Lecture notes 2.1

Propositional calculus: formulas, interpretations, logical equivalence

COMP 2411, session 1, 2004

Propositional atoms

The propositional calculus is a formal language that can be used to formalise statements expressed in natural languages.

Whereas the meaning of an English statement is complex, involving many aspects of the real world, the meaning of a statement in the propositional calculus can be either true or false.

The easiest way to formalise an English statement in the propositional calculus is to represent it by a propositional atom, and give it one of the truth values true or false.

Example

For example:

English statement	Atom	Truth value
2 is an even natural number	p	true
Earth is as blue as an orange	q	false
If I am walking then I am not running	p_1	true
If 2 times 3 equals 5 then I love apples	p_2	true
I am sitting or I am lying	p_3	false

In propositional logic, there is no notion of 'meaningless,' or 'possibly true,' or 'sometimes true.'

Propositional formulas

A more complex way of formalising English statements in the propositional calculus is to represent them by propositional formulas that are boolean combinations of propositional atoms, using boolean operators. For example:

English statement	Formula
2 is an even natural number	p
Earth is as blue as an orange	q
If I am walking then I am not running	$q_1 \rightarrow q_2$
If I am walking then I am not running	$q_1 \rightarrow \neg q_2'$
If 2 times 3 equals 5 then I love apples	$q_3 \rightarrow q_4$
I am standing or I am sitting	$r \vee r'$

Meaning of propositional formulas

The meaning of a propositional formula is also true or false, and is determined by the meaning of the constituting atoms, together with the fixed meaning of the boolean operators, that map truth values to either true or false.

For instance, $\rightarrow$ is a boolean operator such that:

- if p is true and q is false then $p \rightarrow q$ is false;
- otherwise $p \rightarrow q$ is true.

Hence:

- if q_3 is false then $q_3 \rightarrow q_4$ is true, irrespective of the truth value of q_4;
- if q_1 is true and q_2 is true then $q_1 \rightarrow q_2$ is true.

Meaning of boolean operators

$\neg$ is a unary boolean operator, whereas $\rightarrow$ and $\vee$ are binary boolean operators.

More generally, given a nonnull $n \in \mathbb{N}$, the meaning of an n-ary boolean operator is given by a function whose type is:

$$\{\mathrm{true}, \mathrm{false}\}^n \rightarrow \{\mathrm{true}, \mathrm{false}\}$$

Hence there are 2^{2^n} n-ary boolean operators.

- true is also represented by T, $\top$ or 1;
- false is also represented by F, $\perp$ or 0.

Usual boolean operators

The usual boolean operators are the following, where the first one in the list is unary, and the others are binary.

- negation: $\neg$
- disjunction, also called inclusive or: $\vee$
- conjunction: $\wedge$
- (material) implication: $\rightarrow$
- equivalence: $\leftrightarrow$
- exclusive or: $\oplus$
- nor: $\downarrow$
- nand: $\uparrow$

Unary boolean operators

p		p	$\neg p$	
T	T	T	F	F
F	T	F	T	F

Binary boolean operators

p	q		$p \vee q$	$q \rightarrow p$	p	$p \rightarrow q$	q	$p \leftrightarrow q$	$p \wedge q$
T	T	T	T	T	T	T	T	T	T
T	F	T	T	T	T	F	F	F	F
F	T	T	T	F	F	T	T	F	F
F	F	T	F	T	F	T	F	T	F

p	q	$p \uparrow q$	$p \oplus q$	$\neg q$		$\neg p$		$p \downarrow q$	
T	T	F	F	F	F	F	F	F	F
T	F	T	T	T	T	F	F	F	F
F	T	T	T	F	F	T	T	F	F
F	F	T	F	T	F	T	F	T	F

A few properties

We will verify the following claims.

- The fact that some unary and binary boolean operators have no name (corresponding to the columns with an empty cell) is not a problem because they can all be defined in terms of those that have a name.
- More generally, for all $n \in \mathbb{N}$, all n-ary boolean operators are definable in terms of either:
  - $\neg$ and $\vee$
  - $\neg$ and $\wedge$
  - $\downarrow$
  - $\uparrow$
- $\downarrow$ and $\uparrow$ are the only binary boolean operators from which all boolean operators can be defined.

Formulas defined formally

Fix an infinite set $\mathbb{P}$ of propositional atoms, *e.g.*:

$$\mathbb{P} = \{p, q, r, p_1, q_1, r_1, p_2, q_2, r_2, \ldots\}$$

Choose a set Op of boolean operators that can generate all boolean operators, *e.g.*:

$$\mathrm{Op} = \{\neg, \vee, \wedge, \rightarrow, \leftrightarrow, \oplus\}$$

The set of propositional formulas (built from $\mathbb{P}$ and Op is the smallest set $\mathcal{F}$ such that:

- all members of $\mathbb{P}$ belong to $\mathcal{F}$;
- for all $\varphi \in \mathcal{F}$, $\neg\varphi$ belongs to $\mathcal{F}$;
- for all $\varphi_1, \varphi_2 \in \mathcal{F}$, $(\varphi_1 \vee \varphi_2)$, $(\varphi_1 \wedge \varphi_2)$, $(\varphi_1 \rightarrow \varphi_2)$, $(\varphi_1 \leftrightarrow \varphi_2)$, and $(\varphi_1 \oplus \varphi_2)$ all belong to $\mathcal{F}$.

Simplified syntax

Redundant parentheses can be removed, taking into account the convention that:

- $\wedge$ had precedence over $\vee$;
- $\vee$ has precedence over $\rightarrow$;
- $\rightarrow$ has precedence over $\leftrightarrow$ and $\oplus$;
- all binary operators associate from right to left.

For instance:

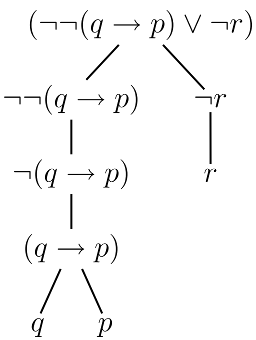
- $(p \rightarrow (q \rightarrow r))$ can be simplified to $p \rightarrow q \rightarrow r$;
- $(p \vee (q \wedge r))$ can be simplified to $p \vee q \wedge r$;
- $((p \rightarrow q) \leftrightarrow (\neg q \rightarrow \neg p))$ can be simplified to $p \rightarrow q \leftrightarrow \neg q \rightarrow \neg p$.

Parse trees

Propositional formulas can be represented by parse trees.

For instance, the following is a parse tree representation of

$$(\neg\neg(q \to p) \lor \neg r)$$



Assignments and interpretations (1)

An assignment is a (total) mapping from $\mathbb{P}$ (set of propositional atoms) into $\{T, F\}$.

An interpretation is a (total) mapping from $\mathcal{F}$ (set of propositional formulas) into $\{T, F\}$ that respects the meaning of the boolean operators. This means that:

- for all formulas φ, $\overline{\nu}(\neg\varphi) = T$ iff $\overline{\nu}(\varphi) = F$;
- for all formulas φ_1, φ_2, $\overline{\nu}(\varphi_1 \land \varphi_2) = T$ iff $\overline{\nu}(\varphi_1) = T$ and $\overline{\nu}(\varphi_2) = T$;
- Idem for all other boolean operators.

Assignments and interpretations (2)

Proposition: An assignment ν can be extended to a unique interpretation $\overline{\nu}$.
Moreover, for all $\varphi \in \mathcal{F}$, $\overline{\nu}(\varphi)$ depends only on $\nu(\alpha)$ where α varies over the set of propositional atoms that occur in φ.

For example, consider an assignment ν such that $\nu(p) = F$ and $\nu(q) = T$. Then:

- $\overline{\nu}(p \to q) = T$
- $\overline{\nu}(\neg q) = F$ and $\overline{\nu}(\neg p) = T$
- $\overline{\nu}(\neg q \to \neg p) = T$
- $\overline{\nu}((p \to q) \leftrightarrow (\neg q \to \neg p)) = T$.

Logical equivalence

Two formulas φ and ψ are logically equivalent, written $\varphi \equiv \psi$, iff they agree on all interpretations.

The following is a small list of pairwise logically equivalent formulas. See Figure 2.6 in textbook for more, including two wrong ones.

- $\varphi \to \psi \equiv \neg\varphi \lor \psi$
- $\neg(\varphi \to \psi) \equiv \varphi \land \neg\psi$
- $\varphi \leftrightarrow \psi \equiv \varphi \to \psi \land \psi \to \varphi$
- $\varphi \leftrightarrow \psi \equiv \varphi \land \psi \lor \neg\varphi \land \neg\psi$
- $\varphi \leftrightarrow \psi \equiv (\varphi \lor \neg\psi) \land (\neg\varphi \lor \psi)$
- $\neg(\varphi \lor \psi) \equiv \neg\varphi \land \neg\psi$
- $\neg(\varphi \land \psi) \equiv \neg\varphi \lor \neg\psi$

Substitution

Given 3 formulas φ, ψ and ξ, we denote by $\varphi\{\psi \leftarrow \xi\}$ the result of substituting all occurrence of ψ in φ by ξ.

For instance, if $\varphi = (p \rightarrow q) \leftrightarrow (\neg p \rightarrow \neg q)$, $\psi = p \rightarrow q$ and $\xi = \neg p \vee q$ then:

$$\varphi\{\psi \leftarrow \xi\} = (\neg p \vee q) \leftrightarrow (\neg p \rightarrow \neg q).$$

Proposition: For all formulas φ, ψ, ξ_1 and ξ_2, if $\xi_1 \equiv \xi_2$ then $\varphi\{\psi \leftarrow \xi_1\} \equiv \varphi\{\psi \leftarrow \xi_2\}$.
In other words, substituting all occurrences of ψ in φ by logically equivalent formulas yields logically equivalent formulas.