Nat. Ded. Introduction Sequent Rules

\( \Gamma \) is set of propositions (assumptions/hypotheses)

Hypothesis Introduction:

\[ \Gamma \cup \{A\} \vdash A \quad \text{Hyp} \]

Truth Introduction: And Introduction:

\[ \Gamma \vdash \top \quad \text{I} \]
\[ \Gamma \vdash A \quad \Gamma \vdash B \quad \text{And I} \]

Or Introduction:

\[ \Gamma \vdash A \quad \text{Orl I} \]
\[ \Gamma \vdash A \lor B \quad \text{Org I} \]

Not Introduction: Implication Introduction:

\[ \Gamma \cup \{A\} \vdash \bot \quad \text{Not I} \]
\[ \Gamma \vdash \{A\} \vdash B \quad \text{Imp I} \]
\[ \Gamma \vdash A \Rightarrow B \quad \text{Imp} \]

Nat. Ded. Elimination Sequent Rules

\( \Gamma \) is set of propositions (assumptions/hypotheses)

Not Elimination: Implication Elimination:

\[ \Gamma \vdash \neg A \quad \Gamma \vdash A \quad \text{Not E} \]
\[ \Gamma \vdash B \quad \Gamma \vdash \{B\} \vdash C \quad \text{Imp E} \]

And Elimination:

\[ \Gamma \vdash A \land B \quad \Gamma \vdash \{A\} \vdash C \quad \text{And E} \]
\[ \Gamma \vdash A \land B \quad \Gamma \vdash \{B\} \vdash C \quad \text{And E} \]

False Elimination: Or Elimination:

\[ \Gamma \vdash \bot \quad \text{F E} \]
\[ \Gamma \vdash A \lor B \quad \Gamma \vdash \{A\} \vdash C \quad \Gamma \vdash \{B\} \vdash C \quad \text{Or E} \]
\[ \Gamma \vdash A \lor B \quad \text{Or E} \]

Proof implies Truth

**Theorem (Soundness)**

Suppose \( \{H_1, \ldots, H_n\} \vdash P \) is provable. Then, for every valuation \( v \), if for every \( i \) we have \( v \models H_i \), then \( v \models P \).

**Proof.**

- Fix a proof of \( \{H_1, \ldots, H_n\} \vdash P \)
- Proceed by induction on the structure of the proof tree of \( \{H_1, \ldots, H_n\} \vdash P \).
- **Ind Hyp:** We may assume that, for every subproof of the proof of \( \{H_1, \ldots, H_n\} \vdash P \), if \( v \) satisfies all the hypotheses of the result of the subproof, then \( v \) satisfies the consequent of the result of the subproof.

Proof implies Truth

**Theorem (Soundness)**

Suppose \( \{H_1, \ldots, H_n\} \vdash P \) is provable. Then, for every valuation \( v \), if for every \( i \) we have \( v \models H_i \), then \( v \models P \).

**Proof.**

- Fix a proof of \( \{H_1, \ldots, H_n\} \vdash P \)
- Proceed by induction on the structure of the proof tree of \( \{H_1, \ldots, H_n\} \vdash P \).
- **Ind Hyp:** We may assume that, for every subproof of the proof of \( \{H_1, \ldots, H_n\} \vdash P \), if \( v \) satisfies all the hypotheses of the result of the subproof, then \( v \) satisfies the consequent of the result of the subproof.
Proof implies Truth

Theorem (Soundness)
Suppose \( \{H_1, \ldots, H_n\} \vdash P \) is provable. Then, for every valuation \( v \), if for every \( i \) we have \( v \models H_i \), then \( v \models P \).

Proof.
- Fix a proof of \( \{H_1, \ldots, H_n\} \vdash P \)
- Proceed by induction on the structure of the proof tree of \( \{H_1, \ldots, H_n\} \vdash P \).
  - Ind Hyp: We may assume that, for every subproof of the proof of \( \{H_1, \ldots, H_n\} \vdash P \), if \( v \) satisfies all the hypotheses of the result of the subproof, then \( v \) satisfies the consequent of the result of the subproof.
- Proceed by case analysis on the last rule used in the proof.
- Case: Hyp
  - The \( P \) is among the \( H_i \), so by assumption \( v \models P \).

Proof implies Truth

Case: \( T I \)
- Then \( P = T \) and \( v \models T \) always.

Proof implies Truth

Case: \( \text{And I} \)
- Then \( P = A \land B \) and \( v \models A \lor B \) so \( v \models P \).

Case Or \( R I \) same.
Proof implies Truth

Proof.

Case: T I

Then $P = T$ and $v \models T$ always.

Case: And I

Then there exist $A$ and $B$ s.t. $P = A \land B$ and $(H_1, \ldots, H_n) \vdash A$ and $(H_1, \ldots, H_n) \vdash B$ are provable by subproofs of the proof of $(H_1, \ldots, H_n) \vdash P$.

By inductive hypothesis, since $v \models H_i$ for $i = 1, \ldots, n$, have $v \models A$ and $v \models B$.

Thus $v \models A \land B$ so $v \models P$.

Case Or L I

Then there exist $A$ and $B$ s.t. $P = A \lor B$ and $(H_1, \ldots, H_n) \vdash A$ is provable by a subproof of $(H_1, \ldots, H_n) \vdash P$.

Proof implies Truth

Proof.

Case: T I

Then $P = T$ and $v \models T$ always.

Case: And I

Then there exist $A$ and $B$ s.t. $P = A \land B$ and $(H_1, \ldots, H_n) \vdash A$ and $(H_1, \ldots, H_n) \vdash B$ are provable by subproofs of the proof of $(H_1, \ldots, H_n) \vdash P$.

By inductive hypothesis, since $v \models H_i$ for $i = 1, \ldots, n$, have $v \models A$ and $v \models B$.

Thus $v \models A \land B$ so $v \models P$.

Case Or R I same.
Proof implies Truth

Proof.

- Case: \( T \)
  - Then \( P = T \) and \( v \models T \) always.

- Case: \( \text{And I} \)
  - Then there exist \( A \) and \( B \) s.t. \( P = A \land B \) and \( (H_1, \ldots, H_n) \vdash A \) and \( (H_1, \ldots, H_n) \vdash B \) are provable by subproofs of the proof of \( (H_1, \ldots, H_n) \vdash P \).
  - By inductive hypothesis, since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \) and \( v \models B \).
  - Thus \( v \models A \land B \) so \( v \models P \).

- Case: \( \text{Not I} \)
  - Then there exist \( A \) s.t. \( P = \neg A \) and \( \{ H_1, \ldots, H_n, A \} \vdash F \) is provable by a subproof of the proof of \( \{ H_1, \ldots, H_n \} \vdash P \).
  - By inductive hypothesis, since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \) and \( v \models F \).
  - Thus \( v \models \neg A \land B \) so \( v \models P \).

Case: \( \text{Or} \)

- Case: \( \text{Or I} \)
  - Then there exist \( A \) and \( B \) s.t. \( P = A \lor B \) and \( (H_1, \ldots, H_n) \vdash A \) is provable by a subproof of \( (H_1, \ldots, H_n) \vdash P \).
  - By inductive hypothesis, since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \).
  - Thus \( v \models A \lor B \) so \( v \models P \).

- Case: \( \text{Or} \) same.
Proof.

Case: Not I

Then there exists \( A \) s.t. \( P = \neg A \) and \( \{H_1, \ldots, H_n, A\} \vdash F \) is provable by a subproof of the proof of \( \{H_1, \ldots, H_n\} \vdash P \).

- Have \( v \models H_i \) for \( i = 1, \ldots, n \), but not \( v \models F \).
- By Ind. Hyp. must have \( v \not\models A \).
- Thus \( v \models \neg A \).

Case: Imp I

Then there exist \( A \) and \( B \) s.t. \( P = A \Rightarrow B \) and \( \{H_1, \ldots, H_n, A\} \vdash B \) is provable by a subproof of the proof of \( \{H_1, \ldots, H_n\} \vdash P \).

- By inductive hypothesis, since \( v \models H_i \) for \( i = 1, \ldots, n \), if \( v \models A \) then \( v \models B \), so either have have \( v \models B \) or \( v \not\models A \).
- Thus \( v \models A \Rightarrow B \) so \( v \models P \).
Proof implies Truth

Proof.

Case: Imp E

Then there exist A s.t. \( \{H_1, \ldots, H_n\} \vdash A \) and \( \{H_1, \ldots, H_n\} \vdash A \) are provable by subproofs of \( \{H_1, \ldots, H_n\} \vdash P \).

By inductive hypothesis, since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \) and \( v \models A \), which is impossible.

Thus either the last rule is not Not E or for some \( i \) we have \( v \not\models H_i \), contradicting theorem assumption.

Case: Not E

Then there exist A s.t. \( \{H_1, \ldots, H_n\} \vdash A \) and \( \{H_1, \ldots, H_n\} \vdash A \) are provable by subproofs of \( \{H_1, \ldots, H_n\} \vdash P \).

By inductive hypothesis, since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \) and \( v \models A \), which is impossible.

Thus either the last rule is not Not E or for some \( i \) we have \( v \not\models H_i \), contradicting theorem assumption.

Case: Imp E

Then there exist A and B s.t. \( \{H_1, \ldots, H_n\} \vdash A \Rightarrow B \) and \( \{H_1, \ldots, H_n\} \vdash A \) and \( \{H_1, \ldots, H_n, B\} \vdash P \) are provable by subproofs of \( \{H_1, \ldots, H_n\} \vdash P \).

By hypothesis, for \( i = 1 \ldots n \), have \( v \models A \Rightarrow B \) and \( v \models A \).
Proof implies Truth

Proof.

Case Not E
- Then there exist $A$ s.t. $\{H_1, \ldots, H_n\} \vdash A$ and $\{H_1, \ldots, H_n\} \vdash \neg A$ are provable by subproofs of $\{H_1, \ldots, H_n\} \vdash P$.
- By inductive hypothesis, since $\nu \models H_i$ for $i = 1 \ldots n$, have $\nu \models A$ and $\nu \models \neg A$, which is impossible.
- Thus either the last rule is not Not E or for some $i$ we have $\nu \not\models H_i$, contradicting theorem assumption.

Case: Imp E
- Then there exist $A$ and $B$ s.t. $\{H_1, \ldots, H_n\} \vdash A \Rightarrow B$ and $\{H_1, \ldots, H_n, A\} \vdash P$ are provable by subproofs of $\{H_1, \ldots, H_n\} \vdash P$.
- By Ind. Hyp., since $\nu \models H_i$ for $i = 1 \ldots n$, have $\nu \models A$ and $\nu \models B$, therefore $\nu \models P$.

Case: And L E
- Then there exist $A$ and $B$ s.t. $\{H_1, \ldots, H_n\} \vdash A \land B$ and $\{H_1, \ldots, H_n, A\} \vdash P$, have $\nu \models \neg Hi$, which violates theorem assumption.
- Hence, either last rule in proof not And L E or for some $i$ we have $\nu \not\models H_i$, contradicting theorem assumption.

Case: Imp E
- Then there exist $A$ and $B$ s.t. $\{H_1, \ldots, H_n\} \vdash A \Rightarrow B$ and $\{H_1, \ldots, H_n, A\} \vdash P$ are provable by subproofs of $\{H_1, \ldots, H_n\} \vdash P$.
- By Ind. Hyp., since $\nu \models H_i$ for $i = 1 \ldots n$, have $\nu \models A$ and $\nu \models B$, therefore $\nu \models P$.
- Again by Ind. Hyp., $\nu \models P$. 

Therefore, either last rule in proof not And L E or for some $i$ we have $\nu \not\models H_i$, contradicting theorem assumption.
Proof implies Truth

Proof.

Case: \( \text{And}_E \)
- Then there exist \( A \) and \( B \) s.t. \( (H_1, \ldots, H_n) \vdash A \land B \) and 
  \( (H_1, \ldots, H_n, A) \vdash P \) are provable by subproofs of \( (H_1, \ldots, H_n) \vdash P \).
- By Ind. Hyp., since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \land B \) and \( v \models A \) 
  (and \( v \models B \)).
- Again by Ind. Hyp, \( v \models P \).
- Case: \( \text{And}_E \) same.
- Case: \( \text{F} \) same.

Case: \( \text{F} \) same.

Proof implies Truth

Proof.

Case: \( \text{And}_E \)
- Then there exist \( A \) and \( B \) s.t. \( (H_1, \ldots, H_n) \vdash A \land B \) and 
  \( (H_1, \ldots, H_n, A) \vdash P \) are provable by subproofs of \( (H_1, \ldots, H_n) \vdash P \).
- By Ind. Hyp., since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \land B \) and \( v \models A \) 
  (and \( v \models B \)).
- Again by Ind. Hyp, \( v \models P \).
- Case: \( \text{And}_E \) same.
- Case: \( \text{F} \) same.

Case: \( \text{F} \) same.

Proof implies Truth

Proof.

Case: \( \text{And}_E \)
- Then there exist \( A \) and \( B \) s.t. \( (H_1, \ldots, H_n) \vdash A \land B \) and 
  \( (H_1, \ldots, H_n, A) \vdash P \) are provable by subproofs of \( (H_1, \ldots, H_n) \vdash P \).
- By Ind. Hyp., since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \land B \) and \( v \models A \) 
  (and \( v \models B \)).
- Again by Ind. Hyp, \( v \models P \).
- Case: \( \text{And}_E \) same.
- Case: \( \text{F} \) same.

Case: \( \text{F} \) same.

Proof implies Truth

Proof.

Case: \( \text{And}_E \)
- Then there exist \( A \) and \( B \) s.t. \( (H_1, \ldots, H_n) \vdash A \land B \) and 
  \( (H_1, \ldots, H_n, A) \vdash P \) are provable by subproofs of \( (H_1, \ldots, H_n) \vdash P \).
- By Ind. Hyp., since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \land B \) and \( v \models A \) 
  (and \( v \models B \)).
- Again by Ind. Hyp, \( v \models P \).
- Case: \( \text{And}_E \) same.
- Case: \( \text{F} \) same.

Case: \( \text{F} \) same.

Proof implies Truth

Proof.

Case: \( \text{And}_E \)
- Then there exist \( A \) and \( B \) s.t. \( (H_1, \ldots, H_n) \vdash A \land B \) and 
  \( (H_1, \ldots, H_n, A) \vdash P \) are provable by subproofs of \( (H_1, \ldots, H_n) \vdash P \).
- By Ind. Hyp., since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \land B \) and \( v \models A \) 
  (and \( v \models B \)).
- Again by Ind. Hyp, \( v \models P \).
- Case: \( \text{And}_E \) same.
- Case: \( \text{F} \) same.

Case: \( \text{F} \) same.

Proof implies Truth

Proof.

Case: \( \text{And}_E \)
- Then there exist \( A \) and \( B \) s.t. \( (H_1, \ldots, H_n) \vdash A \land B \) and 
  \( (H_1, \ldots, H_n, A) \vdash P \) are provable by subproofs of \( (H_1, \ldots, H_n) \vdash P \).
- By Ind. Hyp., since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \land B \) and \( v \models A \) 
  (and \( v \models B \)).
- Again by Ind. Hyp, \( v \models P \).
- Case: \( \text{And}_E \) same.
- Case: \( \text{F} \) same.

Case: \( \text{F} \) same.
Case: Or E
Then there exist A and B s.t. \( \{H_1, \ldots, H_n\} \vdash A \lor B \) and \( \{H_1, \ldots, H_n, A\} \vdash P \) and \( \{H_1, \ldots, H_n, B\} \vdash P \) are all provable by subproofs of \( \{H_1, \ldots, H_n\} \vdash P \).
By Ind. Hyp., since \( v \models H_i \) for \( i = 1 \ldots n \), have \( v \models A \lor B \).
Case: \( v \models A \)
* Ind. Hyp. implies \( v \models P \).
Case: \( v \models B \)
Truth does not imply Proof . . .

- For given rules, can not prove \( A \lor \neg A \)
- Need an axiom.

Model Checking for Propositions

- Problem: Would like an efficient way to answer for a given proposition \( P \):
  - Is \( P \) a tautology?
  - Is \( P \) satisfiable?
    - Note: A general algorithm to answer the first can be used to answer the second and vice versa.
    - Does a given valuation satisfy \( P \)?
- Difficulty: Answering if \( P \) is satisfiable is NP-complete
- Algorithms exist with good performance in general practice
- BDDs are one such

Binary Decision Trees

- Binary decision tree is a (rooted, directed) edge and vertex labeled tree with two types of vertices – internal nodes, and leaves – such that:
  - Internal nodes have exactly two out edges
  - Leaves have no out edges
  - Internal nodes are labeled by atomic propositions (variables)
  - Leaves are labeled by true or false.
  - Left edges labeled false and right edges labeled true.
  - For each path (branch) in the tree, each atomic proposition may label at most one vertex of that path.

Example:

\[(A \land B) \lor (\neg C)\]

Example: Different Variable Ordering - Different Tree

\[(A \land B) \lor (\neg C)\]
Translating Original Propositions into if_then_else

- Start with proposition $P_0$ with variables $v_1, \ldots, v_n$
- $P[c/v]$ is the proposition resulting from replacing all occurrences of variable $v$ with constant $c$
- Let $P$ be the result of evaluating every subexpression of $P$ containing no variables
  - Let $P_1 = \text{if } v_1 \text{ then } P_0[T/v_1] \text{ else } P_0[F/v_1]$
  - Let $P_2 = \text{if } v_2 \text{ then } P_{11}[T/v_2] \text{ else } P_{11}[F/v_2]$
- $P_n$ is logically equivalent to $P$, but only uses if_then_else_
  - Valuation satisfies $P$ if and only if it satisfies $P_n$
  - $P_n$ depends on the order of variables $v_1, \ldots, v_n$
  - $P_n$ directly corresponds to a binary decision tree

Example:

$P = (A \land B) \lor (\neg C)$, variables $\{A, B, C\}$

$P_0 = (A \land B) \lor (\neg C)$
$P_1 = \text{if } A \text{ then } (T \land B) \lor (\neg C) \text{ else } (F \land B) \lor (\neg C)$

Example:

$P = (A \land B) \lor (\neg C)$, variables $\{A, B, C\}$

$P_0 = (A \land B) \lor (\neg C)$
$P_1 = \text{if } A \text{ then } (T \land B) \lor (\neg C) \text{ else } (F \land B) \lor (\neg C)$
$P_2 = \text{if } B \text{ then } (A \land T) \lor (\neg C) \text{ else } (F \land T) \lor (\neg C)$
$P_3 = \text{if } A \text{ then } (T \land F) \lor (\neg C) \text{ else } (F \land F) \lor (\neg C)$
Example:

\[ P = (A \land B) \lor (\neg C) \], variables \{ A, B, C \}

\[ P_0 = (A \land B) \lor (\neg C) \]
\[ P_1 = \text{if } A \text{ then } (T \land B) \lor (\neg C) \text{ else } (F \land B) \lor (\neg C) \]
\[ P_2 = \text{if } B \text{ then } (if \ A \text{ then } (T \land T) \lor (\neg C) \text{ else } (F \land T) \lor (\neg C)) \text{ else } (if \ A \text{ then } (T \land F) \lor (\neg C) \text{ else } (F \land F) \lor (\neg C)) \]
\[ P_3 = \text{if } B \text{ then } (if \ A \text{ then } T \lor (\neg C) \text{ else } F \lor (\neg C)) \]
\[ P_4 = \text{if } B \text{ then } (if \ A \text{ then } F \lor (\neg C) \text{ else } F \lor (\neg C)) \]
\[ P_5 = \text{if } C \text{ then } (if \ B \text{ then } (if \ A \text{ then } T \text{ else } F) \text{ else } (if \ A \text{ then } F \text{ else } F)) \]
\[ P_6 = \text{if } C \text{ then } (if \ B \text{ then } (if \ A \text{ then } T \text{ else } F) \text{ else } (if \ A \text{ then } F \text{ else } F)) \]

Example, cont.

\[ P_3 \text{ corresponds to second binary decision tree given earlier} \]
\[ \bullet \text{ Any proposition in strict if}_a\text{then}_b\text{else}_c form corresponds directly to a binary decision tree that accepts exactly the valuations that satisfy (model) the proposition.} \]

Binary Decision Diagram

- Binary decision trees may contain (much) redundancy
- Binary Decision Diagram (BDD): Replace trees by (rooted) directed acyclic graphs
- Require all other conditions still hold
- Generalization of binary decision trees
- Allows for sharing of common subtrees.
- Accepts / rejects valuations as with binary decision trees.
Reduced Ordered Binary Decision Diagrams

- Problem: given proposition may correspond to many different BDDs
- How to create a (compact) canonical BDD for a proposition such that two different propositions are logically equivalent if and only if they have the same (isomorphic) canonical BDD
- Start: order propositional variables $v_i < v_j$
- Bryant showed you can obtain such a canonical BDD by requiring
  - Variables should appear in order on each path for root to leaf
  - No distinct duplicate (isomorphic) subtrees (including leaves)

Achieving Canonical Form

- Start with an Ordered BDD (all edges in correct order)
- Repeat following until none apply
- Remove duplicate leaves: Eliminate all but one leaf with a given label and redirect all edges to the eliminated leaves to the remaining one
- Remove duplicate nonterminals: If node $n$ and $m$ have the same variable label, their left edges point to the same node and their right edges point to the same node, remove one and redirect edges that pointed to it to the other
- Remove redundant tests: If both out edges node of $n$ point to node $m$, eliminate $n$ and redirect all edges coming into $n$ to $m$
- Bryant gave procedure to do the above that terminates in linear time

Example