MAU22C00 Lecture 11

John Stalker

Trinity College Dublin

Tableaux for FOL as non-deterministic computations

- The initial state has the statement whose validity we're trying to prove to the right of the line. There are variants for proving unsatisfiability, logical consequences, etc.
- The allowed actions are applying the tableau rules, ignoring ones which produce statements we've already generated (on that side of the line for that branch). This includes branching actions, where we follow both branches.
- The termination condition is that all branches have a contradiction or no further allowed actions.
- Successful termination means all branches have a contradiction.
- Unsuccessful means all branches have a contradiction or no further allowed actions, and at least one has no further allowed actions. This is very unusual. Usually the algorithm just fails to terminate.

If the algorithm *can* terminate successfully then the statement is valid (or unsatisfiable, or a logical consequences, depending on what we're doing).

Comments

This is the “tree of trees” scenario discussed last time. Each computational path produces a tableau, which has a tree structure. The set of all computational paths is also a tree, which each node being the partially completed tableau.

The number of possible actions in most states is infinite, since we have infinitely many parameters to replace a variable with. There are two ways to deal with this.

- Accept that some nodes in the program tree will have infinitely many children and use a hybrid tree traversal method.
- Note that all “new” parameters, i.e. ones which haven’t appeared on that branch are functionally equivalent and only allow using the one which is next in the queue. In this case there are finitely many options and we can do breadth first traversal.

Because the algorithm generally just runs forever in the unsuccessful case the validity of statements in first order logic is only semi-decidable. If a statement is valid we can prove that it is but if it’s not we generally can’t prove that it isn’t. In zeroeth order logic the question of whether a statement is a tautology was decidable.

Natural deduction for first order logic

We can treat first order logic with a traditional axiomatic system or a natural deduction system. I'll only discuss the natural deduction option.

We copy over all the rules of inference from zeroeth order logic. We've lost Boolean variables but only the rule of substitution referred to them.

There are no rules for atomic expressions, i.e. for predicates and their arguments.

There are four rules of inference for quantifiers. These correspond to our four tableau rules for quantifiers. Two are simple and two are more complicated.

One simple rule is that from a statement like $(\forall x.P)$ we can derive one like Q, where Q is the result of replacing all free occurrences of x in P with a.

Here x could be replaced by any variable, a by any parameter, and P by any Boolean expression.

The other simple rule is that from a statement like Q we can derive one like $(\exists x.P)$ where Q is the result of replacing all free occurrences of x in P with a.

Natural deduction, continued

Applying those rules in succession we can derive $(\exists x.P)$ from $(\forall x.P)$.

This is a reminder that first order logic can only be expected to work in settings where the domain is known to be non-empty.

There are logics without this restriction, but not in this module.

The more complicated rules involve creating a new scope, like the rule of fantasy.

One rule creates a new scope *without any corresponding hypothesis*. If we derive Q within that scope it allows us leave the scope and derive $(\forall x.P)$ outside of it, subject to some restrictions.

As previously, Q is the result of replacing all free occurrences of x in P with a.

The restrictions are that within the scope we are not allowed to use any statements from outside the scope in which the parameter a appears and that we can't replace variables from existentially quantified statements with that parameter.

Natural deduction, continued

The idea behind the rule is that *if a parameter is not special in any way* then anything you can prove about that parameter is true for everything.

This is a formal counterpart to informal arguments like "Suppose p is a prime number … so for all prime numbers p we have …"

The "not special in any way" requirement is similar to the problem we had with substitution in zeroeth order logic but there it was the variables appearing in the conclusion which needed special treatment and here it's the parameter appearing in the conclusion.

The restrictions on importing statements where a appears into the scope from outside and on using it to replace variables in existentially quantified statements are there to preserve this lack of specialness.

The idea of entering a new scope without introducing a hypothesis is a bit weird. There is really an implicit hypothesis though, namely that the domain is non-empty. It's just that first order logic makes that assumption implicitly.

Natural deduction, continued

The last quantifier rule is that after a statement of the form $\exists x.P$ we can introduce a hypothesis Q, derive a statement R within the scope of that hypothesis, discharge it, and keep the statement R, provided that the same restrictions on the use of a are observed and a does not appear in R.

If you find these rules confusing you are not alone. Published textbooks often get these wrong!

I find it easier to remember the principle than the rules: If you want to show something holds for all values of a variable then you're allowed to name an arbitrary object of the same type and prove that statement for that object, but it really has to be arbitrary. No further assumptions can be smuggled in.

In informal proofs this is often indicated by something like "Suppose Bob is a wombat. Then … So Bob has the property that … Bob was chosen arbitrarily, so all wombats have the property that …".

It's an error in such a proof to do anything which distinguishes Bob from other wombats.

A formal proof

Here's a proof of $[([\forall x.(fx)] \wedge \{\exists x.[(fx) \supset (gx)]\}) \supset \{\exists x.(gx)\}]$, which we saw an informal proof of and a tableau for earlier.

$$
\begin{array}{lll}
1 & . & ([\forall x.(fx)] \wedge \{\exists x.[(fx) \supset (gx)]\}) \\
2 & . & [\forall x.(fx)] \\
3 & . & \{\exists x.[(fx) \supset (gx)]\} \\
4 & . & .\quad [(fa) \supset (ga)] \\
5 & . & .\quad (fa) \\
6 & . & .\quad (ga) \\
7 & . & .\quad \{\exists x.(gx)\} \\
8 & . & \{\exists x.(gx)\} \\
9 & \multicolumn{2}{l}{[([\forall x.(fx)] \wedge \{\exists x.[(fx) \supset (gx)]\}) \supset \{\exists x.(gx)\}]}
\end{array}
$$

The first line is the rule of fantasy from ZOL, the second and third are the separation rule from ZOL, the fourth is the last of our quantifier rules, the fifth is the first of our quantifier rules, the sixth is modus ponens from ZOL, the seventh is our second quantifier rule, the eighth discharges the hypothesis from the fourth line, and the ninth discharges the hypothesis from the first line.

Formal proofs as non-deterministic computations

Formal proofs are non-deterministic computations

- The initial state is a blank page
- The set of possible actions is given by the axioms and the rules of inference. There may be infinitely may, if there are infinitely many axioms or if there are rules inference like substitution or fantasy. That's okay, as long as we can list the axioms and list the possible results of applying each rule of inference.
- The computation terminates successfully if we derive the given statement (in global scope, if scopes are a thing).
- The computation can't terminate unsuccessfully unless our formal system is quite bad, but it will often fail to terminate.

The given statement is a theorem if the computation *can* terminate successfully.

This applies to the given system for first order logic but also to any formal system.

Soundness, consistency, completeness

(Our formal system for) first order logic is *sound*, i.e. can only prove valid statements. This is about 80% obvious. There are no axioms and most of the rules of inference are obviously sound.

The non-obvious ones are substitution and the two restricted quantifier rules.

If you believe it's sound and you believe truth and falsehood are incompatible then it must be *consistent*, i.e. it can't be used to prove a contradiction.

It's also possible to prove consistency directly, without introducing the notion of truth. You do need to specify what negation looks like though.

It's also *complete*, i.e. it can prove any valid statement. This is definitely not obvious.

It's the last complete formal system we'll see in this module. Essentially everything more complicated than first order logic is incomplete.