

Internship report M2  
Parisian Master of Research in Computer Science Level 2

Non-singleton Elimination

**Soudant Léo**  
fourth year student at ENS Paris Saclay

2025

Supervisor  
**Pierre-Merie Pédrot**, chargé de recherche at Inria Rennes-Bretagne-Atlantique,

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	The outline . . . . .	2
<b>2</b>	<b>Sheaves</b>	<b>2</b>
2.1	Topoi and sheaves in topoi . . . . .	2
2.2	Sheaves in type theory . . . . .	2
<b>3</b>	<b>Models</b>	<b>3</b>
<b>4</b>	<b>System T and MLTT</b>	<b>3</b>
4.1	MLTT . . . . .	3
4.2	System T . . . . .	4
<b>5</b>	<b>Meta-informations</b>	<b>4</b>
5.1	Time expenditure . . . . .	4
5.2	Difficulties . . . . .	4
<b>6</b>	<b>Conclusion</b>	<b>4</b>
<b>A</b>	<b>Dummy</b>	<b>5</b>

# 1 Introduction

## 1.1 The outline

## 2 Sheaves

### 2.1 Topoï and sheaves in topoï

Proofs, results and details for this section can often be found in *Sheaves in geometry and logic: A first introduction to topos theory* by Saunders MacLane and Ieke Moerdijk [1].

**Definition 2.1** (Subobject). *In a category, a subobject of  $X$  is an equivalence class of monomorphism  $m : A \rightarrowtail X$ , where the equivalence comes from the preorder where  $m : A \rightarrowtail X$  is smaller than  $m' : A' \rightarrowtail X$  when there is a map  $f : A \rightarrow A'$  with  $m' \circ f = m$ .*

We deduce a presheaf **Sub** where **Sub**( $X$ ) is the the set of subobjects of  $X$ , and **Sub**( $f$ ) : **Sub**( $Y$ )  $\rightarrow$  **Sub**( $X$ ) sends  $m : A \rightarrow Y$  to its pullback by  $f : X \rightarrow Y$ .

**Definition 2.2** (Topos). *A topos is a cartesian closed category with all finite limits and a subobject classifier  $\Omega$  and an isomorphism **Sub**( $X$ )  $\cong$  **Hom**( $X, \Omega$ ) natural in  $X$ .*

We note that topoï also have finite colimits.

A topos serves to give models of intuitionistic logic in classical mathematical language. It has an internal logic which is higher order.

For example, **Set** is a topos, and given a topos  $\mathcal{E}$ ,  $\mathcal{E}/X$ , the category of maps with codomain  $X$  and commuting triangles, as well as  $\mathcal{E}^{\mathbf{C}^{op}}$ , the category of contravariant functors from a small category **C** and natural transformation, are all topoï. In particular categories of presheaves are topoï, and correspond to Kripke models.

The subobject classifier  $\Omega$  is equipped with an internal meet-semilattice structure inherited from the meet-semilattice structure on each **Sub**( $X$ ), which is natural in  $X$ .

**Definition 2.3** (Lawvere-Tierney topology). *A Lawvere-Tierney topology is a left exact idempotent monad  $j$  on the internal meet-semilattice on  $\Omega$ .*

- $id_{\Omega} \leq j$ ,
- $j \circ j \leq j$
- $j \circ \wedge = \wedge \circ j \times j$

From a topology  $j$  we extract a closure operator  $J_X$  of **Sub**( $X$ ) for any  $X$ .

**Definition 2.4** (Dense subobject). *A subobject  $U$  of  $X$  is dense if  $J_X U = X$*

A topology can be lifted to a left exact idempotent monad on the entirety of the topos, the sheafification monad.

**Definition 2.5** ( $j$ -Sheaf in topos). *An object  $F$  is a  $j$ -sheaf in a topos if for any dense subobject  $U$  of any object  $X$ , the morphism **Hom**( $X, F$ )  $\rightarrow$  **Hom**( $U, F$ ) obtained by precomposition is an isomorphism.*

A  $j$ -Sheaf is up to isomorphism the result of sheafifying an object.

$j$ -Sheaves form a topos. The sheaves on a presheaf topos correspond to Beth semantics.

### 2.2 Sheaves in type theory

Consider a type theory with a notion of proof irrelevant propositions **Prop**, e.g. book-HoTT with mere propositions, or ROCQ with **SProp**.

In this case, a Lawvere-Tierney topology may be similarly defined, as a monad:

- $J : \mathbf{Prop} \rightarrow \mathbf{Prop}$
- $\eta : \Pi(P : \mathbf{Prop}). P \rightarrow J P$
- $\text{bind} : \forall(PQ : \mathbf{Prop}). J P \rightarrow (P \rightarrow J Q) \rightarrow J Q$

Then a sheaf is just a type  $T$  with

- A map  $\text{ask}_T : \Pi(P : \mathbf{Prop}). J P \rightarrow (P \rightarrow T) \rightarrow T$
- A coherence  $\varepsilon_T : \Pi(P : \mathbf{Prop}) (j : J P) (x : T). \text{ask}_T P j (\lambda p : P.x) = x$

Now, the sheafified of a type doesn't exist in general, if the theory admits quotient inductive types, it can then be defined as follow :

$$\begin{aligned} \text{Inductive } \mathcal{S}_J T : \text{Type} := \\ & | \text{ret} : T \rightarrow \mathcal{S}_J T \\ & | \text{ask} : \Pi(P : \text{Prop}), J P \rightarrow (P \rightarrow \mathcal{S}_J T) \rightarrow \mathcal{S}_J T \\ & | \varepsilon : \Pi(P : \text{Prop}) (j : J P) (x : \mathcal{S}_J T). \text{ask } P j (\lambda p : P.x) = x \end{aligned}$$

We note that by taking  $I := \Sigma(P : \text{Prop}). J P$  and  $O (P, j) : P$ , a sheaf is then a

- A type  $T$
- A map  $\text{ask}_T : \Pi(i : I), (O i \rightarrow T) \rightarrow T$
- A coherence map  $\varepsilon_T : \Pi(i : I) (x : T), \text{ask}_T i (\lambda o : O i.x) = x$

and similarly for sheafification. This is marginally simpler, and make sheaves appear as quotient dialogue trees, hence why we will henceforth consider  $(I, O)$ -sheaves instead of  $J$ -sheaves.

### 3 Models

A significant part of my internship was dedicated to constructing models of type theory in ROCQ.

1. A model of a variant of Baelofen TT using dialogue trees. Predicates must be linearized before eliminating an inductive into them.
2. An exceptional model, with a type of exceptions  $E$ . A special type of dialogue trees where  $I = E$  and  $Oi = \mathbf{0}$ , the resulting theory is inconsistent (when  $E$  is inhabited), as always when  $Oi \rightarrow \mathbf{0}$  for some  $i$ .
3. A model using sheaves, which requires univalence, and quotient inductive types to model positive types.

## 4 System T and MLTT

### 4.1 MLTT

We sought to extend the following variant of MLTT.

We consider a type of levels containing two elements  $\mathbf{s}$  and  $\mathbf{l}$  for small and large, with generic  $\ell$   
With terms :

$$M, N ::= x | \lambda x.M | MN | 0 | S | \mathbf{N}_{\text{rec}} | \perp_{\text{rec}} | \mathbf{N} | \perp | \Pi x : A. B | \square_{\mathbf{s}} | \square_{\mathbf{l}}$$

Contexts :

$$\Gamma ::= \Gamma, x : A | \cdot$$

And conversion rules :

$$\begin{aligned} & \text{WF-EMPTY} \frac{}{\cdot \vdash \text{well-formed}} \quad \text{WF-EXT} \frac{\Gamma \vdash A \equiv A \quad \Gamma \vdash \text{well-formed}}{\Gamma, x : A \vdash \text{well-formed}} \\ & \text{INT-TYP} \frac{\Gamma \vdash \text{well-formed}}{\Gamma \vdash \mathbf{N} \equiv \mathbf{N} : \square_{\ell}} \quad \text{EMP-TYP} \frac{\Gamma \vdash \text{well-formed}}{\Gamma \vdash \perp \equiv \perp : \square_{\ell}} \\ & \text{FUN-TYP} \frac{\Gamma \vdash A \equiv A' : \square_{\ell} \quad \Gamma, x : A \vdash B \equiv B' : \square_{\ell}}{\Gamma \vdash \Pi x : A, B \equiv \Pi x : A', B' : \square_{\ell}} \quad \text{TYP-TYP} \frac{\Gamma \vdash \text{well-formed}}{\Gamma \vdash \square_{\mathbf{s}} : \square_{\mathbf{l}}} \\ & \text{FUN-INTRO} \frac{\Gamma, x : A \vdash M \equiv M' : B \quad \Gamma \vdash A \equiv A : \square_{\ell}}{\Gamma \vdash \lambda x.M \equiv \lambda x.M' : \Pi x : A, B} \quad \text{FUN-ELIM} \frac{\Gamma \vdash M \equiv M' : \Pi x : A, B \quad \Gamma \vdash N \equiv N' : A}{\Gamma \vdash MN \equiv M'N' : B(N/x)} \\ & \text{AXIOM} \frac{\Gamma \vdash \text{well-formed} \quad x : A \in \Gamma}{\Gamma \vdash x \equiv x : A} \quad \text{BETA} \frac{\Gamma, x : A \vdash M \equiv M' : B \quad \Gamma \vdash N \equiv N' : A}{\Gamma \vdash (\lambda x.M)N \equiv M'(N'/x) : B(N/x)} \\ & \text{INT-ZERO} \frac{\Gamma \vdash \text{well-formed}}{\Gamma \vdash 0 \equiv 0 : \mathbf{N}} \quad \text{INT-SUCC} \frac{\Gamma \vdash \text{well-formed}}{\Gamma \vdash S \equiv S : \mathbf{N} \rightarrow \mathbf{N}} \\ & \text{INT-REC} \frac{\Gamma \vdash \text{well-formed}}{\Gamma \vdash \mathbf{N}_{\text{rec}} \equiv \mathbf{N}_{\text{rec}} : \Pi A : \mathbf{N} \rightarrow \square_{\mathbf{s}}, A0 \rightarrow (\Pi n : \mathbf{N}, An \rightarrow A(Sn)) \rightarrow \Pi n : \mathbf{N}, An} \\ & \text{INT-REC-ZERO} \frac{\Gamma \vdash A \equiv A : \mathbf{N} \rightarrow \text{square}_{\mathbf{s}} \quad \Gamma \vdash N_0 \equiv N'_0 \equiv A0 \quad \Gamma \vdash N_S \equiv N_S : \Pi n : \mathbf{N}, An \rightarrow A(Sn)}{\Gamma \vdash \mathbf{N}_{\text{rec}} A N_0 N_S 0 \equiv N'_0 : A0} \end{aligned}$$

$$\begin{array}{c}
\text{INT-REC-SUCC} \frac{\Gamma \vdash A \equiv A' : \mathbf{N} \rightarrow \text{square}_{\mathbf{s}} \quad \Gamma \vdash N_0 \equiv N'_0 \equiv A0 \quad \Gamma \vdash N_S \equiv N'_S : \Pi n : \mathbf{N}, An \rightarrow A(Sn) \quad \Gamma \vdash N \equiv N' : \mathbf{N}}{\Gamma \vdash \mathbf{N}_{\text{rec}} AN_0 N_S(SN) \equiv N'_S N'(\mathbf{N}_{\text{rec}} A' N'_0 N'_S N') : A(SN)} \\
\\
\text{EMP-REC} \frac{\Gamma \vdash \text{well-formed}}{\Gamma \vdash \perp_{\text{rec}} \equiv \perp_{\text{rec}} : \Pi A : \perp \rightarrow \Box_{\mathbf{s}}, \Pi e : \perp, Ae} \\
\\
\text{SYM} \frac{\Gamma \vdash M \equiv M' : A}{\Gamma \vdash M' \equiv M : A} \quad \text{TRANS} \frac{\Gamma \vdash M \equiv M' : A \quad \Gamma \vdash M' \equiv M'' : A}{\Gamma \vdash M \equiv M'' : A} \\
\\
\text{CONV} \frac{\Gamma \vdash M \equiv M' : A \quad \Gamma \vdash A \equiv A' : \Box_{\ell}}{\Gamma M \equiv M' : A'}
\end{array}$$

## 4.2 System T

To identify and solve problems in a simpler environment, we studied a modified System T before, based on the following variant.

With types :

$$A ::= A \rightarrow A | \mathbf{N} | \perp$$

Terms :

$$M, N ::= x | \lambda x. M | MN | 0 | S | \mathbf{N}_{\text{rec}} | \perp_{\text{rec}}$$

Contexts :

$$\Gamma ::= \Gamma, x : A | \cdot$$

And conversion rules :

$$\begin{array}{c}
\text{FUN-INTRO} \frac{\Gamma, x : A \vdash M \equiv M' : B}{\Gamma \vdash \lambda x. M \equiv \lambda x. M' : A \rightarrow B} \quad \text{FUN-ELIM} \frac{\Gamma \vdash M \equiv M' : A \rightarrow B \quad \Gamma \vdash N \equiv N' : A}{\Gamma \vdash MN \equiv M'N' : B} \\
\\
\text{AXIOM} \frac{x : A \in \Gamma}{\Gamma \vdash x \equiv x : A} \quad \text{BETA} \frac{\Gamma, x : A \vdash M \equiv M' : B \quad \Gamma \vdash N \equiv N' : A}{\Gamma \vdash (\lambda x. M)N \equiv M' : B} \\
\\
\text{INT-ZERO} \frac{\Gamma \vdash \text{well-formed}}{\Gamma \vdash 0 \equiv 0 : \mathbf{N}} \quad \text{INT-SUCC} \frac{\Gamma \vdash \text{well-formed}}{\Gamma \vdash S \equiv S : \mathbf{N} \rightarrow \mathbf{N}} \\
\\
\text{INT-REC} \frac{\Gamma \vdash \text{well-formed}}{\Gamma \vdash \mathbf{N}_{\text{rec}} \equiv \mathbf{N}_{\text{rec}} : A \rightarrow (\mathbf{N} \rightarrow A \rightarrow A) \rightarrow \mathbf{N} \rightarrow A} \\
\\
\text{INT-REC-ZERO} \frac{\Gamma \vdash N_0 \equiv N'_0 \equiv A \quad \Gamma \vdash N_S \equiv N'_S : \mathbf{N} \rightarrow A \rightarrow A}{\Gamma \vdash \mathbf{N}_{\text{rec}} N_0 N_S 0 \equiv N'_0 : A} \\
\\
\text{INT-REC-SUCC} \frac{\Gamma \vdash N_0 \equiv N'_0 \equiv A \quad \Gamma \vdash N_S \equiv N'_S : \mathbf{N} \rightarrow A \rightarrow A \quad \Gamma \vdash N \equiv N' : \mathbf{N}}{\Gamma \vdash \mathbf{N}_{\text{rec}} N_0 N_S(SN) \equiv N'_S N'(\mathbf{N}_{\text{rec}} N'_0 N'_S N') : A} \\
\\
\text{EMP-REC} \frac{}{\Gamma \vdash \perp_{\text{rec}} \equiv \perp_{\text{rec}} : \perp \rightarrow A} \\
\\
\text{SYM} \frac{\Gamma \vdash M \equiv M' : A}{\Gamma \vdash M' \equiv M : A} \quad \text{TRANS} \frac{\Gamma \vdash M \equiv M' : A \quad \Gamma \vdash M' \equiv M'' : A}{\Gamma \vdash M \equiv M'' : A}
\end{array}$$

## 5 Meta-informations

### 5.1 Time expenditure

### 5.2 Difficulties

The subject is rather vast and a bit unclear.

Doing logical relations for MLTT is a large task, even before adding sheaves. My next attempt will probably start from logrel rocq insted of nothing

## 6 Conclusion

## A Dummy

### References

- [1] Saunders MacLane and Ieke Moerdijk. *Sheaves in geometry and logic: A first introduction to topos theory*. Springer Science & Business Media, 2012.