MTAT.05.105 Type Theory

The λ-cube and pure type systems

Towards the λ-cube

- The simply typed λ-calculus $\lambda\!\rightarrow$ provides a possibility to define terms depending on terms using λ-abstraction and application.
- The second-order λ-calculus $\lambda 2$ adds the posibility to define terms depending on types.
- But what about types depending on types or types depending on terms?
- Indeed there are several λ-calculi which allow these kinds of dependencies.
- In 1991, Henk Barendregt proposed a classification of 8 different calculi into λ-cube and provided a uniform description of these calculi.

Towards the λ-cube: the system $\lambda\underline{\omega}$

- $\lambda\underline{\omega}$ extends $\lambda\!\to$ with types depending on types.
  - I.e., provides type-level abstraction and application constructs.
- Now, types should also have "types".
  - "Types" for types are called kinds.

$$\mathbb{K} \;=\; * \;\mid\; \mathbb{K} \to \mathbb{K}$$

  - "Ordinary" types have the kind $*$, unary type constructors have the kind $* \to *$, etc.
  - There is also a "type" of kinds, $\Box$, i.e., $k \in \mathbb{K} \iff k : \Box$.

Towards the λ-cube: the system $\lambda\underline{\omega}$

- Sorts:

$$
\begin{array}{lcll}
s & ::= & * & \text{star} \\
  & | & \Box & \text{square}
\end{array}
$$

- Expressions:

$$
\begin{array}{lcll}
e & ::= & x & \text{variable} \\
  & | & s & \text{sort constant} \\
  & | & e_1\, e_2 & \text{application} \\
  & | & \lambda x : e_1 .\, e_2 & \text{abstraction} \\
  & | & e_1 \rightarrow e_2 & \text{function}
\end{array}
$$

- Contexts are finite linearly ordered sets of statements $x : e$ with distinct variables as subjects.

$$\Gamma = [x_1 : e_1,\ x_2 : e_2,\ \ldots,\ x_n : e_n]$$

Towards the λ-cube: the system $\lambda\underline{\omega}$

- Everything is an expression; there is no separate term, type or kind grammar.
- The typing rules determine which expressions are terms, types or kinds.

$$
\begin{array}{lcl}
K \text{ is a kind} & \Longleftrightarrow & \Gamma \vdash K : \Box \\
T \text{ is a type} & \Longleftrightarrow & \Gamma \vdash T : K : \Box \\
E \text{ is a term} & \Longleftrightarrow & \Gamma \vdash E : T : K : \Box
\end{array}
$$

- β-reduction:

$$(\lambda x{:}e_1.\ e_2)\, e_3 \ \to_\beta\ e_2[x \mapsto e_3]$$

Towards the λ-cube: the system $\lambda\underline{\omega}$

- Typing rules:

$$\frac{}{\vdash * : \Box}\ [\texttt{AXIOM}] \qquad \frac{\Gamma \vdash e_1 : A \to B \quad \Gamma \vdash e_2 : A}{\Gamma \vdash e_1\, e_2 : B}\ [\texttt{APP}]$$

$$\frac{\Gamma \vdash A : s}{\Gamma, x{:}A \vdash x : A}\ [\texttt{START}] \qquad \frac{\Gamma, x{:}A \vdash e : B \quad \Gamma \vdash A \to B : s}{\Gamma \vdash (\lambda x{:}A.\, e) : A \to B}\ [\texttt{ABS}]$$

$$\frac{\Gamma \vdash e : A \quad \Gamma \vdash B : s}{\Gamma, x{:}B \vdash e : A}\ [\texttt{WEAK}] \qquad \frac{\Gamma \vdash e : B \quad \Gamma \vdash A : s \quad A =_\beta B}{\Gamma \vdash e : A}\ [\texttt{CONV}]$$

$$\frac{\Gamma \vdash A : s \quad \Gamma \vdash B : s}{\Gamma \vdash A \to B : s}\ [\texttt{ARROW}]$$

  – Here $s \in \{*, \Box\}$.

Towards the λ-cube: the system $\lambda\underline{\omega}$

Examples:

$$
\begin{array}{rcl}
\alpha{:}*, \beta{:}* & \vdash & \alpha \to \beta : * \\
\alpha{:}*, \beta{:}*, x{:}\alpha \to \beta & \vdash & x : \alpha \to \beta \\
\alpha{:}*, \beta{:}* & \vdash & (\lambda x{:}\alpha \to \beta.\ x) : (\alpha \to \beta) \to (\alpha \to \beta) \\
 & \vdash & (\lambda \alpha{:}*.\ \alpha \to \alpha) : * \to * \\
\alpha{:}* & \vdash & (\lambda x{:}D\alpha.\ x) : D(D\alpha)
\end{array}
$$

where $D \equiv \lambda\alpha{:}*.\ \alpha \to \alpha$.

Towards the λ-cube: the system λP

- λP extends $\lambda{\rightarrow}$ with dependent types (i.e. types depending on terms).
- The dependent product $\Pi x : A.\, B$ is the type of functions from values x of type A to values of type B, in which the type B may depend on the argument value x.
- Note: $A \rightarrow B \;\equiv\; \Pi\_{:}A.\, B$.

Towards the λ-cube: the system λP

- Expressions:

$$
\begin{array}{rcll}
e & ::= & x & \text{variable} \\
 & | & * & \text{star} \\
 & | & \Box & \text{square} \\
 & | & e_1\, e_2 & \text{application} \\
 & | & \lambda x : e_1 .\, e_2 & \text{abstraction} \\
 & | & \Pi x : e_1 .\, e_2 & \text{product}
\end{array}
$$

- β-reduction:

$$(\lambda x{:}e_1.\ e_2)\, e_3 \ \rightarrow_\beta\ e_2[x \mapsto e_3]$$

Towards the λ-cube: the system λP

- Typing rules:

$$\frac{}{\vdash * : \Box}\ [\text{AXIOM}] \qquad \frac{\Gamma \vdash e_1 : (\Pi x{:}A.B) \quad \Gamma \vdash e_2 : A}{\Gamma \vdash e_1\, e_2 : B[x \mapsto e_2]}\ [\text{APP}]$$

$$\frac{\Gamma \vdash A : s}{\Gamma,\, x{:}A \vdash x : A}\ [\text{START}] \qquad \frac{\Gamma,\, x{:}A \vdash e : B \quad \Gamma \vdash (\Pi x{:}A.B : s)}{\Gamma \vdash (\lambda x{:}A.\, e) : (\Pi x{:}A.B)}\ [\text{ABS}]$$

$$\frac{\Gamma \vdash e : A \quad \Gamma \vdash B : s}{\Gamma,\, x{:}B \vdash e : A}\ [\text{WEAK}] \qquad \frac{\Gamma \vdash e : B \quad \Gamma \vdash A : s \quad A =_\beta B}{\Gamma \vdash e : A}\ [\text{CONV}]$$

$$\frac{\Gamma \vdash A : * \quad \Gamma,\, x{:}A \vdash B : s}{\Gamma \vdash (\Pi x{:}A.B) : s}\ [\text{PROD}]$$

  – Here $s \in \{*, \Box\}$.

Towards the λ-cube: the system λP

Examples:

$$
\begin{array}{rcl}
A{:}* & \vdash & (A \to *) : \Box \\
A{:}*,\ P{:}A{\to}*,\ a{:}A & \vdash & P\,a : * \\
A{:}*,\ P{:}A{\to}*,\ a{:}A & \vdash & (P\,a \to *) : \Box \\
A{:}*,\ P{:}A{\to}* & \vdash & (\Pi a{:}A.\ P\,a \to *) : \Box \\
A{:}*,\ P{:}A{\to}* & \vdash & (\lambda a{:}A.\ \lambda x{:}P a.\ x) : (\Pi a{:}A.\ P\,a \to\ P\,a)
\end{array}
$$

The λ-cube

- Expressions:

$$
\begin{array}{rcll}
e & ::= & x & \text{variable} \\
 & | & * & \text{star} \\
 & | & \Box & \text{square} \\
 & | & e_1\, e_2 & \text{application} \\
 & | & \lambda x : e_1 .\, e_2 & \text{abstraction} \\
 & | & \Pi x : e_1 .\, e_2 & \text{product}
\end{array}
$$

- β-reduction:

$$(\lambda x{:}e_1.\ e_2)\, e_3 \ \to_\beta\ e_2[x \mapsto e_3]$$

- Abbrevations:

$$
\begin{array}{lcl}
A \to B & \equiv & \Pi\_{:}A.\ B \\
\forall \alpha . A & \equiv & \Pi \alpha{:}*.\ A
\end{array}
$$

- Dependency rules:

$$\mathcal{R} \subseteq \{(*,*),\ (*,\Box),\ (\Box,*),\ (\Box,\Box)\}$$

The λ-cube

- Typing rules:

$$\frac{}{\vdash * : \Box}\ [\texttt{AXIOM}] \qquad \frac{\Gamma \vdash e_1 : (\Pi x{:}A.B) \quad \Gamma \vdash e_2 : A}{\Gamma \vdash e_1\, e_2 : B[x \mapsto e_2]}\ [\texttt{APP}]$$

$$\frac{\Gamma \vdash A : s}{\Gamma,\ x{:}A \vdash x : A}\ [\texttt{START}] \qquad \frac{\Gamma,\ x{:}A \vdash e : B \quad \Gamma \vdash (\Pi x{:}A.B) : s}{\Gamma \vdash (\lambda x{:}A.\, e) : (\Pi x{:}A.B)}\ [\texttt{ABS}]$$

$$\frac{\Gamma \vdash e : A \quad \Gamma \vdash B : s}{\Gamma,\ x{:}B \vdash e : A}\ [\texttt{WEAK}] \qquad \frac{\Gamma \vdash e : B \quad \Gamma \vdash A : s \quad A =_\beta B}{\Gamma \vdash e : A}\ [\texttt{CONV}]$$

$$\frac{\Gamma \vdash A : s_1 \quad \Gamma,\ x{:}A \vdash B : s_2}{\Gamma \vdash (\Pi x{:}A.B) : s_2}\ [\texttt{PROD}]$$

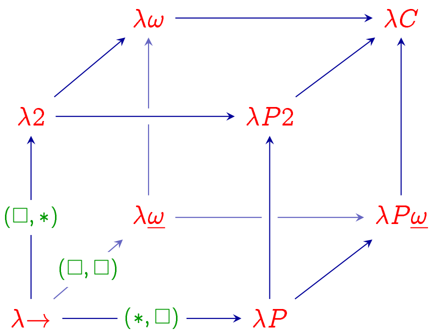
  - Here $s \in \{*, \Box\}$ and $(s_1, s_2) \in \mathcal{R}$.

The λ-cube

The systems of the λ-cube:

System	Set of rules ($\mathcal{R}$)			
$\lambda{\rightarrow}$	$(*,*)$			
$\lambda 2$	$(*,*)$	$(\Box,*)$		
λP	$(*,*)$		$(*,\Box)$	
$\lambda P2$	$(*,*)$	$(\Box,*)$	$(*,\Box)$	
$\lambda\underline{\omega}$	$(*,*)$			$(\Box,\Box)$
$\lambda\omega$	$(*,*)$	$(\Box,*)$		$(\Box,\Box)$
$\lambda P\underline{\omega}$	$(*,*)$		$(*,\Box)$	$(\Box,\Box)$
$\lambda P\omega = \lambda C$	$(*,*)$	$(\Box,*)$	$(*,\Box)$	$(\Box,\Box)$

The λ-cube



Properties of the systems of the λ-cube

- Church-Rosser: For any expressions A, B_1 and B_2,

$$[A \twoheadrightarrow B_1 \wedge A \twoheadrightarrow B_2] \quad \Longrightarrow \quad \exists C.\ [B_1 \twoheadrightarrow C \wedge B_2 \twoheadrightarrow C]$$

- Subject Reduction: For any system in the λ-cube,

$$\Gamma \vdash A : B \ \wedge \ A \twoheadrightarrow A' \quad \Longrightarrow \quad \Gamma \vdash A' : B$$

- Strong Normalization: For any system in the λ-cube, if $\Gamma \vdash A : B$, then all reductions starting from A or B terminate.

- Unicity of types: For any system in the λ-cube,

$$\Gamma \vdash A : B \ \wedge \ \Gamma \vdash A : B' \quad \Longrightarrow \quad B =_\beta B'$$

The λ-cube: examples

- In $\lambda\!\rightarrow$ the following can be derived:

$$
\begin{array}{lrcl}
(1.1) & A{:}* & \vdash & (\Pi x{:}A.\ A) : * \\
(1.2) & A{:}* & \vdash & (\lambda a{:}A.\ a) : (\Pi x{:}A.\ A) \\
(1.3) & A{:}*,\ b{:}A & \vdash & ((\lambda a{:}A.\ a)\, b) : A
\end{array}
$$

The λ-cube: examples

- In $\lambda\!\to$ the following can be derived:

$$
\begin{array}{lrcl}
(1.1) & A{:}* & \vdash & (\Pi x{:}A.\ A) : * \\
(1.2) & A{:}* & \vdash & (\lambda a{:}A.\ a) : (\Pi x{:}A.\ A) \\
(1.3) & A{:}*,\ b{:}A & \vdash & ((\lambda a{:}A.\ a)\, b) : A
\end{array}
$$

- Derivation of (1.1):

$$
\dfrac{
  \dfrac{\dfrac{}{\vdash * : \Box}\ [\texttt{AXIOM}]}{A{:}* \vdash A : *}\ [\texttt{START}]
  \qquad
  \dfrac{
    \dfrac{\dfrac{}{\vdash * : \Box}\ [\texttt{AXIOM}]}{A{:}* \vdash A : *}\ [\texttt{START}]
    \quad
    \dfrac{\dfrac{}{\vdash * : \Box}\ [\texttt{AXIOM}]}{A{:}* \vdash A : *}\ [\texttt{START}]
  }{A{:}*,\ x{:}A \vdash A : *}\ [\texttt{WEAK}]
}{A{:}* \vdash (\Pi x{:}A.\ A) : *}\ [\texttt{PROD}(*,*)]
$$

The λ-cube: examples

- In $\lambda\!\rightarrow$ the following can be derived:

$$
\begin{array}{lrcl}
(1.1) & A{:}* & \vdash & (\Pi x{:}A.\ A) : * \\
(1.2) & A{:}* & \vdash & (\lambda a{:}A.\ a) : (\Pi x{:}A.\ A) \\
(1.3) & A{:}*,\ b{:}A & \vdash & ((\lambda a{:}A.\ a)\, b) : A
\end{array}
$$

- Derivation of (1.2):

$$
\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash * : \Box}\ [\texttt{AXIOM}]}{A{:}* \vdash A : *}\ [\texttt{START}]}{A{:}*,\ a{:}A \vdash a : A}\ [\texttt{START}] \qquad \dfrac{(1.1)}{A{:}* \vdash (\Pi x{:}A.\ A) : *}}{A{:}* \vdash (\lambda a{:}A.\ a) : (\Pi x{:}A.\ A)}\ [\texttt{ABS}]
$$

The λ-cube: examples

- In $\lambda\!\to$ the following can be derived:

$$
\begin{array}{lrcl}
(1.1) & A{:}* & \vdash & (\Pi x{:}A.\ A) : * \\
(1.2) & A{:}* & \vdash & (\lambda a{:}A.\ a) : (\Pi x{:}A.\ A) \\
(1.3) & A{:}*,\ b{:}A & \vdash & ((\lambda a{:}A.\ a)\, b) : A
\end{array}
$$

- Derivation of (1.3):

$$
\dfrac{
  \dfrac{
    \dfrac{(1.2)}{A{:}* \vdash (\lambda a{:}A.\ a) : (\Pi x{:}A.\ A)}
    \qquad
    \dfrac{\dfrac{}{\vdash * : \Box}\,[\texttt{AXIOM}]}{A{:}* \vdash A : *}\,[\texttt{START}]
  }{A{:}*,\ b{:}A \vdash (\lambda a{:}A.\ a) : (\Pi x{:}A.\ A)}\,[\texttt{WEAK}]
  \qquad
  \dfrac{\dfrac{\dfrac{}{\vdash * : \Box}\,[\texttt{AXIOM}]}{A{:}* \vdash A : *}\,[\texttt{START}]}{A{:}*,\ b{:}A \vdash b : A}\,[\texttt{START}]
}{A{:}*,\ b{:}A \vdash ((\lambda a{:}A.\ a)\, b) : A}\,[\texttt{APP}]
$$

The λ-cube: examples

- In $\lambda 2$ the following can be derived:

$$
\begin{array}{lrcl}
(2.1) & & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a) : (\Pi\alpha{:}*.\ \alpha \to \alpha) \\
(2.2) & A{:}* & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A : A \to A \\
(2.3) & A{:}*,\ b{:}A & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A\, b : A
\end{array}
$$

The λ-cube: examples

- In $\lambda 2$ the following can be derived:

$$\begin{array}{llcl}
(2.1) & & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a) : (\Pi\alpha{:}*.\ \alpha \to \alpha) \\
(2.2) & A{:}* & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A : A \to A \\
(2.3) & A{:}*,\ b{:}A & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A\, b : A
\end{array}$$

- Derivation of (2.1):

$$\dfrac{\dfrac{(1.2)}{\alpha{:}* \vdash (\lambda a{:}\alpha.\ a) : \alpha \to \alpha} \qquad \dfrac{\dfrac{}{\vdash * : \Box}\,[\texttt{AXIOM}] \qquad \dfrac{(1.1)}{\alpha{:}* \vdash \alpha \to \alpha : *}}{\vdash (\Pi\alpha{:}*.\ \alpha \to \alpha) : *}\,[\texttt{PROD}(\Box,*)]}{\vdash (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a) : (\Pi\alpha{:}*.\ \alpha \to \alpha)}\,[\texttt{ABS}]$$

The λ-cube: examples

- In $\lambda 2$ the following can be derived:

$$
\begin{array}{lrcl}
(2.1) & & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a) : (\Pi\alpha{:}*.\ \alpha \to \alpha) \\
(2.2) & A{:}* & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A : A \to A \\
(2.3) & A{:}*,\, b{:}A & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A\, b : A
\end{array}
$$

- Derivation of (2.2):

$$
\dfrac{\dfrac{\dfrac{(2.1)}{\vdash (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a) : (\Pi\alpha{:}*.\ \alpha \to \alpha)} \quad \dfrac{}{\vdash * : \square}\,[\mathtt{AXIOM}]}{A{:}* \vdash (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a) : (\Pi\alpha{:}*.\ \alpha \to \alpha)}\,[\mathtt{WEAK}] \quad \dfrac{\dfrac{}{\vdash * : \square}\,[\mathtt{AXIOM}]}{A{:}* \vdash A : *}\,[\mathtt{START}]}{A{:}* \vdash (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A : A \to A}\,[\mathtt{APP}]
$$

The λ-cube: examples

- In $\lambda 2$ the following can be derived:

$$
\begin{array}{lrcl}
(2.1) & & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a) : (\Pi\alpha{:}*.\ \alpha \to \alpha) \\
(2.2) & A{:}* & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A : A \to A \\
(2.3) & A{:}*,\, b{:}A & \vdash & (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A\, b : A
\end{array}
$$

- Derivation of (2.3):

$$
\dfrac{
  \dfrac{
    \dfrac{(2.2)}{A{:}* \vdash (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A : A \to A}
    \quad
    \dfrac{\dfrac{}{\vdash * : \Box}\,[\texttt{AXIOM}]}{A{:}* \vdash A : *}\,[\texttt{START}]
  }{A{:}*,\, b{:}A \vdash (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A : A \to A}\,[\texttt{WEAK}]
  \quad
  \dfrac{\dfrac{\dfrac{}{\vdash * : \Box}\,[\texttt{AXIOM}]}{A{:}* \vdash A : *}\,[\texttt{START}]}{A{:}*,\, b{:}A \vdash b : A}\,[\texttt{START}]
}{A{:}*,\, b{:}A \vdash (\lambda\alpha{:}*.\ \lambda a{:}\alpha.\ a)\, A\, b : A}\,[\texttt{APP}]
$$

The λ-cube: examples

- In $\lambda\omega$ the following can be derived:

$$
\begin{array}{llcl}
(3.1) & & \vdash & * \to * : \Box \\
(3.2) & \alpha{:}*,\ \beta{:}* & \vdash & \alpha\&\beta : * \\
(3.3) & & \vdash & (\lambda\alpha{:}*.\,\lambda\beta{:}*.\,\alpha\&\beta) : * \to * \to *
\end{array}
$$

where $\alpha\&\beta \equiv \Pi\gamma{:}*.\,(\alpha \to \beta \to \gamma) \to \gamma$.

The λ-cube: examples

- In $\lambda\omega$ the following can be derived:

$$
\begin{array}{lrcl}
(3.1) & & \vdash & * \to * : \Box \\
(3.2) & \alpha{:}*,\ \beta{:}* & \vdash & \alpha \& \beta : * \\
(3.3) & & \vdash & (\lambda\alpha{:}*.\, \lambda\beta{:}*.\, \alpha \& \beta) : * \to * \to *
\end{array}
$$

where $\alpha \& \beta \equiv \Pi\gamma{:}*.\,(\alpha \to \beta \to \gamma) \to \gamma$.

- Derivation of (3.1):

$$
\dfrac{\dfrac{}{\vdash * : \Box}\,[\texttt{AXIOM}] \qquad \dfrac{\dfrac{}{\vdash * : \Box}\,[\texttt{AXIOM}] \quad \dfrac{}{\vdash * : \Box}\,[\texttt{AXIOM}]}{\_ : * \vdash * : \Box}\,[\texttt{WEAK}]}{\vdash * \to * : \Box}\,[\texttt{PROD}(\Box,\Box)]
$$

The λ-cube: examples

- In $\lambda\omega$ the following can be derived:

$$
\begin{array}{lrll}
(3.1) & & \vdash & * \to * : \Box \\
(3.2) & \alpha{:}*,\ \beta{:}* & \vdash & \alpha\&\beta : * \\
(3.3) & & \vdash & (\lambda\alpha{:}*.\ \lambda\beta{:}*.\ \alpha\&\beta) : * \to * \to *
\end{array}
$$

where $\alpha\&\beta \equiv \Pi\gamma{:}*.\,(\alpha \to \beta \to \gamma) \to \gamma$.

- Derivation of (3.2):

$$
\dfrac{
  \dfrac{\vdots}{\alpha{:}*,\ \beta{:}* \vdash * : \Box}
  \qquad
  \dfrac{
    \dfrac{
      \dfrac{\vdots}{\Gamma \vdash \alpha : *}
      \quad
      \dfrac{
        \dfrac{\vdots}{\Gamma_2 \vdash \beta : *}
        \quad
        \dfrac{\vdots}{\Gamma_3 \vdash \gamma : *}
      }{\Gamma_2 \vdash \beta \to \gamma : *}\,[\text{PROD}(*,*)]
    }{\Gamma \vdash \alpha \to \beta \to \gamma : *}\,[\text{PROD}(*,*)]
    \quad
    \dfrac{\vdots}{\Gamma_1 \vdash \gamma : *}
  }{\Gamma \vdash (\alpha \to \beta \to \gamma) \to \gamma : *}\,[\text{PROD}(*,*)]
}{\alpha{:}*,\ \beta{:}* \vdash \alpha\&\beta : *}\,[\text{PROD}(\Box,*)]
$$

where $\Gamma = \alpha{:}*,\ \beta{:}*,\ \gamma{:}*$ and $\Gamma_i = \Gamma,\ \_ : \ldots$.

The λ-cube: examples

- In $\lambda\omega$ the following can be derived:

$$
\begin{array}{lrcl}
(3.1) & & \vdash & * \to * : \square \\
(3.2) & \alpha{:}*,\ \beta{:}* & \vdash & \alpha \& \beta : * \\
(3.3) & & \vdash & (\lambda\alpha{:}*.\ \lambda\beta{:}*.\ \alpha\&\beta) : * \to * \to *
\end{array}
$$

where $\alpha\&\beta \equiv \Pi\gamma{:}*.\,(\alpha \to \beta \to \gamma) \to \gamma$.

- Derivation of (3.3):

$$
\dfrac{
\dfrac{
\dfrac{(3.2)}{\alpha{:}*,\ \beta{:}* \vdash \alpha\&\beta : *}
\quad
\dfrac{(3.1)}{* \to * : \square}
}{\alpha{:}* \vdash (\lambda\beta{:}*.\ \alpha\&\beta) : * \to *}\ [\texttt{ABS}]
\quad
\dfrac{
\dfrac{}{\vdash * : \square}
\quad
\dfrac{
\dfrac{(3.1)}{\vdash * \to * : \square}
\quad
\dfrac{}{\vdash * : \square}
}{\_ : * \vdash * \to * : \square}\ [\texttt{WEAK}]
}{\vdash * \to * \to * : \square}\ [\texttt{PROD}(\square,\square)]
}{\vdash (\lambda\alpha{:}*.\ \lambda\beta{:}*.\ \alpha\&\beta) : * \to * \to *}\ [\texttt{ABS}]
$$

Pure type systems

- Expressions:

$$
\begin{array}{rcll}
e & ::= & x & \text{variable} \\
 & | & c & \text{constant} \\
 & | & e_1\, e_2 & \text{application} \\
 & | & \lambda x : e_1 .\, e_2 & \text{abstraction} \\
 & | & \Pi x : e_1 .\, e_2 & \text{product}
\end{array}
$$

- The specification of a pure type system consists of a triple $\mathsf{S} = (\mathcal{S}, \mathcal{A}, \mathcal{R})$ where
  - $\mathcal{S}$ is a subset of constants C, called the sorts;
  - $\mathcal{A}$ is a set of axioms of the form $c : s$, where $c \in C$ and $s \in \mathcal{S}$;
  - $\mathcal{R}$ is a set of rules of the form (s_1, s_2, s_3), where $s_i \in \mathcal{S}$.
- Notation: $(s_1, s_2) = (s_1, s_2, s_2)$.

Pure type systems

- Typing rules:

$$\frac{(c : s) \in \mathcal{A}}{\vdash c : s}\ [\mathtt{AXIOM}] \qquad \frac{\Gamma \vdash e_1 : (\Pi x{:}A.B) \quad \Gamma \vdash e_2 : A}{\Gamma \vdash e_1\, e_2 : B[x \mapsto e_2]}\ [\mathtt{APP}]$$

$$\frac{\Gamma \vdash A : s}{\Gamma,\, x{:}A \vdash x : A}\ [\mathtt{START}] \qquad \frac{\Gamma,\, x{:}A \vdash e : B \quad \Gamma \vdash (\Pi x{:}A.B) : s}{\Gamma \vdash (\lambda x{:}A.\, e) : (\Pi x{:}A.B)}\ [\mathtt{ABS}]$$

$$\frac{\Gamma \vdash e : A \quad \Gamma \vdash B : s}{\Gamma,\, x{:}B \vdash e : A}\ [\mathtt{WEAK}] \qquad \frac{\Gamma \vdash e : B \quad \Gamma \vdash A : s \quad A =_\beta B}{\Gamma \vdash e : A}\ [\mathtt{CONV}]$$

$$\frac{\Gamma \vdash A : s_1 \quad \Gamma,\, x{:}A \vdash B : s_2}{\Gamma \vdash (\Pi x{:}A.B) : s_3}\ [\mathtt{PROD}]$$

  – Here $s \in \mathcal{S}$ and $(s_1, s_2, s_3) \in \mathcal{R}$.

Properties of pure type systems

- The PTS $\lambda(\mathcal{S}, \mathcal{A}, \mathcal{R})$ is full if $\mathcal{R} = \{(s_1, s_2) \mid s_1, s_2 \in \mathcal{S}\}$.
- The PTS $\lambda(\mathcal{S}, \mathcal{A}, \mathcal{R})$ is singly sorted if

$$\begin{array}{lll} 1. & (c : s_1), (c : s_2) \in \mathcal{A} & \Longrightarrow \quad s_1 \equiv s_2 \\ 2. & (s_1, s_2, s_3), (s_1, s_2, s_3') \in \mathcal{R} & \Longrightarrow \quad s_3 \equiv s_3' \end{array}$$

- Subject Reduction: For any PTS,

$$\Gamma \vdash A : B \ \wedge \ A \twoheadrightarrow A' \quad \Longrightarrow \quad \Gamma \vdash A' : B$$

- Unicity of types: For any singly sorted PTS,

$$\Gamma \vdash A : B \ \wedge \ \Gamma \vdash A : B' \quad \Longrightarrow \quad B =_\beta B'$$

Pure type systems: examples

- $\lambda 2$ is the PTS determined by:

$$\lambda 2 \quad \boxed{\begin{array}{ll} \mathcal{S} & *,\ \Box \\ \mathcal{A} & * : \Box \\ \mathcal{R} & (*,*),\ (\Box,*) \end{array}}$$

- λC is the full PTS determined by:

$$\lambda C \quad \boxed{\begin{array}{ll} \mathcal{S} & *,\ \Box \\ \mathcal{A} & * : \Box \\ \mathcal{R} & (*,*),\ (\Box,*),\ (*,\Box),\ (\Box,\Box) \end{array}}$$

Pure type systems: examples

- $\lambda\!\rightarrow$ is the PTS determined by:

$$\lambda\!\rightarrow \quad \boxed{\begin{array}{ll} \mathcal{S} & *,\ \Box \\ \mathcal{A} & * : \Box \\ \mathcal{R} & (*, *) \end{array}}$$

- λ^{τ} is the PTS determined by:

$$\lambda^{\tau} \quad \boxed{\begin{array}{ll} \mathcal{S} & * \\ \mathcal{A} & 0 : * \\ \mathcal{R} & (*, *) \end{array}}$$

Pure type systems: examples

- λHOL is the PTS determined by:

$$\lambda\mathrm{HOL}\quad \begin{array}{|ll|}\hline \mathcal{S} & *,\ \Box,\ \Delta \\ \mathcal{A} & * : \Box,\ \Box : \Delta \\ \mathcal{R} & (*,*),\ (\Box,*),\ (\Box,\Box) \\ \hline \end{array}$$

- $\lambda *$ is the PTS determined by:

$$\lambda *\quad \begin{array}{|ll|}\hline \mathcal{S} & * \\ \mathcal{A} & * : * \\ \mathcal{R} & (*,*) \\ \hline \end{array}$$