PL - Semantics and Truth Tables

HE Mingxin, Max

CS104: program07 @ yeah.net

CS108: mxhe1 @ yeah.net

I2ML(H) Spring 2023 (CS104|CS108)

Exercises 04 : Reading and More

Record your time spent (in 0.1 hours) with brief tasks and durations in your learning log by hand writing!

1) Read textB-ch01-1.4-SemanticsOfPL.pdf (in 2 weeks).
2) Finish Assignment 1 on sakai before 1.00pm, March 12.
3) Assignment 2 is to be launched ...

Topic 4.1

Semantics - Meaning of the Formulas

Truth Values

We denote the set of truth values as $\mathcal{B} \triangleq \{0, 1\}$.

0 and 1 are only distinct objects without any intuitive meaning.

We may view 0 as false and 1 as true, but it is only our emotional response to the symbols.

Model

Definition 4.1

A model is an element of $\mathrm{Vars} \to \mathcal{B}$.

Example 4.1

$\{p_1 \mapsto 1, p_2 \mapsto 0, p_3 \mapsto 0, \ldots\}$ *is a model*

Since Vars is countably infinite, the set of models is non-empty and infinite.

A model m may or may not satisfy a formula F.
The satisfaction relation is usually denoted by $m \models F$ in infix notation.

Propositional Logic Semantics

Definition 4.2

The satisfaction relation $\models$ between models and formulas is the smallest relation that satisfies the following conditions.

- $m \models \top$
- $m \models p$ *if* $m(p) = 1$
- $m \models \neg F$ *if* $m \not\models F$
- $m \models F_1 \vee F_2$ *if* $m \models F_1$ *or* $m \models F_2$,   *at least one, may be both*
- $m \models F_1 \wedge F_2$ *if* $m \models F_1$ *and* $m \models F_2$,   *just both*
- $m \models F_1 \oplus F_2$ *if* $m \models F_1$ *or* $m \models F_2$,   *but not both*
- $m \models F_1 \Rightarrow F_2$ *if if* $m \models F_1$ *then* $m \models F_2$
- $m \models F_1 \Leftrightarrow F_2$ *if* $m \models F_1$ *iff* $m \models F_2$

Thinking Exercise 4.1

Why $\perp$ is not explicitly mentioned in the above definition?

Example: Satisfaction Relation

Example 4.2

Consider model $m = \{p_1 \mapsto 1, p_2 \mapsto 0, p_3 \mapsto 0, \ldots\}$ *and formula* $(p_1 \Rightarrow (\neg p_2 \Leftrightarrow (p_1 \wedge p_3)))$

$$\begin{tikzcd}
 & m \not\models (p_1 \Rightarrow (\neg p_2 \Leftrightarrow (p_1 \wedge p_3))) \arrow[dl] \arrow[dr] & & & \\
m \models p_1 & & m \not\models (\neg p_2 \Leftrightarrow (p_1 \wedge p_3)) \arrow[dl] \arrow[dr] & & \\
 & m \models \neg p_2 \arrow[d] & & m \not\models (p_1 \wedge p_3) \arrow[dl] \arrow[dr] & \\
 & m \not\models p_2 & m \models p_1 & & m \not\models p_3
\end{tikzcd}$$

Thinking Exercise 4.2

Formally, write the satisfiability checking procedure .

Remark:

$m \models$ *might be T or 1,*

$m \not\models$ *might be F or 0.*

Satisfiable, Valid, Unsatisfiable

We say

- m *satisfies* F if $m \models F$,
- F is *satisfiable* if there is a model m such that $m \models F$,
- F is *valid* (written $\models F$) if for each model m $m \models F$, and
- F is *unsatisfiable* (written $\not\models F$) if there is no model m such that $m \models F$.

Thinking Exercise 4.3

If F is sat then $\neg F$ is_____.
If F is valid then $\neg F$ is_____.
If F is unsat then $\neg F$ is_____.

A valid formula is also called a tautology.

Overloading $\models$: set of models

We extend the usage of $\models$ in the following natural ways.

Definition 4.3

Let M be a (possibly infinite) set of models. $M \models F$ if for each $m \in M$, $m \models F$.

Example 4.3

$\{\{p \to 1, q \to 1\}, \{p \to 1, q \to 0\}\} \models p \lor q$

Thinking Exercise 4.4

Which of the following hold?

- $\{\{p \to 1, q \to 1\}, \{p \to 0, q \to 0\}\} \models p$
- $\{\{p \to 1, q \to 1\}\} \models p \land q$
- $\{\{p_i \to (k = i) | i \in \mathbb{N}\} | k \in \mathbb{N}\} \models p_1$

Overloading $\models$: set of formulas

Definition 4.4

Let Σ be a (possibly infinite) set of formulas.
$\Sigma \models F$ *if for each model m that satisfies each formula in Σ, $m \models F$.*

- $\Sigma \models F$ is read Σ entails (logically implies) F.
- If $\{G\} \models F$ then we may write $G \models F$.

Thinking Exercise 4.5

Which of the following hold?

- $\{p, q\} \models p \wedge q$
- $\{p \Rightarrow q, q \Rightarrow p\} \models p \Leftrightarrow q$
- $\{p \Rightarrow q, q\} \models p \oplus q$
- $\{p \Rightarrow q, \neg q, p\} \models p \oplus q$

Example 4.4

$\{p, q\} \models p \vee q$

Commentary: If Σ is finite, the definition of $\Sigma \models F$ means $\bigwedge \Sigma \Rightarrow F$ is valid. Why are we inventing a new notation? Because, Σ can be an infinite set. $\wedge$ is not applicable on an infinite set. (why?)

Equivalent

Definition 3.5

Let $F \equiv G$ *if for each model* m

$$m \models F \text{ iff } m \models G.$$

Example 3.5

$(p \vee q) \vee r \equiv p \vee (q \vee r)$

Equisatisfiable and Equivalid

Definition 4.6
Formulas F and G are equisatisfiable if

$$F \text{ is sat} \quad \text{iff} \quad G \text{ is sat.}$$

Definition 4.7
Formulas F and G are equivalid if

$$\models F \quad \text{iff} \quad \models G.$$

Commentary: The concept of equisatisfiable is used in formula transformations. We often say that after a transformation the formula remained equisatisfiable. Equivalid is the dual concept, rarely used in practice.

Topic 3.2

Decidability of SAT

Notation alert: decidable

A problem is decidable if there is an algorithm to solve the problem.

Propositional satisfiability problem

The following problem is called the satisfiability problem

$$\text{For a given } F \in \mathrm{P}\text{, is } F \text{ satisfiable?}$$

Theorem 4.1

The propositional satisfiability problem is decidable.

Proof.

Let $n = |\mathrm{Vars}(F)|$.
We need to enumerate 2^n elements of $\mathrm{Vars}(F) \to \mathcal{B}$.

If any of the models satisfy the formula, then F is sat. Otherwise, F is unsat. $\square$

Thinking Exercise 4.6*

Give a procedure to decide the validity of a formula.

Complexity of the decidability question?

- If we enumerate all models to check satisfiability, the cost is exponential
- We do not know if we can do better.
- However, there are several tricks that have made satisfiability checking practical for the real-world formulas.

Topic 3.3

Truth Tables

Truth Tables

Truth tables was one of the methods to decide propositional logic.

The method is usually presented in slightly different notation. We need to assign a truth value to every formula.

Truth Function

A model m is in $\mathrm{Vars} \to \mathcal{B}$.

We can extend m to $\mathrm{P} \to \mathcal{B}$ in the following way.

$$m(F) = \begin{cases} 1 & m \models F \\ 0 & \textit{otherwise}. \end{cases}$$

The extended m is called truth function.

Since truth functions are natural extensions of models, we did not introduce new symbols.

Truth functions for logical connectives

Let F and G be logical formulas, and m be a model.
Due to the semantics of the propositional logic, the following holds for the truth functions.

$m(F)$	$m(\neg F)$
0	1
1	0

$m(F)$	$m(G)$	$m(F \wedge G)$	$m(F \vee G)$	$m(F \oplus G)$	$m(F \Rightarrow G)$	$m(F \Leftrightarrow G)$
0	0	0	0	0	1	1
0	1	0	1	1	1	0
1	0	0	1	1	0	0
1	1	1	1	0	1	1

Truth Table

For a formula F, a truth table consists of $2^{|\mathrm{Vars}(F)|}$ rows. Each row considers one of the models and computes the truth value of F for each of them.

Example 4.6

Consider $(p_1 \Rightarrow (\neg p_2 \Leftrightarrow (p_1 \wedge p_3)))$. *We will not write* $m(.)$ *in the top row for brevity.*

p_1	p_2	p_3	$(p_1$	$\Rightarrow$	$(\neg$	p_2	$\Leftrightarrow ($	p_1	$\wedge$	p_3 $)))$
0	0	0	0	1	1	0	0	0	0	0
0	0	1	0	1	1	0	0	0	0	1
0	1	0	0	1	0	1	1	0	0	0
0	1	1	0	1	0	1	1	0	0	1
1	0	0	1	0	1	0	0	1	0	0
1	0	1	1	1	1	0	1	1	1	1
1	1	0	1	1	0	1	1	1	0	0
1	1	1	1	0	0	1	0	1	1	1

The column under the leading connective has 1s therefore the formula is sat. But, there are some 0s in the column therefore the formula is not valid.

Example : DeMorgan Law

Example 4.7

Let us show $p \lor q \equiv \neg(\neg p \land \neg q)$.

p	q	$(p \lor q)$	$\neg$	$(\neg$	p	$\land$	$\neg$	$q)$
0	0	0	0	1	0	1	1	0
0	1	1	1	1	0	0	0	1
1	0	1	1	0	1	0	1	0
1	1	1	1	0	1	0	0	1

Since the truth values of both the formulas are same in each row, the formulas are equivalent.

Thinking Exercise 4.7

Show $p \land q \equiv \neg(\neg p \lor \neg q)$ *using a truth table.*

Commentary: $p \land q \equiv \neg(\neg p \lor \neg q)$ and $p \lor q \equiv \neg(\neg p \land \neg q)$ are called DeMorgan law.

Example : definition of $\Rightarrow$

Example 4.8

Let us show $p \Rightarrow q \equiv (\neg p \lor q)$.

p	q	$(p \Rightarrow q)$	$(\neg$	p	$\lor$	$q)$
0	0	1	1	0	1	0
0	1	1	1	0	1	1
1	0	0	0	1	0	0
1	1	1	0	1	1	1

Since the truth values of both the formulas are same in each row, the formulas are equivalent.

It appears that $\Rightarrow$ is a redundant symbol. We can write it in terms of the other symbols.

Question

Why do we assign "1" to $p \to q$ whenever p is "0"?

Simply we can say that "anything you want follows from falsity".

$$\frac{\perp}{A}\ \perp\mathrm{E}$$

We may think of it in another way from a special case ...

Question

Why do we assign “1” to $p \to q$ whenever p is “0”?

Want

p	q	$p \wedge q \to^* p$
1	1	1
1	0	1
0	1	1
0	0	1

Question

Why do we assign "1" to $p \to q$ whenever p is "0"?

Suppose

p	q	$p \wedge q$	
1	1	1	
1	0	0	
0	1	0	
0	0	0	

Question

Why do we assign "1" to $p \to q$ whenever p is "0"?

Then

p	q	$p \wedge q$	$p \to^* q$
1	1	1	1
1	0	0	0
0	1	0	a
0	0	0	b

Question

Why do we assign "1" to $p \to q$ whenever p is "0"?

And

p	q	$p \wedge q$	$p \to^* q$	$p \wedge q \to^* p$
1	1	1	1	1
1	0	0	0	a
0	1	0	a	b
0	0	0	b	b

Question

Why do we assign "1" to $p \to q$ whenever p is "0"?

Hence

p	q	$p \wedge q$	$p \to^* q$	$p \wedge q \to^* p$
1	1	1	1	1
1	0	0	0	a
0	1	0	$a = 1$	b
0	0	0	$b = 1$	b

Example : definition of $\Leftrightarrow$

Example 4.9

Let us show $p \Leftrightarrow q \equiv (p \Rightarrow q) \land (q \Rightarrow p)$.

p	q	$(p \Leftrightarrow q)$	$(p$	$\Rightarrow$	$q)$	$\land$	$(q$	$\Rightarrow$	$p)$
0	0	1	0	1	0	1	0	1	0
0	1	0	0	1	1	0	1	0	0
1	0	0	1	0	0	0	0	1	1
1	1	1	1	1	1	1	1	1	1

Example: definition $\oplus$

Example 4.10

Let us show $(p \oplus q) \equiv (\neg p \wedge q) \vee (p \wedge \neg q)$ *using truth table.*

p	q	$(p \oplus q)$	$(\neg$	p	$\wedge$	$q)$	$\vee$	$(p$	$\wedge$	$\neg$	$q)$
0	0	0	1	0	0	0	0	0	0	1	0
0	1	1	1	0	1	1	1	0	0	0	1
1	0	1	0	1	0	0	1	1	1	1	0
1	1	0	0	1	0	1	0	1	0	0	1

Thinking Exercise 4.8

Show $(p \oplus q) \equiv (\neg p \vee \neg q) \wedge (p \vee q)$.

Example: associativity

Example 4.11

Let us show $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$

p	q	r	$(p$	$\wedge$	$q)$	$\wedge$	r	p	$\wedge$	$(q$	$\wedge$	$r)$
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1	0	0	0	0	1
0	1	0	0	0	1	0	0	0	0	1	0	0
0	1	1	0	0	1	0	1	0	0	1	1	1
1	0	0	1	0	0	0	0	1	0	0	0	0
1	0	1	1	0	0	0	1	1	0	0	0	1
1	1	0	1	1	1	0	0	1	0	1	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1

Exercise: associativity

Thinking Exercise 4.9

Prove/disprove using truth tables

- $(p \vee q) \vee r \equiv p \vee (q \vee r)$
- $(p \oplus q) \oplus r \equiv p \oplus (q \oplus r)$
- $(p \Leftrightarrow q) \Leftrightarrow r \equiv p \Leftrightarrow (q \Leftrightarrow r)$
- $(p \Rightarrow q) \Rightarrow r \equiv p \Rightarrow (q \Rightarrow r)$

Exercise: distributivity

Thinking Exercise 4.10

Prove/disprove using truth tables prove that $\wedge$ distributes over $\vee$ and vice-versa.

- $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
- $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$

Tedious Truth Tables

- We need to write 2^n rows even if a simple observation about the formula can prove (un)satisfiability.

  For example,
  - $(a \vee (c \wedge a))$ is sat (why? - no negation)
  - $(a \vee (c \wedge a)) \wedge \neg(a \vee (c \wedge a))$ is unsat (why?- contradiction at the top level)

- We should be able to take such shortcuts?

We saw and will see methods that will allow us to take such shortcuts.

Topic 4.4

Expressive Power of Propositional Logic

Boolean functions

A finite boolean function is in $\mathcal{B}^n \to \mathcal{B}$.

A formula F with $\mathrm{Vars}(F) = \{p_1, \dots, p_n\}$ can be viewed as a Boolean function f that is defined as follows.

$$\text{for each model } m, f(m(p_1), \dots, m(p_n)) = m(F)$$

We say F represents f.

Example 4.12

Formula $p_1 \vee p_2$ represents the following function

$$f = \{(0,0) \to 0, (0,1) \to 1, (1,0) \to 1, (1,1) \to 1\}$$

A Boolean function is another way of writing truth table.

Expressive power

Theorem 4.2

For each finite boolean function f, there is a formula F that represents f.

Proof.

Let $f : \mathcal{B}^n \to \mathcal{B}$. We construct a formula F to represent f.

Let $p_i^0 \triangleq \neg p_i$ and $p_i^1 \triangleq p_i$.

$$\text{For } (b_1, \ldots, b_n) \in \mathcal{B}^n, \text{ let } \quad F_{(b_1,\ldots,b_n)} \triangleq \begin{cases} (p_1^{b_1} \wedge \cdots \wedge p_n^{b_n}) & \text{if } f(b_1, \ldots, b_n) = 1 \\ \bot & \text{otherwise.} \end{cases}$$

$$F \triangleq \underbrace{F_{(0,\ldots,0)} \vee \cdots \vee F_{(1,\ldots,1)}}_{\text{All Boolean combinations}}$$

We used only three logical connectives to construct F

Thinking Exercise 4.11

Workout if F really represents f.

Insufficient expressive power

If we do not have sufficiently many logical connectives, we cannot represent all Boolean functions.

Example 4.13

$\wedge$ alone can not express all boolean functions.

To prove this we show that Boolean function $f = \{0 \rightarrow 1, 1 \rightarrow 1\}$ can not be achieved by any combination of $\wedge$s.

We setup induction over the sizes of formulas consisting a variable p and $\wedge$.

Commentary: We are assuming that only one variable occurs in the formula, since there is exactly one input to f. Our definition of "represents" requires the number of variables must match the arity of the function.

Insufficient expressive power II

base case:
Only choice is p.(why?) For $p = 0$, the function does not match.

induction step:
Let us assume that formulas F and G of size less than $n-1$ do not represent f.
We can construct a longer formula in the following way.

$$(F \wedge G)$$

The formula does not represent f, because we can always pick(why?) a model when F or G produces 0.

Therefore $\wedge$ alone is not expressive enough.

Minimal logical connectives

We used

- 2 0-ary,
- 1 unary, and
- 5 binary

connectives to describe the propositional logic.

However, it is not the minimal set needed for the maximum expressivity.

Example 4.14

$\neg$ and $\vee$ can define the whole propositional logic.

- $\top \equiv p \vee \neg p$ *for some* $p \in \mathrm{Vars}$
- $\bot \equiv \neg \top$
- $(p \wedge q) \equiv \neg(\neg p \vee \neg q)$
- $(p \oplus q) \equiv (p \wedge \neg q) \vee (\neg p \wedge q)$
- $(p \Rightarrow q) \equiv (\neg p \vee q)$
- $(p \Leftrightarrow q) \equiv (p \Rightarrow q) \wedge (q \Rightarrow p)$

Thinking Exercise 4.12

a. Show $\neg$ and $\wedge$ can define all the other connectives

b. Show $\oplus$ alone can not define $\neg$

Topic 3.5

Problems

Truth Tables

Thinking Exercise 4.13

Prove/disprove validity of the following formulas using truth tables.

1. $(p \Rightarrow (q \Rightarrow r)) \Leftrightarrow ((p \wedge q) \Rightarrow r))$
2. $p \wedge (q \oplus r) \Leftrightarrow (p \wedge q) \oplus (q \wedge r)$
3. $(p \vee q) \wedge (\neg q \vee r) \Leftrightarrow (p \vee r)$
4. $\perp \Rightarrow F$ *for any* F

Expressive power

Thinking Exercise 4.14

Show $\neg$ *and* $\oplus$ *is not as expressive as propositional logic.*

Thinking Exercise 4.15

Prove/disprove that the following subsets of connectives are fully expressive.

- $\lor, \oplus$
- $\bot, \oplus$
- $\Rightarrow, \oplus$
- $\lor, \land$
- $\Rightarrow, \bot$

Expressive power(2)

Thinking Exercise 4.16
Prove/disprove: if-then-else is fully expressive.

Thinking Exercise 4.17
Show $\Rightarrow$ alone can not express all the Boolean functions.

All Minimal Combinations*

Thinking Exercise 4.18

List all minimal subsets of the logical connectives that are fully expressive.

Recall: Other connectives

Other connectives $\vee$, $\wedge$, $\leftrightarrow$ are treated as abbreviations of formulas (involving $\{\neg, \to\}$ only) as follows:

$$\begin{array}{rcl} p \vee q & \text{iff} & \neg p \to q \\ p \wedge q & \text{iff} & \neg(p \to \neg q) \\ p \leftrightarrow q & \text{iff} & (p \to q) \wedge (q \to p) \end{array}$$

The following sets of connectives are complete:

- $\{\neg, \wedge, \vee\}$ and $\{\neg, \wedge, \vee, \to, \leftrightarrow\}$, by NDF
- $\{\neg, \to\}$,
- $\{\neg, \wedge\}$ and $\{\neg, \vee\}$.

Prove that $\{\vee\}$, $\{\wedge\}$, $\{\to\}$, $\{\neg, \leftrightarrow\}$ are not complete.

Recall: Another list of binary connectives:

Symbol	Equivalent	Remarks
$\wedge$	$p \wedge q$	且
$\vee$	$p \vee q$	或
$\rightarrow$	$p \rightarrow q$	如果 p 就 q
$\leftarrow$	$p \leftarrow q$	如果 q 就 p
$\leftrightarrow$	$p \leftrightarrow q$	p 当且仅当 q
$\oplus$	$(p \vee q) \wedge \neg(p \wedge q)$	异或，相当于对称差
$\uparrow$	$\neg(p \wedge q)$	与非
$\downarrow$	$\neg(p \vee q)$	或非
$<$	$\neg p \wedge q$	相当于 $q \setminus p$
$>$	$p \wedge \neg q$	相当于 $p \setminus q$

Show that $\{\uparrow\}$ and $\{\downarrow\}$ are complete.

$\models$ vs. $\Rightarrow$

Thinking Exercise 4.19

Using truth table prove the following

- $F \models G$ *if and only if* $\models (F \Rightarrow G)$.
- $F \equiv G$ *if and only if* $\models (F \Leftrightarrow G)$.

Exercise: Downward Saturation

Thinking Exercise 4.20

Let us suppose we only have connectives $\land$, $\lor$, *or* $\neg$ *in our formulas. Consider a set* Σ *of formulas such that*

1. *for each* $p \in \mathrm{Vars}$, $p \notin \Sigma$ *or* $\neg p \notin \Sigma$
2. *if* $\neg\neg F \in \Sigma$ *then* $F \in \Sigma$
3. *if* $(F \land G) \in \Sigma$ *then* $F \in \Sigma$ *and* $G \in \Sigma$
4. *if* $\neg(F \lor G) \in \Sigma$ *then* $\neg F \in \Sigma$ *and* $\neg G \in \Sigma$
5. *if* $(F \lor G) \in \Sigma$ *then* $F \in \Sigma$ *or* $G \in \Sigma$
6. *if* $\neg(F \land G) \in \Sigma$ *then* $\neg F \in \Sigma$ *or* $\neg G \in \Sigma$

Show that Σ *is satisfiable, i.e., there is a model that satisfies every formula in* Σ*.*

Thinking Exercise 4.21

Given algorithm that extends a set Σ *into a set of the formula that satisfy the above. Can we use the algorithm as a satisfiability checker?*

Commentary: Please note that the above does not hold if we drop any of the six conditions. You need to show that all six are needed.

Exercise: Counting Models

Thinking Exercise 4.22

Let propositional variables p, q, are r be relevant to us. There are eight possible models to the variables. Out of the eight, how many satisfy the following formulas?

1. p
2. $p \vee q$
3. $p \vee q \vee r$
4. $p \vee \neg p \vee r$

Exercise: Universal Connective

Let $\overline{\wedge}$ be a binary connective with the following truth table (called "and not" 与非)

$m(F)$	$m(G)$	$m(F\overline{\wedge}G)$
0	0	1
0	1	1
1	0	1
1	1	0

Thinking Exercise 4.23

a. Show $\overline{\wedge}$ can define all other connectives.

b. Are there other universal connectives?

Topic 4.6

Extra slides: Sizes of Models

Size of models

A model must assign value to all the variable, since it is a complete function.

However, we may not want to handle such an object.

In practice, we handle partial models. Often, without explicitly mentioning this.

Partial models

Let $m|_{\mathsf{Vars}(F)} : \mathsf{Vars}(F) \to \mathcal{B}$ and for each $p \in \mathsf{Vars}(F)$, $m|_{\mathsf{Vars}(F)}(p) = m(p)$

Theorem 4.3

If $m|_{\mathsf{Vars}(F)} = m'|_{\mathsf{Vars}(F)}$ then $m \models F$ iff $m' \models F$

Proof sketch.

The procedure to check $m \models F$ only looks at the $\mathsf{Vars}(F)$ part of m. Therefore, any extension of $m|_{\mathsf{Vars}(F)}$ will have same result either $m \models F$ or $m \not\models F$. $\square$

Definition 4.8

We will call elements of $\mathsf{Vars} \to \mathcal{B}$ *as partial models.*

Thinking Exercise 4.24

Write the above proof formally.

End of Lecture 4