Completeness: literature

Completeness proofs for propositional logic are in

- “Logic in Computer Science” by Huth and Ryan, Chapter 1.
- “Logic and Structure” by van Dalen, Chapter 1.

The two proofs differ. The proof presented in this course is a slightly modified version of van Dalen’s.

About maximally consistent sets

Lemma. Every consistent set Γ is contained in a maximally consistent set Γ^*.

Proof of lemma (part 1/2)

Proof. Let $A_0, A_1, A_2, \ldots$ be an enumeration of all formulæ. We define a sequence $\Gamma_0, \Gamma_1, \Gamma_2, \ldots$ of sets of formulæ such that the union is maximally consistent:

$$\Gamma_0 = \Gamma$$

$$\Gamma_{n+1} = \begin{cases} \Gamma \cup \{A_n\} & \text{if } \Gamma \cup \{A_n\} \text{ is consistent.} \\ \Gamma & \text{otherwise} \end{cases}$$

$$\Gamma^* = \bigcup \{\Gamma : n \geq 0\}$$

Proof of lemma (part 2/2)

1. All Γ_n are consistent: this follows immediately from induction on n.
2. Γ^* is consistent: suppose not, i.e. $\Gamma^* \vdash \bot$. The proof of $\bot$ needs only finitely many assumptions from Γ^*, so we have $\Gamma_n \rightarrow \bot$ for some n. But this is impossible because of (1).
3. Γ^* is **maximally** consistent: suppose not, i.e. $\Gamma^* \cup \{B\}$ is consistent for some $B \notin \Gamma^*$. We have $B = A_n$ for some n, and $A_n \in \Gamma_{n+1} \subseteq \Gamma^*$. Contradiction!

Q.e.d.

The Model Existence Lemma

To prove completeness, it remains to prove the Model Existence Lemma.

Lemma. Every consistent set Γ of formulæ has a model.

Proof. Blackboard or van Dalen.

This concludes the completeness proof for propositional logic.

Proof of MEL (part 1/3)

Proof. By earlier lemma, Γ is contained in a maximally consistent Γ^*. Define a situation M be letting

$$[\![p]\!]_M = \begin{cases} 1 & \text{if } p \in \Gamma^* \\ 0 & \text{otherwise} \end{cases} .$$

Now we prove by induction on the size of A that

$$A \in \Gamma^* \text{ if and only if } M \models A.$$

Proof of MEL (part 2/3)

- $A = p$: by definition of M, we have $[\![p]\!]_M = 1$, and therefore $M \models p$.
- $A = \bot$: the formula A is never in Γ^* because Γ^* is consistent, and M is never a model of $\bot$.
- $A = B \wedge C$:

$$\begin{array}{lll} A \in \Gamma^* & \text{iff } B \in \Gamma^* \text{ and } C \in \Gamma^* & \text{(by } \wedge e \text{ and } \wedge i\text{)} \\ & \text{iff } M \models B \text{ and } M \models C & \text{(ind. hyp.)} \\ & \text{iff } M \models B \wedge C & \text{(by definition of } \models\text{).} \end{array}$$

Proof of MEL (part 3/3)

- $A = B \rightarrow C$:

$$\begin{array}{lll} A \in \Gamma^* & \text{iff } B \in \Gamma^* \text{ implies } C \in \Gamma^* & \text{(previous lemma)} \\ & \text{iff } M \models B \text{ implies } M \models C & \text{(ind. hyp.)} \\ & \text{iff } M \models B \rightarrow C & \text{(by definition of } \models\text{).} \end{array}$$

Here ends the induction proof of

$$A \in \Gamma^* \quad \text{iff} \quad M \models A.$$

In particular, it follows that M is a model of Γ^*, and therefore of Γ. This concludes the proof of the Model Existence Lemma, and thereby the completeness proof.

Natural deduction for $\vee$

- Recall that we decided to not include $\vee$ into the language of formulæ, because

$$A \vee B = \neg(\neg A \wedge \neg B).$$

- Still, it is good to know the introduction and elimination rules for $\vee$.

$\vee$-introduction

$$\frac{A_1}{A_1 \vee A_2} \vee i \qquad \frac{A_2}{A_1 \vee A_2} \vee i$$

The soundness of these rules is evident.

$\vee$-elimination

The version without explicit assumptions is

$$\frac{A \vee B \quad \begin{array}{c}[A]\\ \vdots \\ C\end{array} \quad \begin{array}{c}[B]\\ \vdots \\ C\end{array}}{C} \vee e.$$

Intuitively,

$$\frac{\begin{array}{c}\text{everything is an } A \text{ or a } B\\ \text{every } A \text{ is a } C\\ \text{every } B \text{ is a } C\end{array}}{\text{everything is a } C}$$

$\vee$-elimination

The version with explicit assumptions is

$$\frac{\Gamma \vdash A \vee B \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C} \vee e.$$

The soundness proof goes as follows: let $\Gamma \models A \vee B$ and $\Gamma, A \models C$ and $\Gamma, B \models C$, and $M \models \Gamma$. By definition of $\models$, we have $M \models A$ or $M \models B$. In the first case, $M \models \Gamma, A$ and therefore $M \models C$. In the second case, $M \models \Gamma, B$ and therefore $M \models C$.

RAA and excluded middle

- To demonstrate the inference rules for $\vee$, we show the important fact that the law of the excluded middle

$$\frac{}{A \vee \neg A}\,EM$$

is interderivable with RAA.

- This is significant, because from a constructivist's point of view it means that EM is as dubious as RAA.

From RAA to EM

Here is how to derive EM with the help of RAA.

$$\cfrac{\cfrac{[\neg(A \vee \neg A)]_2 \qquad \cfrac{\cfrac{\cfrac{[\neg(A \vee \neg A)]_2 \qquad \cfrac{[A]_1}{A \vee \neg A}\,\vee i}{\bot}\,\to e}{\neg A}\,\to i_1}{A \vee \neg A}\,\vee i}{\bot}\,\to e}{A \vee \neg A}\,RAA_2$$

From EM to RAA

Suppose that we have EM. To show that we have RAA, we must be able to derive A from any derivation D as below.

$$\begin{array}{c} \neg A \\ \vdots\, D \\ \bot \end{array}$$

Here is how it works:

$$\cfrac{\cfrac{}{A \vee \neg A}\,EM \qquad [A] \qquad \cfrac{\begin{array}{c}[\neg A] \\ \vdots\, D \\ \bot\end{array}}{A}\,\bot e}{A}\,\vee e.$$

Soundness and completeness with $\vee$

Theorem.

- If $\Gamma \vdash A$ is provable in the "ND with $\vee$", then $\Gamma \models A$ (soundness).
- If $\Gamma \models A$, then $\Gamma \vdash A$ is provable in "ND with $\vee$".

Proof. Soundness is straightforward. Completeness holds essentially because $B \vee C$ is equivalent with $\neg(\neg B \wedge \neg C)$ and we already have completeness in the absence of $\vee$; the details are somewhat technical and we omit them here.

Predicate logic (revision)

Predicate logic: motivation

- Propositional logic is insufficient for many applications. E.g. it cannot express the sentence

  "Every student is younger than some supervisor".

- To state this sort of sentence, we need **predicate logic**. E.g. the sentence above could be expressed as follows:

$$\forall x.student(x) \to \exists y.supervisor(y) \wedge age(x) < age(y).$$

$\forall$ means "for all" and $\exists$ means "exists".

Predicate logic: new features

Example again:

$$\forall x.student(x) \rightarrow \exists y.supervisor(y) \wedge age(x) < age(y).$$

Predicate logic can be seen as propositional logic plus:

- **variables** (e.g. x, y),
- $(\forall, \exists)$,
- **quantifiers**,
- **functions** (e.g. age), and
- **relations** (e.g. $<$).

Predicate logic and maths

- In particular, predicate logic is ubiquitous in mathematics. E.g. consider

$$\forall \epsilon.\epsilon > 0 \rightarrow \exists \delta.$$
$$\delta > 0 \wedge \forall y.abs(x - y) < \delta \rightarrow abs(f(x) - f(y)) < \epsilon.$$

- Quiz: does this ring a bell?

Syntax

The syntax of predicate logic uses two kinds of expressions:

- **Terms**, e.g. $x, y, age(x), 0, \epsilon, \delta, x - y, f(x), abs(f(x) - f(y))$.
- **Formulæ**, e.g. $supervisor(y)$, $\delta > 0$, $age(x) < age(y)$, $\forall x.student(x)$, $\exists \delta.\delta > 0$.
- Formulæ are those expressions that can be true or false.
- Terms stand for **individuals** of some **universe**.

Signatures

The well-formed terms and formulæ are described by the **signature**:

Definition. A **signature** consists of

- A set of **function symbols** $f, g, h, \ldots$, such that each symbol f has an **arity** $ar(f) \geq 1$ (i.e. a number describing how many arguments f takes).
- A set of **constants** $c, d, \ldots$.
- A set of **relation symbols** $p, q, r, \ldots$, such that each symbol r has an **arity** $ar(r) \geq 0$.

Signatures

Examples.

- "$+$" is a function symbol or arity 2.
- "7" is a constant.
- "$supervisor$" is a relation symbol of arity 1.
- "$<$" is a relation symbol of arity 2.

Terms

Definition. The **terms** for a given signature are given as follows:

- Every variable is a term. (We assume enumerably many variables $x_1, x_2, x_3, \ldots$.)
- Every constant is a term.
- If $t_1, \ldots, t_n$ are terms and f is a function symbol of arity n, then $f(t_1, \ldots, t_n)$ is a term.

Formulæ

Definition. The formulæ of predicate logic are given as follows:

- If $t_1, \ldots, t_n$ are terms and p is a predicate symbol of arity n, then $p(t_1, \ldots, t_n)$ is a formula.
- If A and B are formulæ, then so are $(A \wedge B)$ and $(A \vee B)$ and $(A \to B)$;
- if A is a formula, then so is $(\neg A)$.
- $\top$ and $\bot$ are formula.
- If x is a variable and A is a formula, then $(\forall x.A)$ and $(\exists x.A)$ are formulæ.

Semantics

- A situation for predicate logic is a pair consisting of a **structure** and a **variable assignment**.
- The structure describes the functions and relations corresponding to the the function symbols and relation symbols.
- The variable assignment sends each variable to an element of the **universe** on which the functions and relations are defined.

Structures

Definition. A **structure** M for a given signature consists of

- a non-empty set U called **universe**,
- for every constant c, an element of U,
- for every function symbol f of arity n, an n-ary function f^M, and
- for every relation symbol p of arity n, an n-ary relation p^M.

Examples of structures

- The ring of integers: the universe U is the set of integers; functions are $+$, $*$, unary $-$. Constants are 1 and 0. No relations.
- The ordered set of natural numbers: the universe U is the set of natural numbers; there is one relation, $<$, and no functions or constants.

Situations

Definition. A **situation** M is a structure together with, for every variable x, an element x^M of U.

Semantics of terms

Before we define the satisfaction relation, we must describe the meaning of terms.

Definition. In a situation M, a term t denotes an element $[\![t]\!]_M$ of the universe as follows:

$$[\![x]\!]_M = x^M$$
$$[\![c]\!]_M = c^M$$
$$[\![f(t_1, \ldots, t_n)]\!]_M = f^M([\![t_1]\!]_M, \ldots, [\![t_n]\!]_M)$$

Semantics of formulæ

Definition. The satisfaction relation for predicate logic is defined as follows, where $M[a/x]$ stands for the situation that is like M except that the variable x is interpreted as a.

$$M \models p(t_1, \ldots, t_n) \text{ if } ([\![t_1]\!]_M, \ldots, [\![t_n]\!]_M) \in p^M$$
$$M \models \forall x.A \text{ if for all } a \in U \text{ it holds that } M[a/x] \models A$$
$$M \models \exists x.A \text{ if there exists an } a \in U \text{ such that } M[a/x] \models A$$
$$M \models A \wedge B \text{ if } M \models A \text{ and } M \models B$$
$$M \models A \vee B \text{ if } M \models A \text{ or } M \models B$$
$$M \models A \rightarrow B \text{ if } M \models A \text{ implies } M \models B$$
$$M \models \neg A \text{ if } M \not\models A$$
$$M \models \bot \text{ never}$$
$$M \models \top \text{ never}$$

Predicate logic vs. propositional logic

- By definition of the semantics, for a nullary predicate symbol p we have

$$M \models p() \text{ if } () \in p^M$$

- Such a p has only two possible behaviours: $M \models p()$ or $M \not\models p()$.
- So nullary relation symbols take over the rôle of the propositional atoms.
- Thus propositional logic can be seen as the simplified case of predicate logic where all predicate symbols are nullary.

Validity, satisfiability, semantic entailment

The definitions of validity, satisfiability, and semantic entailment for predicate logic look exactly the same as for propositional logic.