4CCS1ELA–Elementary Logic with Applications

3–Important Semantical Notions

3.2–Quine's Method and Satisfiability

Dr. Odinaldo Rodrigues

odinaldo.rodrigues@kcl.ac.uk
Room BH(S) TBC, +44 (0)20 7848 2087
Department of Informatics
King's College London

Outline

1. Quine's Method
2. Satisfiability

QUINE'S METHOD

Quine's Method

For any formula W and propositional variable P:

- W is a tautology if and only if $W(P/\mathbf{0})$ and $W(P/\mathbf{1})$ are tautologies.
- W is a contradiction if and only if $W(P/\mathbf{0})$ and $W(P/\mathbf{1})$ are contradictions.

Quine's method can be described graphically with a binary tree (Hein, Section 6.2).

Constructing Quine's Tree for a Formula W

1. Start with W as the root of the tree.
2. Take the first level in the tree with a propositional symbol, say P, in any of the level's nodes n. If none are left, then finish.
3. Let the left child of n be $n(P/\mathbf{1})$ and let its right child be $n(P/\mathbf{0})$.
4. Repeat from 2.

When no propositional symbols remain:

- W is a tautology if all of the leaves in the tree are true (i.e., **1**)
- W is a contradiction if all leaves in the tree are false (i.e., **0**)
- Otherwise, W is a *contingency* (i.e., sometimes true, sometimes false)

Example

$$\underline{P} \to Q \land \underline{P}$$

$$\mathbf{1} \to Q \land \mathbf{1} \qquad \mathbf{0} \to Q \land \mathbf{0}$$

$$\mathbf{1} \to Q \qquad \mathbf{0} \to \mathbf{0}$$

$$\underline{Q} \qquad \mathbf{1}$$

$$\mathbf{1} \quad \mathbf{0}$$

Conclusion. The formula $P \to Q \land P$ is a contingency, because neither all leaves in the tree are true, nor all of them are false.

SATISFIABILITY

Satisfiability

A formula F is **satisfiable** if there is an interpretation v that makes the formula F true. In this case, we say that v **satisfies** F.

A set $\mathcal{S} = \{A_1, \ldots, A_n\}$ of propositional formulae is **satisfiable** (**consistent**) if there is an interpretation v satisfying *every* formula in $\mathcal{S}$.

	p_1	$\ldots$	p_m	A_1	$\ldots$	A_n
v	e_1	$\ldots$	e_m	1	$\ldots$	1

$p_1, \ldots, p_m$ are all the propositional symbols appearing in $\mathcal{S}$.

The set of formulae $\{A_1, \ldots, A_n\}$ is satisfiable if, and only if, the conjunction $A_1 \wedge \ldots \wedge A_n$ is satisfiable

Example

Let $A_1 = P \to Q$, $A_2 = Q \to R$ and $A_3 = R \to P$ and $\mathcal{S}$ be the set of formulae $\{A_1, A_2, A_3\}$. The combined truth-table for $\mathcal{S}$ is:

	p_1	p_2	p_3	A_1	A_2	A_3
	P	Q	R	$P \to Q$	$Q \to R$	$R \to P$
v_0	0	0	0	1	1	1
v_1	0	0	1	1	1	0
v_2	0	1	0	1	0	1
v_3	0	1	1	1	1	0
v_4	1	0	0	0	1	1
v_5	1	0	1	0	1	1
v_6	1	1	0	1	0	1
v_7	1	1	1	1	1	1

Thus, $\mathcal{S}$ is satisfiable, since v_0 and v_7 satisfy every formula in $\mathcal{S}$ (we can also say that $\mathcal{S}$ is *consistent*).

Models

A *model* is an interpretation that makes a formula (or set of formulae) true.

We denote the fact that v is a model of A by $v \Vdash A$.

The set of all models of a formula A is denoted by $\mathtt{mod}(A)$. We use the same symbol for a set of formulae $\mathcal{S}$, as before.

Thus, in the Example of Slide 3.2.8 we have that $\mathtt{mod}(\mathcal{S}) = \{v_0, v_7\}$.

Example

Let $\mathcal{S}$ be the set of formulae $\{P \leftrightarrow \neg Q, Q \leftrightarrow R, R \leftrightarrow P\}$. The truth-table for its formulae is:

	P	Q	R	$P \leftrightarrow \neg Q$	$Q \leftrightarrow R$	$R \leftrightarrow P$
v_0	0	0	0	0	1	1
v_1	0	0	1	0	0	0
v_2	0	1	0	1	0	1
v_3	0	1	1	1	1	0
v_4	1	0	0	1	1	0
v_5	1	0	1	1	0	1
v_6	1	1	0	0	0	0
v_7	1	1	1	0	1	1

$\mathcal{S}$ is **not** satisfiable (i.e., $\mathcal{S}$ is inconsistent) and thus, $\mathtt{mod}(\mathcal{S}) = \varnothing$.

To know more...

- Quine's method is shown in Section 6.2 of Hein's "Discrete Structures, Logic, and Computability", 4th edition.
- Satisfiability can be found in Sections 1.2 and 1.3 of Gabbay and Rodrigues' "Elementary Logic with Applications, 1st edition.