Computational Logic

Computational Logic

Vincent van Oostrom
Course/slides by Aart Middeldorp

Department of Computer Science
University of Innsbruck

SS 2020

Outline

Tableaux and Hilbert Systems are proof calculi, just as Natural Deduction (seen in Ba logic course), and resolution (also in the book but not part of this course). There are many proof calculi, each describing formally how proofs are structured and what operations are permitted on them. Whereas before we have focused on the meta-theoretical aspects (soundness, completeness, interpolation etc.) of the calculi, this week and next week we will focus more on the structural and representational aspects of proofs themselves, in particular for Hilbert Systems (this week) and Natural Deduction (next week).
In mathematics proofs are stated at an informal level. When implementing proofs appropriate formal representations and operations on these representations must be chosen. For instance, tableaux could be formalised as trees whose nodes are formulas and whose leaves can be expanded, and Hilbert System proofs can be represented as lists whose elements (its lines) are either instances of Axiom Schemes or inferences of (2) previous lines (by Modus Ponens) and we may add such lines at the end of the list. Today we will introduce combinatory logic as a term representation of the proofs of propositional logic, more precisely, of proofs in Hilbert Systems restricted to only Axiom Schemes 1 and 2 and where implication is the only connective.

Combinatory logic (CL) terms are constructed from two constants, K and S, and one operation application which is left implicit (denoted by juxtaposition). For instance, $(SK)K$ is a CL-term comprising two applications. Representing Hilbert System proofs as CL-terms goes in two steps:

- From lists to trees (the correspondence between $\vdash_{\mathsf{ph}}$ and $\vdash_{\mathsf{H}}$ on slide 15): Hilbert System proofs were represented above as lists where lines may refer to (2) previous lines (in case of Modus Ponens). Viewing elements as nodes, this turns the list into a (directed acyclic) graph, and if lines were not reused even into a tree. Observe that by copying lines reuse can always be avoided (at the expense of making the proof longer) so that Hilbert System proof lists can always be represented as Hilbert System proof trees.
- From trees to terms (slides 19–26): Hilbert Systems proof trees have nodes of two types: leaves that are instances of Axioms Schemes and internal Modus-Ponens-nodes with two edges to other nodes. Observe that we may assume the edges of the latter to be in a fixed order (since $X \supset Y$ is larger, as formula, than X). That is, we may assume the tree to be an ordered binary tree. From such a tree a CL-term is obtained by representing Axiom Schemes 1 and 2 (when restricted to that fragment) by constants K and S and Modus Ponens by a binary function symbol called application.

For instance, the inference that the term SKK of (simply typed) combinatory logic is of type $\alpha \to \alpha$ as inferred on slide 24, is a term representation of the proof in Hilbert Systems on page 80 of Fitting's book that $P \supset P$. Each application (denoted by juxtaposition) in the former corresponds to a usage of modus ponens in the latter, and each K and S in the former correspond to usage of Axiom Schemes 1 respectively 2 in the latter. (Both the CL-term and the HS-proof have size 5: the former comprises 2 applications, 2 Ks and 1 S, whereas the latter comprises 2 modus ponens, 1 instance of Axiom Scheme 1 and 2 instances of Axiom Scheme 2.

That is, we can view proofs as terms. This correspondence is half of the Curry–Howard isomorphism, the other half being propositions as types, e.g. that the proposition $X \supset Y$ can be viewed as the type $X \to Y$ (of functions from X to Y). Curry–Howard expresses a correspondence between the proof system for propositional logic and type inference systems. For instance, Modus Ponens expressing that from $X \supset Y$ and X we may infer Y can be viewed as (in functional programming) inferring that applying a function of type $X \to Y$ to an argument of type X yields a result of type Y. Weak reduction $\to_w$ on CL-terms is similar to cut-elimination on proofs in that it 'eliminates cuts' (but for K, S) possibly at the expense of lengthening terms/proofs.

As it turns out, restricting to Axiom Schemes 1 and 2 makes the proof calculus incomplete for propositional logic, even when restricted to just implicational formulas. That is, there are propositional tautologies that are not provable (in the restricted system), with Peirce's law $((P \supset Q) \supset P) \supset P$ being an example. Looking at it from the other end, one may ask whether there is a semantic characterisation of the formulas provable in the restricted system, i.e. a logic for which the restricted inference system is complete. Such a logic does indeed exist and is known as intuitionistic logic. Trying to prove Peirce's law in the unrestricted system, one notices that the law of the excluded middle $X \vee \neg X$ (LEM; or any one of its equivalent formulations such as double-negation-elimination) is used. Intuitionistic logic arises by removing/not accepting LEM.

Instead of the usual truth-table semantics of classical propositional logic, intuitionistic propositional logic has (must have!) different semantics. We present Kripke semantics (slides 8–16). Whereas truth-table semantics can be thought of as based on giving truth-values to all propositional letters in one state, Kripke semantics allows truth-values to evolve (as captured by the order $\leqslant$ on states $\mathcal{C}$), e.g. although P is not known in this state it may evolve to become true in the next state (in particular the interpretation of $\supset$ on slide 10 is based on this). We show the Hilbert System restricted to Axiom Schemes 1 and 2 is both sound and complete with respect to Kripke semantics.

Outline

Syntax

- basic connectives $\supset$ $\wedge$ $\vee$ $\bot$
- derived connectives
  - $\neg\varphi$ abbreviates $\varphi \supset \bot$
  - $\top$ abbreviates $\bot \supset \bot$
  - $\varphi \equiv \psi$ abbreviates $(\varphi \supset \psi) \wedge (\psi \supset \varphi)$
- implicational fragment contains only $\supset$

Formal Semantics

- Heyting algebras
- Kripke models

Definition

Kripke model is triple $\mathcal{C} = \langle C, \leqslant, \Vdash \rangle$ with

- nonempty set C of states
- partial order $\leqslant$ on C
- binary relation $\Vdash$ between elements of C and propositional letters

such that $c' \Vdash p$ whenever $c \Vdash p$ and $c \leqslant c'$

Definition

Kripke model $\mathcal{C} = \langle C, \leqslant, \Vdash \rangle$, $c \in C$

- $c \Vdash \varphi \wedge \psi$ if and only if $c \Vdash \varphi$ and $c \Vdash \psi$
- $c \Vdash \varphi \vee \psi$ if and only if $c \Vdash \varphi$ or $c \Vdash \psi$
- $c \Vdash \varphi \supset \psi$ if and only if $c' \Vdash \psi$ for all $c' \geqslant c$ with $c' \Vdash \varphi$
- $c \nVdash \bot$

Terminology

c forces p if $c \Vdash p$

Example

Kripke model $\mathcal{C} = \langle C, \leqslant, \Vdash \rangle$ with $C = \{a, b, c\}$, $a \leqslant b$, $a \leqslant c$, $b \Vdash p$, $c \Vdash q$

- $a \Vdash (p \supset q) \supset q$
- $a \Vdash \neg\neg(p \vee q)$
- $a \nVdash p \vee \neg p$

Definition

Kripke model $\mathcal{C} = \langle C, \leqslant, \Vdash \rangle$, $c \in C$

- $c \Vdash \Gamma$ if $c \Vdash \varphi$ for all $\varphi \in \Gamma$
- $\mathcal{C} \Vdash \varphi$ if $c \Vdash \varphi$ for all $c \in C$

Definition

$\Gamma \Vdash \varphi$ if $c \Vdash \varphi$ whenever $c \Vdash \Gamma$ for all Kripke models $\mathcal{C} = \langle C, \leqslant, \Vdash \rangle$ and $c \in C$

Lemma (Monotonicity)

if $c \leqslant c'$ and $c \Vdash \varphi$ then $c' \Vdash \varphi$

Lemma

if $\Vdash \varphi \vee \psi$ then $\Vdash \varphi$ or $\Vdash \psi$

Theorem

Hilbert system with Modus Ponens and Axiom Schemes 1 and 2 is sound and complete with respect to Kripke models for implicational fragment:

$$\Gamma \vdash_{ph} \varphi \iff \Gamma \Vdash \varphi$$

Proof (⇒)

suppose $\Gamma \vdash_{ph} \varphi$

we prove $\Gamma \Vdash \varphi$ by induction on length of derivation of $\Gamma \vdash_{ph} \varphi$:

- $\varphi \in \Gamma$

  $\Gamma \Vdash \varphi$ holds trivially

- $\varphi = (\psi_1 \supset (\psi_2 \supset \psi_1))$

  $\Vdash \varphi$ by definition of $\Vdash$ and thus also $\Gamma \Vdash \varphi$

- $\varphi = ((\psi_1 \supset (\psi_2 \supset \psi_3)) \supset ((\psi_1 \supset \psi_2) \supset (\psi_1 \supset \psi_3)))$

  $\Vdash \varphi$ by definition of $\Vdash$ and thus also $\Gamma \Vdash \varphi$

- φ is obtained by Modus Ponens

  $\Gamma \vdash \psi$ and $\Gamma \vdash \psi \supset \varphi$ are shorter derivations

  $\Gamma \Vdash \psi$ and $\Gamma \Vdash \psi \supset \varphi$ by induction hypothesis

  $\Gamma \Vdash \varphi$ by definition of $\Vdash$

Proof (⇐)

suppose $\Gamma \vdash_{ph} \varphi$ does not hold

define Kripke model $\mathcal{C} = \langle C, \subseteq, \Vdash \rangle$ with

- $C = \{\Delta \mid \Gamma \subseteq \Delta \text{ and } \Delta = \{\psi \mid \Delta \vdash_{ph} \psi\}\}$
- $\Delta \Vdash p$ if $p \in \Delta$ for propositional letters p

claim: $\Delta \Vdash \psi \iff \psi \in \Delta$ for all $\Delta \in C$ and implicational formulas ψ

proof of claim (induction on ψ): consider $\psi = (\psi_1 \supset \psi_2)$

$\Rightarrow$ let $\Delta \Vdash \psi$ and define $\Delta' = \{\chi \mid \Delta, \psi_1 \vdash_{ph} \chi\}$

$\psi_1 \in \Delta' \in C$ and thus $\Delta' \Vdash \psi_1$ by induction hypothesis

$\Delta' \Vdash \psi_2$ because $\Delta \subseteq \Delta'$ and thus $\psi_2 \in \Delta'$ by induction hypothesis

$\Delta, \psi_1 \vdash_{ph} \psi_2$

$\Delta \vdash_{ph} \psi$ by deduction theorem

Proof (⇐)

suppose $\Gamma \vdash_{ph} \varphi$ does not hold

define Kripke model $\mathcal{C} = \langle C, \subseteq, \Vdash \rangle$ with

- $C = \{\Delta \mid \Gamma \subseteq \Delta \text{ and } \Delta = \{\psi \mid \Delta \vdash_{ph} \psi\}\}$
- $\Delta \Vdash p$ if $p \in \Delta$ for propositional letters p

claim: $\Delta \Vdash \psi \iff \psi \in \Delta$ for all $\Delta \in C$ and implicational formulas ψ

proof of claim: consider $\psi = (\psi_1 \supset \psi_2)$

$\Leftarrow$ let $\psi \in \Delta$ and consider state $\Delta' \supseteq \Delta$ with $\Delta' \Vdash \psi_1$

$\psi_1 \in \Delta'$ by induction hypothesis and thus $\Delta' \vdash_{ph} \psi_1$

$\Delta' \vdash_{ph} \psi$ because $\Delta \subseteq \Delta'$

$\Delta' \vdash_{ph} \psi_2$ by Modus Ponens

$\Delta' \Vdash \psi_2$ by induction hypothesis

Proof (⇐)

suppose $\Gamma \vdash_{ph} \varphi$ does not hold

define Kripke model $\mathcal{C} = \langle C, \subseteq, \Vdash \rangle$ with

- $C = \{\Delta \mid \Gamma \subseteq \Delta \text{ and } \Delta = \{\psi \mid \Delta \vdash_{ph} \psi\}\}$
- $\Delta \Vdash p$ if $p \in \Delta$ for propositional letters p

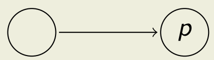
claim: $\Delta \Vdash \psi \iff \psi \in \Delta$ for all $\Delta \in C$ and implicational formulas ψ

define $\Delta = \{\psi \mid \Gamma \vdash_{ph} \psi\}$

$\Delta \in C$ and $\Delta \Vdash \psi$ for all $\psi \in \Gamma$ and $\Delta \nVdash \varphi$

$\Gamma \nVdash \varphi$ by definition of $\Vdash$

Example (Peirce's Law)

$\nVdash ((p \supset q) \supset p) \supset p$ because of Kripke model 

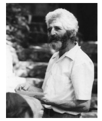
Definition (Hilbert Systems, Tree Variant)

- Assumption $\Gamma, \varphi \vdash \varphi$
- Axiom Scheme 1 $\Gamma \vdash \varphi \supset (\psi \supset \varphi)$
- Axiom Scheme 2 $\Gamma \vdash (\varphi \supset (\psi \supset \chi)) \supset ((\varphi \supset \psi) \supset (\varphi \supset \chi))$
- Modus Ponens $\dfrac{\Gamma \vdash \varphi \supset \psi \qquad \Gamma \vdash \varphi}{\Gamma \vdash \psi}$

$\Gamma \vdash_H \varphi$ if $\Gamma \vdash \varphi$ is derivable

Lemma

$\Gamma \vdash_{ph} \varphi \quad \Longleftrightarrow \quad \Gamma \vdash_H \varphi$

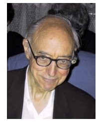


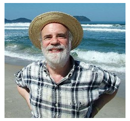
William Craig
(1918 – 2016)

Jacques Herbrand
(1908 – 1931)

David Hilbert
(1862 – 1943)

Jaakko Hintikka
(1929 – 2015)





Saul Kripke
(1940 –)

Leopold Löwenheim
(1878 – 1957)

Thoralf Skolem
(1887 – 1963)

Outline

Definition

set $\mathcal{C}$ of (combinatory) terms is built from

- variables $x, y, z, \ldots$
- constants K S
- application (MN) for combinatory terms M and N

Notational Convention

left association to reduce number of parentheses

Definition

(weak) reduction is smallest relation $\to_w$ on terms such that

$$\dfrac{}{\mathsf{K}\,M\,N \to_w M} \qquad \dfrac{}{\mathsf{S}\,M\,N\,P \to_w M\,P\,(N\,P)} \qquad \dfrac{M \to_w N}{M\,P \to_w N\,P} \qquad \dfrac{M \to_w N}{P\,M \to_w P\,N}$$

for all terms M, N, P

Definitions

- $\rightarrow_w^*$ is transitive and reflexive closure of $\rightarrow_w$
- $\mathsf{I} = \mathsf{SKK}$ $\mathsf{W} = \mathsf{SS(KI)}$ $\mathsf{B} = \mathsf{S(KS)K}$ $\mathsf{C} = \mathsf{S(BBS)(KK)}$

Lemma

$\mathsf{I}x \rightarrow_w^* x$ $\mathsf{W}xy \rightarrow_w^* xyy$ $\mathsf{B}xyz \rightarrow_w^* x(yz)$ $\mathsf{C}xyz \rightarrow_w^* xzy$

Proof

$$\mathsf{I}x \rightarrow_w \mathsf{K}x(\mathsf{K}x) \rightarrow_w x$$

$$\mathsf{W}xy \rightarrow_w \mathsf{S}x(\mathsf{KI}x)y \rightarrow_w xy(\mathsf{KI}xy) \rightarrow_w xy(\mathsf{I}y) \rightarrow_w^* xyy$$

$$\mathsf{B}xyz \rightarrow_w \mathsf{KS}x(\mathsf{K}x)yz \rightarrow_w \mathsf{S}(\mathsf{K}x)yz \rightarrow_w \mathsf{K}xz(yz) \rightarrow_w x(yz)$$

$$\mathsf{C}xyz \rightarrow_w \mathsf{BBS}x(\mathsf{KK}x)yz \rightarrow_w \mathsf{BBS}x\mathsf{K}yz \rightarrow_w^* \mathsf{B}(\mathsf{S}x)\mathsf{K}yz$$
$$\rightarrow_w^* \mathsf{S}x(\mathsf{K}y)z \rightarrow_w xz(\mathsf{K}yz) \rightarrow_w xzy$$

Definitions

- normal form is term M such that $M \rightarrow_w N$ for no term N
- $=_w$ is transitive, reflexive, and symmetric closure of $\rightarrow_w$
- term M is normalizing if $M \rightarrow_w^* N$ for some normal form N
- infinite reduction is sequence $(M_i)_{i \geqslant 0}$ such that $M_i \rightarrow_w M_{i+1}$ for all $i \geqslant 0$
- term M is strongly normalizing if there are no infinite reductions starting at M

Example

term SII(SII) is not strongly normalizing:

$$\mathsf{SII(SII)} \rightarrow_w \mathsf{I(SII)(I(SII))} \rightarrow_w^* \mathsf{SII(I(SII))} \rightarrow_w^* \mathsf{SII(SII)}$$

Theorem (Confluence)

if $M \rightarrow_w^* N_1$ and $M \rightarrow_w^* N_2$ then $N_1 \rightarrow_w^* N_3$ and $N_2 \rightarrow_w^* N_3$ for some term N_3

Definitions

- simple type is implicational propositional formula
- environment is finite set of pairs $\Gamma = \{x_1 : \tau_1, \dots, x_n : \tau_n\}$ with pairwise distinct variables $x_1, \dots, x_n$ and simple types $\tau_1, \dots, \tau_n$
- $\mathrm{dom}(\Gamma) = \{x \mid (x : \tau) \in \Gamma\}$ and $\mathrm{ran}(\Gamma) = \{\tau \mid (x : \tau) \in \Gamma\}$
- judgement $\Gamma \vdash M : \tau$ (term M has type τ in environment Γ) is defined by type assignment rules
  - variable $\Gamma, x : \tau \vdash x : \tau$
  - K $\Gamma \vdash \mathsf{K} : \sigma \to \tau \to \sigma$
  - S $\Gamma \vdash \mathsf{S} : (\sigma \to \tau \to \rho) \to (\sigma \to \tau) \to \sigma \to \rho$
  - application $\dfrac{\Gamma \vdash M : \sigma \to \tau \qquad \Gamma \vdash N : \sigma}{\Gamma \vdash (MN) : \tau}$

Examples

- $\vdash \mathsf{SKK} : \alpha \to \alpha$ for all simple types α

$$\dfrac{\dfrac{\mathsf{S} : (\alpha \to (\alpha \to \alpha) \to \alpha) \to (\alpha \to \alpha \to \alpha) \to \alpha \to \alpha \qquad \mathsf{K} : \alpha \to (\alpha \to \alpha) \to \alpha}{\mathsf{SK} : (\alpha \to \alpha \to \alpha) \to \alpha \to \alpha} \qquad \mathsf{K} : \alpha \to \alpha \to \alpha}{\mathsf{SKK} : \alpha \to \alpha}$$

- $\vdash \mathsf{B} : (\alpha \to \beta) \to (\gamma \to \alpha) \to \gamma \to \beta$

$$\dfrac{\dfrac{\mathsf{S} : (\mu \to \nu \to \pi) \to (\mu \to \nu) \to (\mu \to \pi) \qquad \dfrac{\mathsf{K} : (\theta \to \mu \to \theta) \qquad \mathsf{S} : \theta}{\mathsf{KS} : \mu \to \theta}}{\mathsf{S(KS)} : (\mu \to \nu) \to \mu \to \pi} \qquad \mathsf{K} : (\mu \to \nu)}{\mathsf{S(KS)K} : \mu \to \pi}$$

with $\theta = (\gamma \to \alpha \to \beta) \to (\gamma \to \alpha) \to \gamma \to \beta$, $\mu = \alpha \to \beta$, $\nu = \gamma \to \alpha \to \beta$, $\pi = (\gamma \to \alpha) \to \gamma \to \beta$

Definitions

- set FV(M) of (free) variables of term M:

$$\mathrm{FV}(M) = \begin{cases} \{M\} & \text{if } M \text{ is variable} \\ \varnothing & \text{if } M \in \{\mathsf{K}, \mathsf{S}\} \\ \mathrm{FV}(M_1) \cup \mathrm{FV}(M_2) & \text{if } M = M_1 M_2 \end{cases}$$

- term M is typable if $\Gamma \vdash M : \tau$ for some environment Γ with $\mathrm{dom}(\Gamma) = \mathrm{FV}(M)$ and simple type τ

Lemma (Subject Reduction)

if $\Gamma \vdash M : \tau$ *and* $M \to_w^* N$ *then* $\Gamma \vdash N : \tau$

Theorem (Strong Normalization)

typable terms are strongly normalizing

Decision Problems

- type checking

  instance: term M, environment Γ, simple type τ

  question: $\Gamma \vdash M : \tau$?

- type inference

  instance: term M

  question: $\Gamma \vdash M : \tau$ for some environment Γ and simple type τ ?

- type inhabitation

  instance: type τ, environment Γ

  question: $\Gamma \vdash M : \tau$ for some term M ?

Theorem

type checking, inference, and inhabitation are decidable problems

Outline

type assignment	Hilbert system
$\Gamma, x : \tau \vdash x : \tau$	$\Gamma, \varphi \vdash \varphi$
$\Gamma \vdash \mathsf{K} : \sigma \to \tau \to \sigma$	$\Gamma \vdash \varphi \supset (\psi \supset \varphi)$
$\Gamma \vdash \mathsf{S} : (\sigma \to \tau \to \rho) \to (\sigma \to \tau) \to \sigma \to \rho$	$\Gamma \vdash (\varphi \supset (\psi \supset \chi)) \supset ((\varphi \supset \psi) \supset (\varphi \supset \chi))$
$\dfrac{\Gamma \vdash M : \sigma \to \tau \qquad \Gamma \vdash N : \sigma}{\Gamma \vdash (MN) : \tau}$	$\dfrac{\Gamma \vdash \varphi \supset \psi \qquad \Gamma \vdash \varphi}{\Gamma \vdash \psi}$

$\to$ and $\supset$ are identified

Theorem (Curry–Howard)

1. *if* $\Gamma \vdash M : \tau$ *then* $\mathrm{ran}(\Gamma) \vdash_H \tau$
2. *if* $\Gamma \vdash_H \varphi$ *then* $\Delta \vdash M : \varphi$ *for some* M *and* Δ *with* $\mathrm{ran}(\Delta) = \Gamma$

Theorem (Curry–Howard)

1. *if* $\Gamma \vdash M : \tau$ *then* $\mathrm{ran}(\Gamma) \vdash_H \tau$

Proof

induction on derivation of judgement $\Gamma \vdash M : \tau$

- $M = x$ and $\Gamma = \Gamma', x : \tau$

  $\mathrm{ran}(\Gamma) = \mathrm{ran}(\Gamma'), \tau$ and thus $\mathrm{ran}(\Gamma) \vdash_H \tau$ by Assumption
- $M = \mathsf{K}$ and $\tau = (\sigma \to \rho \to \sigma)$

  $\mathrm{ran}(\Gamma) \vdash_H \tau$ by Axiom Scheme 1
- $M = \mathsf{S}$ and $\tau = ((\sigma \to \rho \to \chi) \to (\sigma \to \rho) \to \sigma \to \chi$

  $\mathrm{ran}(\Gamma) \vdash_H \tau$ by Axiom Scheme 2
- $M = (NP)$ and $\Gamma \vdash N : \sigma \to \tau$ and $\Gamma \vdash P : \sigma$

  $\mathrm{ran}(\Gamma) \vdash_H \sigma \to \tau$ and $\mathrm{ran}(\Gamma) \vdash_H \sigma$ by induction hypothesis

  $\mathrm{ran}(\Gamma) \vdash_H \tau$ by Modus Ponens

Theorem (Curry–Howard)

2. *if* $\Gamma \vdash_H \varphi$ *then* $\Delta \vdash M : \varphi$ *for some* M *and* Δ *with* $\mathrm{ran}(\Delta) = \Gamma$

Proof

induction on derivation of $\Gamma \vdash_H \varphi$

interesting case: φ is obtained by Modus Ponens

$\Gamma \vdash_H \psi \to \varphi$ and $\Gamma \vdash_H \psi$

induction hypothesis: $\Delta_1 \vdash M_1 : \psi \to \varphi$ and $\Delta_2 \vdash M_2 : \psi$
for some M_1, Δ_1, M_2, Δ_2 with $\mathrm{ran}(\Delta_1) = \mathrm{ran}(\Delta_2) = \Gamma$

suppose $\Gamma = \{\phi_1, \dots, \phi_n\}$

$\Delta_1 = \{x_1 : \phi_1, \dots, x_n : \phi_n\}$
$\Delta_2 = \{y_1 : \phi_1, \dots, y_n : \phi_n\}$

let M_2' be obtained from M_2 by replacing every y_i with x_i

$\Delta_1 \vdash M_2' : \psi$ and thus $\Delta_1 \vdash (M_1 M_2') : \varphi$

Corollary

if $\Gamma, x : \sigma \vdash M : \tau$ *then* $\Gamma \vdash N : \sigma \to \tau$ *for some term* N

Proof

Curry–Howard in combination with deduction theorem

Remark

term N can be computed from M and x by bracket abstraction

Definition (Bracket Abstraction)

term $[x]M$ is defined for all terms M and variables x:

$$[x]M = \begin{cases} \mathsf{I} & \text{if } M = x \\ \mathsf{K}\,M & \text{if } x \notin \mathrm{FV}(M) \\ \mathsf{S}\,([x]M_1)\,([x]M_2) & \text{if } M = M_1 M_2 \text{ and } x \in \mathrm{FV}(M) \end{cases}$$

Definition (Bracket Abstraction)

term $[x]M$ is defined for all terms M and variables x:

$$[x]M = \begin{cases} \mathsf{I} & \text{if } M = x \\ \mathsf{K}\,M & \text{if } x \notin \mathrm{FV}(M) \\ \mathsf{S}\,([x]M_1)\,([x]M_2) & \text{if } M = M_1 M_2 \text{ and } x \in \mathrm{FV}(M) \end{cases}$$

Example

$$\begin{aligned}
[x][y][z](xzy) &= [x][y](\mathsf{S}([z](xz))([z]y)) = [x][y](\mathsf{S}(\mathsf{S}([z]x)([z]z))(\mathsf{K}y)) \\
&= [x][y](\mathsf{S}(\mathsf{S}(\mathsf{K}x)\mathsf{I})(\mathsf{K}y)) = [x](\mathsf{S}([y](\mathsf{S}(\mathsf{S}(\mathsf{K}x)\mathsf{I})))([y](\mathsf{K}y))) \\
&= [x](\mathsf{S}(\mathsf{K}(\mathsf{S}(\mathsf{S}(\mathsf{K}x)\mathsf{I})))(\mathsf{S}([y]\mathsf{K})([y]y))) \\
&= [x](\mathsf{S}(\mathsf{K}(\mathsf{S}(\mathsf{S}(\mathsf{K}x)\mathsf{I})))(\mathsf{S}(\mathsf{K}\mathsf{K})\mathsf{I})) \\
&= \cdots \\
&= \mathsf{S}(\mathsf{S}(\mathsf{K}\mathsf{S})(\mathsf{S}(\mathsf{K}\mathsf{K})(\mathsf{S}(\mathsf{K}\mathsf{S})(\mathsf{S}(\mathsf{S}(\mathsf{K}\mathsf{S})(\mathsf{S}(\mathsf{K}\mathsf{K})\mathsf{I}))(\mathsf{K}\mathsf{I})))))(\mathsf{K}(\mathsf{S}(\mathsf{K}\mathsf{K})\mathsf{I}))
\end{aligned}$$

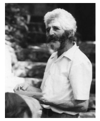
Lemma

$([x]M)N \to_w^* M\{x/N\}$ *for all terms* M *and* N

Lemma

if $\Gamma, x : \sigma \vdash M : \tau$ *then* $\Gamma \vdash [x]M : \sigma \to \tau$

Haskell Curry
(1900 – 1982)

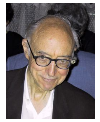


William Craig
(1918 – 2016)

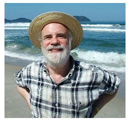
Jacques Herbrand
(1908 – 1931)

David Hilbert
(1862 – 1943)

Jaakko Hintikka
(1929 – 2015)



William Howard
(1926 –)



Saul Kripke
(1940 –)

Leopold Löwenheim
(1878 – 1957)

Thoralf Skolem
(1887 – 1963)

Outline

Earlier Exam

- Exercise 2 of the exam of March 4, 2016.

Intuitionistic Logic

- $\Vdash \varphi \supset \neg\neg\varphi$?
- $\Vdash \neg\neg\varphi \supset \varphi$?
- $\Vdash (\varphi \supset \neg\psi) \supset (\neg\neg\varphi \supset \neg\psi)$?
- Prove that φ is a propositional tautology if and only if $\Vdash \neg\neg\varphi$.

Fitting

- Argue that the Example on slide 32 illustrating the abstraction algorithm gives, via the Curry–Howard correspondence, a solution to Exercise 4.1.1. That is, first show that $x : P \supset (Q \supset R), y : Q, z : P \vdash (xz)y : R$ can be inferred in the type inference system (we identify $\supset$ with $\rightarrow$). Next, show that performing the abstraction algorithm three times to compute $[x][y][z](xz)y$ yields a (closed) term of type $(P \supset (Q \supset R)) \supset (Q \supset (P \subset R))$. Conclude this gives rise to a Hilbert System proof of $(P \supset (Q \supset R)) \supset (Q \supset (P \subset R))$.
- In the solution to Exercise 4.1.1 I had made use of the following extra rule (having priority over the others) for the abstraction algorithm:

  $$[x](Mx) \quad = \quad M \qquad \text{if } x \notin \mathrm{FV}(M)$$

  Show this optimisation to be correct (in the sense of the lemmata on slide 33), and check whether or not I made a mistake in my solution,. Is the extra rule to be preferred or not? Argue why (not).
- Bonus Implement both above versions of the abstraction algorithm and check whether or not slide 32 and the earlier solution to Exercise 4.1.1 are correct.
- Bonus Exercise 4.1.8 (again ...)

Outline

Fitting

- Section 4.1 (revisit from earlier this course, from new C–H-perspective)
- Section 4.2 (revisit from Ba logic course as preparation for next week)
- Section 4.3 (idem)

Additional Literature

- Philip Wadler, Propositions as Types, Communications of the ACM 58(12), pp. 75 – 84, 2015
- Morten Heine Sørensen and Pawel Urzyczyn, Lectures on the Curry–Howard Isomorphism, Studies in Logic and the Foundations of Mathematics, volume 149, Elsevier, 2006 (cached PDF of preliminary version on citeseer)