Lecture 03: Predicates and Quantifiers

Jingjin Yu | Computer Science @ Rutgers

RUTGERS THE STATE UNIVERSITY OF NEW JERSEY

September 11, 2020

Algorithmic Robotics & Control Lab

Outline

- A brief review of lectures 01-02
- Predicate logic
  - Predicates
  - Preconditions and postconditions
- Quantifiers
  - The universal quantifier
  - The existential quantifier
  - Binding variables
- Logical equivalences involving quantifiers
- Negating statements with quantifiers

- A repeating note: **make sure you read the textbook**!

L01: Propositional Logic Basics

- Proposition: a declarative sentence, true or false, not both
- Logical connectives/operators (in truth table format)

p	q	negation $\neg p$	conjunction $p \land q$	disjunction $p \lor q$	conditional $p \to q$	bidirectional $p \leftrightarrow q$
T	T					
T	F					
F	T					
F	F					

- Evaluation of compound statement using truth table

p	q	$\neg q$	$p \to \neg q$	$p \land q$	$(\boldsymbol{p} \to \neg \boldsymbol{q}) \to (\boldsymbol{p} \land \boldsymbol{q})$
T	T				
T	F				
F	T				
F	F				

L02: Logical Equivalences

- **Def^n**: Propositions p and q are **logically equivalent**, denoted by $p \equiv q$, if $p \leftrightarrow q$ is a tautology (always true)
  - May prove using truth tables when # of variables are small
  - Can also use proven equivalences to do the proof
- **Ex**: using truth table
  - $\neg(p \wedge q) \equiv \neg p \vee \neg q$
  - $\neg(p \vee q) \equiv \neg p \wedge \neg q$
- **Ex**: using equivalences
  - $\neg(p \rightarrow q) \equiv p \wedge \neg q$

p	q	$\neg p$	$\neg q$	$p \wedge q$	$\neg(p \wedge q)$	$\neg p \vee \neg q$
T	T					
T	F					
F	T					
F	F					

L02: Commonly Used Equivalences

Name	Equivalence
Identity laws	$p \wedge T \equiv p$, $p \vee F \equiv p$
Domination laws	$p \vee T \equiv T$, $p \wedge F \equiv F$
Idempotent laws	$p \vee p \equiv p$, $p \wedge p \equiv p$
Double negation law	$\neg(\neg p) \equiv p$
Commutative laws	$p \vee q \equiv q \vee p$, $p \wedge q \equiv q \wedge p$
Associative laws	$(p \vee q) \vee r \equiv p \vee (q \vee r)$, $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$
Distributive laws	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$ $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
De Morgan's laws	$\neg(p \vee q) \equiv \neg p \wedge \neg q$, $\neg(p \wedge q) \equiv \neg p \vee \neg q$
Absorption laws	$p \vee (p \wedge q) \equiv p$, $p \wedge (p \vee q) \equiv p$
Negation laws	$p \vee \neg p \equiv T$, $p \wedge \neg p \equiv F$

Equivalence Containing Conditionals
$p \to q \equiv \neg p \vee q$
$p \to q \equiv \neg q \to \neg p$
$p \vee q \equiv \neg p \to q$
$p \wedge q \equiv \neg(p \to \neg q)$
$\neg(p \to q) \equiv p \wedge \neg q$
$(p \to q) \wedge (p \to r) \equiv p \to (q \wedge r)$
$(p \to r) \wedge (q \to r) \equiv (p \vee q) \to r$
$(p \to q) \vee (p \to r) \equiv p \to (q \vee r)$
$(p \to r) \vee (q \to r) \equiv (p \wedge q) \to r$

Equivalence Containing Bidirectionals
$p \leftrightarrow q \equiv (p \to q) \wedge (q \to p)$
$p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$
$p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$
$\neg(p \leftrightarrow q) \equiv p \leftrightarrow \neg q$

Problems with Propositional Logic

- Propositional logic has very limited **expressiveness**.
  - **Ex:**
    - p = "Every computer on campus is functioning properly."
    - q = "math3 (a computer's name) is functioning properly."
    - $p \rightarrow q$ holds, but not by **reasoning** using **propositional logic**
  - What is the problem?
    - Every proposition involves **specific** objects
    - There is way to related p and q logically using propositional logic
    - There is no concept of "some" nor "all
- We need more **expressive** logics (logical systems)

Predicates

- **Def^n**: A **predicate** is a partial proposition where one or more variables replace part of a propositional statement.
  - **Ex:**
    - $x > 3$ is a predicate
    - May write as $P(x)$, i.e., $P(x) =$ "$x > 3$"
    - A proposition can be obtained by replacing x with a number
      - $P(4)$ would be true
      - $P(2)$ would be false
      - But, $P(x)$ applies to **infinite** x's

Usually, use P, Q, R, ... for predicates

- $P(x)$ is also said to be the value of the **proposition function** $P(\cdot)$ at x
- **Def^n**: A predicate is **bound** if its variables are fixed
  - Once bounded, it becomes a full proposition
  - Bound predicates' truth values are fixed

Predicates, cont.

- **Ex**: "Computer x is under attack"
  - Predicate: $Attack(x)$
    - If $MATH6$ is under attack, we have $Attack(MATH6) = true$
    - If $CS2$ is not under attack, we have $Attack(CS2) = false$

- Predicates can be ***k*-nary** (unary, binary, ternary, …). **Ex**:
  - $x^2 + y^2 = 1$ is binary, can be written as $P(x, y)$
  - $x + y = z$ is ternary, can be written as $Q(x, y, z)$
  - Truth value of $P(1,1)$?
  - Truth value of $P(\sin 60, \cos 60)$?
  - Truth value of $Q(-100, 100, 0)$?
  - Truth value of $Q(3, 4, 5)$?

Preconditions and Postconditions

- You may have seen preconditions and postconditions in computer programs, in particular "unit tests". They can have the format $assert(x > 10)$. These help ensure the correctness of computer programs. We can represent these using predicates. **Ex**:
  - $sqrt(x)$
    - Precondition: $x \geq 0$
    - Postcondition: $x \geq 0; y = sqrt(x); y^2 = x$
- Preconditions and postconditions are used to establish the correctness of computer programs
- Read Example 7 in the textbook for more information
- Not a super essential point for this class

Quantifiers

- **Quantification** expresses the extent to which a predicate is true over a range of elements (i.e., choices of variable values). **Ex**:
  - **All** humans are mortal
  - On **some** days, it rains
  - More on these details later

- **Predicate logic** is the area of logic that deals with predicates and quantifiers
- There are many predicate logics. **Ex**:
  - First order: has predicates, but not predicates of predicates
    - I.e., no $P(Q(x))$
  - Second order
  - …

Universal Quantification

- **Def^{n}**: The **universal quantification** of $P(x)$ is the proposition "$P(x)$ for all x in the domain" and written as $\forall x\ P(x)$. If for a fixed x_0, $P(x_0) = false$, then x_0 is a **counterexample** of $\forall x\ P(x)$.
  - We cannot give $true/false$ to a predicate $P(x)$
  - But $\underline{\forall x\ P(x)}$ can be assigned a truth value
  - To evaluate $\forall x\ P(x)$:
    - $\forall x\ P(x) = true$ when $P(x)$ holds for all x in the domain
    - $\forall x\ P(x) = false$ if there is an x such that $P(x)$ is $false$

- **Ex**: $x \in \mathbb{R}$, predicate $Q(x) =$ "$x + 1 > x$"
  - For any $x \in \mathbb{R}$, it is clear $x + 1 > x$ holds
  - So we can write $\forall x\ Q(x) = true$

Universal Quantification, cont.

- **Ex**: "All humans are mortal"
  - Can write as $\forall x\ P(x)$, which is true
    - x refers to a person
    - $P(x) =$ "x is mortal"
- **Ex**: $\forall x \in \mathbb{R}, x \geq 2$
  - This is false because if $x = 1$, then $1 \geq 2$ is false
  - **One counterexample** is enough to show $\forall x\ P(x) = false$
- **Ex**: $R(x) = "x^2 < 10"$ with domain $x \in \{1, 2, 3, 4\}$
  - $\forall x\ R(x) = false$, because $R(4)$ is false
- If domain is finite, e.g., $X = \{x_1, \dots, x_n\}$, then
  - $\forall x\ P(x) \equiv P(x_1) \wedge P(x_2) \wedge \cdots \wedge P(x_n)$
  - **Ex**: $\forall x\ R(x) \equiv R(1) \wedge R(2) \wedge R(3) \wedge R(4)$
- $\forall x\ P(x) = true$ when the domain is empty
  - **Ex**: "All immortals fly"

Existential Quantification

- **Def$^{\boldsymbol{n}}$**: The **existential quantification** of $P(x)$ is the proposition "There exists an element x in the domain such that $P(x)$" and written as $\exists x P(x)$**.**
  - Equivalent statements:
    - "There is an x such that $P(x)$ (is true)"
    - "There is at least one x such that $P(x)$"
    - "For some x, $P(x)$"
  - To evaluate $\exists x P(x)$:
    - $\exists x P(x) = true$ if there is at least one x in the domain such that $P(x)$ is true
    - $\exists x P(x) = false$ if $P(x) = false$ for every single x

Existential Quantification, cont.

- **Ex**: $P(x) =$ "$x \geq 2$" and $x \in \mathbb{R}$
  - $\exists x\, P(x) = true$ because if we let $x = 2 \in \mathbb{R}$, $P(x) = true$
  - We had this before with "∀", which was $false$ then
- **Ex**: $P(x) =$ "$x = x + 1$" and $x \in \mathbb{R}$
- For a finite domain $X = \{x_1, \dots, x_n\}$, we have

$$\exists x\, P(x) \equiv P(x_1) \vee \cdots \vee P(x_n)$$

- **Ex**: $P(x) =$ "$x^2 < 10$", $x \in \{1, 2, 3, 4\}$
  - $\exists x\, P(x) \equiv (1^2 < 10) \vee (2^2 < 10) \vee (3^2 < 10) \vee (4^2 < 10)$
  - This is now true with the existential quantifier
- **Uniqueness quantifier ∃!**
  - $\exists!\, x\, P(x)$: "There is exactly one x such that $P(x)$"
  - **Ex**: $\exists!\, x\, (x^2 = 0)$

Quantifier with Restricted Domains

- Sometimes quantifiers have restricted domains, which can be simplified. **Ex**:
  - $\forall x < 0\ (x^2 > 0)$
  - $\forall x \neq 0\ (x^3 \neq 0)$
  - $\exists x > 0\ (x^2 = 2)$

- For universal quantifications, e.g., $\forall x < 0\ (x^2 > 0)$, they can be simplified to a conditional

$$\forall x < 0\ (x^2 > 0) \equiv \forall x\ (x < 0 \rightarrow x^2 > 0)$$

- For existential quantifications, e.g., $\exists x > 0\ (x^2 = 2)$, they can be simplified to a conjunction

$$\exists x > 0\ (x^2 = 2) \equiv \exists x\ (x > 0 \land x^2 = 2)$$

Quantifier Precedence & Binding Variables

- Quantifiers have higher precedence
  - $\forall, \exists > \neg > \wedge > \vee > \rightarrow > \leftrightarrow$
- **Ex**: $\forall x P(x) \wedge Q(x) \equiv (\forall x P(x)) \wedge Q(x)$
  - Note that this is not the same as $\forall x (P(x) \wedge Q(x))$
- **Bound** and **free** variables. **Ex**:
  - $\exists x > 0\ (x^2 > 0)$
  - $\exists x\ (x + y = 1)$
  - $\exists x (P(x) \wedge Q(x)) \vee \forall x R(x)$
    - Same as $\exists x (P(x) \wedge Q(x)) \vee \forall y R(y)$

Logical Equivalence Involving Quantifiers

- $\mathbf{Def^n}$: Two statements S_1 and S_2 are **logically equivalent** if they have the same truth value regardless of which predicates are substituted into these statements and which domain of discourse is used for the variables in these propositional functions. When equivalence holds, it is denoted $S_1 \equiv S_2$

- A new "proof" approach showing $S_1 \equiv S_2$
  - Show $S_1 \rightarrow S_2 = true$ and $S_2 \rightarrow S_1 = true$
  - Therefore, $(S_1 \rightarrow S_2) \land (S_2 \rightarrow S_1) = true$
  - We know $(S_1 \rightarrow S_2) \land (S_2 \rightarrow S_1) \equiv S_1 \leftrightarrow S_2$
  - Therefore, $S_1 \leftrightarrow S_2 = true$, a tautology, so $S_1 \equiv S_2$

Example: $\forall x\big(P(x) \wedge Q(x)\big) \equiv \forall x P(x) \wedge \forall x Q(x)$

Negating Statements with Quantifiers

- **Ex**: Every student in CS 205 has taken Calc II.
  - $\forall x\, P(x)$ = "for every student x in CS 205, x took Calc II"
  - Negation: $\neg(\forall x\, P(x))$
    - English: **it is not the case** that every student in CS 205 took Calc II
    - Equivalent: some student(s) in CS 205 did not take Calc II
    - Equivalent: $\exists x\, (\neg P(x))$

- So we have (De Morgan's laws for quantifiers)
  - $\neg\big(\forall x\, P(x)\big) \equiv \exists x\, (\neg P(x))$
  - $\neg\big(\exists x\, P(x)\big) \equiv \forall x\, (\neg P(x))$
  - Finite domain $X = \{x_1, \dots, x_n\}$:
    - $\neg\big(\forall x\, P(x)\big) = \neg\big(P(x_1) \wedge \cdots \wedge P(x_n)\big) = \neg P(x_1) \vee \cdots \vee \neg P(x_n) = \exists x\, (\neg P(x))$
    - $\neg\big(\exists x\, P(x)\big) = \neg\big(P(x_1) \vee \cdots \vee P(x_n)\big) = \neg P(x_1) \wedge \cdots \wedge \neg P(x_n) = \forall x\, (\neg P(x))$

English Translations

- **Ex**: There is a completely honest politician
  - $P(x) =$ "politician x is completely honest"
  - Quantified statement: $\exists x\, P(x)$
  - Negation: $\neg\big(\exists x\, P(x)\big) = \forall x\, (\neg P(x))$
  - Translation: "all politicians are somewhat dishonest"