CS 477: Background and Propositional Logic

Sasa Misailovic

Based on previous slides by Elsa Gunter, which were based on earlier slides by Gul Agha, and Mahesh Viswanathan

University of Illinois at Urbana-Champaign

Propositional Logic

The Language of Propositional Logic

- Constants $\{\mathrm{T}, \mathrm{F}\}$
- Countable set AP of propositional variables (x,y,z), a.k.a. propositional atoms, a.k.a. atomic propositions
- logical connectives: $\wedge$ (and); $\vee$ (or); $\neg$ (not); $\Rightarrow$ (implies);$\Leftrightarrow$ (if and only if)

Propositional Logic (cont.)

The set of propositional formulae PROP is the inductive closure of the previous elements as follows:

- $\{T,F\} \subseteq PROP$
- $AP \subseteq PROP$
- if $A \in PROP$ then $(A) \in PROP$ and $\neg A \in PROP$
- if $A \in PROP$ and $B \in PROP$ then $(A \wedge B) \in PROP$, $(A \vee B) \in PROP$, $(A \Rightarrow B) \in PROP$, $(A \Leftrightarrow B) \in PROP$.
- Nothing else is in PROP

- *Informal definition; formal definition requires math foundations, set theory, fixed point theorem ...*

Propositional Logic

We can write it as a grammar too:

- C ::= T | F
- AP ::= x | y | z | …
- PROP ::= C | AP | (PROP) | ¬PROP
  | PROP ∧ PROP | PROP ∨ PROP
  | PROP ⇒ PROP | PROP ⇔ PROP

We can get various “sentences” in this language.

E.g. $x \wedge y$, $(x \wedge y) \Rightarrow (x \vee y)$, $x \vee \neg x \Leftrightarrow T$

But what is their meaning?

Toward Propositional Logic Semantics

Model for Propositional Logic has three parts

- Mathematical set of values used as meaning of propositions
- Interpretation function giving meaning to props built from logical connectives, via structural recursion
- Standard Model of Propositional Logic
  - **Boolean values** B = {true,false}
  - **a valuation** v : AP → B

Example valuation:

AP	B
x	true
y	false
z	true

Background read:
https://courses.engr.illinois.edu/cs498mv/fa2018/PropositionalLogic.pdf

Semantics of Propositional Logic

Standard interpretation I_v defined by structural induction on formulae:

- I_v (T) = true and I_v (F) = false
- If a ∈ AP then I_v (a) = v (a)
- For p ∈ PROP , if I_v (p) = true then I_v (¬p) = false, and if I_v (p) = false then I_v (¬p) = true
- For p,q ∈ PROP :
  - If I_v (p) = true and I_v (q) = true, then I_v (p∧q) = true, else I_v (p∧q) = false
  - If I_v (p) = true or I_v (q) = true, then I_v (p∨q) = true, else I_v (p∨q) = false
  - If I_v (q) = true or I_v (p) = false, then I_v (p⇒q) = true, else I_v (p⇒q) = false
  - If I_v (p) = I_v (q) then I_v (p⇔q) = true, else I_v (p⇔q) = false

Example

- I_v (T) = true and I_v (F) = false
- If a ∈ AP then I_v (a) = v (a)
- For p ∈ PROP , if I_v (p) = true then I_v (¬p) = false, and if I_v (p) = false then I_v (¬p) = true
- For p,q ∈ PROP :
  - If I_v (p) = true and I_v (q) = true, then I_v (p ∧q) = true, else I_v (p ∧q) = false
  - If I_v (p) = true or I_v (q) = true, then I_v (p ∨q) = true, else I_v (p ∨q) = false
  - If I_v (q) = true or I_v (p) = false, then I_v (p ⇒q) = true, else I_v (p ⇒q) = false
  - If I_v (p) = I_v (q) then I_v (p ⇔q) = true, else I_v (p ⇔q) = false

p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \Rightarrow q$	$p \Leftrightarrow q$
true	true					
true	false					
false	true					
false	false					

Example

- I_v (T) = true and I_v (F) = false
- If a ∈ AP then I_v (a) = v (a)
- For p ∈ PROP , if I_v (p) = true then I_v (¬p) = false, and if I_v (p) = false then I_v (¬p) = true
- For p,q ∈ PROP :
  - If I_v (p) = true and I_v (q) = true, then I_v (p ∧q) = true, else I_v (p ∧q) = false
  - If I_v (p) = true or I_v (q) = true, then I_v (p ∨q) = true, else I_v (p ∨q) = false
  - If I_v (q) = true or I_v (p) = false, then I_v (p ⇒q) = true, else I_v (p ⇒q) = false
  - If I_v (p) = I_v (q) then I_v (p ⇔q) = true, else I_v (p ⇔q) = false

p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \Rightarrow q$	$p \Leftrightarrow q$
true	true	false	true	true	true	true
true	false	false	false	true	false	false
false	true	true	false	true	true	false
false	false	true	false	false	true	true

Semantics of Propositional Logic

(B,I) is the standard model of proposition logic

- **Satisfaction relation** $\models$ **:** Given valuation v and proposition p ∈PROP, we write $\boxed{v \models p \text{ iff } I_v(p) = \text{true}}$ (the $\models$ symbol name is called "double turnstile")
  - More fully written as $B,I,v \models p$.
  - Can also write $(B,I,v,p) \in \models$
  - Say valuation v **satisfies** p, or v **models** p
  - Write $v \not\models p$ if $I_v(p) = \text{false}$
- p is **satisfiable** if there exists valuation v such that $v \models p$
- p is **valid**, a.k.a. a **tautology** if for every valuation v we have v $\models$ p
- p is **logically equivalent** to q, $p \equiv q$ if for every valuation, v , we have $v \models p \text{ iff } v \models q$. Claim: Logical equivalence is an equivalence relation

We can have other models of this logic, e.g. defined via sets

Example Tautology

$$A \Rightarrow ((A \Rightarrow B) \Rightarrow B)$$

A	B	$A \Rightarrow B$	$(A \Rightarrow B) \Rightarrow B$	$A \Rightarrow ((A \Rightarrow B) \Rightarrow B)$
true	true	true	true	true
true	false	false	true	true
false	true	true	true	true
false	false	true	false	true

Some Useful Logical Equivalences

$$\neg\neg A \equiv A \qquad \neg\mathbf{T} \equiv \mathbf{F} \qquad \neg\mathbf{F} \equiv \mathbf{T}$$

$$(A \vee A) \equiv A \qquad (A \vee B) \vee C \equiv A \vee (B \vee C)$$

$$(A \wedge A) \equiv A \qquad (A \wedge B) \wedge C \equiv A \wedge (B \wedge C)$$

$$A \vee B \equiv B \vee A \qquad \neg(A \vee B) \equiv (\neg A) \wedge (\neg B)$$

$$A \wedge B \equiv B \wedge A \qquad \neg(A \wedge B) \equiv (\neg A) \vee (\neg B)$$

$$(A \wedge \neg A) \equiv \mathbf{F} \qquad A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$$

$$(A \vee \neg A) \equiv \mathbf{T} \qquad (A \wedge B) \vee C \equiv (A \vee C) \wedge (B \vee C)$$

$$(\mathbf{T} \wedge A) \equiv A \qquad A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$$

$$(\mathbf{T} \vee A) \equiv \mathbf{T} \qquad (A \wedge B) \vee C \equiv (A \wedge C) \vee (B \wedge C)$$

$$(\mathbf{F} \wedge A) \equiv \mathbf{F} \qquad (\mathbf{F} \vee A) \equiv A$$

Normal Forms

Conjunctive normal form (CNF):

$$\bigwedge_{i \in I} \bigvee_{j \in J_i} AP_{i,j}$$

Disjunctive normal form (DNF):

$$\bigvee_{i \in I} \bigwedge_{j \in J_i} AP_{i,j}$$

where I and J are index sets

Question:
What is the computational complexity of finding a satisfying assignment of variables?

Notions of Logical Consequence

Semantic entailment: $\mathrm{v} \vDash \mathrm{p}$

- As we defined on the previous slides

Syntactic entailment: $\Gamma \vdash p$ is true iff there is a proof from the formulas in Γ to the formula p

- ***Deductive system***: a list of rules that express which formulas can legally follow which.
- ***Proof (aka derivation):*** a sequence of formulas that follow the rules of the deductive apparatus

Proofs in Propositional Logic

Constructing the full truth table is expensive!

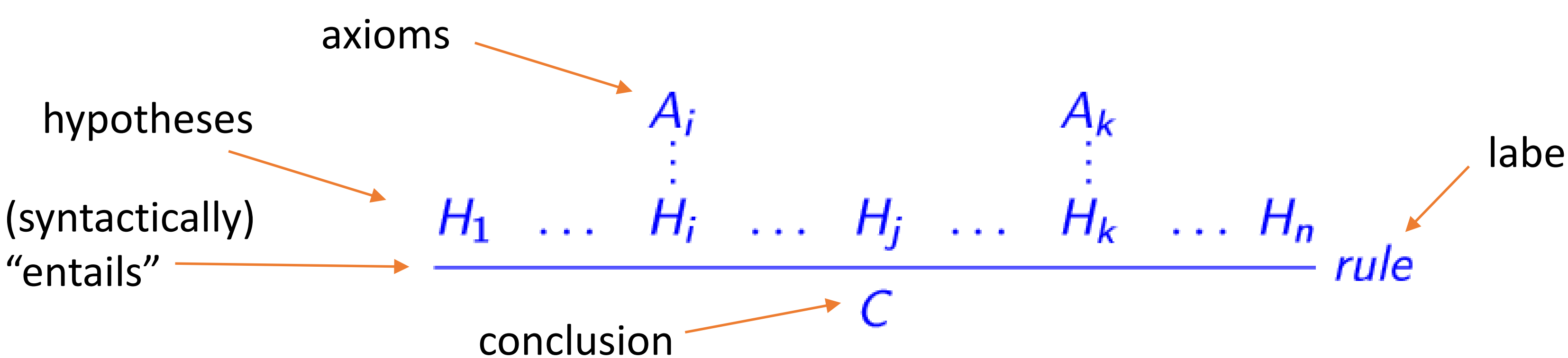
Natural Deduction proof is tree and a discharge function

- Nodes are instances of inference rules
- Leaves are assumptions of subproofs
- Discharge function maps each leaf of the tree to an ancestor as prescribed by the inference rules

See also: https://en.wikipedia.org/wiki/Propositional_calculus#Example_2._Natural_deduction_system

Proofs in Propositional Logic

- Inference rule has hypotheses and conclusion
- Conclusion **(C)** is a single proposition
- Hypotheses **(H) are** zero or more propositions, possibly with (discharged) hypotheses
- Rule with no hypotheses is called an axiom **(A)**
- Inference rule graphically presented as



Natural Deduction Inference Rules

Proof system: Inference rules associated with connectives
Two main kinds of inference rules in natural deduction:

- **Introduction:** says how to conclude proposition made from connective is true

$$\frac{\begin{array}{c}A\\ \vdots\\ B\end{array}}{A \Rightarrow B}\ \text{Imp I}$$

- **Eliminations:** says how to use a proposition made from connective to prove result

$$\frac{A \Rightarrow B \qquad A \qquad \begin{array}{c}B\\ \vdots\\ C\end{array}}{C}\ \text{Imp E}$$

Introduction Rules

Truth Introduction:

$$\frac{}{\mathbf{T}}\ \mathbf{T}\ \mathrm{I}$$

And Introduction:

$$\frac{A \quad B}{A \wedge B}\ \mathrm{And\ I}$$

Or Introduction:

$$\frac{A}{A \vee B}\ \mathrm{Or}_L\ \mathrm{I} \qquad \frac{B}{A \vee B}\ \mathrm{Or}_R\ \mathrm{I}$$

Not Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ \mathbf{F} \end{array}}{\neg A}\ \mathrm{Not\ I}$$

Implication Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ B \end{array}}{A \Rightarrow B}\ \mathrm{Imp\ I}$$

No False Introduction

Elimination Rules

False Elimination:

$$\frac{\mathbf{F}}{C}\ \mathbf{F}\ \mathrm{E}$$

Not Elimination:

$$\frac{\neg A \quad A}{C}\ \text{Not E}$$

And Elimination:

$$\frac{A \wedge B \qquad \begin{array}{c} A \\ \vdots \\ C \end{array}}{C}\ \text{And}_L\ \text{E}$$

$$\frac{A \wedge B \qquad \begin{array}{c} B \\ \vdots \\ C \end{array}}{C}\ \text{And}_R\ \text{E}$$

Or Elimination:

$$\frac{A \vee B \qquad \begin{array}{c} A \\ \vdots \\ C \end{array} \qquad \begin{array}{c} B \\ \vdots \\ C \end{array}}{C}\ \text{Or E}$$

Implication Elimination:

$$\frac{A \Rightarrow B \quad A \qquad \begin{array}{c} B \\ \vdots \\ C \end{array}}{C}\ \text{Imp E}$$

Desired Properties of a Proof System

- Soundness: if something is provable, it is valid
- Completeness: if something is valid, it is provable

Why: Conjunction? (slightly simplified)

Introduction:

$$\frac{A \quad B}{A \wedge B}\ \textsf{And I}$$

Elimination: *simplified

$$\frac{A \wedge B}{A}\ \textsf{And}_L\ \textsf{E}$$

$$\frac{A \wedge B}{B}\ \textsf{And}_R\ \textsf{E}$$

- **Local soundness:** if we introduce a conjunction and then eliminate it, we should be able to remove the whole subtree [local reduction $\Rightarrow_R$]
- **Local completeness:** we can eliminate the conjunction such that we can still reconstruct it by applying introduction [local expansion $\Rightarrow_E$]

*From https://www.cs.cmu.edu/~fp/courses/atp/handouts/ch2-natded.pdf for full description

Why: Conjunction? (slightly simplified)

Introduction:

$$\frac{A \quad B}{A \wedge B}\ \text{And I}$$

Elimination: *simplified

$$\frac{A \wedge B}{A}\ \text{And}_L\ \text{E}$$

$$\frac{A \wedge B}{B}\ \text{And}_R\ \text{E}$$

- **Local soundness:** if we introduce a conjunction and then eliminate it, we should be able to remove the whole subtree [local reduction $\Rightarrow_R$]
- **Local completeness:** we can eliminate the conjunction such that we can still reconstruct it by applying introduction [local expansion $\Rightarrow_E$]

*local soundness:

$$\cfrac{\cfrac{\stackrel{\mathcal{D}}{A\ true} \qquad \stackrel{\mathcal{E}}{B\ true}}{A \wedge B\ true}\wedge I}{A\ true}\wedge E_L \Longrightarrow_R \stackrel{\mathcal{D}}{A\ true} \qquad \cfrac{\cfrac{\stackrel{\mathcal{D}}{A\ true} \qquad \stackrel{\mathcal{E}}{B\ true}}{A \wedge B\ true}\wedge I}{B\ true}\wedge E_R \Longrightarrow_R \stackrel{\mathcal{E}}{B\ true}$$

*local completeness:

$$\stackrel{\mathcal{D}}{A \wedge B\ true} \Longrightarrow_E \cfrac{\cfrac{\stackrel{\mathcal{D}}{A \wedge B\ true}}{A\ true}\wedge E_L \qquad \cfrac{\stackrel{\mathcal{D}}{A \wedge B\ true}}{B\ true}\wedge E_R}{A \wedge B\ true}\wedge I$$

*From https://www.cs.cmu.edu/~fp/courses/atp/handouts/ch2-natded.pdf for full description

Why: Conjunction? (full rule)

Introduction:

$$\frac{A \quad B}{A \wedge B}\ \text{And I}$$

Elimination:

$$\frac{A \wedge B \qquad \begin{array}{c} A \\ \vdots \\ C \end{array}}{C}\ \text{And}_L\ \text{E}$$

$$\frac{A \wedge B \qquad \begin{array}{c} B \\ \vdots \\ C \end{array}}{C}\ \text{And}_R\ \text{E}$$

- **Local soundness:** if we introduce a conjunction and then eliminate it, we should be able to remove the whole subtree [local reduction $\Rightarrow_R$]
- **Local completeness:** we can eliminate the conjunction such that we can still reconstruct it by applying introduction [local expansion $\Rightarrow_E$]

(Your turn)

Introduction Rules

Truth Introduction:

$$\frac{}{\mathbf{T}}\ \mathbf{T}\ \mathsf{I}$$

And Introduction:

$$\frac{A \quad B}{A \wedge B}\ \text{And I}$$

Or Introduction:

$$\frac{A}{A \vee B}\ \text{Or}_L\ \text{I} \qquad \frac{B}{A \vee B}\ \text{Or}_R\ \text{I}$$

Not Introduction:

$$\begin{array}{c} A \\ \vdots \\ \mathbf{F} \\ \hline \neg A \end{array}\ \text{Not I}$$

Implication Introduction:

$$\begin{array}{c} A \\ \vdots \\ B \\ \hline A \Rightarrow B \end{array}\ \text{Imp I}$$

No False Introduction

Example 1

Truth Introduction:

$$\frac{}{\mathbf{T}}\ \mathbf{T}\ \mathrm{I}$$

And Introduction:

$$\frac{A \quad B}{A \wedge B}\ \text{And I}$$

Or Introduction:

$$\frac{A}{A \vee B}\ \mathrm{Or}_L\ \mathrm{I} \qquad \frac{B}{A \vee B}\ \mathrm{Or}_R\ \mathrm{I}$$

Not Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ \mathbf{F} \end{array}}{\neg A}\ \text{Not I}$$

Implication Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ B \end{array}}{A \Rightarrow B}\ \text{Imp I}$$

No False Introduction

$$\frac{}{A \Rightarrow (B \Rightarrow (A \wedge B))}$$

Truth Introduction:

$$\frac{}{\mathbf{T}}\,\mathbf{T}\ \mathrm{I}$$

And Introduction:

$$\frac{A \quad B}{A \wedge B}\ \mathrm{And\ I}$$

Or Introduction:

$$\frac{A}{A \vee B}\ \mathrm{Or}_L\ \mathrm{I} \qquad \frac{B}{A \vee B}\ \mathrm{Or}_R\ \mathrm{I}$$

Not Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ \mathbf{F} \end{array}}{\neg A}\ \mathrm{Not\ I}$$

Implication Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ B \end{array}}{A \Rightarrow B}\ \mathrm{Imp\ I}$$

No False Introduction

Example 1

$$\dfrac{\dfrac{\dfrac{A \qquad B}{A \wedge B}\ \mathrm{And\ I}}{B \Rightarrow (A \wedge B)}\ \mathrm{Imp\ I}}{A \Rightarrow (B \Rightarrow (A \wedge B))}\ \mathrm{Imp\ I}$$

Example 1

Truth Introduction:

$$\frac{}{\mathbf{T}}\ \mathbf{T}\ \mathrm{I}$$

And Introduction:

$$\frac{A \quad B}{A \wedge B}\ \text{And I}$$

Or Introduction:

$$\frac{A}{A \vee B}\ \text{Or}_L\ \text{I} \qquad \frac{B}{A \vee B}\ \text{Or}_R\ \text{I}$$

Not Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ \mathbf{F} \end{array}}{\neg A}\ \text{Not I}$$

Implication Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ B \end{array}}{A \Rightarrow B}\ \text{Imp I}$$

No False Introduction

$$\cfrac{\cfrac{\cfrac{\boxed{A} \quad \boxed{B}}{A \wedge B}\ \text{And I}}{\boxed{B} \Rightarrow (A \wedge B)}\ \text{Imp I}}{\boxed{A} \Rightarrow (B \Rightarrow (A \wedge B))}\ \text{Imp I}$$

All assumptions discharged; proof complete

Example 2

Truth Introduction:

$$\frac{}{\mathbf{T}}\ \mathbf{T}\ \mathsf{I}$$

And Introduction:

$$\frac{A \quad B}{A \wedge B}\ \mathsf{And\ I}$$

Or Introduction:

$$\frac{A}{A \vee B}\ \mathsf{Or}_L\ \mathsf{I} \qquad \frac{B}{A \vee B}\ \mathsf{Or}_R\ \mathsf{I}$$

Not Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ \mathbf{F} \end{array}}{\neg A}\ \mathsf{Not\ I}$$

Implication Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ B \end{array}}{A \Rightarrow B}\ \mathsf{Imp\ I}$$

No False Introduction

$$\frac{}{B \Rightarrow (A \wedge B)}$$

Example 2

Truth Introduction:

$$\frac{}{\mathbf{T}}\ \mathbf{T}\ \mathrm{I}$$

And Introduction:

$$\frac{A \quad B}{A \wedge B}\ \text{And I}$$

Or Introduction:

$$\frac{A}{A \vee B}\ \mathrm{Or}_L\ \mathrm{I} \qquad \frac{B}{A \vee B}\ \mathrm{Or}_R\ \mathrm{I}$$

Not Introduction:

$$\begin{array}{c} A \\ \vdots \\ \mathbf{F} \\ \hline \neg A \end{array}\ \text{Not I}$$

Implication Introduction:

$$\begin{array}{c} A \\ \vdots \\ B \\ \hline A \Rightarrow B \end{array}\ \text{Imp I}$$

No False Introduction

$$\dfrac{\dfrac{\boxed{?} \quad \boxed{B}}{A \wedge B}\ \text{And I}}{\boxed{B} \Rightarrow (A \wedge B)}\ \text{Imp I}$$

Truth Introduction:

$$\frac{}{\mathbf{T}}\ \mathbf{T}\ \mathrm{I}$$

And Introduction:

$$\frac{A \quad B}{A \wedge B}\ \mathrm{And\ I}$$

Or Introduction:

$$\frac{A}{A \vee B}\ \mathrm{Or}_L\ \mathrm{I} \qquad \frac{B}{A \vee B}\ \mathrm{Or}_R\ \mathrm{I}$$

Not Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ \mathbf{F} \end{array}}{\neg A}\ \mathrm{Not\ I}$$

Implication Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ B \end{array}}{A \Rightarrow B}\ \mathrm{Imp\ I}$$

No False Introduction

Example 2

$$\dfrac{\dfrac{A \qquad B}{A \wedge B}\ \mathrm{And\ I}}{B \Rightarrow (A \wedge B)}\ \mathrm{Imp\ I}$$

- Closed proofs must discharge all hypotheses
- Otherwise have theorem relative to undischarged hypotheses
- Here we have proved “Assuming A, we have B $\Rightarrow$(A $\wedge$ B)”

Example 3

$$\cfrac{\cfrac{A \qquad A}{A \wedge A}\ \text{And I}}{A \Rightarrow (A \wedge A)}\ \text{Imp I} \qquad\qquad \cfrac{\cfrac{A}{B \Rightarrow A}\ \text{Imp I}}{A \Rightarrow (B \Rightarrow A)}\ \text{Imp I}$$

- The rules may discharge multiple instances of hypotheses
- Or they may discharge none
- Each (implicit) assumption* can be discharged only once

* in the sense of the previous slide

Need for Elimination Rules

- So far, have rules to “introduce” logical connectives into propositions
- No rules for how to “use” logical connectives
  No assumptions with logical connectives
- Need “elimination” rules
  Example: Can’t prove $(A \Rightarrow B) \Rightarrow ((B \Rightarrow C) \Rightarrow (A \Rightarrow C))$
  with what we have so far
- Elimination rules assume assumption with a connective; have general conclusion
- Generally, needs additional hypotheses

Elimination Rules

False Elimination:

$$\frac{\mathbf{F}}{C}\ \mathbf{F}\ \mathrm{E}$$

Not Elimination:

$$\frac{\neg A \quad A}{C}\ \text{Not E}$$

And Elimination:

$$\frac{A \land B \qquad \begin{array}[b]{c} A \\ \vdots \\ C \end{array}}{C}\ \text{And}_L\ \text{E} \qquad\qquad \frac{A \land B \qquad \begin{array}[b]{c} B \\ \vdots \\ C \end{array}}{C}\ \text{And}_R\ \text{E}$$

Or Elimination:

$$\frac{A \lor B \qquad \begin{array}[b]{c} A \\ \vdots \\ C \end{array} \qquad \begin{array}[b]{c} B \\ \vdots \\ C \end{array}}{C}\ \text{Or E}$$

Implication Elimination:

$$\frac{A \Rightarrow B \quad A \quad \begin{array}[b]{c} B \\ \vdots \\ C \end{array}}{C}\ \text{Imp E}$$

Example with Elimination

And Introduction:

$$\frac{A \quad B}{A \wedge B}\ \text{And I}$$

Implication Introduction:

$$\frac{\begin{array}{c} A \\ \vdots \\ B \end{array}}{A \Rightarrow B}\ \text{Imp I}$$

And Elimination:

$$\frac{A \wedge B \quad \begin{array}{c} A \\ \vdots \\ C \end{array}}{C}\ \text{And}_L\ \text{E}$$

$$\frac{A \wedge B \quad \begin{array}{c} B \\ \vdots \\ C \end{array}}{C}\ \text{And}_R\ \text{E}$$

Implication Elimination:

$$\frac{A \Rightarrow B \quad A \quad \begin{array}{c} B \\ \vdots \\ C \end{array}}{C}\ \text{Imp E}$$

$$\frac{}{(A \Rightarrow B) \Rightarrow ((B \Rightarrow C) \Rightarrow (A \Rightarrow C))}$$

Example with Elimination

And Introduction:

$$\dfrac{A \quad B}{A \wedge B}\ \text{And I}$$

Implication Introduction:

$$\dfrac{\begin{array}{c} A \\ \vdots \\ B \end{array}}{A \Rightarrow B}\ \text{Imp I}$$

And Elimination:

$$\dfrac{A \wedge B \quad \begin{array}{c} A \\ \vdots \\ C \end{array}}{C}\ \text{And}_L\ \text{E}$$

$$\dfrac{A \wedge B \quad \begin{array}{c} B \\ \vdots \\ C \end{array}}{C}\ \text{And}_R\ \text{E}$$

Implication Elimination:

$$\dfrac{A \Rightarrow B \quad A \quad \begin{array}{c} B \\ \vdots \\ C \end{array}}{C}\ \text{Imp E}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{A \Rightarrow B \quad A \quad \dfrac{B \Rightarrow C \quad B \quad C}{C}\ \text{Imp E}}{C}\ \text{Imp E}}{A \Rightarrow C}\ \text{Imp I}}{(B \Rightarrow C) \Rightarrow (A \Rightarrow C)}\ \text{Imp I}}{(A \Rightarrow B) \Rightarrow ((B \Rightarrow C) \Rightarrow (A \Rightarrow C))}\ \text{Imp I}$$

Some Derived Rules

Modus Ponens

$$\frac{A \Rightarrow B \quad A}{B}\ \mathrm{MP} \qquad\qquad \frac{A \Rightarrow B \quad A \quad B}{B}\ \mathrm{Imp\ E}$$

Left Conjunct

$$\frac{A \wedge B}{A}\ \mathrm{AndL} \qquad\qquad \frac{A \wedge B \quad A}{A}\ \mathrm{And}_L\ \mathrm{E}$$

Right Conjunct

$$\frac{A \wedge B}{B}\ \mathrm{AndR} \qquad\qquad \frac{A \wedge B \quad B}{B}\ \mathrm{And}_R\ \mathrm{E}$$

Assumptions in Natural Deduction

Problem: Keeping track of hypotheses and their discharge in Natural Deduction is HARD!

- Solution: Use sequents to track hypotheses

A sequent is a pair of

- Γ: a set of propositions (called assumptions, or hypotheses of sequent) and
- **A**: a proposition (called conclusion of sequent)
- Notation:

$$\Gamma \vdash A$$

- Note: $\vdash$ expresses syntactic derivation ($\models$ was semantic)

Sequent Rules: Introduction

Γ is set of propositions (assumptions/hypotheses)

Hypothesis Introduction:

$$\frac{}{\Gamma \cup \{A\} \vdash A}\ \text{Hyp}$$

Truth Introduction:

$$\frac{}{\Gamma \vdash \mathbf{T}}\ \mathbf{T}\ \text{I}$$

And Introduction:

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ \text{And I}$$

Or Introduction:

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}\ \text{Or}_L\ \text{I} \qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}\ \text{Or}_R\ \text{I}$$

Not Introduction:

$$\frac{\Gamma \cup \{A\} \vdash \mathbf{F}}{\Gamma \vdash \neg A}\ \text{Not I}$$

Implication Introduction:

$$\frac{\Gamma \cup \{A\} \vdash B}{\Gamma \vdash A \Rightarrow B}\ \text{Imp I}$$

Sequent Rules: Elimination

Γ is set of propositions (assumptions/hypotheses)

Not Elimination:

$$\frac{\Gamma \vdash \neg A \quad \Gamma \vdash A}{\Gamma \vdash C} \text{ Not E}$$

Implication Elimination:

$$\frac{\Gamma \vdash A \Rightarrow B \quad \Gamma \vdash A \quad \Gamma \cup \{B\} \vdash C}{\Gamma \vdash C} \text{ Imp E}$$

And Elimination:

$$\frac{\Gamma \vdash A \wedge B \quad \Gamma \cup \{A\} \vdash C}{\Gamma \vdash C} \text{ And}_L \text{ E} \qquad \frac{\Gamma \vdash A \wedge B \quad \Gamma \cup \{B\} \vdash C}{\Gamma \vdash C} \text{ And}_R \text{ E}$$

False Elimination:

$$\frac{\Gamma \vdash \mathbf{F}}{\Gamma \vdash C} \mathbf{F} \text{ E}$$

Or Elimination:

$$\frac{\Gamma \vdash A \vee B \quad \Gamma \cup \{A\} \vdash C \quad \Gamma \cup \{B\} \vdash C}{\Gamma \vdash C} \text{ Or E}$$

Example Revisited

Hypothesis Introduction:

$$\frac{}{\Gamma \cup \{A\} \vdash A}\ \text{Hyp}$$

And Introduction:

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ \text{And I}$$

Implication Introduction:

$$\frac{\Gamma \cup \{A\} \vdash B}{\Gamma \vdash A \Rightarrow B}\ \text{Imp I}$$

And Elimination:

$$\frac{\Gamma \vdash A \wedge B \quad \Gamma \cup \{A\} \vdash C}{\Gamma \vdash C}\ \text{And}_L\ \text{E}$$

$$\frac{\Gamma \vdash A \wedge B \quad \Gamma \cup \{B\} \vdash C}{\Gamma \vdash C}\ \text{And}_R\ \text{E}$$

Implication Elimination:

$$\frac{\Gamma \vdash A \Rightarrow B \quad \Gamma \vdash A \quad \Gamma \cup \{B\} \vdash C}{\Gamma \vdash C}\ \text{Imp E}$$

$$\frac{}{\{\ \} \vdash (A \Rightarrow B) \Rightarrow ((B \Rightarrow C) \Rightarrow (A \Rightarrow C))}$$

Example Revisited

Hypothesis Introduction:

$$\frac{}{\Gamma \cup \{A\} \vdash A}\,\text{Hyp}$$

And Introduction:

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\,\text{And I}$$

Implication Introduction:

$$\frac{\Gamma \cup \{A\} \vdash B}{\Gamma \vdash A \Rightarrow B}\,\text{Imp I}$$

And Elimination:

$$\frac{\Gamma \vdash A \wedge B \quad \Gamma \cup \{A\} \vdash C}{\Gamma \vdash C}\,\text{And}_L\text{ E}$$

$$\frac{\Gamma \vdash A \wedge B \quad \Gamma \cup \{B\} \vdash C}{\Gamma \vdash C}\,\text{And}_R\text{ E}$$

Implication Elimination:

$$\frac{\Gamma \vdash A \Rightarrow B \quad \Gamma \vdash A \quad \Gamma \cup \{B\} \vdash C}{\Gamma \vdash C}\,\text{Imp E}$$

$$\Gamma_3 = \{A \Rightarrow B,\ B \Rightarrow C,\ A\}$$
$$\Gamma_4 = \{A \Rightarrow B,\ B \Rightarrow C,\ A,\ B\}$$
$$\Gamma_5 = \{A \Rightarrow B,\ B \Rightarrow C,\ A,\ B,\ C\}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\text{Hyp}}{\Gamma_3 \vdash A \Rightarrow B} \quad \dfrac{\text{Hyp}}{\Gamma_3 \vdash A} \quad \dfrac{\dfrac{\text{Hyp}}{\Gamma_4 \vdash B \Rightarrow C} \quad \dfrac{\text{Hyp}}{\Gamma_4 \vdash B} \quad \dfrac{\text{Hyp}}{\Gamma_5 \vdash C}}{\Gamma_4 \vdash C}\,\text{Imp E}}{\Gamma_3 = \{A \Rightarrow B,\ B \Rightarrow C,\ A\} \vdash C}\,\text{Imp E}}{\{A \Rightarrow B,\ B \Rightarrow C\} \vdash A \Rightarrow C}\,\text{Imp I}}{\{A \Rightarrow B\} \vdash (B \Rightarrow C) \Rightarrow (A \Rightarrow C)}\,\text{Imp I}}{\{\,\} \vdash (A \Rightarrow B) \Rightarrow ((B \Rightarrow C) \Rightarrow (A \Rightarrow C))}\,\text{Imp I}$$

Desired Properties of a Proof System

- Soundness**: if something is provable, it is valid**

Suppose $\{\mathrm{H}_1, \dots, \mathrm{H}_\mathrm{n}\} \vdash \mathrm{P}$ is provable. Then, for every valuation v , if for every i we have $\mathrm{v} \vDash \mathrm{H}_\mathrm{i}$, then $\mathrm{v} \vDash \mathrm{P}$.

True for natural deduction.

- Completeness: **if something is valid, it is provable**

For given rules, can not prove A ∨ ¬A. Need an axiom.

More: https://courses.grainger.illinois.edu/cs477/sp2020/lectures/04-prop-proof-soundness.pdf

Tool Support: Boolean Satisfiability

SAT Solver

- Takes a logical formula
- Returns a satisfying valuation (or unsat)
- Difficulty: Answering if P is satisfiable is NP-complete

Power-horses of many of today's analysis

- Finding the solution is NP-hard in general
- There are many good heuristics for common formulas arising from analysis of programs
- We will use Z3 (which can do much more) in our project assignments.

Basic Solution Algorithm

- DPLL (Davis–Putnam–Logemann–Loveland), 1961: many modern algorithms derive from it in some way
- Operates on the formula in CNF
- Core – Backtracking:
  - Recursively, choose a literal, set a truth value, check if the simplified formula was satisfied;
  - if not invert the truth value of the literal, and try again
- Aggressive simplification:
  - Unit clauses: contains only a single unassigned literal – it can be set in only one way to make the clause true!
  - Pure literal elimination: only x (or $\neg$x) occur in all clauses – assign it to make all clauses that contain it true

DPLL algorithm

```
Algorithm DPLL
    Input: A set of clauses Φ in CNF.
    Output: A Truth Value.

function DPLL(Φ)
    if Φ is a consistent set of literals then
        return true;
    if Φ contains an empty clause then
        return false;
    for every unit clause {l} in Φ do
        Φ ← unit-propagate(l, Φ);
    for every literal l that occurs pure in Φ do
        Φ ← pure-literal-assign(l, Φ);
    l ← choose-literal(Φ);
    return DPLL(Φ ∧ {l}) or DPLL(Φ ∧ {not(l)});
```

From Wikipedia: https://en.wikipedia.org/wiki/DPLL_algorithm

Some Practical Consequences

- Some classes of SAT formulas are easier to solve than others. Practical solvers apply many ‘tricks’ under the hood, e.g.,
    - Pick better order of variables (e.g., conflict driven resolution)
    - Estimate which version of algorithm will be better for the input formula, and run one algorithm from the “portfolio”
    - Employ machine learning

- We didn’t yet introduce quantifiers -> First-order logic (soon)
- We don’t yet know how to solve formulas that also should obey the rules of arithmetic and similar theories (e.g., uninterpreted functions, theory of arrays and others)
    - We will introduce SMT solving (later in the class)