

Notes on Dependent Type Theory

Alyson Mei

January 18, 2026

These notes are based on [video recordings](#) of R. Harper's lectures.

Conventions.

- Bullet lists are used for strongly logically connected items (primarily in definitions), while dashed lists are used for (somewhat) distinct items;
- Remarks marked with * are additions by the present author.

1 Introduction

Boolean algebra is associated with classical logic, Heyting algebra – with intuitionistic logic.

Definition 1.1 (Boolean algebra). A *Boolean algebra* can be defined as a complemented distributive lattice:

- Pre-order:

$$x \leq x, \quad x \leq y \& y \leq z \rightarrow x \leq z;$$

- Has finite meets and joins:

$$x \leq 1, \quad z \leq x \& z \leq y \rightarrow z \leq (x \wedge y), \quad x \wedge y \leq x, \quad x \wedge y \leq y;$$

$$0 \leq x, \quad x \leq z \& y \leq z \rightarrow x \vee y \leq z, \quad x \leq x \vee y, \quad y \leq x \vee y;$$

- Has complements:

$$1 \leq \bar{x} \vee x, \quad \bar{x} \wedge x \leq 0;$$

- Distributive:

$$x \wedge (y \vee z) \equiv (x \wedge y) \vee (x \wedge z), \quad x \vee (y \wedge z) \equiv (x \vee y) \wedge (x \vee z).$$

Additionally, the exponential is defined as

$$y^x := \bar{x} \vee y.$$

Definition 1.2 (Heyting algebra). A *Heyting algebra* is defined as a lattice with exponentials:

- Pre-order;
- Has finite meets and joins;

- Has exponentials:

$$y^x \wedge x \leq y, \quad z \wedge x \leq y \rightarrow z \leq y^x.$$

Exercise 1.3.

1. “Yoneda lemma”: $x \leq y$ iff $\forall z (z \leq x \rightarrow z \leq y)$;
2. Every Heyting algebra is distributive.

Quotes:

- “Boolean algebra (closed world) = Heyting algebra (open world) with complements”;
- “Classical logic is a logic with complete information”.

The definitions provide us with standard rules:

- Weakening:

$$x \leq x \vee y \quad (x \wedge y \leq x);$$

- Contraction:

$$x \leq x \wedge x;$$

- Exchange:

$$x \wedge y \equiv y \wedge x.$$

Relation to classical logic:

- Sequent $\Gamma \vdash A$, where $\Gamma = A_1, \dots, A_n$, corresponds to:

$$A_1 \wedge \dots \wedge A_n \leq A;$$

- Rules for \wedge :

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \quad \frac{}{\Gamma, A \wedge B \vdash A} \quad \frac{}{\Gamma, A \wedge B \vdash B};$$

- ...and so on.

Definition 1.4 (Lindenbaum algebra). A *Lindenbaum algebra* is defined as the algebra of equivalence classes of a given theory:

- $[A] = \{B \mid B \equiv A\}$;
- $[A] \wedge [B] := [A \wedge B]$;
- ... and so on.

Theorem 1.5 (Soundness and Completeness for Intuitionistic Propositional Logic). Let Γ be a context and A a formula. Then

$$\Gamma \vdash A \quad \text{iff} \quad \forall H ([\Gamma]_H \leq [A]_H),$$

where H ranges over all Heyting algebras, and $[\neg]_H$ denotes the interpretation of formulas as elements of H under a valuation of propositional variables.

2 Simple Type Theory

Definition 2.1 (Simple type theory). We define the *simple type theory* as follows:

- Unit 1:

$$\frac{}{\Gamma \vdash 1 \text{ type}} \text{ 1-}F \quad \frac{}{\Gamma \vdash \langle \rangle : 1} \text{ 1-}I \quad (\text{no 1-E}) ;$$

- Product \times :

$$\frac{\Gamma \vdash A \text{ type} \quad \Gamma \vdash B \text{ type}}{\Gamma \vdash A \times B \text{ type}} \text{ } \times\text{-}F \quad \frac{\Gamma \vdash M : A \quad \Gamma \vdash N : B}{\Gamma \vdash \langle M, N \rangle : A \times B} \text{ } \times\text{-}I \quad \frac{\Gamma \vdash M : A \times B}{\begin{array}{l} \Gamma \vdash \text{fst}(M) : A \\ \Gamma \vdash \text{snd}(M) : B \end{array}} \text{ } \times\text{-}E ;$$

- Exponential \rightarrow :

$$\frac{\Gamma \vdash A \text{ type} \quad \Gamma \vdash B \text{ type}}{\Gamma \vdash A \rightarrow B \text{ type}} \text{ } \rightarrow\text{-}F \quad \frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x. M : A \rightarrow B} \text{ } \rightarrow\text{-}I$$

$$\frac{\Gamma \vdash M : A \rightarrow B \quad \Gamma \vdash N : A}{\Gamma \vdash M(N) : B} \text{ } \rightarrow\text{-}E ;$$

- Void 0:

$$\frac{}{\Gamma \vdash 0 \text{ type}} \text{ 0-}F \quad (\text{no 0-I}) \quad \frac{\Gamma \vdash M : 0}{\Gamma \vdash \text{abort}(M) : A} \text{ 0-}E ;$$

- Coproduct $+$:

$$\frac{\Gamma \vdash A \text{ type} \quad \Gamma \vdash B \text{ type}}{\Gamma \vdash A + B \text{ type}} \text{ } +\text{-}F \quad \frac{\Gamma \vdash M : A}{\Gamma \vdash \text{inl}(M) : A + B} \text{ } +\text{-}I_1 \quad \frac{\Gamma \vdash N : B}{\Gamma \vdash \text{inr}(N) : A + B} \text{ } +\text{-}I_2 ,$$

$$\frac{\Gamma, x : A \vdash N : C \quad \Gamma, y : B \vdash P : C}{\Gamma, z : A + B \vdash \text{case}(x.N, y.P)(z) : C} \text{ } +\text{-}E .$$

Remark 2.2 (Categorical interpretation*). The type theory given above corresponds to a category \mathcal{C} that is both cartesian closed and cocartesian. We also assume that \mathcal{C} has all morphisms from the terminal object 1 to the objects corresponding to the types of context variables (this is somewhat experimental).

- Types are the objects of \mathcal{C} ;
- A context $\Gamma = x_1 : A_1, \dots, x_n : A_n$ corresponds to the product of its types, together with a morphism from the terminal object naming the variables:

$$\Gamma := A_1 \times \cdots \times A_n, \quad \langle x_1, \dots, x_n \rangle : 1 \rightarrow \Gamma;$$

- Terms are morphisms in \mathcal{C} :

$$\Gamma \vdash M : A \quad \mapsto \quad M : \Gamma \rightarrow A;$$

- Unit 1 is a terminal object; the 1-I rule corresponds to arrows *to* 1. Void 0 is an initial object; the 0-E rule corresponds to arrows *from* 0:

$$\begin{array}{c} \Gamma \\ \downarrow \langle \rangle \\ 1 \end{array} \quad \begin{array}{ccc} \Gamma & \xrightarrow{M} & 0 \\ \searrow \text{abort}(M) & & \downarrow \text{abort} ; \\ & A & \end{array}$$

- Product \times and coproduct $+$ are the usual categorical limits and colimits (finite in our case):

$$\begin{array}{ccc} \begin{array}{c} \Gamma \\ \downarrow (M,N) \\ A \times B \end{array} & \begin{array}{ccc} \Gamma & & \\ \downarrow & & \\ A & \xleftarrow{\text{fst}} & A \times B \xrightarrow{\text{snd}} B \end{array} & \begin{array}{ccc} \Gamma & & \\ \downarrow M & & \searrow \text{snd}(M) \\ A \times B & \xleftarrow{\text{fst}} & B \end{array} , \\ \begin{array}{ccc} A & \xrightarrow{\text{inl}} & A + B \\ \uparrow \text{inl}(M) & & \uparrow \text{inr} \\ \Gamma & & \end{array} & & \begin{array}{ccc} A + B & & \\ \uparrow \text{inr}(N) & & \uparrow \text{inl} \\ B & \xleftarrow{N} & \Gamma \end{array} , \\ \begin{array}{ccccc} \Gamma \times 1 & & & & \Gamma \times B \\ \downarrow \Gamma \times z & & & & \searrow \Gamma \times y \\ \Gamma \times (A + B) & & & & \Gamma \times B ; \\ \downarrow \text{case}(x.N,y.P) & & & & \downarrow P \\ N & \xrightarrow{\Gamma \times x} & C & \xleftarrow{P} & \end{array} & & \end{array}$$

- Exponential \rightarrow is the right adjoint to product \times :

$$(A \times -) \dashv (A \rightarrow -)$$

$$\begin{array}{ccc} \begin{array}{c} \Gamma \\ \downarrow \lambda x.M \\ A \rightarrow B \end{array} & \begin{array}{ccc} \Gamma \times A & \xleftarrow{\text{id}_{\Gamma \times x}} & \Gamma \times 1 \\ \downarrow M & & \downarrow \text{Moid}_{\Gamma \times x} \\ B & \xleftarrow{\text{Moid}_{\Gamma \times x}} & \end{array} ; & \begin{array}{ccc} \Gamma & & \\ \downarrow \langle M,N \rangle & & \searrow \varepsilon_B \circ \langle M,N \rangle \\ A \times (A \rightarrow B) & \xrightarrow[\varepsilon_B]{} & B \end{array} . \end{array}$$

Remark 2.3 (Categorical notation*). Formally, we should've been writing $1 \rightarrow \Gamma$ instead of just Γ for each diagram. But for brevity, we explicitly defined only those variables that are mentioned in the type definitions. Maybe I will refine this later.

Definition 2.4 (β and η equivalences).

– Unit 1:

$$(\beta) \text{ none} \quad (\eta) \Gamma \vdash \langle \rangle \equiv M : 1;$$

– Product \times :

$$(\beta) \begin{array}{l} \Gamma \vdash \text{fst}(\langle M, N \rangle) \equiv M : A \\ \Gamma \vdash \text{snd}(\langle M, N \rangle) \equiv N : B \end{array} \quad (\eta) \Gamma \vdash \langle \text{fst } M, \text{snd } M \rangle \equiv M : A \times B;$$

– Exponential \rightarrow :

$$(\beta) \Gamma \vdash (\lambda x.M)(N) \equiv [N/x]M : B \quad (\eta) \Gamma \vdash (\lambda x.M(x)) \equiv M : A \rightarrow B;$$

– Zero 0:

$$(\beta) \text{ none} \quad (\eta) \Gamma, z : 0 \vdash R \equiv \text{abort}(M) : C;$$

– Coproduct $+$:

$$\frac{\begin{array}{c} (\beta) \begin{array}{l} \text{case}(x.M; y.N)(\text{inl}(P)) \equiv [P/x]M : C \\ \text{case}(x.M; y.N)(\text{inr}(Q)) \equiv [Q/y]N : C \end{array} \\ \Gamma \vdash [\text{inl}(P)/z]R \equiv [P/x]M \quad \Gamma \vdash [\text{inr}(Q)/z]R \equiv [Q/y]N \end{array}}{\Gamma, z : A + B \vdash R \equiv \text{case}(x.M; y.N)(z)} \quad (\eta).$$

Next, we're going to augment this STT with "data" types.

Definition 2.5 (Natural numbers type). The *natural numbers type* is defined as follows:

- Introduction: zero is a Nat, succ($x : \text{Nat}$) is a Nat:

$$\frac{}{\Gamma \vdash \text{zero} : \text{Nat}} \quad \frac{}{\Gamma, x : \text{Nat} \vdash \text{succ}(x) : \text{Nat}} ;$$

- Elimination:

$$\frac{\Gamma \vdash M : C \quad \Gamma, x : C \vdash N : C}{\Gamma, z : \text{Nat} \vdash \text{iter}(M, x.N)(z) : C},$$

given by the recursion:

$$\text{iter}(M, x.N)(\text{zero}) = M, \quad \text{iter}(M, x.N)(\text{succ}(n)) = [\text{iter}(M, x.N)(n)/x]N.$$

Note: (η) here is not easy to formulate, because it would require ω -rule.

(next there were brief comments by R.Harper about inductive types and categorical notation)

Remark 2.6 (Explaining the iter*). Given $M \in C$, $f \in \text{Hom}(C, C)$ and $z \in \mathbb{N}$, iter essentially applies function f to M z times:

$$\text{iter} : C \times \text{Hom}(C, C) \times \mathbb{N} \rightarrow C, \quad \text{iter} : (M, f, z) \mapsto f_{(z)}(\dots(f_{(1)}(M))).$$

The recursion takes the following form:

$$\text{iter}(M, f, 0) = M, \quad \text{iter}(M, f, \text{succ}(n)) = f(\text{iter}(M, f, n)).$$

Remark 2.7 (Natural numbers object*). In a category \mathcal{C} with a terminal object 1 , a *natural number object (NNO)*

$$(N, z : 1 \rightarrow N, s : N \rightarrow N)$$

is an initial object in the category induced by morphisms $a : (x, g) \rightarrow (y, h) :$

$$\begin{array}{ccc} 1 & \xrightarrow{x} & X & \xrightarrow{g} & X \\ & \searrow y & \downarrow a & & \downarrow a \\ & & Y & \xrightarrow{h} & Y \end{array} \quad \begin{array}{ccc} 1 & \xrightarrow{z} & N & \xrightarrow{s} & N \\ & \searrow q & \downarrow u & & \downarrow u \\ & & A & \xrightarrow{f} & A \end{array} .$$

3 Families of types