Semantics, Soundness, and Completeness for Propositional Natural Deduction

Robert Keller
January 2013
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 \( \bot \) (“top” and “bottom”):
  - \( \nu(T) = T \)
  - \( \nu(\bot) = F \)

- 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:

<table>
<thead>
<tr>
<th><strong>Formula $\varphi$</strong></th>
<th><strong>Value $\nu(\varphi)$</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>single proposition symbol $p$</td>
<td>$\nu(p)$ as given by the valuation $\nu$</td>
</tr>
<tr>
<td>$\neg F$</td>
<td>$h_{\neg}(\nu(F))$</td>
</tr>
<tr>
<td>$F \land G$</td>
<td>$h_{\land}(\nu(F), \nu(G))$</td>
</tr>
<tr>
<td>$F \lor G$</td>
<td>$h_{\lor}(\nu(F), \nu(G))$</td>
</tr>
<tr>
<td>$F \rightarrow G$</td>
<td>$h_{\rightarrow}(\nu(F), \nu(G))$</td>
</tr>
<tr>
<td>$F \leftrightarrow G$</td>
<td>$h_{\leftrightarrow}(\nu(F), \nu(G))$</td>
</tr>
</tbody>
</table>
## Truth Function Summary

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>( h_\sim(y) )</th>
<th>( h_\wedge(x, y) )</th>
<th>( h_\vee(x, y) )</th>
<th>( h_\rightarrow(x, y) )</th>
<th>( h_\leftrightarrow(x, y) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>


Example of Induced Value

- Formula: \( p \lor q \rightarrow \neg p \land q \)

- Valuation: \( \nu(p) = F, \nu(q) = T, \ldots \)

- Induced Value: 
  \[
  \nu(p \lor q \rightarrow \neg p \land q) = h \rightarrow (\nu(p \lor q), \nu(\neg p \land q))
  = h \rightarrow (h_\lor(\nu(p), \nu(q)), h_\land(h_\neg(\nu(p)), \nu(q)))
  = h \rightarrow (h_\lor(F, T), h_\land(h_\neg(F), T))
  = h \rightarrow (T, h_\land(T, T))
  = h \rightarrow (T, T)
  = T
  \]
Another Example of Induced Value

- Formula: \( p \lor q \rightarrow \neg p \land q \)

- Valuation: \( \nu(p) = T, \nu(q) = F, \ldots \)

- Induced Value: \( \nu(p \lor q \rightarrow \neg p \land q) \)
  \[ = h_{\rightarrow}(\nu(p) \lor \nu(q), \nu(\neg p \land q)) \]
  \[ = h_{\rightarrow}(h_{\rightarrow}(\nu(p), \nu(q)), h_{\land}(h_{\rightarrow}(\nu(p)), \nu(q))) \]
  \[ = h_{\rightarrow}(h_{\rightarrow}(T, F), h_{\land}(h_{\rightarrow}(T), F)) \]
  \[ = h_{\rightarrow}(T, h_{\land}(F, F)) \]
  \[ = h_{\rightarrow}(T, F) \]
  \[ = F \]
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?)
  • \( p \land \neg p \) 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 \( p \lor q, \neg q \lor r \models p \lor r \)
- We need to look at at most 8 valuations \( \nu \), one for each possible value of \( \nu(p), \nu(q), \nu(r) \).

<table>
<thead>
<tr>
<th>( \nu(p) )</th>
<th>( \nu(q) )</th>
<th>( \nu(r) )</th>
<th>( \nu(p \lor q) )</th>
<th>( \nu(\neg q \lor r) )</th>
<th>* holds (LHS)</th>
<th>( \checkmark ) holds (RHS)</th>
<th>( \nu(p \lor r) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td></td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>( \checkmark )</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td></td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>( \checkmark )</td>
<td>( \checkmark )</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>( \checkmark )</td>
<td>( \checkmark )</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>( \checkmark )</td>
<td>( \checkmark )</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>( \checkmark )</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>( \checkmark )</td>
<td>( \checkmark )</td>
<td>T</td>
</tr>
</tbody>
</table>
Example of $|=\$

- Determine whether or not $p \lor q, \neg q \lor r |= p \lor r$

- **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 $\nu(p \lor r) = T$, i.e. $\S$ holds.
    - If $\nu(q) = T$, then * holds iff $\nu(r) = T$, and in that case $\nu(p \lor r) = T$, i.e. $\S$ holds.
  - Since $\S$ holds whenever * holds, we have entailment.

For every valuation $\nu$ such that

$\nu$ satisfies each of $\phi_1, ..., \phi_n, (*)$

$\nu$ also satisfies $\psi$. ($\S$)
Validity and Tautology

• $\models \psi$ 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) \textbf{set} of formulas

- A valuation \( \nu \) \textbf{satisfies} \( \Gamma \)

  iff \( \nu \) satisfies \textbf{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 \( \neg (\Gamma |\models \bot) \).

- **Proof** follows on next slide.

- **Corollary**: \( \Gamma \) is **unsatisfiable** iff \( \Gamma |\models \bot \).
Satisfiability of a Set of Formulas

- **Proof**: The following statements are equivalent:

  - $\Gamma$ is satisfiable.
  
  - $\Gamma$ is satisfied by some $\nu$.
  
  - $\Gamma$ is satisfied by some $\nu$ that does not satisfy $\bot$ (because no valuation satisfies $\bot$).
  
  - $\neg (\Gamma \models \bot)$. 
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 $\Gamma \models \psi$

- **Completeness**: Every entailment is provable:
  
  (for every set $\Gamma$ and formula $\psi$):
  
  $\Gamma \models \psi$ 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 $\Gamma$.

- In order to make this as clean as possible, we develop another way of representing proof trees, called **contextual representation**.
Contextual Representation of Natural Deduction Rules

- In the previous representation of natural deduction rules, the **full context** of antecedents is **implicit**.

- For example, with \( \land \) Introduction, formulas that lead to \( \varphi \) and \( \psi \) in the proof are not shown explicitly.

\[
\begin{array}{c}
\varphi \\
\psi \\
\hline
\varphi \land \psi
\end{array}
\quad \land \text{I}
\]

- For the soundness proof, however, it will be helpful to show the context of each formula explicitly.

- So we **restate** this rule **with context** (sets of formulas \( \Gamma, \Delta \)) as follows:

\[
\frac{
\Gamma \vdash \varphi \\
\Delta \vdash \psi
}{
\Gamma \cup \Delta \vdash \varphi \land \psi
\}
\quad \land \text{I}
\]
Advantage of Contextual Representation of Natural Deduction Rules

The contextual form will have advantages when temporary assumptions are involved, such as in the →I rule:

\[
\frac{\Gamma \cup \{\phi\} \vdash \psi}{\Gamma \vdash \phi \rightarrow \psi} \quad \rightarrow I
\]

This expresses the fact that, in implies-introduction proving \(\phi \rightarrow \psi\), the context for the proof of \(\psi\) inside the box consists of the context outside the box, augmented with the assumption \(\phi\).
### Natural Deduction Rules in Contextual Form

<table>
<thead>
<tr>
<th></th>
<th>Introduction</th>
<th>Elimination</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\land)</td>
<td>(\Gamma \vdash \phi \quad \Delta \vdash \psi) (\Gamma \cup \Delta \vdash \phi \land \psi)</td>
<td>(\Gamma \vdash \phi \land \psi) (\Gamma \vdash \phi \land \psi) (\Gamma \vdash \phi) (\Gamma \vdash \psi)</td>
</tr>
<tr>
<td>(\lor)</td>
<td>(\Gamma \vdash \phi) (\Gamma \vdash \psi) (\Gamma \vdash \phi \lor \psi)</td>
<td>(\Gamma \vdash \phi \lor \psi) (\Delta \cup {\phi} \vdash \xi) (\Omega \cup {\psi} \vdash \xi) (\Gamma \cup \Delta \cup \Omega \vdash \xi)</td>
</tr>
<tr>
<td>(\rightarrow)</td>
<td>(\Gamma \cup {\phi} \vdash \psi) (\Gamma \vdash \phi \rightarrow \psi)</td>
<td>(\Gamma \vdash \phi) (\Delta \vdash \phi \rightarrow \psi) (\Gamma \cup \Delta \vdash \psi)</td>
</tr>
<tr>
<td>(\neg)</td>
<td>(\Gamma \cup {\phi} \vdash \bot) (\Gamma \vdash \neg \phi)</td>
<td>(\Gamma \vdash \phi) (\Delta \vdash \neg \phi) (\Gamma \cup \Delta \vdash \bot)</td>
</tr>
<tr>
<td>RAA</td>
<td>(\Gamma \vdash {\neg \phi} \vdash \bot)</td>
<td></td>
</tr>
<tr>
<td>(\bot E)</td>
<td>(\Gamma \vdash \phi)</td>
<td>(\Gamma \cup {\bot} \vdash \phi)</td>
</tr>
<tr>
<td>Ax</td>
<td>(\Gamma \vdash \top) (\Gamma \cup {\phi} \vdash \phi)</td>
<td></td>
</tr>
</tbody>
</table>

Note: This form is related to Gentzen’s “Sequent Calculus”.  

---

22
Example of a Proof with Contexts
Build tree from the bottom up
Lines with nothing above are Ax

\[
\begin{align*}
\text{Rules} & \quad \text{\text{\{\neg F\} | - \neg F \quad \text{\{F\} | - F}} \\
\text{\{EvF\} | - EvF} & \quad \text{\{E\} | - E \quad \text{\{-F, F\} | - E}} \\
\text{\{EvF, \neg F\} | - E} & \quad \neg E \quad \bot E \quad \vee E
\end{align*}
\]

1: E ∨ F, ¬F premises
2: E assumption
3: F assumption
4: ⊥ ¬ elim 3,1.2
5: E contra (constructive) 4
6: E ∨ elim 1.1,2–2,3–5
Example of Box, Tree, and Contextual Forms for the same proof

**Box**

1: \((E \land F) \rightarrow G\) **premise**

2: \(E\) **assumption**

3: \(F\) **assumption**

4: \(E \land F\) **\(\land\) intro 2, 3**

5: \(G\) **\(\rightarrow\) elim 1, 4**

6: \(F \rightarrow G\) **\(\rightarrow\) intro 3–5**

7: \(E \rightarrow F \rightarrow G\) **\(\rightarrow\) intro 2–6**

**Tree**

\[
\begin{aligned}
\{E\} & \vdash E \\
\{F\} & \vdash F \\
\{E, F\} & \vdash E \land F \\
\{E \land F \rightarrow G, E, F\} & \vdash G \\
\{E \land F \rightarrow G, E\} & \vdash F \rightarrow G \\
\{E \land F \rightarrow G\} & \vdash E \rightarrow (F \rightarrow G)
\end{aligned}
\]

**Contextual**

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

**Contextual Rule Applications**

(justification is below the lines)

\[
\begin{aligned}
\{E\}_1 & \vdash \{F\}_2 \\
E \land F & \vdash E \land F \rightarrow G \\
G & \vdash F \rightarrow G \\
E \rightarrow (F \rightarrow G)
\end{aligned}
\]
Step by Step

1: \((E \land F) \rightarrow G\) premise

\[\cdots\]

2: \(E \rightarrow F \rightarrow G\)

\{E \land F \rightarrow G\} \models E \rightarrow (F \rightarrow G)
Step by Step

1: \((E \land F) \rightarrow G\) premise
2: \(E\) assumption
3: \(F \rightarrow G\)
4: \(E \rightarrow F \rightarrow G\) → intro 2-3

\[
\text{Box}
\]

\[
\text{Tree}
\]

\[
\begin{align*}
[E]_1 & \quad \text{E} \land \text{F} \rightarrow \text{G} \\
\text{F} \rightarrow \text{G} & \quad \text{E} \rightarrow \text{(F} \rightarrow \text{G}) \\
\text{E} \rightarrow \text{(F} \rightarrow \text{G}) & \quad \rightarrow \text{I}_1
\end{align*}
\]

\[
\text{Contextual}
\]

\[
\begin{align*}
\{E \land F \rightarrow G, E\} & \quad \vdash F \rightarrow G \\
\{E \land F \rightarrow G\} & \quad \vdash E \rightarrow (F \rightarrow G) \\
\{E \land F \rightarrow G\} & \quad \vdash E \rightarrow (F \rightarrow G) \quad \rightarrow \text{I}
\end{align*}
\]
Step by Step

1: \((E \land F) \rightarrow G\) premise

2: \([E]\)
   assumption

3: \([F]\)
   assumption

4: \(G\)

5: \(F \rightarrow G\) → intro 3–4

6: \(E \rightarrow F \rightarrow G\) → intro 2–5

\[
\{E \land F \rightarrow G, E, F\} \vdash G
\]
\[
\{E \land F \rightarrow G, E\} \vdash F \rightarrow G \quad \rightarrow I
\]
\[
\{E \land F \rightarrow G\} \vdash E \rightarrow (F \rightarrow G) \quad \rightarrow I
\]
Step by Step

Box

1: \((E \land F) \rightarrow G\) premise

2: \(E\) assumption

3: \(F\) assumption

4: \(E \land F\)

5: \(G\) → elim 1, 4

6: \(F \rightarrow G\) → intro 3–5

7: \(E \rightarrow F \rightarrow G\) → intro 2–6

Tree

\[
\begin{align*}
[E]_1 & \quad [F]_2 \\
E \land F & \quad E \land F \rightarrow G & \rightarrow E \\
G & \quad \rightarrow I_2 \\
F \rightarrow G & \quad \rightarrow I_1 \\
E \rightarrow (F \rightarrow G)&
\end{align*}
\]

Contextual

\[
\begin{align*}
\{E \land F \rightarrow G, E, F\} & \rightarrow E \\
\{E \land F \rightarrow G, E, F\} & \rightarrow G \rightarrow E \\
\{E \land F \rightarrow G, E\} & \rightarrow F \rightarrow G \rightarrow I \\
\{E \land F \rightarrow G\} & \rightarrow E \rightarrow (F \rightarrow G) \rightarrow I
\end{align*}
\]
Example of Box, Tree, and Contextual Forms for the same proof

**Box**

1: \((E \land F) \rightarrow G\) premise
2: \(E\) assumption
3: \(F\) assumption
4: \(E \land F\) \(\land\) intro 2,3
5: \(G\) \(\rightarrow\) elim 1,4
6: \(F \rightarrow G\) \(\rightarrow\) intro 3–5
7: \(E \rightarrow F \rightarrow G\) \(\rightarrow\) intro 2–6

**Tree**

\[
\begin{array}{c}
\{E\} \quad E \\
\{F\} \quad F \\
\{E, F\} \quad E \land F \\
\{E \land F \rightarrow G, E, F\} \quad G \\
\{E \land F \rightarrow G, E\} \quad F \rightarrow G \\
\{E \land F \rightarrow G\} \quad E \rightarrow (F \rightarrow G)
\end{array}
\]

\[
\begin{array}{c}
[E]_1 \\
[F]_2 \\
E \land F \\
E \land F \rightarrow G \\
G \\
F \rightarrow G \\
E \rightarrow (F \rightarrow G)
\end{array}
\]

\[
\begin{array}{c}
\land I \\
\rightarrow E \\
\rightarrow I_2 \\
\rightarrow I_1
\end{array}
\]

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

**Contextual**

\[
\begin{array}{c}
\{E\} \quad E \\
\{F\} \quad F \\
\{E, F\} \quad E \land F \\
\{E \land F \rightarrow G, E, F\} \quad G \\
\{E \land F \rightarrow G, E\} \quad F \rightarrow G \\
\{E \land F \rightarrow G\} \quad E \rightarrow (F \rightarrow G)
\end{array}
\]

\[
\begin{array}{c}
\land I \\
\rightarrow E \\
\rightarrow I \\
\rightarrow I
\end{array}
\]

(Contextual Rule Applications)

(justification is below the lines)

29
Proof of Soundness

• We are proving: \( \Gamma \vdash \psi \) implies \( \Gamma \models \psi \).

• We will show by **structural induction** that in the nodes of **any** contextual tree formed by following the rules of inference, we can replace \( \vdash \) with \( \models \).

• For example:

\[
\begin{align*}
\{E\} & \vdash E & \{F\} & \vdash F \\
\{E, F\} & \vdash E \land I & \{E \land F \rightarrow G\} & \vdash E \land F \rightarrow G \\
\{E \land F \rightarrow G, E, F\} & \vdash G & \rightarrow E \\
\{E \land F \rightarrow G, E\} & \vdash F \rightarrow G & \rightarrow I \\
\{E \land F \rightarrow G\} & \vdash E \rightarrow (F \rightarrow G) & \rightarrow I
\end{align*}
\]

becomes

\[
\begin{align*}
\{E\} & \models E & \{F\} & \models F \\
\{E, F\} & \models E \land F & \{E \land F \rightarrow G\} & \models E \land F \rightarrow G \\
\{E \land F \rightarrow G, E, F\} & \models G & \rightarrow E \\
\{E \land F \rightarrow G, E\} & \models F \rightarrow G & \rightarrow I \\
\{E \land F \rightarrow G\} & \models E \rightarrow (F \rightarrow G) & \rightarrow I
\end{align*}
\]
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 \( \nu \) such that \( \nu(\Gamma) = T \), also \( \nu(\psi) = T \).

• **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:
  • \( \Gamma \vdash T \)
  • \( \Gamma \cup \{\varphi\} \vdash \varphi \)
  • \( \Gamma \cup \{\bot\} \vdash \varphi \)

  • In the first case, \( \nu(T) = T \) for any \( \nu \), thus \( \Gamma \models T \).

  • In the second case, if \( \nu \) satisfies \( \Gamma \cup \{\varphi\} \) then \( \nu(\varphi) = T \), thus \( \Gamma \models \varphi \).

  • In the third case, \( \nu \) 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: \( \land \) Introduction rule:

\[
\frac{\Gamma \vdash \varphi \quad \Delta \vdash \psi}{\Gamma \cup \Delta \vdash \varphi \land \psi}
\]

- **The induction hypothesis** is that \( \Gamma \vdash \varphi \) implies \( \Gamma \models \varphi \), and \( \Delta \vdash \psi \) implies \( \Delta \models \psi \).

- We must show that \( \Gamma \cup \Delta \vdash \varphi \land \psi \) implies \( \Gamma \cup \Delta \models \varphi \land \psi \).

- Assume \( \Gamma \cup \Delta \vdash \varphi \land \psi \), to show \( \Gamma \cup \Delta \models \varphi \land \psi \).

- Suppose \( \nu \) satisfies \( \Gamma \cup \Delta \), to show \( \nu(\varphi \land \psi) = T \).

  Then \( \nu \) satisfies both \( \Gamma \) and \( \Delta \).

  From the induction hypotheses, \( \nu(\varphi) = \nu(\psi) = T \).

  Thus from the truth table for \( h_\land \), \( \nu(\varphi \land \psi) = T \).
Proof of Soundness Continued

• **Induction Step, Continued:**

• The steps for \( \land E, \lor I, \rightarrow E, \neg E, \bot E \) (rules that don’t introduce assumptions) are analogous to that for \( \land I \), and are left to the reader.
Proof of Soundness Continued

- **Induction Step, Continued:**

  - Example: → 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: if \( \Gamma \cup \{ \varphi \} \vdash \psi \) then \( \Gamma \cup \{ \varphi \} \models \psi \).

- We must show the conclusion: if \( \Gamma \models \varphi \rightarrow \psi \) then \( \Gamma \models \varphi \rightarrow \psi \).

- Suppose \( \Gamma \models \varphi \rightarrow \psi \), to show \( \Gamma \models \varphi \rightarrow \psi \).

- From →I, we know \( \Gamma \cup \{ \varphi \} \vdash \psi \). So by the induction hypothesis, \( \Gamma \cup \{ \varphi \} \models \psi \).

- Now suppose that \( \nu \) satisfies \( \Gamma \). There are two cases:
  - **If** \( \nu(\varphi) = T \), then \( \nu \) satisfies \( \Gamma \cup \{ \varphi \} \), and from the induction hypothesis, \( \nu(\psi) = T \), so \( \nu(\varphi \rightarrow \psi) = T \), from the truth table for \( \rightarrow \).
  - **If** \( \nu(\varphi) = F \), then \( \nu(\varphi \rightarrow \psi) = T \), also from the truth table for \( \rightarrow \).

- The step for \( \lor E \) (which also introduces assumptions) is analogous to the above.
Proof of Soundness Continued

• **Induction Step:**

• RAA

\[
\frac{\Gamma \cup \{\neg \varphi\} \models \bot}{\Gamma \models \varphi}
\]

• **The induction hypothesis** is that if \(\Gamma \cup \{\neg \varphi\} \models \bot\) then \(\Gamma \cup \{\neg \varphi\} \models \bot\).

• To be shown is \(\Gamma \models \varphi\) implies \(\Gamma \models \bot\).

• Suppose \(\Gamma \models \varphi\). Then by the \(\neg\)E rule, \(\Gamma \cup \{\neg \varphi\} \models \bot\), and by the induction hypothesis \(\Gamma \cup \{\neg \varphi\} \models \bot\).

• We must show that if \(\nu\) satisfies \(\Gamma\) then \(\nu(\varphi) = T\).

  • Suppose \(\nu\) satisfies \(\Gamma\). Then by lemma S, \(\nu\) **cannot** satisfy \(\Gamma \cup \{\neg \varphi\}\).
  
  • So we must have \(\nu(\neg \varphi) = F\).
  
  • But then \(\nu(\varphi) = T\), from the truth table for \(h_{\neg}\).

• The proof 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

\[ \Gamma \vdash \psi \Rightarrow \Gamma \models \psi \]

- 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

\[ \varphi_1, \ldots, \varphi_n \models \psi \text{ implies } \varphi_1, \ldots, \varphi_n \vdash \psi : \]

1. \[ \varphi_1, \ldots, \varphi_n \models \psi \text{ implies } \models (\varphi_1 \rightarrow (\varphi_2 \rightarrow \ldots (\varphi_n \rightarrow \psi ))) \ldots \]

2. For any formula \( \eta \), \( \models \eta \) implies \( \vdash \eta \).
   \( \eta \) could be \( (\varphi_1 \rightarrow (\varphi_2 \rightarrow \ldots (\varphi_n \rightarrow \psi ))) \ldots \), for example.

3. \( \vdash (\varphi_1 \rightarrow (\varphi_2 \rightarrow \ldots (\varphi_n \rightarrow \psi ))) \ldots \) 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 $\eta$

\[ |\eta| \text{ implies } \vdash \eta \]

• Assume $|\eta|$. 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 $\lor$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

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>$p_1 \lor \neg p_1$</td>
<td>LEM</td>
</tr>
<tr>
<td>2.</td>
<td>$p_1$</td>
<td>Assumption</td>
</tr>
<tr>
<td>3.</td>
<td>$p_2 \lor \neg p_2$</td>
<td>LEM</td>
</tr>
<tr>
<td>4.</td>
<td>$p_2$</td>
<td>Assumption</td>
</tr>
<tr>
<td></td>
<td>... steps in the proof of $p_1, p_2 \vdash \eta$</td>
<td></td>
</tr>
<tr>
<td>5.</td>
<td>$\eta$</td>
<td></td>
</tr>
<tr>
<td>6.</td>
<td>$\neg p_2$</td>
<td>Assumption</td>
</tr>
<tr>
<td></td>
<td>... steps in the proof of $p_1, \neg p_2 \vdash \eta$</td>
<td></td>
</tr>
<tr>
<td>7.</td>
<td>$\eta$</td>
<td></td>
</tr>
<tr>
<td>8.</td>
<td>$\eta$</td>
<td>$\lor E$ 3, 4-5, 6-7</td>
</tr>
<tr>
<td>9.</td>
<td>$\neg p_1$</td>
<td>Assumption</td>
</tr>
<tr>
<td>10.</td>
<td>$p_2 \lor \neg p_2$</td>
<td>LEM</td>
</tr>
<tr>
<td>11.</td>
<td>$p_2$</td>
<td>Assumption</td>
</tr>
<tr>
<td></td>
<td>... steps in the proof of $\neg p_1, p_2 \vdash \eta$</td>
<td></td>
</tr>
<tr>
<td>12.</td>
<td>$\eta$</td>
<td></td>
</tr>
<tr>
<td>13.</td>
<td>$\neg p_2$</td>
<td>Assumption</td>
</tr>
<tr>
<td></td>
<td>... steps in the proof of $\neg p_1, \neg p_2 \vdash \eta$</td>
<td></td>
</tr>
<tr>
<td>14.</td>
<td>$\eta$</td>
<td></td>
</tr>
<tr>
<td>15.</td>
<td>$\eta$</td>
<td>$\lor E$ 10, 11-12, 13-14</td>
</tr>
<tr>
<td>16.</td>
<td>$\eta$</td>
<td>$\lor E$ 1, 2-8, 9-15</td>
</tr>
</tbody>
</table>
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:

- $p_1, p_2, \ldots, p_k \models \eta$
- $\neg p_1, p_2, \ldots, p_k \models \eta$
- $p_1, \neg p_2, \ldots, p_k \models \eta$
- $\neg p_1, \neg p_2, \ldots, p_k \models \eta$
  
  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).
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

\[
p^*_{1}, p^*_{2}, \ldots, p^*_{k} \vdash \varphi \quad \text{and} \quad p^*_{1}, p^*_{2}, \ldots, p^*_{k} \vdash \psi
\]

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** $\nu$, let $p^*_1, p^*_2, \ldots, p^*_k$ be the proposition symbols or their negations (depending on $\nu$, e.g. $\neg p_1, p_2, \ldots, \neg p_k$) such that $\{p^*_1, p^*_2, \ldots, p^*_k\}$ is satisfied by $\nu$ (e.g. $\nu(p_1) = F, \nu(p_2) = T$, etc.)

- **Lemma**: For any formula $\eta$ and valuation $\nu$:

  $A(\eta)$: If $\nu(\eta) = T$, then $p^*_1, p^*_2, \ldots, p^*_k \models \eta$.  
  $B(\eta)$: If $\nu(\eta) = F$, then $p^*_1, p^*_2, \ldots, p^*_k \models (\neg \eta)$.
Proving

\( A(\eta) \): If \( \nu(\eta) = T \) then \( p^*_1, p^*_2, \ldots, p^*_k \models \eta \).

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

• This is done by **structural induction** on the **structure** of the **formula** \( \eta \).

• **Basis:** If \( \eta \) is a **single proposition symbol** \( p \), then:
  - If \( \nu(p) = T \), then \( p^* \) must be \( p \), and we certainly have \( p \models p \) (case A).
  - If \( \nu(p) = F \), then \( p^* \) must be \( \neg p \), and we have \( \neg p \models (\neg p) \) (case B).

• If \( \eta \) is \( T \), then \( \nu(T) = T \) always, but also \( \models T \) (by Ti) (case A).

• If \( \eta \) is \( \bot \), then \( \nu(\bot) = F \) always, but also \( \models \neg \bot \) (by \( \neg I \)) (case B).
**Proving**  

\( A(\eta) \): If \( \nu(\eta) = T \) then \( p^{*}_1, p^{*}_2, \ldots, p^{*}_k \models \eta \).

\( B(\eta) \): If \( \nu(\eta) = F \) then \( p^{*}_1, p^{*}_2, \ldots, p^{*}_k \models ( \neg \eta ) \).

- **Induction Step**: We have to show that the inductive hypothesis \( A(\eta) \) and \( B(\eta) \) implies the conclusion for each possible operator: \( \neg \land \lor \rightarrow \) forming \( \eta \) at the top level.

- For example, if \( \eta \) is \( \varphi \lor \psi \), then we show:

  if \( A(\varphi) \) and \( B(\varphi) \), and \( A(\psi) \) and \( B(\psi) \),
  then also \( A(\varphi \lor \psi) \) and \( B(\varphi \lor \psi) \).

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: $\nu(\eta) = T$
- Then $\nu(\rho) = F$.
- By the induction hypothesis, $B(\rho)$:
  
  \[ p^*_1, p^*_2, \ldots, p^*_k \models (\neg \rho), \]

  but that is the same as $A(\eta)$:
  
  \[ p^*_1, p^*_2, \ldots, p^*_k \models \eta \]
Case where $\eta$ is of form $\neg \rho$ for some $\rho$:

- Case 2: $\nu(\eta) = F$
- Then $\nu(\rho) = T$.
- By the induction hypothesis, $A(\rho)$:
  
  \[ p^{*}_{1}, p^{*}_{2}, \ldots, p^{*}_{k} \vdash \rho. \]

Using derived rule $\neg \neg I$ 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 $\rho_1 \land \rho_2$ :

- We need to consider 4 cases:
  $\nu(\rho_1, \rho_2) = FF, FT, TF$, and $TT$.

- If $\nu(\rho_1) = F$: in which case $\nu(\rho_1 \land \rho_2) = F$:
  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
  $p^*_1, p^*_2, \ldots, p^*_k \vdash \neg (\rho_1 \land \rho_2)$

  This conforms to case B.

- A similar argument applies if $\nu(\rho_2) = F$.
  So three of four cases have now been covered.
Case where $\eta$ is of form $\rho_1 \land \rho_2$:

- Now address the remaining case.

- If $\nu(\rho_1) = \nu(\rho_2) = T$, we have by the induction hypothesis
  
  $p^*_1, p^*_2, \ldots, p^*_k \models \rho_1$
  
  $p^*_1, p^*_2, \ldots, p^*_k \models \rho_2$

  These proofs can be combined using $\land I$ to get a proof of $\rho_1 \land \rho_2$ (case A).

- The steps for the other operators ($\lor$, $\rightarrow$) 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: \( \neg\neg p \rightarrow q \)

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

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

<table>
<thead>
<tr>
<th></th>
<th>1: or(p, not(p)) [lem]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2: p [assumption(or-el)]</td>
</tr>
<tr>
<td></td>
<td>3: implies(not(not(p)), p) [implies-intro(2)]</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4: not(p) [assumption(or-el)]</td>
</tr>
<tr>
<td></td>
<td>5: not(not(not(p))) [not-not-intro(4)]</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6: not(not(p)) [assumption(implies-intro)]</td>
</tr>
<tr>
<td></td>
<td>7: bottom [not-el(5, 6)]</td>
</tr>
<tr>
<td></td>
<td>8: p [bottom(7)]</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>9: implies(not(not(p)), p) [implies-intro(6-8)]</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10: implies(not(not(p)), p) [or-el(1, 2-3, 4-9)]</td>
</tr>
</tbody>
</table>
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)

```
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: \(\text{implies}(\text{implies}(\text{implies}(p, q), p), p)\):

1. or(p, not(p)) [lem]
2. p [assumption(or-elim)]
3. or(q, not(q)) [lem]
4. q [assumption(or-elim)]
5. \(\text{implies}(\text{implies}(\text{implies}(p, q), p), p)\) [implies-intro(2)]
6. not(q) [assumption(or-elim)]
7. \(\text{implies}(\text{implies}(\text{implies}(p, q), p), p)\) [implies-intro(2)]
8. \(\text{implies}(\text{implies}(\text{implies}(p, q), p), p)\) [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. \(\text{implies}(p, q)\) [implies-intro(12-14)]
16. \(\text{implies}(\text{implies}(p, q), p)\) [assumption(not-intro)]
17. p [implies-elim(15, 16)]
18. bottom [not-elim(9, 17)]
19. not(\(\text{implies}(\text{implies}(p, q), p)\)) [not-intro(16-18)]
```
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 $\Gamma$ is **consistent** provided

\[
\text{not } \Gamma \models \bot.
\]

• Note the parallel:

  • **Consistency** of $\Gamma$: Not $\Gamma \models \bot$.

  • **Satisfiability** of $\Gamma$: Not $\Gamma \models \bot$. 
Lemma A

- For any $\Gamma$, $\varphi$

\[ \Gamma \vdash \varphi \quad \text{iff} \quad \Gamma \cup \{\neg \varphi\} \vdash \bot \]

- Proof:
  - Suppose $\Gamma \vdash \varphi$. Then also $\Gamma \cup \{\neg \varphi\} \vdash \varphi$. Trivially $\Gamma \cup \{\neg \varphi\} \vdash \neg \varphi$. So $\Gamma \cup \{\neg \varphi\} \vdash \bot$ by $\neg$E.
  
  - Suppose $\Gamma \cup \{\neg \varphi\} \vdash \bot$. Then by RAA, $\Gamma \vdash \varphi$. 


Lemma B

- For any $\Gamma$, $\varphi$
  \[ \Gamma \models \varphi \iff \Gamma \cup \{\neg \varphi\} \models \bot. \]

- Proof: The following statements are equivalent:
  - $\Gamma \models \varphi$.
  - If $\nu$ is a valuation satisfying $\Gamma$ then satisfies $\varphi$.
  - If $\nu$ is a valuation satisfying $\Gamma$ then $\nu$ doesn’t satisfy $\neg \varphi$.
  - There is no valuation satisfying $\Gamma \cup \{\neg \varphi\}$.
  - Every valuation satisfying $\Gamma \cup \{\neg \varphi\}$ satisfies $\bot$ (vacuously).
  - $\Gamma \cup \{\neg \varphi\} \models \bot$. 
Lemma C

• The following are equivalent:
  a) Completeness.
  b) For all $\Gamma$, $\varphi$, $\Gamma |- \varphi$ implies $\Gamma |- \neg \varphi$.
  c) For all $\Gamma$, $\Gamma |- \bot$ implies $\Gamma |- \bot$.
  d) For all $\Gamma$, not $\Gamma |- \bot$ implies not $\Gamma |- \bot$.
  e) For all $\Gamma$, $\Gamma$ is consistent implies $\Gamma$ is satisfiable by some valuation ("$\Gamma$ 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 $\Gamma$)
    $\Gamma$ consistent implies $\Gamma$ 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 $\Gamma_{\text{max}}$:
  - **Enumerate** every possible propositional formula $\varphi_0, \varphi_1, \varphi_2, ...$ defining sets $\Gamma_0, \Gamma_1, \Gamma_2, ...$ 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 $\bigcup \{\Gamma_0, \Gamma_1, \Gamma_2, ... \} = \Gamma_{\text{max}}$.
  - Then show that $\Gamma_{\text{max}}$ is consistent, and in fact, maximally consistent.
Sketch, continued

- $\Gamma_{\text{max}}$ 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 $\Gamma_{\text{max}} \vdash \varphi$, then in fact $\varphi \in \Gamma_{\text{max}}$.

- 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 $p \in \Gamma_{\text{max}}$ then $\nu(p) = T$, otherwise $\nu(p) = F$.

  - Then argue that $\nu$ **satisfies** $\Gamma_{\text{max}}$ using closure under derivability (using a soundness-like argument).

- Finally, $\nu$ also satisfies $\Gamma_0$, since $\Gamma_0 \subseteq \Gamma_{\text{max}}$. So $\Gamma_0$ is satisfiable.