Reasoning in Propositional Logic

Lecture 3

February 6, 2018

Arguments

- An argument is a pair $(\mathcal{P}, \phi)$.
  - $\mathcal{P} = \{\psi_1, \psi_2, \ldots, \psi_n\}$ is a finite set of PL WFFs called **hypotheses** (or **premises**).
  - ϕ is a WFF called the **conclusion**.
- It is common to display arguments as

$$\begin{array}{c} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_n \\ \hline \phi \end{array}$$

Valid Arguments

- An argument $(\mathcal{P}, \phi)$ is valid if $\mathcal{P} \models \phi$.
- Are the following arguments valid?

The British PM is either a man or a woman
If the British PM is a father, then the British PM is a man

The British PM is a woman

The British PM is a man
If the British PM is a man, then the British PM is a father

The British PM is a father

Valid Arguments

- An argument $(\mathcal{P}, \phi)$ is valid if $\mathcal{P} \models \phi$.
- Are the following arguments valid?

Invalid

The British PM is either a man or a woman
If the British PM is a father, then the British PM is a man

The British PM is a woman

The British PM is a man
If the British PM is a man, then the British PM is a father

The British PM is a father

Valid Arguments

- An argument $(\mathcal{P}, \phi)$ is valid if $\mathcal{P} \models \phi$.
- Are the following arguments valid?

Invalid

The British PM is either a man or a woman

If the British PM is a father, then the British PM is a man

The British PM is a woman

Valid

The British PM is a man

If the British PM is a man, then the British PM is a father

The British PM is a father

Example

Prove that $(\{(P \Rightarrow Q), P\}, Q)$ is a valid argument.

- This argument is known as **modus ponens**.
- Do it yourself.

Note

- $(\mathcal{P}, \phi)$ is a valid argument if and only if

$$(\{\}, (\bigwedge_{\psi_i \in \mathcal{P}} \psi_i) \Rightarrow \phi)$$

  is a valid argument.

- $(\{\}, \phi)$ is a valid argument if and only if ϕ is a tautology.

Semantic Inference

- The term **semantic inference** is used to refer to the process of identifying valid arguments of a logic (and hence tautologies).
- A straightforward semantic inference method is that of constructing truth tables: given an argument, you can always use truth tables to determine whether it is valid. **(How?)**
- The running time of this algorithm is exponential in the number of propositional variables.
- This is extremely bad!

Sad News

- At this point in the history of computing there are no sub-exponential algorithms known for this problem.
- **SAT** is reducible to this problem of determining whether a given argument is valid.
  - SAT is the problem of determining whether a WFF of PL is satisfiable.
- SAT is a classical **NP-complete** problem.
- Thus, you either wait until someone comes up with an efficient algorithm, or you come up with one and gain the Turing award.

Wang's Algorithm

- But we can enhance the average-case complexity.
- One algorithm that does this is **Wang's algorithm.**[a]
- Given an argument $(\mathcal{P}, \phi)$, Wang's algorithm proceeds by trying to find an assignment $\mathcal{A}$ such that, for every $\psi \in \mathcal{P}$, $[\![\psi]\!]^{\mathcal{A}} = \top$ and $[\![\phi]\!]^{\mathcal{A}} = \bot$.
  - If it succeeds, then the argument is not valid.
  - If it fails, then the argument is valid.

[a]There is a link on the course web site to Wang's original paper.

The Operation of Wang's Algorithm

- The algorithm operates on two sets $\mathcal{T}$ and $\mathcal{F}$.
- It returns "True" if there is no assignment that satisfies $\mathcal{T}$ and falsifies $\mathcal{F}$.
- It returns "False" otherwise.
- The algorithm may be recursively defined as follows.

Wang: The Algorithm

Wang$(\mathcal{T}, \mathcal{F})$

1. If $\mathbf{T} \in \mathcal{F}$ or $\mathbf{F} \in \mathcal{T}$ or $\mathcal{T} \cap \mathcal{F} \neq \{\}$, return "True";

2. If $\mathcal{T} \cup \mathcal{F} \subseteq \mathcal{V}$, return "False";

3. If $\neg\phi \in \mathcal{T}$, return Wang$(\mathcal{T} - \{\neg\phi\},\ \mathcal{F} \cup \{\phi\})$;

4. If $\neg\phi \in \mathcal{F}$, return Wang$(\mathcal{T} \cup \{\phi\},\ \mathcal{F} - \{\neg\phi\})$;

5. If $(\phi \wedge \psi) \in \mathcal{T}$, return Wang$((\mathcal{T} - \{\phi \wedge \psi\}) \cup \{\phi, \psi\},\ \mathcal{F})$;

6. If $(\phi \wedge \psi) \in \mathcal{F}$,
return Wang$(\mathcal{T},\ (\mathcal{F} - \{\phi \wedge \psi\}) \cup \{\phi\})$
and Wang$(\mathcal{T},\ (\mathcal{F} - \{\phi \wedge \psi\}) \cup \{\psi\})$;

Wang: The Algorithm

7. If $(\phi \vee \psi) \in \mathcal{T}$,
return Wang$((\mathcal{T} - \{\phi \vee \psi\}) \cup \{\phi\},\ \mathcal{F})$
and Wang$(\mathcal{T} - \{\phi \vee \psi\}) \cup \{\psi\},\ \mathcal{F})$;

8. If $(\phi \vee \psi) \in \mathcal{F}$, return Wang$(\mathcal{T},\ (\mathcal{F} - \{\phi \vee \psi\}) \cup \{\phi, \psi\})$;

9. If $(\phi \Rightarrow \psi) \in \mathcal{T}$,
return Wang$((\mathcal{T} - \{\phi \Rightarrow \psi\}) \cup \{\psi\},\ \mathcal{F})$
and Wang$(\mathcal{T} - \{\phi \Rightarrow \psi\},\ \mathcal{F} \cup \{\phi\})$;

10. If $(\phi \Rightarrow \psi) \in \mathcal{F}$, return Wang$(\mathcal{T} \cup \{\phi\},\ (\mathcal{F} - \{\phi \Rightarrow \psi\}) \cup \{\psi\})$;

Wang: The Algorithm

11. If $(\phi \Leftrightarrow \psi) \in \mathcal{T}$,
return Wang$((\mathcal{T} - \{\phi \Leftrightarrow \psi\}) \cup \{\psi, \phi\},\ \mathcal{F})$
and Wang$(\mathcal{T} - \{\phi \Leftrightarrow \psi\},\ \mathcal{F} \cup \{\phi, \psi\})$;

12. If $(\phi \Leftrightarrow \psi) \in \mathcal{F}$,
return Wang$(\mathcal{T} \cup \{\phi\},\ (\mathcal{F} - \{\phi \Leftrightarrow \psi\}) \cup \{\psi\})$
and Wang$(\mathcal{T} \cup \{\psi\},\ (\mathcal{F} - \{\phi \Leftrightarrow \psi\}) \cup \{\phi\})$;

Examples

- Using Wang's algorithm, determine whether the following are valid arguments.
  - $\models ((P \Rightarrow Q) \Rightarrow (\neg Q \Rightarrow \neg P))$.
  - $(P \Rightarrow Q) \models (\neg P \Rightarrow \neg Q)$.
- **Do it yourself.**
- The tree structure resulting from applying Wang's algorithm is called a **semantic tableau**.

The Light Switch World

- Recall the single-switch light switch world.
- Let $\mathbb{K} = \{SD \Leftrightarrow \neg SU, LF \Leftrightarrow \neg LN, SU \Leftrightarrow LN\}$ be the set of domain axioms.
- Prove that $\mathbb{K} \cup \{LN\} \models \neg SD$.
- Note that, if we think of $\mathbb{K}$ as a knowledge base, then the above is equivalent to the following sequence:
  1. Tell(LN)
  2. Ask($\neg SD$)
- (If you do not want to permanently add LN to the KB, then you should Ask($LN \Rightarrow \neg SD$).)

Some Important Properties

- Wang's algorithm is **sound**.
  - If Wang($\mathcal{P}, \{\phi\}$) = "True", then $\mathcal{P} \models \phi$.
- Wang's algorithm is **complete**.
  - If $\mathcal{P} \models \phi$, then Wang($\mathcal{P}, \{\phi\}$) = "True".

Syntactic Inference

- An **inference rule** is a rule that licences the **derivation** of WFFs of a certain form from a (possibly empty) set of WFFs of certain forms.
- **Syntactic inference** is the process of identifying correct derivations, based on some set of inference rules.
- The important point is that syntactic inference depends solely on the *form* of the WFFs—the syntax, not the semantics.
- For a set of WFFs $\mathcal{P}$, a WFF ϕ, and a set $\mathcal{I}$ of inference rules; $\mathcal{P} \vdash_{\mathcal{I}} \phi$ means that ϕ is derivable from $\mathcal{P}$ using the rules in $\mathcal{I}$.
- If $\mathcal{P} = \{\}$, we write $\vdash_{\mathcal{I}} \phi$, and ϕ is said to be a **theorem**.
- When clear from context, the subscript $\mathcal{I}$ will be omitted.

Natural Deduction

- In a **natural deduction** syntactic inference system we typically have a large set of rules of inference:
  - Two rules for each connective: an **introduction rule** and an **elimination rule**.
- Inference rules are typically represented as

$$\frac{\Gamma_1, \Gamma_2, \ldots, \Gamma_n}{\phi}$$

where

  - ϕ is a PL WFF; and
  - Γ_i is either
    1. a WFF or
    2. $\Delta \vdash \psi$, for a set of WFFs Δ and a WFF ψ.

Interpretation of Inference Rules

- You can interpret the above inference-rule schema as follows:
  - Assume some set of WFFs (KB) $\mathbb{K}$.
  - Add ϕ to $\mathbb{K}$ if
    1. all Γ_is that are WFFs are in $\mathbb{K}$, and
    2. for all Γ_is of the form $\Delta \vdash \psi$, indeed $\Delta \vdash \psi$.
- Note that, a derivation may, thus, make use of a **sub-derivation**.

$\wedge$-Rules

- $\wedge$-Introduction:

$$\frac{\phi, \psi}{\phi \wedge \psi}$$

- $\wedge$-Elimination: (two rules)

$$\frac{\phi \wedge \psi}{\phi \text{ (or } \psi)}$$

$\vee$-Rules

- $\vee$-Introduction:

$$\frac{\phi}{\phi \vee \psi}$$

- $\vee$-Elimination: (two rules)

$$\frac{\phi \vee \psi, \neg\psi \text{ (or } \neg\phi)}{\phi \text{ (or } \psi)}$$

$\Leftrightarrow$-Rules

- $\Leftrightarrow$-Introduction:

$$\frac{\phi \Rightarrow \psi, \psi \Rightarrow \phi}{\phi \Leftrightarrow \psi}$$

- $\Leftrightarrow$-Elimination: (two rules)

$$\frac{\phi \Leftrightarrow \psi}{\phi \Rightarrow \psi \text{ (or } \psi \Rightarrow \phi)}$$

$\Rightarrow$-Rules

- $\Rightarrow$-Introduction:

$$\frac{\mathbb{K} \cup \{\phi\} \vdash \psi}{\phi \Rightarrow \psi}$$

- $\Rightarrow$-Elimination:

$$\frac{\phi \Rightarrow \psi, \phi}{\psi}$$

¬-Rules

- ¬-Introduction:

$$\frac{\mathbb{K} \cup \{\phi\} \vdash \psi \wedge \neg\psi}{\neg\phi}$$

- ¬-Elimination:

$$\frac{\neg\neg\phi}{\phi}$$

T and F Rules

- **T**-Rule:

$$\frac{}{\mathbf{T}}$$

- **F**-Rules:

$$\frac{\mathbf{F}}{\phi}$$

$$\frac{\phi \wedge \neg\phi}{\mathbf{F}}$$

Proofs and Derivations

- A proof of $\mathcal{P} \vdash \phi$ is a proof by construction: construct a **derivation** of ϕ from $\mathcal{P}$.
- Such a derivation is a sequence of items ending with ϕ.
- Each item is either a WFF or a sub-derivation.
- Each WFF in the sequence is either in $\mathcal{P}$, a repetition of a WFF that appears earlier in the sequence, or follows from earlier WFFs and sub-derivations by one of the inference rules.
- If $\mathcal{P} = \{\}$, then the derivation is a **proof** of the theorem ϕ.

Example

Prove that $\{A, (B \Rightarrow \neg C), ((A \wedge B) \Rightarrow (D \vee C)), B\} \vdash D$

Example

Prove that $\{A, (B \Rightarrow \neg C), ((A \wedge B) \Rightarrow (D \vee C)), B\} \vdash D$

1.A (hypothesis)

Example

Prove that $\{A, (B \Rightarrow \neg C), ((A \wedge B) \Rightarrow (D \vee C)), B\} \vdash D$

1. A	(hypothesis)
2. $(B \Rightarrow \neg C)$	(hypothesis)

Example

Prove that $\{A, (B \Rightarrow \neg C), ((A \wedge B) \Rightarrow (D \vee C)), B\} \vdash D$

1.A	(hypothesis)
2.$(B \Rightarrow \neg C)$	(hypothesis)
3.$(A \wedge B) \Rightarrow (D \vee C)$	(hypothesis)

Example

Prove that $\{A, (B \Rightarrow \neg C), ((A \wedge B) \Rightarrow (D \vee C)), B\} \vdash D$

1.A	(hypothesis)
2.$(B \Rightarrow \neg C)$	(hypothesis)
3.$(A \wedge B) \Rightarrow (D \vee C)$	(hypothesis)
4.B	(hypothesis)

Example

Prove that $\{A, (B \Rightarrow \neg C), ((A \wedge B) \Rightarrow (D \vee C)), B\} \vdash D$

1.A	(hypothesis)
2.$(B \Rightarrow \neg C)$	(hypothesis)
3.$(A \wedge B) \Rightarrow (D \vee C)$	(hypothesis)
4.B	(hypothesis)
5.$\neg C$	(2, 4, $\Rightarrow$-Elim)

Example

Prove that $\{A, (B \Rightarrow \neg C), ((A \wedge B) \Rightarrow (D \vee C)), B\} \vdash D$

1.A	(hypothesis)
2.$(B \Rightarrow \neg C)$	(hypothesis)
3.$(A \wedge B) \Rightarrow (D \vee C)$	(hypothesis)
4.B	(hypothesis)
5.$\neg C$	(2, 4, $\Rightarrow$-Elim)
6.$A \wedge B$	(1, 4, $\wedge$-Intro)

Example

Prove that $\{A, (B \Rightarrow \neg C), ((A \wedge B) \Rightarrow (D \vee C)), B\} \vdash D$

1.A	(hypothesis)
2.$(B \Rightarrow \neg C)$	(hypothesis)
3.$(A \wedge B) \Rightarrow (D \vee C)$	(hypothesis)
4.B	(hypothesis)
5.$\neg C$	(2, 4, $\Rightarrow$-Elim)
6.$A \wedge B$	(1, 4, $\wedge$-Intro)
7.$D \vee C$	(3, 6, $\Rightarrow$-Elim)

Example

Prove that $\{A, (B \Rightarrow \neg C), ((A \wedge B) \Rightarrow (D \vee C)), B\} \vdash D$

1.A	(hypothesis)
2.$(B \Rightarrow \neg C)$	(hypothesis)
3.$(A \wedge B) \Rightarrow (D \vee C)$	(hypothesis)
4.B	(hypothesis)
5.$\neg C$	(2, 4, $\Rightarrow$-Elim)
6.$A \wedge B$	(1, 4, $\wedge$-Intro)
7.$D \vee C$	(3, 6, $\Rightarrow$-Elim)
8.D	(5, 7, $\vee$-Elim)

Example

Prove that $\vdash ((P \Rightarrow Q) \Rightarrow (\neg Q \Rightarrow \neg P))$.

Important Properties

- A logic is
  - **sound** iff $\vdash \phi$ implies $\models \phi$,
  - **complete** iff $\models \phi$ implies $\vdash \phi$, and
  - **consistent** iff $\vdash \phi$ implies $\not\vdash \neg\phi$.
- The PL we considered is both sound and complete.
- **Which is more important, soundness or completeness?**
- **Why is inconsistency dangerous?**