Logik für Informatiker
Logic for computer scientists

Boolean Connectives and Formal Proofs

Till Mossakowski

WiSe 2009/10
Formal proofs in Fitch

\[ \begin{array}{c}
P \\
Q \\
R \\
S_1 \\
\vdots \\
\vdots \\
S_n \\
S \\
\end{array}\]

\[ \begin{array}{c}
\text{Justification 1} \\
\text{Justification n} \\
\text{Justification n+1} \\
\end{array}\]
Identity Introduction ($= \text{Intro}$):

$\triangleright \quad n = n$
Identity Elimination (= Elim):

\[
\begin{align*}
P(n) \\
\vdots \\
n = m \\
\vdots \\
\vdash P(m)
\end{align*}
\]
Identity Elimination (= Elim):

\[ P(n) \]

\[ \ldots \]

\[ n = m \]

\[ \ldots \]

\[ P(m) \]

When we apply this rule, it does not matter which of \( P(n) \) and \( n = m \) occurs first in the proof, as long as they both appear before the inferred step. In justifying the step, we cite the name of the rule, followed by the steps in which \( P(n) \) and \( n = m \) occur, in that order.

We could also introduce rules justified by the meanings of other predicates besides \( = \) into the system \( F \). For example, we could introduce a formal rule of the following sort:

Bidirectionality of Between:

\[ \text{Between}(a, b, c) \]

\[ \ldots \]

\[ \Delta \]

\[ \text{Between}(a, c, b) \]

We don’t do this because there are just too many such rules. We could state them for a few predicates, but certainly not all of the predicates you will encounter in first-order languages.

There is one rule that is not technically necessary, but which will make Reiteration some proofs look more natural. This rule is called Reiteration, and simply allows you to repeat an earlier step, if you so desire.

Reiteration (Reit):

\[ P \]

\[ \ldots \]

\[ \Delta \]

\[ P \]

To use the Reiteration rule, just repeat the sentence in question and, on the right, write "Reit: x," where \( x \) is the number of the earlier occurrence of the sentence.
Example proof in fitch
Properties of predicates in Tarski’s world

\[ \begin{align*} 
& \text{Larger}(a, b) \\
& \text{Larger}(b, c) \\
& \quad \underline{\text{Larger}(a, c)} \\
& \text{RightOf}(b, c) \\
& \quad \underline{\text{LeftOf}(c, b)} \\
\end{align*} \]

Such arguments can be proved in Fitch using the special rule \textit{Ana Con}.
This rule is only valid for reasoning about Tarski’s world!
Properties of predicates in Tarski’s world

Larger(a, b)
Larger(b, c)
\hline
Larger(a, c)

RightOf(b, c)
\hline
LeftOf(c, b)

Such arguments can be proved in Fitch using the special rule **Ana Con**.

This rule is only valid for reasoning about Tarski’s world!
Properties of predicates in Tarski’s world

Larger(a, b)
Larger(b, c)

\[ \rightarrow \]
Larger(a, c)

RightOf(b, c)

\[ \rightarrow \]
LeftOf(c, b)

Such arguments can be proved in Fitch using the special rule **Ana Con**.

This rule is only valid for reasoning about Tarski’s world!
Al Gore is a politician
Hardly any politicians are honest
Al Gore is dishonest

Imagine a situation where there are 10,000 politicians, and that Al Gore is the only honest one of the lot. In such circumstances both premises would be true but the conclusion would be false. This demonstrates that the argument is invalid.
Showing invalidity using counterexamples

- Al Gore is a politician
- Hardly any politicians are honest
- Al Gore is dishonest

Imagine a situation where there are 10,000 politicians, and that Al Gore is the only honest one of the lot. In such circumstances both premises would be true but the conclusion would be false.

This demonstrates that the argument is *invalid*. 
Showing invalidity using counterexamples

- Al Gore is a politician
- Hardly any politicians are honest
- Al Gore is dishonest

Imagine a situation where there are 10,000 politicians, and that Al Gore is the only honest one of the lot. In such circumstances both premises would be true but the conclusion would be false. This demonstrates that the argument is invalid.
Are the following arguments valid?

\[
\begin{align*}
\text{Small}(a) & \\
\text{Larger}(b, a) & \\
\Rightarrow & \\
\text{Large}(b) & \\
\text{Small}(a) & \\
\text{Larger}(a, b) & \\
\Rightarrow & \\
\text{Large}(b) & \\
\end{align*}
\]
Negation — Truth table

<table>
<thead>
<tr>
<th>P</th>
<th>¬P</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRUE</td>
<td>FALSE</td>
</tr>
<tr>
<td>FALSE</td>
<td>TRUE</td>
</tr>
</tbody>
</table>
The Henkin-Hintikka game

"Checkmate!"

© Original Artist
Reproduction rights obtainable from
www.CartoonStock.com
The Henkin-Hintikka game

Is a sentence true in a given world?

- Players: you and the computer (Tarski’s world)
- You claim that a sentence is true (or false), Tarski’s world will claim the opposite
- In each round, the sentence is reduced to a simpler one
- When an atomic sentence is reached, its truth can be directly inspected in the given world

You have a winning strategy exactly in those cases where your claim is correct.
Negation — Game rule

<table>
<thead>
<tr>
<th>Form</th>
<th>Your commitment</th>
<th>Player to move</th>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\neg P$</td>
<td>either</td>
<td>$\to$</td>
<td>Replace $\neg P$ by $P$ and switch commitment</td>
</tr>
</tbody>
</table>
### Conjunction — Truth table

<table>
<thead>
<tr>
<th>P</th>
<th>Q</th>
<th>$P \land Q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRUE</td>
<td>TRUE</td>
<td>TRUE</td>
</tr>
<tr>
<td>TRUE</td>
<td>FALSE</td>
<td>FALSE</td>
</tr>
<tr>
<td>FALSE</td>
<td>TRUE</td>
<td>FALSE</td>
</tr>
<tr>
<td>FALSE</td>
<td>FALSE</td>
<td>FALSE</td>
</tr>
</tbody>
</table>
Conjunction — Game rule

<table>
<thead>
<tr>
<th>Form</th>
<th>Your commitment</th>
<th>Player to move</th>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P \land Q$</td>
<td>TRUE</td>
<td>Tarski’s World</td>
<td>Choose one of $P$, $Q$ that is false.</td>
</tr>
<tr>
<td></td>
<td>FALSE</td>
<td>you</td>
<td></td>
</tr>
</tbody>
</table>
Disjunction — Truth table

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>Q</td>
<td>P \lor Q</td>
</tr>
<tr>
<td>TRUE</td>
<td>TRUE</td>
<td>TRUE</td>
</tr>
<tr>
<td>TRUE</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>FALSE</td>
<td>TRUE</td>
<td>TRUE</td>
</tr>
<tr>
<td>FALSE</td>
<td>FALSE</td>
<td>FALSE</td>
</tr>
</tbody>
</table>
## Disjunction — Game rule

<table>
<thead>
<tr>
<th>Form</th>
<th>Your commitment</th>
<th>Player to move</th>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P \lor Q$</td>
<td>TRUE</td>
<td>you</td>
<td>Choose one of $P$, $Q$ that is true.</td>
</tr>
<tr>
<td></td>
<td>FALSE</td>
<td>Tarski’s World</td>
<td></td>
</tr>
</tbody>
</table>
Sometimes, natural language double negation means logical single negation.

The English expression *and* sometimes suggests a temporal ordering; the FOL expression $\land$ never does.

The English expressions *but*, *however*, *yet*, *nonetheless*, and *moreover* are all stylistic variants of *and*.

Natural language disjunction can mean *inclusive-or* ($\lor$) or *exclusive-or*: $A \text{ xor } B \iff (A \lor B) \land (\neg A \lor \neg B)$.
A sentence is

- *logically necessary*, or *logically valid*, if it is true in all circumstances (worlds),
- *logically possible*, if it is true in some circumstances (worlds),
- *logically impossible*, if it is true in no circumstances (worlds).
A sentence is

- *logically necessary*, or *logically valid*, if it is true in all circumstances (worlds),
- *logically possible*, if it is true in some circumstances (worlds),
- *logically impossible*, if it is true in no circumstances (worlds).
A sentence is

- *logically necessary*, or *logically valid*, if it is true in all circumstances (worlds),
- *logically possible*, if it is true in some circumstances (worlds),
- *logically impossible*, if it is true in no circumstances (worlds).
Logically possible

Logically and physically possible

Logically impossible

Logically necessary

\[ P \land \neg P \quad a \neq a \]

\[ P \lor \neg P \quad a = a \]
Formal Proofs in Fitch
The Boolean Connectives

Logically possible

Logically and physically possible

Logically impossible

Logically necessary

$P \land \neg P \quad a \neq a$

$P \lor \neg P \quad a = a$
Logically possible

Logically and physically possible

Logically impossible

$P \land \neg P$

$a \neq a$

Logically necessary

$P \lor \neg P$

$a = a$
Formal Proofs in Fitch
The Boolean Connectives

Logically possible

Logically and physically possible

Logically impossible

Logically necessary
A sentence is

- *logically necessary*, or *logically valid*, if it is true in all circumstances (worlds),
- *TW-necessary*, if it is true in all worlds of Tarski’s world,
- a *tautology*, if it is true in all valuations of the atomic sentences with \{TRUE, FALSE\}.
A sentence is

- **logically necessary**, or **logically valid**, if it is true in all circumstances (worlds),
- **TW-necessary**, if it is true in all worlds of Tarski’s world,
- a **tautology**, if it is true in all valuations of the atomic sentences with \{TRUE, FALSE\}. 
A sentence is

- **logically necessary**, or **logically valid**, if it is true in all circumstances (worlds),
- **TW-necessary**, if it is true in all worlds of Tarski’s world,
- a **tautology**, if it is true in all valuations of the atomic sentences with \{TRUE, FALSE\}.
A sentence is a tautology if and only if it evaluates to TRUE in all rows of its complete truth table.

Truth tables can be constructed with the program Boole.
Tautological equivalence and consequence

- Two sentences $P$ and $Q$ are \emph{tautologically equivalent}, if they evaluate to the same truth value in all valuations (rows of the truth table).
- $Q$ is a \emph{tautological consequence} of $P_1, \ldots, P_n$ if and only if every row that assigns TRUE to each of $P_1, \ldots, P_n$ also assigns TRUE to $Q$.
- If $Q$ is a tautological consequence of $P_1, \ldots, P_n$, then $Q$ is also a \emph{logical consequence} of $P_1, \ldots, P_n$.
- Some logical consequences are not tautological ones.
Tautological equivalence and consequence

- Two sentences $P$ and $Q$ are *tautologically equivalent*, if they evaluate to the same truth value in all valuations (rows of the truth table).
- $Q$ is a *tautological consequence* of $P_1, \ldots, P_n$ if and only if every row that assigns TRUE to each of $P_1, \ldots, P_n$ also assigns TRUE to $Q$.
- If $Q$ is a tautological consequence of $P_1, \ldots, P_n$, then $Q$ is also a *logical consequence* of $P_1, \ldots, P_n$.
- Some logical consequences are not tautological ones.
Two sentences $P$ and $Q$ are *tautologically equivalent*, if they evaluate to the same truth value in all valuations (rows of the truth table).

$Q$ is a *tautological consequence* of $P_1, \ldots, P_n$ if and only if every row that assigns TRUE to each of $P_1, \ldots, P_n$ also assigns TRUE to $Q$.

If $Q$ is a tautological consequence of $P_1, \ldots, P_n$, then $Q$ is also a *logical consequence* of $P_1, \ldots, P_n$.

Some logical consequences are not tautological ones.
Two sentences $P$ and $Q$ are *tautologically equivalent*, if they evaluate to the same truth value in all valuations (rows of the truth table).

$Q$ is a *tautological consequence* of $P_1, \ldots, P_n$ if and only if every row that assigns TRUE to each of $P_1, \ldots P_n$ also assigns TRUE to $Q$.

If $Q$ is a tautological consequence of $P_1, \ldots P_n$, then $Q$ is also a *logical consequence* of $P_1, \ldots, P_n$.

Some logical consequences are not tautological ones.
The **Con** rules in Fitch

- **Taut Con** proves all tautological consequences.
- **FO Con** proves all first-order consequences (like $a = c$ follows from $a = b \land b = c$).
- **Ana Con** proves (almost) all Tarski’s world consequences.
The **Con** rules in Fitch

- **Taut Con** proves all tautological consequences.
- **FO Con** proves all first-order consequences (like $a = c$ follows from $a = b \land b = c$).
- **Ana Con** proves (almost) all Tarski’s world consequences.
The **Con** rules in Fitch

- **Taut Con** proves all tautological consequences.
- **FO Con** proves all first-order consequences
  (like $a = c$ follows from $a = b \land b = c$).
- **Ana Con** proves (almost) all Tarski’s world consequences.
de Morgan’s laws and double negation

$$\neg(P \land Q) \iff (\neg P \lor \neg Q)$$

$$\neg(P \lor Q) \iff (\neg P \land \neg Q)$$

$$\neg\neg P \iff P$$

Note: $\neg$ binds stronger than $\land$ and $\lor$. Brackets are needed to override this.
Negation normal form

- Substitution of equivalents: If $P$ and $Q$ are logically equivalent: $P \Leftrightarrow Q$ then the results of substituting one for the other in the context of a larger sentence are also logically equivalent: $S(P) \Leftrightarrow S(Q)$

- A sentence is in negation normal form (NNF) if all occurrences of $\neg$ apply directly to atomic sentences.

- Any sentence built from atomic sentences using just $\land$, $\lor$, and $\neg$ can be put into negation normal form by repeated application of the de Morgan laws and double negation.
**Negation normal form**

- *Substitution of equivalents*: If $P$ and $Q$ are logically equivalent: $P \Leftrightarrow Q$ then the results of substituting one for the other in the context of a larger sentence are also logically equivalent: $S(P) \Leftrightarrow S(Q)$.

- A sentence is in *negation normal form* (NNF) if all occurrences of $\neg$ apply directly to atomic sentences.

- Any sentence built from atomic sentences using just $\land$, $\lor$, and $\neg$ can be put into *negation normal form* by repeated application of the de Morgan laws and double negation.
Negation normal form

- **Substitution of equivalents**: If $P$ and $Q$ are logically equivalent: $P \iff Q$ then the results of substituting one for the other in the context of a larger sentence are also logically equivalent: $S(P) \iff S(Q)$

- A sentence is in *negation normal form* (NNF) if all occurrences of $\neg$ apply directly to atomic sentences.

- Any sentence built from atomic sentences using just $\land$, $\lor$, and $\neg$ can be put into *negation normal form* by repeated application of the de Morgan laws and double negation.
Distributive laws

For any sentences $P$, $Q$, and $R$:

- **Distribution of $\land$ over $\lor$:**

  \[ P \land (Q \lor R) \iff (P \land Q) \lor (P \land R). \]

- **Distribution of $\lor$ over $\land$:**

  \[ P \lor (Q \land R) \iff (P \lor Q) \land (P \lor R). \]
Distributive laws

For any sentences $P$, $Q$, and $R$:

- **Distribution of $\land$ over $\lor$**:

  $P \land (Q \lor R) \iff (P \land Q) \lor (P \land R)$.

- **Distribution of $\lor$ over $\land$**:

  $P \lor (Q \land R) \iff (P \lor Q) \land (P \lor R)$.
A sentence is in *conjunctive normal form* (CNF) if it is a conjunction of one or more disjunctions of one or more literals.

Distribution of $\lor$ over $\land$ allows you to *transform* any sentence in negation normal form into conjunctive normal form.
A sentence is in *conjunctive normal form* (CNF) if it is a conjunction of one or more disjunctions of one or more literals.

Distribution of $\lor$ over $\land$ allows you to *transform* any sentence in negation normal form into conjunctive normal form.
A sentence is in *disjunctive normal form* (DNF) if it is a disjunction of one or more conjunctions of one or more literals.

Distribution of $\land$ over $\lor$ allows you to *transform* any sentence in negation normal form into disjunctive normal form.

Some sentences are in both CNF and DNF.
A sentence is in *disjunctive normal form* (DNF) if it is a disjunction of one or more conjunctions of one or more literals.

Distribution of $\land$ over $\lor$ allows you to *transform* any sentence in negation normal form into disjunctive normal form.

Some sentences are in both CNF and DNF.
A sentence is in disjunctive normal form (DNF) if it is a disjunction of one or more conjunctions of one or more literals.

Distribution of $\land$ over $\lor$ allows you to transform any sentence in negation normal form into disjunctive normal form.

Some sentences are in both CNF and DNF.