The Importance of Being Formal

Martin Henz

February 12, 2014
1. Review: Syntax and Semantics of Propositional Logic
2. Natural Deduction
3. Soundness and Completeness
Review: Syntax and Semantics of Propositional Logic
- Propositional Atoms and Propositions
- Semantics of Formulas
- Validity, Satisfiability, Truth Tables

Natural Deduction

Soundness and Completeness
Atoms

Anything goes

We allow any kind of proposition, for example “The sun is shining today”.
Atoms

Anything goes

We allow any kind of proposition, for example “The sun is shining today”.

Convention

We usually use $p$, $q$, $p_1$, etc, instead of sentences like “The sun is shining today”.
Atoms

Anything goes

We allow any kind of proposition, for example “The sun is shining today”.

Convention

We usually use $p$, $q$, $p_1$, etc, instead of sentences like “The sun is shining today”.

Atoms

More formally, we fix a set $A$ of propositional atoms.
Meaning of Atoms

Models assign truth values

A *model* assigns truth values (*F* or *T*) to each atom.
Meaning of Atoms

Models assign truth values

A *model* assigns truth values \((F\text{ or } T)\) to each atom.

More formally

A model for a propositional logic for the set \(A\) of atoms is a mapping from \(A\) to \(\{T, F\}\).
Meaning of Atoms

Models assign truth values

A *model* assigns truth values (*F* or *T*) to each atom.

More formally

A model for a propositional logic for the set *A* of atoms is a mapping from *A* to \{*T*, *F*\}.

How do you call them?

Models for propositional logic are called *valuations*. 
Definition

For a given set $A$ of propositional atoms, the set of *well-formed formulas in propositional logic* is the smallest set $\mathcal{F}$ that fulfills the following rules:

- The constant symbols $\bot$ and $\top$ are in $\mathcal{F}$.
- Every element of $A$ is in $\mathcal{F}$.
- If $\phi$ is in $\mathcal{F}$, then $(\neg \phi)$ is also in $\mathcal{F}$.
- If $\phi$ and $\psi$ are in $\mathcal{F}$, then $(\phi \land \psi)$ is also in $\mathcal{F}$.
- If $\phi$ and $\psi$ are in $\mathcal{F}$, then $(\phi \lor \psi)$ is also in $\mathcal{F}$.
- If $\phi$ and $\psi$ are in $\mathcal{F}$, then $(\phi \rightarrow \psi)$ is also in $\mathcal{F}$.
Example

\[((\lnot p) \land q) \to (\top \land (q \lor (\lnot r)))\]

is a well-formed formula in propositional logic.

If *the train arrives late* and *there are no taxis at the station*, then
*John is late for his meeting.*

can be represented by \((p \land q) \to r\), with the respective abbreviations \(p, q, r\).
Evaluation of Formulas

**Definition**

The result of *evaluating* a well-formed propositional formula $\phi$ with respect to a valuation $v$, denoted $v(\phi)$ is defined as follows:

- If $\phi$ is the constant $\bot$, then $v(\phi) = F$.
- If $\phi$ is the constant $\top$, then $v(\phi) = T$.
- If $\phi$ is a propositional atom $p$, then $v(\phi) = p^v$.
- If $\phi$ has the form $(\neg \psi)$, then $v(\phi) = \neg v(\psi)$.
- If $\phi$ has the form $(\psi \land \tau)$, then $v(\phi) = v(\psi) \& v(\tau)$.
- If $\phi$ has the form $(\psi \lor \tau)$, then $v(\phi) = v(\psi) \mid v(\tau)$.
- If $\phi$ has the form $(\psi \rightarrow \tau)$, then $v(\phi) = v(\psi) \Rightarrow v(\tau)$.
Validity

A formula is *valid* if it computes $\top$ for all its valuations.
Validity

A formula is *valid* if it computes $\top$ for all its valuations.

Satisfiability

A formula is *satisfiable* if it computes $\top$ for at least one of its valuations.
### Example of Truth Table

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$(\neg p)$</th>
<th>$\neg q$</th>
<th>$p \rightarrow \neg q$</th>
<th>$q \lor \neg p$</th>
<th>$(p \rightarrow \neg q) \rightarrow (q \lor \neg p)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</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>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>
1. Review: Syntax and Semantics of Propositional Logic

2. Natural Deduction
   - The Problem with Truth Tables
   - Sequents
   - Basic Rules
   - Basic and Derived Rules

3. Soundness and Completeness
The Problem with Truth Tables

The problem

Truth tables get very large.

How large?

Consider for example prop.txt from verifying the correctness of a railroad station.
The problem
Truth tables get very large.
The Problem with Truth Tables

The problem

Truth tables get very large. How large?
The Problem with Truth Tables

The problem
Truth tables get very large. How large?

Consider for example
prop.txt from verifying the correctness of a railroad station
Practical Examples

Timetabling and scheduling

Satisfiability of propositional formulas is used for solving scheduling problems (timetabling, sports tournaments etc)
Practical Examples

Timetabling and scheduling

Satisfiability of propositional formulas is used for solving scheduling problems (timetabling, sports tournaments etc) (hundreds or thousands of atoms, thousands of operators)
Practical Examples

**Timetabling and scheduling**

Satisfiability of propositional formulas is used for solving scheduling problems (timetabling, sports tournaments etc) (hundreds or thousands of atoms, thousands of operators)

**Computer hardware design**

Satisfiability of propositional formulas is used for designing and verifying the correctness of the hardware (millions of atoms, millions of operators)
Practical Examples

Timetabling and scheduling
Satisfiability of propositional formulas is used for solving scheduling problems (timetabling, sports tournaments etc) (hundreds or thousands of atoms, thousands of operators)

Computer hardware design
Satisfiability of propositional formulas is used for designing and verifying the correctness of the hardware (millions of atoms, millions of operators)

Software verification using model checking
Satisfiability of propositional formulas is used for verifying the correctness of software (embedded systems etc) (hundreds of thousands of atoms and operators)
Objective

We would like to develop a calculus for reasoning about propositions, so that we can establish the validity of statements such as “If the train arrives late...”.
Objective

We would like to develop a calculus for reasoning about propositions, so that we can establish the validity of statements such as “If the train arrives late...”.

Idea

We introduce proof rules that allow us to derive a formula $\psi$ from a number of other formulas $\phi_1, \phi_2, \ldots \phi_n$. 
Objective

We would like to develop a calculus for reasoning about propositions, so that we can establish the validity of statements such as “If the train arrives late...”.

Idea

We introduce proof rules that allow us to derive a formula \( \psi \) from a number of other formulas \( \phi_1, \phi_2, \ldots, \phi_n \).

Notation

We write a sequent \( \phi_1, \phi_2, \ldots, \phi_n \vdash \psi \) to denote that we can derive \( \psi \) from \( \phi_1, \phi_2, \ldots, \phi_n \).
Example—Revisited

English

If the train arrives late and there are no taxis at the station, then John is late for his meeting.

John is not late for his meeting. The train did arrive late.

Therefore, there were taxis at the station.
Example—Revisited

**English**

If *the train arrives late* and *there are no taxis at the station*, then *John is late for his meeting.*

*John is not late for his meeting. The train did arrive late.*

*Therefore, there were taxis at the station.*

**Sequent**

\[ p \land \neg q \rightarrow r, \neg r, p \vdash q \]
Example—Revisited

English

If *the train arrives late* and *there are no taxis at the station*, then *John is late for his meeting.*

*John is not late for his meeting.* *The train did arrive late.*

Therefore, *there were taxis at the station.*

Sequent

\[ p \land \neg q \rightarrow r, \neg r, p \vdash q \]

Remaining task

Find proof rules that allow us to establish such sequents.
Rules for Conjunction

Introduction of Conjunction

\[ \frac{\phi \quad \psi}{\phi \wedge \psi} \quad [\wedge i] \]
Rules for Conjunction

Introduction of Conjunction

\[ \frac{\phi \quad \psi}{\phi \land \psi} ^{\land i} \]

Elimination of Conjunction

\[ \frac{\phi \land \psi}{\phi} ^{\land e_1} \]
\[ \frac{\phi \land \psi}{\psi} ^{\land e_2} \]
Example of Proof

To show

\[ p \land q, r \vdash q \land r \]

How to start?

\[ p \land q \quad r \]

\[ q \land r \]
Proof Step-by-Step

1. \( p \land q \) (premise)
Proof Step-by-Step

1. \( p \land q \) (premise)
2. \( r \) (premise)
Proof Step-by-Step

1. \( p \land q \) (premise)
2. \( r \) (premise)
3. \( q \) (by using Rule \( \land e_2 \) and Item 1)
Proof Step-by-Step

1. \( p \land q \) (premise)
2. \( r \) (premise)
3. \( q \) (by using Rule \( \land e_2 \) and Item 1)
4. \( q \land r \) (by using Rule \( \land i \) and Items 3 and 2)
Graphical Representation of Proof

\[ p \land q \]
\[ \quad \quad \quad \quad [\land e_2] \quad r \]
\[ q \]
\[ \quad \quad \quad \quad \quad \quad [\land e_2] \quad \quad \quad [\land i] \]
\[ q \land r \]
Graphical Representation of Proof

\[ \frac{p \land q}{\vdash q \land r} \]

Find the parts of the corresponding sequent:

\[ p \land q, r \vdash q \land r \]
The Importance of Being Formal 03—Propositional Logic II

Graphical Representation of Proof

\[ p \land q \]

\[ \frac{r}{q} \quad [\land e_2] \]

\[ q \]

\[ \frac{q \land r}{q \land r} \quad [\land i] \]
Graphical Representation of Proof

\[ p \land q \]

\[ [\land e_2] \quad r \]

\[ q \]

\[ [\land i] \quad q \land r \]

Find the parts of the corresponding proof:

1. \( p \land q \) (premise)
2. \( r \) (premise)
3. \( q \) (by using Rule \( \land e_2 \) and Item 1)
4. \( q \land r \) (by using Rule \( \land i \) and Items 3 and 2)
Where are we heading with this?

- We would like to prove sequents of the form

\[ \phi_1, \phi_2, \ldots, \phi_n \vdash \psi \]
Where are we heading with this?

- We would like to prove sequents of the form
  \[ \phi_1, \phi_2, \ldots, \phi_n \vdash \psi \]
- We introduce rules that allow us to form “legal” proofs
Where are we heading with this?

- We would like to prove sequents of the form
  \[ \phi_1, \phi_2, \ldots, \phi_n \vdash \psi \]
- We introduce rules that allow us to form “legal” proofs
- Then any proof of any formula \( \psi \) using the premises \( \phi_1, \phi_2, \ldots, \phi_n \) is considered “correct”.
Where are we heading with this?

- We would like to prove sequents of the form
  \[ \phi_1, \phi_2, \ldots, \phi_n \vdash \psi \]
- We introduce rules that allow us to form “legal” proofs
- Then any proof of any formula \( \psi \) using the premises \( \phi_1, \phi_2, \ldots, \phi_n \) is considered “correct”.
- Can we say that sequents with a correct proof are somehow “valid”, or “meaningful”? 
Where are we heading with this?

- We would like to prove sequents of the form 
  \[ \phi_1, \phi_2, \ldots, \phi_n \vdash \psi \]
- We introduce rules that allow us to form “legal” proofs
- Then any proof of any formula \( \psi \) using the premises 
  \( \phi_1, \phi_2, \ldots, \phi_n \) is considered “correct”.
- Can we say that sequents with a correct proof are somehow “valid”, or “meaningful”?
- What does it mean to be meaningful?
Where are we heading with this?

- We would like to prove sequents of the form
  \[ \phi_1, \phi_2, \ldots, \phi_n \vdash \psi \]
- We introduce rules that allow us to form “legal” proofs
- Then any proof of any formula \( \psi \) using the premises \( \phi_1, \phi_2, \ldots, \phi_n \) is considered “correct”.
- Can we say that sequents with a correct proof are somehow “valid”, or “meaningful”?
- What does it mean to be meaningful?
- Can we say that any meaningful sequent has a valid proof?
Where are we heading with this?

- We would like to prove sequents of the form
  \[ \phi_1, \phi_2, \ldots, \phi_n \vdash \psi \]
- We introduce rules that allow us to form “legal” proofs
- Then any proof of any formula \( \psi \) using the premises \( \phi_1, \phi_2, \ldots, \phi_n \) is considered “correct”.
- Can we say that sequents with a correct proof are somehow “valid”, or “meaningful”?
- What does it mean to be meaningful?
- Can we say that any meaningful sequent has a valid proof?
- ...but first back to the proof rules...
Rules of Double Negation

\[
\begin{align*}
\neg\neg\phi & \quad \rightarrow \quad \neg\neg i \\
\phi & \quad \rightarrow \quad \neg\neg e
\end{align*}
\]
Rule for Eliminating Implication

\[
\phi \quad \phi \rightarrow \psi \\
\hline
\psi \\
\]
Rule for Eliminating Implication

$\phi \quad \phi \rightarrow \psi$

$\frac{}{\psi \quad \rightarrow \ e}$

Example

$p$: It rained.

$p \rightarrow q$: If it rained, then the street is wet.

We can conclude from these two that the street is indeed wet.
Another Rule for Eliminating Implication

The rule

\[
\frac{\phi \quad \phi \rightarrow \psi}{\psi}
\]

is often called “Modus Ponens” (or MP)
Another Rule for Eliminating Implication

The rule

\[ \phi \quad \phi \rightarrow \psi \]

\[ \therefore \psi \]

is often called “Modus Ponens” (or MP)

**Origin of term**

“Modus ponens” is an abbreviation of the Latin “modus ponendo ponens” which means in English “mode that affirms by affirming”. More precisely, we could say “mode that affirms the antecedent of an implication”.

The Twin Sister of Modus Ponens

The rule

\[ \phi \quad \phi \rightarrow \psi \]

\[ \frac{}{\psi} \] \[\rightarrow e\]

is called “Modus Ponens” (or MP)
The Twin Sister of Modus Ponens

The rule

\[
\begin{align*}
\phi & \\
\phi \to \psi & \\
\hline
\psi
\end{align*}
\]

is called “Modus Ponens” (or MP)

A similar rule

\[
\begin{align*}
\phi & \to \psi \\
\neg \psi & \\
\hline
\neg \phi
\end{align*}
\]

is called “Modus Tollens” (or MT).
The Twin Sister of Modus Ponens

The rule

\[ \phi \rightarrow \psi \quad \neg \psi \]

\[ \frac{}{\neg \phi \quad [MT]} \]

is called “Modus Tollens” (or MT).
The Twin Sister of Modus Ponens

The rule

\[
\phi \rightarrow \psi, \neg \psi \quad \Rightarrow \quad \neg \phi
\]

is called “Modus Tollens” (or MT).

Origin of term

“Modus tollens” is an abbreviation of the Latin “modus tollendo tollens” which means in English “mode that denies by denying”. More precisely, we could say “mode that denies the consequent of an implication”.
Example

\[ p \rightarrow (q \rightarrow r), p, \neg r \vdash \neg q \]
Example

\[ p \rightarrow (q \rightarrow r), p, \neg r \vdash \neg q \]

1. \[ p \rightarrow (q \rightarrow r) \] premise
Example

\[ \begin{align*}
\vdash & \quad p \rightarrow (q \rightarrow r), p, \neg r \vdash \neg q \\
1 & \quad p \rightarrow (q \rightarrow r) \quad \text{premise} \\
2 & \quad p \quad \text{premise}
\end{align*} \]
Example

\[ p \rightarrow (q \rightarrow r), p, \neg r \vdash \neg q \]

1. \( p \rightarrow (q \rightarrow r) \) premise
2. \( p \) premise
3. \( \neg r \) premise
Example

\[ p \rightarrow (q \rightarrow r), \ p, \ \neg r \vdash \neg q \]

1. \( p \rightarrow (q \rightarrow r) \)  
   - premise
2. \( p \)  
   - premise
3. \( \neg r \)  
   - premise
4. \( q \rightarrow r \)  
   - \( \rightarrow_e \, 1,2 \)
Example

\[ p \rightarrow (q \rightarrow r), p, \neg r \vdash \neg q \]

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( p \rightarrow (q \rightarrow r) )</td>
<td>premise</td>
</tr>
<tr>
<td>2</td>
<td>( p )</td>
<td>premise</td>
</tr>
<tr>
<td>3</td>
<td>( \neg r )</td>
<td>premise</td>
</tr>
<tr>
<td>4</td>
<td>( q \rightarrow r )</td>
<td>( \rightarrow_e 1,2 )</td>
</tr>
<tr>
<td>5</td>
<td>( \neg q )</td>
<td>MT 4,3</td>
</tr>
</tbody>
</table>
How to *introduce* implication?

Compare the sequent (MT)

\[ p \rightarrow q, \neg q \vdash \neg p \]

with the sequent

\[ p \rightarrow q \vdash \neg q \rightarrow \neg p \]
How to *introduce* implication?

Compare the sequent (MT)

\[ p \rightarrow q, \neg q \vdash \neg p \]

with the sequent

\[ p \rightarrow q \vdash \neg q \rightarrow \neg p \]

The second sequent should be provable, but we don’t have a rule to introduce implication yet!
### A Proof We Would Like To Have

\[ p \rightarrow q \vdash \neg q \rightarrow \neg p \]

<table>
<thead>
<tr>
<th></th>
<th>( p \rightarrow q )</th>
<th>premise</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( p \rightarrow q )</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>( \neg q )</td>
<td>assumption</td>
</tr>
<tr>
<td>3</td>
<td>( \neg p )</td>
<td>MT 1,2</td>
</tr>
<tr>
<td>4</td>
<td>( \neg q \rightarrow \neg p )</td>
<td>( \rightarrow i ) 2–3</td>
</tr>
</tbody>
</table>
A Proof We Would Like To Have

\[ p \rightarrow q \vdash \neg q \rightarrow \neg p \]

1. \( p \rightarrow q \)  \( \text{premise} \)
2. \( \neg q \)  \( \text{assumption} \)
3. \( \neg p \)  \( \text{MT 1,2} \)
4. \( \neg q \rightarrow \neg p \)  \( \rightarrow i \ 2–3 \)

We can start a box with an *assumption*, and use previously proven propositions (including premises) from the outside in the box.
A Proof We Would Like To Have

\[ p \to q \vdash \neg q \to \neg p \]

1. \( p \to q \) premise
2. \( \neg q \) assumption
3. \( \neg p \) MT 1,2
4. \( \neg q \to \neg p \) \( \to i \ 2-3 \)

We can start a box with an assumption, and use previously proven propositions (including premises) from the outside in the box.
We cannot use assumptions from inside the box in rules outside the box.
Rule for Introduction of Implication

\[ \phi \rightarrow \psi \]

\[ \phi \rightarrow \psi \]
Rules for Introduction of Disjunction

\[ \begin{align*}
\phi & \quad & [\lor i_1] \\
\phi \lor \psi & \quad & \\
\end{align*} \]

\[ \begin{align*}
\psi & \quad & [\lor i_2] \\
\phi \lor \psi & \quad & \\
\end{align*} \]
Rule for Elimination of Disjunction

\[ \phi \lor \psi \]

\[ \vdots \]

\[ \chi \]

\[ \psi \]

\[ \vdots \]

\[ \chi \]

\[ [\lor e] \]

\[ \chi \]
Example

1. $p \land (q \lor r)$  
   - premise
2. $p$  
   - $\land e_1$ 1
3. $q \lor r$  
   - $\land e_2$ 1

4. $q$  
   - assumption
5. $p \land q$  
   - $\land i$ 2,4
6. $(p \land q) \lor (p \land r)$  
   - $\lor i_1$ 5

7. $r$  
   - assumption
8. $p \land r$  
   - $\land i$ 2,7
9. $(p \land q) \lor (p \land r)$  
   - $\lor i_2$ 8

10. $(p \land q) \lor (p \land r)$  
    - $\lor e$ 3, 4–6, 7–9

The Importance of Being Formal
Rules for Eliminating Implication

\[
\begin{align*}
\phi & \quad \phi \rightarrow \psi \\
\hline
\phi \rightarrow \psi & \quad \neg \psi \\
\hline
\psi & \quad \neg \phi
\end{align*}
\]
Rule for Introduction of Implication

\[ \phi \rightarrow \psi \]
Elimination of Negation

\[ \phi \quad \neg \phi \]

\[ \frac{}{\bot} \quad \neg \text{e} \]
Introduction of Negation

\[
\neg \phi
\]

The Importance of Being Formal
Elimination of $\bot$

\[
\begin{align*}
\bot \\
\hline
\phi
\end{align*}
\]

$\bot e$
Basic Rules (conjunction and disjunction)

\[
\begin{align*}
\phi & \quad \psi \\
\hline
\phi \land \psi & \quad [\land i] \quad \phi \land \psi \\
\hline
\phi & \\
\hline
\psi & \\
\hline
\phi \land \psi & \quad [\land e_1] \\
\hline
\phi & \\
\hline
\psi & \\
\hline
\phi \land \psi & \quad [\land e_2] \\
\hline
\phi & \\
\hline
\psi & \\
\hline
\phi \lor \psi & \quad [\lor i_1] \\
\hline
\phi & \\
\hline
\psi & \\
\hline
\phi \lor \psi & \quad [\lor e_1] \\
\hline
\phi & \\
\hline
\psi & \\
\hline
\phi \lor \psi & \\
\hline
\phi \lor \psi & \quad [\lor i_2] \\
\hline
\phi & \\
\hline
\psi & \\
\hline
\phi \lor \psi & \\
\hline
\phi \lor \psi & \quad [\lor e] \\
\hline
\phi & \\
\hline
\psi & \\
\hline
\chi & \\
\hline
\chi & \\
\hline
\chi & \\
\end{align*}
\]
Basic Rules (implication)

\[
\begin{align*}
\phi & \quad \vdash \rightarrow i \\
\phi \rightarrow \psi & \\
\phi & \quad \vdash \rightarrow e \\
\psi & \quad \phi \rightarrow \psi \\
\end{align*}
\]
Basic Rules (negation)

\[ \begin{array}{c}
\phi \\
\vdots \\
\bot \\
\hline
\neg \phi \\
\end{array} \quad \text{[\neg i]} \\
\]

\[ \begin{array}{c}
\phi \\
\neg \phi \\
\bot \\
\hline
\end{array} \quad \text{[\neg e]} \\
\]
Basic Rules (⊥ and double negation)

\[
\begin{align*}
\bot & \quad [\bot e] \\
\phi & \\
\neg \neg \phi & \quad [\neg \neg e] \\
\phi &
\end{align*}
\]
Some Derived Rules: Introduction of Double Negation

\[ \phi \]

\[ \underline{\phi} \]

\[ \underline{\underline{\neg \neg \phi}} \]
Example: Deriving $[\neg\neg \phi]$ from $[\neg \phi]$ and $[\neg \psi]$.

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\phi$</td>
<td>premise</td>
</tr>
<tr>
<td>2</td>
<td>$\neg \phi$</td>
<td>assumption</td>
</tr>
<tr>
<td>3</td>
<td>$\bot$</td>
<td>$\neg \psi$ 1,2</td>
</tr>
<tr>
<td>4</td>
<td>$\neg \neg \phi$</td>
<td>$\neg \phi$ 2–3</td>
</tr>
</tbody>
</table>
Some Derived Rules: Modus Tollens

\[ \phi \rightarrow \psi \quad \neg \psi \]

\[ \frac{}{\neg \phi} \quad \text{[MT]} \]
Some Derived Rules: Proof By Contradiction

\[ \neg \phi \]

\[ \vdots \]

\[ \bot \]

\[ \phi \]

\[ \text{[PBC]} \]
Some Derived Rules: Law of Excluded Middle

\[ \phi \lor \neg \phi \]

[LEM]
1. Review: Syntax and Semantics of Propositional Logic
2. Natural Deduction
3. Soundness and Completeness
Definition

If there is a natural deduction proof of $\psi$ using the premises $\phi_1, \phi_2, \ldots, \phi_n$, we say that $\psi$ is provable from $\phi_1, \phi_2, \ldots, \phi_n$ and write

$$\phi_1, \phi_2, \ldots, \phi_n \vdash \psi$$
Semantic Entailment

Definition

If, for all valuations in which all $\phi_1, \phi_2, \ldots, \phi_n$ evaluate to $\top$, the formula $\psi$ evaluates to $\top$ as well, we say that $\phi_1, \phi_2, \ldots, \phi_n$ semantically entail $\psi$, written:

$$\phi_1, \phi_2, \ldots, \phi_n \models \psi$$
Semantic equivalence

Let $\phi$ and $\psi$ be formulas of propositional logic. We say that $\phi$ and $\psi$ are semantically equivalent iff $\phi \models \psi$ and $\psi \models \phi$ hold. We write $\psi \equiv \phi$. 
Semantic equivalence

Let \( \phi \) and \( \psi \) be formulas of propositional logic. We say that \( \phi \) and \( \psi \) are semantically equivalent iff \( \phi \models \psi \) and \( \psi \models \phi \) hold. We write \( \psi \equiv \phi \).

Validity

If \( \models \phi \) holds, we call \( \phi \) valid.
Soundness

Let $\phi_1, \phi_2, \ldots, \phi_n$ and $\psi$ be propositional formulas. If $\phi_1, \phi_2, \ldots, \phi_n \vdash \psi$, then $\phi_1, \phi_2, \ldots, \phi_n \models \psi$. 
Completeness

Let $\phi_1, \phi_2, \ldots, \phi_n$ and $\psi$ be propositional formulas. If $\phi_1, \phi_2, \ldots, \phi_n \models \psi$, then $\phi_1, \phi_2, \ldots, \phi_n \vdash \psi$. 
Questions about Propositional Formula

- Is a given formula valid?
- Is a given formula satisfiable?
- Is a given formula invalid?
- Is a given formula unsatisfiable?
- Are two formulas equivalent?
Definition

A decision problem is a question in some formal system with a yes-or-no answer.
Definition

A decision problem is a question in some formal system with a yes-or-no answer.

Examples

The question whether a given propositional formula is satisifiable (unsatisfiable, valid, invalid) is a decision problem.

The question whether two given propositional formulas are equivalent is also a decision problem.
How to Solve the Decision Problem?

Question

How do you decide whether a given propositional formula is satisfiable/valid?
How to Solve the Decision Problem?

**Question**
How do you decide whether a given propositional formula is satisfiable/valid?

**The good news**
We can construct a truth table for the formula and check if some/all rows have $\top$ in the last column.
An algorithm for satsifiability

Using a truth table, we can implement an algorithm (systematic method) that returns “yes” if the formula is satisifiable, and that returns “no” if the formula is unsatisfiable.
Satisfiability is Decidable

An algorithm for satisfiability

Using a truth table, we can implement an algorithm (systematic method) that returns “yes” if the formula is satisfiable, and that returns “no” if the formula is unsatisfiable.

Decidability

Decision problems for which there is an algorithm computing “yes” whenever the answer is “yes”, and “no” whenever the answer is “no”, are called decidable.
Satisifiability is Decidable

An algorithm for satisifiability

Using a truth table, we can implement an algorithm (systematic method) that returns “yes” if the formula is satisifiable, and that returns “no” if the formula is unsatisfiable.

Decidability

Decision problems for which there is an algorithm computing “yes” whenever the answer is “yes”, and “no” whenever the answer is “no”, are called decidable.

Decidability of satisifiability

The question, whether a given propositional formula is satisifiable, is decidable.
Next homework: Homework 3: Due Wednesday morning before class

Wednesday material: Predicate Logic

After that: Homework 4: Due Friday morning before class