Sequent calculus, proof search, & logic programming (addendum)

Resolution proofs: correction

The explanation of resolution proofs in the last handout is flawed. The following slides contain the corrected version.

Towards resolution

Suppose our goal sequent is

$$\Gamma, \forall x.G \rightarrow p \vdash \exists y.q,$$

an that t, t' are terms such that $p[t/x] = q[t'/y]$. By uniform proof, we get

$$\dfrac{\dfrac{\dfrac{\begin{matrix}\vdots \\ \Gamma \vdash G[t/x]\end{matrix} \qquad \dfrac{}{\Gamma, p[t/x] \vdash q[t'/y]}\,Ax}{\Gamma, G[t/x] \rightarrow p[t/x] \vdash q[t'/y]}\,L\rightarrow}{\Gamma, \forall x.G \rightarrow p \vdash q[t'/y]}\,L\forall}{\Gamma, \forall x.G \rightarrow p \vdash \exists y.q}\,R\exists$$

Resolution

In other words, to prove

$$D_1, \ldots, D_n \vdash \exists y.q,$$

find a D_i of the form

$$D_i = \forall x.G \to p$$

and a terms t, t' such that $p[t/x] = q[t'/y]$. Then it suffices to prove $D_1, \ldots, D_n \vdash G[t/x]$. Such t, t' can be found by **unification**; Prolog essentially works in that way (recall CM20019).

A resolution proof

Given the logic program $\Gamma = \{D_1, D_2\}$ with

$$D_1 = p(0,0)$$

$$D_2 = \forall x.p(x,x) \rightarrow p(s(x), s(x)),$$

we have the resolution proof

$$\dfrac{\dfrac{\dfrac{}{D_1, D_2 \vdash p(0,0)}\,Ax}{D_1, D_2 \vdash p(s(0), p(s(0)))}\ \text{res. with}[0/x]\ \text{(no } y\text{)}}{D_1, D_2 \vdash \exists y.p(y, s(s(0))}\ \text{res. with } [s(0)/x, s(s(0)/y].$$

Completeness of resolution

A **resolution proof** is a uniform proof such that existential goals $\exists x.A$ are treated as we just described. (The precise description is a bit more complicated, because A need not be atomic.)

Theorem. Resolution proofs of HH formulæ are complete w.r.t. minimal predicate logic.

Proof. By showing that every uniform proof can be re-written into a resolution proof.

Modal logic

Motivation

- In natural language, we often use **modes** of truth, e.g. “possibly true ”, “necessarily true”, “known to be true”, “believed to be true”, “true in the future”.
- E.g. the sentence

Tony Blair is prime minister.

is true, but will be false at some point in the future.

Motivation

- Consider the sentence

  There are nine planets in the solar system.

  It is is possibly true, but not necessarily true, because there might be more planets.

- The sentence

  The square root of 9 is 3.

  is necessarily true, and true in the future. But it does not enjoy all modes of truth: it may not be believed to be true (if the believer is mistaken).

Modal logic: overview

- We shall study **modal logics**, which can express modes of truth.
- Modal logics are extremely useful in computer science.
- E.g. it can be used to reason about the knowledge of agents.
- It can also be used to specify the behaviour of computer programs and reactive systems (see e.g. the modal logic CTL in the "Verification by model checking" essay).

Modal formulæ

The language of basic modal logic is that of propositional logic with two extra connectives $\Box$ and $\Diamond$ ("box" and "diamond").

Definition. The formulæ of basic modal logic are defined by the following grammar:

$$A, B ::= \bot | p | A \wedge B | A \vee B | A \to B | \Box A | \Diamond A,$$

where p is an atomic formula.

$\Box$ and $\Diamond$

- In basic modal logic, $\Box$ and $\Diamond$ are read "box" and "diamond".
- But when we express a mode of truth, we may read them appropriately.
- E.g. in the logic for necessity and possibility, $\Box$ is read "necessarily" and $\Diamond$ "possibly".
- In the logic of agent Q's knowledge, $\Box$ is read "agent Q knows" and $\Diamond$ is read "it is consistent with agent Q's knowledge that".
- We shall see later why this makes sense.