Propositional Equivalences

Section 1.3
Tautologies, Contradictions, and Contingencies

- A **tautology** is a proposition which is always true.
  - Example: $p \lor \lnot p$

- A **contradiction** is a proposition which is always false.
  - Example: $p \land \lnot p$

- A **contingency** is a proposition which is neither a tautology nor a contradiction, such as $p$

<table>
<thead>
<tr>
<th>$P$</th>
<th>$\lnot p$</th>
<th>$p \lor \lnot p$</th>
<th>$p \land \lnot p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
</tbody>
</table>
Logically Equivalent

Definition:

- Two compound propositions $p$ and $q$ are logically equivalent if $p \leftrightarrow q$ is a tautology.
- Alternatively, two compound propositions $p$ and $q$ are equivalent if and only if the columns in a truth table giving their truth values agree.
- We write this as $p \leftrightarrow q$ or as $p \equiv q$ where $p$ and $q$ are compound propositions.
Logically Equivalent

Example:
- This truth table shows $\neg p \lor q$ is equivalent to $p \rightarrow q$.

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$\neg p$</th>
<th>$\neg p \lor q$</th>
<th>$p \rightarrow q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>
De Morgan’s Laws

\[ \neg(p \land q) \equiv \neg p \lor \neg q \]

\[ \neg(p \lor q) \equiv \neg p \land \neg q \]

This truth table shows that De Morgan’s Second Law holds.

<table>
<thead>
<tr>
<th>(p)</th>
<th>(q)</th>
<th>(\neg p)</th>
<th>(\neg q)</th>
<th>(p \lor q)</th>
<th>(\neg(p \lor q))</th>
<th>(\neg p \land \neg q)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>
Key Logical Equivalences

- **Identity Laws:** \( p \land T \equiv p \), \( p \lor F \equiv p \)

- **Domination Laws:** \( p \lor T \equiv T \), \( p \land F \equiv F \)

- **Idempotent laws:** \( p \lor p \equiv p \), \( p \land p \equiv p \)

- **Double Negation Law:** \( \neg(\neg p) \equiv p \)

- **Negation Laws:** \( p \lor \neg p \equiv T \), \( p \land \neg p \equiv F \)
Key Logical Equivalences (cont)

- **Commutative Laws:** \( p \lor q \equiv q \lor p \), \( p \land q \equiv q \land p \)

- **Associative Laws:**
  \[
  (p \land q) \land r \equiv p \land (q \land r) \\
  (p \lor q) \lor r \equiv p \lor (q \lor r)
  \]

- **Distributive Laws:**
  \[
  (p \lor (q \land r)) \equiv (p \lor q) \land (p \lor r) \\
  (p \land (q \lor r)) \equiv (p \land q) \lor (p \land r)
  \]

- **Absorption Laws:**
  \[
  p \lor (p \land q) \equiv p \\
  p \land (p \lor q) \equiv p
  \]
More Logical Equivalences

**TABLE 7** Logical Equivalences Involving Conditional Statements.

<table>
<thead>
<tr>
<th>Equation</th>
<th>Equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p \rightarrow q )</td>
<td>(~p \lor q )</td>
</tr>
<tr>
<td>( p \rightarrow q )</td>
<td>(~q \rightarrow ~p )</td>
</tr>
<tr>
<td>( p \lor q )</td>
<td>(~p \rightarrow q )</td>
</tr>
<tr>
<td>( p \land q )</td>
<td>(~(p \rightarrow ~q) )</td>
</tr>
<tr>
<td>(~(p \rightarrow q) )</td>
<td>(p \land ~q )</td>
</tr>
<tr>
<td>((p \rightarrow q) \land (p \rightarrow r) )</td>
<td>(p \rightarrow (q \land r) )</td>
</tr>
<tr>
<td>((p \rightarrow r) \land (q \rightarrow r) )</td>
<td>((p \lor q) \rightarrow r )</td>
</tr>
<tr>
<td>((p \rightarrow q) \lor (p \rightarrow r) )</td>
<td>(p \rightarrow (q \lor r) )</td>
</tr>
<tr>
<td>((p \rightarrow r) \lor (q \rightarrow r) )</td>
<td>((p \land q) \rightarrow r )</td>
</tr>
</tbody>
</table>

**TABLE 8** Logical Equivalences Involving Biconditional Statements.

<table>
<thead>
<tr>
<th>Equation</th>
<th>Equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p \leftrightarrow q )</td>
<td>((p \rightarrow q) \land (q \rightarrow p) )</td>
</tr>
<tr>
<td>( p \leftrightarrow q )</td>
<td>(~p \leftrightarrow ~q )</td>
</tr>
<tr>
<td>( p \leftrightarrow q )</td>
<td>((p \land q) \lor (~p \land ~q) )</td>
</tr>
<tr>
<td>(~(p \leftrightarrow q) )</td>
<td>(p \leftrightarrow ~q )</td>
</tr>
</tbody>
</table>
Constructing New Logical Equivalences

- We can show that two expressions are logically equivalent by developing a series of logically equivalent statements.
- To prove that $A \equiv B$ we produce a series of equivalences beginning with $A$ and ending with $B$.

\[
A \equiv A_1 \\
\vdots \\
A_n \equiv B
\]
Equivalence Proofs

Example: Show that $\neg(p \lor (\neg p \land q))$ is logically equivalent to $\neg p \land \neg q$.

Solution:

\[
\begin{align*}
\neg(p \lor (\neg p \land q)) & \equiv \neg p \land \neg (\neg p \land q) & \text{by the second De Morgan law} \\
& \equiv \neg p \land [\neg (\neg p) \lor \neg q] & \text{by the first De Morgan law} \\
& \equiv \neg p \land (p \lor \neg q) & \text{by the double negation law} \\
& \equiv (\neg p \land p) \lor (\neg p \land \neg q) & \text{by the second distributive law} \\
& \equiv F \lor (\neg p \land \neg q) & \text{because } \neg p \land p \equiv F \\
& \equiv (\neg p \land \neg q) \lor F & \text{by the commutative law for disjunction} \\
& \equiv (\neg p \land \neg q) & \text{by the identity law for } F
\end{align*}
\]
Equivalence Proofs

Example: Show that \((p \land q) \rightarrow (p \lor q)\)

is a tautology.
Propositional Satisfiability

- A compound proposition is *satisfiable* if there is an assignment of truth values to its variables that makes it true. When no such assignments exist, the compound proposition is *unsatisfiable*.
- A compound proposition is unsatisfiable if and only if its negation is a tautology.
Questions on Propositional Satisfiability

**Example:** Determine the satisfiability of the following compound propositions:

\[(p \lor \neg q) \land (q \lor \neg r) \land (r \lor \neg p)\]

**Solution:** Satisfiable. Assign \(T\) to \(p, q,\) and \(r\).

\[(p \lor q \lor r) \land (\neg p \lor \neg q \lor \neg r)\]

**Solution:** Satisfiable. Assign \(T\) to \(p\) and \(F\) to \(q\).

\[(p \lor \neg q) \land (q \lor \neg r) \land (r \lor \neg p) \land (p \lor q \lor r) \land (\neg p \lor \neg q \lor \neg r)\]

**Solution:** Not satisfiable. Check each possible assignment of truth values to the propositional variables and none will make the proposition true.
Solving Satisfiability Problems

- Solving large satisfiability problems can lead to very complex computations...
- A truth table can always be used to determine the satisfiability of a compound proposition. But this is too complex even for modern computers for large problems.
- There has been much work on developing efficient methods for solving satisfiability problems as many practical problems can be translated into satisfiability problems.
- The area of theory of computation studies this in much detail.