Mathematical Logic 2016

Lecture 5: Proof methods - Tableaux and Resolution

Instructor: Ashutosh Gupta

TIFR, India

Compile date: 2016-08-21

Where are we and where are we going?

We have seen

- propositional logic syntax and semantics
- truth tables for deciding SAT
- normal forms

We will see

- proof methods - tableaux and resolution
- implementation issues in resolution

Topic 5.1

Proof Methods

Proof methods for entailment

Consider a (in)finite set of formulas Σ and a formula F.

A proof method establishes if the following proof query holds.

$$\Sigma \models F$$

Refutation proof method

We will present refutation proof methods that only apply to the following queries.

$$\Sigma \models \bot.$$

This is not a restriction.

For queries $\Sigma \models F$, we pass the following input to the refutation methods.

$$\Sigma \cup \{\neg F\} \models \bot$$

Two proof systems

We will present the following two refutation proof methods.

- Tableaux
- Resolution

Some additional notation on formulas

We assume that $\top$, $\bot$, $\Leftrightarrow$ and $\oplus$ are removed using the following equivalences

- $p \Leftrightarrow q \equiv (p \wedge q) \vee (p \wedge q)$
- $p \oplus q \equiv (\neg p \wedge q) \vee (p \wedge \neg q)$
- $\top \equiv (p \wedge \neg p)$ for some $p \in$ **Vars**
- $\bot \equiv (p \vee \neg p)$ for some $p \in$ **Vars**

In order to avoid writing many cases we will use a uniform notation.

Conjunctive			Disjunctive		
α	α_1	α_2	β	β_1	β_2
$(F \wedge G)$	F	G	$\neg(F \wedge G)$	$\neg F$	$\neg G$
$\neg(F \vee G)$	$\neg F$	$\neg G$	$(F \vee G)$	F	G
$\neg(F \Rightarrow G)$	F	$\neg G$	$(F \Rightarrow G)$	$\neg F$	G

A non-literal formula can be from one of the following three types.

- α
- β
- $\neg\neg F$

We can further reduced the number of cases by removing $\Rightarrow$ and applying NNF transformation. However, $\Rightarrow$ helps in human readability and NNF destroys the high level structure.

Game of symbol pushing

A proof method is essentially a game of symbol pushing with an objective.

In the end, we need to show that the game has a meaning and connects to the semantics.

Topic 5.2

Tableaux

Tableaux proof method

Tableaux proof method

- takes a set of formula Σ as input and
- produces a finite labelled tree called tableaux as a proof.

Tableaux

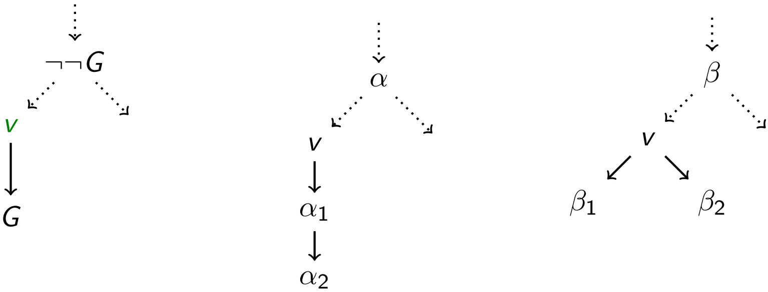
- The tree is labelled with formulas.
- Branching in the tree represents the cases in proofs.
- The goal of the method is to find two nodes in each branch of the tree such that their labels are F and $\neg F$ for some formula F.

Tableaux

Definition 5.1

A tableaux T for a set of formulas Σ is a finite labelled tree that is initially empty and expanded according to the following tableaux expansion rules.

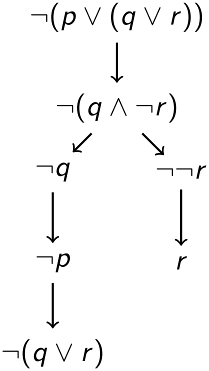
1. *$F \in \Sigma$ labelled node is added as a child to a leaf or root if empty tree*
2. *Let v be a leaf of T. If an ancestor of v is labelled with F then children to v are added using the following rules.*
   - *$F = \neg\neg G$: a child is added to v with label G*
   - *$F = \alpha$: a child and grand child to v added with labels α_1 and α_2*
   - *$F = \beta$: two children to v are added with labels β_1 and β_2*



Example : Tableaux

Example 5.1

Consider $\Sigma = \{\neg(p \vee (q \vee r)), \neg(q \wedge \neg r)\}$



All intermediate states are tableaux.

Closed Tableaux

Definition 5.2

A branch of a tableaux is closed if it contains two nodes with labels F and $\neg F$ for a formula F.

Definition 5.3

A branch of a tableaux is atomically closed if it contains two nodes with labels p and $\neg p$ for a variable p.

Definition 5.4

A tableaux is (atomically) closed if each branch of the tableaux is (atomically) closed.

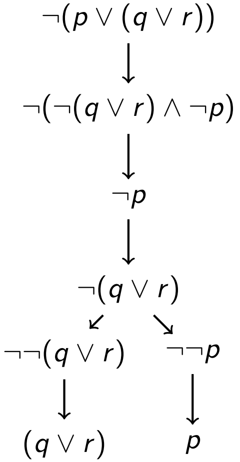
Exercise 5.1

Prove that a closed tableaux can always be expanded to a atomically closed tableaux.

Example : Closed Tableaux

Example 5.2

Consider $\Sigma = \{\neg(p \vee (q \vee r)), \neg(\neg(q \vee r) \wedge \neg p)\}$



Proven by tableaux

Definition 5.5

For a set of formulas Σ, if there exists a closed tableaux then we say Σ is tableaux inconsistent. Otherwise, tableaux consistent.

Definition 5.6

If $\{\neg F\}$ is tableaux inconsistent then we write $\vdash_{pt} F$, i.e., F is a theorem of the tableaux proof method. We say a closed tableaux for $\{\neg F\}$ is a tableaux proof of F.

We will later show that $\models F$ iff $\vdash_{pt} F$.

Exercise 5.2

a. Describe the structure of a tableaux of a DNF formula

b. Describe the structure of a tableaux of a CNF formula

Commentary: pt stands for "propositional tableaux".

Practice tableaux

Exercise 5.3

Prove that the following formulas are theorems in tableaux proof method

1. $(p \Rightarrow q) \land (p \lor q) \Rightarrow q$
2. $(p \Rightarrow q) \land (q \Rightarrow r) \Rightarrow \neg(\neg r \land p)$
3. $(q \lor (r \land s)) \land (q \Rightarrow t) \land (t \Rightarrow s) \Rightarrow s$

Topic 5.3

Resolution

Clauses as sets

Definition 5.7 (clause redefined)

A clause is a finite set of formulas $\{F_1, \ldots, F_n\}$ *and interpreted as* $F_1 \vee .. \vee F_n$.

Here we do not require F_i to be a literal.

For a clause C and a formula F, we will write $F \cup C$ to denote $\{F\} \cup C$.

Example 5.3

$\{(p \Rightarrow q), q\}$ *is a clause.*

Resolution proof method

Resolution proof method takes a set of formulas Σ and produces a sequence of clauses as a proof.

Clauses in the proof are either from Σ or consequences of previous clauses.

The goal of the proof method is to find the empty clause, which stands for inconsistency.

Resolution derivation

Definition 5.8

A resolution derivation R for a set of formulas Σ is a finite sequence of clauses that are generated by the following resolution expansion rules.

1. *$\{F\}$ is appended in R if $F \in \Sigma$*
2. *If a clause $F \cup C$ is in R then new clauses are appended using the following rules*
   2.1 *$F = \neg\neg G$: $G \cup C$ is appended*
   2.2 *$F = \beta$: $\beta_1 \cup \beta_2 \cup C$ is appended*
   2.3 *$F = \alpha$: $\alpha_1 \cup C$ and $\alpha_2 \cup C$ are appended*
3. *If clauses $F \cup C$ and $\neg F \cup D$ are in R then $C \cup D$ is appended.*

Rules 2.1-3 are expansion of the structure of the formulas
Rule 3 is called resolution which is valid due to the following implication

$$(p \vee q) \wedge (\neg p \vee r) \models (q \vee r)$$

p is called the pivot, and $(p \vee q)$ and $(\neg p \vee r)$ are called resolvents.

Example: resolution derivation

Example 5.4

Consider $\Sigma = \{\neg(p \vee (\neg p \wedge q)), (q \vee (p \Rightarrow q))\}$

Resolution derivation

1. $\{\neg(p \vee (\neg p \wedge q))\}$ // *Since* $\neg(p \vee (\neg p \wedge q)) \in \Sigma$
2. $\{(q \vee (p \Rightarrow q))\}$ // *Since* $(q \vee (p \Rightarrow q)) \in \Sigma$
3. $\{\neg p\}$ // *applying* α *expansion on 1*
4. $\{\neg(\neg p \wedge q)\}$ // *applying* α *expansion on 1*
5. $\{\neg\neg p, \neg q\}$ // *applying* β *expansion on 4*
6. $\{p, \neg q\}$ // *applying* $\neg$ *expansion on 5*
7. $\{\neg q\}$ // *applying resolution on p,3,6*

Closed Resolution

Definition 5.9
A resolution derivation is closed if it contains the empty clause.

Definition 5.10
A closed resolution derivation is atomically closed if resolution is applied only using literal pivots.

Exercise 5.4
Prove that a closed resolution derivation can always be extended to a atomically closed resolution derivation.

Example 5.5 (closed resolution)

Consider $\Sigma = \{\neg((p \wedge q) \vee (r \wedge s) \Rightarrow ((p \vee (r \wedge s)) \wedge (q \vee (r \wedge s))))\}$

1. $\{\neg(\ (p \wedge q) \vee (r \wedge s) \Rightarrow ((p \vee (r \wedge s)) \wedge (q \vee (r \wedge s)))\ )\}$
2. $\{(p \wedge q) \vee (r \wedge s)\}$ *// applying* α *expansion on 1*
3. $\{\neg(\ (p \vee (r \wedge s)) \wedge (q \vee (r \wedge s))\ )\}$ *// applying* α *expansion on 1*
4. $\{(p \wedge q), (r \wedge s)\}$ *// applying* β *expansion on 2*
5. $\{\neg(p \vee (r \wedge s)), \neg(q \vee (r \wedge s))\}$ *// applying* β *expansion on 3*
6. $\{\neg p, \neg(q \vee (r \wedge s))\}$ *// applying* α *expansion on 5*
7. $\{\neg(r \wedge s), \neg(q \vee (r \wedge s))\}$ *// applying* α *expansion on 5*
8. $\{\neg p, \neg q\}$ *// applying* α *expansion on 6*
9. $\{\neg p, \neg(r \wedge s)\}$ *// applying* α *expansion on 6*
10. $\{\neg(r \wedge s)\}$ *// applying* α *expansion on 7*
11. $\{\neg(r \wedge s), \neg q\}$ *// applying* α *expansion on 7*
12. $\{p \wedge q\}$ *// applying resolution on* $(r \wedge s)$*,10,4*
13. $\{p\}$ *// applying* α *expansion on 12*
14. $\{q\}$ *// applying* α *expansion on 12*
15. $\{\neg q\}$ *// applying resolution on p,13,8*
16. $\{\}$ *// applying resolution on q,14,15*

Proven by resolution

Definition 5.11

For a set of formulas Σ, if there exists a closed resolution derivation then we say Σ is resolution inconsistent. Otherwise, resolution consistent.

Definition 5.12

If $\{\neg F\}$ is resolution inconsistent then we write $\vdash_{pr} F$, i.e., F is a theorem of the proof method. We say a closed resolution derivation of $\{\neg F\}$ is a resolution proof of F.

We will later show that $\models F$ iff $\vdash_{pr} F$.

Exercise 5.5

a. Describe the structure of a resolution proof of a CNF formula

b. Describe the structure of a resolution proof of a DNF formula

Commentary: pr stands for "propositional resolution".

Practice resolution

Exercise 5.6

Prove that the following formulas are theorems in the resolution proof method

1. $(p \Rightarrow q) \wedge (p \vee q) \Rightarrow q$
2. $(p \Rightarrow q) \wedge (q \Rightarrow r) \Rightarrow \neg(\neg r \wedge p)$
3. $(q \vee (r \wedge s)) \wedge (q \Rightarrow t) \wedge (t \Rightarrow s) \Rightarrow s$
4. $(p \vee q) \wedge (r \vee s) \Rightarrow ((p \wedge r) \vee q \vee s)$

Topic 5.4

Implementation issues in resolution

Efficient implementation of proof methods

A proof method implicitly defines a non-deterministic proof search algorithm

In implementing such a algorithm, one needs to ensure that one is not doing unnecessary work.

We will discuss some simple observations that may cut huge search spaces.

This discussion is a preview of much detailed discussion about SAT solvers.

We are ignoring Tableaux here. Similar issues can be discussed for Tableaux.

Expand only once

We need not apply structural expansion rules multiple times on same formulas in resolution derivation.

In a resolution derivation, the order of application of the expansions on a clause is irrelevant.

Example 5.6

Consider clause $\{\alpha, \beta\}$
First α *then* β *expansion results in*

1. $\{\alpha, \beta\}$
2. $\{\alpha_1, \beta\}$
3. $\{\alpha_2, \beta\}$
4. $\{\alpha_1, \beta_1, \beta_2\}$
5. $\{\alpha_2, \beta_1, \beta_2\}$

First β *then* α *expansion results in*

1. $\{\alpha, \beta\}$
2. $\{\alpha, \beta_1, \beta_2\}$
3. $\{\alpha_1, \beta_1, \beta_2\}$
4. $\{\alpha_2, \beta_1, \beta_2\}$

Superset clauses are redundant

Theorem 5.1

For clauses C and D, if $D \subset C$ and the empty clause can be derived using C then it can be derived using D.

If clause C is superset of clause D, then C is redundant.

Exercise 5.7

Prove the above theorem.

Ignore valid clauses in resolution

Definition 5.13

If a clause contains both F and $\neg F$ then the clause is syntactically valid.

If a syntactically valid clause contributes in deriving the empty clause, the descendents clause must participate in some resolution with pivot F.

However, that is impossible.

Example 5.7

$$\frac{\{F, C\} \quad \{\neg F, F, D\}}{\{F, C, D\}} Resolution$$

Note that the resolution fails to remove F in the consequence.

If a syntactically valid clause is generated then we can ignore it for any further expansions, without loss of completeness.

Pure literals

Definition 5.14

If a literal occurs in a CNF formula and its negation does not then it is a pure literal.

Theorem 5.2

The removal of clauses containing the pure literals in a CNF preserves satisfiability.

Exercise 5.8

Prove the above theorem

Unit clause propagation

If $\{F\}$ occurs in a resolution proof, we can remove $\neg F$ from every clause, which is valid because of the following resolutions.

$$\frac{\{F\} \quad \neg F \cup D}{\{D\}}\text{Resolution}$$

Other proof systems

We are skipping the following important proof systems

- Hilbert system
- Natural deduction
- Sequent Calculus

Topic 5.5

Problems

Practice

Exercise 5.9

Use both tableaux and resolution to prove the following theorems

1. $(((p \Rightarrow q) \Rightarrow q) \Rightarrow q) \Rightarrow (p \Rightarrow q)$
2. $(p \Rightarrow (q \vee r)) \vee (r \Rightarrow \neg p)$

Use the full formal notations to write the proofs.

Dual

Exercise 5.10

Give a dual of Tableaux that proves validity of formulas.

Exercise 5.11

Give a dual of Resolution that proves validity of formulas.

Resolution rule

Exercise 5.12

Give a minimal example that require at least two application of the resolution rule on some clause to derive the empty clause.

End of Lecture 5