Semantic Tableau Proof System for First-Order Logic

Khair Eddin Sabri
McMaster University

Abstract

Semantic tableau is a proof system used to prove the validity of a formula, it can also be used to prove if a formula is a logic consequence of a set of formulas. Tableau is used in both propositional and predicate logic. In this report, it is shown how Tableau proof system can be used in predicate logic. It is also shown the proof of the soundness and completeness of this proof system.

Keywords: Semantic Tableau proof system, Predicate Logic, Soundness, Completeness

1. Introduction

A semantic tableau is a tree representing all the ways the conjunction of the formulas at the root can be true. We expand the formulas based on the structure of the compound formulas. This expansion forms a tree. If all branches in the tableau lead to a contradiction, then there is no way the conjunction of the formulas at the root can be true. A path of the tree represents the conjunction of the formulas along the path. Semantic tableaux was invented by E.W. Beth and J. Hintikka (1965).

A semantic tableau is a proof system used to:

1. Test a formula *A* for validity.

2. Test whether *B* is a logical consequence of $A_1 \ldots . A_k$.

3. Test $A_1 \ldots\ A_k$ for satisfiability.

Definition 1 A *signed formula* is an expression TX or FX, where X is an (unsigned) formula. Under a given valuation, a signed formula TX is called *true* if X is true, and *false* if X is false. Also, a signed formula FX is called *true* if X is false, and *false* if X is true [1].

Definition 2 A *signed tableau* is a rooted dyadic tree where each node carries a signed formula[2].

If τ is a signed tableau, an *immediate extension* of τ is a larger tableau τ' obtained by applying a tableau rule to a finite path of τ.

Definition 3 A path of a tableau is said to be *closed* if it contains a conjugate pair of formulas, i.e., a pair such as T*A*, F*A*. A path of a tableau is said to be *open* if it is not closed. A tableau is said to be *closed* if each of its paths is closed[2].

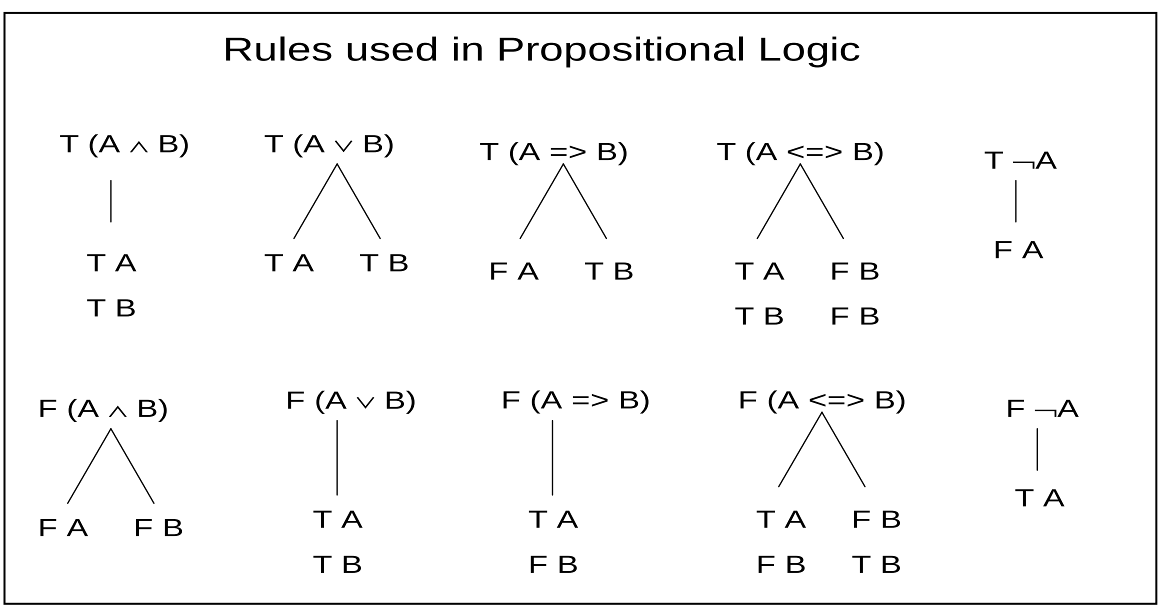
The tableau method:
We will see how tableau can be used to prove each of the mentioned formulae

1. To test a formula *A* for validity, form a signed tableau starting with FA. If the tableau closes off then *A* is logically valid.

2. To test whether *B* is a logical consequence of A_1 …. A_k, form a signed tableau starting with TA_1 . . . TA_k, F*B*, If the tableau closes off then *B* is indeed a logical consequence of A_1 … A_k.

3. To test A_1 …. A_k for satisfiability, form a signed tableau starting with TA_1. . .TA_k, If the tableau closes off then A_1 … A_k is not satisfiable. If the tableau does not close off then A_1 … A_k is satisfiable, and from any open path we can read off an assignment satisfying A_1 … A_k.

There are ten rules used to construct the signed tableau in the propositional logic as shown in the following figure[3].



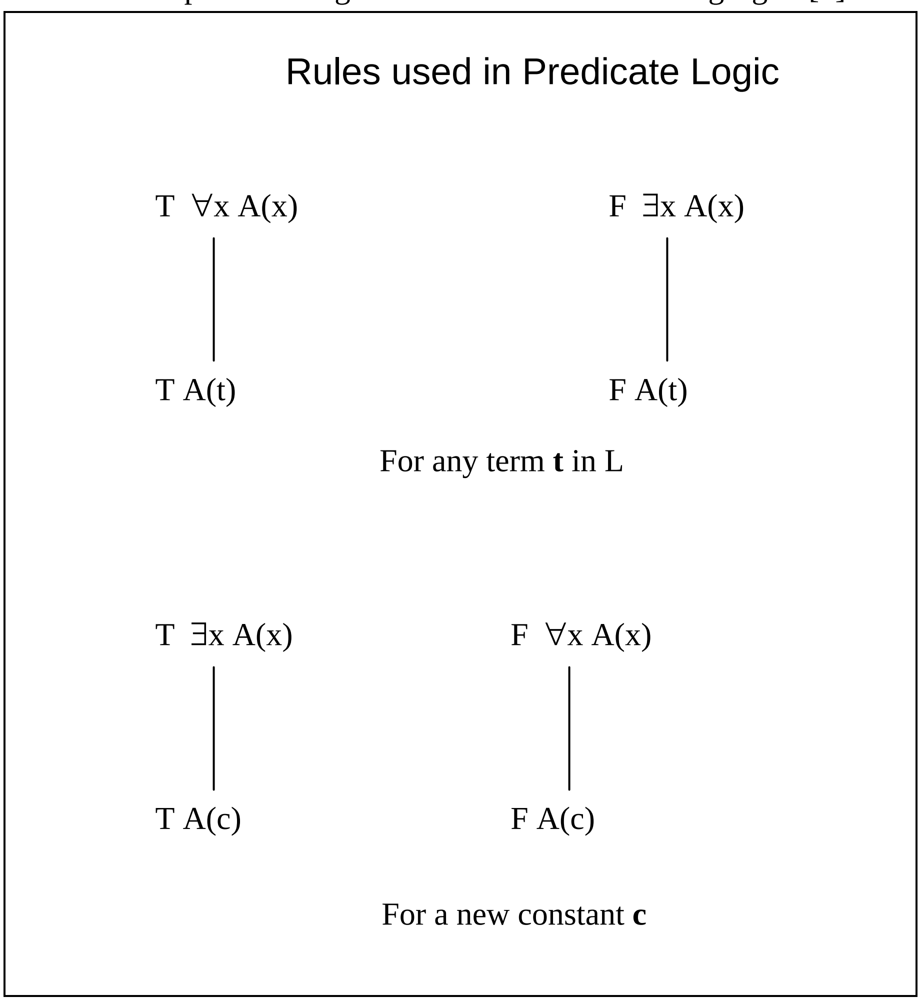
2. Predicate Logic

Tableau proof is used also in predicate logic by adding rules to cope with the universal and existential quantifiers.

Definition 4 Fix a countably infinite set *V* = {*a*1; *a*2; ….. ; *an*;……}. The elements of *V* will be called *parameters*. If *L* is a language, *L*-*V* -sentences will be called *sentences with parameters*[2].

Definition 5 A *signed tableau* is a rooted dyadic tree where each node carries a signed *L*-*V* -sentence. The tableau rules for predicate logic are the same as those for propositional logic, with additional rules[2].

The rules used in predicate logic are shown in the following figure[3]



The following example shows how tableau proof system is used in proving

$$F\ ((\exists x\ (P(x) \vee Q(x)) \Leftrightarrow ((\exists x\ P(x)) \vee \exists x\ Q(x))$$

$T\ \exists x\ (P(x) \vee Q(x)$

$F\ ((\exists x\ P(x)) \vee \exists x\ Q(x))$

$F\ \exists x\ (P(x))$

$F\ \exists x\ (Q(x))$

$T\ (P(a) \vee Q(a))$

$T\ P(a)$	$T\ Q(a)$
$F\ P(a)$	$F\ Q(a)$

$F\ \exists x\ (P(x) \vee Q(x))$

$T\ (\exists x\ P(x)) \vee \exists x\ Q(x)$

$T\ \exists x\ (P(x))$	$T\ \exists x\ (Q(x))$
$T\ P(p)$	$T\ Q(c)$
$F(P(p) \vee Q(b)$	$F(P(c) \vee Q(c))$
$F\ P(p)$	$F\ P(c)$
$F\ Q(b)$	$F\ Q(c)$

In this example all the branches are closed. Therefore, the original formula is valid

3. Decidability

While in propositional logic the tableau method could be used as decision procedure, this will certainly not work in first-order logic anymore. If a formula is not valid, the systematic method may lead to an infinite tableau. This is, however, not a deficiency of the tableau method. In fact, there is no correct and complete proof method for first-order logic that always terminates, as first-order logic is known to undecidable. Nevertheless in some cases, the tableau method can decide that a formula is invalid although the proof is not finished yet. Whenever we have constructed a branch μ that represents a Hintkka set (over the finite domain of the parameters that occur on μ), then we know that the origin FX of the tableau is satisfiable and hence X must be invalid. In these rare cases, the Hintikka branch gives us a *counterexample* for the validity of the formula [4].

Example

$$F\ (\forall x\ (P(x) \vee Q(x))) \Rightarrow ((\forall x\ P(x) \vee (\forall x\ Q(x)).$$

$$T\ \forall x\ (P(x) \vee Q(x))$$

$$F\ ((\forall x\ P(x) \vee (\forall x\ Q(x))$$

$$F\ \forall x\ P(x)$$

$$F\ \forall x\ Q(x)$$

$$F\ P(a)$$

$$F\ Q(b)$$

$$T\ (P(a) \vee Q(a))$$

$$T\ (P(b) \vee Q(b))$$

$T\ P(a)$	$T\ Q(a)$

$T\ P(b)$	$T\ Q(b)$

This tableau cannot be extended anymore in any meaningful way and has one open branch. We can assume a language with 2 elements in the domain U = {a,b}.and we can assign T to Q(a) and P(b) and F to Q(b) and P(a)

4. Soundness[1]

Definition 6

1. An *L-V -structure* consists of an *L*-structure *M* together with a mapping Φ: $V \rightarrow U_M$. If *A* is an *L-V* -sentence, we write

$$A^{\Phi} = A[a_1=\_(a_1 / \Phi(a_1),\ldots, a_k / \Phi(a_k)]$$

where $a_1;\ldots ; a_k$ are the parameters occurring in *A*. Note that A^{Φ} is an L-U_M-sentence. Note also that, if *A* is an *L*-sentence, then $A^{\Phi} = A$.

2. Let *S* be a finite or countable set of (signed or unsigned) *L-V* -sentences. An *L-V* -structure *M, Φ* is said to *satisfy S* if $v_M(A^{\Phi}) = T$ for all $A \in S$. *S* is said to be *satisfiable* if there exists an *L-V* -structure satisfying *S*.

3. Let τ be an *L*-tableau. We say that τ is *satisfiable* if at least one path of τ is satisfiable.

Lemma 1. Let τ and τ′ be tableaux such that τ′ is an *immediate extension* of τ, i.e., τ′ is obtained from τ by applying a tableau rule to a path of τ. If τ is satisfiable, then τ′ is satisfiable.

Proof. The proof consists of one case for each tableau rule. We consider some representative cases.

Case 1: Suppose that τ′ is obtained from τ′ by applying the rule

$$\begin{array}{c} \vdots \\ A \vee B \\ \vdots \\ / \quad \backslash \\ A \qquad B \end{array}$$

to the path θ in τ′. Since τ is satisfiable, it has at least one satisfiable path, S. If S $\neq$ θ, then S is a path of τ′, so τ′ is satisfiable. If S = θ, then θ is satisfiable, so let M and $\Phi : V \rightarrow U_M$ satisfy θ. In particular $v_M((A \vee B)^{\Phi}) = T$, so we have at least one of $v_M(A^{\Phi}) = T$ and $v_M(B^{\Phi}) = T$. Thus M and Φ satisfy at least one of θ, A and θ, B. Since these are paths of τ′, it follows that τ′ is satisfiable.

---

[1] The soundness and completeness proof are taken from the following reference:
S. Simpson, “Mathematical Logic”, http://www.math.psu.edu/simpson, 2004

Case 2: Suppose that τ' is obtained from τ by applying the rule

$$\begin{array}{c} \vdots \\ \forall x\, A \\ \vdots \\ | \\ A[x/a] \end{array}$$

to the path θ in τ, where a is a parameter. Since τ is satisfiable, it has at least one satisfiable path, S. If $S \neq \theta$, then S is a path of τ', so τ' is satisfiable. If $S = \theta$, then θ is satisfiable, so let M and $\Phi : V \rightarrow U_M$ satisfy θ. In particular $v_M(\forall x\ (A^{\Phi})) = v_M((\forall xA)^{\Phi}) = T$, so $v_M(A^{\Phi}[x/c]) = T$ for all $c \in U_M$. In particular

$$v_M(A[x/a]^{\Phi}) = v_M(A^{\Phi}[x/(a)]) = T.$$

Thus M and Φ satisfy θ, A[x/a]. Since this is a path of τ', it follows that τ' is satisfiable.

Case 3: Suppose that τ' is obtained from τ by applying the rule

$$\begin{array}{c} \vdots \\ \exists x\, A \\ \vdots \\ | \\ A[x/a] \end{array}$$

to the path θ in τ, where a is a new parameter. Since τ is satisfiable, it has at least one satisfiable path, S. If $S \neq \theta$, then S is a path of τ', so τ' is satisfiable. If $S = \theta$, then θ is satisfiable, so let M and $\Phi : V \rightarrow U_M$ satisfy θ. In particular $v_M(\exists x\ (A^{\Phi})) = v_M((\exists xA)^{\Phi}) = T$, so $v_M(A^{\Phi}[x/c]) = T$ for at least one $c \in U_M$. Fix such a c and define Φ': $V \rightarrow U_M$ by putting $\Phi'(a) = c$, and $\Phi'(b) = \Phi(b)$ for all $b \neq a$, $b \in V$. Since a is new, we have $B^{\Phi'} = B^{\Phi}$ for all $B \in \theta$, and $A^{\Phi'} = A^{\Phi}$, hence $A[x/a]^{\Phi'} = A^{\Phi'}[x/\Phi'(a)] = A^{\Phi}[x/c]$. Thus $v_M(B^{\Phi'}) = v_M(B^{\Phi}) = T$ for all $B \in \Phi$, and $v_M(A[x/a]^{\Phi'}) = v_M(A^{\Phi}[x/c]) = T$. Thus M and Φ' satisfy θ, A[x/a]. Since this is a path of τ', it follows that τ' is satisfiable.

The Soundness Theorem: Let $X_1,\ldots, X_k$ be a finite set of (signed or unsigned) sentences with parameters. If there exists a finite closed tableau starting with $X_1,\ldots,X_k$, then $X_1,\ldots,X_k$ is not satisfiable.

Proof. Let τ be a closed tableau starting with $X_1,\ldots,X_k$. Thus there is a finite sequence of tableaux $\tau_0; \tau_1,\ldots, \tau_n = \tau$ such that

$$\tau_0 = \begin{array}{c} X_1 \\ \vdots \\ X_k \end{array}$$

and each τ_{i+1} is an immediate extension of τ_i. Suppose $X_1,\ldots, X_k$ is satisfiable. Then τ_0 is satisfiable, and by induction on i using Lemma 1, it follows that all of the τ_i are satisfiable. In particular $\tau_n = \tau$ is satisfiable, but this is impossible since τ is closed.

5. Completeness

Let U be a nonempty set, and let S be a set of (signed or unsigned) L-U-sentences.

Definition 7: S is closed if S contains a conjugate pair of L-U-sentences. In other words, for some L-U-sentence A, S contains TA, FA.

Definition 8: S is U-replete if S "obeys the tableau rules" with respect to U. We list some representative clauses of the definition.

1. If S contains T $\neg$A, then S contains FA. If S contains F $\neg$A, then S contains TA. If S contains $\neg\neg$A, then S contains A.

2. If S contains TA&B, then S contains both TA and TB. If S contains FA&B, then S contains at least one of FA and FB.

3. If S contains T $\exists$xA, then S contains TA[x/a] for at least one $a \in U$. If S contains F $\exists$xA, then S contains FA[x/a] for all $a \in U$.

4. If S contains T $\forall$xA, then S contains TA[x/a] for all $a \in U$. If S contains F $\forall$xA, then S contains FA[x/a] for at least one $a \in U$.

Lemma 2 (Hintikka's Lemma). If *S* is *U*-replete and open, then *S* is satisfiable. In fact, *S* is satisfiable in the domain *U*.

Proof. Assume S is U-replete and open. We define an L-structure M by putting $U_M = U$ and, for each n-ary predicate P of L,

$$PM = \{(a_1,\ldots, a_n) \in U^n : T\, Pa_1\ldots a_n \in S\}$$

We claim that for all L-U-sentences A,

(a) if S contains TA, then $v_M(A) = T$
(b) if S contains FA, then $v_M(A) = F$

The claim is easily proved by induction on the degree of A. We give the proof for some representative cases.

1. deg(A) = 0. In this case A is atomic, say A = $Pa_1\dots a_n$.
   a) If S contains T $Pa_1\dots a_n$, then by definition of M we have $(a_1,\dots, a_n) \in P_M$, so $v_M(Pa_1\dots a_n) = T$.
   b) If S contains F $Pa_1\dots a_n$, then S does not contain T $Pa_1\dots a_n$ since S is open. Thus by definition of M we have $(a_1,\dots, a_n) \notin P_M$, so $v_M(Pa_1\dots a_n) = F$.

2. deg(A) > 0 and A = ¬B. Note that deg(B) < deg(A) so the inductive hypothesis applies to B.

3. deg(A) > 0 and A = B &C. Note that deg(B) and deg(C) are < deg(A) so the inductive hypothesis applies to B and C.
   (a) If S contains T B &C, then by repleteness of S we see that S contains both TB and TC. Hence by inductive hypothesis we have $v_M(B) = v_M(C) = T$. Hence $v_M(B \,\&C) = T$.
   (b) If S contains F B &C, then by repleteness of S we see that S contains at least one of FB and FC. Hence by inductive hypothesis we have at least one of $v_M(B) = F$ and $v_M(C) = F$. Hence $v_M(B \,\&C) = F$

4. deg(A) > 0 and A = ∃xB. Note that for all a ∈ U we have deg(B[x/a]) < deg(A), so the inductive hypothesis applies to B[x/a].

5. deg(A) > 0 and A = ∀xB. Note that for all a ∈ U we have deg(B[x/a]) <deg(A), so the inductive hypothesis applies to B[x/a].

We shall now use Hintikka's Lemma to prove the completeness of the tableau method. Let V = $\{a_1,\dots,a_n,\dots.\}$ be the set of parameters. Recall that a tableau is a tree whose nodes carry L-V -sentences.

Lemma 3. Let τ_0 be a finite tableau. By applying tableau rules, we can extend τ_0 to a (possibly infinite) tableau τ with the following properties: every closed path of τ is finite, and every open path of τ is V -replete.

Proof: The idea is to start with τ_0 and use tableau rules to construct a sequence of finite extensions $\tau_0, \tau_1,\dots, \tau_I,\dots$. If some τ_i is closed, then the construction halts, i.e., $\tau_j = \tau_i$ for all $j \geq i$, and we set $\tau = \tau_i$. In any case, we set $\tau = \tau_\infty = \bigcup_{i=0}^{\infty}\tau_i$. In the course of the construction, we apply tableau rules systematically to ensure that τ_1 will have the desired properties, using the fact that V = $\{a_1, a_2,\dots, a_n,\dots\}$ is countably infinite.

Here are the details of the construction. Call a node X of τ_i quasiuniversal if it is of the form T∀xA or F ∃xA or 8∀A or ¬∃xA. Our construction begins with τ_0. Suppose we have constructed τ_{2i}. For each quasiuniversal node X of τ_{2i} and each $n \leq 2i$, apply the appropriate tableau rule to extend each open path of τ_{2i} containing X by TA[x/a_n] or

FA[x/a_n] or A[x/a_n] or ¬A[x/a_n] as the case may be. Let τ_{2i+1} be the finite tableau so obtained. Next, for each non-quasiuniversal node X of τ_{2i+1}, extend each open path containing X by applying the appropriate tableau rule. Again, let τ_{2i+2} be the finite tableau so obtained.

In this construction, a closed path is never extended, so all closed paths of τ_∞ are finite. In addition, the construction ensures that each open path of τ_∞ is V-replete. Thus τ_∞ has the desired properties. This proves our lemma.

The Completeness Theorem: Let $X_1,\ldots,X_k$ be a finite set of (signed or unsigned) sentences with parameters. If $X_1,\ldots,X_k$ is not satisfiable, then there exists a finite closed tableau starting with $X_1,\ldots,X_k$. If $X_1,\ldots,X_k$ is satisfiable, then $X_1,\ldots,X_k$ is satisfiable in the domain V.

Proof: By Lemma 3 there exists a (possibly infinite) tableau τ starting with $X_1,\ldots,X_k$ such that every closed path of τ is finite, and every open path of τ is V-replete. If τ is closed, then by Konig's Lemma, τ is finite. If τ is open, let S be an open path of τ. Then S is V-replete. By Hintikka's Lemma 2, S is satisfiable in V . Hence $X_1,\ldots,X_k$ is satisfiable in V .

6. Conclusion

1. Tableau proof system is an easy to used system for proving the validity of a formula.
2. Tableau proof system is not used only to prove the validity of a formula but can also be used to find a counterexample of formula that is not valid
3. Tableau proof is a Sound and Complete system.

References

[1] P. Selinger, “Analytic Tableaux”,2004, http://quasar.mathstat.uottawa.ca/~selinger/3361/

[2] S. Simpson, “Mathematical Logic”, http://www.math.psu.edu/simpson, 2004

[3] Nerode and R. Shore, “Logic for Applications”, Springer-Verlag, 1993

[4] C. Kreitz, “First-Order Tableaux (Completeness and Compactness)” http://www.cs.cornell.edu/Info/People/kreitz/Teaching/CS486/current.html