Mechanising your proofs in Coq

From zero to Cut-Admissibility in less than three months

Marco Carbone
IT University of Copenhagen, Denmark

Goal of this talk…

- Promote proof mechanisation for our community
- Talk about my (short) experience

- Not about
  - How to prove Cut Admissibility
  - How to prove Cut Admissibility in Coq
  - Showing Off my Coq skills

My Coding Background

- Not much coding experience (Basic, Pascal, C, **Java, Haskell**)
- Almost null experience with Theorem Provers/Proof Assistants:
    - Pre-historic Coq (undergraduate, meaningless exercises, 1999)
    - HLF/TWELF (failed attempt, nothing proven, 2013)

The Ravara's challenge

- @POPL 2019, Cascais (Portugal)
- It's time for our community to start mechanising
- Start with known proofs
- Meet again in Prague (Czech Rep.)

Initial Objective

- Pi-calculus with Binary Sessions + Types
  - ==> Multiparty Sessions + Types

Why Coq?

- Considered possibilities: Coq, Isabelle, Agda, Twelf.
- Most popular (perception)
- Interest from industry
- Talked to Jesper Begtson (Psi Calculi)

Learning Phase

Benjamin Pierce’s book(s)

softwarefoundations.cis.upenn.edu

- Brilliant (me)

- Not right structures (others)

Learning Phase

Functional Programming in Coq (*Basics*)

Proof by Induction (*Induction*)

Working with Structured Data (*Lists*)

Polymorphism and Higher-Order Functions (*Poly*)

More Basic Tactics (*Tactics*)

Logic in Coq (*Logic*)

Inductively Defined Propositions (*IndProp*)

PUMPING LEMMA!!!!!

Total and Partial Maps (*Maps*)

The Curry-Howard Correspondence (*ProofObjects*)

Induction Principles (*IndPrinciples*)

Properties of Relations (*Rel*)

Simple Imperative Programs (*Imp*)

Lexing and Parsing in Coq (*ImpParser*)

An Evaluation Function for Imp (*ImpCEvalFun*)

Extracting ML from Coq (*Extraction*)

Intuitionistic Linear Logic

Sequent Calculus

$$\frac{}{A \vdash A}\,\mathrm{ax} \qquad \frac{\Gamma \vdash A \qquad \Delta, A \vdash C}{\Gamma, \Delta \vdash C}\,\mathrm{cut}$$

$$\frac{\Gamma \vdash A \qquad \Delta \vdash B}{\Gamma, \Delta \vdash A \otimes B}\,\otimes R \qquad \frac{\Gamma, A, B \vdash C}{\Gamma, A \otimes B \vdash C}\,\otimes L \qquad \frac{}{\vdash 1}\,1R \qquad \frac{\Gamma \vdash C}{\Gamma, 1 \vdash C}\,1L$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \multimap B}\multimap R \qquad \frac{\Gamma \vdash A \qquad \Delta, B \vdash C}{\Gamma, \Delta, A \multimap B \vdash C}\multimap L$$

$$\frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \,\&\, B}\,\& R \qquad \frac{\Gamma, A \vdash C}{\Gamma, A \,\&\, B \vdash C}\,\&_1 L \qquad \frac{\Gamma, B \vdash C}{\Gamma, A \,\&\, B \vdash C}\,\&_2 L$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \oplus B}\,\oplus_1 R \qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \oplus B}\,\oplus_2 R \qquad \frac{\Gamma, A \vdash C \qquad \Gamma, B \vdash C}{\Gamma, A \oplus B \vdash C}\,\oplus L$$

Sequent Calculus in Coq

Cut Admissibility

Theorem. If $\Gamma \vdash A$ and $\Delta, A \vdash C$ then $\Gamma, \Delta \vdash C$.

My proof idea before Coq:

- consider cut as a rule
- permute rules (commuting conversions/structural congruence)
- reduce cuts to cuts on smaller formulas (reduction semantics)

===> very intuitive reasoning.

Commuting Conversion

$$\dfrac{\dfrac{\begin{matrix}\mathcal{D}_1 \\ \Delta_1, B_1, B_2 \Rightarrow A\end{matrix}}{\Delta_1, B_1 \otimes B_2 \Rightarrow A}\otimes L \qquad \begin{matrix}\mathcal{E} \\ \Delta', A \Rightarrow C\end{matrix}}{\Delta_1, B_1 \otimes B_2, \Delta' \Rightarrow C}(\mathsf{cut}_A)$$

$$\dfrac{\dfrac{\begin{matrix}\mathcal{D}_1 \\ \Delta_1, B_1, B_2 \Rightarrow A\end{matrix} \qquad \begin{matrix}\mathcal{E} \\ \Delta', A \Rightarrow C\end{matrix}}{\Delta_1, B_1, B_2, \Delta' \Rightarrow C}(\mathsf{cut}_A)}{\Delta_1, B_1 \otimes B_2, \Delta' \Rightarrow C}\otimes L$$

from F. Pfenning Lecture Notes on Linear Logic

Cut Reductions

$$
\dfrac{\dfrac{\Delta, A \vdash B}{\Delta \vdash A \multimap B}\multimap R \qquad \dfrac{\Delta_1 \vdash A \quad \Delta_2, B \vdash C}{\Delta_1, \Delta_2, A \multimap B \vdash C}\multimap L}{\Delta, \Delta_1, \Delta_2 \vdash C}\,\mathsf{cut}
$$

$$\longrightarrow$$

$$
\dfrac{\dfrac{\Delta_1 \vdash A \quad \Delta, A \vdash B}{\Delta, \Delta_1 \vdash B}\,\mathsf{cut} \qquad \Delta_2, B \vdash C}{\Delta, \Delta_1, \Delta_2 \vdash C}\,\mathsf{cut}
$$

Cut Admissibility

$$\dfrac{\stackrel{\mathcal{D}}{\Delta \Rightarrow A} \quad \stackrel{\mathcal{E}}{\Delta', A \Rightarrow C}}{\Delta, \Delta' \Rightarrow C}$$

Proof: By a nested induction, first on the structure of A and second simultaneously on the structures of $\mathcal{D}$ and $\mathcal{E}$. This means we can appeal to the induction hypothesis

1. when the cut formula A becomes smaller, or
2. the cut formula A stays the same and
   (a) either $\mathcal{D}$ becomes smaller and $\mathcal{E}$ stays the same, or
   (b) $\mathcal{D}$ stays the same and $\mathcal{E}$ becomes smaller.

from F. Pfenning Lecture Notes on Linear Logic

Cut Admissibility in Coq

Current/Future Work

- Restriction of Linear Logic for modelling forwarders (BehAPI - started at CMU - now with C. Schurmann)
  - Formalised (proved): 7 different cut theorems on top of this formalisation
- Subject Reduction for Binary Session Types (*see you in Leicester*)
- Subject Reduction for Multiparty Session Types
- **Subject Reduction for Asynch. Multiparty Session Types**