EECS 598-008 & EECS 498-008: Intelligent Programming Systems

Lecture 4

Announcements

- **Remote & live** discussion section 3-4pm Friday September 10 (tomorrow)
  - Top-down search algorithm, A1, Z3 (briefly)
  - Zoom (same password as lectures)
- A1 due **midnight Tuesday September 14**
  - Questions? Come to discussion section (Friday) and/or GSI OH (Monday)

This and Next Lectures

- Logics
  - Understand SYGUS framework
  - Deduction-based pruning (this is more important)
    - A2
    - Upcoming paper presentations
    - Potentially final project
- Propositional Logic (today)
- First-Order Logic (today/next lecture)
- First-Order Theories (next lecture)

Agenda

- **Propositional Logic**
- First-Order Logic
- First-Order Theories

Propositional Logic

- E.g., $(p \wedge q) \rightarrow (p \vee \neg q)$
- Should already know this stuff — quick refresher!

Propositional Logic Syntax

- What are valid propositional logic formulas? How to write them?

Propositional Logic Syntax

- What are valid propositional logic formulas? How to write them?

- Basic building blocks:
  - Logical constants: $\top$ (“true”, 1) and $\bot$ (“false”, 0)
  - Propositional variables: $p, q, r, x, y, z, p_1, q_1, r_1, \ldots$
  - Logic connectives: $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$

Propositional Logic Syntax

- Basic building blocks: logical constants, propositional variables, logical connectives

- Use these blocks to build formulas, recursively:
  - Each logical constant is a formula (base case)
  - Each propositional variable is a formula (base case)
  - If F_1 and F_2 are formulas, all of the following are also formulas (recursive case)

  $(F_1), \neg F_1, F_1 \wedge F_2, F_1 \vee F_2, F_1 \rightarrow F_2, F_1 \leftrightarrow F_2$

Propositional Logic Syntax

- Quiz:
  - Is x a propositional logic formula?

Propositional Logic Syntax

- Quiz:
  - Is x a propositional logic formula?
  - Is $x = y \wedge p$ a propositional logic formula?

Propositional Logic Syntax

- Quiz:
  - Is x a propositional logic formula?
  - Is $x = y \wedge p$ a propositional logic formula?
  - Is $\forall x \,.\, x > 5$ a propositional logic formula?

Propositional Logic Syntax

- Quiz:
  - Is x a propositional logic formula?
  - Is $x = y \land p$ a propositional logic formula?
  - Is $\forall x\,.\,x > 5$ a propositional logic formula?
  - Is $(x \land \neg x) \lor y$ a propositional logic formula?

Propositional Logic Semantics

- What is the meaning of a propositional logic formula?
  - What truth value does it “evaluate to”?

Propositional Logic Semantics

- What is the meaning of a propositional logic formula?
  - What truth value does it "evaluate to"?

- A formula is evaluated under an "**interpretation**"
  - Interpretation I: assignment of boolean values to propositional variables

$$I : \{p \mapsto \top\ , q \mapsto \bot\ , \ldots\}$$

Propositional Logic Semantics

- Interpretation I: assignment of boolean values to propositional variables

$$I : \{p \mapsto \top , q \mapsto \bot , \ldots\}$$

- A formula F evaluates to a truth value under an interpretation I
  - $I \models F$: F evaluates to $\top$ under I (i.e., I is a satisfying assignment/model)
  - $I \not\models F$: F evaluates to $\bot$ under I (i.e., I is a falsifying assignment/counter-model)

- $\models$, “entailment”, is defined inductively

Propositional Logic Semantics

- Base cases
  - $I \models \top$ $\quad I \not\models \bot$ $\quad I \models p$ iff $I[p] = \top$ $\quad I \not\models p$ iff $I[p] = \bot$

Propositional Logic Semantics

- Base cases
  - $I \models \top$ $\quad I \not\models \bot$ $\quad I \models p$ iff $I[p] = \top$ $\quad I \not\models p$ iff $I[p] = \bot$

- Inductive cases
  - $I \models (F)$ iff $I \models F$
  - $I \models \neg F$ iff $I \not\models F$
  - $I \models F_1 \wedge F_2$ iff $I \models F_1$ and $I \models F_2$
  - $I \models F_1 \vee F_2$ iff $I \models F_1$ or $I \models F_2$
  - $I \models F_1 \rightarrow F_2$ iff $I \not\models F_1$ or $I \models F_2$
  - $I \models F_1 \leftrightarrow F_2$ iff $I \models F_1$ and $I \models F_2$
    or $I \not\models F_1$ and $I \not\models F_2$

Propositional Logic Semantics

- Consider $F : (p \vee q) \rightarrow (p \wedge q)$
  - What does F evaluate to under $I : \{p \mapsto \top, q \mapsto \top\}$

Propositional Logic Semantics

- Consider $F : (p \vee q) \rightarrow (p \wedge q)$
  - What does F evaluate to under $I : \{p \mapsto \top, q \mapsto \top\}$
  - What does F evaluate to under $I : \{p \mapsto \bot, q \mapsto \bot\}$

Propositional Logic Semantics

- Consider $F : (p \vee q) \rightarrow (p \wedge q)$
  - What does F evaluate to under $I : \{p \mapsto \top, q \mapsto \top\}$
  - What does F evaluate to under $I : \{p \mapsto \bot, q \mapsto \bot\}$
  - What does F evaluate to under $I : \{p \mapsto \top, q \mapsto \bot\}$

Satisfiability and Validity

- F is **satisfiable** iff **there exists** an interpretation I such that $I \models F$
  - I is a **satisfying assignment**
- F is **unsatisfiable** iff **for all** interpretations I, $I \not\models F$

Satisfiability and Validity

- F is **satisfiable** iff **there exists** an interpretation I such that $I \models F$
  - I is a **satisfying assignment**
- F is **unsatisfiable** iff **for all** interpretations $I, I \not\models F$

- F is **valid** iff **for all** interpretations $I, I \models F$
- F is **not valid** iff **there exists** an interpretations I such that $I \not\models F$

Satisfiability and Validity

- F is **satisfiable** iff **there exists** an interpretation I such that $I \models F$
  - I is a **satisfying assignment**
- F is **unsatisfiable** iff **for all** interpretations I, $I \not\models F$

- F is **valid** iff **for all** interpretations I, $I \models F$
- F is **not valid** iff **there exists** an interpretations I such that $I \not\models F$

- **Duality between satisfiability and validity: F is valid iff $\neg F$ is unsatisfiable**
  - … which means: if we know how to check satisfiability, we can check validity as well

Satisfiability and Validity

- SAT, UNSAT, Valid, Not Valid?

	SAT?	UNSAT?	Valid?	Not Valid?
• p				
• $(p \land q) \to p$				
• $(p \to q) \to (\neg(p \land \neg q))$				

Deciding Satisfiability and Validity

- Two ways (ideas):
  - Truth table method: Brute-force all interpretations and check
  - Semantic argument method: Use logical deduction

Method 1: Truth Table Method

- E.g., consider $F : (p \wedge q) \rightarrow p$

p	q	$p \wedge q$	$(p \wedge q) \rightarrow p$
T	T	T	T
T	F	F	T
F	T	F	T
F	F	F	T

Method 1: Truth Table Method

- Can be automated, but bad idea!
  - Slow: Exponentially many interpretations to # propositional variables
  - Impossible when the domain is infinite, e.g., First-Order Logic

Method 2: Semantic Argument Method

- Essentially, proof by contradiction
- Key idea: Assume F is not valid and try to find a falsifying interpretation I *s.t.* $I \not\models F$
  - If found, then not valid
  - If not, then contradiction, so F is valid

Method 2: Semantic Argument Method

- Essentially, proof by contradiction
- Key idea: Assume F is not valid and try to find a falsifying interpretation I *s.t.* $I \not\models F$
  - If found, then not valid
  - If not, then contradiction, so F is valid
- How to "find" falsifying interpretation?
  - Apply (a predefined set of) proof rules
  - If we derive a contradiction in **every** branch of the proof, then F is valid

Method 2: Semantic Argument Method

- Proof Rule 1: $I \models \neg F \Rightarrow I \not\models F$
- Proof Rule 2: $I \not\models \neg F \Rightarrow I \models F$
- Proof Rule 3: $I \models F \land G \Rightarrow I \models F$ and $I \models G$
- Proof Rule 4: $I \not\models F \land G \Rightarrow I \not\models F$ or $I \not\models G$ **Creating two branches!**
- Proof Rule 5: $I \models F \lor G \Rightarrow I \models F$ or $I \models G$ **Creating two branches!**
- Proof Rule 6: $I \not\models F \lor G \Rightarrow I \not\models F$ and $I \not\models G$
- Proof Rule 7: $I \models F \rightarrow G \Rightarrow I \not\models F$ or $I \models G$ **Creating two branches!**
- Proof Rule 8: $I \not\models F \rightarrow G \Rightarrow I \models F$ and $I \not\models G$
- Proof Rule 9 (Contradiction): $I \models F$ and $I \not\models F \Rightarrow I \models \bot$

Method 2: Semantic Argument Method

- Proof rules as **inference rules**:
  - $I \models \neg F \Rightarrow I \not\models F$

$$\frac{I \models \neg F}{I \not\models F}$$

  - $I \not\models \neg F \Rightarrow I \models F$

$$\frac{I \not\models \neg F}{I \models F}$$

  - $I \not\models F \wedge G \Rightarrow I \not\models F$ or $I \not\models G$

$$\frac{I \not\models F \wedge G}{I \not\models F \quad \mid \quad I \not\models G}$$

  - $I \models F$ and $I \not\models F \Rightarrow I \models \bot$

$$\frac{\begin{array}{c} I \models F \\ I \not\models F \end{array}}{I \models \bot}$$

Method 2: Semantic Argument Method

- Prove $F : \big(\neg(p \wedge q)\big) \vee p$ is valid
- Recall: Assume F is not valid and try to find a falsifying interpretation I *s.t.* $I \not\models F$
  - If found, then not valid
  - If not, then contradiction, so F is valid

Method 2: Semantic Argument Method

- Prove $F : \big(\neg(p \land q)\big) \lor p$ is valid

- $I \vDash \neg F \Rightarrow I \nvDash F$
- $I \nvDash \neg F \Rightarrow I \vDash F$
- $I \vDash F \land G \Rightarrow I \vDash F$ and $I \vDash G$
- $I \nvDash F \land G \Rightarrow I \nvDash F$ or $I \nvDash G$
- $I \vDash F \lor G \Rightarrow I \vDash F$ or $I \vDash G$
- $I \nvDash F \lor G \Rightarrow I \nvDash F$ and $I \nvDash G$
- $I \vDash F \rightarrow G \Rightarrow I \nvDash F$ or $I \vDash G$
- $I \nvDash F \rightarrow G \Rightarrow I \vDash F$ and $I \nvDash G$
- $I \vDash F$ and $I \nvDash F \Rightarrow I \vDash \bot$

Method 2: Semantic Argument Method

- Prove $F : \big(\neg(p \wedge q)\big) \vee p$ is valid

$$I \not\models \big(\neg(p \wedge q)\big) \vee p$$

$$I \not\models \neg(p \wedge q)$$
$$I \not\models p$$

$$I \models p \wedge q$$
$$I \not\models p$$

$$I \models p$$
$$I \models q$$
$$I \not\models p$$

$$I \models \bot$$

- $I \models \neg F \Rightarrow I \not\models F$
- $I \not\models \neg F \Rightarrow I \models F$
- $I \models F \wedge G \Rightarrow I \models F$ and $I \models G$
- $I \not\models F \wedge G \Rightarrow I \not\models F$ or $I \not\models G$
- $I \models F \vee G \Rightarrow I \models F$ or $I \models G$
- $I \not\models F \vee G \Rightarrow I \not\models F$ and $I \not\models G$
- $I \models F \rightarrow G \Rightarrow I \not\models F$ or $I \models G$
- $I \not\models F \rightarrow G \Rightarrow I \models F$ and $I \not\models G$
- $I \models F$ and $I \not\models F \Rightarrow I \models \bot$

Method 2: Semantic Argument Method

- Prove $F : \big((p \to q) \land (q \to r)\big) \to (p \to r)$ is valid

- $I \vDash \neg F \Rightarrow I \nvDash F$
- $I \nvDash \neg F \Rightarrow I \vDash F$
- $I \vDash F \land G \Rightarrow I \vDash F$ and $I \vDash G$
- $I \nvDash F \land G \Rightarrow I \nvDash F$ or $I \nvDash G$
- $I \vDash F \lor G \Rightarrow I \vDash F$ or $I \vDash G$
- $I \nvDash F \lor G \Rightarrow I \nvDash F$ and $I \nvDash G$
- $I \vDash F \to G \Rightarrow I \nvDash F$ or $I \vDash G$
- $I \nvDash F \to G \Rightarrow I \vDash F$ and $I \nvDash G$
- $I \vDash F$ and $I \nvDash F \Rightarrow I \vDash \bot$

Deciding Satisfiability and Validity

- Semantic argument method
  - Use deduction
  - Can be automated, applicable when domain is infinite
- Truth table method
  - Brute-force search, exponential, impractical, not applicable for infinite domains

Deciding Satisfiability and Validity

- Semantic argument method
  - Use deduction
  - Can be automated, applicable when domain is infinite
- Truth table method
  - Brute-force search, exponential, impractical, not applicable for infinite domains

- We can automate this process!
  - Modern SAT solvers combine search and deduction
  - NP-Complete, but modern SAT solvers (e.g., Microsoft z3) are working pretty well!

Microsoft Z3 Demo

- https://compsys-tools.ens-lyon.fr/z3/index.php
- Use SMT-LIB to express formulas
  - https://compsys-tools.ens-lyon.fr/z3/smt-lib-reference-v2.5-r2015-06-28.pdf
  - https://link.springer.com/content/pdf/bbm%3A978-3-662-50497-0%2F1.pdf

Microsoft Z3 Demo

- Prove $F : \big(\neg(p \land q)\big) \lor p$ is valid

```
; Declare variables
(declare-fun p () Bool)
(declare-fun q () Bool)

; Create constraints to be checked
(assert (or (not (and p q)) p))

; Check
; (check-sat)
; (get-model)
; (get-value (p q))
```

Microsoft Z3 Demo

- Prove $F : \big((p \to q) \land (q \to r)\big) \to (p \to r)$ is valid

```
; Variable declarations
(declare-fun p () Bool)
(declare-fun q () Bool)
(declare-fun r () Bool)

; Constraints
(assert (=> (and (=> p q) (=> q r)) (=> p r)))

; Solve
; (check-sat)
; (get-model)
; (get-value (p q r))
```

Propositional Logic Recap

- Syntax: Logical constants, propositional variables, logical connectives
- Semantics: Interpretation, truth value, evaluation
- Satisfiability and Validity
- Deciding Satisfiability and Validity