Logic in computer Science

Proof Procedures for Propositional Logic

1

Proof Procedures

- A proof procedure P(A, B) takes a set A of axioms and a formula B as input, and returns true if it claims B is a theorem from A.

Two properties of P(A, B):

- Soundness: If P(A, B) returns true, then B is indeed a theorem of A.
- Completeness: If B is a theorem of A, then P(A, B) will return true in a finite number of steps.

2

Special Purposes of Proof Procedures

- Theorem Prover P(X, Y)
  - P(X, Y) returns true iff Y is a logical consequence of X (i.e., X entails Y0, or Y is a theorem of X).
- Tautology Prover T(A)
  - T(A) returns true iff A is a tautology (i.e., A is valid, or |= A)
- Refutation Prover R(B)
  - R(B) returns true iff B is unsatisfiable.

3

Equivalences of P(X,Y), T(A), R(B)

- Theorem Prover P(X, Y)
  - P(X, Y) as T(A): P(1, A), since |= A iff 1 |= A
  - P(X, Y) as R(B): P(B, 0), since B is unsat. iff B |= 0
- Tautology Prover T(A)
  - T(A) as P(X, Y): T(X→Y), since X|=Y iff |= X→Y
  - T(A) as R(B): T(¬B), since B is unsatifiable iff |= ¬B
- Refutation Prover R(B)
  - R(B) as P(X, Y): R(X∧ ¬Y), since X|=Y iff X∧ ¬Y is unsat.
  - R(B) as T(A): R(¬A), since |= A iff ¬A is unsatisfiable.

4

Proof Methods

- Truth Table
- Algebraic Substitution
- Normal Forms
- Semantic Tableau
- Inference Systems

5

Proof Methods

- Truth Table
  - Can be used for constructing theorem prover, tautology prover, refutation prover.
  - Work only for a small number of variables

6

Proof Methods

- Truth Table
- Algebraic Substitution
  - Apply algebraic laws to replace equal by equal
  - Can be used for constructing tautology prover
  - Difficulty to use

7

Proof Methods

- Truth Table
- Algebraic Substitution
- Norm Forms
  - CNF: as a tautology prover
    - CNF A is a tautology iff every clause is a tautology.
  - DNF: as a refutation prover
    - DNF A is unsatisfiable iff every minterm is unsatisfiable.
  - INF (ROBDD): as a tautology prover (reduced to 1) and a refutation prover (reduced to 0).

8

Proof Methods

- Truth Table
- Algebraic Substitution
- Normal Forms
- Semantic Tableau
  - As a refutation prover, similar to DNF

9

Proof Methods

- Truth Table
- Algebraic Substitution
- Canonical Forms
- Semantic Tableau
- Inference Systems
  - Hilbert systems: as a tautology prover
  - Natural deduction: as a theorem prover
  - Resolution: as a refutation prover

10

Semantic Tableau

- A graphic tool showing the conversion of DNF, which can serve as a refutation prover.
- The graph starts with a single node containing the original formula and uses logical equivalences to expand the graph.
- Equivalences used in the conversion are divided into two sets: α-rules and β-rules.
- α-rule produces a conjunction of formulas and creates one successor node ($\wedge$ is replaced by ",")
- β-rule produces a disjunction of formulas and creates two successor nodes (for each disjunct)

11

11

Semantic Tableau: Example

$A = p \wedge (\neg q \vee \neg p)$

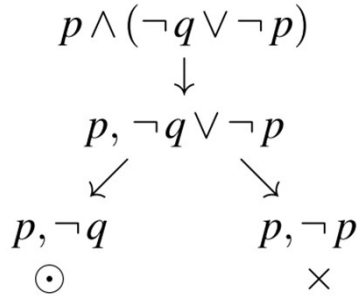
$\equiv p, \neg q \vee \neg p$ // "," for $\wedge$

$\equiv \{p, \neg q\} \mid \{p, \neg p\}$ // "|" for $\vee$

$\equiv \{p, \neg q\} \mid 0$

$\equiv p, \neg q$ // DNF

$\neq 0$



A is not unsatisfiable.

12

Semantic Tableau: Example

$B = (p \vee q) \wedge (\neg p \wedge \neg q)$

$\equiv p \vee q, \neg p \wedge \neg q$

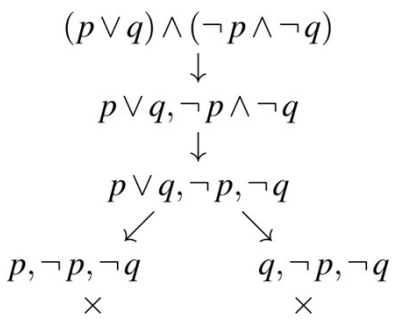
$\equiv p \vee q, \neg p, \neg q$

$\equiv \{p, \neg p, \neg q\} | \{q, \neg p, \neg q\}$

$\equiv 0 \mid 0$

$\equiv 0$ // DNF

B is unsatisfiable.



13

α-rules

- α-rule produces a conjunction of formulas and creates one successor node (∧ is replaced by ",")
  - A ∧ B ≡ A, B // "," for ∧
  - A ↓ B ≡ ¬ A, ¬ B
  - A ⊕ B ≡ (A ∨ B), (¬ A ∨ ¬ B)
  - A ↔ B ≡ (¬ A ∨ B), (A ∨ ¬ B)
  - ¬ (A ∨ B) ≡ ¬A, ¬B
  - ¬ (A → B) ≡ A, ¬ B
  - ¬ (A ↑ B) ≡ A, B
  - ¬ ¬ p ≡ p

14

β-rules

- β-rule produces a disjunction of formulas and creates two successor nodes (for each disjunct)
  - $A \vee B \equiv A \mid B$ // "|" for ∨, two branches
  - $A \rightarrow B \equiv \neg A \mid B$
  - $A \uparrow B \equiv \neg A \mid \neg B$
  - $\neg (A \wedge B) \equiv \neg A \mid \neg B$
  - $\neg (A \downarrow B) \equiv A \mid B$
  - $\neg (A \oplus B) \equiv (A \wedge B) \mid (\neg A \wedge \neg B)$
  - $\neg (A \leftrightarrow B) \equiv (\neg A \wedge B) \mid (A \wedge \neg B)$

15

Tableau Rules

- α-rules (AND-rules)

$$\frac{U \cup \{\alpha\}}{U \cup \{\alpha_1, \alpha_2\}}$$

$\alpha \equiv \alpha_1 \wedge \alpha_2$ implies
$U \wedge \alpha \equiv U \wedge \alpha_1 \wedge \alpha_2$

- β-rules (OR-rules)

$$\frac{U \cup \{\beta\}}{U \cup \{\beta_1\} \quad U \cup \{\beta_2\}}$$

distribution law is used.
$\beta \equiv \beta_1 \vee \beta_2$ implies
$U \wedge \beta \equiv (U \wedge \beta_1) \vee (U \wedge \beta_2)$

Tableau Rules preserve logical equivalence of each node with (the disjunction of) its children.

16

Semantic Tableaux

- The user has freedom to choose any formula in a node to apply rules.
- α-rules and β-rules always apply to top operators.
- If a node contains a pair of contradictory formulas, it is equivalent to 0 (false) and called closed.
- If every leaf node is closed, the tableau is said closed and the original formula is unsatisfiable.

$$\neg[A \rightarrow (B \rightarrow A)]$$
$$\downarrow$$
$$A, \neg(B \rightarrow A)$$
$$\downarrow$$
$$A, B, \neg A$$
$$\times$$

17

17

Formal Properties

- Termination

  For every A, the tableau will stop expansion when no rules can apply. This tableau is *finished*.
- Soundness

  If the tableau is *closed,* then A is *unsatisfiable*.
- Completeness

  If the tableaux is finished and not closed (i.e., *open*) then A is *satisfiable*.

  Every open leaf gives us at least one model.

18

18

Use of Semantic Tableau

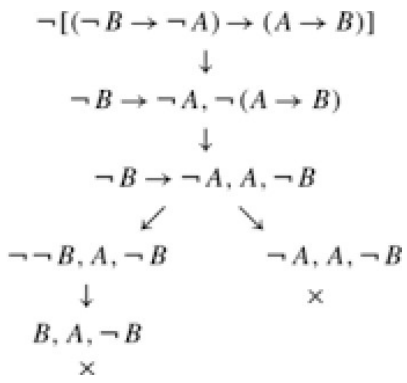
- Refutation Prover: To show that B is unsatisfiable, show that the tableau of B is closed.
- Tautology Prover: To show that A is valid, show that the tableau of ¬A is closed.
- Theorem Prover: To show that Y is a theorem of X, show that the tableau of (X∧¬Y) is closed.

19

Semantic Tableaux: Example

- To show that $(\neg B \rightarrow \neg A) \rightarrow (A \rightarrow B)$ is a tautology, we show that the tableau of $\neg((\neg B \rightarrow \neg A) \rightarrow (A \rightarrow B))$ is closed.

All branches are closed ⇨
$(\neg B \rightarrow \neg A) \rightarrow (A \rightarrow B)$ is valid.

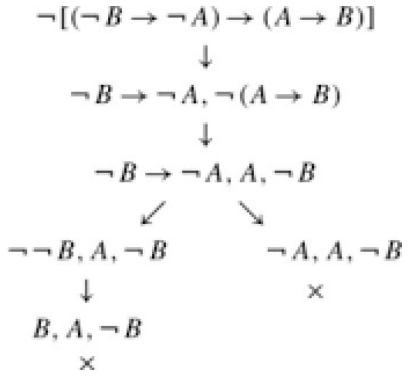


20

Semantic Tableaux: Example

- Tree nodes can be labelled by a string of 1 and 2:
  - The root's label is ε (the empty string)
  - If the parent's label is x, the 1st child's label is x1 and the 2nd child's label is x2.

- $\neg((\neg B \to \neg A) \to (A \to B))$
- 1: $\neg B \to \neg A, \neg(A \to B)$
- 11: $\neg B \to \neg A, A, \neg B$
- 111: $\neg \neg B, A, \neg B$
- 1111: $B, A, \neg B$ closed
- 112: $\neg A, A, \neg B$ closed



21

Semantic Tableaux: Example

- Tree nodes are labelled by a string of 1 and 2
- $\neg(((p \to q) \wedge (q \to r)) \to (p \to r))$ by α-¬ →
- 1: $(p \to q) \wedge (q \to r), \neg(p \to r)$ by α-¬ →
- 11: $(p \to q) \wedge (q \to r), p, \neg r$ by α-∧
- 111: $p \to q, q \to r, p, \neg r$ by β-→
- 1111: $\neg p, q \to r, p, \neg r$ closed
- 1112: $q, q \to r, p, \neg r$ by β-→
- 11121: $q, \neg q, p, \neg r$ closed
- 11122: $q, r, p, \neg r$ closed

- All branches are closed ⇨ the original is unsatisfiable.
- $((p \to q) \wedge (q \to r)) \to (p \to r)$ is a tautology.

22

Tableaux versus Truth tables

- Tableaux can be extended to handle many logics.
- Tableaux are a more convenient presentation of the familiar truth table analysis.
- "Tableaux are more efficient than truth tables."
- … not exactly:

  $(p \vee q \vee r) \wedge (p \vee q \vee \sim r) \wedge (p \vee \sim q \vee r) \wedge (p \vee \sim q \vee \sim r) \wedge$
  $(\sim p \vee q \vee r) \wedge (\sim p \vee q \vee \sim r) \wedge (\sim p \vee \sim q \vee r) \wedge (\sim p \vee \sim q \vee \sim r)$

  - This formula has a tableau of at least 15 nodes.
  - There are formulas with n variables of length $O(2^n)$
  - ➔ truth tables have 2^n rows
  - ➔ closed tableaux has O(n!) nodes, n! grows faster than 2^n

23

23

Inference Systems

- Expressed as a set of rules (inference rules), which represent an entailment relation between the conditions and conclusions.
- Used to design a theorem prover P(A, B), often denoted by A |- B.
- Properties of premises (axioms):
  - Consistency: the premises have no contradiction.
  - Independence: no premise can be proved from other premises.
- Properties of an inference system:
  - Soundness: every inference rule must be a theorem.
  - Completeness: every theorem can be proved by the given inference system.

24

Hilbert System

- Every instance of the following axioms is a theorem:
  - $A \rightarrow (B \rightarrow A)$
  - $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$
  - $(\neg B \rightarrow \neg A) \rightarrow (A \rightarrow B)$
- Every formula derived from Modus Ponens is a theorem:
- $\dfrac{A,\ \ A \rightarrow B}{B}$ (or $A,\ \ A \rightarrow B \mid\!- B$)

Note: $A \wedge B =_{df} \neg(A \rightarrow \neg B)$; $A \vee B =_{df} (\neg A \rightarrow B)$

25

Definition: Proof

- A *proof* of B from A for an inference system S is a list of formulas $F_1, F_2, \ldots, F_n=B$, such that for each F_i, either F_i is in A, or F_i is generated by an inference rule of S from $F_1, F_2, \ldots, F_{i-1}$.
- A proof of $A \rightarrow A$ in Hilbert System:

1. **$(A \rightarrow ((A \rightarrow A) \rightarrow A)) \rightarrow ((A \rightarrow (A \rightarrow A)) \rightarrow (A \rightarrow A))$**
   from $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$ with $B/A \rightarrow A, C/A$
2. **$A \rightarrow ((A \rightarrow A) \rightarrow A)$** from $(A \rightarrow (B \rightarrow A))$ with $B/A \rightarrow A$
3. **$(A \rightarrow (A \rightarrow A)) \rightarrow (A \rightarrow A)$** MP:2,1
4. **$A \rightarrow (A \rightarrow A)$** from $(A \rightarrow (B \rightarrow A))$ with B/A
5. **$A \rightarrow A$** MP:4,3

Hence: **$\mid\!- A \rightarrow A$** .

26

Hilbert System is Sound

- Every axiom is a tautology:
  - $A \rightarrow (B \rightarrow A)$
  - $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$
  - $(\neg B \rightarrow \neg A) \rightarrow (A \rightarrow B)$
- Modus ponens is sound: $A \wedge (A \rightarrow B) \models B$

27

Inference Rules in Natural Deduction

- For each operator op of 0, 1, $\neg$, $\bot$, $\wedge$, $\vee$, $\rightarrow$, and each formula p with outermost operator op, we give:
- A set of Introduction rules for op, describing under which conditions p is true;
- A set of Elimination rules for op, describing what we may infer from the truth of p.

28

Inference Rules in Natural Deduction

- **Rules for 0**
- **Introduction rules:** p, ¬p |- 0.
- **Elimination rules:** 0 |- p.
- **Rules for ∧**
- **Introduction rules:** *p, q* |- *p* ∧ *q*
- ***Elimination rules:*** *p* ∧ *q* |- *p*, *p* ∧ *q* |- *q*
- **Rules for ∨**
- **Introduction rules:** *p* |- *p* ∨ *q*, *p* |- *q* ∨ *p*,
- ***Elimination rules:*** *p* ∨ *q*, ¬*q* |- *p*, *p* ∨ *q*, ¬*p* |- *q*

29

Inference Rules in Natural Deduction

- **Rules for ¬**
- **Introduction rules:** *p* |- ¬ ¬ *p*
- ***Elimination rules:*** ¬ ¬ *p* |- *p*
- **Rules for →**
- **Introduction rules:** *If p* |- *q*, *then* |- *p* → *q*
- ***Elimination rules:*** *p* → *q*, *p* |- *q* (*modus ponens*)

30

Inference Rules in Natural Deduction

- More rules can be added as needed
- $p \rightarrow q, p \mid\!- q$ *modus ponens* (*MP*)
- $p \rightarrow q, \neg q \mid\!- \neg p$ *modus tollens* (*MT*)
  - $(p \rightarrow q) \wedge \neg q \mid\!= \neg p$
- $p \rightarrow q \mid\!- \neg q \rightarrow \neg p$ *contraposition* (*CP*)
  - $(p \rightarrow q) \equiv \neg q \rightarrow \neg p$
- $p \rightarrow q, q \rightarrow r \mid\!- p \rightarrow r$ *imply transitivity* (*IT*)
  - $(p \rightarrow q) \wedge (q \rightarrow r) \mid\!= (p \rightarrow r)$
- $p \vee q, \neg q \vee r \mid\!- p \vee r$ *resolution* (R)
  - $(p \vee q) \wedge (\neg q \vee r) \mid\!= (p \vee r)$
- These rules are sound.

31

Soundness of IT and R

- *imply transitivity* (*IT*): A = $(p \rightarrow q) \wedge (q \rightarrow r) \rightarrow (p \rightarrow r)$
- *resolution* (R): B = $(p \vee q) \wedge (\neg q \vee r) \rightarrow (p \vee r)$

p	q	r	$p \rightarrow q$	$q \rightarrow r$	$p \rightarrow r$	$p \vee q$	$\neg q \vee r$	$p \vee r$	A	B
0	0	0	1	1	1	0	1	0	1	1
0	0	1	1	1	1	0	1	1	1	1
0	1	0	1	0	1	1	0	0	1	1
0	1	1	1	1	1	1	1	1	1	1
1	0	0	0	1	0	1	1	1	1	1
1	0	1	0	1	1	1	1	1	1	1
1	1	0	1	0	0	1	0	1	1	1
1	1	1	1	1	1	1	1	1	1	1

Replace p by $\neg p$ in B, $(\neg p \vee q) \wedge (\neg q \vee r) \rightarrow (\neg p \vee r)$
which is equivalent to $(p \rightarrow q) \wedge (q \rightarrow r) \rightarrow (p \rightarrow r)$

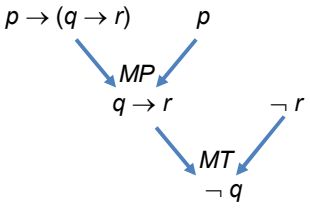
32

A Proof in Natural Deduction

- $p \rightarrow q, p \vdash q$ *modus ponens* (*MP*)
- $p \rightarrow q, \neg q \vdash \neg p$ *modus tollens* (*MT*)

- A proof of $\neg q$ from $p \rightarrow (q \rightarrow r), p, \neg r$

1. $p \rightarrow (q \rightarrow r)$ assumed
2. p assumed
3. $q \rightarrow r$ MP, 1, 2
4. $\neg r$ assumed
5. $\neg q$ MT, 3, 4



33

A Proof in Natural Deduction

$p \wedge \neg\neg q \vdash \neg\neg p \wedge q$

#	Formulas		Reason
(1)	$p \wedge \neg\neg q$		Axiom
(2)	p		1 ∧E
(3)	$\neg\neg q$		1 ∧E
(4)	q		3 ¬¬E
(5)	$\neg\neg p$		2 ¬¬I
(6)	$\neg\neg p \wedge q$		4, 5 ∧I

34

Formal Properties of Natural Deduction

- **Soundness**: every proved formula must be a theorem.

  If |- p then |= p

- **Completeness**: every theorem can be proved by the natural deduction system.

  If |= p then |- p

35

Proof Comparison

A proof of A, ¬ A |– B in Hilbert System

1.	A	assumption
2.	¬A	assumption
3.	(¬B → ¬A) → (A → B)	Axiom3
4.	¬A → (¬B → ¬A)	Axiom1
5.	¬B → ¬A	MP: 2,4
6.	A → B	MP: 5,3
7.	B	MP: 1,6

A proof of A, ¬ A |– B in Natural Deduction

1.	A	assumption
2.	¬A	assumption
3.	0	0-Introduction
4.	B	0-Elimination

36

36

Comparison of Inference Systems

	Hilbert System	Natural Deduction	Resolution
Axiom schemes	three	zero	zero
Inference Rules	one	many	one
Soundness	Yes	Yes	Yes
Completeness	Yes	Yes	Yes
Convenience	No	Yes	Yes
Practicableness	No	No	Yes

37