Tableaux Calculus
Propositional Logic

A compact version of sequent calculus
The idea

What’s “wrong” with sequent calculus:

Why do we have to copy (?) Γ and ∆ with every rule application?

The answer: tableaux calculus.
The idea:

Describe *backward* sequent calculus rule application but leave Γ and ∆ implicit/shared

Comparison:

**Sequent**  
Proof is a tree labeled by sequents,  
trees grow upwards

**Tableaux**  
Proof is a tree labeled by formulas,  
trees grow downwards

Terminology: *tableau* = tableaux calculus proof tree
Tableaux rules (examples)

Notation: 

\( +F \approx F \) occurs on the right of \( \Rightarrow \)

\( -F \approx F \) occurs on the left of \( \Rightarrow \)

<table>
<thead>
<tr>
<th>S.C.</th>
<th>Tab.</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>( F, \Gamma \Rightarrow \Delta )</td>
<td>( \Gamma \Rightarrow \neg F, \Delta )</td>
<td>( +\neg F )</td>
</tr>
<tr>
<td>( \Gamma \Rightarrow \neg F, \Delta )</td>
<td>( -F )</td>
<td>( -F )</td>
</tr>
<tr>
<td>( \Gamma \Rightarrow F, G, \Delta )</td>
<td>( +F \lor G )</td>
<td>( +F \vee G )</td>
</tr>
<tr>
<td>( \Gamma \Rightarrow F \lor G, \Delta )</td>
<td>( +F \lor G )</td>
<td>( +F )</td>
</tr>
<tr>
<td>( \Gamma \Rightarrow F \lor G, \Delta )</td>
<td>( +F \lor G )</td>
<td>( +G )</td>
</tr>
<tr>
<td>( \Gamma \Rightarrow F, \Delta ) ( \Gamma \Rightarrow G, \Delta )</td>
<td>( +F \land G )</td>
<td>( +F \land G )</td>
</tr>
<tr>
<td>( \Gamma \Rightarrow F \land G, \Delta )</td>
<td>( +F \land G )</td>
<td>( +F )</td>
</tr>
<tr>
<td>( \Gamma \Rightarrow F \land G, \Delta )</td>
<td>( +F \land G )</td>
<td>( +G )</td>
</tr>
<tr>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
</tr>
<tr>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
</tr>
<tr>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
</tr>
<tr>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
</tr>
<tr>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
</tr>
<tr>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
<td>( +F \lor G )</td>
</tr>
</tbody>
</table>
Interpretation of tableaux rule

\[
\frac{F}{FGH}
\]

if \( F \) matches the formula at some node in the tableau extend the end of some branch starting at that node according to \( FGH \).
Example

- $A \rightarrow B$
- $B \rightarrow C$
- $A$
- $C$

$A \rightarrow B, B \rightarrow C, A \Rightarrow C$
From tableau to sequents:

- Every path from the root to a leaf in a tableau represents a sequent.
- The set of all such sequents represents the set of leaves of the corresponding sequent calculus proof.

A branch is closed (proved) if both $+F$ and $-F$ occur on it or $-\bot$ occurs on it.

The root sequent is proved if all branches are closed.

Algorithm to prove $F_1, \ldots \Rightarrow G_1, \ldots$:

1. Start with the tableau $-F_1, \ldots, +G_1, \ldots$.
2. While there is an open branch do
   - Pick some non-atomic formula on that branch,
   - Extend the branch according to the matching rule.
Termination

No formula needs to be used twice on the same branch. But possibly on *different* branches:

$$+\neg A \wedge \neg B$$

$$+ A \lor B$$

A formula occurrence in a tableau can be deleted if it has been used in every unclosed branch starting from that occurrence.
Tableaux rules

\[
\begin{align*}
-\neg F & \ \Rightarrow \ +F \\
+\neg F & \ \Rightarrow \ -F \\
-\neg F \land G & \ \Rightarrow \ -F \\
-\neg F \lor G & \ \Rightarrow \ -F \land -G \\
-\neg F \rightarrow G & \ \Rightarrow \ +F \land -G \\
+\neg F & \ \Rightarrow \ -F \\
+F & \ \Rightarrow \ -F \\
\end{align*}
\]