From Axioms to Rules:
The Factory of Modal Proof Systems

3. Lecture

First-Order Predicate Logic

Sonia Marin and Lutz Straßburger

Formulas of First-Order Logic

Terms and *formulas*:

$$t \;::=\; x \mid f(t_1, \ldots, t_m)$$
$$A, B \;::=\; \top \mid \bot \mid p(t_1, \ldots, t_n) \mid A \wedge B \mid A \vee B \mid A \supset B \mid \neg A \mid \forall x. A \mid \exists x. A$$

where

- x is a *first-order variable*
- f is an *m-ary function symbol*
- p is an *n-ary predicate symbol*
- $\forall x. A$ is read as *"for all x, we have A"*
- $\exists x. A$ is read as *"there exists a x, such that A"*
- $\forall$ and $\exists$ are called *quantifiers*
  they *bind* the first-order variable x.

- In the language od first-order logic, we have
  - a countable set $\mathcal{X} = \{x, y, z, \ldots\}$ of first-order variables,
  - a countable set $\mathcal{F} = \{f, g, \ldots\}$ of m-ary function symbols (for each $m \geq 0$), and
  - a countable set $\mathcal{P} = \{p, q, \ldots\}$ of n-ary predicate symbols (for each $n \geq 0$).

Free and Bound Variables and Substitution

Definition: The *free* and *bound* variable occurrences in a formula are defined inductively as follows.

- x occurs *free* in an atomic formula A iff x occurs in A.
  There are *no bound* variables in any atomic formula.
- x occurs *free* in $\neg A$ iff x occurs free in A.
  x occurs bound in $\neg A$ iff x occurs bound in A.
- x occurs *free* in $A \wedge B$, $A \vee B$, $A \supset B$ iff x occurs free in A or in B.
  x occurs *bound* in $A \wedge B$, $A \vee B$, $A \supset B$ iff x occurs bound in A or in B.
- x occurs *free* in $\forall y. A$ or $\exists y. A$, iff x occurs free in A and x is a different variable symbol from y.
  x occurs *bound* in $\forall y. A$ or $\exists y. A$, iff x is y or x occurs bound in A.

Substitution:
$A[x/t]$ is the result of replacing all free occurrences of x in A by t.

- In a formula, a variable may occur free or bound (or both). A variable occurrence is *bound* in a formula if it is quantified. Otherwise it is *free*.
- **Exercise 3.1:** What are the free and bound variables in the following formulas:
  - $\forall x. \forall y. (p(x) \supset q(x, f(x), z))$
  - $p(x) \supset \forall x. q(x)$
- **Exercise 3.2:** Let $A = p(x) \supset \forall x. q(x)$. What is $A[x/f(y)]$?

Axioms for First-Order Logic

- Axioms for propositional logic (for $\supset, \bot$):
  - $A \supset (B \supset A)$
  - $(A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C))$
  - $((A \supset \bot) \supset \bot) \supset A$
- Axioms for $\forall$:
  - $(\forall x.\,(A \supset B)) \supset (A \supset \forall x.\,B)$ (provided x is not free in A)
  - $(\forall x.\,A) \supset A[x/t]$
- The other connectives are defined via $\bot, \supset, \forall$:
  - $\neg A \equiv A \supset \bot$ $\quad$ $\exists x.\,A \equiv \neg\forall x.\,(\neg A)$
  - $A \vee B \equiv \neg A \supset B$ $\quad$ $A \wedge B \equiv \neg(\neg A \vee \neg B)$
- Inference rules:

$$\text{mp}\ \frac{A \qquad A \supset B}{B} \qquad\qquad \text{gen}\ \frac{A}{\forall x.\,A}$$

Definition: A formula is *provable* (or *a theorem*) if it is either (a substitution instance of) an axiom, or can be derived via an instance of a rule mp or gen from provable formulas.

- As for propositional logic, there are many different Hilbert-style systems for first order logic.
- We picked here one that defines only $\forall$, to keep the list of axioms short. It can be found, for example, in
  - *Chang, C.C. and Keisler, H.J.:* **"Model Theory".** *North-Holland, Amsterdam, 1973*
- **Exercise 3.3:** Can you prove $\exists x.\,\forall y.\,p(x) \supset p(y)$?
- **Exercise 3.4:** The formula above is also called the *Drinker's formula*. Can you figure out why?

Semantics for First-Order Logic

Definition: A *model* $\mathcal{M}$ of first order logic is a non-empty set $\mathcal{D}$ (the *domain* of $\mathcal{M}$) together with an *interpretation* $[\![\cdot]\!]$:

- for each m-ary function symbol f:
  a function $[\![f]\!] : \mathcal{D}^m \to \mathcal{D}$
- for each n-ary predicate symbol p:
  a function $[\![p]\!] : \mathcal{D}^n \to \{\text{true}, \text{false}\}$

Definition: Given a model $\mathcal{M}$, a *variable assignment* μ is a mapping assigning each variable x to an element $\mu(x) \in \mathcal{D}$.

Semantics for First-Order Logic (cont.)

Given a model $\mathcal{M}$ and a variable assignment μ, we can extend the interpretation $[\![\cdot]\!]$ to all terms:

- $[\![x]\!] = \mu(x)$
- $[\![f(t_1, \ldots, t_m)]\!] = [\![f]\!]([\![t_1]\!], \ldots, [\![t_m]\!])$

and to all formulas:

- $[\![\bot]\!] = \text{false}$
- $[\![\top]\!] = \text{true}$
- $[\![p(t_1, \ldots, t_n)]\!] = [\![p]\!]([\![t_1]\!], \ldots, [\![t_n]\!])$
- $[\![A \wedge B]\!]$, $[\![A \vee B]\!]$, $[\![A \supset B]\!]$, $[\![\neg A]\!]$
  as in truth tables for propositional logic
- $[\![\forall x.\,A]\!] = \text{true}$ iff $[\![A]\!] = \text{true}$ for every variable assignment μ' with $\mu'(y) = \mu(y)$ for all variable symbols y different from x.
- $[\![\exists x.\,A]\!] = \text{true}$ iff there is a variable assignment μ' with $\mu'(y) = \mu(y)$ for all variable symbols y different from x, such that $[\![A]\!] = \text{true}$.

- The definition of $[\![\forall x.\,A]\!]$ captures the idea that $\forall x.\,A$ is true if every possible choice of a value for x causes A to be true.
- The definition of $[\![\exists x.\,A]\!]$ captures the idea that $\exists x.\,A$ is true if there is a possible choice of a value for x such that A is true.

Soundness and Completeness for First-Order Logic

We write $\mathcal{M}, \mu \vDash A$ iff $[\![A]\!] =$ true for model $\mathcal{M}$ and assigment μ.

Definition: A formula A is *satisfiable* if there is a model $\mathcal{M}$ and a variable assigment μ, such that $\mathcal{M}, \mu \vDash A$.

Definition: A formula A is *valid* iff $\mathcal{M}, \mu \vDash A$ for all models $\mathcal{M}$ and variable assigments μ.

Theorem: *(Soundness)* Every formula that is provable is also valid.

Theorem: *(Completeness)* Every formula that is valid is also provable.

- **Exercise 3.5:** Show that the formula $\forall x. \forall y. p(x) \wedge p(y)$ is not valid.
- **Exercise 3.6:** Show that $\forall x. p(x) \vee \neg p(x)$ is valid.
- **Exercise 3.7:** Is $\forall x. \forall y. p(x) \vee p(y)$ valid?
- **Exercise 3.8:** What about $\forall x. \forall y. p(x) \vee \neg p(y)$?

For the exercises above work only with the models (don't use Soundness/Completeness).

- **Exercise 3.9:** Prove soundness.
- Completeness for first-order logic is more involved than for modal logics. It has first been shown by Gödel in his PhD thesis:
  - *Kurt Gödel:* **"Die Vollständigkeit der Axiome des logischen Funktionenkalküls".** *Monatshefte für Mathematik und Physik 37, 1930, p.349–360*

Sequent Calculus for First-Order Logic

- *Sequents:*

$$A_1, \ldots, A_m \vdash B_1, \ldots, B_n$$

- *Corresponding Formula:*

$$(A_1 \wedge \cdots \wedge A_m) \supset (B_1 \vee \cdots \vee B_n)$$

- if $m = 0$ then the corresponding formula is

$$B_1 \vee \cdots \vee B_n \qquad \text{or} \qquad \top \supset (B_1 \vee \cdots \vee B_n)$$

- if $n = 0$ then the corresponding formula is

$$(A_1 \wedge \cdots \wedge A_m) \supset \bot \qquad \text{or} \qquad \bar{A}_1 \vee \ldots \vee \bar{A}_n$$

- if $m = 0$ and $n = 0$ then the corresponding formula is *falsum*:

$$\top \supset \bot \qquad \text{or} \qquad \bot$$

- A *sequent* is essentially a pair of finite lists of formulas. This is how Gentzen introduced them:
  - *Gerhard Gentzen:* **"Untersuchungen über das logische Schließen. I".** *Mathematische Zeitschrift (39), 1935, p.176–210*
- Nowadays authors also often use multisets (the order of the formulas is irrelevant) or sets (also the number of occurrences of a formula is irrelevant).

Sequent Calculus for First-Order Logic

- *Initial sequents / Axioms:*

$$\bot\ \frac{}{\bot \vdash} \qquad \mathsf{id}\ \frac{}{A \vdash A} \qquad \top\ \frac{}{\vdash \top}$$

- *Structural rules:*

$$\mathsf{weak_L}\ \frac{\Gamma \vdash \Theta}{A, \Gamma \vdash \Theta} \qquad \mathsf{weak_R}\ \frac{\Gamma \vdash \Theta}{\Gamma \vdash \Theta, A}$$

$$\mathsf{con_L}\ \frac{A, A, \Gamma \vdash \Theta}{A, \Gamma \vdash \Theta} \qquad \mathsf{con_R}\ \frac{\Gamma \vdash \Theta, A, A}{\Gamma \vdash \Theta, A}$$

$$\mathsf{exch_L}\ \frac{\Delta, B, A, \Gamma \vdash \Theta}{\Delta, A, B, \Gamma \vdash \Theta} \qquad \mathsf{exch_R}\ \frac{\Gamma \vdash \Theta, A, B, \Lambda}{\Gamma \vdash \Theta, A, B, \Lambda}$$

- *Cut:*

$$\mathsf{cut}\ \frac{\Gamma \vdash \Theta, A \qquad A, \Delta \vdash \Lambda}{\Gamma, \Delta \vdash \Theta, \Lambda}$$

- We use Capital Greek letters, like $\Gamma, \Delta, \Theta, \Lambda, \ldots$ to denote finite lists of formulas.
- The structural rules are called like this because they work on the structure of the sequent. When multisets are used instead of lists, the *exchange* rules $\mathsf{exch_L}$ and $\mathsf{exch_R}$ are not needed. And when sets are used instead, the rules for *weakening* ($\mathsf{weak_L}$ and $\mathsf{weak_R}$) and *contraction* ($\mathsf{con_L}$ and $\mathsf{con_R}$) are not needed.
- However, lists are used in order to have more control over the structure of proofs.
- The *cut* rule can be seen as generalisation of *modus ponens*. The formula A in that rule is also called the *cut formula*.

Sequent Calculus for First-Order Logic

- *Logical rules:*

$$\vee_L\ \frac{A,\Gamma\vdash\Theta \qquad B,\Gamma\vdash\Theta}{A\vee B,\Gamma\vdash\Theta} \qquad \vee_R\ \frac{\Gamma\vdash\Theta,A}{\Gamma\vdash\Theta,A\vee B} \quad \vee_R\ \frac{\Gamma\vdash\Theta,B}{\Gamma\vdash\Theta,A\vee B}$$

$$\wedge_L\ \frac{A,\Gamma\vdash\Theta}{A\wedge B,\Gamma\vdash\Theta} \quad \wedge_L\ \frac{B,\Gamma\vdash\Theta}{A\wedge B,\Gamma\vdash\Theta} \qquad \wedge_R\ \frac{\Gamma\vdash\Theta,A \qquad \Gamma\vdash\Theta,B}{\Gamma\vdash\Theta,A\wedge B}$$

$$\supset_L\ \frac{\Gamma\vdash\Theta,A \qquad B,\Gamma\vdash\Theta}{A\supset B,\Gamma\vdash\Theta} \qquad \supset_R\ \frac{A,\Gamma\vdash\Theta,B}{\Gamma\vdash\Theta,A\supset B}$$

$$\neg_L\ \frac{\Gamma\vdash\Theta,A}{\neg A,\Gamma\vdash\Theta} \qquad \neg_R\ \frac{A,\Gamma\vdash\Theta}{\Gamma\vdash\Theta,\neg A}$$

$$\forall_L\ \frac{A[x/t],\Gamma\vdash\Theta}{\forall x.\,A,\Gamma\vdash\Theta} \qquad \forall_R\ \frac{\Gamma\vdash\Theta,A}{\Gamma\vdash\Theta,\forall x.\,A}\ x \text{ not free in } \Gamma,\Theta$$

$$\exists_L\ \frac{A,\Gamma\vdash\Theta}{\exists x.\,A,\Gamma\vdash\Theta}\ x \text{ not free in } \Gamma,\Theta \qquad \exists_R\ \frac{\Gamma\vdash\Theta,A[x/t]}{\Gamma\vdash\Theta,\exists x.\,A}$$

- The side condition that x must not be free in Γ and Θ in the rules $\forall_R$ and $\exists_L$ is important to avoid *variable capturing*. In these two rules, the variable x is called *Eigenvariable*.
- In the rules $\forall_L$ and $\exists_R$, the term t is arbitrary. It might or might not contain x.
- Gentzen called this calculus LK.
- Observe that while in Hilbert systems there are many axioms and only a few rules, in the sequent calculus we have only a few (trivial) axioms and many rules. This is what allows us to eventually get more control over the structure of proofs.
- **Exercise 3.10:** Are the following fromulas provable in the sequent calculus?
  - $\forall x.\,p(x)\vee\neg p(x)$
  - $\forall x.\,\forall y.\,p(x)\vee p(y)$
  - $\forall x.\,\forall y.\,p(x)\vee\neg p(y)$
  - $\exists x.\,\forall y.\,p(x)\vee\neg p(y)$
  - $\forall x.\,\exists y.\,p(x)\vee\neg p(y)$

Soundness and Completeness of the Sequent Calculus

- **Definition:** A formula A is *provable* in LK if the sequent $\vdash A$ is derivable with the rules shown in the previous two slides.
- **Definition:** A formula A is *cut-free provable* in LK if the sequent $\vdash A$ is derivable without the use of the cut-rule.
- **Theorem:** For every formula A, the following are equivalent:
  1. A is a theorem of first-order logic.
  2. A is valid.
  3. A is provable in LK.
  4. A is cut-free provable in LK.

- **Exercise 3.11:** Prove $1 \Longrightarrow 3$.
- **Exercise 3.12:** Prove $3 \Longrightarrow 2$.
- The implication $3 \Longrightarrow 4$ is Gentzen's contribution, proved in
  - *Gerhard Gentzen:* **"Untersuchungen über das logische Schließen. I".** *Mathematische Zeitschrift (39), 1935, p.176–210*

  ($4 \Longrightarrow 3$ is trivial.)
- With this theorem, completess is easier to show, as it is shown via contrapositive: $\neg 4 \Longrightarrow \neg 2$ is simpler than $\neg 3 \Longrightarrow \neg 2$ or $\neg 1 \Longrightarrow \neg 2$.
- The implication $3 \Longrightarrow 4$ is also known as *cut elimination*.

Cut elimination

- Basic idea: permute instances of cut upwards in the proof until they meet an axiom.
  - *commutation cases:*

$$\text{cut}\ \frac{\text{r}\ \dfrac{\Gamma'\vdash\Theta',A}{\Gamma\vdash\Theta,A} \qquad A,\Delta\vdash\Lambda}{\Gamma,\Delta\vdash\Theta,\Lambda} \quad\rightsquigarrow\quad \text{r}\ \frac{\text{cut}\ \dfrac{\Gamma'\vdash\Theta',A \qquad A,\Delta\vdash\Lambda}{\Gamma',\Delta\vdash\Theta',\Lambda}}{\Gamma,\Delta\vdash\Theta,\Lambda}$$

  - *key cases:*

$$\text{cut}\ \frac{\wedge_R\ \dfrac{\Gamma\vdash\Theta,A \qquad \Gamma\vdash\Theta,B}{\Gamma\vdash\Theta,A\wedge B} \qquad \wedge_L\ \dfrac{A,\Delta\vdash\Lambda}{A\wedge B,\Delta\vdash\Lambda}}{\Gamma,\Delta\vdash\Theta,\Lambda}$$

$$\rightsquigarrow\quad \text{cut}\ \frac{\Gamma\vdash\Theta,A \qquad A,\Delta\vdash\Lambda}{\Gamma,\Delta\vdash\Theta,\Lambda}$$

- **Exercise 3.13:** Write down all key cases.
- **Exercise 3.14:** What would be the base case?

Cut elimination (cont.)

- cut meets structural rules:
  - *cases for weakening:*

$$
\text{cut}\;\dfrac{\text{weak}_R\;\dfrac{\Gamma \vdash \Theta}{\Gamma \vdash \Theta, A} \qquad A, \Delta \vdash \Lambda}{\Gamma, \Delta \vdash \Theta, \Lambda}
\quad\rightsquigarrow\quad
\text{weak}_L\;\dfrac{\text{weak}_R\;\dfrac{\Gamma \vdash \Theta}{\Gamma \vdash \Theta, \Lambda}}{\Gamma, \Delta \vdash \Theta, \Lambda}
$$

  - *cases for contraction:*

$$
\text{cut}\;\dfrac{\text{con}_R\;\dfrac{\Gamma \vdash \Theta, A, A}{\Gamma \vdash \Theta, A} \qquad A, \Delta \vdash \Lambda}{\Gamma, \Delta \vdash \Theta, \Lambda}
\quad\rightsquigarrow\quad
\text{con}_L\;\dfrac{\text{con}_R\;\dfrac{\text{cut}\;\dfrac{\text{exch}_R\;\dfrac{\text{cut}\;\dfrac{\Gamma \vdash \Theta, A, A \qquad A, \Delta \vdash \Lambda}{\Gamma, \Delta \vdash \Theta, A, \Lambda}}{\Gamma, \Delta \vdash \Theta, \Lambda, A} \qquad A, \Delta \vdash \Lambda}{\Gamma, \Delta, \Delta \vdash \Theta, \Lambda, \Lambda}}{\Gamma, \Delta, \Delta \vdash \Theta, \Lambda}}{\Gamma, \Delta \vdash \Theta, \Lambda}
$$

- Because of the contraction cases, this naive cut elimination process is not terminating.
- **Exercise 3.15:** To see this, write down the reduction for this case:

$$
\text{cut}\;\dfrac{\text{con}_R\;\dfrac{\Gamma \vdash \Theta, A, A}{\Gamma \vdash \Theta, A} \qquad \text{con}_L\;\dfrac{A, A, \Delta \vdash \Lambda}{A, \Delta \vdash \Lambda}}{\Gamma, \Delta \vdash \Theta, \Lambda}
$$

Cut elimination (cont.)

- *the super-cut rule:*

$$
\text{scut}\;\dfrac{\Gamma \vdash \Theta' \qquad \Delta' \vdash \Lambda}{\Gamma, \Delta \vdash \Theta, \Lambda}
$$

  where
  - Θ' and Δ' are list of formulas that both contain at least once the cut formula A.
  - Θ and Δ are obtained from Θ' and Δ', respectively, by removing all occurrences of the cut formula A.

- Note that the cut rule is a special case of the scut rule.
- The scut rule and its variations have many different names in the literature. Gentzen called it *Mischung*. Therefore it is often called *Mix* or *Merge*. Very common is also *multi-cut* because multiple occurrences of the cut formula occur in the premises.
- **Exercise 3.16:** Write down the super-cut reduction cases for the structural rules. What do you observe?
- **Exercise 3.17:** Write down the key cases for super-cut. What do you observe?

Undecidability

- Most modal logics are *decidable*:
  We can implement a terminating *proof search* using cut-free sequent calculus.
- First-order logic is *undecidable*:
  Neither cut-free sequent calculus nor any other complete proof system can give us a terminating *proof search*.