PROPOSITIONAL LOGIC II

Proof Theory

- What is logic used for? A number of things, but most importantly, it is a *language for representing the properties of things.*
- But also, we hope it will give us a *method for establishing the properties of things*.
- To see how logic may be used to establish the properties of things, it helps to look at its history.
- Logic was originally developed to make the notion of an *argument* precise.

  (We do *not* mean argument as in fighting here!)

- Here is a classic argument:

All men are mortal
Socrates is a man
―――――――――――
Socrates is mortal

- This example serves to illustrate a number of features of arguments:
  - The argument has a number of *premises* — these are the statements that appear *before* the horizontal line;
  - The argument has a *conclusion* — this is the statement that appears *after* the horizontal line;
  - The argument has the form

If
  you accept that
  the premises are true
then
  you must accept that
  the conclusion is true.

- In mathematics, we are concerned with when arguments are *sound*.
- To formalise the notion of a sound argument, we need some extra terminology…
- **Definition:** If $\phi \in \mathcal{W}$, then:
  1. if there is *some* interpretation π such that
  $$\pi \models \phi$$
  then ϕ is said to be *satisfiable*, otherwise ϕ is *unsatisfiable*.
  2. if
  $$\pi \models \phi$$
  for *all* interpretations π, then ϕ is said to be *valid*.
- Valid formulae of propositional logic are called *tautologies*.

- **Theorem 16.1** *1. If ϕ is a valid formula, then $\neg\phi$ is unsatisfiable;*
  2. If $\neg\phi$ is unsatisfiable, then ϕ is valid.
- We indicate that a formula ϕ is valid by writing
$$\models \phi.$$
- We can now define the *logical consequence*.

- **Definition:** If
$$\{\phi_1, \ldots, \phi_n, \phi\} \subseteq \mathcal{W}$$
then ϕ is said to be a *logical consequence* of $\{\phi_1, \ldots, \phi_n\}$ iff ϕ is satisfied by all interpretations that satisfy
$$\phi_1 \wedge \cdots \wedge \phi_n.$$
- We indicate that ϕ is a logical consequence of $\phi_1, \ldots, \phi_n$ by writing
$$\{\phi_1, \ldots, \phi_n\} \models \phi.$$
- An expression like this is called a *semantic sequent*.

- **Theorem 16.2**
$$\{\phi_1, \ldots, \phi_n\} \models \phi.$$
iff
$$\models (\phi_1 \wedge \cdots \wedge \phi_n) \Rightarrow \phi.$$
- So we have a method for determining whether ϕ is a logical consequence of $\phi_1, \ldots \phi_n$: we use a truth table to see whether $\phi_1 \wedge \cdots \wedge \phi_n \Rightarrow \phi$ is a tautology. If it is, then ϕ is a logical consequence of $\phi_1, \ldots, \phi_n$.
- *Our main concern in proof theory is thus to have a technique for determining whether a given formula is valid, as this will then give us a technique for determining whether some formula is a logical consequence of some others.*

- EXAMPLE. Show that

$$p \wedge q \models p \vee q.$$

To do this, we construct a truth-table for

$$(p \wedge q) \Rightarrow (p \vee q).$$

Here it is:

		(1)	(2)	
p	q	$p \wedge q$	$p \vee q$	$(1) \Rightarrow (2)$
F	F	F	F	T
F	T	F	T	T
T	F	F	T	T
T	T	T	T	T

Since

$$(p \wedge q) \Rightarrow (p \vee q).$$

is true under every interpretation, we have that $p \vee q$ is a logical consequence of $p \wedge q$.

- The notion of logical consequence we have defined above is acceptable for a *definition* of a sound argument, but is not very helpful for checking whether a particular argument is sound or not.
- The problem is that we must look at all the possible interpretations of the primitive propositions. While this is acceptable for, say, 4 primitive propositions, it will clearly be unacceptable for 100 propositions, as it would mean checking 2^{100} interpretations.

  (Moreover, for first-order logic, there will be an *infinite* number of such interpretations.)
- What we require instead is an alternative version of logical consequence, that does not involve this kind of checking. This leads us to the idea of *syntactic* proof.

'Syntactic' Proof

- The idea of syntactic proof is to replace the semantic checking to determine whether a formula is valid by a procedure that involves purely *syntactic* manipulation.
- The kinds of techniques that we shall use are similar to those that we use when solving problems in algebra.
- The basic idea is that to show that ϕ is a logical consequence of $\phi_1, \ldots, \phi_n$, we use a set of *rules* to manipulate formulae.

  If we can derive ϕ from $\phi_1, \ldots, \phi_n$ by using these rules, then ϕ is said to be *proved* from $\phi_1, \ldots, \phi_n$, which we indicate by writing

$$\phi_1, \ldots, \phi_n \vdash \phi.$$

- The symbol $\vdash$ is called the *syntactic turnstile*.
- An expression of the form

$$\phi_1, \ldots, \phi_n \vdash \phi.$$

  is called a *syntactic sequent*.
- A rule has the general form:

$$\frac{\vdash \phi_1; \cdots; \vdash \phi_n}{\vdash \phi} \text{ rule name}$$

  Such a rule is read:

  If
  $\phi_1, \ldots, \phi_n$ are proved
  then
  ϕ is proved.

- EXAMPLE. Here is an example of such a rule:

$$\frac{\vdash \phi; \vdash \psi}{\vdash \phi \wedge \psi} \wedge\text{-I}$$

This rule is called *and introduction*. It says that if we have proved ϕ, and we have also proved ψ, then we can prove $\phi \wedge \psi$.

- EXAMPLE. Here is another rule:

$$\frac{\vdash \phi \wedge \psi}{\vdash \phi; \vdash \psi} \wedge\text{-E}$$

This rule is called *and elimination*. It says that if we have proved $\phi \wedge \psi$, then we can prove both ϕ and ψ; it allows us to eliminate the $\wedge$ symbol from between them.

- Let us now try to define precisely what we mean by *proof*.
- **Definition:** (Proof) If

$$\{\phi_1, \ldots, \phi_m, \phi\} \subseteq \mathcal{W}$$

then there is a proof of ϕ from $\phi_1, \ldots, \phi_m$ iff there exists some sequence of formulae

$$\psi_1, \ldots, \psi_n$$

such that $\psi_n = \phi$, and each formula ψ_k, for $1 \leq k < n$ is either one of the formula $\phi_1, \ldots, \phi_m$, or else is the conclusion of a rule whose antecedents appeared earlier in the sequence.

- If there is a proof of ϕ from $\phi_1, \ldots, \phi_n$, then we indicate this by writing:

$$\phi_1, \ldots, \phi_m \vdash \phi.$$

- It should be clear that the symbols $\vdash$ and $\models$ are related. We now have to state exactly *how* they are related.
- There are two properties of $\vdash$ to consider:
  - *soundness*;
  - *completness*.
  - Intuitively, $\vdash$ is said to be *sound* if it is correct, in that it does not let us derive something that is not true.
  - Intuitively, *completeness* means that $\vdash$ will let us prove anything that is true.

- **Definition:** (Soundness) A proof system $\vdash$ is said to be *sound* with respect to semantics $\models$ iff

$$\phi_1, \ldots, \phi_n \vdash \phi$$

implies

$$\phi_1, \ldots, \phi_n \models \phi.$$

- **Definition:** (Completeness) A proof system $\vdash$ is said to be *complete* with respect to semantics $\models$ iff

$$\phi_1, \ldots, \phi_n \models \phi$$

implies

$$\phi_1, \ldots, \phi_n \vdash \phi.$$

A Proof System

- There are many proof systems for propositional logic; we shall look at a simple one.
- First, we have an unusual rule that allows us to introduce any tautology.

$$\frac{}{\vdash \phi}\ \text{TAUT} \quad \text{if } \phi \text{ is a tautology}$$

- Because a tautology is true there is no problem bringing it into the proof.

- Next, rules for *eliminating* connectives.

$$\frac{\vdash \phi \wedge \psi}{\vdash \phi; \vdash \psi}\ \wedge\text{-E}$$

$$\frac{\begin{array}{l}\vdash \phi_1 \vee \cdots \vee \phi_n;\\ \phi_1 \vdash \phi;\\ \quad \cdots;\\ \phi_n \vdash \phi\end{array}}{\vdash \phi}\ \vee\text{-E}$$

- An alternative $\vee$ elimination rule is:

$$\frac{\begin{array}{l}\vdash \phi \vee \psi;\\ \vdash \phi \Rightarrow \chi;\\ \vdash \psi \Rightarrow \chi\end{array}}{\vdash \chi}\ \vee\text{-E}$$

- Next, a rule called *modus ponens*, which lets us eliminate $\Rightarrow$.

$$\frac{\vdash \phi \Rightarrow \psi; \vdash \phi}{\vdash \psi}\ \Rightarrow\text{-E}$$

- Next, rules for *introducing* connectives.

$$\frac{\vdash \phi_1; \cdots; \vdash \phi_n}{\vdash \phi_1 \wedge \cdots \wedge \phi_n}\ \wedge\text{-I}$$

$$\frac{\vdash \phi_1; \cdots; \phi_n}{\vdash \phi_1 \vee \cdots \vee \phi_n}\ \vee\text{-I}$$

- We have a rule called the *deduction theorem*. This rule says that if we can prove ψ from ϕ, then we can prove that $\phi \Rightarrow \psi$.

$$\frac{\phi \vdash \psi}{\vdash \phi \Rightarrow \psi}\ \Rightarrow\text{-I}$$

- There are a whole range of other rules, which we shall not list here.

Proof Examples

- In this section, we give some examples of proofs in the propositional calculus.
- Example 1:

$p \wedge q \vdash q \wedge p$

1.	$p \wedge q$	Given
2.	p	From 1 using $\wedge$-E
3.	q	1,$\wedge$-E
4.	$q \wedge p$	2, 3, $\wedge$-I

- Example 2:

$p \wedge q \vdash p \vee q$

1.	$p \wedge q$	Given
2.	p	1, $\wedge$-E
3.	$p \vee q$	2, $\vee$-I

- Example 3:

$p \wedge q, p \Rightarrow r \vdash r$

1.	$p \wedge q$	Given
2.	p	1, $\wedge$-E
3.	$p \Rightarrow r$	Given
4.	r	2, 3, $\Rightarrow$-E

- Example 4:

$p \Rightarrow q, q \Rightarrow r \vdash p \Rightarrow r$

1.	$p \Rightarrow q$	Given	
2.	$q \Rightarrow r$	Given	
3.	p	Ass	|
4.	q	1, 3, $\Rightarrow$-E	|
5.	r	2, 4, $\Rightarrow$-E	|
6.	$p \Rightarrow r$	3, 5, $\Rightarrow$-I	

- Example 5:

$(p \wedge q) \Rightarrow r \vdash p \Rightarrow (q \Rightarrow r)$

1.	$(p \wedge q) \Rightarrow r$	Given	
2.	p	Ass	$\mid$
3.	q	Ass	$\mid\mid$
4.	$p \wedge q$	2, 3, $\wedge$-I	$\mid\mid$
5.	r	1, 4, $\Rightarrow$-I	$\mid\mid$
6.	$q \Rightarrow r$	3–5, $\Rightarrow$-I	$\mid$
7.	$p \Rightarrow (q \Rightarrow r)$	2–6, $\Rightarrow$-I	

- Example 6:

$p \Rightarrow (q \Rightarrow r) \vdash (p \wedge q) \Rightarrow r$

1.	$p \Rightarrow (q \Rightarrow r)$	Given	
2.	$p \wedge q$	Ass	$\mid$
3.	p	2, $\wedge$-E	$\mid$
4.	q	2, $\wedge$-E	$\mid$
5.	$q \Rightarrow r$	1, 3, $\Rightarrow$-E	$\mid$
6.	r	4, 5, $\Rightarrow$-E	$\mid$
7.	$(p \wedge q) \Rightarrow r$	2–6, $\Rightarrow$-I	

- Example 7:

$p \Rightarrow q, \neg q \vdash \neg p$

1.	$p \Rightarrow q$	Given	
2.	$\neg q$	Given	
3.	p	Ass	$\mid$
4.	q	1, 3, $\Rightarrow$-E	$\mid$
5.	$q \wedge \neg q$	2, 4, $\wedge$-I	$\mid$
6.	$\neg p$	3, 5, $\neg$-I	

- Example 8:

$p \Rightarrow q \vdash \neg(p \wedge \neg q)$

1.	$p \Rightarrow q$	Given	
2.	$p \wedge \neg q$	Ass	$\mid$
3.	p	2, $\wedge$-E	$\mid$
4.	$\neg q$	2, $\wedge$-E	$\mid$
5.	q	1, 3, $\Rightarrow$-E	$\mid$
6.	$q \wedge \neg q$	4, 5, $\wedge$-I	$\mid$
7.	$\neg(p \wedge \neg q)$	6, $\neg$-I	

- Example 9:

  Jim will party all night and pass AI? That must be wrong. If he works hard he won't have time to party. If he doesn't work hard he's not going to pass AI.

  Let:

  p Jim will party all night
  q Jim will pass AI
  r Jim works hard

  Formalisation of argument:

$$r \Rightarrow \neg p, \neg r \Rightarrow \neg q \vdash \neg(p \wedge q)$$

1.	$r \Rightarrow \neg p$	Given	
2.	$\neg r \Rightarrow \neg q$	Given	
3.	$p \wedge q$	Ass	|
4.	r	Ass	||
5.	$\neg p$	1, 4, $\Rightarrow$-E	||
6.	p	3, $\wedge$-I	||
7.	$p \wedge \neg p$	5, 6, $\wedge$-I	||
8.	$\neg r$	4, 7, $\neg$-I	|
9.	$\neg q$	2, 9, $\Rightarrow$-E	|
10.	q	3, $\wedge$-E	|
11.	$q \wedge \neg q$	9, 10, $\wedge$-I	|
12.	$\neg(p \wedge q)$	3, 11, $\neg$-I	

Proof as Search

- Proof problems can easily be formulated as *search*, in the way that we formulated other problems.
- Suppose we want to establish whether $\phi_1, \ldots, \phi_n \vdash \psi$.
- State space: sequence of formulae.
- Initial state: $\phi_1, \ldots, \phi_n$.
- Goal: sequence of formulae with last element ψ.
- Operators: rules, which when applied to some elements in sequence generate new formula appended to state.

- Problems:
  - no solution guaranteed — perhaps non-terminating;
  - no way of knowing "right" rule to apply.
- Huge amounts of work on *heuristics for proof*.
- Small sets of sound & complete rules better ...
- *resolution*
  - sound & complete proof method with just 1 rule.

Resolution

- Based on *checking satisfiability* of formulae.
- Relies on the fact that

$$\phi_1, \ldots, \phi_n \vdash \psi$$

iff

$$\phi_1 \wedge \cdots \wedge \phi_n \wedge \neg\psi$$

is unsatisfiable.

- So, negate what you want to show, add it to what you know, and try to show unsatisfiability.

- The resolution rule itself is:

$$\frac{\begin{array}{c}\vdash \phi \vee \psi \\ \vdash \chi \vee \neg\psi\end{array}}{\vdash \phi \vee \chi} \text{ resolution}$$

- Unsatisfiability is proved when we there is nothing left after we resolve two formulae together.

Summary

- This lecture continued our look at propositional logic.
- It concentrated on proof theory, and gave examples of a number of different kinds of proof.
- We also touched on the relationship between proof and search.
- And finally we looked briefly at resolution.