

Lecture 15: Soundness, Completeness and Compactness

Lecturer: Yi Li

1 Overview

In this lecture, we will prove the soundness and completeness of tableau proof system, as we did for proposition logic proof system. These two theorems are the fundamental theorems of the first order logic. You can think it as two generals Hen and Ha, who are always posted on the door traditionally. We hope that they can protect our home.

The proof of the theorems is the same as the one used in proposition logic but the technique is special on how to handle quantifiers. We can also use this technique to construct a model, a structure, which can satisfy all the sentences.

Furthermore, we discuss the model size, actually the cardinality of universe in structure. Then we proved Skolem-Löwenheim theorem and compactness theorem.

Finally, we can establish a connection between size of model and compactness theorem by introducing infinite new sentences. Actually, it is just an application of compactness theorem.

2 Soundness theorem

Once you are told about something. You may say it sounds reasonable. It means that you may think the fact you know may be true. How about the proof system. It's the role of soundness theorem, which guarantees the validity of our proof.

Gong Sunlong, a famous Chinese old philosopher, once made a very ridiculous argument that a white horse is not a horse. Will you agree with his view? If you can not be persuaded by him, can you find the flaw in his proof? In the point of view of Philosophy, Gong's argument just reflects the relation between the essence and the specifics.

However, we does not care about the view of philosophy here. His proof is our focus. As we know, we may know how to write a program and we have written so many correct programs. But we may still make mistakes and write wrong program. Sometimes, you may even doubt the compiler is wrong. And actually, compiler sometimes really generate wrong binary but your program is correct.

Similarly, we may make wrong proof. And furthermore, would you doubt the validity of our proof itself? Soundness is the very theorem to eliminate our doubts.

Basically, a proof is just a procedure. Especially, it is a very mechanical approach to prove that a sentence is a consequence of a set by using tableau proof. We just reduce and generate the nodes according to the rules: atomic tableaux and reduce rules.

In predicate language, we introduce predicate, quantifier, function, constant, and variable to enhance the expressing power. In order to accommodate the changes, several new tableaux are introduced. The soundness of the proof system looks broken because of these new ones. So we need to reprove the soundness of the new tableau proof system. However the proof of soundness theorem is very similar to the one in propositional logic. We need only pay attention to the case of quantifiers. Here, we mean how to establish the connection between syntax and semantics of tableaux related with quantifiers. Especially, the interpretations associated with all-style quantifier are very difficult to understand. Because we want to use information as little as possible to check all cases.

Lemma 1. *If $\tau = \cup \tau_n$ is a tableau from a set of sentences S with root $F\alpha$, then any \mathcal{L} -structure \mathcal{A} that is a model of $S \cup \{\neg\alpha\}$ can be extended to one agreeing with every entry on some path P through τ . (Recall that \mathcal{A} agrees with $T\alpha(F\alpha)$ if α is true(false) in \mathcal{A} .)*

Theorem 2 (Soundness). *If there is a tableau proof τ of α from S , then $S \models \alpha$.*

3 Completeness theorem

3.1 Proof

In our life, sometimes you know something is definitely right. But you may find it is hard to persuade others for you cannot give a reasonable argument. Is there any problem which is right but we can not give a proof? The answer is positive. Completeness theorem guarantees that there must exist a proof if it is right in first order logic. If you cannot prove it, it just means that you are unlucky and you need do better and harder work.

Theorem 3. *Suppose P is a noncontradictory path through a complete systematic tableau τ from S with root $F\alpha$. There is then a structure \mathcal{A} in which α is false and every sentence in S is true.*

1. *If $F\beta$ occurs on P , then β is false in \mathcal{A} .*
2. *If $T\beta$ occurs on P , then β is true in \mathcal{A} .*

Corollary 4. *For every sentence α and set of sentences S of \mathcal{L} , either*

1. *the CST from S with root $F\alpha$ is a tableau proof of α from S .*

or

2. *there is a noncontradictory branch through the complete systematic tableau that yields a structure in that α is false and every element of S is true.*

3.2 Model construction

Actually, the proof show us a way to construct a real model to show that a sentence is true or false, while the model is still abstract because we may not find a real example.

Example 1. Is the sentence $(\exists y(\neg R(y, y) \vee P(y, y)) \wedge \forall x R(x, x))$ invalid?

Solution. If a sentence φ is invalid, then $\neg\varphi$ must be valid. So we can have the tableau proof in figure 1.

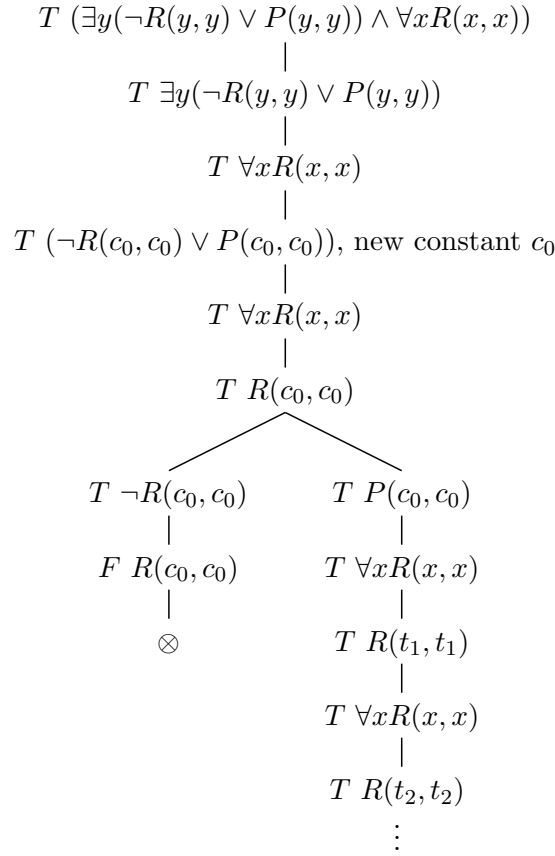


Figure 1: An example of tableau proof

The tableau is not contradictory. So the sentence is not invalid. We can construct a model along the non-contradictory path. Let domain $A = \{c_0, t_1, t_2\}$, $R = \{(c_0, c_0), (t_1, t_1), (t_2, t_2)\}$, $P = \{(c_0, c_0)\}$. This is a model of the sentence. \square

Here, we should be careful about the termination of a tableau proof. In practice, we always reduce \exists -style first if possible and try to leads contradiction by reducing \forall -style. The detail argument of when the proof is terminable. Please refer to the phrase in our text book on the page 115 below Figure 33.

4 Size of model

Given a language \mathcal{L} , we can construct a model \mathcal{A} (structure) which interprets constants, functions, and predicates. As we have shown that \mathcal{L} could have infinite constants, which means \mathcal{L} is infinite and \mathcal{A} is also infinite.

Definition 5 (size of a model). *The size of a model is the cardinality of the universe A in the structure \mathcal{A} .*

Example 2. *Let $\mathcal{A} = \langle \{c_0, c_1, \dots, c_n\}, \{P = \{ \langle c_0, c_0 \rangle \}, R = \{ \langle c_0, c_0 \rangle, \langle c_1, c_1 \rangle, \dots, \langle c_n, c_n \rangle \} \rangle$. It is easy to check it is a model of*

$$\alpha = (\exists y)(\neg R(y, y) \vee P(y, y)) \wedge (\forall x)R(x, x).$$

Here, we give a finite model. You can also construct a infinite model.

Example 3. *Suppose we have a language $\mathcal{L} = \langle \{P, \}, \{f(x, y)\}, \{c, d\} \rangle$. Given two sentences $(\forall x)P(c, x)$ and $(\forall x)(P(x, c) \rightarrow P(x, d))$.*

Remark. We know that the structure $\mathcal{A} = \langle \mathcal{N}, \{P = \leq\}, \{f(x, y)\}, \{c = 0, d = 2\} \rangle$ is a infinite model of them.

You can also find a finite model. □

4.1 Skolem-Löwenheim theorem

In previous case, we have only finite sentence. We are curious about the case that there are maybe infinite sentences. We have the following theorem.

Theorem 6. *If a countable set of sentences S is satisfiable, then it has a countable model.*

In the proof of completeness theorem, we construct the model by collecting all constants introduced by \exists -style node. As any sentence α could be taken as a sequence of symbols. It is obvious that a sentence could only contributes finite possible new constants.

Proof. Consider the tableau proof with the root $F(\alpha \wedge \neg\alpha)$. The tree is not finite otherwise S is unsatisfiable. It means there is a non-contradictory path. We can construct a model along that path. There are at most countable new constants. □

5 Compactness theorem

Theorem 7 (Completeness and Soundness).

1. α is a tableau provable from $S \Leftrightarrow \alpha$ is a logical consequence of S .
2. If we take α to be any contradiction such as $\beta \wedge \neg\beta$ in 1, we see that S is inconsistent if and only if S is unsatisfiable.

Theorem 8. *Let $S = \{\alpha_1, \alpha_2, \dots\}$ be a set of sentences of predicate logic. S is satisfiable if and only if every finite subset of S is satisfiable.*

Proof. Consider the tableau proof with the root $F(\alpha \wedge \neg\alpha)$. The tree should not be finite otherwise it is contradictory, which means $\alpha \wedge \neg\alpha$ could be proved from S and hence S is unsatisfiable.

However, if we have a tableau proof, it means that the tree only has finite path. We can follow the methods used in the proof of completeness theorem to construct a model which can only satisfy a set of sentences with a fixed number. It contradicts to our premises that every finite subset of S is satisfiable. \square

Theorem 9. *Let L be a first-order language. Any set S of sentences of L that is satisfiable in arbitrarily large finite models is satisfiable in some infinite model.*

Suppose S is satisfiable in arbitrary large finite models. Let R be a two-ary relation symbol that is not part of the language L , and enlarge L to L' by adding R .

We can modify the interpretation of R without affecting the truth values of members of S , since R does not occur in members of S . So we can write a sentence σ_n that asserts there are at least n different elements.

After we construct a new structure, which can satisfy all sentences. Now we can get rid of newly introduced sentences and predicates. It can be implied that this model can satisfy S .

Proof. We can introduce a new predicates $=$ and a set of new constants $\{c_1, c_2, \dots, c_i, \dots\}$. Now we construct a set of sentences $\Sigma = \{\sigma_i, i \geq 2\}$, where

$$\sigma_i = (\exists x) \left(\bigwedge_{k=1}^{i-1} \neg(x = c_k) \wedge \bigwedge_{1 \leq i < j \leq i-1} \neg(c_i = c_j) \right),$$

which means there are at least i elements in the domain.

Now consider the set of sentence $S \cup \Sigma$. If we can prove that its any finite subset of sentence is satisfiable we can prove that itself is satisfiable by applying Compactness Theorem. It is obvious that the size of model is infinite. Now we just eliminate all newly introduced constants, predicate, and sentences. We get a infinite model which also satisfies S .

Given any finite subset of $S' \cup \Sigma'$, Σ' implies there is a finite model. Premises guarantee that S' also has a model with the same number of elements. We just prove that any finite subset of $S \cup \Sigma$ is satisfiable. \square

6 Conclusion

In the proof, we use induction methods. Top-down approach is adopted for soundness theorem. For the base step, the root node satisfies. Given a noncontradictory path, we can prove the augmented path also satisfies if it satisfy. While bottom-up method is used in the proof of completeness theorem. Where we consider the simplest symbols first like constant and ground terms. Then we consider the atomic predicates and sentences with quantifiers and connectives.

Finally, we introduce the model theory of first order logic. Compactness Theorem and an application of it are given.

Exercises

Caution: In the question 4 and 5, we do not just consider a very special model. For the sentence or sentences you are given, you should consider all possible models, which means all possible interpretation of your predicates and functions.

1. 6/P126
2. 9/P126
3. 10/P126
4. 2/P125
5. 3/P125
6. 4/P125