Lecture 12, Apr 2, 2024

Proof Procedures

- Given a knowledge base KB, we want to know whether $KB \models \alpha$ where α is some formula (whether the knowledge base *entails* α)
  - We can show that $KB \wedge \neg\alpha$ is unsatisfiable
- If we can manipulate a formula into the empty formula, denoted by $\square$, then we know it is unsatisfiable
  - We know that if $\varphi = \alpha \vee \beta$, then $\text{models}(\varphi) = \text{models}(\alpha) \cup \text{models}(\beta)$
  - If the formula is empty, we have nothing to union, so nothing models it
  - Therefore the empty formula is unsatisfiable
- We are interested in formulas in their *clausical form*, $(C_1 \wedge C_2 \wedge C_3 \wedge \cdots \wedge C_m)$
  - Each clause $C_i = (l'_1 \vee l'_2 \vee \cdots \vee l'_k)$
  - Each l'_i is a *literal*, which is a proposition p or $\neg p$
  - Any formula can be written in clausal form
  - Using the Tseytin transformation, any formula can be converted into an equisatisfiable formula that is linear in size
    - This is as opposed to the naive method of just expanding it out, which leads to exponential size formulas
- In propositional logic, each step of a *proof* is derived from *resolution* $\dfrac{(\alpha \vee p) \wedge (\neg p \vee \beta)}{(\alpha \vee \beta)}$ and $\dfrac{p \wedge \neg p}{\square}$
- Suppose we want to prove by resolution that formula φ is false, i.e. a *resolution refutation*; resolution refutation is a sequence of clauses $C_1, \dots, C_t$ where $C_t = \square$, and all $C_i \in \varphi$ or $\dfrac{C_{i_1} C_{i_2}}{C_k}$ where $i_1, i_2 < k$
  - All the clauses are either the original formula, or implied by previous formulas, leading to an empty formula that is unsatisfiable
- In first-order logic we do this over predicates instead, $\dfrac{(\alpha(x) \vee \neg P(y))(P(y) \vee \beta(z))}{\alpha(x) \vee \beta(x)}$, but quantifiers may be involved
  - If the quantifies are the same, we can do this
  - But if quantifiers are different, this isn't true anymore
  - $\exists x, y, z(\varphi(x, y, z))$ is equivalent to $\neg\forall x, y, z = \neg\varphi(x, y, z)$
- *Skolemization*: we can get rid of one set of quantifiers, e.g. replacing all $\exists$ with $\forall$ or vice versa
- Given any formula we want it first in a form $\forall x, y, z(C_1 \wedge C_2 \wedge \cdots \wedge C_m)$
  - First we get all the quantifiers out of the formula, and then apply skolemization so we are only left with $\forall$
    - Note with $\exists y P(y) \wedge \exists y R(y)$ we cannot simply take out $\exists y$ because the y in P and R are not guaranteed to be the same
    - To avoid this, first we make sure all variables in quantifiers have unique names
    - Once we have unique names for all of them, we can pull them out
    - Note the order of quantifiers matters – we cannot swap them
  - Once we take it out, we can apply resolution just like in propositional logic