Advanced Software Security

11. First-order Theories

Kihong Heo

KAIST

Motivation

- Full first-order logic: functions and predicates are uninterpreted (i.e., determined by I)
- Validity of full FOL: valid in all interpretations
- Do we really care about all interpretations?
- NO. Only some specific classes (theory) of interpretations depending on applications
  - E.g., numbers, lists, arrays, strings, etc
- Another good news:
  - Validity of FOL: generally undecidable
  - Validity in particular theories: sometimes decidable

First-order Theory

- Theory T : A restricted class of FOL
  - Signature Σ_T : a set of constants, functions, and predicate symbols
  - Axioms A_T : a set of FOL sentences over Σ_T
- Σ_T formula: formula constructed from
  - Symbols of Σ_T
  - Variables, logical connectives, and quantifiers

Example: The theory of equality T_E

- $\Sigma_E : \{ =, a, b, c, \ldots, f, g, h, \ldots, p, q, r, \ldots \}$
- Equality “=” is an interpreted predicate symbol
  - The conventional interpretation of “=” and other uninterpreted symbols
- The other functions, predicates, and constants are uninterpreted

Example: The theory of equality T_E

- Axioms A_E
  - Reflexivity: $\forall x \,.\, x = x$
  - Symmetry: $\forall x, y \,.\, x = y \rightarrow y = x$
  - Transitive: $\forall x, y, z \,.\, x = y \wedge y = z \rightarrow x = z$
  - Function congruence: $\forall \overrightarrow{x}, \overrightarrow{y} \,.\, (\bigwedge_{i=1}^{n} x_i = y_i) \rightarrow f(\overrightarrow{x}) = f(\overrightarrow{y})$
  - Predicate congruence: $\forall \overrightarrow{x}, \overrightarrow{y} \,.\, (\bigwedge_{i=1}^{n} x_i = y_i) \rightarrow (p(\overrightarrow{x}) \leftrightarrow p(\overrightarrow{y}))$

Validity and Satisfiability Modulo Theory

- Given a first-order theory T, a Σ_T formula F
- T-interpretation: an interpretation that satisfies all the axioms of T
- F is valid in the theory T if all T-interpretations satisfy F
  - $T \models F$
- F is satisfiable in the theory T if there exists a T-interpretation that satisfies F

Example

- Prove $F : a = b \wedge b = c \rightarrow g(f(a), b) = g(f(c), a)$ is T_E-valid

Decidability

- A theory T is decidable if $T \models F$ is decidable for every Σ_T-formula F
- Many theories are undecidable
  - E.g., the theory of equality is undecidable
- Some theories become decidable with further restrictions
  - Quantifier-free fragment: formulae without quantifiers
  - Conjunctive fragment: formulae with only conjunctions

First-order Theories for Programs

- Equality
- Integers, rationals, and reals
- Lists
- Arrays
- Pointers
- Bit-vectors
- etc

Theory of Peano Arithmetic

- $\Sigma_{PA} : \{ 0,1, +, \cdot, = \}$
  - 0 and 1 : constants
  - + (addition) and · (multiplication) are binary functions
  - and = (equality) is a binary predicate

Theory of Peano Arithmetic

- Axioms of T_{PA}
  - Zero: $\forall x \,.\, \neg(x + 1 = 0)$
  - Successor: $\forall x, y, .\, x + 1 = y + 1 \rightarrow x = y$
  - Plus zero: $\forall x \,.\, x + 0 = x$
  - Plus successor: $\forall x, y \,.\, x + (y + 1) = (x + y) + 1$
  - Times zero: $\forall x \,.\, x \cdot 0 = 0$
  - Times successor: $\forall x, y, z \,.\, x \cdot (y + 1) = x \cdot y + x$
  - Induction: $F[0] \wedge (\forall x \,.\, F[x] \rightarrow F[x + 1]) \rightarrow \forall x \,.\, F[x]$

An axiom schema for every Σ_{PA}-formula F with one free variable

Example

- $3x + 5 = 2y \ : \ (1 + 1 + 1) \cdot x + 1 + 1 + 1 + 1 + 1 = (1 + 1) \cdot y$
- $3x + 5 > 2y \ : \ \exists z \,.\, z \neq 0 \wedge 3x + 5 = 2y + z$
- $\exists x, y, z \,.\, x \neq 0 \wedge y \neq 0 \wedge z \neq 0 \wedge xx + yy = zz \ : \ T_{PA}$-valid
- $\{ \forall x, y, z \,.\, x \neq 0 \wedge y \neq 0 \wedge z \neq 0 \rightarrow x^n + y^n \neq z^n \mid n > 2 \wedge n \in \mathbb{Z} \} : T_{PA}$-valid

Theory of Array

- $\Sigma_A : \{ \cdot [\cdot], \cdot \langle \cdot \triangleleft \cdot \rangle, = \}$
  - $a[i]$ (read) is a binary function: the value of array a at position i)
  - $a \langle i \triangleleft v \rangle$ (write) is a ternary function: the modified array a in which position i has value v
  - and = (equality) is a binary predicate

Theory of Array

- Axioms of T_A
  - Reflexivity: $\forall x \,.\, x = x$
  - Symmetry: $\forall x, y \,.\, x = y \rightarrow y = x$
  - Transitive: $\forall x, y, z \,.\, x = y \wedge y = z \rightarrow x = z$
  - Array congruence: $\forall a, i, j \,.\, i = j \rightarrow a[i] = a[j]$
  - Read-over-write 1: $\forall a, v, i, j \,.\, i = j \rightarrow a\langle i \lhd v \rangle[j] = v$
  - Read-over-write 2: $\forall a, v, i, j \,.\, i \neq j \rightarrow a\langle i \lhd v \rangle[j] = a[j]$

Decidability of Theories

Description	Full	QFF
equality	no	yes
Peano arithmetic	no	no
Presburger arithmetic	yes	yes
linear integers	yes	yes
reals with multiplication	yes	yes
rationals without multplication	yes	yes
recursive data structures	no	yes
acyclic recursive data structures	yes	yes
arrays	no	yes
arrays with extentionality	no	yes

Summary

- First-order theories: instances of FOL
- Many useful theories for program verification
  - E.g., equality, integers, arrays, pointers, etc
- Some theories are decidable but some are not