Propositional Logic

Interested in statements about the behaviour of computer systems & programs

specify & check

$\Rightarrow$ need a calculus of reasoning

Propositional Logic

* Declarative sentences
(or propositions)
those which can be true or false.

[It is raining

[Every natural number > 2
is a sum of two prime
numbers.

Non-declarative sentences

Happy birthday!

(We'll not deal with them)

Syntax versus Semantics

Syntax is about how we write formulas (with the goal of using them in formal proofs). It is about string manipulations, without regard to their meaning.

Semantics is about the formal meaning of those strings.

Well-Formed Formulas (WFFs) of Propositional Logic

The set of WFFs of prop. logic is defined inductively

Base case: each prop. atom is a WFF.

Ind. step: If A and B are WFFs then so are $\neg A$, $(A \vee B)$, $(A \wedge B)$.

Nothing else is a WFF.

Define $(A \to B)$ "A implies B"

as an abbreviation for $(\neg A \vee B)$

$A \leftrightarrow B$ abbreviates $(A \to B) \wedge (B \to A)$

Syntax of propositional logic

Alphabet

- propositional atoms

  $P_1, P_2, \ldots$ (a countably infinite set)

  they will denote declarative sentences

- connectives

  $\wedge, \vee, \neg$ - "and", "or", "not"

  other connectives will be definable

- parentheses:

  (,)

  Sometimes we also use [,] for readability

Sometimes we use English words for proposition names

e.g. MadeOfWood instead of P_{24}

In many cases, we omit parentheses, assuming

$\neg$ binds most tightly

then $\wedge$, $\vee$

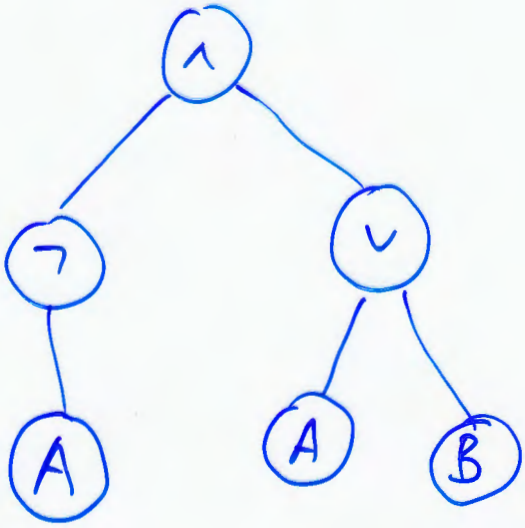
then $\rightarrow$, $\leftrightarrow$

$(A \leftrightarrow B)$

$(A \rightarrow B) \wedge (B \rightarrow A)$

It is convenient to view a formula as a tree T_A

$(\neg A \wedge (A \vee B))$



Semantics of Propositional Logic

A truth assignment for a set S of atoms is a function mapping each element of S to {true, false}

"tau" → $\tau : \overbrace{Atoms}^{S} \rightarrow \{true, false\}$

Def. Let τ be a truth assignment for set S of atoms, and A a formula over the atoms in S. We say that τ **satisfies** A, written $\tau \models A$, if the following holds:

- If A is an atom, then
  $\mathcal{T} \models A$ iff $\mathcal{T}(A) = \text{true}$;
- If A is of the form $\neg B$, then
  $\mathcal{T} \models A$ iff $\mathcal{T} \not\models B$;
- If A is of the form $(B \vee C)$, then
  $\mathcal{T} \models A$ iff $\mathcal{T} \models B$ or $\mathcal{T} \models C$

Thus, each truth assignment (t.a.) for S either satisfies A or does not.

We write: $\mathcal{T} \models A$, $\mathcal{T}(A) = \text{true}$

$\mathcal{T}$ satisfies A, $\mathcal{T}$ makes A true.

Satisfiability and Validity

$A :=$ denotes, is by def.

Example: $(P \vee Q) \rightarrow (\neg P \vee R)$

Consider the following t.a. $\mathcal{I}_1$:

$\mathcal{I}_1(P) = 0$ (=false)

$\mathcal{I}_1(Q) = 1$ $\mathcal{I}_1$ satisfies A

$\mathcal{I}_1(R) = 1$ (=true)

$\mathcal{I}_2(P) = 1$

$\mathcal{I}_2(Q) = 1$ $\mathcal{I}_2$ falsifies A

$\mathcal{I}_2(R) = 0$

Def. We say that formula A is

- Satisfiable, if there is some t.a. for the atoms of A that satisfies A;
- Unsatisfiable, if no truth assign. satisfies it
- Valid (or is a tautology) if every t.a. satisfies it.

EXAMPLE:

$$P \vee Q \rightarrow P$$

satisfiable?

yes, there is a t.a. that satisfies it

valid?

no. there is a t.a that falsifies it

$\mathcal{I}(P) = \text{false}$

$\mathcal{I}(Q) = \text{true}$.

Example: $P \wedge \neg(P \vee Q)$

valid?

no

contradiction (unsatisfiable)?

yes, because every t.a. falsifies it

P	Q	$P \wedge \neg(P \vee Q)$
0	0	0
0	1	0
1	0	0
1	1	0

t.a → (each row)

value of f-la under each t.a.

$\{P \wedge Q, P\}$ is satisfiable

because $\tau(P) = \tau(Q) =$ true

satisfies all f-las in the set

We generalize the notion of satisfiability to sets of f-las.

Let S be a set of atoms, and $\Gamma = \{A_1, A_2, \ldots\}$ a set of f-las over the atoms in S.

We say that a t.a. τ for S satisfies Γ iff τ satisfies every formula in Γ.

Γ is satisfiable if there is a t.a. that satisfies it.

Mod X - (for a formula X or a set of f-las X) is the set of t.a. to the atoms of X that satisfy it (are models of X)

Def. Propositional Formula Satisfiability (SAT)

Given: A propositional formula A

Question: Is A satisfiable?

SAT is hard in general
(NP-complete) - Cook's theorem

- many practical problems can be represented as SAT

  including bounded model checking

- no polytime algorithm known
  (now, maybe ever)

But: - there are polytime special cases

- there are general techniques that often work in practice

SAT competitions

NP is the "new tractable"

So far:

Syntax of prop. logic (WFFs)

Semantics of prop. logic

- notion of a truth assignment
  - satisfying a f-la
  - falsifying a f-la

satisfiable,
valid formulas, = tautology
contradictions.