# Logics and type systems

## 1 Introduction

This document summarizes some logics and type systems from the course Type Theory and Coq at the Radboud University Nijmegen. The logics and type systems in this document are (most presented in two different ways):

| | | |
|---|---|---|
| Propositional logic (Sections 2.1 and 2.2) | $\lambda\!\to$ | (Sections 3.2 an 3.4) |
| Second order propositional logic (Sections 2.3 and 2.4) | $\lambda 2$ | (Sections 3.3 and 3.4) |
| Predicate logic (Sections 2.5 and 2.6) | $\lambda P$ | (Section 3.4) |

## 2 Logics

These are all intuitionistic logics. The minimal variant of the logic is indicated by coloring red the parts that belong to this. The repeated part of a logic when a logic is extended is colored green in the syntax, and omitted in the rules.

### 2.1 Propositional logic (implicit contexts)

**Syntax**

$$A, B ::= a \mid A \to B \mid \top \mid \bot \mid \neg A \mid A \wedge B \mid A \vee B$$

**Rules**

$$\frac{\begin{matrix}[A^x] \\ \vdots \\ B\end{matrix}}{A \to B}\, I[x]\!\to \qquad \frac{\begin{matrix}\vdots \\ A \to B\end{matrix} \qquad \begin{matrix}\vdots \\ A\end{matrix}}{B}\, E\!\to$$

$$\frac{}{\top}\, I\top \qquad \frac{\begin{matrix}\vdots \\ \bot\end{matrix}}{A}\, E\bot$$

$$\dfrac{\begin{matrix}[A^x]\\ \vdots \\ \bot\end{matrix}}{\neg A}\, I[x]\neg \qquad \dfrac{\begin{matrix}\vdots \\ \neg A\end{matrix} \quad \begin{matrix}\vdots \\ A\end{matrix}}{\bot}\, E\neg$$

$$\dfrac{\begin{matrix}\vdots \\ A\end{matrix} \quad \begin{matrix}\vdots \\ B\end{matrix}}{A \wedge B}\, I\wedge \qquad \dfrac{\begin{matrix}\vdots \\ A \wedge B\end{matrix}}{A}\, El\wedge \qquad \dfrac{\begin{matrix}\vdots \\ A \wedge B\end{matrix}}{B}\, Er\wedge$$

$$\dfrac{\begin{matrix}\vdots \\ A\end{matrix}}{A \vee B}\, Il\vee \qquad \dfrac{\begin{matrix}\vdots \\ B\end{matrix}}{A \vee B}\, Ir\vee \qquad \dfrac{\begin{matrix}\vdots \\ A \vee B\end{matrix} \quad \begin{matrix}\vdots \\ A \to C\end{matrix} \quad \begin{matrix}\vdots \\ B \to C\end{matrix}}{C}\, E\vee$$

## 2.2 Propositional logic (explicit contexts)

**Syntax**

$$\begin{aligned} A, B &::= a \mid A \to B \mid \top \mid \bot \mid \neg A \mid A \wedge B \mid A \vee B \\ \Gamma &:= \cdot \mid \Gamma,\, A \end{aligned}$$

**Rules**

$$\dfrac{}{\Gamma \vdash A}\, \mathsf{ass} \qquad \text{for } A \in \Gamma$$

$$\dfrac{\Gamma,\, A \vdash B}{\Gamma \vdash A \to B}\, I{\to} \qquad \dfrac{\Gamma \vdash A \to B \quad \Gamma \vdash A}{\Gamma \vdash B}\, E{\to}$$

$$\dfrac{}{\Gamma \vdash \top}\, I\top \qquad \dfrac{\Gamma \vdash \bot}{\Gamma \vdash A}\, E\bot$$

$$\dfrac{\Gamma,\, A \vdash \bot}{\Gamma \vdash \neg A}\, I\neg \qquad \dfrac{\Gamma \vdash \neg A \quad \Gamma \vdash A}{\Gamma \vdash \bot}\, E\neg$$

$$\dfrac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\, I\wedge \qquad \dfrac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\, El\wedge \qquad \dfrac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\, Er\wedge$$

$$\dfrac{\Gamma \vdash A}{\Gamma \vdash A \vee B}\, Il\vee \qquad \dfrac{\Gamma \vdash B}{\Gamma \vdash A \vee B}\, Ir\vee \qquad \dfrac{\Gamma \vdash A \vee B \quad \Gamma \vdash A \to C \quad \Gamma \vdash B \to C}{\Gamma \vdash C}\, E\vee$$

## 2.3 Second order propositional logic (implicit contexts)

**Syntax**

$$A, B ::= a \mid A \to B \mid \top \mid \bot \mid \neg A \mid A \wedge B \mid A \vee B \mid \forall a.\, A \mid \exists a.\, A$$

**Rules**

See Section 2.1 for the rules for the propositional connectives.

$$\frac{\begin{array}{c}\vdots \\ A\end{array}}{\forall a.\, A}\, I\forall \qquad \frac{\begin{array}{c}\vdots \\ \forall a.\, A\end{array}}{A[a := B]}\, E\forall$$

$$\frac{\begin{array}{c}\vdots \\ A[a := B]\end{array}}{\exists a.\, A}\, I\exists \qquad \frac{\begin{array}{c}\vdots \\ \exists a.\, A\end{array} \qquad \begin{array}{c}\vdots \\ \forall a.\, A \to C\end{array}}{C}\, E\exists$$

The variable condition for $I\forall$ is that $a$ should not be free in open assumptions. The variable condition for $E\exists$ is that $a$ should not be free in $C$.

## 2.4 Second order propositional logic (explicit contexts)

**Syntax**

$$A, B ::= a \mid A \to B \mid \top \mid \bot \mid \neg A \mid A \wedge B \mid A \vee B \mid \forall a.\, A \mid \exists a.\, A$$
$$\Gamma ::= \cdot \mid \Gamma,\, A$$

**Rules**

See Section 2.2 for the assumption rule and the rules for the propositional connectives.

$$\frac{\Gamma \vdash A}{\Gamma \vdash \forall a.\, A}\, I\forall \qquad \frac{\Gamma \vdash \forall a.\, A}{\Gamma \vdash A[a := B]}\, E\forall$$

$$\frac{\Gamma \vdash A[a := B]}{\Gamma \vdash \exists a.\, A}\, I\exists \qquad \frac{\Gamma \vdash \exists a.\, A \qquad \Gamma \vdash \forall a.\, A \to C}{\Gamma \vdash C}\, E\exists$$

The variable condition for $I\forall$ is that $a$ should not be free in $\Gamma$. The variable condition for $E\exists$ is that $a$ should not be free in $C$.

## 2.5 Predicate logic (implicit contexts)

### Syntax

$$
\begin{aligned}
M, N &:= f(\overrightarrow{M}) \\
\overrightarrow{M} &:= \cdot \mid \overrightarrow{M},\, N \\
A, B &::= p(\overrightarrow{M}) \mid A \to B \mid \top \mid \bot \mid \neg A \mid A \land B \mid A \lor B \mid \forall x.\, A \mid \exists x.\, A
\end{aligned}
$$

### Rules

See Section 2.1 for the rules for the propositional connectives.

$$
\frac{\begin{matrix}\vdots \\ A\end{matrix}}{\forall x.\, A}\; I\forall \qquad \frac{\begin{matrix}\vdots \\ \forall x.\, A\end{matrix}}{A[x := M]}\; E\forall
$$

$$
\frac{\begin{matrix}\vdots \\ A[x := M]\end{matrix}}{\exists x.\, A}\; I\exists \qquad \frac{\begin{matrix}\vdots \\ \exists x.\, A\end{matrix} \qquad \begin{matrix}\vdots \\ \forall x.\, A \to C\end{matrix}}{C}\; E\exists
$$

The variable condition for $I\forall$ is that $x$ should not be free in open assumptions. The variable condition for $E\exists$ is that $x$ should not be free in $C$.

## 2.6 Predicate logic (explicit contexts)

### Syntax

$$
\begin{aligned}
M, N &:= f(\overrightarrow{M}) \\
\overrightarrow{M} &:= \cdot \mid \overrightarrow{M}, N \\
A, B &::= p(\overrightarrow{M}) \mid A \to B \mid \top \mid \bot \mid \neg A \mid A \land B \mid A \lor B \mid \forall x.\, A \mid \exists x.\, A \\
\Gamma &:= \cdot \mid \Gamma,\, A
\end{aligned}
$$

### Rules

See Section 2.2 for the assumption rule and the rules for the propositional connectives.

$$
\frac{\Gamma \vdash A}{\Gamma \vdash \forall x.\, A}\; I\forall \qquad \frac{\Gamma \vdash \forall x.\, A}{\Gamma \vdash A[x := M]}\; E\forall
$$

$$
\frac{\Gamma \vdash A[x := M]}{\Gamma \vdash \exists x.\, A}\; I\exists \qquad \frac{\Gamma \vdash \exists x.\, A \qquad \Gamma \vdash \forall x.\, A \to C}{\Gamma \vdash C}\; E\exists
$$

The variable condition for $I\forall$ is that $x$ should not be free in $\Gamma$. The variable condition for $E\exists$ is that $x$ should not be free in $C$.

# 3 Type systems

The systems given in this section are all Church-style type systems. To obtain Curry-style versions, omit the red parts. The type systems correspond to logics. To get a logic from a type system, omit the blue proof term parts. The repeated part of a type system when a type system is extended is colored green in the syntax.

A type system is given by syntax, typing rules and reduction rules, but this document does not include the reduction rules of the systems (which in all cases is just beta reduction).

## 3.1 Untyped lambda calculus

This is of course not a type system, but the syntax is here for comparison.

**Syntax**

$$M, N := x \mid \lambda x.M \mid MN$$

## 3.2 Simply typed lambda calculus = Simple Type Theory

This is the non-PTS version of $\lambda{\to}$.

**Syntax**

$$\begin{aligned} A, B &:= a \mid A \to B \\ M, N &:= x \mid \lambda x : A.\, M \mid MN \\ \Gamma &:= \cdot \mid \Gamma, x : A \end{aligned}$$

**Rules**

$$\frac{}{\Gamma \vdash x : A} \quad \text{for } (x : A) \in \Gamma$$

$$\frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash (\lambda x : A.\, M) : A \to B} \qquad \frac{\Gamma \vdash M : A \to B \quad \Gamma \vdash N : A}{\Gamma \vdash MN : B}$$

### 3.3 Polymorphic lambda calculus = System F

This is the non-PTS version of $\lambda 2$.

$$
\begin{aligned}
A, B &:= a \mid A \to B \mid \forall a.\, A \\
M, N &:= x \mid \lambda x : A.\, M \mid M N \mid \Lambda a.\, M \mid M A \\
\Gamma &:= \cdot \mid \Gamma,\, x : A
\end{aligned}
$$

**Rules**

$$\frac{}{\Gamma \vdash x : A} \quad \text{for } (x : A) \in \Gamma$$

$$\frac{\Gamma,\, x : A \vdash M : B}{\Gamma \vdash (\lambda x : A.\, M) : A \to B} \qquad \frac{\Gamma \vdash M : A \to B \quad \Gamma \vdash N : A}{\Gamma \vdash M N : B}$$

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash (\Lambda a.\, M) : \forall a.\, A} \qquad \frac{\Gamma \vdash M : \forall a.\, A}{\Gamma \vdash M B : A[a := B]}$$

The variable condition for the typing rule for $(\Lambda a.\, M)$ is that $a$ should not occur in $\Gamma$.

### 3.4 The Pure Type Systems of the lambda cube

**Syntax**

$$
\begin{aligned}
s, s' &:= * \mid \square \\
M, N, A, B &:= s \mid x \mid \lambda x : A.\, M \mid M N \mid \Pi x : A.\, B \\
\Gamma &:= \cdot \mid \Gamma,\, x : A
\end{aligned}
$$

$$A \to B := \Pi x : A.\, B$$

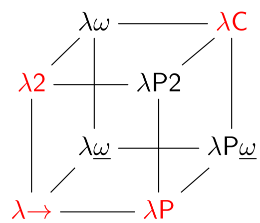
**Rules**

$$\frac{}{\vdash * : \square} \qquad \frac{\Gamma \vdash A : s}{\Gamma,\, x : A \vdash x : A} \qquad \frac{\Gamma \vdash M : A \quad \Gamma \vdash B : s}{\Gamma,\, y : B \vdash M : A}$$

$$\frac{\Gamma,\, x : A \vdash M : B \quad \Gamma \vdash (\Pi x : A.\, B) : s}{\Gamma \vdash (\lambda x : A.\, M) : \Pi x : A.\, B} \qquad \frac{\Gamma \vdash F : \Pi x : A.\, B \quad \Gamma \vdash M : A}{\Gamma \vdash F M : B[x := M]}$$

$$\frac{\Gamma \vdash A : s \quad \Gamma,\, x : A \vdash B : s'}{\Gamma \vdash (\Pi x : A.\, B) : s'} \quad \text{for } (s, s', s') \in \mathcal{R}$$

$$\frac{\Gamma \vdash M : A \quad \Gamma \vdash A' : s}{\Gamma \vdash M : A'} \qquad \text{if } A =_\beta A'$$




| | |
|---|---|
| $\lambda\!\rightarrow$ | $\mathcal{R} = \{(*, *, *)\}$ |
| $\lambda 2$ | $\mathcal{R} = \{(*, *, *), (\Box, *, *)\}$ |
| $\lambda P$ | $\mathcal{R} = \{(*, *, *), \qquad\qquad (*, \Box, \Box)\}$ |
| $\lambda C$ | $\mathcal{R} = \{(*, *, *), (\Box, *, *), (*, \Box, \Box), (\Box, \Box, \Box)\}$ |

## 3.5 Pure Type Systems

### Syntax

$$\begin{aligned} M, N, A, B &:= s \mid x \mid \lambda x : A.\, M \mid M N \mid \Pi x : A.\, B \\ \Gamma &:= \cdot \mid \Gamma,\, x : A \end{aligned}$$

### Rules

A PTS is defined by giving:

| | |
|---|---|
| a set of sorts | $\mathcal{S}$ |
| a set of axioms | $\mathcal{A} \subseteq \mathcal{S} \times \mathcal{S}$ |
| a set of rules | $\mathcal{R} \subseteq \mathcal{S} \times \mathcal{S} \times \mathcal{S}$ |

Five of the seven typing rules are the same as in Section 3.4. The only two different ones, which are generalizations of the rules in the lambda cube, are the first and the sixth: the start/axiom rule and the product rule.

$$\frac{}{\vdash s : s'} \qquad \text{for } (s, s') \in \mathcal{A}$$

$$\frac{\Gamma \vdash A : s \quad \Gamma, x : A \vdash B : s'}{\Gamma \vdash (\Pi x : A.\, B) : s''} \qquad \text{for } (s, s', s'') \in \mathcal{R}$$