

Lecture 7: Tableau Proof System

Lecturer: Yi Li

1 Overview

In last lecture, we have defined consequence, the relationship between a proposition and a set of propositions, on side of semantics. we prove the validity of a proposition semantically, according to truth table. This lecture will consider how to prove the validity of a tautology. But the validity would be guaranteed in the next lecture.

2 Atomic tableaux

For a given proposition, we can guess it true or false. In order to show our guess, we just associate it with a sign T or F in front of it. A string like $F\alpha$ and $T\alpha$ is called *signed* proposition.

In preceding lectures, we have known that a compound proposition is constructed by some simpler ones and the truth valuation between them must obey truth table. Because of the Adequacy Theorem, all connective function can be represent by $\{\neg, \wedge, \vee\}$. For convenience, there are totally five connectives in proposition logic.

We first figure out 12 atomic tableaux according to truth table as shown in Figure 1. We just take $(\alpha \rightarrow \beta)$ as the example. If a truth valuation \mathcal{V} make $(\alpha \rightarrow \beta)$ true, we must have either α is false or β is true. Otherwise, we have α true or β false simultaneously. Vividly, we can use a branch to represent “or” and a chain to represent “both ... and ...”.

With these atomic tableaux, we can expand a given proposition with a sign following this way:

Definition 1 (Tableaux). *A finite tableau is a binary tree, labeled with signed propositions called entries, such that:*

1. All atomic tableaux are finite tableaux.
2. If τ is a finite tableau, P a path on τ , E an entry of τ occurring on P and $\iota\tau$ is obtained from τ by adjoining the unique atomic tableau with root entry E to τ at the end of the path P , then $\iota\tau$ is also a finite tableau.

If $\tau_0, \tau_1, \dots, \tau_n, \dots$ is a (finite or infinite) sequence of the finite tableaux such that, for each $n \geq 0$, τ_{n+1} is constructed from τ_n by an application of (2), then $\tau = \cup \tau_n$ is a tableau.

Example 1. A tableau with the signed proposition $F(((\alpha \rightarrow \beta) \vee (\gamma \vee \delta)) \wedge (\alpha \vee \beta))$.

In order to characterize a tableaux, we define the following terms:

Definition 2. Let τ be a tableau, P a path on τ and E an entry occurring on P .

TA	FA	$ \begin{array}{c} T(\alpha \wedge \beta) \\ \swarrow \quad \searrow \\ F\alpha \quad T\beta \end{array} $	$ \begin{array}{c} F(\alpha \wedge \beta) \\ \\ T\alpha \\ \\ F\beta \end{array} $
$ \begin{array}{c} T(\neg\alpha) \\ \\ F\alpha \end{array} $	$ \begin{array}{c} F(\neg\alpha) \\ \\ T\alpha \end{array} $	$ \begin{array}{c} F(\alpha \vee \beta) \\ \\ F\alpha \\ \\ F\beta \end{array} $	$ \begin{array}{c} T(\alpha \vee \beta) \\ \swarrow \quad \searrow \\ T\alpha \quad T\beta \end{array} $
$ \begin{array}{c} T(\alpha \rightarrow \beta) \\ \swarrow \quad \searrow \\ F\alpha \quad T\beta \end{array} $	$ \begin{array}{c} F(\alpha \rightarrow \beta) \\ \\ T\alpha \\ \\ F\beta \end{array} $	$ \begin{array}{c} T(\alpha \leftrightarrow \beta) \\ \swarrow \quad \searrow \\ T\alpha \quad F\alpha \\ \quad \quad \\ T\beta \quad F\beta \end{array} $	$ \begin{array}{c} F(\alpha \leftrightarrow \beta) \\ \swarrow \quad \searrow \\ T\alpha \quad F\alpha \\ \quad \quad \\ F\beta \quad T\beta \end{array} $

Figure 1: Atomic tableaux of propositional logic

1. E has been reduced on P if all the entries on one path through the atomic tableau with root E occur on P .
2. P is contradictory if, for some proposition α , $T\alpha$ and $F\alpha$ are both entries on P . P is finished if it is contradictory or every entry on P is reduced on P .
3. τ is finished if every path through τ is finished.
4. τ is contradictory if every path through τ is contradictory.

Based on these terms, we can define what is a proof in a way with tableaux. Actually, we just prove a proposition valid by contradiction, which means we just negate its every contrary side.

Definition 3. 1. A tableau proof of a proposition α is a contradictory tableau with root entry $F\alpha$. A proposition is tableau provable, written $\vdash \alpha$, if it has a tableau proof.

2. A tableau refutation for a proposition α is a contradictory tableau starting with $T\alpha$. A proposition is tableau refutable if it has a tableau refutation.

Of course, proof and refutation are dual to each other. Consider the following question, is refutation necessary in our proof system?

3 Complete systematic tableaux

As we already know, there are many possible ways to expand the tableaux, which we mean which entry is chosen and be reduced. If the proposition is complicated enough, the parsing tree would

be big and it means there many nodes to handle. The we just run into a problem, how can you guarantee that every entry in generated tableau are all reduced, which means the tableau is really finished.

As a tableau is at most a binary-branching tree and also be a partial ordered set. However, if we can make it order we can reduce node by node without omitting. Then we just assure a lexical graphical order, which is defined level by level and left to right on a given tableau. We just call this approach Complete Systematic Tableaux (CST) as the following.

Definition 4 (Complete systematic tableaux). *Let R be a signed proposition. We define the complete systematic tableau(CST) with root entry R by induction.*

1. *Let τ_0 be the unique atomic tableau with R at its root.*
2. *Assume that τ_m has been defined. Let n be the smallest level of τ_m and let E be the leftmost such entry of level n .*
3. *Let τ_{m+1} be the tableau gotten by adjoining the unique atomic tableau with root E to the end of every noncontradictory path of τ_m on which E is unreduced.*

The union of the sequence τ_m is our desired complete systematic tableau.

This method is tedious for us when the size of a proposition is small. However, it is proper for a program to reduce a given proposition with sign mechanically. CST can guarantee that no unreduced nodes are left for every entry is ordered with a number and CST just reduce them one by one according to that order.

We can check the procedure in detail. The entries are examined level by level and from left to right in the same level. Suppose CST is processing the entries on level n . The atomic tableau corresponding to the current unreduced entry must be adjoined to the entries on the level no less than n .

However, there are still some issue should be taken into consideration if we really think algorithmically. What's the time complexity for a node to be reduced and be adjoined in every noncontradictory path which pass that node?

4 Properties of CST

There are some good properties of CST, which will be introduced in the next class.

Theorem 5. *Every CST is finished.*

Proof. Reduce the E level by level and from left to right until that there is no E unreduced for any fixed level. Every entry has a number. Given an unreduced entry E_n , say n . If the current processing entry is m . After finite step, actually $n - m$ steps, it will be reduced. \square

Theorem 6. *If $\tau = \cup \tau_n$ is a contradictory tableau, then for some m , τ_m is a finite contradictory tableau. Thus, in particular, if a CST is a proof, it is a finite tableau.*

Proof. By König lemma. Because a contradictory path must be finite and infinite tree with finite branch must has a infinite path. It is contradictory. \square

In order to prove the finiteness of every CST, we first give the following definition.

Definition 7. We define $d(\alpha)$, the degree of a proposition α by induction.

1. if α is a propositional letter, then $d(\alpha) = 0$.
2. if α is $\neg\beta$, then $d(\alpha) = d(\beta) + 1$.
3. if α is $\beta \vee \gamma, \beta \wedge \gamma$, or $\beta \rightarrow \gamma$, then $d(\alpha) = d(\beta) + d(\gamma) + 1$.

The degree of a signed proposition $T\alpha$ or $F\alpha$ is the degree of α . If P is a path in a tableau τ , then $d(P)$ the degree of P is the sum of the degree of the signed propositions on P that are not reduced on P .

Theorem 8. Every CST is finite.

Proof. While path extending, an entry could be split into two branches. whatever, the degree of every path decrease for each reduction. So we have $d(P_{m+1}) < d(P_m)$ where P_{m+1} is an extension of P_m . \square

With this Theorem, we do know something wrong if we run into a tableau which can never be finished.

Exercises

1. Ex 4/ page 36
2. Ex 6/ page 36
3. Ex 9/ page 36
4. Prove the following tautology by tableau proof
 - (a) $(q \rightarrow r) \rightarrow ((\neg q \rightarrow \neg p) \rightarrow (p \rightarrow r))$
 - (b) $((p \wedge q) \rightarrow r) \leftrightarrow (p \rightarrow (q \rightarrow r))$
5. Prove or refute the following proposition.(If negative, find the counterexample)
 - (a) $((\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow \gamma)) \rightarrow (\alpha \rightarrow (\beta \rightarrow \gamma))$
 - (b) $(\alpha \rightarrow \beta) \rightarrow ((\beta \rightarrow \gamma) \rightarrow (\alpha \rightarrow \gamma))$