Back to Basics

Denis Kuperberg, Luigi Santocanale & Alexis Saurin

EPIT 2025 Opening Lecture
May 19th 2025, Aussois

Background on Proof Systems

On Logics and Proof systems

- Statements: A language of formulas
- Consequence relation: $F \models G$ when every model of F is a model of G
- Valid Statements: $\models F$

Design a Proof System that is able:

- To deduce only valid statement (Soundness Property)
- To deduce enough valid statement (if not all)
- Completeness property: when all valid statement are provable.

That allows for tools to analyze the properties of the provability relation

Natural Deduction and Sequent Calculus

- Both introduced by Gentzen in the 30's
- Natural Deduction:
  - Close to mathematical reasoning;
  - A duality between introduction rules (backward reasoning) and elimination rules (forward reasoning);
  - Hypothetical reasoning: can I derive A under an environment of hypotheses Γ.
  - Deductions may take detours when an introduction is followed by an elimination.
- Sequent Calculus:
  - less close to usual reasoning;
  - intro / elim duality is replaced by left / right symmetry;
  - An explicit cut inference
- Both are based on a notion of logical judgements, sequents, which focus on hypothetical reasoning, reasoning under a set of current assumptions.

Natural Deduction : Proof Dynamics

$$\frac{}{\Gamma, A \vdash A}\ Ax \qquad \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ \wedge I \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}\ \Rightarrow I$$

$$\frac{\Gamma \vdash A_1 \wedge A_2}{\Gamma \vdash A_j}\ \wedge Ej\ (j \in \{1, 2\}) \qquad \frac{\Gamma \vdash A \Rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B}\ \Rightarrow E$$

Example

With $\Gamma = A \wedge B \Rightarrow C, A, B$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma \vdash A \wedge B \Rightarrow C}\ Ax \quad \dfrac{\dfrac{}{\Gamma \vdash A}\ Ax \quad \dfrac{}{\Gamma \vdash B}\ Ax}{\Gamma \vdash A \wedge B}\ \wedge I}{A \wedge B \Rightarrow C, A, B \vdash C}\ \Rightarrow E}{A \wedge B \Rightarrow C, A \vdash B \Rightarrow C}\ \Rightarrow I}{A \wedge B \Rightarrow C \vdash A \Rightarrow (B \Rightarrow C)}\ \Rightarrow I}{\vdash (A \wedge B \Rightarrow C) \Rightarrow (A \Rightarrow (B \Rightarrow C))}\ \Rightarrow I$$

Natural Deduction : Proof Dynamics

$$\dfrac{}{\Gamma, A \vdash A}\,Ax \qquad \dfrac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\,\wedge I \qquad \dfrac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}\,\Rightarrow I$$

$$\dfrac{\Gamma \vdash A_1 \wedge A_2}{\Gamma \vdash A_j}\,\wedge Ej\ (j \in \{1,2\}) \qquad \dfrac{\Gamma \vdash A \Rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B}\,\Rightarrow E$$

Example

With $\Gamma = A \wedge B \Rightarrow C, A, B$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma \vdash A \wedge B \Rightarrow C}\,Ax \quad \dfrac{\dfrac{}{\Gamma \vdash A}\,Ax \quad \dfrac{}{\Gamma \vdash B}\,Ax}{\Gamma \vdash A \wedge B}\,\wedge I}{A \wedge B \Rightarrow C, A, B \vdash C}\,\Rightarrow E}{A \wedge B \Rightarrow C, A \vdash B \Rightarrow C}\,\Rightarrow I}{A \wedge B \Rightarrow C \vdash A \Rightarrow (B \Rightarrow C)}\,\Rightarrow I}{\vdash (A \wedge B \Rightarrow C) \Rightarrow (A \Rightarrow (B \Rightarrow C))}\,\Rightarrow I$$

Detours (cuts)

$$\dfrac{\dfrac{\dfrac{\mathcal{D}_A}{\Gamma \vdash A} \quad \dfrac{\mathcal{D}_B}{\Gamma \vdash B}}{\Gamma \vdash A \wedge B}\,\wedge I}{\Gamma \vdash B}\,\wedge E2$$

$$\dfrac{\dfrac{\dfrac{\mathcal{D}_1}{\Gamma, A \vdash B}}{\Gamma \vdash A \Rightarrow B}\,\Rightarrow I \quad \dfrac{\mathcal{D}_2}{\Gamma \vdash A}}{\Gamma \vdash B}\,\Rightarrow E$$

$$\tilde{\mathcal{D}} = \mathcal{D}_1 \left\{ \mathcal{D}_2 / \overline{\Gamma, A, \Delta \vdash A}\ Ax \right\}$$

Natural Deduction : Proof Dynamics

$$\frac{}{\Gamma, A \vdash A}\ Ax \qquad \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ \wedge I \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}\ \Rightarrow I$$

$$\frac{\Gamma \vdash A_1 \wedge A_2}{\Gamma \vdash A_j}\ \wedge Ej\ (j \in \{1, 2\}) \qquad \frac{\Gamma \vdash A \Rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B}\ \Rightarrow E$$

Example

With $\Gamma = A \wedge B \Rightarrow C, A, B$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma \vdash A \wedge B \Rightarrow C}\ Ax \quad \dfrac{\dfrac{}{\Gamma \vdash A}\ Ax \quad \dfrac{}{\Gamma \vdash B}\ Ax}{\Gamma \vdash A \wedge B}\ \wedge I}{A \wedge B \Rightarrow C, A, B \vdash C}\ \Rightarrow E}{A \wedge B \Rightarrow C, A \vdash B \Rightarrow C}\ \Rightarrow I}{A \wedge B \Rightarrow C \vdash A \Rightarrow (B \Rightarrow C)}\ \Rightarrow I}{\vdash (A \wedge B \Rightarrow C) \Rightarrow (A \Rightarrow (B \Rightarrow C))}\ \Rightarrow I$$

Detours (cuts)

$$\dfrac{\dfrac{\dfrac{\mathcal{D}_A}{\Gamma \vdash A} \quad \dfrac{\mathcal{D}_B}{\Gamma \vdash B}}{\Gamma \vdash A \wedge B}\ \wedge I}{\Gamma \vdash B}\ \wedge E2 \quad \longrightarrow \quad \dfrac{\mathcal{D}_B}{\Gamma \vdash B}$$

$$\dfrac{\dfrac{\dfrac{\mathcal{D}_1}{\Gamma, A \vdash B}}{\Gamma \vdash A \Rightarrow B}\ \Rightarrow I \quad \dfrac{\mathcal{D}_2}{\Gamma \vdash A}}{\Gamma \vdash B}\ \Rightarrow E \longrightarrow \dfrac{\tilde{\mathcal{D}}}{\Gamma \vdash B}$$

$$\tilde{\mathcal{D}} = \mathcal{D}_1 \left\{ \mathcal{D}_2 / \dfrac{}{\Gamma, A, \Delta \vdash A}\ Ax \right\}$$

Very good properties: confluence, termination, ...

Towards the Sequent calculus

- Cut becomes an inference rules of the proof system,
- Structural rules serve only to manage formula erasure and duplication,
- Left-right symmetry, which is the counterpart of the intro/elim symmetry in natural deduction; elimination rules becomes left introduction rules,
- A sequent has the form: $H_1, \ldots, H_m \vdash C_1, \ldots, C_n$. with potential restructions.
- A logical judgement meaning that the conjunction of the H_i has as logical consequence the disjunction of the C_j, i.e. :

$$\left(\bigwedge_{1 \leq i \leq m} H_i \right) \Rightarrow \left(\bigvee_{1 \leq j \leq n} C_j \right)$$

(Some of) LK Inference Rules

Group	
Identity	$\dfrac{}{A \vdash A}\ Ax \quad \dfrac{\Gamma_1 \vdash A, \Delta_1 \quad \Gamma_2, A \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}\ Cut$
Structural	$\dfrac{\Gamma_1, B, A, \Gamma_2 \vdash \Delta}{\Gamma_1, A, B, \Gamma_2 \vdash \Delta}\ LEx \quad \dfrac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}\ LW \quad \dfrac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}\ LC$
Logical	$\dfrac{\Gamma, A_j \vdash \Delta}{\Gamma, A_1 \wedge A_2 \vdash \Delta}\ L\wedge j \quad (j \in \{1, 2\}) \quad \dfrac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta}\ R\wedge$
	$\dfrac{\Gamma_1 \vdash A, \Delta_1 \quad \Gamma_2, B \vdash \Delta_2}{\Gamma_1, \Gamma_2, A \Rightarrow B \vdash \Delta_1, \Delta_2}\ L\Rightarrow \quad \dfrac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \Rightarrow B, \Delta}\ R\Rightarrow$

Some formal theorems

- $A \vee B \vdash B \vee A$ Commutativity of disjunction
- $\vdash A \vee \neg A$ *Tertium non datur*
- $\vdash ((A \Rightarrow B) \Rightarrow A) \Rightarrow A$ Peirce's Law
- $\vdash (\neg\neg A) \Rightarrow A$ Double Negation Elimination
- $\vdash \exists x \forall y (P(x) \Rightarrow P(y))$ Drinker Poperty
- $A \vee B \vdash \neg(\neg A \wedge \neg B)$ de Morgan Law
- $\vdash (A \Rightarrow B) \vee (B \Rightarrow A)$
- $\vdash \neg\neg(A \vee \neg A)$ intuitionistic *Tertium non datur*
- $(p \Rightarrow q) \vdash (\neg q \Rightarrow \neg p)$
- $(\neg q \Rightarrow \neg p) \vdash (p \Rightarrow q)$

Prove them!

Gentzen's Cut-Elimination Theorem

Gentzen's Hauptsatz

The cut rule is admissible in the cut-free sequent calculus.

Gentzen provided more than a proof of *admissibility of cut* since it also provides *an explicit procedure to eliminate cuts from a proof*:

⋆ Key case: A cut on $A \wedge B$:

$$\dfrac{\dfrac{\dfrac{\pi_A}{\Gamma \vdash \Delta, A} \quad \dfrac{\pi_B}{\Gamma \vdash \Delta, B}}{\Gamma \vdash \Delta, A \wedge B}\, R\wedge \quad \dfrac{\dfrac{\pi'}{A, \Gamma' \vdash \Delta'}}{A \wedge B, \Gamma' \vdash \Delta'}\, L\wedge_1}{\Gamma, \Gamma' \vdash \Delta, \Delta'}\, Cut$$

is reduced into a cut on A:

$$\dfrac{\dfrac{\pi_A}{\Gamma \vdash \Delta, A} \quad \dfrac{\pi'}{A, \Gamma' \vdash \Delta'}}{\Gamma, \Gamma' \vdash \Delta, \Delta'}\, Cut$$

⋆ Cut commutation (some inferences may commutes below the cut):

$$\dfrac{\dfrac{\dfrac{\pi_B}{\Gamma \vdash \Delta, B, C}}{\Gamma \vdash \Delta, A \vee B, C}\, R\vee_2 \quad \dfrac{\pi'}{C, \Gamma' \vdash \Delta'}}{\Gamma, \Gamma' \vdash \Delta, A \vee B, \Delta'}\, Cut \quad \longmapsto \quad \dfrac{\dfrac{\dfrac{\pi_B}{\Gamma \vdash \Delta, B, C} \quad \dfrac{\pi'}{C, \Gamma' \vdash \Delta'}}{\Gamma \vdash \Delta, B}\, Cut}{\Gamma, \Gamma' \vdash \Delta, A \vee B, \Delta'}\, R\vee_2$$

Consequences of cut-elimination

Sub-formula property, Consistency and Interpolation

Subformula Property

A provable sequent can be derived by using only sub-formulas of the formulas occurring in the conclusion of the sequent.

Reduces search space and sometimes ensures decidability of provability.

Theorem (Consistency of LK)

The empty sequent is not derivable in LK.
In particular, there is no proof of $\vdash \bot$ in LK and one cannot prove both $\vdash A$ and $\vdash \neg A$ for any formula.

Theorem (Craig Interpolation Theorem)

If $\vdash_{LK} A \Rightarrow B$, then there exists a formula I using only atomic propositions that A and B have in common and such that $\vdash_{LK} A \Rightarrow I$ and $\vdash_{LK} I \Rightarrow B$.

Non-Constructivism of LK

Constructivism: a proof provides a method for constructing an object representing the property. (think of a geometric statement asserting the existence of some geometrical object.)

Some proofs are constructive, other are not. Let us consider the following non-constructive proof of the following proposition:

Proposition

There exists two irrational numbers a, b such that a^b is rational.

Proof

Consider $\sqrt{2}$ which is irrational. Either $\sqrt{2}^{\sqrt{2}}$ is rational or it isn't.
In the first case, let $a = b = \sqrt{2}$, the result follows. In the second case, let $a = sqrt2^{\sqrt{2}}$ and $b = \sqrt{2}$ one obtains
$a^b = (\sqrt{2}^{\sqrt{2}})^{\sqrt{2}} = \sqrt{2}^{\sqrt{2}\times\sqrt{2}} = \sqrt{2}^2 = 2.$

$\square$

Non-Constructivism of *LK*

Constructivism: a proof provides a method for constructing an object representing the property. (think of a geometric statement asserting the existence of some geometrical object.)

Some proofs are constructive, other are not. Let us consider the following non-constructive proof of the following proposition:

Proposition

There exists two irrational numbers a, b such that a^b is rational.

Proof

Consider $\sqrt{2}$ which is irrational. Either $\sqrt{2}^{\sqrt{2}}$ is rational or it isn't.
In the first case, let $a = b = \sqrt{2}$, the result follows. In the second case, let $a = sqrt2^{\sqrt{2}}$ and $b = \sqrt{2}$ one obtains
$a^b = (\sqrt{2}^{\sqrt{2}})^{\sqrt{2}} = \sqrt{2}^{\sqrt{2} \times \sqrt{2}} = \sqrt{2}^2 = 2.$

$\square$

Is $\sqrt{2}^{\sqrt{2}}$ rational or not?

LK **is symmetrical but non constructive.**

Brouwer-Heyting-Kolmogorov Interpretation of proofs

- A proof of $A \vee B$ is a pair (i, π) with $i \in \{1; 2\}$ and if $i = 1$ then π proves A, otherwise it proves B;
- A proof of $A \wedge B$ is a pair (π, π') of a proof of A and a proof of B;
- A proof of $A \Rightarrow B$ is a *function* which associates to any proof of A a proof of B (therefore, it is a proof transformation, or a mapping on proofs);
- A proof of $\exists x A$ is a pair (t, π) with t a term and π a proof of $A[t/x]$;
- A proof of $\forall x A$ is a function which associates to any term t a proof of $A[t/x]$;
- A proof of $\neg A$ is a function which associates, to any proof of A, a proof of $F \wedge \neg F$.

$\Longrightarrow$ **Natural Deduction**

LJ Sequent Calculus

Gentzen modelled intuitionistic in sequent calculus not by modifying the logical rules (as in natural deduction) but by imposing constraints on the shape of sequents: $\Gamma \vdash A$ or $\Gamma \vdash$.

Prevents using the right contraction rule. (But it is still available on the left!)

Disjunction and existence property
Thanks to cut-elimination, we have:

Disjunction Property

If $\vdash_{LJ} A \vee B$, then $\vdash_{LJ} A$ or $\vdash_{LJ} B$

Witness Property

If $\vdash_{LJ} \exists x A$, then there exists a term t such that $\vdash_{LJ} A[t/x]$

LJ **is constructive but asymmetrical .**

Intuitionistic Excluded-middle

$A \vee \neg A$ is not provable in LJ but $\neg\neg(A \vee \neg A)$ is:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash A}\ \mathit{Axiom}}{\vdash A, \neg A}\ R\neg}{\vdash A, A \vee \neg A}\ R\vee}{\vdash A \vee \neg A, A \vee \neg A}\ R\vee}{\vdash A \vee \neg A}\ RC$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash A}\ \mathit{Axiom}}{A \vdash A \vee \neg A}\ R\vee}{\neg(A \vee \neg A), A \vdash}\ L\neg}{\neg(A \vee \neg A) \vdash \neg A}\ R\neg}{\neg(A \vee \neg A) \vdash A \vee \neg A}\ R\vee}{\neg(A \vee \neg A), \neg(A \vee \neg A) \vdash}\ L\neg}{\neg(A \vee \neg A) \vdash}\ LC}{\vdash \neg\neg(A \vee \neg A)}\ R\neg$$

LJ is weaker that LK: $\Gamma \vdash_{LJ} A$ **implies** $\Gamma \vdash_{LK} A$**.**

... but one can get a converse if A contains enough double negations:

- $(\_)^\star$ is a formula translation which introduces double negations at special places (e.g. atoms, $\vee$ and $\exists$);
- $\vdash_{LK} A \Leftrightarrow A^\star$;
- $\Gamma \vdash_{LK} A$ iff $\Gamma^\star \vdash_{LJ} A^\star$

Towards linear logic

Can we restrict all the structural rules?

Removing all structural rules: logic very very good properties but poorly expressive.

Disjunction and conjunctions splits into two connectives each!

$\wedge \longmapsto \ \&$ and $\otimes$ $\qquad\qquad\qquad$ $\vee \longmapsto \invamp$ and $\oplus$.

Reintroduce structural rules in a controlled way, via a modality, ! and ?, playing the role of double negation for LJ.

The modalities relate the copies of conjunction and disjunction:

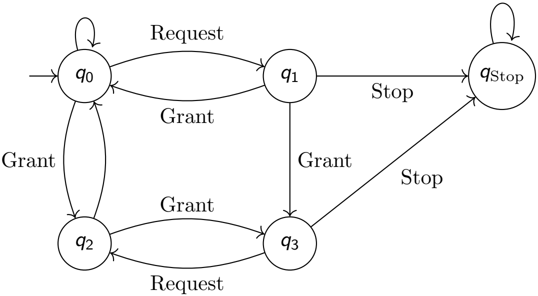
$$!(A \& B) \ \simeq \ !A \otimes !B$$

LK and LJ can be decomposed into LL.

Background on Finite Automata and Model-Checking

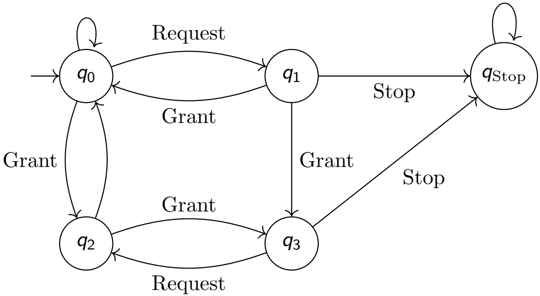
- **Automata**: powerful tool for modeling and verifying systems.
- Used to represent the behavior of systems in a compact form.
- **Model-checking**: technique that uses finite automata to verify properties of systems.
- Goal: ensure that all behaviors of the system behaves are correct.

Example



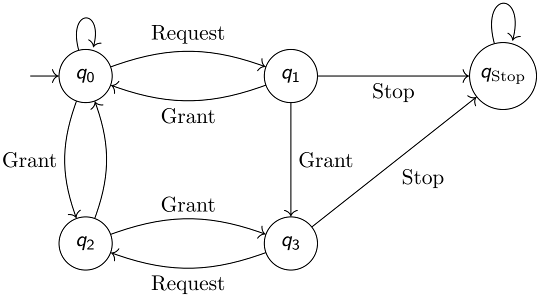
q_0
q_1
q_2
q_3
q_{Stop}
Request
Grant
Stop
Grant
Grant
Stop
Grant
Request

Example



Specification: "Every request is eventually granted, unless stop"

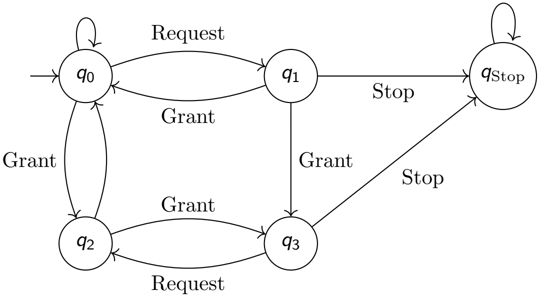
Example



Specification: "Every request is eventually granted, unless stop"

LTL Formula: $\varphi = F(\text{Stop}) \vee (G(\text{Request} \rightarrow F(\text{Grant})))$

Example

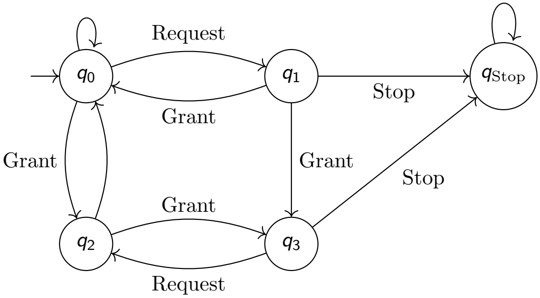


Specification: "Every request is eventually granted, unless stop"

LTL Formula: $\varphi = F(\text{Stop}) \vee (G(\text{Request} \rightarrow F(\text{Grant})))$

Model-checking: "Does the automaton satisfy the LTL formula φ ?"

Example

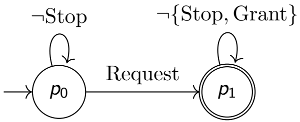


Specification: "Every request is eventually granted, unless stop"

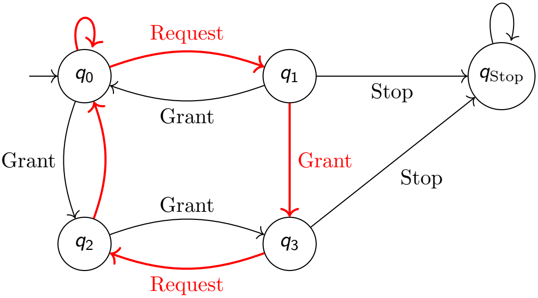
LTL Formula: $\varphi = F(\mathrm{Stop}) \vee (G(\mathrm{Request} \to F(\mathrm{Grant})))$

Model-checking: "Does the automaton satisfy the LTL formula φ ?"

Automaton for $\neg\varphi$:



Example

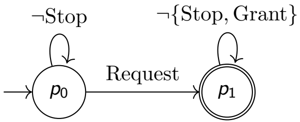


Specification: "Every request is eventually granted, unless stop"

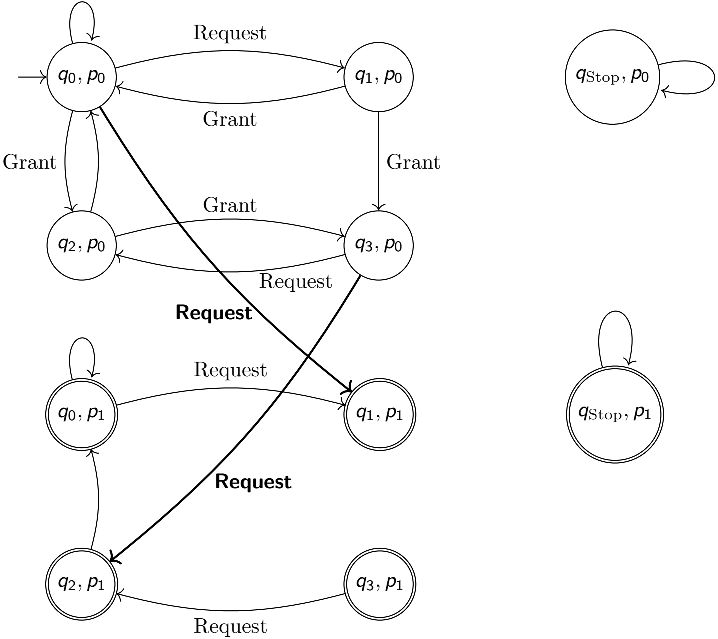
LTL Formula: $\varphi = F(\text{Stop}) \vee (G(\text{Request} \to F(\text{Grant})))$

Model-checking: "Does the automaton satisfy the LTL formula φ ?"

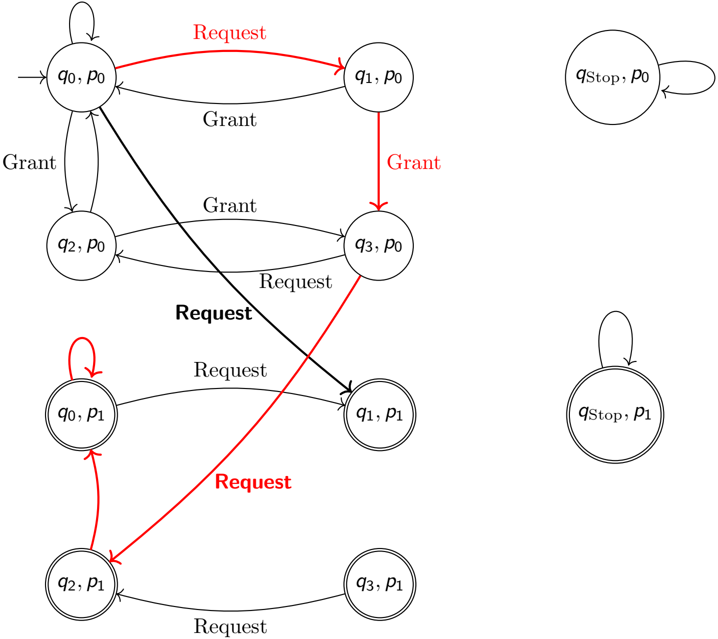
Automaton for $\neg\varphi$:



Detailed construction of the automaton



Detailed construction of the automaton



Background on Fixed-Point Theorems

Tarski's Fixed Point Theorem - Statement

Theorem (Tarski 1955; Knaster 1928)

Let L be a *complete lattice* and $f : L \to L$ a *monotone* function. Then, the set of *fixed points* of f forms a complete lattice.

In particular, there exists a least fixed point $\mu.f$ and a greatest fixed point $\nu.f$.

- **Fixed point**: $x \in L$ such that $x = f(x)$.
- **Monotone**: $x \leq y$ implies $f(x) \leq f(y)$.
- **Complete lattice**: see next.

Complete Lattices

Definition

A **complete lattice** is a partially ordered set L such that every subset $S \subseteq L$ has:

- a *supremum* ($\sup S$) and
- an *infimum* ($\inf S$).

Examples

- $(\mathcal{P}(X) \subseteq)$, the powerset of a set X,
- $\mathbb{Z}^+ = \mathbb{Z} \cup \{\bot, \top\}$,
- $\mathbb{R}^+ = \mathbb{R} \cup \{\bot, \top\}$,
- $\mathcal{I}(\mathbb{R}^+) = \{\, [x, y] \mid x, y \in \mathbb{R}^+ \,\}$.

Tarski's Theorem: proof's sketch

- Define $Pre_f = \{ x \in L \mid f(x) \leq x \}$.

Tarski's Theorem: proof's sketch

- Define $Pre_f = \{\, x \in L \mid f(x) \leq x \,\}$.
- Let $p = \bigwedge Pre_f$.

Tarski's Theorem: proof's sketch

- Define $Pre_f = \{ x \in L \mid f(x) \leq x \}$.
- Let $p = \bigwedge Pre_f$.
- Then $f(p) = p$:

Tarski's Theorem: proof's sketch

- Define $Pre_f = \{\, x \in L \mid f(x) \leq x \,\}$.
- Let $p = \bigwedge Pre_f$.
- Then $f(p) = p$:
  - $f(p) \leq p$, using monotonicity.

Tarski's Theorem: proof's sketch

- Define $Pre_f = \{ x \in L \mid f(x) \leq x \}$.
- Let $p = \bigwedge Pre_f$.
- Then $f(p) = p$:
  - $f(p) \leq p$, using monotonicity.
  - $p \leq f(p)$, since $p \in Pre_f$ and Pre_f is closed under f.

Tarski's Theorem: proof's sketch

- Define $Pre_f = \{ x \in L \mid f(x) \leq x \}$.
- Let $p = \bigwedge Pre_f$.
- Then $f(p) = p$:
  - $f(p) \leq p$, using monotonicity.
  - $p \leq f(p)$, since $p \in Pre_f$ and Pre_f is closed under f.
- Therefore, p is the *least fixed point* of f:

$$p = \mu.f \,.$$

Tarski's Theorem: proof's sketch

- Define $Pre_f = \{ x \in L \mid f(x) \leq x \}$.
- Let $p = \bigwedge Pre_f$.
- Then $f(p) = p$:
  - $f(p) \leq p$, using monotonicity.
  - $p \leq f(p)$, since $p \in Pre_f$ and Pre_f is closed under f.
- Therefore, p is the *least fixed point* of f:

$$p = \mu.f\ .$$

- More then that: p is the **least prefixed point** of f,
  that is, the least element of Pre_f.

Kozen's axiomatisation of the least fixed point

Proposition

The least fixed point of a monotone $f : L \to L$ is uniquely characterized by the following properties:

$$
\begin{array}{rcll}
 & & f(\,\mu.f\,) \leq f\,, & \\
f(x) \leq x & \Rightarrow & \mu.f \leq x & \quad \textit{(for all } x \in L\textit{)}.
\end{array}
$$

NB : if L is not complete, then the least fixed point of f mnight not be its least prefixed point.

A Converse of Tarski's Theorem

Theorem (Davis 1955)

Let L be a partially ordered set.
The following two statements are equivalent:

1. L is a **complete lattice**.
2. Every **monotone** function $f : L \to L$ has at least one **fixed point**.

Kleene's Iteration (another fixed point theorem)

Theorem (Scott 1972, after Kleene 1952)

Let L be a complete lattice with least element $\bot$.
Let $f : L \to L$ be a monotone and **continuous** function.
Then, the least fixed point of f is given by:

$$\mu.f = \bigvee \{\, f^n(\bot) \mid n \in \mathbb{N} \,\}$$

- **Continuous**: preserves suprema of ω-chains, that is

$$f(\bigvee_n x_n) = \bigvee_n f(x_n)\,,$$

if $\{\, x_n \mid n \geq 0 \,\}$ is a chain.

Kleene's Iteration

Construction: Start from $\perp$ and iterate f:

$$\perp, \quad f(\perp), \quad f(f(\perp)), \quad f^3(\perp), \quad \dots$$

- The sequence $(f^n(\perp))$ is increasing.
- Its limit is the least fixed point.
- **Constructive method** used in programming.
- And in Logic?

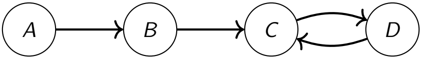
Idea of the Proof

- Since f is monotone, the chain $\perp \leq f(\perp) \leq f^2(\perp) \leq \ldots$ is well-ordered.
- Because f is continuous, we have:

$$f\left(\bigvee_n f^n(\perp)\right) = \bigvee_n f^{n+1}(\perp)$$

- Thus $\bigvee_n f^n(\perp)$ is a fixed point.
- It is the **least** fixed point as it is built solely from $\perp$ by applications of f.

Example: Kleene's Iteration for Graph Reachability



Computing reachable nodes from A.

- Domain: $\mathcal{P}(\{A, B, C, D\})$
- Monotone function:

$$F(S) = \{ A \} \cup \{y \mid \exists x \in S \text{ such that } x \to y\}$$

$F^0(\emptyset) = \emptyset$

$F^1(\emptyset) = F(\emptyset) = \{ A \}$

$F^2(\emptyset) = F(\{ A \}) = \{ A, B \}$

$F^3(\emptyset) = F(\{ A, B \}) = \{ A, B, C \}$

$F^4(\emptyset) = F(\{ A, B, C \}) = \{ A, B, C, D \}$

$F^5(\emptyset) = F(\{ A, B, C, D \}) = \{ A, B, C$

Generalized Kleene's Iteration

Theorem

Let L be a complete lattice with least element $\bot$.
Let $f : L \to L$ be a monotone function.
Then, the least fixed point of f is given by:

$$\mu.f = \bigvee \{\, f^{\alpha}(\bot) \mid \alpha \in \mathtt{Ord} \,\}$$

Here, for an ordinal α,

$$f^{\alpha} = \begin{cases} f(f^{\beta}(bot))\,, & \text{if } \alpha = \beta + 1, \\ \bigvee_{\beta < \alpha} f^{\beta}(\bot)\,, & \text{if } \alpha \text{ is a limit ordinal.} \end{cases}$$

One more Kleene's Iteration

Theorem

Let L be a **DCPO** with a least element $\perp$.
Let $f : L \to L$ be a monotone continuous function.
Then, the least fixed point of f is given by:

$$\mu.f = \bigvee \{ f^n(\perp) \mid n \in \mathbb{N} \}$$

- **DCPO** : a poset for which every chain has a supremum.

Constructive or not ?

- Which one is a constructive fixed point theorem ?
    - Tarski-Knaster?
    - Kleene's ?

Monotonicity of Least Fixed Points

Proposition

- **Monotonicity:** *If $f, g : L \longrightarrow L$ are monotone and $f \leq g$, then*

$$\mu.f \leq \mu.g$$

- **Parameters:** *Let*

$$f : L \times M \longrightarrow L$$

  be a monotone function of the variables x, y.
  Then, the mapping

$$y \mapsto \mu x.f(x, y)$$

  is a monotone function $M \longrightarrow L$.

The Diagonal Property

Proposition

Let L be a complete lattice, and

$$f : L \times L \to L$$

be a monotone function.

Then:

$$\mu x.f(x,x) = \mu y.\mu x.f(x,y)\,.$$

The Rolling Property

Proposition

Let L and M be two complete lattices, and

$$f : L \to M \quad \text{and} \quad g : M \to L$$

be monotone functions.

Then:

$$\mu. g \circ f = g(\, \mu.f \circ g\,)\,.$$

The Bekič Property (1969)

Proposition (Bekić 1984)

Let L and M be two complete lattices, and

$$f : L \times M \to L \quad \text{and} \quad g : L \times M \to M$$

two monotone functions.

Then:

$$\begin{aligned} &\mu(x, y).\,(f(x, y), g(x, y)) \\ &\qquad = (\mu x.\, f(x, \mu y.\, g(x, y)),\ \mu y.\, g(\mu x.\, f(x, y), y)) \end{aligned}$$

Consequences

- A fixed point over a product $L \times M$ can be **decomposed** into two simpler fixed points.
- Vectorial μ-calculi (automata) can be 'simulated' by scalar ones.

Pataraia's theorem

Recall the generalized Scott-Kleene Fixed Point Theorem:

Theorem

If $(P, \leq)$ is a DCPO with a least element $\perp$, then each monotone $f : P \to P$ has a least fixed point.

Theorem (Pataraia 1997)

The above statement is constructive (that is, it holds in a topos).

Bourbaki–Witt Fixed Point Theorem

Theorem (Bourbaki–Witt Fixed Point Theorem)

Let P be a non-empty DCPO. Let $f : P \to P$ be an **inflationary** *function. that is,*

$$\forall x \in P, \quad f(x) \geq x.$$

Then f has at least one fixed point.

- **Inflationary** : $\forall x \in P, \quad x \leq f(x)$.
- This theorem is not constructive : Bauer and Lumsdaine 2013

References

- Bauer, Andrej and Peter Lefanu Lumsdaine (2013). "On the Bourbaki-Witt principle in toposes". English. In: *Math. Proc. Camb. Philos. Soc.* 155.1, pp. 87–99. ISSN: 0305-0041. DOI: 10.1017/S0305004113000108.
- Bekić, Hans (1984). "Definable Operation in General Algebras, and the Theory of Automata and Flowcharts". In: *Programming Languages and Their Definition - Hans Bekic (1936-1982)*. Ed. by Cliff B. Jones. Vol. 177. Lecture Notes in Computer Science. Springer, pp. 30–55. DOI: 10.1007/BFB0048939. URL: https://doi.org/10.1007/BFb0048939.
- Davis, Anne C. (1955). "A characterization of complete lattices". English. In: *Pac. J. Math.* 5, pp. 311–319. ISSN: 1945-5844. DOI: 10.2140/pjm.1955.5.311.
- Kleene, S. C. (1952). *Introduction to metamathematics.* English. 9th repr. Amsterdam: North-Holland. ISBN: 0-7204-2103-9.
- Knaster, B. (1928). "Un théorème sur les fonctions d'ensembles.". French. In: *Ann. Soc. Polon. Math.* 6,

Background on (typed) λ-calculus

Church's λ-calculus

Introduced by A. Church, independently of nat. deduction.
Only 3 constructions for λ-terms:

1. variable, $x, y, z, \dots \in \mathcal{V}$,
2. abstraction, $\lambda x.p$: function associating p to x and
3. application, (pq), of function p to an argument q.

$\Longrightarrow$ $$\boxed{(\lambda x.p)q \longrightarrow_\beta p\{q/x\}} \qquad (\longrightarrow_\beta^\star, =_\beta)$$

(possibly extended with constants $(a, b, c\dots)$ and δ-rules.)

Very good properties:

- Confluence: if $p \longrightarrow_\beta^\star q$ & $p \longrightarrow_\beta^\star q'$, $\exists r$ s.t. $q \longrightarrow_\beta^\star r$ & $q' \longrightarrow_\beta^\star r$.
- Uniqueness of the normal form (if it exists) + strategy to reach nf.
- Diverging computations: *(with $\Delta = \lambda x.xx$, $\Omega = \Delta\Delta$)*
  $\Omega \longrightarrow_\beta (x)x\{\Delta/x\} = \Omega \longrightarrow_\beta \dots$ exploits self-application.
- Normalizability is undecidable.

Basis of functional and recursive programming.

Encoding data

Representing numbers with λ-terms:

- $\overline{n} = \lambda s.\lambda z.s^n z = \lambda s.\lambda z.(s(s...(sz)))$
- $\text{add} = \lambda n.\lambda m.\lambda s.\lambda z.ns(msz)$ $\qquad \text{add}\ \overline{n}\ \overline{m} \longrightarrow_\beta^\star \overline{n+m}$

Representing pairs:

- $\overline{\langle u, v\rangle} = \lambda p.(p\ uv)$ and $\overline{\pi_i} = \lambda p.(p\ \lambda y_1.\lambda y_2.y_i)$.

Representing booleans (and finite sets):

- $\overline{tt} = \lambda x_1.\lambda x_2.x_1$ $\qquad \overline{ff} = \lambda x_1.\lambda x_2.x_2$
- $\overline{ifte} = \lambda b.\lambda c_1.\lambda c_2.(b\ c_1 c_2)$

Representing words (Given an alphabet $A = \{a_1, \ldots, a_k\}$):

- $\overline{\epsilon} = \lambda a_1, \ldots, a_k.\lambda e.e$ $\qquad \overline{a_i \cdot w} = \lambda a_1, \ldots, a_k.\lambda e.a_i\overline{w}$
- $\overline{concat} = \lambda v, w.\lambda \vec{a_j}.\lambda e.(v\ \vec{a_j})(w\ \vec{a_j}e)$.

With Universality, comes non-termination

Non-termination and Fixed-point combinators

- $\Delta_C = \lambda x.C[xx], \Omega_C = \Delta_C\Delta_C$; *(C not binding x)*
- $\Omega_C \longrightarrow_\beta C[\Omega_C]$ $(\Omega = \Omega_\Box)$
- Let p with $x \notin FV(p)$ and $C = (p\ \Box)$, then $\Omega_C =_\beta p\ \Omega_C$

$\Longrightarrow$ Every λ-term p has a fixed-point

- With $\mathcal{Y} = \lambda f.\Omega_{(f\Box)}$, for any p, $\mathcal{Y}p =_\beta p(\mathcal{Y}p)$
- There are λ-terms computing fixed-points of their arguments, fixed-point combinators.

Head normalization and Böhm trees

Fact

Every λ-term is of the following form: $\lambda x_1, \ldots \lambda x_k.(h\ p_1 \ldots p_l) \quad k, l \geq 0$
with h being either (i) a (head) variable y or (ii) a (head) redex $(\lambda y.q)r$.

- Head reduction: (deterministic) sub-relation $\longrightarrow_h$ of $\longrightarrow_\beta$ which reduces the head redex – if there is one – and stops otherwise.
- A head normal form (hnf) is a term without head redex.

Head normalization and Böhm trees

Fact

Every λ-term is of the following form: $\lambda x_1, \ldots \lambda x_k.(h\ p_1 \ldots p_l) \quad k, l \geq 0$
with h being either (i) a (head) variable y or (ii) a (head) redex $(\lambda y.q)r$.

- Head reduction: (deterministic) sub-relation $\longrightarrow_h$ of $\longrightarrow_\beta$ which reduces the head redex – if there is one – and stops otherwise.
- A head normal form (hnf) is a term without head redex.
- Stable prefix if $p = \lambda x_1, \ldots \lambda x_k.(y\ p_1 \ldots p_l) \longrightarrow_\beta^\star q$, then q is in hnf. Moreover, $q = \lambda x_1, \ldots \lambda x_k.(y\ q_1 \ldots q_l)$ with $p_i \longrightarrow_\beta^\star q_i, 1 \leq i \leq l$.
$$\mathcal{H}_p = \lambda x_1, \ldots \lambda x_k.(y\ \Box_1 \ldots \Box_l)$$
- Böhm trees: The *Böhm tree* $\mathrm{BT}(p)$ of a λ-term p is coinductively defined as follows. ($\perp$ represents a lack of information.)

$$\mathrm{BT}(p) = \begin{cases} \begin{array}{ccc} & \lambda x_1 \ldots x_n.\, y & \\ \mathrm{BT}(p_1) & \cdots & \mathrm{BT}(p_k) \end{array} & \text{if } p \longrightarrow_h^\star \lambda x_1 \ldots x_n.y\ p_1 \cdots p_k \\ \perp & \text{otherwise} \end{cases}$$

Head normalization and Böhm trees

$$\mathrm{BT}(p) = \begin{cases} \begin{array}{c} \lambda x_1 \ldots x_n.\, y \\ \mathrm{BT}(p_1) \quad \cdots \quad \mathrm{BT}(p_k) \end{array} & \text{if } p \longrightarrow_h^\star \lambda x_1 \ldots x_n.y \ p_1 \cdots p_k \\ \perp & \text{otherwise} \end{cases}$$

Examples ($\mathrm{BT}(\mathcal{Y})$, $\mathrm{BT}(\mathcal{Y}I)$, with $I = \lambda x.x$)

- $\mathcal{Y} = \lambda f.\Omega_{f\Box} \longrightarrow_h \lambda f.f\Omega_{f\Box} \longrightarrow_\beta \lambda f.f(f\Omega_{f\Box}) \longrightarrow_\beta \lambda f.f(f(f\Omega_{f\Box}))\ldots$
- $\mathcal{Y}I \longrightarrow_h (\lambda f.f\Omega_{f\Box})I \longrightarrow_h I\Omega_{I\Box} \longrightarrow_h \Omega_{I\Box} \longrightarrow_h I\Omega_{I\Box} \longrightarrow_h \Omega_{I\Box}\ldots$

Head normalization and Böhm trees

$$\mathrm{BT}(p) = \begin{cases} \begin{array}{c} \lambda x_1 \ldots x_n .\, y \\ \mathrm{BT}(p_1) \quad \cdots \quad \mathrm{BT}(p_k) \end{array} & \text{if } p \longrightarrow_h^{\star} \lambda x_1 \ldots x_n . y \;\; p_1 \cdots p_k \\ \perp & \text{otherwise} \end{cases}$$

Examples ($\mathrm{BT}(\mathcal{Y})$, $\mathrm{BT}(\mathcal{Y} I)$, with $I = \lambda x.x$)

- $\mathcal{Y} = \lambda f.\Omega_{f\square} \longrightarrow_h \lambda f.f\Omega_{f\square} \longrightarrow_\beta \lambda f.f(f\Omega_{f\square}) \longrightarrow_\beta \lambda f.f(f(f\Omega_{f\square}))\ldots$
- $\mathcal{Y} I \longrightarrow_h (\lambda f.f\Omega_{f\square})I \longrightarrow_h I\Omega_{I\square} \longrightarrow_h \Omega_{I\square} \longrightarrow_h I\Omega_{I\square} \longrightarrow_h \Omega_{I\square}\ldots$

$$\mathrm{BT}(\mathcal{Y}) = \begin{array}{c} \lambda f .\, f \\ | \\ f \\ | \\ f \\ | \\ f \\ | \\ \vdots \end{array} \qquad \mathrm{BT}(\mathcal{Y}\lambda x.x) = \perp$$

Head normalization and Böhm trees

$$\mathrm{BT}(p) = \begin{cases} \begin{tikzcd}[column sep=small] & \lambda x_1 \ldots x_n . y \arrow[dl, no head] \arrow[dr, no head] & \\ \mathrm{BT}(p_1) & \cdots & \mathrm{BT}(p_k) \end{tikzcd} & \text{if } p \longrightarrow_h^\star \lambda x_1 \ldots x_n.y \ p_1 \cdots p_k \\ \bot & \text{otherwise} \end{cases}$$

Examples ($\mathrm{BT}(\mathcal{Y})$, $\mathrm{BT}(\mathcal{Y}I)$, with $I = \lambda x.x$)

- $\mathcal{Y} = \lambda f.\Omega_{f\Box} \longrightarrow_h \lambda f.f\Omega_{f\Box} \longrightarrow_\beta \lambda f.f(f\Omega_{f\Box}) \longrightarrow_\beta \lambda f.f(f(f\Omega_{f\Box}))\ldots$
- $\mathcal{Y}I \longrightarrow_h (\lambda f.f\Omega_{f\Box})I \longrightarrow_h I\Omega_{I\Box} \longrightarrow_h \Omega_{I\Box} \longrightarrow_h I\Omega_{I\Box} \longrightarrow_h \Omega_{I\Box}\ldots$

$$\mathrm{BT}(\mathcal{Y}) = \begin{tikzcd}[row sep=small] \lambda f.f \arrow[d, no head] \\ f \arrow[d, no head] \\ f \arrow[d, no head] \\ f \arrow[d, no head] \\ \vdots \end{tikzcd} \qquad \mathrm{BT}(\mathcal{Y}\lambda x.x) = \bot$$

1. BT of normalizable terms are finite, $\bot$-free and isomorphic to their normal form;
2. All FP combinators have the same BT;
3. BTs induce an equivalence over λ-terms ("having the same BT"): it is a model of the λ-calculus;
4. BT model equates all terms w/o hnf.

Types as Specification

Tame the non-termination of the pure λ-calculus by constraining how terms can be plugged together. Specifying domain/codomain.

Types	Typing context	Typing Judgement
$A, B ::= \mathrm{At} \mid A \to B$	$\Gamma ::= \emptyset \mid x : A, \Gamma$	$\Gamma \vdash p : A$

Typing rules:

$$\frac{}{\Gamma, x : A \vdash x : A}\ Ax \qquad \frac{\Gamma, x : A \vdash p : B}{\Gamma \vdash \lambda x.p : A \Rightarrow B} \Rightarrow I \qquad \frac{\Gamma \vdash p : A \Rightarrow B \quad \Gamma \vdash q : A}{\Gamma \vdash (pq) : B} \Rightarrow E$$

Theorem (Type preservation)

If $\Gamma \vdash p : A$ and $p \longrightarrow_\beta q$, then $\Gamma \vdash q : A$.

Theorem (Normalization)

- *Weak N. If $\Gamma \vdash p : A$ then p has a normal form.*
- *Strong N. If $\Gamma \vdash p : A$ then all reduction sequences from p are finite.*

Types as Specification

Tame the non-termination of the pure λ-calculus by constraining how terms can be plugged together. Specifying domain/codomain.

Types	Typing context	Typing Judgement
$A, B ::= \mathrm{At} \mid A \to B$	$\Gamma ::= \emptyset \mid x : A, \Gamma$	$\Gamma \vdash p : A$

Typing rules:

$$\frac{}{\Gamma, x : A \vdash x : A}\ Ax \qquad \frac{\Gamma, x : A \vdash p : B}{\Gamma \vdash \lambda x.p : A \Rightarrow B} \Rightarrow I \qquad \frac{\Gamma \vdash p : A \Rightarrow B \quad \Gamma \vdash q : A}{\Gamma \vdash (pq) : B} \Rightarrow E$$

Theorem (Type preservation)

If $\Gamma \vdash p : A$ and $p \longrightarrow_\beta q$, then $\Gamma \vdash q : A$.

Theorem (Normalization)

- *Weak N. If $\Gamma \vdash p : A$ then p has a normal form.*
- *Strong N. If $\Gamma \vdash p : A$ then all reduction sequences from p are finite.*

$\lambda x.xx$ cannot be typed, and therefore neither Ω nor $\mathcal{Y}$.

Similarity:

Formulas $\leftrightarrow$ Types;
Nat Ded $\leftrightarrow$ Typed λ-terms;
Cut red. $\leftrightarrow$ β-reduction

Fast growing functions

Ackermann-Péter function:

$$AP(m,n) = \begin{cases} n+1 & \text{if } m=0 \\ AP(m-1,1) & \text{if } m>0, n=0 \\ AP(m-1, AP(m,n-1)) & \text{if } m>0, n>0 \end{cases}$$

non definable in simply typed λ-calculus, because it grows to fast: the termination proof of simply typed λ-calcul is too weak for such a computation.

Fast growing functions

Ackermann-Péter function:

$$AP(m,n) = \begin{cases} n+1 & \text{if } m=0 \\ AP(m-1,1) & \text{if } m>0, n=0 \\ AP(m-1, {\color{red}AP(m,n-1)}) & \text{if } m>0, n>0 \end{cases}$$

non definable in simply typed λ-calculus, because it grows to fast: the termination proof of simply typed λ-calcul is too weak for such a computation.

Gödel's System T extend simply-typed λ-calculus and can define AP. Contains primitive integers, a higher-order recursor whose type corresponds to the axiom of recurrence of alertPeano arithmetic.

Girard's System F allows to define all functions definable in T and much more. F extends Curry-Howard correspondence to second-order logic (quantification over predicates).

Whatever effort we make to extend those typed (and **terminating**) systems, there are total recursive functions that one cannot represent.

Typed λ-calculus and Curry-Howard correspondence

Deduction	Programming
Formula	Data Type
Natural deduction	Typed λ-term
Cut	β-redex
Cut simplification	β-reduction
Cut-free proof	value, evaluated program
Implication	Function type
Conjunction	Product Type (pairs)
Disjunction	Sum Type (disjoint union)
True	Singleton Type (unit)
Recurrence Axiom	Recursion
Second-order Quantification	Polymorphism
...	...

Another way to enrich the simply-typed λ-calculus: λY-calculus

Consider simply typed λ-calculus extended with a family constants $(Y_A)_{A \in Type}$ having the following typing rule and δ-rule:

$$\frac{}{\vdash Y_A : (A \to A) \to A}\; Y \qquad\qquad Y\,p \longrightarrow_Y p(Y\,p)$$

- Corresponds to a let rec.
- Confluence, Type preservation;
- Obviously loses termination: $\vdash Y_{A \to A}\,\lambda x^A.x : A$ loops;
- Head reduction / hnf extend naturally;
- λY-Böhm trees.

In which cases is $\mathrm{BT}(p)$ finite? $\perp$-free?
Can we find sufficient conditions?

(Enriched with primitive integers and booleans, one gets PCF.)

Some logical systems

LK Inference Rules (1)

Identity Group (*Axiom and Cut***)**

$$\frac{}{A \vdash A}\ Ax \qquad\qquad \frac{\Gamma_1 \vdash A, \Delta_1 \quad \Gamma_2, A \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}\ Cut$$

Structural Group (*Exchange, Weakening and Contraction***)**

$$\frac{\Gamma_1, B, A, \Gamma_2 \vdash \Delta}{\Gamma_1, A, B, \Gamma_2 \vdash \Delta}\ LEx \qquad\qquad \frac{\Gamma \vdash \Delta_1, B, A, \Delta_2}{\Gamma \vdash \Delta_1, A, B, \Delta_2}\ REx$$

$$\frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}\ LW \qquad\qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash A, \Delta}\ RW$$

$$\frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}\ LC \qquad\qquad \frac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta}\ RC$$

LK Inference Rules (2)

Logical Group ($\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\forall$, $\exists$)

$$\frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}\ L\neg \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta}\ R\neg$$

$$\frac{\Gamma, A_j \vdash \Delta}{\Gamma, A_1 \wedge A_2 \vdash \Delta}\ L\wedge j \quad (j \in \{1,2\}) \qquad \frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta}\ R\wedge$$

$$\frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}\ L\vee \qquad \frac{\Gamma \vdash A_j, \Delta}{\Gamma \vdash A_1 \vee A_2, \Delta}\ R\vee j \quad (j \in \{1,2\})$$

$$\frac{\Gamma_1 \vdash A, \Delta_1 \quad \Gamma_2, B \vdash \Delta_2}{\Gamma_1, \Gamma_2, A \Rightarrow B \vdash \Delta_1, \Delta_2}\ L\Rightarrow \qquad \frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \Rightarrow B, \Delta}\ R\Rightarrow$$

$$\frac{\Gamma, A[t/x] \vdash \Delta}{\Gamma, \forall x A \vdash \Delta}\ L\forall \qquad \frac{\Gamma \vdash A, \Delta}{\Gamma \vdash \forall x A, \Delta}\ R\forall \quad (*)$$

$$\frac{\Gamma, A \vdash \Delta}{\Gamma, \exists x A \vdash \Delta}\ L\exists \quad (*) \qquad \frac{\Gamma \vdash A[t/x], \Delta}{\Gamma \vdash \exists x A, \Delta}\ R\exists$$

(*) For those rules, $x \notin FV(\Gamma, \Delta)$.

Some formal theorems

- $A \vee B \vdash B \vee A$ — Commutativity of disjunction
- $\vdash A \vee \neg A$ — *Tertium non datur*
- $\vdash ((A \Rightarrow B) \Rightarrow A) \Rightarrow A$ — Peirce's Law
- $\vdash (\neg\neg A) \Rightarrow A$ — Double Negation Elimination
- $\vdash \exists x \forall y (P(x) \Rightarrow P(y))$ — Drinker Poperty
- $A \vee B \vdash \neg(\neg A \wedge \neg B)$ — de Morgan Law
- $\vdash (A \Rightarrow B) \vee (B \Rightarrow A)$
- $\vdash \neg\neg(A \vee \neg A)$ — intuitionistic *Tertium non datur*
- $(p \Rightarrow q) \vdash (\neg q \Rightarrow \neg p)$
- $(\neg q \Rightarrow \neg p) \vdash (p \Rightarrow q)$

Prove them!

LJ Sequent Calculus

Identity

$$\frac{}{A \vdash A}\ ax \qquad \frac{\Gamma_1 \vdash A \quad \Gamma_2, A \vdash \Xi}{\Gamma_1, \Gamma_2 \vdash \Xi}\ cut$$

Struct.

$$\frac{\Gamma_1, B, A, \Gamma_2 \vdash \Xi}{\Gamma_1, A, B, \Gamma_2 \vdash \Xi}\ LEx \quad \frac{\Gamma \vdash \Xi}{\Gamma, A \vdash \Xi}\ LW \quad \frac{\Gamma \vdash}{\Gamma \vdash A}\ RW \quad \frac{\Gamma, A, A \vdash \Xi}{\Gamma, A \vdash \Xi}\ LC$$

Logical

$$\frac{\Gamma \vdash A}{\Gamma, \neg A \vdash}\ L\neg \qquad \frac{\Gamma, A \vdash}{\Gamma \vdash \neg A}\ R\neg$$

$$\frac{\Gamma_1 \vdash A \quad \Gamma_2, B \vdash \Xi}{\Gamma_1, \Gamma_2, A \Rightarrow B \vdash \Xi}\ L \Rightarrow \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}\ R \Rightarrow$$

$$\frac{\Gamma, A \vdash \Xi}{\Gamma, A \wedge B \vdash \Xi}\ L \wedge 1 \qquad \frac{\Gamma, B \vdash \Xi}{\Gamma, A \wedge B \vdash \Xi}\ L \wedge 2 \qquad \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ R\wedge$$

$$\frac{\Gamma, A \vdash \Xi \quad \Gamma, B \vdash \Xi}{\Gamma, A \vee B \vdash \Xi}\ L\vee \qquad \frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}\ R \vee 1 \qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}\ R \vee 2$$

$$\frac{\Gamma, A[t/x] \vdash \Xi}{\Gamma, \forall x A \vdash \Xi}\ L\forall \quad \frac{\Gamma \vdash A}{\Gamma \vdash \forall x A}\ R\forall\ (*) \qquad \frac{\Gamma, A \vdash \Xi}{\Gamma, \exists x A \vdash \Xi}\ L\exists\ (**) \quad \frac{\Gamma \vdash A[t/x]}{\Gamma \vdash \exists x A}\ R\exists$$

(*) For $R\forall$, $x \notin FV(\Gamma)$. (**) For $L\exists$, $x \notin FV(\Gamma, \Xi)$.

LL Sequent Calculus

Indentity Group:

$$\frac{}{\vdash A^\perp, A}\ ax \qquad \frac{\vdash A, \Gamma \quad \vdash A^\perp, \Delta}{\vdash \Gamma, \Delta}\ cut$$

Structural Group:

$$\frac{\vdash \Gamma, B, A, \Delta}{\vdash \Gamma, A, B, \Delta}\ Ex$$

Logical Group:

$$\frac{\vdash F, G, \Gamma}{\vdash F \parr G, \Gamma}\ \parr \qquad \frac{\vdash F, \Gamma \quad \vdash G, \Delta}{\vdash F \otimes G, \Gamma, \Delta}\ \otimes$$

$$\frac{\vdash F, \Gamma \quad \vdash G, \Gamma}{\vdash F \& G, \Gamma}\ \& \qquad \frac{\vdash F, \Gamma}{\vdash F \oplus G, \Gamma}\ \oplus 1 \qquad \frac{\vdash G, \Gamma}{\vdash F \oplus G, \Gamma}\ \oplus 2$$

$$\frac{}{\vdash 1}\ 1 \qquad \frac{\vdash \Gamma}{\vdash \perp, \Gamma}\ \perp \qquad \frac{}{\vdash \top, \Gamma}\ \top$$

$$\frac{\vdash F, \Gamma}{\vdash ?F, \Gamma}\ ? \qquad \frac{\vdash F, ?\Gamma}{\vdash !F, ?\Gamma}\ !$$

$$\frac{\vdash \Gamma}{\vdash ?F, \Gamma}\ ?W \qquad \frac{\vdash ?F, ?F, \Gamma}{\vdash ?F, \Gamma}\ ?C$$