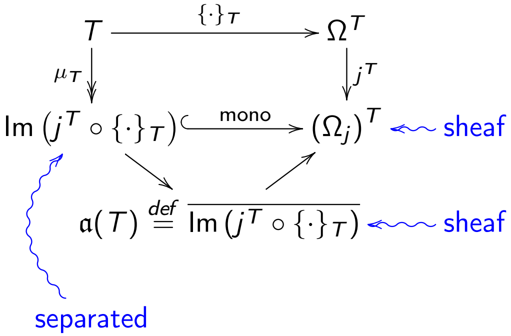
Recall: Sheafification in topos

Let j be a Lawvere-Tierney topology on a topos $\mathcal{T}$, with subobject classifier Ω.

Close this subobject

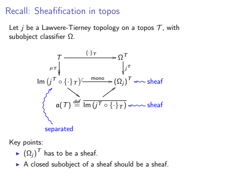
Recall: Sheafification in topos

Let j be a Lawvere-Tierney topology on a topos $\mathcal{T}$, with subobject classifier Ω.



Key points:

- $(\Omega_j)^T$ has to be a sheaf.
- A closed subobject of a sheaf should be a sheaf.



The predicate "is n-modal" on homotopy level n will be "is a Lawvere-Tierney n-sheaf", and the required modality will be the n-sheafification.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

The predicate "is n-modal" on homotopy level n will be "is a Lawvere-Tierney n-sheaf", and the required modality will be the n-sheafification.

1. We do this by induction on the homotopy level n. At the moment, we don't know how to extend it to not truncated types
2. From *Sets in HoTT* (Rijke-Spitters), we know that n-Type can be seen as an object classifier. We will use this property ;
   HProp will be a common object classifier for all levels, and n-Type will be an object classifier for $-$-$\mathrm{Type}(n+1)$ sheafification.

Context

We work in homotopy type theory, i.e, Martin-Löf type theory, with univalence axiom (thus functional extensionality) and higher inductive types (although at the moment, we only need propositional truncation).

Context

We work in homotopy type theory, i.e, Martin-Löf type theory, with univalence axiom (thus functional extensionality) and higher inductive types (although at the moment, we only need propositional truncation).

Context

Let $\bigcirc_{-1}$ be a left exact modality on HProp (homotopy level -1), $n \geqslant -1$ a truncation index, and $\bigcirc_n$ a left exact modality on n-Type (homotopy level n), coherent with $\bigcirc_{-1}$:

If T : HProp, then $\bigcirc_n T = \bigcirc_{-1} T$ where we still note T the image of T via the inclusion HProp $\hookrightarrow$ n-Type.

Context

Let $\bigcirc_{-1}$ be a left exact modality on HProp (homotopy level -1), $n \geqslant -1$ a truncation index, and $\bigcirc_n$ a left exact modality on n-Type (homotopy level n), coherent with $\bigcirc_{-1}$:

If T : HProp, then $\bigcirc_n T = \bigcirc_{-1} T$ where we still note T the image of T via the inclusion HProp $\hookrightarrow$ n-Type.

There, by cumulativity, T can be seen as a n-Type.

Questions

When generalizing construction in topos, several questions arises:

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Questions
When generalizing construction in topos, several questions arises:

Questions

When generalizing construction in topos, several questions arises:

- Do we generalize subobjects as n-subobjects (maps with n-truncated fibers) or (-1)-subobjects (embeddings) ?

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Questions

When generalizing construction in topos, several questions arises:

- Do we generalize subobjects as n-subobjects (maps with n-truncated fibers) or (-1)-subobjects (embeddings) ?

Questions

When generalizing construction in topos, several questions arises:

- Do we generalize subobjects as n-subobjects (maps with n-truncated fibers) or (-1)-subobjects (embeddings) ?
- The proof involves kernel pair of a surjection. How to generalize it ?

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Questions

When generalizing construction in topos, several questions arises:

- Do we generalize subobjects as n-subobjects (maps with n-truncated fibers) or (-1)-subobjects (embeddings) ?
- The proof involves kernel pair of a surjection. How to generalize it ?

Questions

When generalizing construction in topos, several questions arises:

- Do we generalize subobjects as n-subobjects (maps with n-truncated fibers) or (-1)-subobjects (embeddings) ?
- The proof involves kernel pair of a surjection. How to generalize it ?
- Do we use usual image, or a n-image arising from n-connected/n-truncated factorization system ?

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Questions

When generalizing construction in topos, several questions arises:

- Do we generalize subobjects as n-subobjects (maps with n-truncated fibers) or (-1)-subobjects (embeddings) ?
- The proof involves kernel pair of a surjection. How to generalize it ?
- Do we use usual image, or a n-image arising from n-connected/n-truncated factorization system ?

Questions

When generalizing construction in topos, several questions arises:

- Do we generalize subobjects as n-subobjects (maps with n-truncated fibers) or (-1)-subobjects (embeddings) ? Solved
- The proof involves kernel pair of a surjection. How to generalize it ? In progress
- Do we use usual image, or a n-image arising from n-connected/n-truncated factorization system ? Solved

Questions

When generalizing construction in topos, several questions arises:

- Do we generalize subobjects as n-subobjects (maps with n-truncated fibers) or (-1)-subobjects (embeddings) ? Solved
- The proof involves kernel pair of a surjection. How to generalize it ? In progress
- Do we use usual image, or a n-image arising from n-connected/n-truncated factorization system ? Solved

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Dense subobject I

Definition

Let E be a type. The closure *of a subobject of E with m-truncated homotopy fibers (or m-subobject of E, for short), classified by $\chi : E \to m\text{-Type}$, is the m-subobject of E classified by $\bigcirc_m \circ \chi$.*
An m-subobject of E classified by χ is said to be closed *in E if it is equal to its closure, i.e, $\chi = \bigcirc_m \circ \chi$.*

Practically, a m-subobject of E is just $\{e : E \ \& \ \chi \ e\}$, and its closure is $\{e : E \ \& \ \bigcirc_m (\chi \ e)\}$.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Dense subobject I
Definition
Let E be a type. The closure of a subobject of E with m-truncated homotopy fibers (or m-subobject of E, for short), classified by $\chi : E \to m\text{-Type}$, is the m-subobject of E classified by $\bigcirc_m \circ \chi$.
An m-subobject of E classified by χ is said to be closed in E if it is equal to its closure, i.e, $\chi = \bigcirc_m \circ \chi$.
Practically, a m-subobject of E is just $\{e : E \ \& \ \chi \ e\}$, and its closure is $\{e : E \ \& \ \bigcirc_m (\chi \ e)\}$.

1. The closure operator is just postcomposition of characteristic with the modality.
2. A is closed in E if its closure is E.

Dense subobject II

Definition

Let E be a type, and $\chi : E \to m\text{-Type}$. The m-subobject of E classified by χ is dense *in E when its $\bigcirc_m$-closure is equivalent to χ_E, i.e,*

$$\forall e : E,\ \ \bigcirc_m (\chi\ e) \simeq \mathbf{1}.$$

Practically, a m-subobject A of E is dense if, from the $\bigcirc_m$ point of view, you cannot make a difference between A and E.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Dense subobject II
Definition
Let E be a type, and $\chi : E \to m\text{-Type}$. The m-subobject of E classified by χ is dense in E when its $\bigcirc_m$-closure is equivalent to χ_E, i.e,
$\forall e : E,\ \ \bigcirc_m (\chi\ e) \simeq \mathbf{1}.$
Practically, a m-subobject A of E is dense if, from the $\bigcirc_m$ point of view, you cannot make a difference between A and E.

Restriction

Definition

For any type E, characteristic map $\chi : E \to m\text{-Type}$ and $F : (n+1)\text{-Type}$, we define

$$\Phi_E^{\chi,m} : (E \to F) \to (\{e : E \ \& \ \chi\ e\} \to F)$$

as the map sending an arrow $f : E \to F$ to its restriction $f \circ \pi_1$.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Restriction
Definition
For any type E, characteristic map $\chi : E \to m\text{-Type}$ and $F : (n+1)\text{-Type}$, we define
$\Phi_E^{\chi,m} : (E \to F) \to (\{e : E \ \& \ \chi\ e\} \to F)$
as the map sending an arrow $f : E \to F$ to its restriction $f \circ \pi_1$.

Requirements

We want a predicate on $(n+1)$-Type, which we call *sheaf property*, satisfying:

- if $\bigcirc_n$ is the identity modality, then everybody should be a sheaf
- a closed (-1)-subobject of a sheaf should be a sheaf
- the type of modal n-Type should be a sheaf
- if T is a sheaf, then $X \to T$ should be a sheaf, for any X

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Requirements
We want a predicate on $(n+1)$-Type, which we call *sheaf property*, satisfying:
- if $\bigcirc_n$ is the identity modality, then everybody should be a sheaf
- a closed (-1)-subobject of a sheaf should be a sheaf
- the type of modal n-Type should be a sheaf
- if T is a sheaf, then $X \to T$ should be a sheaf, for any X

Requirements

We want a predicate on $(n+1)$-Type, which we call *sheaf property*, satisfying:

- if $\bigcirc_n$ is the identity modality, then everybody should be a sheaf
- a closed (-1)-subobject of a sheaf should be a sheaf
- the type of modal n-Type should be a sheaf
- if $T : X \to (n+1)$-Type such that any $T\,x$ is a sheaf, then $\prod_{x:X} T\,x$ should be a sheaf.

Requirements

We want a predicate on $(n+1)$-Type, which we call *sheaf property*, satisfying:

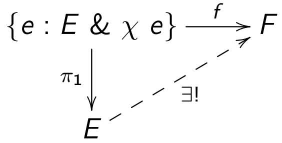
- if $\bigcirc_n$ is the identity modality, then everybody should be a sheaf
- a closed (-1)-subobject of a sheaf should be a sheaf
- the type of modal n-Type should be a sheaf
- if $T : X \to (n+1)$-Type such that any $T\,x$ is a sheaf, then $\prod_{x:X} T\,x$ should be a sheaf.

Sheaves

Following the topos-theoretic idea, we use:

Definition (Sheaves)

A type F of $(n+1)$-Type is a $(n+1)$-sheaf for any type E and all dense (-1)-subobject $\chi : E \to (-1)$-Type, $\Phi_E^{\chi,-1}$ is an equivalence. In other words, the dotted arrow exists and is unique.



Lawvere-Tierney Sheafification in Homotopy Type Theory

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Sheaves
Following the topos-theoretic idea, we use:
Definition (Sheaves)
A type F of (n+1)-Type is a (n+1)-sheaf for any type E and all dense (−1)-subobject χ : E → (−1)-Type, Φ is an equivalence. In other words, the dotted arrow exists and is unique.

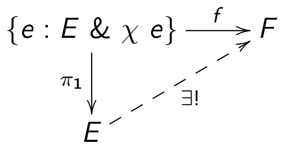
1. Here, we take (-1)-subobjects, because we want every type to be a sheaf for the identity modality.
2. The conditions are not satisfied that way; sheaves are not stable by dependent products.

Sheaves

Following the topos-theoretic idea, we use:

Definition (Sheaves)

A type F of $(n+1)$-Type is a $(n+1)$-sheaf if it is separated, and for any type E and all dense (-1)-subobject $\chi : E \to (-1)\text{-Type}$, $\Phi_E^{\chi,-1}$ is an equivalence. In other words, the dotted arrow exists and is unique.



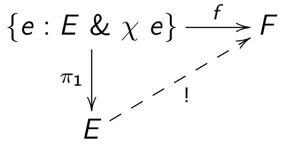
Sheaves
Following the topos-theoretic idea, we use:
Definition (Sheaves)
A type F of (n+1)-Type is a (n+1)-sheaf if it is separated, and for any type E and all dense (−1)-subobject χ : E → (−1)-Type, Φ is an equivalence. In other words, the dotted arrow exists and is unique.

1. Here, we take (-1)-subobjects, because we want every type to be a sheaf for the identity modality.
2. The conditions are not satisfied that way; sheaves are not stable by dependent products.

Separated type

Definition (Separated Type)

A type F in $(n+1)$-Type is separated *if for any type E, and all dense* n-subobject $\chi : E \to n\text{-Type}$, $\Phi_E^{\chi,n}$ *is an embedding. In other words, the dotted arrow, if exists, is unique.*



Lawvere-Tierney Sheafification in Homotopy Type Theory

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Separated type

Definition (Separated Type)

A type F in (n+1)-Type is separated if for any type E, and all dense n-subobject χ : E → n-Type, Φ_E^{χ,n} is an embedding. In other words, the dotted arrow, if exists, is unique.

Two steps

We will proceed in two steps:

(i) *separation:* From any T in $(n+1)\text{-Type}$, we construct its *free* separated object $\square_{n+1} T$.

(ii) *completion:* We add what is missing for the free separated type to be a sheaf by using closure.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Two steps

We will proceed in two steps:
(i) *separation:* From any T in $(n+1)\text{-Type}$, we construct its *free* separated object $\square_{n+1} T$.
(ii) *completion:* We add what is missing for the free separated type to be a sheaf by using closure.

1. Not equivalent with + construction. We define the free separated object, while Grothendieck not.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Introduction
The construction
Idea and context
Definitions
From types to separated types
From separated types to sheaves
Consequences
Future works
References

Let $T : (n+1)\text{-Type}$. We define $\square_{n+1}\, T$ as the image of $\bigcirc_n^T \circ \{\cdot\}_T$, as in

$$\begin{array}{ccc} T & \xrightarrow{\{\cdot\}_T} & n\text{-Type}^T \\ {\scriptstyle \mu_T}\downarrow & & \downarrow{\scriptstyle \bigcirc_n^T} \\ \square_{n+1}\, T & \longrightarrow & \left(n\text{-Type}^{\bigcirc}\right)^T \end{array}$$

where $\{\cdot\}_T$ is the singleton map $\lambda(t : T),\ \lambda(t' : T),\ t = t'$.

Lawvere-Tierney Sheafification in Homotopy Type Theory

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Let $T : (n+1)\text{-Type}$. We define $\square_{n+1}\, T$ as the image of $\bigcirc_n^T \circ \{\cdot\}_T$, as in
where $\{\cdot\}_T$ is the singleton map $\lambda(t : T),\ \lambda(t' : T),\ t = t'$.

We note that, as μ_T is the surjection-embedding factorization, μ_T is indeed a surjection.

Let $T : (n+1)\text{-Type}$. We define $\square_{n+1}\ T$ as the image of $\bigcirc_n^T \circ \{\cdot\}_T$, as in

$$\begin{array}{ccc} T & \xrightarrow{\{\cdot\}_T} & n\text{-Type}^T \\ {\scriptstyle \mu_T}\downarrow & & \downarrow{\scriptstyle \bigcirc_n^T} \\ \square_{n+1}\ T & \longrightarrow & \left(n\text{-Type}^{\bigcirc}\right)^T \end{array}$$

where $\{\cdot\}_T$ is the singleton map $\lambda(t : T),\ \lambda(t' : T),\ t = t'$.
$\square_{n+1}\ T$ can be given explicitly by

$$\begin{aligned} \square_{n+1}\ T &\overset{def}{=} \mathrm{Im}(\lambda\ t : T,\ \lambda\ t',\ \bigcirc_n (t = t')) \\ &\overset{def}{=} \textstyle\sum_{u : T \to n\text{-Type}^{\bigcirc}} \left\| \sum_{a : T} (\lambda t,\ \bigcirc_n (a = t)) = u \right\| . \end{aligned}$$

Let $T : (n+1)\text{-Type}$. We define $\square_{n+1}\ T$ as the image of $\bigcirc_n^T \circ \{\cdot\}_T$, as in
where $\{\cdot\}_T$ is the singleton map $\lambda(t : T),\ \lambda(t' : T),\ t = t'$.
$\square_{n+1}\ T$ can be given explicitly by

We note that, as μ_T is the surjection-embedding factorization, μ_T is indeed a surjection.

At first, we prove that:

Proposition

For any $T : (n+1)\text{-Type}$, $\Box_{n+1}\, T$ *is separated.*

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

1. That's indeed the least we can ask.

At first, we prove that:
Proposition
For any $T : (n+1)\text{-Type}$, $\Box_{n+1}\, T$ is separated.

At first, we prove that:

Proposition

For any $T : (n+1)\text{-Type}$, $\Box_{n+1}\, T$ *is separated.*

Then, we want

Theorem

$(\Box_{n+1}, \mu)$ *defines a modality on* $(n+1)\text{-Type}$.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

At first, we prove that:
Proposition
For any T : (n+1)-Type, □n+1 T is separated.
Then, we want
Theorem
(□n+1, μ) defines a modality on (n+1)-Type.

1. That's indeed the least we can ask.
2. This actually is the hard part of the construction ; especially the universal property for the reflective subuniverse.

Sketch of proof

In topoi, the proof goes this way:

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Sketch of proof
In topoi, the proof goes this way:

Sketch of proof

In topoi, the proof goes this way:

- μ_T is a surjection, thus it coequalizes its kernel pair

$$T \times_{\Box_{n+1} T} T \overset{\pi_1}{\underset{\pi_2}{\rightrightarrows}} T \xrightarrow{\mu_T} \Box_{n+1} T$$

Sketch of proof

In topoi, the proof goes this way:
- μ_T is a surjection, thus it coequalizes its kernel pair

$$T \times_{\Box_{n+1} T} T \overset{\pi_1}{\underset{\pi_2}{\rightrightarrows}} T \xrightarrow{\mu_T} \Box_{n+1} T$$

Sketch of proof

In topoi, the proof goes this way:

- μ_T is a surjection, thus it coequalizes its kernel pair

$$T \times_{\Box_{n+1} T} T \overset{\pi_1}{\underset{\pi_2}{\rightrightarrows}} T \xrightarrow{\mu_T} \Box_{n+1} T$$

- Then $T \times_{\Box_{n+1} T} T = \overline{\Delta}$, where $\Delta = \{(x, y) : T^2 \ \& \ x = y\}$. The following is a coequalizer

$$\overline{\Delta} \overset{\pi_1}{\underset{\pi_2}{\rightrightarrows}} T \xrightarrow{\mu_T} \Box_{n+1} T$$

Sketch of proof
In topoi, the proof goes this way:
μ_T is a surjection, thus it coequalizes its kernel pair
$T \times_{\Box_{n+1} T} T \rightrightarrows T \xrightarrow{\mu_T} \Box_{n+1} T$
Then $T \times_{\Box_{n+1} T} T = \overline{\Delta}$, where $\Delta = \{(x, y) : T^2 \ \& \ x = y\}$. The following is a coequalizer
$\overline{\Delta} \rightrightarrows T \xrightarrow{\mu_T} \Box_{n+1} T$

Sketch of proof

Then, if Q is any separated type and $f : T \to Q$, it makes the diagram

$$\overline{\Delta} \underset{\pi_2}{\overset{\pi_1}{\rightrightarrows}} T \xrightarrow{f} Q$$

commute, thus f factors through $\square_{n+1} T$.

Sketch of proof

Then, if Q is any separated type and $f : T \to Q$, it makes the diagram

$$\overline{\Delta} \underset{\pi_2}{\overset{\pi_1}{\rightrightarrows}} T \xrightarrow{f} Q$$

commute, thus f factors through $\square_{n+1} T$.

We would like to use the same idea, replacing the kernel pair by the Čech nerve.

At the moment, we only assumed as an axiom that surjections are colimits of their Čech nerves, seen as graphs. It allows us to finish the proof.

We would like to use the same idea, replacing the kernel pair by the Čech nerve.

At the moment, we only assumed as an axiom that surjections are colimits of their Čech nerves, seen as graphs. It allows us to finish the proof.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Introduction
The construction
Idea and context
Definitions
From types to separated types
From separated types to sheaves
Consequences
Future works
References

For any T in $(n+1)$-Type, $\bigcirc_{n+1} T$ is defined as the closure of $\square_{n+1}\ T$, seen as a subobject of $T \to n$-$\mathrm{Type}^{\bigcirc}$. $\bigcirc_{n+1} T$ can be given explicitly by

$$\bigcirc_{n+1} T \stackrel{def}{=} \sum_{u:T \to n\text{-}\mathrm{Type}^{\bigcirc}} \bigcirc_{-1} \left\| \sum_{a:T} (\lambda t,\ \ \bigcirc_n (a = t)) = u \right\|.$$

For any T in $(n+1)$-Type, $\bigcirc_{n+1} T$ is defined as the closure of $\square_{n+1}\ T$, seen as a subobject of $T \to n$-$\mathrm{Type}^{\bigcirc}$. $\bigcirc_{n+1} T$ can be given explicitly by
$$\bigcirc_{n+1} T \stackrel{def}{=} \sum_{u:T \to n\text{-}\mathrm{Type}^{\bigcirc}} \bigcirc_{-1} \left\| \sum_{a:T} (\lambda t,\ \ \bigcirc_n (a = t)) = u \right\|.$$

As above, we first prove that:

Proposition

For any $T : (n+1)\text{-Type}$, $\bigcirc_{n+1} T$ is a sheaf.

It is true because of the requirement we asked about sheaves:

Lemma

Let $X : (n+1)\text{-Type}$ and U be a sheaf. If X embeds in U, and is closed in U, then X is a sheaf.

As above, we first prove that:
Proposition
For any $T : (n+1)\text{-Type}$, $\bigcirc_{n+1} T$ is a sheaf.
It is true because of the requirement we asked about sheaves:
Lemma
Let $X : (n+1)\text{-Type}$ and U be a sheaf. If X embeds in U, and is closed in U, then X is a sheaf.

1. Again, we need this.

As above, we first prove that:

Proposition

For any $T : (n+1)\text{-Type}$, $\bigcirc_{n+1} T$ *is a sheaf.*

It is true because of the requirement we asked about sheaves:

Lemma

Let $X : (n+1)\text{-Type}$ *and* U *be a sheaf. If* X *embeds in* U*, and is closed in* U*, then* X *is a sheaf.*

Then:

Theorem

$(\bigcirc_{n+1}, \nu)$ *defines a left-exact modality.*

As above, we first prove that:
Proposition
For any $T : (n+1)\text{-Type}$, $\bigcirc_{n+1} T$ is a sheaf.
It is true because of the requirement we asked about sheaves:
Lemma
Let $X : (n+1)\text{-Type}$ and U be a sheaf. If X embeds in U, and is closed in U, then X is a sheaf.
Then:
Theorem
$(\bigcirc_{n+1}, \nu)$ defines a left-exact modality.

1. Again, we need this.
2. This time, it's pretty easy...

Sketch of proof

Let $T, Q : (n+1)\text{-}\mathrm{Type}$ such that Q is a sheaf. Let $f : T \to Q$. Because Q is a sheaf, it is in particular separated; thus we can extend f to $\square_{n+1} f : \square_{n+1} T \to Q$.

Sketch of proof

Let $T, Q : (n+1)\text{-}\mathrm{Type}$ such that Q is a sheaf. Let $f : T \to Q$. Because Q is a sheaf, it is in particular separated; thus we can extend f to $\square_{n+1} f : \square_{n+1} T \to Q$.

Sketch of proof

Let $T, Q : (n+1)\text{-Type}$ such that Q is a sheaf. Let $f : T \to Q$. Because Q is a sheaf, it is in particular separated; thus we can extend f to $\square_{n+1} f : \square_{n+1} T \to Q$.

But as $\bigcirc_{n+1} T$ is the closure of $\square_{n+1} T$, $\square_{n+1} T$ is dense into $\bigcirc_{n+1} T$, so the sheaf property of Q allows to extend $\square_{n+1} f$ to $\bigcirc_{n+1} f : \bigcirc_{n+1} T \to Q$.
As all these steps are universal, the composition is.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Sketch of proof

Let $T, Q : (n+1)\text{-Type}$ such that Q is a sheaf. Let $f : T \to Q$. Because Q is a sheaf, it is in particular separated; thus we can extend f to $\square_{n+1} f : \square_{n+1} T \to Q$.

But as $\bigcirc_{n+1} T$ is the closure of $\square_{n+1} T$, $\square_{n+1} T$ is dense into $\bigcirc_{n+1} T$, so the sheaf property of Q allows to extend $\square_{n+1} f$ to $\bigcirc_{n+1} f : \bigcirc_{n+1} T \to Q$.
As all these steps are universal, the composition is.

Sketch of proof

Let $T, Q : (n+1)\text{-Type}$ such that Q is a sheaf. Let $f : T \to Q$. Because Q is a sheaf, it is in particular separated; thus we can extend f to $\Box_{n+1} f : \Box_{n+1} T \to Q$.

But as $\bigcirc_{n+1} T$ is the closure of $\Box_{n+1} T$, $\Box_{n+1} T$ is dense into $\bigcirc_{n+1} T$, so the sheaf property of Q allows to extend $\Box_{n+1} f$ to $\bigcirc_{n+1} f : \bigcirc_{n+1} T \to Q$.
As all these steps are universal, the composition is.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Sketch of proof
Let $T, Q : (n+1)\text{-Type}$ such that Q is a sheaf. Let $f : T \to Q$. Because Q is a sheaf, it is in particular separated; thus we can extend f to $\Box_{n+1} f : \Box_{n+1} T \to Q$.
But as $\bigcirc_{n+1} T$ is the closure of $\Box_{n+1} T$, $\Box_{n+1} T$ is dense into $\bigcirc_{n+1} T$, so the sheaf property of Q allows to extend $\Box_{n+1} f$ to $\bigcirc_{n+1} f : \bigcirc_{n+1} T \to Q$.
As all these steps are universal, the composition is.

again, the modality thing is just technical, and the left-exactness comes from the compatibility.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Introduction
The construction
Idea and context
Definitions
From types to separated types
From separated types to sheaves
Consequences
Future works
References

Starting from the left-exact modality $\bigcirc_{-1} P = \neg\neg P$, this allows us to build a model satisfying excluded middle for HProp, without axiom.

Starting from the left-exact modality $\bigcirc_{-1} P = \neg\neg P$, this allows us to build a model satisfying excluded middle for HProp, without axiom.

Starting from the left-exact modality $\bigcirc_{-1} P = \neg\neg P$, this allows us to build a model satisfying excluded middle for HProp, without axiom.

With the same modality $\neg\neg$, we hope to be able to formalize the proof of independance of continuum hypothesis (actually, just the consistance of $\neg \mathrm{HC}$).

Starting from the left-exact modality $\bigcirc_{-1} P = \neg\neg P$, this allows us to build a model satisfying excluded middle for HProp, without axiom.

With the same modality $\neg\neg$, we hope to be able to formalize the proof of independance of continuum hypothesis (actually, just the consistance of $\neg \mathrm{HC}$).

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Introduction
The construction
Idea and context
Definitions
From types to separated types
From separated types to sheaves
Consequences
Future works
References

Universes

The construction can be written inductively:

$\bigcirc : \forall\ (n : nat),\ n\text{-Type} \to n\text{-Type}$

- $\bigcirc_{-1}$ is a left exact modality on HProp
- $\bigcirc_{n+1} \stackrel{def}{=} \lambda T : (n+1)\text{-Type},$

$$\sum_{u : T \to n\text{-Type}^{\bigcirc}} \bigcirc_{-1} \left\| \sum_{a : T} u = (\lambda t,\ \ \bigcirc_n (a = t)) \right\|$$

Here , the universe level increases strictly at each step, hence it is impossible to take the fixpoint: we would need universes to be indexed by (non-finite) ordinals.

Universes

The construction can be written inductively:

$\bigcirc : \forall\ (n : nat),\ n\text{-Type} \to n\text{-Type}$
- $\bigcirc_{-1}$ is a left exact modality on HProp
- $\bigcirc_{n+1} \stackrel{def}{=} \lambda T : (n+1)\text{-Type},$

$$\sum_{u : T \to n\text{-Type}^{\bigcirc}} \bigcirc_{-1} \left\| \sum_{a : T} u = (\lambda t,\ \ \bigcirc_n (a = t)) \right\|$$

Here , the universe level increases strictly at each step, hence it is impossible to take the fixpoint: we would need universes to be indexed by (non-finite) ordinals.

Čech nerve

The main step to finish the construction is to define Čech nerve in HoTT, as well as the computation of their colimits.

We will rather try to define general simplicial objects.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France

Čech nerve

The main step to finish the construction is to define Čech nerve in HoTT, as well as the computation of their colimits.

We will rather try to define general simplicial objects.

Simplicial types

Hugo Herbelin [Her14] gives an inductive definition of semi-simplicial types, which can probably be adapted to define simplicial types, but is quite unusable for n-types with $n \geqslant 4$.

Make a joke about the previous talk...

Simplicial types

Hugo Herbelin [Her14] gives an inductive definition of semi-simplicial types, which can probably be adapted to define simplicial types, but is quite unusable for n-types with $n \geqslant 4$.

Homotopy type system

One idea is to use homotopy type system, introduced by V.V., to see Type as a model category. Then, we should be able to formalize homotopy colimits in type theory.

Homotopy type system

One idea is to use homotopy type system, introduced by V.V., to see Type as a model category. Then, we should be able to formalize homotopy colimits in type theory.

Introduction
The construction
Idea and context
Definitions
From types to separated types
From separated types to sheaves
Consequences
Future works
References

- P.J. Cohen and M. Davis, *Set theory and the continuum hypothesis*, WA Benjamin New York, 1966.
- Hugo Herbelin, *A dependently-typed construction of semi-simplicial types*, Mathematical Structures in Computer Science (2014), to appear.
- Jacob Lurie, *Higher topos theory*, Annals of mathematics studies, Princeton University Press, Princeton, N.J., Oxford, 2009.
- Saunders MacLane and Ieke Moerdijk, *Sheaves in geometry and logic*, Springer-Verlag, 1992.
- Univalent Foundations Project, *Homotopy type theory: Univalent foundations for mathematics*, `http://homotopytypetheory.org/book`, 2013.

Kevin Quirin and Nicolas Tabareau
Inria, Mines de Nantes
Nantes, France