Logic in Computer Science (COMP118)
Tutorial Problems 1

1. Let
   • \( p_1 \) denote the proposition: Tom’s house is red;
   • \( p_2 \) denote the proposition: Jim’s house is red;
   • \( p_3 \) denote the proposition: Mary’s house is red.

   Translate into propositional logic:
   • Tom’s house is red or Jim’s house is red;
   • If Jim’s house is red, then Tom’s house is red and Mary’s house is red;
   • Tom’s house is not red;
   • Jim’s house is red if, and only if, Tom’s house is not red;
   • Neither Jim’s nor Mary’s house is red.

   Let \( P_1, P_2, \ldots, P_5 \) be the translations into propositional logic of the five sentences above. For each \( P_i, 1 \leq i \leq 5 \):
   • give an interpretation \( I \) such that \( I(P_i) = 1 \) and
   • give an interpretation \( J \) such that \( J(P_i) = 0 \).

2. Give a definition of satisfiable formulas.

3. Which of the following formulas are satisfiable? Check this
   (a) using truth tables and
   (b) using the tableau algorithm.
   • \((\neg \neg p \land \neg(p \lor q))\);
   • \((\neg \neg p \lor q) \land \neg r\);
   • \(((p \rightarrow q) \land \neg(q \rightarrow p))\);
   • \(((p \rightarrow q) \land \neg(p \rightarrow q))\);
• \(((\neg (p \land q) \land p) \land q)\).

(Hint: We do not have tableau rules for \(\rightarrow\). Thus, \(\rightarrow\) has to be replaced by its definition before the tableau rules can be applied.)

4. Call a propositional formula \(P\) positive if it does not contain the symbol \(\neg\). Give an inductive definition of positive formulas. Show that every positive formula is satisfiable.

Solution for 1.

• \(P_1 = (p_1 \lor p_2)\). Let \(I(p_1) = 1\) and \(I(p_2) = 1\). Then \(I(P_1) = 1\). Let \(J(p_1) = 0\) and \(J(p_2) = 0\). Then \(J(P_1) = 0\).

• \(P_2 = (p_2 \rightarrow (p_1 \land p_3))\). Let \(I(p_1) = 1\) and \(I(p_2) = 1\) and \(I(p_3) = 1\). Then \(I(P_2) = 1\). Let \(J(p_1) = 1\) and \(J(p_2) = 0\) and \(J(p_3) = 0\). Then \(J(P_2) = 0\).

• \(P_3 = \neg p_1\). Let \(I(p_1) = 0\). Then \(I(P_3) = 1\). Let \(J(p_1) = 1\). Then \(J(P_3) = 0\).

• \(P_4 = (p_2 \leftrightarrow \neg p_1)\). Let \(I(p_2) = 1\) and \(I(p_1) = 0\). Then \(I(P_4) = 1\). Let \(J(p_1) = 1\) and \(J(p_2) = 1\). Then \(J(P_4) = 0\).

• \(P_5 = (\neg p_2 \land \neg p_3)\). Let \(I(p_2) = 0\) and \(I(p_3) = 0\). Then \(I(P_5) = 1\). Let \(J(p_2) = 1\) and \(J(p_3) = 1\). Then \(J(P_5) = 0\).

Solution for 3.

• For \(P_1 = (\neg \neg p \land \neg (p \lor q))\) we have the following truth table:

<table>
<thead>
<tr>
<th>(p)</th>
<th>(q)</th>
<th>(\neg p)</th>
<th>(\neg (p \lor q))</th>
<th>(\neg (p \lor q))</th>
<th>(P_1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

It follows that \(I(P_1) = 0\) for all interpretations \(I\). Hence \(P_1\) is not satisfiable.

Using the tableau algorithm, we set \(S_0 = \{P_1\}\).
An application of $\implies_\land$ to $S_0$ gives

$$S_1 = S_0 \cup \{\neg\neg p, \neg (p \lor q)\}$$

An application of $\implies_\lor$ to $S_1$ gives

$$S_2 = S_1 \cup \{\neg p, \neg q\}$$

$S_2$ contains the clash $\{\neg\neg p, \neg p\}$. Since we obtained a tableau path with a clash and we only applied non-branching rules, it follows that $P_1$ is not satisfiable.

Note that we applied only non-branching rules. Nevertheless, there are of course more tableau paths starting with $S_0 = \{P_1\}$. Namely,

- An application of $\implies_\land$ to $S_0$ gives

$$S_1 = S_0 \cup \{\neg\neg p, \neg (p \lor q)\}$$

- An application of $\implies_\lor$ to $S_1$ gives

$$S_2 = S_1 \cup \{p\}$$

- An application of $\implies_\lor$ to $S_2$ gives

$$S_3 = S_2 \cup \{\neg p, \neg q\}$$

$S_3$ contains a clash (actually two). Thus, again we obtain that $P_1$ is not satisfiable. This type of non-determinism of the algorithm (caused by different orders in which the rules are applied) is called “don’t care non-determinism”: it does not depend on which rule we apply first whether we obtain a clash or not.

- For $P_2 = \neg (\neg(p \lor q) \land \neg r)$ we have the following truth table:
It follows that $I(P_2) = 1$ for at least one interpretation $I$. Thus, $P_2$ is satisfiable.

Using the tableau algorithm, we set $S_0 = \{P_2\}$.

- An application of $\implies \land$ to $S_0$ gives
  
  $$S_1 = S_0 \cup \{\neg(p \lor q), \neg r\}$$

- An application of $\implies \neg \lor$ to $S_1$ gives
  
  $$S_2 = S_1 \cup \{\neg \neg p, \neg q\}$$

- An application of $\implies \neg \neg$ to $S_2$ gives
  
  $$S_3 = S_2 \cup \{p\}$$

$s_3$ is complete and contains no clash. Thus, $P_2$ is satisfiable.

- First we unfold the abbreviation for $\to$ and obtain $P_3 = (\neg p \lor q) \land \neg (\neg q \lor p)$. We obtain the following truth table:

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$r$</th>
<th>$\neg p$</th>
<th>$\neg r$</th>
<th>$\neg(p \lor q)$</th>
<th>$\neg(q \lor p)$</th>
<th>$P_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
It follows that $I(P_3) = 1$ for at least one interpretation $I$. Hence $P_3$ is satisfiable.

Using the tableau algorithm, we set $S_0 = \{P_3\}$.

- An application of $\implies \land$ to $S_0$ gives
  \[ S_1 = S_0 \cup \{\neg p \lor q, \neg (\neg q \lor p)\} \]

- An application of $\implies \neg \lor$ to $S_1$ gives
  \[ S_2 = S_1 \cup \{\neg q, \neg p\} \]

- An application of $\implies \neg \neg$ to $S_2$ gives
  \[ S_3 = S_2 \cup \{q\} \]

No rule is applicable to $S_3 = \{P_3, \neg p \lor q, \neg (\neg q \lor p), \neg q, \neg p, q\}$. $S_3$ does not contain a clash. Thus, $P_3$ is satisfiable.

• First we unfold the abbreviation for $\to$ and obtain $P_4 = ((\neg p \lor q) \land \neg (\neg p \lor q))$. We obtain the following truth table:

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>\neg p</th>
<th>\neg (\neg p \lor q)</th>
<th>\neg (\neg p \lor q)</th>
<th>P_4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

It follows that $I(P_4) = 0$ for all interpretations $I$. Hence $P_4$ is not satisfiable.

Using the tableau algorithm, we set $S_0 = \{P_4\}$.

- An application of $\implies \land$ to $S_0$ gives
  \[ S_1 = S_0 \cup \{\neg p \lor q, \neg (\neg p \lor q)\} \]

$S_1$ contains a clash. Thus, $P_4$ is not satisfiable.

• We set $P_5 = ((\neg (p \land q) \land p) \land q)$ and obtain the truth table:
It follows that $I(P_5) = 0$ for all interpretations $I$. Hence $P_5$ is not satisfiable.

Using the tableau algorithm, we set $S_0 = \{P_5\}$.

- An application of $\implies \wedge$ to $S_0$ gives

  $$S_1 = S_0 \cup \{(- (p \wedge q) \wedge p), q\}$$

- An application of $\implies \wedge$ to $S_1$ gives

  $$S_2 = S_1 \cup \{(- (p \wedge q), p\}$$

- An application of $\implies - \wedge$ to $S_2$ gives

  $$S_3 = S_2 \cup \{-p\}$$

  which contains the clash $\{p, -p\}$.

- The other application of $\implies - \wedge$ to $S_2$ gives

  $$S_4 = S_3 \cup \{-q\}$$

  which contains the clash $\{q, -q\}$.

Thus, all complete tableaus starting with $\{P_5\}$ contain a clash. Hence $P_5$ is not satisfiable.

**Solution for 4.**

A definition of positive formulas is as follows:

- all atomic formulas are positive formulas;
- if $P$ and $Q$ are positive formulas, then $(P \wedge Q)$ is a positive formula;
• if \( P \) and \( Q \) are positive formulas, then \( (P \lor Q) \) is a positive formula;

• Nothing else is a positive formula.

Every positive formula is satisfiable: Let \( P \) be a positive formula. We set \( I(p) = 1 \) for every atomic formula \( p \) and show by induction over the construction of positive formulas \( Q \) that \( I(Q) = 1 \). For atomic formulas this follows from the definition.

• Assume \( Q = (Q_1 \land Q_2) \). By induction hypothesis, \( I(Q_1) = 1 \) and \( I(Q_2) = 1 \). Thus \( I(Q) = 1 \).

• Assume \( Q = (Q_1 \lor Q_2) \). By induction hypothesis, \( I(Q_1) = 1 \) and \( I(Q_2) = 1 \). Thus \( I(Q) = 1 \).