cse541
LOGIC FOR COMPUTER SCIENCE

Professor Anita Wasilewska

Spring 2015
LECTURE 10a
Chapter 10
CLASSICAL AUTOMATED PROOF SYSTEMS

PART 3: GENTZEN SYSTEMS
Gentzen Sequent Calculus GL

The proof system GL for the classical propositional logic presented now is a version of the original Gentzen (1934) systems LK.

A constructive proof of the Completeness Theorem for the system GL is very similar to the proof of the Completeness Theorem for the system RS.

Expressions of the system are like in the original Gentzen system LK are Gentzen sequents.

Hence we use also a name Gentzen sequent calculus for it.
Gentzen Sequent Calculus GL

Language of GL

\[ \mathcal{L} = \mathcal{L}\{\cup, \cap, \Rightarrow, \neg\} \]

We add a new symbol to the alphabet \( \rightarrow \) called a Gentzen arrow

The sequents are built out of finite sequences (empty included) of formulas, i.e. elements of \( \mathcal{F}^* \), and the additional symbol \( \rightarrow \)

We denote, as in the RS system, the finite sequences of formulas by Greek capital letters

\[ \Gamma, \Delta, \Sigma, \ldots \]

with indices if necessary
Gentzen Sequents

Definition Any expression

\[ \Gamma \rightarrow \Delta \]

where \( \Gamma, \Delta \in \mathcal{F}^* \) is called a sequent

Intuitively, we interpret semantically a sequent

\[ A_1, ..., A_n \rightarrow B_1, ..., B_m \]

where \( n, m \geq 1 \), as a formula

\[ (A_1 \cap ... \cap A_n) \Rightarrow (B_1 \cup ... \cup B_m) \]
Gentzen Sequents

The sequent

\[ A_1, \ldots, A_n \rightarrow \]

(where \( m \geq 1 \)) means that \( A_1 \cap \ldots \cap A_n \) yields a contradiction

The sequent

\[ \rightarrow B_1, \ldots, B_m \]

(where \( m \geq 1 \)) means semantically \( T \Rightarrow (B_1 \cup \ldots \cup B_m) \)

The empty sequent

\[ \rightarrow \]

means a contradiction
Gentzen Sequents

Given non empty sequences $\Gamma$, $\Delta$

We denote by $\sigma_\Gamma$ any conjunction of all formulas of $\Gamma$

We denote by $\delta_\Delta$ any disjunction of all formulas of $\Delta$

The intuitive semantics of a non-empty sequent $\Gamma \rightarrow \Delta$ is

$$\Gamma \rightarrow \Delta \equiv (\sigma_\Gamma \Rightarrow \delta_\Delta)$$
Formal Semantics

**Formal semantics** for sequents of GL is defined as follows:

Let \( v : \text{VAR} \rightarrow \{T, F\} \) be a truth assignment and \( v^* \) its extension to the set of formulas \( \mathcal{F} \).

We *extend* \( v^* \) to the set

\[
\text{SQ} = \{ \Gamma \rightarrow \Delta : \Gamma, \Delta \in \mathcal{F}^* \}
\]

of all sequents as follows:

For any sequent \( \Gamma \rightarrow \Delta \in \text{SQ} \),

\[
v^*(\Gamma \rightarrow \Delta) = v^*(\sigma_\Gamma) \Rightarrow v^*(\delta_\Delta)
\]
Formal Semantics

In the case when $\Gamma = \emptyset$ or $\Delta = \emptyset$ we define

$$v^*(\Gamma \rightarrow \Delta) = (T \Rightarrow v^*(\delta_\Delta))$$

$$v^*(\Gamma \rightarrow ) = (v^*(\sigma_\Gamma) \Rightarrow F)$$

The sequent $\Gamma \rightarrow \Delta$ is **satisfiable** if there is a truth assignment $v : \text{VAR} \rightarrow \{T, F\}$ such that

$$v^*(\Gamma \rightarrow \Delta) = T$$
Formal Semantics

**Model** for $\Gamma \rightarrow \Delta$ is any $v$ such that

$$v^*(\Gamma \rightarrow \Delta) = T$$

We write it $v \models \Gamma \rightarrow \Delta$

**Counter-model** is any $v$ such that

$$v^*(\Gamma \rightarrow \Delta) = F$$

We write it $v \not\models \Gamma \rightarrow \Delta$

**Tautology** is any sequent $\Gamma \rightarrow \Delta$ such that

$$v^*(\Gamma \rightarrow \Delta) = T$$ for all truth assignments $v : VAR \rightarrow \{T, F\}$

We write it

$$\models \Gamma \rightarrow \Delta$$
Example

Example
Let $\Gamma \rightarrow \Delta$ be a sequent

$$a, (b \cap a) \rightarrow \neg b, (b \Rightarrow a)$$

The truth assignment $v$ for which $v(a) = T$ and $v(b) = T$

is a model for $\Gamma \rightarrow \Delta$ as shows the following computation

$v^*(a, (b \cap a) \rightarrow \neg b, (b \Rightarrow a)) = v^*(\sigma_{\{a, (b\cap a)\}}) \Rightarrow v^*(\delta_{\{-b, (b\Rightarrow a)\}})$

$$= v(a) \cap (v(b) \cap v(a)) \Rightarrow \neg v(b) \cup (v(b) \Rightarrow v(a))$$

$$= T \cap T \cap T \Rightarrow \neg T \cup (T \Rightarrow T) = T \Rightarrow (F \cup T) = T \Rightarrow T = T$$
Example

**Observe** that the truth assignment $v$ for which

$$v(a) = T \text{ and } v(b) = T$$

is the **only one** for which

$$v^*(\Gamma) = v^*(a, (b \cap a)) = T$$

and we proved that it is a **model** for

$$a, (b \cap a) \rightarrow \neg b, (b \Rightarrow a)$$

It is hence **impossible** to find $v$ which would **falsify it**, what proves that

$$\models a, (b \cap a) \rightarrow \neg b, (b \Rightarrow a)$$
Gentzen System \textbf{GL}

\textbf{Definition of GL}

\textbf{Logical Axioms LA}

We adopt as an \textit{axiom} any sequent of \textit{variables (positive literals)} which contains a propositional variable that appears on \textit{both sides} of the sequent arrow $\rightarrow$, i.e any sequent of the form

$$\Gamma', a, \Gamma' \rightarrow \Delta', a, \Delta'$$

for any $a \in VAR$ and any sequences $\Gamma', \Gamma', \Delta', \Delta' \in VAR^*$
Gentzen System GL

Inference rules of GL
Let $\Gamma', \Delta' \in \text{VAR}^*$ and $\Gamma, \Delta \in \mathcal{F}^*$

Conjunction rules

\[
(\cap \rightarrow) \quad \begin{array}{c}
\Gamma', A, B, \Gamma \rightarrow \Delta' \\
\Gamma', (A \cap B), \Gamma \rightarrow \Delta'
\end{array}
\]

\[
(\rightarrow \cap) \quad \begin{array}{c}
\Gamma \rightarrow \Delta, A, \Delta' ; \Gamma \rightarrow \Delta, B, \Delta'
\end{array}
\]

\[
\Gamma \rightarrow \Delta, (A \cap B) \Delta'
\]
Gentzen System **GL**

**Disjunction rules**

\[(→ ∪) \quad \frac{\Gamma \rightarrow \Delta, A, B, \Delta'}{\Gamma \rightarrow \Delta, (A ∪ B), \Delta'}\]

\[(∪ →) \quad \frac{\Gamma', A, \Gamma \rightarrow \Delta'; \Gamma', B, \Gamma \rightarrow \Delta'}{\Gamma', (A ∪ B), \Gamma \rightarrow \Delta'}\]
Gentzen System GL

Implication rules

\[(\rightarrow\rightarrow)\]

\[
\frac{\Gamma', A, \Gamma \rightarrow \Delta, B, \Delta'}{\Gamma', \Gamma \rightarrow \Delta, (A \Rightarrow B), \Delta'}
\]

\[(\Rightarrow\Rightarrow)\]

\[
\frac{\Gamma', \Gamma \rightarrow \Delta, A, \Delta'}{\Gamma', (A \Rightarrow B), \Gamma \rightarrow \Delta, \Delta'}
\]
Gentzen System GL

Negation rules

\[(\neg \rightarrow) \quad \Gamma', \Gamma \rightarrow \Delta, A, \Delta' \]
\[
\frac{\Gamma', \neg A, \Gamma \rightarrow \Delta, \Delta'}{\Gamma', \Gamma \rightarrow \Delta, A, \Delta'}
\]

\[(\rightarrow \neg) \quad \Gamma', A, \Gamma \rightarrow \Delta, \Delta' \]
\[
\frac{\Gamma', \Gamma \rightarrow \Delta, \neg A, \Delta'}{\Gamma', \Gamma \rightarrow \Delta, \Delta'}
\]
Gentzen System $\textbf{GL}$

We define the Gentzen System $\textbf{GL}$

\[ \textbf{GL} = (\mathcal{L}_{\{\cup, \cap, \Rightarrow, \neg}\}, \ SQ, \ LA, \ \mathcal{R}) \]

for

\[ \mathcal{R} = \{(\cap \rightarrow), (\rightarrow \cap), (\cup \rightarrow), (\rightarrow \cup), (\Rightarrow \rightarrow), (\rightarrow \Rightarrow)\} \]

\[ \cup\{(\neg \rightarrow), (\rightarrow \neg)\} \]

We write, as usual,

\[ \vdash_{\textbf{GL}} \Gamma \rightarrow \Delta \]

to denote that $\Gamma \rightarrow \Delta$ has a formal proof in $\textbf{GL}$

A formula $A \in \mathcal{F}$, has a proof in $\textbf{GL}$ if the sequent $\rightarrow A$ has a proof in $\textbf{GL}$, i.e.

\[ \vdash_{\textbf{GL}} A \quad \text{if ad only if} \quad \rightarrow A \]
We consider, as we did with RS the proof trees for GL, i.e. we define
A proof tree, or GL-proof of $\Gamma \rightarrow \Delta$ is a tree $T_{\Gamma \rightarrow \Delta}$
of sequents satisfying the following conditions:
1. The topmost sequent, i.e the root of $T_{\Gamma \rightarrow \Delta}$ is $\Gamma \rightarrow \Delta$
2. All leafs are axioms
3. The nodes are sequents such that each sequent on the tree follows from the ones immediately preceding it by one of the rules.
Gentzen System **GL**

**Exercise 1**

We define, in a similar way as in **RS** the **GL** the notions of decomposable and indecomposable sequences, the decomposition rules and the decomposition tree

**Remark**

The proof search in **GL** as defined by the decomposition tree for a given formula $A$ is not always **unique**

We show it on an example on the next slide
Example

A tree-proof in $\textbf{GL}$ of the de Morgan Law

\[
\rightarrow (\neg(a \land b) \Rightarrow (\neg a \lor \neg b))
\]

\[
| (\rightarrow \Rightarrow)
\]

\[
\neg(a \land b) \rightarrow (\neg a \lor \neg b)
\]

\[
| (\rightarrow \lor)
\]

\[
\neg(a \land b) \rightarrow \neg a, \neg b
\]

\[
| (\rightarrow \neg)
\]

\[
b, \neg(a \land b) \rightarrow \neg a
\]

\[
| (\neg \rightarrow)
\]

\[
b, a, \neg(a \land b) \rightarrow
\]

\[
| (\neg \rightarrow)
\]

\[
b, a \rightarrow (a \land b)
\]

\[
\land (\rightarrow \land)
\]

\[
b, a \rightarrow a \quad b, a \rightarrow b
\]
Example

Here is another tree-proof in GL of the de Morgan Law

\[ \rightarrow (\neg(a \land b) \Rightarrow (\neg a \lor \neg b)) \]

\[ \mid (\rightarrow \Rightarrow) \]
\[ \neg(a \land b) \rightarrow (\neg a \lor \neg b) \]
\[ \mid (\rightarrow \lor) \]
\[ \neg(a \land b) \rightarrow \neg a, \neg b \]
\[ \mid (\rightarrow \neg) \]
\[ b, \neg(a \land b) \rightarrow \neg a \]
\[ \mid (\neg \rightarrow) \]
\[ b \rightarrow \neg a, (a \land b) \]
\[ \land(\rightarrow \land) \]

\[ b \rightarrow \neg a, a \]
\[ b \rightarrow \neg a, b \]
\[ \mid (\rightarrow \neg) \]
\[ b, a \rightarrow a \]
\[ b, a \rightarrow b \]
Exercises

Exercise 2
Write all other proofs in GL of

\((\neg(a \cap b) \Rightarrow (\neg a \cup \neg b))\)

Exercise 3
Find a formula which has a unique decomposition tree

Exercise 4
Describe for which kind of formulas the decomposition tree is unique

Exercise 5
Formulate and prove a Decomposition Tree Theorem for GL that corresponds to the similar theorem for RS
Strong Soundness and Completeness of \( \text{GL} \)

Exercise 6

Prove, in a similar way as we did in the case of \( \text{RS} \) type systems, the **Strong Soundness** of \( \text{GL} \)

Exercise 7

Prove, in a similar way as we did in the case of \( \text{RS} \), the **Completeness Theorem** for \( \text{GL} \), i.e.

**Completeness Theorem**

For any sequent \( \Gamma \rightarrow \Delta \in SQ \)

\[ \vdash_{\text{GL}} \Gamma \rightarrow \Delta \quad \text{if and only if} \quad \models \Gamma \rightarrow \Delta \]