Contents

1. Non-classical logic

1.1. Intuitionistic logic

Idea: a statement is true if there is a proof of it. A proof of $\varphi \Rightarrow \psi$ is a "procedure" that can convert a proof of φ to a proof of ψ. A proof of $\neg\varphi$ is a proof that there is no proof of φ.

In particular, $\neg\neg\varphi$ is not always the same as φ.

Fact 1.1 The Law of Excluded Middle (LEM) ($\varphi \vee \neg\varphi$) is not (generally) intuitionistically valid.

Moreover, the Axiom of Choice is incompatible with intuitionistic set theory.

In intuitionistic logic, $\exists$ means an explicit element can be found.

Why bother with intuitionistic logic?
- Intuitionistic mathematics is more general, as we assume less (no LEM or AC).
- Several notions that are conflated in classical mathematics are genuinely different constructively.
- Intuitionistic proofs have a computable content that may be absent in classical proofs.
- Intuitionistic logic is the internal logic of an arbitrary topos.

We will inductively define a provability relation by enforcing rules that implement the BHK-interpretation.

Definition 1.2 A set is **inhabited** if there is a proof that it is non-empty.

Axiom 1.3 (Choice - Intutionistic Version) Any family of inhabited sets admits a choice function.

Theorem 1.4 (Diaconescu) The Law of Excluded Middle can be intutionistically deduced from the Axiom of Choice.

Proof (Hints).
- Proof should use Axioms of Separation, Extensionality and Choice.
- For proposition φ, consider $A = \{x \in \{0,1\} : \varphi \vee (x = 0)\}$ and $B = \{x \in \{0,1\} : \varphi \vee (x = 1)\}$.
- Show that we have a proof of $f(A) = 0 \vee f(A) = 1$, similarly for $f(B)$.
- Consider the possibilities that arise from above, show that they lead to either a proof of φ or a proof of $\neg\varphi$.

□

Proof.
- Let φ be a proposition. By the Axiom of Separation, the following are sets:

$$A = \{x \in \{0,1\} : \varphi \vee (x = 0)\},$$
$$B = \{x \in \{0,1\} : \varphi \vee (x = 1)\}.$$

- Since $0 \in A$ and $1 \in B$, we have a proof that $\{A, B\}$ is a family of inhabited sets, thus admits a choice function $f : \{A, B\} \to A \cup B$ by the Axiom of Choice.
- f satisfies $f(A) \in A$ and $f(B) \in B$ by definition.
- So we have $f(A) = 0$ or φ is true, and $f(B) = 1$ or φ is true. Also, $f(A), f(B) \in \{0, 1\}$.
- Now $f(A) \in \{0, 1\}$ means we have a proof of $f(A) = 0 \vee f(A) = 1$ and similarly for $f(B)$.
- There are the following possibilities:
  1. We have a proof that $f(A) = 1$, so $\varphi \vee (1 = 0)$ has a proof, so we must have a proof of φ.
  2. We have a proof that $f(B) = 0$, so $\varphi \vee (0 = 1)$ has a proof, so we must have a proof of φ.
  3. We have a proof that $f(A) = 0 \wedge f(B) = 1$, in which case we can prove $\neg\varphi$: assume there is a proof of φ, we can prove that $A = B$ (by the Axiom of Extensionality), in which case $0 = f(A) = f(B) = 1$: contradiction.
- So we can always specify a proof of φ or a proof of $\neg\varphi$.

□

Notation 1.5 We write $\Gamma \vdash \varphi$ to mean that φ is a consequence of the formulae in the set Γ. Γ is called the **set of hypotheses or open assumptions**.

Notation 1.6 Notation for assumptions and deduction.

Definition 1.7 The rules of the **intuitionistic propositional calculus (IPC)** are:

- conjunction introduction,
- conjunction elimination,
- disjunction introduction,
- disjunction elimination,
- implication introduction,
- implication elimination,
- assumption,
- weakening,
- construction,
- and for any A,

$$\frac{\Gamma \vdash \perp}{\Gamma \vdash A}.$$

as defined below.

Definition 1.8 The **conjunction introduction (∧-I)** rule:

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}.$$

Definition 1.9 The **conjunction elimination (∧-E)** rule:

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}, \quad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}.$$

Definition 1.10 The **disjunction introduction ($\vee$-I)** rule:

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}, \quad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}.$$

Definition 1.11 The **disjunction elimination (proof by cases) ($\vee$-E)** rule:

$$\frac{\Gamma, A \vdash C \quad \Gamma, B \vdash C \quad \Gamma \vdash A \vee B}{\Gamma \vdash C}.$$

Definition 1.12 The **implication/arrow introduction ($\to$-I)** rule (note the similarity to the deduction theorem):

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}.$$

Definition 1.13 The **implication/arrow elimination ($\to$-E)** rule (note the similarity to modus ponens):

$$\frac{\Gamma \vdash A \to B \quad \Gamma \vdash A}{\Gamma \vdash B}.$$

Definition 1.14 The **assumption ($\boldsymbol{Ax}$)** rule: for any A,

$$\frac{}{\Gamma, A \vdash A}$$

Definition 1.15 The **weakening** rule:

$$\frac{\Gamma \vdash B}{\Gamma, A \vdash B}.$$

Definition 1.16 The **construction** rule:

$$\frac{\Gamma, A, A \vdash B}{\Gamma, A \vdash B}.$$

Remark 1.17 We obtain classical propositional logic (CPC) from IPC by adding either:

- $\Gamma \vdash A \vee \neg A$:

$$\frac{}{\Gamma \vdash A \vee \neg A},$$

  or
- If $\Gamma, \neg A \vdash \perp$, then $\Gamma \vdash A$:

$$\frac{\Gamma, \neg A \vdash \perp}{\Gamma \vdash A}.$$

Notation 1.18 see scan

Definition 1.19 We obtain **intuitionistic first-order logic (IQC)** by adding the following rules to IPC for quantification:

- existental inclusion,
- existential elimination,
- universal inclusion,
- universal elimination

as defined below.

Definition 1.20 The **existential inclusion (∃-I)** rule: for any term t,

$$\frac{\Gamma \vdash \varphi[t/x]}{\Gamma \vdash \exists x.\varphi(x)}.$$

Definition 1.21 The **existential elimination (∃-I)** rule:

$$\frac{\Gamma \vdash \exists x.\varphi \quad \Gamma, \varphi \vdash \psi}{\Gamma \vdash \psi},$$

where x is not free in Γ or ψ.

Definition 1.22 The **universal inclusion (∀-I)** rule:

$$\frac{\Gamma \vdash \varphi}{\Gamma \vdash \forall x.\varphi},$$

where x is not free in Γ.

Definition 1.23 The **universal exclusion (∀-E)** rule:

$$\frac{\Gamma \vdash \forall x.\varphi(x)}{\Gamma \vdash \varphi[t/x]},$$

where t is a term.

Definition 1.24 We define the notion of **discharging/closing** open assumptions, which informally means that we remove them as open assumptions, and append them to the consequence by adding implications. We enclose discharged assumptions in square brackets [] to indicate this, and add discharged assumptions in parentheses to the right of the proof. For example, →-I is written as

$$\begin{array}{c} \Gamma, [A] \\ \vdots \\ \dfrac{B}{\Gamma \vdash A \to B}(A) \end{array}$$

Example 1.25 A natural deduction proof that $A \wedge B \to B \wedge A$ is given below:

$$\dfrac{\dfrac{\dfrac{[A \wedge B]}{A} \quad \dfrac{[A \wedge B]}{B}}{B \wedge A}}{A \wedge B \to B \wedge A}(A \wedge B)$$

Example 1.26 A natural deduction proof of $\varphi \to (\psi \to \varphi)$ is given below (note clearly we must use $\to$-I):

$$\dfrac{\dfrac{[\varphi] \quad [\psi]}{\psi \to \varphi}}{\varphi \to (\psi \to \varphi)}$$

Example 1.27 A natural deduction proof of $(\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))$ (note clearly we must use $\to$-I):

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\varphi \to (\psi \to \chi)] \quad [\varphi \to \psi] \quad [\varphi]}{\psi \to \chi \qquad \psi}}{\chi}}{\varphi \to \chi}}{(\varphi \to \psi) \to (\varphi \to \chi)}}{(\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))}$$

Notation 1.28 If Γ is a set of propositions, φ is a proposition and $L \in \{\text{IPC}, \text{IQC}, \text{CPC}, \text{CQC}\}$, write $\Gamma \underset{L}{\vdash} \varphi$ if there is a proof of φ from Γ in the logic L.

Lemma 1.29 If $\Gamma \underset{\text{IPC}}{\vdash} \varphi$, then $\Gamma, \psi \underset{\text{IPC}}{\vdash} \varphi$ for any proposition ψ. If p is a primitive proposition (doesn't contain any logical connectives or quantifiers) and ψ is any proposition, then $\Gamma[\psi/p] \underset{\text{IPC}}{\vdash} \varphi[\psi/p]$.

Proof. Induction on number of lines of proof (exercise). □

1.2. The simply typed λ-calculus

Definition 1.30 The set Π of **simple types** is generated by the grammar

$$\Pi \coloneqq U \mid \Pi \to \Pi$$

where U is a countable set of **type variables (primitive types)** together with an infinite set of V of **variables**. So Π consists of U and is closed under $\to$: for any $a, b \in \Pi$, $a \to b \in \Pi$.

Definition 1.31 The set Λ_Π of **simply typed $\boldsymbol{\lambda}$-terms** is defined by the grammar

$$\Lambda_\Pi \coloneqq V \mid \lambda V : \Pi \,.\, \Lambda_\Pi \mid \Lambda_\Pi \; \Lambda_\Pi$$

In the term $\lambda x : \tau.M$, x is a variable, τ is type and M is a λ-term. Forming terms of this form is called **$\boldsymbol{\lambda}$-abstraction**. Forming terms of the form $\Lambda_\Pi \Lambda_\Pi$ is called **$\boldsymbol{\lambda}$-application**.

Example 1.32 The λ-term $\lambda x : \mathbb{Z}.x^2$ should represent the function $x \mapsto x^2$ on $\mathbb{Z}$.

Definition 1.33 A **context** is a set of pairs $\Gamma = \{x_1 : \tau_1, ..., x_n : \tau_n\}$ where the x_i are distinct variables and each τ_i is a type. So a context is an assignment of a type to each variable in a given set. Write C for the set of all possible contexts. Given a context $\Gamma \in C$, write $\Gamma, x : \tau$ for the context $\Gamma \cup \{x : \tau\}$ (if x does not appear in Γ).

The **domain** of Γ is the set of variables $\{x_1, ..., x_n\}$ that occur in it, and its **range**, $|\Gamma|$, is the set of types $\{\tau_1, ..., \tau_n\}$ that it manifests.

Definition 1.34 Recursively define the **typability relation** $\Vdash \subseteq C \times \Lambda_\Pi \times \Pi$ via:
1. For every context Γ, variable x not occurring in Γ and type τ, we have $\Gamma, x : \tau \Vdash x : \tau$.
2. For every context Γ, variable x not occurring in Γ, types $\sigma, \tau \in \Pi$, and λ-term M, if $\Gamma, x : \sigma \Vdash M : \tau$, then $\Gamma \Vdash (\lambda x : \sigma.M) : (\sigma \to t)$.
3. For all contexts Γ, types $\sigma, \tau \in \Pi$, and terms $M, N \in \Lambda_\Pi$, if $\Gamma \Vdash M : (\sigma \to t)$ and $\Gamma \Vdash N : \sigma$, then $\Gamma \Vdash (MN) : \tau$.

Definition 1.35 For $\Gamma \in C$, we say a λ-term $M \in \Lambda_\Pi$ is **typable** if for some type $\tau \in \Pi$, $\Gamma \Vdash M : \tau$.

Notation 1.36 We will refer to the λ-calculus of Λ_Π with this typability relation as $\lambda(\to)$.

Definition 1.37 A variable x occurring in a λ-abstraction $\lambda x : \sigma.M$ is **bound** and is **free** otherwise. A term with no free variables is called **closed**.

Definition 1.38 Terms M and N are **α-equivalent** if they differ only in the names of their bound variables.

Definition 1.39 If M and N are λ-terms and x is a variable, then we define the **substitution of N for x in M** by the following rules:
- $x[x := N] = N$.
- $y[x := N] = y$ for $y \neq x$.
- $(PQ)[x := N] = P[x := N]Q[x := N]$ for λ-terms P, Q.
- $(\lambda y : \sigma.P)[x := N] = \lambda y : \sigma.(P[x := N])$ for $x \neq y$ and y not free in N.

Definition 1.40 The **β-reduction** relation is the smallest relation $\xrightarrow[\beta]{}$ on Λ_Π closed under the following rules:
- $(\lambda x : \sigma.P)Q \xrightarrow[\beta]{} P[x := Q]$. The term being reduced is called a **β-redex**, and the result is called its **β-contraction**.
- If $P \xrightarrow[\beta]{} P'$, then for all variables x and types $\sigma \in \Pi$, we have $\lambda x : \sigma.P \xrightarrow[\beta]{} \lambda x : \sigma.P'$.
- If $P \xrightarrow[\beta]{} P'$ and Z is a λ-term, then $PZ \xrightarrow[\beta]{} P'Z$ and $ZP \xrightarrow[\beta]{} ZP'$.

Definition 1.41 We define **β-equivalence**, $\underset{\beta}{\equiv}$, as the smallest equivalence relation containing $\xrightarrow[\beta]{}$.

Example 1.42 We have $(\lambda x : \mathbb{Z}.(\lambda y : \tau.x))2 \xrightarrow[\beta]{} (\lambda y : \tau.2)$.

Lemma 1.43 (Free Variables Lemma) Let $\Gamma \Vdash M : \sigma$. Then

- If $\Gamma \subseteq \Gamma'$, then $\Gamma' \Vdash M : \sigma$.
- The free variables of M occur in Γ.
- There is a context $\Gamma^* \subseteq \Gamma$ whose variables are exactly the free variables in M, with $\Gamma^* \Vdash M : \sigma$.

Proof. By induction on the grammar (exercise). □

Lemma 1.44 (Generation Lemma)

1. For every variable $x \in V$, context Γ and type $\sigma \in \Pi$: if $\Gamma \Vdash x : \sigma$, then $x : \sigma \in \Gamma$.
2. If $\Gamma \Vdash (MN) : \sigma$, then there is a type $\tau \in \Pi$ such that $\Gamma \Vdash M : \tau \to \sigma$ and $\Gamma \Vdash N : \tau$.
3. If $\Gamma \Vdash (\lambda x.M) : \sigma$, then there are types $\tau, \rho \in \Pi$ such that $\Gamma, x : \tau \Vdash M : \rho$ and $\sigma = (\tau \to \rho)$.

Proof. By induction on the grammar (exercise). □

Lemma 1.45 (Substitution Lemma)

1. If $\Gamma \Vdash M : \sigma$ and $\alpha \in U$ is a type variable, then $\Gamma[\alpha := \tau] \Vdash M : \sigma[\alpha := \tau]$.
2. If $\Gamma, x : \tau \Vdash M : \sigma$ and $\Gamma \Vdash N : \tau$, then $\Gamma \Vdash M[x := N] : \sigma$.

Proof. By induction on the grammar (exercise). □

Proposition 1.46 (Subject Reduction) If $\Gamma \Vdash M : \sigma$ and $M \underset{\beta}{\longrightarrow} N$, then $\Gamma \Vdash N : \sigma$.

Proof.

- By induction on the derivation of $M \underset{\beta}{\longrightarrow} N$, using Generation and Substitution Lemmas (exercise).

□

Definition 1.47 A λ-term $M \in \Lambda_\Pi$ is an **β-normal form (β-NF)** if there is no term $N \neq M$ such that $M \underset{\beta}{\longrightarrow} N$.

Notation 1.48 Write $M \underset{\beta}{\twoheadrightarrow} N$ if M reduces to N after (potentially) multiple β-reductions.

Theorem 1.49 (Church-Rosser for $\lambda(\to)$) Suppose that $\Gamma \Vdash M : \sigma$. If $M \underset{\beta}{\twoheadrightarrow} N_1$ and $M \underset{\beta}{\twoheadrightarrow} N_2$, then there is a λ-term L such that $N_1 \underset{\beta}{\twoheadrightarrow} L$ and $N_2 \underset{\beta}{\twoheadrightarrow} L$, and $\Gamma \Vdash L : \sigma$.

Remark 1.50 In Church-Rosser, the fact that $M \underset{\beta}{\twoheadrightarrow} N_1$ and $M \underset{\beta}{\twoheadrightarrow} N_2$ implies that $N_1, N_2 \underset{\beta}{\twoheadrightarrow} L$ is called **confluence**, and can be represented diagramatically as

$$\begin{tikzcd}
 & M \arrow[dl, two heads] \arrow[dr, two heads] & \\
N_1 \arrow[dr, two heads] & & N_2 \arrow[dl, two heads] \\
 & L &
\end{tikzcd}$$

Corollary 1.51 (Uniqueness of normal form) If a simply-typed λ-term admits a β-NF, then this form is unique.

Proposition 1.52 (Uniqueness of types)
1. If $\Gamma \Vdash M : \sigma$ and $\Gamma \Vdash M : \tau$, then $\sigma = \tau$.
2. If $\Gamma \Vdash M : \sigma$, $\Gamma \Vdash N : \tau$, and $M \underset{\beta}{\equiv} N$, then $\sigma = \tau$.

Proof.
1. Induction (exercise).
2. By Church-Rosser, there is a λ-term L which both M and N reduce to (since β-equivalence means there is a finite sequence of alternating up and down $\underset{\beta}{\twoheadrightarrow}$ relations). By Subject Reduction, we have $\Gamma \Vdash L : \sigma$ and $\Gamma \Vdash L : \tau$, so $\sigma = \tau$ by 1.

$\square$

Example 1.53 There is no way to assign a type to $\lambda x.xx$: let x be of type τ, then by the Generation Lemma, in order to apply x to x, x must be of type $\tau \to \sigma$ for some type σ. But $\tau \neq \tau \to \sigma$, which contradicts Uniqueness of Types.

Definition 1.54 The **height function** is the recursively defined map $h : \Pi \to \mathbb{N}$ that maps all type variables $u \in U$ to 0, and a function type $\sigma \to \tau$ to $1 + \max\{h(\sigma), h(\tau)\}$:

$$\begin{aligned} h &: \Pi \to \mathbb{N}, \\ h(\alpha) &= 0 \quad \forall \alpha \in U, \\ h(\sigma \to \tau) &= 1 + \max\{h(\sigma), h(\tau)\} \quad \forall \sigma, \tau \in \Pi. \end{aligned}$$

The **height** of a redex is defined as the height of the type of its λ-abstraction:

$$h\big((\lambda x : \sigma.P^{\tau})^{\sigma \to \tau} Q\big) = h(\sigma \to \tau).$$

Notation 1.55 $(\lambda x : \sigma.P^{\tau})^{\sigma \to \tau}$ denotes that P has type τ and the λ-abstraction has type $\sigma \to \tau$.

Theorem 1.56 (Weak normalisation for $\lambda(\to)$) Let $\Gamma \Vdash M : \sigma$. Then there is a finite reduction path $M \coloneqq M_0 \underset{\beta}{\longrightarrow} M_1 \underset{\beta}{\longrightarrow} \dots \underset{\beta}{\longrightarrow} M_n$, where M_n is in β-normal form.

Proof "Taming the Hydra".
- Idea is to apply induction on the complexity of M.
- Define a function $m : \Lambda_{\Pi} \to \mathbb{N} \times \mathbb{N}$ by

$$m(M) := \begin{cases} (0,0) & \text{if } M \text{ is in } \beta\text{-NF} \\ (h(M), \mathrm{redex}(M)) & \text{otherwise} \end{cases}$$

where $h(M)$ is the maximal height of a redex in M, and $\mathrm{redex}(M)$ is the number of redexes in M of that height.

- We use induction over $\omega \times \omega$ to show that if M is typable, then it admits a reduction to β-NF.
- The problem is that reductions can copy redexes or create new ones, so our strategy is to always reduce the right-most redex of maximal height.
- We will argue that, by following this strategy, any new redexes that we generate have a strictly lower height than the height of the redex we chose to reduce.
- If $\Gamma \Vdash M : \sigma$ and M is already in β-NF, then we are done.
- So assume M is not in β-NF. Let Δ be the rightmost redex of maximal height h.
- By reducing Δ, we may introduce copies of existing redexes or create new ones.
- Creation of new redexes by β-reduction of Δ in one of the following ways:
  1. If Δ is of the form $(\lambda x : (\rho \to \mu)...xP^{\rho}...)(\lambda y : \rho.Q^{\mu})^{\rho\to\mu}$, then it reduces to $...(\lambda y : \rho.Q^{\mu})^{\rho\to\mu}P^{\rho}...$, in which case there is a new redex of height $h(\rho \to \mu) < h$.
  2. We have $\Delta = (\lambda x : \tau.(\lambda y : \rho.R^{\mu}))P^{\tau}$ occurring in M in the scenario $\Delta^{\rho\to\mu}Q^{\rho}$. Say Δ reduces to $\lambda y : \rho.R_1^{\mu}$. Then we create a new redex $(\lambda y : \rho.R_1^{\mu})Q^{\rho}$ of height $h(\rho \to \mu) < h(\tau \to (\rho \to \mu)) = h$.
  3. $\Delta = (\lambda x : (\rho \to \mu).x)(\lambda y : \rho.P^{\mu})$, which occurs in M as $\Delta^{\rho\to\mu}Q^{\rho}$. Reduction then gives the redex $(\lambda y : \rho.P^{\mu})Q^{\rho}$ of height $h(\rho \to \mu) < h$.
- Now Δ itself no longer appears in M, (lowering the count of redexes of maximal height by 1), and any newly created redexes have height $< h$.
- If we have $\Delta = (\lambda x : \tau.P^{\rho})Q^{\tau}$ and P contains multiple free occurrences of x, then all the redexes in Q are multiplied when performing β-reduction.
- However, our choice of Δ ensures that the height of any such redex in Q has height $< h$ (since these redexes are to the right of Δ in M).
- Thus, it is always the case that for the new term M', $m(M') < m(M)$ in the lexicographic order. So by the induction hypothesis, since M' can be reduced to β-NF, so can M.

□

Theorem 1.57 (Strong Normalisation for λ(→)) Let $\Gamma \Vdash M : \sigma$. Then there is no infinite reduction sequence $M \underset{\beta}{\longrightarrow} M_1 \longrightarrow \beta....$

Proof. Exercise (sheet 1). □

1.3. The Curry-Howard correspondence

Remark 1.58 We can think of a proposition φ as the "type of its proofs". The properties of simply-typed λ(→) match the rules of IPC rather precisely. We first show a correspondence between λ(→) and the implicational fragment IPC(→) of IPC that includes only the → connective, the axiom scheme, and the (→-I) and (→-E) rules. We later extend this to all of IPC by introducing more complex types to λ(→).

Start with IPC($\to$) and build a simply-typed λ-calculus out of it whose set of type variables U is precisely the set of primitive propositions of the logic. Clearly, the set of types Π then matches the set of propositions in the logic.

Proposition 1.59 (Curry-Howard correspondence for IPC($\to$)) Let Γ be a context for $\lambda(\to)$ and φ be a proposition. Then:

1. If $\Gamma \Vdash M : \varphi$, then $|\Gamma| = \{\tau \in \Pi : (x : \tau) \in \Gamma \text{ for some } x\} \underset{\mathrm{IPC}(\to)}{\vdash} \varphi$.
2. If $\Gamma \underset{\mathrm{IPC}(\to)}{\vdash} \varphi$, then there is a simply-typed λ-term $M \in \lambda(\to)$ such that $\{\left(x_\varphi : \varphi\right) : \varphi \in \Gamma\} \Vdash M : \varphi$.

Proof.

1. - Use induction on the derivation of $\Gamma \Vdash M : \varphi$.
   - Let x be a variable not occurring in Γ' and the derivation is of the form $\Gamma', x : \varphi \Vdash x : \varphi$, then we have that $|\Gamma', x : \varphi| \underset{\mathrm{IPC}(\to)}{\vdash} \varphi$ since $\varphi \underset{\mathrm{IPC}(\to)}{\vdash} \varphi$ (as $|\Gamma', x : \varphi| = |\Gamma'| \cup \{\varphi\}$).
   - If the derivation has M of the form $\lambda x : \sigma.N$ and $\varphi = (\sigma \to \tau)$, then we must have $\Gamma, x : \sigma \Vdash N : \tau$. By the induction hypothesis, we have that $|\Gamma, x : \sigma| \vdash \tau$, i.e. $|\Gamma|, \sigma \vdash \tau$. But then $|\Gamma| \vdash \sigma \to \tau$ by ($\to$-I).
   - If the derivation is of the form $\Gamma \Vdash (PQ) : \varphi$, then we must have $\Gamma \Vdash P : (\sigma \to \varphi)$ and $\Gamma \Vdash Q : \sigma$. By the induction hypothesis, we have $|\Gamma| \vdash (\sigma \to \varphi)$ and $|\Gamma| \vdash \sigma$, so $|\Gamma| \vdash \varphi$ by ($\to$-E).
2. - Use induction on the derivation of $\Gamma \vdash \varphi$.
   - Write $\Delta = \left\{\left(x_\psi : \psi\right) : \psi \in \Gamma\right\}$. Then we only have a few ways to construct a proof at a given stage. Say the derivation is of the form $\Gamma, \varphi \vdash \varphi$. If $\varphi \in \Gamma$, then clearly $\Delta \Vdash x_\varphi : \varphi$. If $\varphi \notin \Gamma$, then $\Delta, x_\varphi : \varphi \Vdash x_\varphi : \varphi$.
   - Suppose the derivation is at a stage of the form
   $$\frac{\Gamma \vdash \varphi \to \psi, \quad \Gamma \vdash \varphi}{\Gamma \vdash \psi}$$
   - Then by the induction hypothesis there are λ-terms M and N such that $\Delta \Vdash M : (\varphi \to \psi)$ and $\Delta \Vdash N : \varphi$, from which $\Delta \Vdash (MN) : \psi$.
   - If the stage is given by
   $$\frac{\Gamma, \varphi \vdash \psi}{\Gamma \vdash \varphi \to \psi},$$
   then there are two subcases:
     - If $\varphi \in \Gamma$, then the induction hypothesis gives $\Delta \Vdash M : \psi$ for some term M. By the weakening rule, we have $\Delta, x : \varphi \Vdash M : M : \psi$, where x is a variable not occurring in Δ. But then $\Delta \Vdash (\lambda x : \varphi.M) : (\varphi \to \psi)$.
     - If $\varphi \notin \Gamma$, then the induction hypothesis gives $\Delta, x_\varphi : \varphi \Vdash M : \psi$ for some λ-term M, thus $\Delta \Vdash \left(\lambda x_\varphi : \varphi.M\right) : (\varphi \to \psi)$.

$\square$

Example 1.60 Let φ, ψ be primitive propositions. The λ-term

$$\lambda f : (\varphi \to \psi) \to \varphi . \lambda g : \varphi \to \psi . g(fg)$$

has type $((\varphi \to \psi) \to \varphi) \to ((\varphi \to \psi) \to \psi)$, and therefore encodes a proof of that proposition in IPC($\to$).

$$\dfrac{\dfrac{\dfrac{\dfrac{g : [\varphi \to \psi] \quad f : (\varphi \to \psi) \to \varphi}{fg : \varphi \quad g : [\varphi \to \psi]} \ (\to\text{-E})}{g(fg) : \psi} \ (\to\text{-E})}{\lambda g . g(fg) : (\varphi \to \psi) \to \psi} \ (\to\text{-I}, \varphi \to \psi)}{\lambda f . \lambda g . g(fg) : ((\varphi \to \psi) \to \varphi) \to ((\varphi \to \psi) \to \psi)} \ (\to\text{-I}, (\varphi \to \psi) \to \varphi)$$

Definition 1.61 The **full simply-typed λ-calculus** consists of:

- A set of types Π generated by the grammar

$$\Pi \coloneqq U \mid \Pi \to \Pi \mid \Pi \times \Pi \mid \Pi + \Pi \mid 0 \mid 1$$

  Types of the form $\Pi \times \Pi$ are **product types**, those of the form $\Pi + \Pi$ are **coproduct types**, 0 is the **initial type**, and 1 is the **terminal type**. Again, U is a set of type variables.
- A set of terms Λ_Π generated by the grammar

$$\begin{aligned}\Lambda_\Pi \coloneqq V \mid \lambda V : \Pi . \Lambda_\Pi \mid \Lambda_\Pi \Lambda_\Pi \mid \pi_1(\Lambda_\Pi) \mid \pi_2(\Lambda_\Pi) \mid i_1(\Lambda_\Pi) \mid i_2(\Lambda_\Pi) \\ \mid \text{case}(\Lambda_\Pi ; V . \Lambda_\Pi ; V . \Lambda_\Pi) \mid * \mid !_\Pi \Lambda_\Pi\end{aligned}$$

  where V is a set of variables and $*$ is a constant.

We have the new typing rules:

$$\frac{\Gamma \Vdash M : \psi \times \varphi}{\Gamma \Vdash \pi_1(M) : \psi}$$

$$\frac{\Gamma \Vdash M : \psi \times \varphi}{\Gamma \Vdash \pi_2(M) : \varphi}$$

$$\frac{\Gamma \Vdash M : \psi \quad \Gamma \Vdash N : \varphi}{\Gamma \Vdash \langle M, N \rangle : \psi \times \varphi}$$

$$\frac{\Gamma \Vdash M : \psi}{\Gamma \Vdash \iota_1(M) : \psi + \varphi}$$

$$\frac{\Gamma \Vdash N : \varphi}{\Gamma \Vdash \iota_2(N) : \psi + \varphi}$$

$$\frac{\Gamma \Vdash L : \psi + \varphi \quad \Gamma, x : \psi \Vdash M : \rho \quad \Gamma, y : \varphi \Vdash N : \rho}{\Gamma \Vdash \text{case}\big(L; x^{\psi}.M; y^{\varphi}.N\big) : \rho}$$

$$\frac{}{\Gamma \Vdash * : 1}$$

$$\frac{\Gamma \Vdash M : 0}{\Gamma \Vdash !_{\varphi} M : \varphi}$$

We also have the new reduction rules:

- Projections: $\pi_1\langle M, N\rangle \underset{\beta}{\longrightarrow} M$ and $\pi_2\langle M, N\rangle \underset{\beta}{\longrightarrow} N$.
- Pairs: $\langle \pi_1 M, \pi_2 M\rangle \underset{\eta}{\longrightarrow} M$.
- Definition by cases: $\text{case}(\iota_1(M); x.M; y.L) \underset{\beta}{\longrightarrow} K[x := M]$ and $\text{case}(\iota_2(M); x.K; y.L) \underset{\beta}{\longrightarrow} L[y := M]$
- Unit: if $\Gamma \Vdash M : 1$, then $M \underset{\eta}{\longrightarrow} *$.

Remark 1.62 We can extend the Curry-Howard correspondence with these new types, letting

- $0 \longleftrightarrow \perp$.
- $\times \longleftrightarrow \wedge$.
- $+ \longleftrightarrow \vee$.
- $\rightarrow \longleftrightarrow \rightarrow$.

Example 1.63 Consider the following proof of $(\varphi \wedge \chi) \rightarrow (\psi \rightarrow \varphi)$:

$$\cfrac{\cfrac{\cfrac{[\varphi \wedge \chi] : p \quad [\psi] : b}{\varphi : \pi_1(p)}}{(\psi \rightarrow \varphi) : \lambda b : \psi.\pi_1(p)}}{((\varphi \wedge \chi) \rightarrow (\psi \rightarrow \varphi)) : \lambda p : \varphi \times \chi.\lambda b : \psi.\pi_1(p)}$$

We decorate this proof by turning the assumptions into variables.

Remark 1.64 We have the following correspondence:

Simply-typed λ-calculus	IPC
(Primitive) types	(Primitive) propositions
Variable	Hypothesis
Simply-typed λ-term	Proof
Type construction	Logical connective
Term inhabitation	Provability
Term reduction	Proof normalisation

1.4. Semantics for IPC

Definition 1.65 A **lattice** is a set L equipped with binary operations $\wedge$ and $\vee$ which are commutative and associative and satisfy the **absorption laws**: for all $a, b \in L$,

- $a \vee (a \wedge b) = a$,
- $a \wedge (a \vee b) = a$.

Definition 1.66 A lattice L is **distributive** if for all $a, b, c \in L$, $a \vee (b \wedge c) = (\wedge b) \vee (a \wedge c)$.

Definition 1.67 A lattice L is **bounded** if there are elements $\bot, \top \in L$ such that $a \vee \bot = a$ and $a \wedge \top = a$ for all $a \in L$.

Definition 1.68 A lattice L is **complemented** if it is bounded and for every $a \in L$, there is $a^* \in L$ such that $a \wedge a^* = \bot$ and $a \vee a^* = \top$.

Definition 1.69 A **Boolean algebra** is a complemented distributive lattice.

Remark 1.70 In any lattice, $\wedge$ and $\vee$ are idempotent. Moreover, we can define an ordering by setting $a \leq b$ if $a \wedge b = a$.

Example 1.71

- For every set I, the powerset $\mathbb{P}(I)$ of I with $\wedge = \cap$ and $\vee = \cup$ is the prototypical Boolean algebra.
- More generally, the clopen subsets of a topological space form a Boolean algebra with $\wedge = \cap$ and $\vee = \cup$.
- In particular, the set of finite and cofinite subsets of $\mathbb{Z}$ is a Boolean algebra.

Proposition 1.72 Let L be a bounded lattice and $\leq$ be the order induced by the operations in L ($a \leq b \iff a \wedge b = a$). Then $\leq$ is a partial order with least element $\bot$ and greatest element $\top$, and for all $a, b \in L$, $a \wedge b = \inf\{a, b\}$ and $a \vee b = \sup\{a, b\}$. Conversely, every partial order with all finite inf's and sup's is a bounded lattice.

Proof. Exercise. □

Classically, we say that $\Gamma \vDash t$ if for every valuation $v : L \to \{0, 1\}$ such that $v(p) = 1$ for all $p \in \Gamma$, we have $v(t) = 1$. We might want to replace $\{0, 1\}$ with some other Boolean algebra to get semantics for IPC, with an accompanying completeness theorem. But Boolean algebras believe in the LEM!

Definition 1.73 A **Heyting algebra** H is a bounded lattice equipped with a binary operation $\Rightarrow$ such that for all $a, b, c \in H$,

$$a \wedge b \leq c \text{ iff } a \leq (b \Rightarrow c).$$

This can be thought of as an algebraic version of the deduction theorem. A **Heyting homomorphism** (morphism of Heyting algebras) is a function that preserves all finite meets ($\wedge$), finite joins ($\vee$), and $\Rightarrow$.

Example 1.74

1. Every Boolean algebra is a Heyting algebra: define $a \Rightarrow b := a^* \vee b$ (a^* should be thought of as $\neg a$). Note that we must have $a^* = (a \Rightarrow \perp)$.
2. Every topology on a set X is a Heyting algebra, where $(U \Rightarrow V) := \operatorname{int}((X - U) \cup V)$.
3. A finite distributive lattice is a Heyting algebra.

Definition 1.75 Let H be a Heyting algebra and L be a propositional language with a set of primitive propositions P. An ***H*-valuation** is a function $v : P \to H$, extended recursively to L, by setting:

- $v(\perp) = \perp$.
- $v(A \wedge B) = v(A) \wedge v(B)$.
- $v(A \vee B) = v(A) \vee v(B)$.
- $v(P \to Q) = v(A) \Rightarrow v(B)$.

Definition 1.76 A proposition $A \in L$ is H-valid if $v(A) = \top$ for all H-valuations v, and is an ***H*-consequence** of a (finite) set of propositions Γ if $v(\bigwedge \Gamma) \leq v(A)$ (we write $\Gamma \underset{H}{\models} P$).

Lemma 1.77 (Soundness of Heyting Semantics) Let H be a Heyting algebra and $v : L \to H$ be an H-valuation. If $\Gamma \underset{\text{IPC}}{\vdash} A$, then $\Gamma \underset{H}{\models} A$.

Proof. By induction on the structure of the proof $\Gamma \vdash A$.

- (Ax): $v(\bigwedge \Gamma \wedge A) = v(\bigwedge \Gamma) \wedge v(A) \leq v(A)$.
- ($\wedge$-I): $A = B \wedge C$ and we have derivations $\Gamma_1 \vdash B$ and $\Gamma_2 \vdash C$, with $\Gamma_1, \Gamma_2 \subseteq \Gamma$. By the indutive hypothesis, we have $v(\bigwedge \Gamma) \leq v(\bigwedge \Gamma_1) \wedge v(\bigwedge \Gamma_2) \leq v(B) \wedge v(C) = v(B \wedge C)$, i.e. $\Gamma \underset{H}{\models} A$.
- ($\to$-I): $A = B \to C$, so we must have $\Gamma \cup \{B\} \vdash C$. By the inductive hypothesis, we have $v(\bigwedge \Gamma) \wedge v(B) = v(\bigwedge \Gamma \wedge B) \leq v(C)$. By the definition of $\Rightarrow$, this implies $v(\bigwedge \Gamma) \leq (v(B) \Rightarrow v(C)) = v(B \to C) = v(A)$, i.e. $\Gamma \underset{H}{\models} A$.
- ($\vee$-I): $A = B \vee C$ and WLOG we have a derivation $\Gamma \vdash B$. By the inductive hypothesis, we have $v(\bigwedge \Gamma) \leq v(B)$, but $v(B \vee C) = v(B) \vee v(C) = \sup\{v(B), v(C)\}$, and so $v(B) \leq v(B \vee C)$.

$\square$