First-Order Logic

CS560: Reasoning About Programs

Roopsha Samanta

PURDUE UNIVERSITY

Partly based on slides by Aaron Bradley

Roadmap

Previously

- Propositional logic
- SAT solving

Today

- Syntax and semantics of first-order logic (FOL)
- Semantic argument method for FOL validity
- Properties of FOL

Propositional Logic

$$P \land Q \rightarrow P \lor \neg Q$$

- Simple, not very expressive
- Decidable
  - Automated reasoning about satisfiability/validity

First-Order Logic

(predicate logic/predicate calculus/ relational logic)

$$\forall x. p(x, y) \rightarrow \exists y. \neg q(x, y)$$

- Very expressive
- Semi-decidable
  - Not fully automated

Syntax of FOL

constants: a, b, c
variables: x, y, z
n-ary functions: f, g, h
n-ary predicates: p, q, r

logical connectives: $\neg$, $\lor$, $\land$, $\rightarrow$, $\leftrightarrow$
quantifiers: $\exists$, $\forall$

Term
constant, variable, or,
n-ary function applied to n terms

Atom
$\top$, $\bot$, or,
n-ary predicate applied to n terms

Literal
atom or its negation

FOL formula:
Literal, or, application of logical connectives to an FOL formula, or, application of a quantifier to an FOL formula

Quantifiers

existential quantifier: $\exists x. F(x)$ "there exists an x such that $F(x)$"

universal quantifier: $\forall x. \underbrace{F(x)}$ "for all x, $F(x)$"

Quantified variable

Scope of quantified variable

A variable is **bound** if there exists an occurrence in the scope of some quantifier

A variable is **free** if there exists an occurrence not bound by any quantifier

A variable may be both bound and free!

Closed/Ground formula:
no free variables

Open formula: some free variables

Ground, quantifier-free formula:
no variables

U : people $\cup$ $\mathbb{N}$

constants : { Roopsha, . . . }

vars : people-valued

functions : adviser (arity 1), age (arity 1)

(0-ary functions — constants)

predicates : taller (arity 2), isStudent (arity 1)

(0-ary predicates — prop. vars)

Terms : x, a, $g(a)$, $f(x, g(y,z))$

$age(child(Bob, mother(Alice)))$

Atoms : $p(x)$, $\neg p(f(x))$, $g(x, f(x)))$

$father(mother(Alice), Bob)$

formula : $\forall x. (p(x,y) \rightarrow (\exists y. \neg q(x,y)))$

Open !

free Bound Bound

Closed formula : $\forall x. (\forall y. p(x,y) \rightarrow \exists y. \neg q(x,y)))$

Ground formula : $p(a, f(b, c))$

Semantics of FOL: first-order structure $\langle U, I \rangle$

- Universe of discourse/domain, U:
  - Non-empty set of values or objects of interest
  - May be finite (set of students at Purdue), countably infinite (integers) or uncountable infinite (positive reals)

- Interpretation, I: Mapping of variables, functions and predicates to values in U
  - I maps each variable symbol x to some value $I[x] \in U$
  - I maps each n-ary function symbol f to some function $f_I: U^n \to U$
  - I maps each n-ary predicate symbol p to some predicate $p_I: U^n \to \{true, false\}$

Evaluation of formulas

If F evaluates to T under U, I, we write $\langle U, I \rangle \vDash F$

If F evaluates to $\bot$ under U, I, we write $\langle U, I \rangle \nvDash F$

Evaluation of terms: $I[f(t_1, \dots, t_n)] = I[f](I[t_1], \dots, I[t_n])$

Evaluation of atoms: $I[p(t_1, \dots, t_n)] = I[p](I[t_1], \dots, I[t_n])$

$$U = \{\square, \triangle\}$$

$$I[x] = \triangle, \quad I[a] = \square, \quad I[b] = \triangle$$

$$I[f] = \{\triangle \mapsto \square, \ \square \mapsto \triangle\}$$

$$I[p] = \{(\square, \square) \mapsto \text{true}, \ (\triangle, \triangle) \mapsto \text{false}\}$$

$$F: \ p(f(a), f(x))$$

$$I[f(a)] = I[f](I[a]) = \{\triangle \mapsto \square, \ \square \mapsto \triangle\}(\square) = \triangle$$

$$I[f(x)] = \square$$

$$I[p(f(a), f(x))] = I[p](\triangle, \square) = \text{false}$$

Evaluation of formulas: inductive definition

Base Cases:

$\langle U, I \rangle \vDash \top$
$\langle U, I \rangle \nvDash \bot$
$\langle U, I \rangle \vDash p(t_1, \ldots, t_n)$
iff $I[p(t_1, \ldots, t_n)] = true$

Inductive Cases:

$\langle U, I \rangle \vDash \neg F$ iff $\langle U, I \rangle \nvDash F$
$\langle U, I \rangle \vDash F_1 \vee F_2$ iff $\langle U, I \rangle \vDash F_1$ or $\langle U, I \rangle \vDash F_2$
…
$\langle U, I \rangle \vDash \forall x. F$ iff for all $v \in U, I[x \mapsto v] \vDash F$
$\langle U, I \rangle \vDash \exists x. F$ iff there exists $v \in U, I[x \mapsto v] \vDash F$

x-variant of $\langle \mathrm{U}, \mathrm{I} \rangle$ that agrees with U, I on everything except the variable x, with $\mathrm{I}[\mathrm{x}] = \mathrm{v}$.

$F: \forall x.\, p(x, a) \;\; \begin{cases} \square .\; p(\square, \square) & T \\ \triangle .\; p(\triangle, \square) & F \end{cases} \;\; \Big| \; \langle U, I \rangle \not\models F$

$G: \exists x.\, p(f(x), f(x)) \rightarrow p(x, x)$

$\square .\; p(\triangle, \triangle) \rightarrow p(\triangle, \triangle)$

$\langle U, I \rangle \models G$

Satisfiability and Validity

F is **satisfiable** iff there exists some structure $\langle U, I \rangle : \langle U, I \rangle \vDash F$

F is **valid** iff for all structures $\langle U, I \rangle : \langle U, I \rangle \vDash F$

Duality:
F is valid iff $\neg F$ is unsatisfiable

$\forall x . \exists y . p(x, y)$

satisfiable? satisfying structure?

valid? falsifying structure?

Semantic argument method for validity

Proof by contradiction:

1. Assume F is not valid
2. Apply proof rules
3. Contradiction (i.e, ⊥) along **every** branch of proof tree ⇒ F is valid
4. Otherwise, F is not valid

$$\frac{\langle U,I\rangle \models \neg F}{\langle U,I\rangle \not\models F}$$

$$\frac{\langle U,I\rangle \not\models \neg F}{\langle U,I\rangle \models F}$$

$$\frac{\langle U,I\rangle \models F \wedge G}{\begin{array}{c}\langle U,I\rangle \models F\\ \langle U,I\rangle \models G\end{array}}$$

$$\frac{\langle U,I\rangle \not\models F \wedge G}{\langle U,I\rangle \not\models F \mid \langle U,I\rangle \not\models G}$$

$$\frac{\langle U,I\rangle \models F \rightarrow G}{\langle U,I\rangle \not\models F \mid \langle U,I\rangle \models G}$$

$$\frac{\langle U,I\rangle \not\models F \rightarrow G}{\begin{array}{c}\langle U,I\rangle \models F\\ \langle U,I\rangle \not\models G\end{array}}$$

...

Semantic argument method for validity

$$\frac{\langle U, I\rangle \vDash \forall x. F}{\langle U, I[x \mapsto c]\rangle \vDash F}$$ (for any $c \in U$)

$$\frac{\langle U, I\rangle \nvDash \forall x. F}{\langle U, I[x \mapsto c]\rangle \nvDash F}$$ (for some fresh $c \in U$)

$$\frac{\langle U, I\rangle \vDash \exists x. F}{\langle U, I[x \mapsto c]\rangle \vDash F}$$ (for some fresh $c \in U$)

$$\frac{\langle U, I\rangle \nvDash \exists x. F}{\langle U, I[x \mapsto c]\rangle \nvDash F}$$ (for any $c \in U$)

$$\frac{\langle U,I\rangle \vDash \neg F}{\langle U,I\rangle \nvDash F} \qquad \frac{\langle U,I\rangle \nvDash \neg F}{\langle U,I\rangle \vDash F}$$

$$\frac{\langle U,I\rangle \vDash F \wedge G}{\begin{array}{c}\langle U,I\rangle \vDash F \\ \langle U,I\rangle \vDash G\end{array}} \qquad \frac{\langle U,I\rangle \nvDash F \wedge G}{\langle U,I\rangle \nvDash F \mid \langle U,I\rangle \nvDash G}$$

$$\frac{\langle U,I\rangle \vDash F \rightarrow G}{\langle U,I\rangle \nvDash F \mid \langle U,I\rangle \vDash G} \qquad \frac{\langle U,I\rangle \nvDash F \rightarrow G}{\begin{array}{c}\langle U,I\rangle \vDash F \\ \langle U,I\rangle \nvDash G\end{array}}$$

...

$$\frac{\begin{array}{c}\langle U,I\rangle \vDash p(s_1,\dots,s_n) \\ \langle U,I\rangle \nvDash p(t_1,\dots,t_n) \\ I[s_i] = I[t_i] \text{ for all } i \in [1,n]\end{array}}{\langle U,I\rangle \vDash \bot}$$

$F : (\forall x.\, p(x)) \to (\forall y.\, p(y))$

1. $\langle U, I \rangle \not\models F$ — ass.
2. $\langle U, I \rangle \models \forall x.\, p(x)$ — 1, $\to$
3. $\langle U, I \rangle \not\models \forall y.\, p(y)$ — 1, $\to$
4. $\langle U, I[y \mapsto v] \rangle \not\models p(y)$ — 3, $\not\forall$, $v \in U$
5. $\langle U, I[x \mapsto v] \rangle \models p(x)$ — 2, $\forall$
6. $\langle U, I \rangle \models \bot$

$I[x \mapsto, y \mapsto v]$

Soundness and Completeness of Proof Rules

Soundness:
If every branch of semantic argument proof derives $\perp$, then F is valid

Completeness:
If F is valid, there exists a finite-length semantic argument proof in which every branch derives $\perp$.

Undecidability of FOL

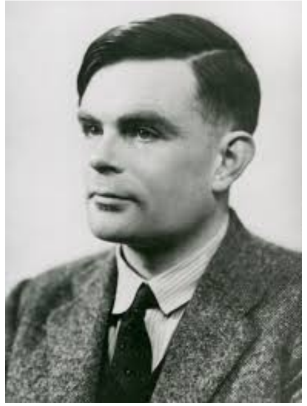
A problem is decidable if there exists a procedure that, for any input:
1. halts and says “yes” if answer is positive, and
2. halts and says “no” if answer is negative
(Such a procedure is called an algorithm or a decision procedure)

Undecidability of FOL [Church and Turing]:
Deciding the validity of an FOL formula is undecidable

Deciding the validity of a PL formula is decidable
The truth table method is a decision procedure

Church

Turing



Semi-decidability of FOL

A problem is semi-decidable iff there exists a procedure that, for any input:
1. halts and says “yes” if answer is positive, and
2. may not terminate if answer is negative.

Semi-decidability of FOL:
For every valid FOL formula, there exists a procedure (semantic argument method) that always terminates and says “yes”.
If an FOL formula is invalid, there exists no procedure that is guaranteed to terminate.

Summary

Today

- Syntax and semantics of first-order logic (FOL)
- Semantic argument method for FOL validity
- Properties of FOL

Next

- First-order theories