Lecture 2
Logic and Proof

David Garlan

Carnegie Mellon University

Models of Software Systems

15-671

Fall 1995

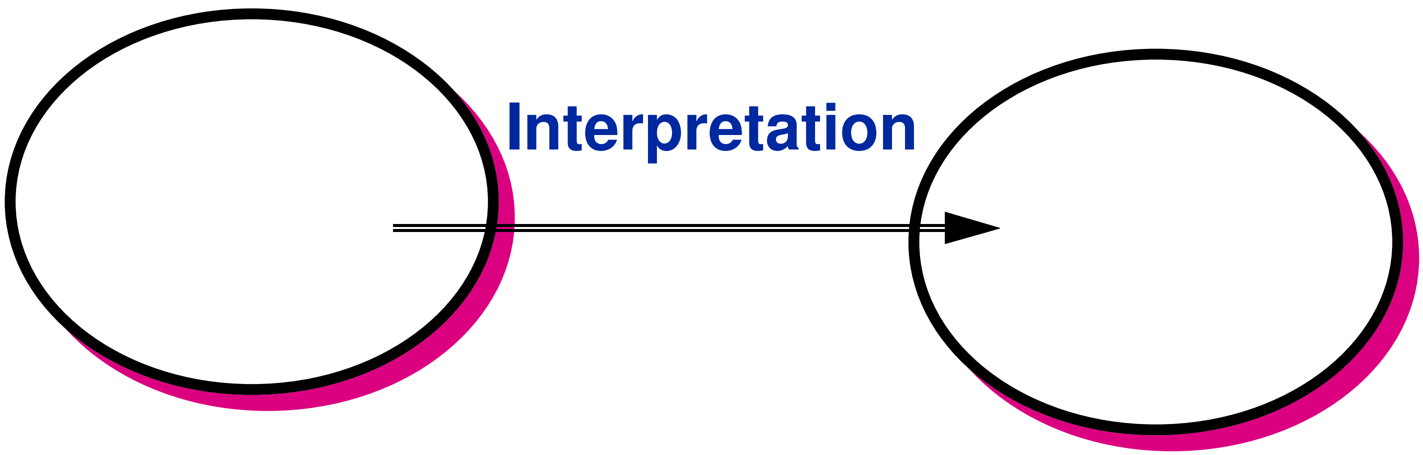
The Story So Far

- Syntax versus semantics
- Formal Systems
  > language: alphabet + grammar
  > inference system: axioms + rules of inference
  > theorem, proof, deduction
- Semantics
  > interpretation
  > consistency (soundness), completeness

Basic Elements

Syntax (language)

Semantics (meaning)



Inference system
(rules of reasoning)

Validity

Next Two Lectures

- **We will look at two specific formal systems:**
  - > **Propositional logic**
  - > **Predicate logic**
- **We will also briefly examine some of the main proof styles used for these systems**
  - > **Equational reasoning**
  - > **Proof by contradiction**
  - > **Proof by case analysis**
  - > **Natural deduction**

Propositions

- A proposition is a statement that is either true or false (but not both).
- Examples
  - Pittsburgh is the home of the Pirates
  - Pittsburgh is the capital of Pennsylvania
- The same proposition can often be expressed in several ways.
- Examples
  - five is bigger than four
  - four is less than five
  - $5 > 4$

(Qn: what's an example of a statement that is neither true nor false?)

Propositional Logic: Syntax

- **Alphabet: {P,Q,R, ..., $\wedge$, $\vee$, ~, $\Rightarrow$, $\Leftrightarrow$}**
- **Grammar:**

sentence = "P" | "Q" | "R" | ...
| "~", sentence
| "(", sentence, "$\vee$", sentence, ")"
| "(", sentence, "$\wedge$", sentence, ")"
| "(", sentence, "$\Rightarrow$", sentence, ")"
| "(", sentence, "$\Leftrightarrow$", sentence, ")"

we usually drop the parentheses when the resulting sentence is unambiguous

Propositional Logic: Semantics

- The *meaning* of a propositional wff is given as follows:
  > each primitive symbol (P, Q, R, ...) is given an interpretation as a proposition (with an associated truth value)

  example: P => today is Thursday, ...

  > the meaning (truth or falsity) of a compound wff is then determined by a set of standard tables:

  example:

p	q	$p \Rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

(NB 1: four interpretations)

(NB 2: this is the semantic domain)

  > In short, we use "truth tables"

Semantic Turnstyle

- **If a sentence W is true whenever a list of sentences P are true, we write P |= W**
- **We say W *is a semantic consequence of P.***
- **Example: $P \wedge Q \models P \vee Q$**
- **Note this is all in the semantic domain**

Terminology & Special Cases

- A sentence of the form
  - $A \wedge B$ is called a *conjunction*
    - A and B are *conjuncts*
  - $A \vee B$ is called a *disjunction*
    - A and B are *disjuncts*
  - $A \Rightarrow B$ is called an *implication*
    - A is the *antecedant*; B is the *consequent*
- Some sentences always interpret to true.
  - these are called *tautologies*
  - example: a sentence of the form $A \vee \sim A$, where A is any proposition
- Some sentences always interpret to false.
  - these are *contradictions,* or *inconsistencies*
  - example: a sentence of the form $A \wedge \sim A$, where A is any proposition

Propositional Calculus

- To complete our system we need a set of inference rules (called a calculus, here)
- One way to represent these rules is in the form:

$$\frac{\text{if we have proved these facts}}{\text{we can conclude these facts}}$$

- Examples

$$\frac{A \wedge B}{A} \qquad \frac{A}{A \vee B} \qquad \frac{A,\ A \Rightarrow B}{B}$$

- These are given names (resp):

  $\wedge$-elimination, $\vee$-introduction, $\Rightarrow$-elimination

Example

- **Prove: $(P \wedge Q) \vdash (P \vee Q)$**
- **Derivation:**
  - $> P \wedge Q$ — premise
  - $> P$ — $\wedge$-elimination
  - $> P \vee Q$ — $\vee$-introduction

About Proofs

- There are many styles (or techniques) of proof.
- SEM presents a style called “natural deduction”
- In this style a proof is constructed as a tree.
- However there are a lot of other styles that we will use as needed.
- Many of these center around the application a particular inference rule.

Natural Deduction

- Example of Derivation 2.4 and Theorem 2.1 done on board
- In practice, proof typically is constructed by working *backwords*, using the *form* of the goal as the guide to rule selection
  - Example: if have to prove $P \Rightarrow Q$, then reach for $\Rightarrow$ -introduction
- Use of indentation helps to keep assumption scopes clear

Some Other Proof Styles

- **Style 1: simple application of rules**
  - > we have already seen this a couple of times
  - > note for each step of proof write justification in terms of rule and lines on which it depends
- **Style 2: proof by contradiction**
  - > assume that the conclusion is not true and derive something that we know to be false
- **Style 3: proof by cases**
  - > break proof in to separate parts and then combine
- **Style 4: modus ponens**
  - > Show P and $P \Rightarrow Q$. Then conclude Q.

Relation to Natural Deduction

- Many of the inference rules for Natural Deduction are better known by the proof technique with which they are associated
  - > $\sim$-introduction = proof by contradiction
  - > $\vee$-elimination = proof by cases
  - > $\Rightarrow$-elimination = modus ponens

Equivalence

- We augment our formal system with a new symbol
- Two sentences are *equivalent* ($\equiv$) iff their truth value is the same under every interpretation.
- Example: $(P \Rightarrow Q) \equiv (\sim P \vee Q)$
- There is a collection of properties of equivalence, summarized on page 26 of SEM.
  - > These can be used in an "equational" style to do proof.
  - > Properties include usual associative, commutative, distributive properties of the connectors

Example

Prove: $\sim (P \wedge \sim Q) \equiv (Q \vee \sim P)$

Proof:

$\sim (P \wedge \sim Q)$

$\equiv$ DeMorgan

$\sim P \vee Q$

$\equiv$ $\vee$–Commutative

$Q \vee \sim P$

Relation Between Syntactic and Semantic Domain

- Propositional Calculus is Consistent
  - > That is, anything that can be proved it is valid
  - > If P |- W then P |= W
- Propositional Calculus is Complete
  - > That is, anything that can be is valid can be proven using the inference rules
  - > If P |= W then P |- W
- This allows us to shift between the two worlds
- In particular we can substitute $\Leftrightarrow$ for $\equiv$

To show |- P $\equiv$ Q we can demonstrate |= P $\equiv$ Q

Predicate Logic

- Sometimes want to assert that all elements of some set (S) satisfy a common property (P).
- For example if S == {Larry, Joe, Moe}, we might want to claim that all elements of S are stooges.
- We could write

  (Larry is a stooge) $\wedge$ (Joe is a stooge) $\wedge$ (Moe is a stooge)

- Alternatively we can create a template: Stooge( )

  Stooge(Larry) $\wedge$ Stooge (Joe) $\wedge$ Stooge (Moe)

- Stooge() is called a <u>predicate.</u>

Predicates (2)

- But for large sets this becomes unwieldy (or impossible).
- So we use a shorthand:

  $\forall$ x:S • Stooge(x)

- Similarly, to assert the existence of an element of some set (S) that satisfies a property (P) we use

  $\exists$ x:S • P(x)

  which is shorthand for

  P(s1) $\vee$ P(s2) $\vee$ P(s3) $\vee$ ...

Predicates (3)

- For the moment we will assume that the sets that are quantified over are understood by context

  examples: $\forall x \bullet P(x)$, $\exists x \bullet Q(x)$

- You can think of a predicate as a boolean valued function: when applied to an argument it returns true or false.
- Predicates may have n arguments
  - examples: >, is-pythagorian

Predicate Logic: Syntax

- **Alphabet: as with propositional logic, but includes quantifiers (∀, ∃) and •**
  - > **∀ is called a "universal" quantifier**
  - > **∃ is called an "existential" quantifier**
- **Grammar: see SEM**
  - > **but note *every* quantified expression has a •**
- **Can nest quantified expressions**
  - > **∀ x • (∀ y • (∃ z • P(x,y,z)))**
  - > **usually leave out the parens**
  - > **combine adjacent variables with same quantifier.**
  - > **so this becomes ∀ x, y • ∃ z • P(x,y,z)**

Predicate Logic: Semantics

- More involved than propositional logic
- Basic idea is that sentence

  $\forall$ x • P(x) is true iff P is true of all values of x, and

  $\exists$ x • Q(x) is true iff Q is true for some value of x

Predicate Calculus

- As with Propositional Logic we introduce a set of deductive rules.
- Similar to situation before
  - > SEM uses natural deduction style

Equality

- We now add one more symbol to Predicate Logic: = to represent that two entities are the same.
- This is equivalent to having a special predicate "equals", but we use the more common infix notation.
- We also introduce several new rules to our deductive apparatus.
- The most important of these is:

Substitution rule:

$$\frac{m = n,\ S(n)}{S[m/n]}$$

Theories

- We can do a lot with just logic.
- But sometimes we want to introduce special axioms to describe a specific phenomenon.
- Then we are interested in the consequences of those axioms (given the other rules of deduction).
- These consequences are called a theory.

Examples

- Some general theories that we will encounter in this course
  - > set theory
  - > theory for sequences
  - > theory for concurrent processes
- Other theories we will make up on the spot to characterize a software system
  - > example: theory of telephones for a switching system
  - > example: theory of devices and event processing for real-time system
  - > homework problems explore some of these