# Semantics, Soundness, and Completeness for Propositional Natural Deduction

Robert Keller
January 2011

## Proof vs. Truth

- So far, we have seen a method (natural deduction) for **proof** of formulas.
- It would be nice if we had an **independent** definition of **truth** of those formulas so that we could ascertain whether
  - Our proofs are proving only *true* statements. (**soundness**)
  - There is nothing lacking in our proof system. (**completeness**)

## Giving Formulas a Meaning

- An **valuation** is a function $\nu$ (Greek "nu") that associates a value in {T, F} to every proposition symbol, with the requirement that, for the special symbols T and $\perp$ ("top" and "bottom"):
  - $\nu(\top) = T$
  - $\nu(\perp) = F$
- In the range of $\nu$, T is intended to represent "true" and F "false".
- A valuation is variously called an *assignment*, *interpretation*, or (in CS) *environment* (depending on the author or text).

## Induced Values for Formulas

- An **valuation** $\nu$ ***induces*** a value $\nu(\varphi)$ in {T, F} in any formula $\varphi$, inductively as follows:

| **Formula** $\varphi$ | **Value** $\nu(\varphi)$ | |
|---|---|---|
| single proposition symbol p | $\nu(p)$ as given by the valuation $\nu$ | |
| $\neg F$ | $h_\neg(\nu(F))$ | where $h_\neg(F) = T$ and $h_\neg(T) = F$ |
| $F \wedge G$ | $h_\wedge(\nu(F), \nu(G))$ | where $h_\wedge(x, y) = T$ iff $x = T$ **and** $y = T$ |
| $F \vee G$ | $h_\vee(\nu(F), \nu(G))$ | where $h_\vee(x, y) = T$ iff $x = T$ **or** $y = T$ |
| $F \rightarrow G$ | $h_\rightarrow(\nu(F), \nu(G))$ | where $h_\rightarrow(x, y) = T$ iff $x = F$ or $y = T$ |
| $F \leftrightarrow G$ | $h_\leftrightarrow(\nu(F), \nu(G))$ | where $h_\leftrightarrow(x, y) = T$ iff $x = y$ |

## Truth Function Summary

| x | y | $h_\neg(y)$ | $h_\wedge(x, y)$ | $h_\vee(x, y)$ | $h_\rightarrow(x, y)$ | $h_\leftrightarrow(x, y)$ |
|---|---|---|---|---|---|---|
| F | F | T | F | F | T | T |
| F | T | F | F | T | T | F |
| T | F | | F | T | F | F |
| T | T | | T | T | T | T |

## Example of Induced Value

- Formula: $p \vee q \rightarrow \neg p \wedge q$
- Valuation: $\nu(p) = F, \nu(q) = T, \ldots$
- Induced Value: $\nu(p \vee q \rightarrow \neg p \wedge q)$

$$\begin{aligned}
&= h_\rightarrow(\nu(p \vee q), && \nu(\neg p \wedge q)) \\
&= h_\rightarrow(h_\vee(\nu(p), \nu(q)), && h_\wedge(h_\neg(\nu(p)), \nu(q)) \\
&= h_\rightarrow(h_\vee(F, T), && h_\wedge(h_\neg(F), T) \\
&= h_\rightarrow(T, && h_\wedge(T, T)) \\
&= h_\rightarrow(T, T) \\
&= T
\end{aligned}$$

## Another Example of Induced Value

- Formula: $p \vee q \rightarrow \neg p \wedge q$
- Valuation: $\nu(p) = T$, $\nu(q) = F$, …
- Induced Value: $\nu(p \vee q \rightarrow \neg p \wedge q)$

$$
\begin{aligned}
&= h_{\rightarrow}(\nu(p \vee q), && \nu(\neg p \wedge q)) \\
&= h_{\rightarrow}(h_{\vee}(\nu(p), \nu(q)), && h_{\wedge}(h_{\neg}(\nu(p)), \nu(q)) \\
&= h_{\rightarrow}(h_{\vee}(T, F), && h_{\wedge}(h_{\neg}(T), F) \\
&= h_{\rightarrow}(T, && h_{\wedge}(F, F)) \\
&= h_{\rightarrow}(T, F) \\
&= F
\end{aligned}
$$

## Language Interpreter

- Note that the determination of the induced value of an expression given a valuation is essentially defining an **interpreter** for the language.
- The valuation would typically be called an "environment" in that context.

## Satisfaction Definition

- A valuation $\nu$ **satisfies** a formula $\varphi$ iff the induced value $\nu(\varphi) = T$.
- A formula is **satisfiable** iff there is some valuation that satisfies it. Otherwise it is **unsatisfiable**.
- Examples:
  - $p \rightarrow \neg p$ is satisfiable (for what valuation?)
  - $p \wedge \neg p$ is unsatisfiable

## Semantic Entailment: Double Turnstile

- Let $\varphi_1, \ldots, \varphi_n, \psi$ be formulas.
- The meaning of

  $$\varphi_1, \ldots, \varphi_n \models \psi$$

  is:

  For every valuation $\nu$ such that

  $\nu$ satisfies each of $\varphi_1, \ldots, \varphi_n$, (*)

  $\nu$ also satisfies $\psi$. (§)

## Example of Entailment $\models$

- Determine whether $p \vee q, \neg q \vee r \models p \vee r$
- We need to look at at most 8 valuations $\nu$, one for each possible value of $\nu(p)$, $\nu(q)$, $\nu(r)$.

| $\nu(p)$ | $\nu(q)$ | $\nu(r)$ | $\nu(p \vee q)$ | $\nu(\neg q \vee r)$ | * holds (LHS) | § holds (RHS) | $\nu(p \vee r)$ |
|---|---|---|---|---|---|---|---|
| F | F | F | F | T | | | F |
| F | F | T | F | T | | √ | T |
| F | T | F | T | F | | | F |
| F | T | T | T | T | √ | √ | T |
| T | F | F | T | T | √ | √ | T |
| T | F | T | T | T | √ | √ | T |
| T | T | F | T | F | | √ | T |
| T | T | T | T | T | √ | √ | T |

## Example of $\models$

- Determine whether or not $p \vee q, \neg q \vee r \models p \vee r$
- **Alternatively**, we could *reason* as follows:
  - $\nu(q) = F$ or $T$.
  - If $\nu(q) = F$, then * holds iff $\nu(p) = T$, and in that case $\nu(p \vee r) = T$, i.e. § holds.
  - If $\nu(q) = T$, then * holds iff $\nu(r) = T$, and in that case $\nu(p \vee r) = T$ , i.e. § holds.
  - Since § holds whenever * holds, we have entailment.

For every valuation $\nu$ such that

$\nu$ satisfies each of $\varphi_1, \ldots, \varphi_n$, (*)

$\nu$ also satisfies $\psi$. (§)

## Validity and Tautology

- $|= \psi$ is the special case for n = 0, and we say $\psi$ is **valid.**

  ***Every*** valuation must induce T for $\psi$, because every valuation vacuously induces T for every formula on the LHS.

  (For the propositional case, we can also say $\psi$ is a **tautology**. For the predicate logic case, not every valid formula is a tautology, although some are.)

- $\vdash \psi$ is the special case for n = 0, meaning that $\psi$ is **provable** from the **empty set** of premises.

## Satisfying a *Set* of Formulas

- Generally, $\Gamma$ is a (possibly-infinite) **set** of formulas
- A valuation $v$ **satisfies** $\Gamma$

  iff $v$ satisfies **each** formula in $\Gamma$.

## Validity vs. Provability

- Generally, $\Gamma$ is a (possibly-infinite) **set** of formulas
- The symbols $\vdash$ and $|=$ are part of the meta-language.
- $\Gamma \vdash \psi$ means $\psi$ is **provable** from formulas $\Gamma$
- $\Gamma |= \psi$ means: Every valuation that satisfies $\Gamma$ also satisfies $\psi$.

## Satisfiability of a *Set* of Formulas

- Set $\Gamma$ is **satisfiable** if there is a valuation that satisfies it.
- **Lemma S**: $\Gamma$ is **satisfiable** iff ***not*** ($\Gamma |= \bot$).
- **Proof** follows on next two slides.

## Satisfiability of a *Set* of Formulas

- **Proof**: The following statements are equivalent:
- $\Gamma$ is satisfiable.
- $\Gamma$ is **satisfied** by some $v$.
- $\Gamma$ is **satisfied** by some $v$ that does *not* satisfy $\bot$ (because no valuation satsifies $\bot$).
- not ($\Gamma |= \bot$).

## Soundness vs. Completeness of a Logical System (such as ND)

- **Soundness**: Every provable sequent is an entailment:

  (for every set $\Gamma$ and formula $\psi$):

  $\Gamma \vdash \psi$ implies $\Gamma |= \psi$

- **Completeness**: Every valid sequent is provable:

  (for every set $\Gamma$ and formula $\psi$):

  $\Gamma |= \psi$ implies $\Gamma \vdash \psi$

## Proof of Soundness

- **Soundness**: Every sequent of Natural Deduction is an entailment:

  (for every $\Gamma$, $\psi$):

  $\Gamma \vdash \psi$ implies $\Gamma \models \psi$

- Assume that $\Gamma \vdash \psi$, to show $\Gamma \models \psi$.
- This will be by **structural induction** on the proof tree of $\psi$ from formulas in $\Gamma$.

## Contextual Representation of Natural Deduction Rules

- In the representation of natural deduction rules, the **context** of premises is **implicit**.
- For example, with $\wedge$ Introduction, premises that lead to $\varphi$ and $\psi$ in the proof are not shown explicitly.

$$\frac{\varphi \quad \psi}{\varphi\wedge\psi}\ \wedge I$$

- For the soundness proof, however, it will be helpful to show the premises explicitly.
- So we **restate** this rule **with contexts** (sets of formulas $\Gamma$, $\Delta$ etc.) as follows:

$$\frac{\Gamma \vdash \varphi \qquad \Delta \vdash \psi}{\Gamma \cup \Delta \vdash \varphi\wedge\psi}\ \wedge I$$

## Contextual Representation of Natural Deduction Rules

- The contextual form will have its advantages when temporary assumptions are involved, such as in the $\to I$ rule:

$$\frac{\Gamma \cup \{\varphi\} \vdash \psi}{\Gamma \vdash \varphi\to\psi}\ \to I$$

| Natural Deduction Rules in Contextual Form | | |
|---|---|---|
| | **Introduction** | **Elimination** |
| $\wedge$ | $\dfrac{\Gamma \vdash \varphi \qquad \Delta \vdash \psi}{\Gamma\cup\Delta \vdash \varphi\wedge\psi}$ | $\dfrac{\Gamma \vdash \varphi\wedge\psi}{\Gamma \vdash \varphi}$ $\quad \dfrac{\Gamma \vdash \varphi\wedge\psi}{\Gamma \vdash \psi}$ |
| $\vee$ | $\dfrac{\Gamma \vdash \varphi}{\Gamma \vdash \varphi\vee\psi}$ $\quad \dfrac{\Gamma \vdash \psi}{\Gamma \vdash \varphi\vee\psi}$ | $\dfrac{\Gamma \vdash \varphi\vee\psi \quad \Delta\cup\{\varphi\} \vdash \xi \quad \Omega\cup\{\psi\} \vdash \xi}{\Gamma\cup\Delta\cup\Omega \vdash \xi}$ |
| $\to$ | $\dfrac{\Gamma\cup\{\varphi\} \vdash \psi}{\Gamma \vdash \varphi\to\psi}$ | $\dfrac{\Gamma \vdash \varphi \qquad \Delta \vdash \varphi\to\psi}{\Gamma\cup\Delta \vdash \psi}$ |
| $\neg$ | $\dfrac{\Gamma\cup\{\varphi\} \vdash \bot}{\Gamma \vdash \neg\varphi}$ | $\dfrac{\Gamma \vdash \varphi \qquad \Delta \vdash \neg\varphi}{\Gamma\cup\Delta \vdash \bot}$ |
| RAA<br>⊥E | $\dfrac{\Gamma\cup\{\neg\varphi\} \vdash \bot}{\Gamma \vdash \varphi}$ | $\dfrac{}{\Gamma\cup\{\bot\} \vdash \varphi}$ |
| | $\dfrac{}{\Gamma \vdash T}$ $\quad \dfrac{}{\Gamma\cup\{\varphi\} \vdash \varphi}$ | |

Note: This form is related to Gentzen's "Sequent Calculus".

## Example of Box, Tree, and Contextual Forms for the same proof

**Box**

| | | |
|---|---|---|
| 1: | $(E\wedge F)\to G$ | premise |
| 2: | $E$ | assumption |
| 3: | $F$ | assumption |
| 4: | $E\wedge F$ | $\wedge$ intro 2,3 |
| 5: | $G$ | $\to$ elim 1,4 |
| 6: | $F\to G$ | $\to$ intro 3–5 |
| 7: | $E\to F\to G$ | $\to$ intro 2–6 |

**Tree**

$$\dfrac{\dfrac{\dfrac{\dfrac{[E]_1 \qquad [F]_2}{E\wedge F}\wedge I \qquad E\wedge F\to G}{G}\to E}{F\to G}\to I_2}{E\to(F\to G)}\to I_1$$

**Contextual**

(Note that leaves can be premises or assumptions. Discharge is implicit.)

$$\dfrac{}{\{E\} \vdash E} \qquad \dfrac{}{\{F\} \vdash F}$$

$$\{E, F\} \vdash E\wedge F \quad \wedge I \qquad\qquad \dfrac{}{\{E\wedge F\to G\} \vdash E\wedge F\to G}$$

| | | |
|---|---|---|
| $\{E, F, E\wedge F\to G\} \vdash G$ | $\to E$ | (note LHS union) |
| $\{E, E\wedge F\to G\} \vdash F\to G$ | $\to I$ | (note reduction in context) |
| $\{E\wedge F\to G\} \vdash E\to(F\to G)$ | $\to I$ | (note reduction in context) |

**Contextual Rule Applications** (justification is *below* the lines)

## Proof of Soundness

- We are proving: $\Gamma \vdash \psi$ implies $\Gamma \models \psi$.
- In effect, we will show by induction that in the nodes of **any** contextual tree formed by following the rules of inference, we can replace $\vdash$ with $\models$.
- For example:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\{E\} \vdash E} \quad \dfrac{}{\{F\} \vdash F}}{\{E, F\} \vdash E\wedge F}\wedge I \qquad \dfrac{}{\{E\wedge F\to G\} \vdash E\wedge F\to G}}{\{E, F, E\wedge F\to G\} \vdash G}\to E}{\{E, E\wedge F\to G\} \vdash F\to G}\to I}{\{E\wedge F\to G\} \vdash E\to(F\to G)}\to I$$

becomes

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\{E\} \models E} \quad \dfrac{}{\{F\} \models F}}{\{E, F\} \models E\wedge F} \qquad \dfrac{}{\{E\wedge F\to G\} \models E\wedge F\to G}}{\{E, F, E\wedge F\to G\} \models G}}{\{E, E\wedge F\to G\} \models F\to G}}{\{E\wedge F\to G\} \models E\to(F\to G)}$$

## Proof of Soundness

- We are proving: $\Gamma \vdash \psi$ implies $\Gamma \models \psi$, i.e.
  **if** there is a proof of $\psi$ from $\Gamma$,
  **then** for any valuation $v$ such that $v(\Gamma) = T$, also $v(\psi) = T$.
- **Structural induction on the tree of the proof in contextual form.**
- The tree is either a single node, or a node combining one or more sub-trees.
- **Basis**: The simplest proof is a tree of **one node**. From the table, it must then be one of these:
  - $\Gamma \vdash \top$
  - $\Gamma \cup \{\varphi\} \vdash \varphi$
  - $\Gamma \cup \{\bot\} \vdash \varphi$
- In the first case, $v(\top) = T$ for any $v$, thus $\Gamma \models \top$.
- In the second case, if $v$ satisfies $\Gamma \cup \{\varphi\}$ then $v(\varphi) = T$, thus $\Gamma \models \varphi$.
- In the third case, $v$ can't satisfy $\Gamma \cup \{\bot\}$, so $\Gamma \cup \{\bot\} \models \varphi$ vacuously.

## Proof of Soundness Continued

- **Induction Step: Adding a root combining one or more subtrees.**
- Suppose that all of the antecedents of a rule in contextual form satisfy the property. We need to show that the consequent satisfies the property as well.
- Example: $\wedge$ Introduction rule:

$$\frac{\Gamma \vdash \varphi \qquad \Delta \vdash \psi}{\Gamma \cup \Delta \vdash \varphi \wedge \psi}$$

- **The induction hypothesis** is that $\Gamma \vdash \varphi$ implies $\Gamma \models \varphi$, and $\Delta \vdash \psi$ implies $\Delta \models \psi$.
- We must show that $\Gamma \cup \Delta \vdash \varphi \wedge \psi$ implies $\Gamma \cup \Delta \models \varphi \wedge \psi$.
- Assume $\Gamma \cup \Delta \vdash \varphi \wedge \psi$, to show $\Gamma \cup \Delta \models \varphi \wedge \psi$.
- Suppose $v$ satisfies $\Gamma \cup \Delta$, to show $v(\varphi \wedge \psi) = T$.
  Then $v$ satisfies both $\Gamma$ and $\Delta$.
  From the induction hypotheses, $v(\varphi) = v(\psi) = T$.
  Thus from the truth table for $h_\wedge$, $v(\varphi \wedge \psi) = T$.

## Proof of Soundness Continued

- **Induction Step, Continued:**
- The steps for $\wedge$E, $\vee$I, $\rightarrow$E, $\neg$E, $\bot$E (rules that don't introduce assumptions) are analogous to that for $\wedge$I, and are left to the reader.

## Proof of Soundness Continued

- **Induction Step, Continued:**
- Example: $\rightarrow$ Introduction rule

$$\frac{\Gamma \cup \{\varphi\} \vdash \psi}{\Gamma \vdash \varphi \rightarrow \psi}$$

- (Here $\varphi$ is the assumption used in natural deduction, which is discharged at the end of the sub-proof.)
- **The induction hypothesis** is $\Gamma \cup \{\varphi\} \vdash \psi$, i.e. if $v$ satisfies $\Gamma \cup \{\varphi\}$ then $v(\psi) = T$.
- We must show that $\Gamma \vdash \varphi \rightarrow \psi$, i.e. if $v$ satisfies $\Gamma$, then $v(\varphi \rightarrow \psi) = T$.
- Suppose that $v$ satisfies $\Gamma$. There are two cases:
  - **If $v(\varphi) = T$**, then $v$ satisfies $\Gamma \cup \{\varphi\}$, and from the induction hypothesis, $v(\psi) = T$, so $v(\varphi \rightarrow \psi) = T$ from the truth table for $h_\rightarrow$.
  - **If $v(\varphi) = F$**, then $v(\varphi \rightarrow \psi) = T$, also from the truth table for $h_\rightarrow$.
- The step for $\vee$E (which also introduces assumptions) is analogous to the above.

## Proof of Soundness Continued

- **Induction Step:**
- RAA

$$\frac{\Gamma \cup \{\neg \varphi\} \vdash \bot}{\Gamma \vdash \varphi}$$

- **The induction hypothesis** is that $v$ satisfies $\Gamma \cup \{\neg \varphi\}$ implies $v(\bot) = T$.
- But $v(\bot) = F$ always, so $v$ **cannot** satisfy $\Gamma \cup \{\neg \varphi\}$.
- We must show that if $v$ satisfies $\Gamma$ then $v(\varphi) = T$.
  - Suppose that $v$ satisfies $\Gamma$.
  - Because $v$ **does not** satisfy $\Gamma \cup \{\neg \varphi\}$, we must have $v(\neg \varphi) = F$.
  - But then $v(\varphi) = T$, from the truth table for $h_\neg$.
- The step for $\neg$I is analogous to the above.
- This concludes the proof of the induction step, and **thus ND is sound**.

## Uses of Soundness

- There are **algorithms** for determining whether or not

$$\varphi_1, \ldots, \varphi_n \models \psi$$

- Thus, one can compute a **necessary** condition of whether there is a proof of

$$\varphi_1, \ldots, \varphi_n \vdash \psi$$

- In other words, before embarking on trying to find a proof of a formula, we could check whether the formula follows on semantic grounds first.

## Completeness

- Completeness says

  (for all $\Gamma$, $\psi$)

  $\Gamma \models \psi$ implies $\Gamma \vdash \psi$

- The general case (where $\Gamma$ could be infinite) will require a "non-constructive" proof.
- The case of $\Gamma$ **finite** is special, and admits a constructive, even algorithmic, proof.

## Finite Completeness

- Finite completeness says (for all $\varphi_1, \ldots, \varphi_n, \psi$)

  $\varphi_1, \ldots, \varphi_n \models \psi$

  implies

  $\varphi_1, \ldots, \varphi_n \vdash \psi$

- **If** this could be established, then the algorithm mentioned for soundness would be a necessary and **sufficient** condition for the existence of a proof. Thus provability could be testable algorithmically.
- Our proof will use LEM, i.e. it applies to a classical rather than an intuitionistic system.

## Proof of Finite Completeness
(following Huth and Ryan)

Three steps are used to show
$\varphi_1, \ldots, \varphi_n \models \psi$ implies $\varphi_1, \ldots, \varphi_n \vdash \psi$ :

1. $\varphi_1, \ldots, \varphi_n \models \psi$ implies $\models (\varphi_1 \to (\varphi_2 \to \ldots (\varphi_n \to \psi)) \ldots)$
2. For any formula $\eta$, $\models \eta$ implies $\vdash \eta$.
   [$\eta$ could be $(\varphi_1 \to (\varphi_2 \to \ldots (\varphi_n \to \psi)) \ldots)$, for example.]
3. $\vdash (\varphi_1 \to (\varphi_2 \to \ldots (\varphi_n \to \psi)) \ldots)$ implies $\varphi_1, \ldots, \varphi_n \vdash \psi$

**Step 2 is the key one**, as only it bridges the gap between $\models$ and $\vdash$. The other two are simplifying steps, showing that we don't need to worry about the LHS of the turnstiles.

Steps 1 and 3 can be proved by induction on n. I leave them to you.

## Proof that for all $\eta$

$\models \eta$ implies $\vdash \eta$

- Assume $\models \eta$. Let $p_1, p_2, \ldots, p_k$ be the set of all proposition symbols that occur in $\eta$.
- For each combination of proposition symbols with and without negation, we show that there is a sequent with that combination on the left and the formula of interest on the right:
  - $p_1, p_2, \ldots, p_k \vdash \eta$
  - $\neg p_1, p_2, \ldots, p_k \vdash \eta$
  - $p_1, \neg p_2, \ldots, p_k \vdash \eta$
  - $\neg p_1, \neg p_2, \ldots, p_k \vdash \eta$ etc.
- Then those sequents will be combined into a single sequent of the required form using LEM and $\vee$E.

## The Combination Process

- Because this constructs a derivation that is of length exponential in k, we will show it by example, for k = 2.
- Given that we have:
  - $p_1, p_2 \vdash \eta$
  - $\neg p_1, p_2 \vdash \eta$
  - $p_1, \neg p_2 \vdash \eta$
  - $\neg p_1, \neg p_2 \vdash \eta$
- The proof constructed for the single sequent is shown on the next page.

## Proof Constructed for the Single Sequent

| | | |
|---|---|---|
| 1. | $p_1 \vee \neg p_1$ | LEM |
| 2. | $p_1$ | Assumption |
| 3. | $p_2 \vee \neg p_2$ | LEM |
| 4. | $p_2$ | Assumption |
| | . . . steps in the proof of $\mathbf{p_1, p_2} \vdash \eta$ | |
| 5. | $\eta$ | |
| 6. | $\neg p_2$ | Assumption |
| | . . . steps in the proof of $\mathbf{p_1, \neg p_2} \vdash \eta$ | |
| 7. | $\eta$ | |
| 8. | $\eta$ | $\vee$E 3, 4-5, 6-7 |
| 9. | $\neg p_1$ | Assumption |
| 10. | $p_2 \vee \neg p_2$ | LEM |
| 11. | $p_2$ | Assumption |
| | . . . steps in the proof of $\mathbf{\neg p_1, p_2} \vdash \eta$ | |
| 12. | $\eta$ | |
| 13. | $\neg p_2$ | Assumption |
| | . . . steps in the proof of $\mathbf{\neg p_1, \neg p_2} \vdash \eta$ | |
| 14. | $\eta$ | |
| 15. | $\eta$ | $\vee$E 10, 11-12, 13-14 |
| 16. | $\eta$ | $\vee$E 1, 2-8, 9-15 |

## Proofs for the Individual Sequents

We want to show that **for any formula** $\eta$,

**if** $\models \eta$

**then each** of the individual sequents below has a proof

- $p_1, \ p_2, \ldots, p_k \vdash \eta$
- $\neg p_1, \ p_2, \ldots, p_k \vdash \eta$
- $p_1, \neg p_2, \ldots, p_k \vdash \eta$
- $\neg p_1, \neg p_2, \ldots, p_k \vdash \eta$ etc. (for every combination of symbols and their negations)

where $p_1, p_2, \ldots, p_k$ are the proposition symbols in $\eta$.

**Approach**: Use **structural induction** on the **structure of the formula** $\eta$ (rather than on the proof tree as before).

## Structural Induction on Formulas

- We'd like to show ***something like***:
  if $\eta$ is constructed of sub-formulas, say $\varphi \circ \psi$ where $\circ$ is some connective,
  then for any combination $p^*_1, p^*_2, \ldots, p^*_k$ of negated and un-negated proposition symbols:
  if
  $p^*_1, p^*_2, \ldots, p^*_k \vdash \varphi$ and
  $p^*_1, p^*_2, \ldots, p^*_k \vdash \psi$ then
  $p^*_1, p^*_2, \ldots, p^*_k \vdash \varphi \circ \psi$.
- However this is not always true
  (for example in the case of $\circ$ being $\neg$).

## Note on Inductive Proofs in General

- In many cases in CS and Math, when faced with an inductive proof, we have to prove a **stronger** statement than the one we wish to "take away".
- This is because the induction hypothesis supplied by the take-away statement itself is **too weak** to enable the inductive conclusion to be drawn. We need a stronger inductive hypothesis, and it has to be able to deduce a stronger conclusion at the same time.
- The present situation is an example. Be on the lookout for others during your career.

## Proofs for the Individual Sequents

- A **revised statement** that will enable structural induction is:
- For any valuation $v$, let $p^*_1, p^*_2, \ldots, p^*_k$ be the proposition symbols or their negations (depending on $v$, e.g. $\neg p_1, p_2, \ldots, \neg p_k$) such that $\{p^*_1, p^*_2, \ldots, p^*_k\}$ is satisfied by $v$ (e.g. $v(p_1) = F$, $v(p_2) = T$, etc.)
- **Lemma:** For any formula $\eta$ and valuation $v$:

  A($\eta$): **If** $v(\eta)$ = **T**, then $p^*_1, p^*_2, \ldots, p^*_k \vdash \eta$. ← take-away

  B($\eta$): **If** $v(\eta)$ = **F**, then $p^*_1, p^*_2, \ldots, p^*_k \vdash (\neg\eta)$. ← strengthener

  ↑ note $\neg$

**Proving** A($\eta$): If $v(\eta)$ = **T** then $p^*_1, p^*_2, \ldots, p^*_k \vdash \eta$.

B($\eta$): If $v(\eta)$ = **F** then $p^*_1, p^*_2, \ldots, p^*_k \vdash (\neg\eta)$.

- This is done by **structural induction** on the **structure** of the **formula** $\eta$.
- **Basis**: If $\eta$ is a **single proposition symbol** p, then:
  - If $v(p)$ = **T**, then p* must be **p**, and we certainly have $p \vdash p$ (case A).
  - If $v(p)$ = **F** , then p* must be **¬p**, and we have $\neg p \vdash (\neg p)$ (case B).
- If $\eta$ is $\bot$, then $v(\bot)$ = **F** always, but also $\vdash \neg\bot$ (by ¬I) (case B).
- If $\eta$ is T, then $v(T)$ = **T** always, but also $\vdash$ T (by TI) (case A).

**Proving** A($\eta$): If $v(\eta)$ = **T** then $p^*_1, p^*_2, \ldots, p^*_k \vdash \eta$.

B($\eta$): If $v(\eta)$ = **F** then $p^*_1, p^*_2, \ldots, p^*_k \vdash (\neg\eta)$.

- **Induction Step**: We have to show that the inductive hypothesis A($\eta$)and B($\eta$) implies the conclusion for each possible operator: $\neg \ \wedge \ \vee \ \rightarrow$ forming $\eta$ at the top level.
- For example, if $\eta$ is $\varphi \vee \psi$, then we show:

  if A($\varphi$) and B($\varphi$), and A($\psi$) and B($\psi$),

  then also A($\varphi \vee \psi$) and B($\varphi \vee \psi$).

  Fortunately one of A or B is vacuously true for any conclusion $\eta$.

## Case where $\eta$ is of form $\neg\rho$ for some $\rho$ :

- Case 1: $v(\eta)$ = **T**
- Then $v(\rho)$ = **F**.
- By the induction hypothesis, $B(\rho)$ :

  $p^*_1, p^*_2, \ldots, p^*_k \vdash (\neg\rho)$,

  but that is the same as $A(\eta)$:

  $p^*_1, p^*_2, \ldots, p^*_k \vdash \eta$

## Case where $\eta$ is of form $\neg\rho$ for some $\rho$ :

- Case 2: $v(\eta)$ = **F**
- Then $v(\rho)$ = **T**.
- By the induction hypothesis, $A(\rho)$ :

  $p^*_1, p^*_2, \ldots, p^*_k \vdash \rho$.

  Using derived rule **¬¬I** to extend the proof, we have

  $p^*_1, p^*_2, \ldots, p^*_k \vdash \neg(\neg\rho)$

  Therefore $B(\eta)$:

  $p^*_1, p^*_2, \ldots, p^*_k \vdash \neg\eta$

## Case where $\eta$ is of form $\rho_1 \wedge \rho_2$ :

- We need to consider 4 cases:
  $v(\rho_1, \rho_2)$ = **FF**, **FT**, **TF**, and **TT**.
- If $v(\rho_1)$ = **F**: in which case $v(\rho_1 \wedge \rho_2)$ = **F:**
  By the induction hypothesis
  $p^*_1, p^*_2, \ldots, p^*_k \vdash (\neg\rho_1)$

  Using ND rules, we derive with a few more steps, a proof of
  $p^*_1, p^*_2, \ldots, p^*_k \vdash \neg(\rho_1 \wedge \rho_2)$

  This conforms to case B.
- A similar argument applies if $v(\rho_2)$ = **F**.
  So three of four cases have now been covered.

## Case where $\eta$ is of form $\rho_1 \wedge \rho_2$ :

- Now address the remaining case.
- If $v(\rho_1) = v(\rho_2)$ = **T,** we have by the induction hypothesis
  $p^*_1, p^*_2, \ldots, p^*_k \vdash \rho_1$
  $p^*_1, p^*_2, \ldots, p^*_k \vdash \rho_2$

  These proofs can be combined using **∧I** to get a proof of
  $\rho_1 \wedge \rho_2$ (case A).
- **The steps for the other operators (∨, →) are similar.**

## Algorithm-Based Proof

- The proof just outlined is sufficiently constructive that we can create an **algorithm** from it:
- Given a valid formula $\eta$, **generate** a natural deduction proof of $\eta$.
- In some sense, such an algorithmic proof is useful, in that it can be **live-tested** by computer for various examples, **unlike an ordinary proof**.

**Example of Algorithmically-Generated Proof by my prover.pro**

**DNE: ¬¬p→q**

```
?- testTautology(implies(not(not(p)), p)).

Proof for tautology: implies(not(not(p)), p):

| 1: or(p, not(p)) [lem]
| --------------------
| | 2: p [assumption(or-elim)]
| | 3: implies(not(not(p)), p) [implies-intro(2)]
| --------------------
| --------------------
| | 4: not(p) [assumption(or-elim)]
| | 5: not(not(not(p))) [not-not-intro(4)]
| | --------------------
| | | 6: not(not(p)) [assumption(implies-intro)]
| | | 7: bottom [not-elim(5, 6)]
| | | 8: p [bottom(7)]
| | --------------------
| | 9: implies(not(not(p)), p) [implies-intro(6-8)]
| --------------------
| 10: implies(not(not(p)), p) [or-elim(1, 2-3, 4-9)]
```

**Example of Algorithmically-Generated Proof by my prover.pro**

**Peirce's law: ((p→q)→p)→p**
(can be proved by a human using RAA rather than LEM in 12 steps)

```
Proof for tautology: implies(implies(implies(p, q), p), p):

| 1: or(p, not(p)) [lem]
| --------------------
| | 2: p [assumption(or-elim)]
| | 3: or(q, not(q)) [lem]
| | --------------------
| | | 4: q [assumption(or-elim)]
| | | 5: implies(implies(implies(p, q), p), p) [implies-intro(2)]
| | --------------------
| | --------------------
| | | 6: not(q) [assumption(or-elim)]
| | | 7: implies(implies(implies(p, q), p), p) [implies-intro(2)]
| | --------------------
| | 8: implies(implies(implies(p, q), p), p) [or-elim(3, 4-5, 6-7)]
| --------------------
| --------------------
| | 9: not(p) [assumption(or-elim)]
| | 10: or(q, not(q)) [lem]
| | --------------------
| | | 11: q [assumption(or-elim)]
| | | --------------------
| | | | 12: p [assumption(implies-intro)]
| | | | 13: bottom [not-elim(9, 12)]
| | | | 14: q [bottom(13)]
| | | --------------------
| | | 15: implies(p, q) [implies-intro(12-14)]
| | | --------------------
| | | | 16: implies(implies(p, q), p) [assumption(not-intro)]
| | | | 17: p [implies-elim(15, 16)]
| | | | 18: bottom [not-elim(9, 17)]
| | | --------------------
| | | 19: not(implies(implies(p, q), p)) [not-intro(16-18)]
| | | --------------------
| | | | 20: implies(implies(p, q), p) [assumption(implies-intro)]
| | | | 21: bottom [not-elim(19, 20)]
| | | | 22: p [bottom(21)]
| | | --------------------
| | | 23: implies(implies(implies(p, q), p), p) [implies-intro(20-22)]
| | --------------------
| | --------------------
| | | 24: not(q) [assumption(or-elim)]
| | | --------------------
| | | | 25: p [assumption(implies-intro)]
| | | | 26: bottom [not-elim(9, 25)]
| | | | 27: q [bottom(26)]
| | | --------------------
| | | 28: implies(p, q) [implies-intro(25-27)]
| | | --------------------
| | | | 29: implies(implies(p, q), p) [assumption(not-intro)]
| | | | 30: p [implies-elim(28, 29)]
| | | | 31: bottom [not-elim(9, 30)]
| | | --------------------
| | | 32: not(implies(implies(p, q), p)) [not-intro(29-31)]
| | | --------------------
| | | | 33: implies(implies(p, q), p) [assumption(implies-intro)]
| | | | 34: bottom [not-elim(32, 33)]
| | | | 35: p [bottom(34)]
| | | --------------------
| | | 36: implies(implies(implies(p, q), p), p) [implies-intro(33-35)]
| | --------------------
| | 37: implies(implies(implies(p, q), p), p) [or-elim(10, 11-23, 24-36)]
| --------------------
| 38: implies(implies(implies(p, q), p), p) [or-elim(1, 2-8, 9-37)]
```

## Sketch of Completeness for the *General* (not-necessarily finite) Propositional Case

**This section is advanced and may be skipped.**

- This sketch follows van Dalen, *Logic and Structure*.
- **Definition**: A set of formulas Γ is **consistent** provided

  not Γ |— ⊥.

- Note the parallel:
  - **Consistency** of Γ: Not Γ |— ⊥.
  - **Satisfiability** of Γ: Not Γ |= ⊥.

## Lemma A

- For any Γ, φ

  Γ |— φ iff Γ ∪ {¬φ} |— ⊥

- Proof:
  - Suppose Γ |— φ. Then also Γ ∪ {¬φ} |— φ. Trivially Γ ∪ {¬φ} |— ¬φ. So Γ ∪ {¬φ} |—⊥ by ¬E.
  - Suppose Γ ∪ {¬φ} |— ⊥. Then by RAA, Γ |— φ.

## Lemma B

- For any Γ, φ

  Γ |= φ iff Γ ∪ {¬φ} |=⊥.

- Proof: The following statements are equivalent:
  - Γ | = φ.
  - If v is a valuation satisfying Γ then satisfies φ.
  - If v is a valuation satisfying Γ then v doesn't satisfy ¬φ.
  - There is no valuation satisfying Γ ∪ {¬φ}.
  - Every valuation satisfying Γ ∪ {¬φ} satisfies ⊥ (vacuously).
  - Γ ∪ {¬φ} |=⊥.

## Lemma C

- The following are equivalent:
  a) Completeness.
  b) For all Γ, φ, Γ |= φ implies Γ |− φ.
  c) For all Γ, Γ |= ⊥ implies Γ |− ⊥.
  d) For all Γ, not Γ |− ⊥ implies not Γ |= ⊥.
  e) For all Γ, Γ is consistent implies Γ is satisfiable by some valuation ("Γ has a model").
- Proof:
  - (b) is a restatement of (a).
  - (c) iff (b) is by Lemmas A and B.
  - (d) is the contrapositive of (c).
  - (e) is a restatement of (d).

## General Completeness Theorem

- We have shown that completeness is equivalent to:
- (For all Γ)
  Γ consistent implies Γ satisfiable.
- "Every consistent set of formulas has a model."
- Sketch of the proof of the above statement:

  We start with a $\Gamma_0$ that is consistent, to eventually show there exists a valuation satisfying $\Gamma_0$, based on ND rules.

## Sketch, continued

- First extend $\Gamma_0$ to a **maximally consistent** set $\Gamma_{max}$:
  - **Enumerate** every possible propositional formula $\varphi_0, \varphi_1, \varphi_2, \ldots$ defining sets $\Gamma_0, \Gamma_1, \Gamma_2, \ldots$ as follows:
    - If $\Gamma_i \cup \{\varphi_i\}$ is consistent, $\Gamma_{i+1}$ is defined as $\Gamma_i \cup \{\varphi_i\}$.

      Otherwise $\Gamma_{i+1}$ is defined as $\Gamma_i$.
  - (The Axiom of Choice is being used here.)
  - The **limit** of this process is $\cup\{\Gamma_0, \Gamma_1, \Gamma_2, \ldots\} = \Gamma_{max}$.
- Then show that $\Gamma_{max}$ is consistent, and in fact, maximally consistent.

## Sketch, continued

- $\mathbf{\Gamma_{max}}$ is ***consistent***, because at no step is a formula added that would destroy its consistency.
- It is ***maximally*** consistent because it can be shown to be **closed under derivability**:
  If $\Gamma_{max} \vdash \varphi$, then in fact $\varphi \in \Gamma_{max}$ .
- We then show that any maximally consistent set has an valuation satisfying it. **Define such a valuation** $\nu$ as follows:
  - For each proposition symbol p, if $p \in \Gamma_{max}$ then $\nu(p) = T$, otherwise $\nu(p) = F$.
- Then argue that $\nu$ **satisfies** $\Gamma_{max}$ using closure under derivability (using a soundness-like argument).
- Finally, $\nu$ also satisfies $\Gamma_0$, since $\Gamma_0 \subseteq \Gamma_{max}$ . So $\Gamma_0$ is satisfiable.