Lecture 2: Initiality and Cut-Admissibility

In this lecture, we will prove Theorems 4 and 5. Before we start, I would like to reflect a bit about why these theorems are important. In the previous courses from last week, you have learned about underlying semantic models to justify the choice of connectives and rules. For linear logic, one could do the same, but this would lead us too far astray from our mission on logical frameworks. We therefore need to convince ourselves that the choice of rules is good — or in other words, that the right rules don’t give more information than the left rules consume, and that the left rules don’t consume more information than the right rules can be provide.

Initiality therefore refers to the property that — for all non-atomic formulas $A$ — the rules are powerful enough to deduce $A$ from $A$, by taking $A$ apart on the left and reconstructing it on the right. The other property, called the cut-admissibility, refers to the property that given two (cut-free) proofs, where one proves $A$, and the other relies on $A$, they can be combined into one cut-free proof. Initiality and cut-admissibility are duals to one another. We prove both.

**Initiality**  The initiality property can be expressed in form of generalized axiom rule

\[
\Gamma; A \Rightarrow A
\]

extending the rule restricted to atoms $P$. We called this rule $pax$. This property guarantees that the axioms on arbitrary formulas do not add more expressiveness to the system.

**Theorem 8**  For all $A$, the rule

\[
\Gamma; A \Rightarrow A
\]

is admissible.

**Proof:** The proof is by induction on the formula

Case: $A = P$ trivial.

Case: $A = B \rightarrow C$

\[
\begin{align*}
\Gamma; A & \Rightarrow A & \Gamma; B & \Rightarrow B \\
\Gamma; A \rightarrow B, A \Rightarrow B & \rightarrow L \\
\Gamma; A \rightarrow B \Rightarrow A \rightarrow B & \rightarrow R
\end{align*}
\]

Case: $A = B \otimes C$

\[
\begin{align*}
\Gamma; A & \Rightarrow A & \Gamma; B & \Rightarrow B \\
\Gamma; A, B & \Rightarrow A \otimes B & \otimes R \\
\Gamma; A \otimes B & \Rightarrow A \otimes B & \otimes L
\end{align*}
\]
Case: $A = !B$

\[
\Gamma; A; A \Rightarrow A \quad \text{ind. hyp}
\]

\[
\frac{}{\Gamma; A; \cdot \Rightarrow A} \quad \text{copy}
\]

\[
\frac{}{\Gamma; A; \cdot \Rightarrow !A} \quad \text{!R}
\]

\[
\frac{}{\Gamma; !A \Rightarrow !A} \quad \text{!L}
\]

Case: $A = \forall x : \tau. B$

\[
\frac{}{\Gamma; A[a/x] \Rightarrow A[a/x]} \quad \text{hyp}
\]

\[
\frac{}{\Gamma; \forall x : \tau. A \Rightarrow A[a/x]} \quad \forall L
\]

\[
\frac{}{\Gamma; \forall x : \tau. A \Rightarrow \forall x : \tau. A} \quad \forall R \quad \text{a fresh}
\]

\[\square\]

\[\square\]

**Cut-Admissibility** The cut-rule is the heart of any logic, in particular here. The cut establishes the soundness of the logic. It provides a natural mechanism for reduction. It defines implicitly the normal form of a proof. It needs to terminate (otherwise cut wouldn’t be admissible) and it cannot get stuck. For many logics it holds for also for the case that we consider infinite derivations (i.e. logic derivations that do not end axioms.) Cut-elimination proofs usually lead to a combinatorial explosion of the number of cut rules that we encounter within a derivation, and yet, it has a simple, elegant and natural proof. We’ll try to go through the cases now. For completeness, we repeat the definition of the cut admissibility theorem here.

**Theorem 9 (Cut-admissibility)**

\[\mathcal{D} \quad \mathcal{E}\]

1. If $\Gamma; \Delta_1 \Rightarrow A$ and $\Gamma; (\Delta_2, A) \Rightarrow B$ then $\Gamma; (\Delta_1, \Delta_2) \Rightarrow B$.

\[\mathcal{D} \quad \mathcal{E}\]

2. If $\Gamma; \cdot \Rightarrow A$ and $(\Gamma, A); \Delta \Rightarrow B$ then $\Gamma; \Delta \Rightarrow B$.

**Proof:** The proof is by simultaneous induction on both cases, lexicographically on the cut-formula and simultaneous induction on the two given derivations. Begin with discussing the first part of the theorem.

The first set of cases to be considered are the axiom cases:

Case: $\mathcal{D} = \frac{}{\Gamma; P \Rightarrow P}$ and $\mathcal{E} = \Gamma; (\Delta_2, P) \Rightarrow B$. Immediate.

Case: $\mathcal{D} = \frac{}{\Gamma; P \Rightarrow P}$ and $\mathcal{E} = \frac{}{\Gamma; P \Rightarrow P}$, Immediate.
The second set of cases are called essential cases. These are the cases, where the action is. An essential case is one where $D$ ends in a right “producing” $A$ and $E$ ends in a left rule “using “A”. These are the interesting cases, because they show how cut elimination can maintain the balance between right and left rules.

Case: $D = \frac{\Delta_1}{\Delta} \Rightarrow B \Rightarrow R, E = \frac{\Delta}{\Delta_1 \Rightarrow A} \Rightarrow L = \frac{\Delta_1, \Delta_2 \Rightarrow A \Rightarrow B}{\Delta_1, \Delta_2 \Rightarrow C}$

Case: $D = \frac{\Delta_1}{\Delta_2} \Rightarrow A \Rightarrow R, E = \frac{\Delta_1, \Delta_2 \Rightarrow A \Rightarrow B}{\Delta_1, \Delta_2 \Rightarrow C}$

Case: $D = \frac{\Delta_1}{\Rightarrow 1} \Rightarrow R, E = \frac{\Delta \Rightarrow C}{\Gamma; (\Delta, 1) \Rightarrow C}$

immediate

Case: $D = \frac{\Delta_1}{\Rightarrow !A} \Rightarrow R, E = \frac{(\Gamma, A); \Delta \Rightarrow C}{\Gamma; (\Delta, !A) \Rightarrow C}$

Case: $D = \frac{\Delta_1}{\Rightarrow \forall x : \tau \cdot A} \Rightarrow R^{\forall \cdot \tau}, E = \frac{\Delta_1, \forall x : \tau \cdot A}{\Delta_1, \forall x : \tau \cdot A \Rightarrow C}$

The next block of cases are those of commutative conversions. These encompass all of the remaining cases. A commutative conversion is triggered if the current the cut formula is not principal in the left and right derivation. By commuting
the cut, we push the cut rule higher up in the derivation by commuting it with
the second last premiss.

Since all the cases are very similar, we give only two, a so-called right com-
mutative conversion

\[
\text{Case: } \Gamma; \Delta_1 \rightarrow A, \varepsilon = \frac{\varepsilon_1}{\varepsilon_1} = \frac{\Gamma; (\Delta_2, A, B) \rightarrow C}{\Gamma; \Delta_2, A \rightarrow B \rightarrow C} \rightarrow \Gamma
\]

\[
\Gamma; (\Delta_1, \Delta_2, B) \rightarrow C \quad \text{by i.h. 1. on } \Gamma \text{ and } \varepsilon_1
\]
\[
\Gamma; (\Delta_1, \Delta_2) \rightarrow B \rightarrow C \quad \text{by reapplying } \rightarrow \Gamma
\]

and a left commutative conversion.

\[
\text{Case: } D = \frac{\varepsilon_1}{\varepsilon} = \frac{D_1}{\varepsilon_1} = \frac{\Gamma; \Delta_1 \rightarrow A}{\Gamma; \Delta_1 \rightarrow A} \rightarrow L, \frac{D_2}{\varepsilon} = \frac{\Gamma; \Delta_2 \rightarrow C}{\Gamma; \Delta_2 \rightarrow C} \rightarrow L, \Gamma; \Delta_3 \rightarrow D
\]

\[
\Gamma; \Delta_2, \Delta_3, B \rightarrow D \quad \text{by i.h. 1. on } D_2 \text{ and } \varepsilon
\]
\[
\Gamma; \Delta_1, \Delta_2, \Delta_3, A \rightarrow B \rightarrow D \quad \text{by reapplying } \rightarrow L
\]

Finally, we discuss the second part of the theorem. Compared to the first,
we only need to consider one rule, the copy rule;

\[
\text{Case: } D = \frac{\varepsilon_1}{\varepsilon} = \frac{\varepsilon_1}{\varepsilon} = \frac{\Gamma; \Delta \rightarrow C}{\Gamma; \Delta \rightarrow C}
\]

\[
\Gamma; \Delta, A \vdash C \quad \text{by i.h. 2. on } \varepsilon \text{ and } D_1
\]
\[
\Gamma; \Delta \vdash C \quad \text{by subsequent i.h. 1. on } \varepsilon
\]

\[\Box\]

The cut-admissibility theorem is a powerful theorem. We now know how to
remove one cut. It can easily be generalized to an inductive procedure called
cut-elimination on how to eliminate several cuts in a derivation.

Cut-elimination has many applications. First, we can show the soundness of
a logic. If we were to add falsehood 0 to the calculus.

\[
\Gamma; \Delta, 0 \rightarrow C
\]

for which there is no right rule, then we could prove that \(\vdash 0\) \(\rightarrow 0\) is not
derivable via cut-elimination. Therefore, there is at least one formula that has
no derivation and the logic is hence consistent.
Example 10 (First past the post) We give an example of the specification of the first past the post voting protocol in linear logic. Assuming that there are \( n \) ballots and \( k \) candidates, this protocol will determine the winner of the election. We first give the rules and then explain them, assuming that there are three sorts, \( \text{nat} \) denoting natural numbers and \( \text{cand} \) denoting candidates.

1. \( \forall N : \text{nat}. \forall C : \text{cand}. \forall M : \text{nat}. \)
   \[ \text{count}(N) \otimes \text{ballot}(C) \otimes \text{hopeful}(C, M) \Rightarrow \text{hopeful}(C, M + 1) \otimes \text{count}(N - 1) \]

2. \( \forall C_1 : \text{cand}. \forall M_1 : \text{nat}. \forall C_2 : \text{cand}. \forall M_2 : \text{nat}. \)
   \[ \text{count}(0) \otimes \text{hopeful}(C_1, M_1) \otimes \text{hopeful}(C_2, M_2) \otimes !(M_1 < M_2) \]
   \[ \Rightarrow \text{hopeful}(C_2, M_2) \otimes \text{count}(0) \otimes !\text{defeated}(C_1) \]

3. \( \forall C : \text{cand}. \forall M : \text{nat}. \)
   \[ \text{count}(0) \otimes \text{hopeful}(C, M) \Rightarrow !\text{elected}(C) \]

A candidate is elected if and only if the following sequent is provable. Let \( \Gamma \) consist of the three rules.

\[
\Gamma; \Rightarrow \text{ballot}(C_1) \otimes \ldots \otimes \text{ballot}(C_n) \\
\otimes \text{hopeful}(C_1, 0) \otimes \ldots \otimes \text{hopeful}(C_k, 0) \otimes \text{count}(n) \\
\Rightarrow !\text{elected}(C)
\]

to figure out who is elected, we need to run a theorem prover. And thus we need to determine a good strategy on how to apply the rules. This is what we are going to do in the next lecture.

References


