Lecture 13

Pete Manolios
Northeastern

Skolem Normal Form Example

For any FO ϕ, we can find a universal ψ in an *expanded* language such that ϕ is satisfiable iff ψ is satisfiable. Try it!

$$\langle \exists x \ \langle \forall w \ \langle \exists y \ \langle \forall u, v \ \langle \exists z \ \phi(x, w, y, u, v, z)\rangle\rangle\rangle\rangle\rangle$$

First, PNF, and push existentials left (2nd order logic)

$$\langle \exists x, F_y \ \langle \forall w, u, v \ \langle \exists z \ \phi(x, w, F_y(w), u, v, z)\rangle\rangle\rangle$$

$$\langle \exists x, F_y, F_z \ \langle \forall w, u, v \ \phi(x, w, F_y(w), u, v, F_z(w, u, v))\rangle\rangle$$

The key idea is the following equivalence

We need the axiom of choice for ping

$$\langle \exists \ldots \ \langle \forall x_1, \ldots x_n \ \langle \exists y \ \phi(\ldots, x_1, \ldots, x_n, y)\rangle\rangle\rangle$$

$$\equiv \ \langle \exists \ldots \ \langle \exists F_y \ \langle \forall x_1, \ldots, x_n \ \phi(\ldots, x_1, \ldots, x_n, F_y(x_1, \ldots, x_n))\rangle\rangle\rangle$$

This allows us to push existential quantifiers to the left

To get back to FO, note that

$$\textbf{Sat}\langle \exists \ldots \ \langle \forall x_1, \ldots x_n \ \langle \exists y \ \phi(\ldots, x_1, \ldots, x_n, y)\rangle\rangle\rangle \ \textbf{iff}$$

$$\textbf{Sat}\langle \forall x_1, \ldots, x_n \ \phi(\ldots, x_1, \ldots, x_n, F_y(x_1, \ldots, x_n))\rangle$$

So, to finish our example, we get, where *c*, F_y, F_z are new symbols,

$$\langle \forall w, u, v \ \phi(c, w, F_y(w), u, v, F_z(w, u, v))\rangle$$

FO Sat/Validity Reductions

Theorem: For any FO ϕ, we can find a universal ψ in an *expanded* language such that ϕ is satisfiable iff ψ is satisfiable. (Proof in previous slide)

Previous example

$$\langle \exists x \ \langle \forall w \ \langle \exists y \ \langle \forall u, v \ \langle \exists z \ \phi(x, w, y, u, v, z) \rangle \rangle \rangle \rangle \rangle$$

$$\langle \forall w, u, v \ \phi(c, w, F_y(w), u, v, F_z(w, u, v)) \rangle$$

Notice that our approach does not give an equi-valid formula. Consider:

$$\langle \forall x \ \langle \exists y \ P(x) \Rightarrow P(y) \rangle \rangle$$

$$\langle \forall x \ P(x) \Rightarrow P(f_y(x)) \rangle$$

Both formulas are satisfiable; the first is valid but the second is not

Corollary: For any FO ϕ, we can find an existential ψ in an *expanded* language such that ϕ is valid iff ψ is valid

Pf: ϕ is valid iff ¬ϕ is unsat iff (universal) ϕ' is unsat iff (existential) ψ=¬ϕ' is valid

$$\phi = \langle \forall x \ \langle \exists y \ P(x) \Rightarrow P(y) \rangle \rangle \quad \rightarrow \quad \neg\phi = \langle \exists x \ \langle \forall y \ P(x) \wedge \neg P(y) \rangle \rangle$$

$$\phi' = \langle \forall y \ P(c) \wedge \neg P(y) \rangle \quad \rightarrow \quad \psi = \langle \exists y \ P(c) \Rightarrow P(y) \rangle$$

So FO Sat reduced to FO universal Sat and FO Validity to FO universal Unsat

Connections with ACL2

For any FO ϕ, we can find a universal ψ in an *expanded* language such that ϕ is satisfiable iff ψ is satisfiable.

$\langle \forall u, v \ \langle \exists z \ \phi(u, v, z) \rangle \rangle$

$\langle \forall u, v \ \langle \exists z \ (App \ u \ v) = (Rev \ z) \rangle \rangle$

First, PNF, and push existentials left (2nd order logic)

$\langle \exists F_z \ \langle \forall u, v \ \phi(u, v, F_z(u, v)) \rangle \rangle$

$\langle \exists F_z \ \langle \forall u, v \ (App \ u \ v) = (Rev \ (F_z \ u \ v)) \rangle \rangle$

Previously, we saw how to go back to FO while preserving SAT with

$\langle \forall u, v \ \phi(u, v, F_z(u, v)) \rangle$

$\langle \forall u, v \ (App \ u \ v) = (Rev \ (F_z \ u \ v)) \rangle$

But what about preserving validity? This method doesn't work, as we've seen.

Can we make it work in a FO setting?

This is how ACL2 handles quantifiers

DEMO

$\langle \forall u, v \ \langle \exists z \ (App \ u \ v) = (Rev \ z) \rangle \rangle$

$\longrightarrow$

$\langle \forall u, v \ (E_z \ u \ v) \rangle$

As above, but not enough

$(E_z \ u \ v) \ \equiv \ (App \ u \ v) = (Rev \ (F_z \ u \ v))$

Constrain F_z:

$(App \ u \ v) = (Rev \ z) \ \Rightarrow \ (E_z \ u \ v)$

if (App *u v*) = (Rev *z*) has solution

then F_z is also a solution

Reduce FOL to Propositional SAT

- We reduced FOL SAT to SAT of the universal fragment
- We now go one step further

ground: quantifier/variable free

- Theorem: A universal FO formula (w/out =) is SAT iff all finite sets of ground instances are (propositionally) SAT (eg $P(x) \vee \neg P(x)$ is propositionally SAT)
- Corollary: A universal FO formula (w/out =) is UNSAT iff some finite set of ground instances is (propositionally) UNSAT
- FO validity checker: Given FO ϕ, negate & Skolemize to get universal ψ s.t. Valid(ϕ) iff UNSAT(ψ). Let G be the set of ground instances of ψ (possibly infinite, but countable). Let G_1, G_2 …, be a sequence of finite subsets of G s.t. $\forall g \subseteq G, |g| < \omega$, $\exists n$ s.t. $g \subseteq G_n$. If $\exists n$ s.t. Unsat G_n, then Unsat ψ and Valid ϕ
- The SAT checking is done via a propositional SAT solver!
- If ϕ is not valid, the checker may never terminate, i.e., we have a semi-decision procedure and we'll see that's all we can hope for
- How should we generate G_i? One idea is to generate all instances over terms with at most 0, 1, … , functions. We'll explore that more later.

Example

$\langle \exists x \; \langle \forall y \; P(x) \Rightarrow P(y) \rangle \rangle$ **is Valid?**

Example

$\langle \exists x\ \langle \forall y\ P(x) \Rightarrow P(y) \rangle\rangle$ **is Valid iff** $\langle \forall x\ \langle \exists y\ P(x) \wedge \neg P(y) \rangle\rangle$ **is UNSAT**

iff $\langle \forall x\ P(x) \wedge \neg P(f_y(x)) \rangle$ **is UNSAT**

with smart Skolemization **iff** $\langle \forall x\ P(x) \wedge \neg P(c) \rangle$ **is UNSAT**

- *Herbrand universe* of FO language L is the set of all ground terms of L, except that if L has no constants, we add c to make the universe non-empty.
- For our example we have $H = \{c, f_y(c), f_y(f_y(c)), \ldots\}$
- So $G = \{P(t) \wedge \neg P(f_y(t)) \mid t \in H\}$
- Notice that $\Delta = \{P(c) \wedge \neg P(f_y(c)), P(f_y(c)) \wedge \neg P(f_y(f_y(c)))\}$ is UNSAT
  - the SAT solver will report UNSAT for: $P(c) \wedge \neg P(f_y(c)) \wedge P(f_y(c)) \wedge \neg P(f_y(f_y(c)))$
- So, for the first G_i that has both $\neg P(f_y(c))$ and $P(f_y(c))$ will lead to termination
- BTW, why do we restrict ourselves to FO w/out equality?
  - Consider $P(c) \wedge \neg P(d) \wedge c{=}d$
  - $H = \{c,d\}$
  - $G = \{P(c) \wedge \neg P(d) \wedge c{=}d\}$, which is propositionally SAT, but FO UNSAT
- This is why smart Skolemization is useful

Propositional Compactness

- A set Γ of propositional formulas is SAT iff every finite subset is SAT
- This is a key theorem justifying the correctness of our FO validity checker
- Proof: Ping is easy. Let p_1, p_2, …, be an enumeration of the atoms (assume the set of atoms is countable). Define Δ_i as follows
  - $\Delta_0 = \Gamma$
  - $\Delta_{n+1} = \Delta_n \cup \{p_{n+1}\}$ if this is finitely SAT
  - $\Delta_{n+1} = \Delta_n \cup \{\neg p_{n+1}\}$ otherwise

  Note: for all i, Δ_i is finitely SAT as is $\Delta = \cup_i \Delta_i$ (any finite subset is in some Δ_i)

  Here is an assignment for Γ: $v(p_i)$ = true iff $p_i \in \Delta$

Herbrand Interpretations

- Theorem: A universal FO formula (w/out =) is SAT iff all finite sets of ground instances are (propositionally) SAT (eg $P(x) \vee \neg P(x)$ is propositionally SAT)
- Let ψ be a universal FO formula w/out equality
- Let H be the Herbrand universe (all ground terms in language of ψ, as before)
- If G (all ground instances of ψ) is propositionally UNSAT then ψ is UNSAT (universal formulas imply all their instances)
- If G is propositionally SAT, say with assignment v, then ψ is SAT
  - Let $\mathcal{I}$ be a canonical interpretation where the universe is H and
  - constants are interpreted autonomously: $a(c) = c$
  - functions are interpreted autonomously: $a(f\ t_1 \ldots t_n) = f\ t_1 \ldots t_n$
  - relations are interpreted as follows: $\langle t_1, \ldots, t_n \rangle \in a.R$ iff $v(R\ t_1, \ldots, t_n) = \text{true}$
  - variables are mapped to terms (how doesn't matter)
- Notice that $\mathcal{I} \models \psi$. We need to check that for all vars $x_1, \ldots, x_n$ in ψ, and for all $t_1, \ldots, t_n$ in H, $\mathcal{J}\frac{t_1 \ldots t_n}{x_1 \ldots x_n} \models \psi$ iff $\mathcal{J}\frac{\mathcal{J}(t_1) \ldots \mathcal{J}(t_n)}{x_1 \ldots x_n} \models \psi$ iff $\mathcal{J} \models \psi\frac{t_1 \ldots t_n}{x_1 \ldots x_n}$

  which holds by construction since G contains all ground instances

FOL Checking

- FO validity checker: Given FO ϕ, negate & Skolemize to get universal ψ s.t. Valid(ϕ) iff UNSAT(ψ). Let G be the set of ground instances of ψ (possibly infinite, but countable). Let G_1, G_2 …, be a sequence of finite subsets of G s.t. $\forall g \subseteq G, |g| < \omega$, $\exists n$ s.t. $g \subseteq G_n$. $\exists n$ s.t. Unsat G_n iff Unsat ψ (and Valid ϕ)
- Question 1: SAT checking
  - Gilmore (1960): Maintain conjunction of instances so far in DNF, so SAT checking is easy, but there is a blowup due to DNF
  - Davis Putnam (1960): Convert ψ to CNF, so adding new instances does not lead to blowup
  - In general, any SAT solver can be used, eg, DPLL much better than DNF
- Question 2: How should we generate G_i?
  - Gilmore: Instances over terms with at most 0, 1, … , functions
  - Any such “naive” method leads to lots of useless work, eg, the book has code for minimizing instances and reductions can be drastic

Unification

- Better idea: intelligently instantiate formulas. Consider the clauses

  $\{P(x, f(y)) \vee Q(x, y), \neg P(g(u), v)\}$

- Instead of blindly instantiating, use $x=g(u)$, $v=f(y)$ so that we can resolve

  $\{P(g(u), f(y)) \vee Q(g(u), y), \neg P(g(u), f(y))\}$

- Now, resolution gives us

  $\{Q(g(u), y)\}$

- Much better than waiting for our enumeration to allow some resolutions
- Unification: Given a set of pairs of terms $S = \{(s_1,t_1), \ldots, (s_n,t_n)\}$ a *unifier* of S is a substitution σ such that $s_i|\sigma = t_i|\sigma$
- We want an algorithm that finds a *most general* unifier if it exists
  - σ is *more general* than τ, $\sigma \leq \tau$, iff $\tau = \delta \circ \sigma$ for some substitution δ
  - Notice that if σ is a unifier, so is $\delta \circ \sigma$
- Similar to solving a set of simultaneous equations, e.g., find unifiers for
  - $\{(P(f(w), f(y)), P(x, f(g(u)))), (P(x,u), P(v,g(v)))\}$ and $\{(x, f(y)), (y, g(x))\}$