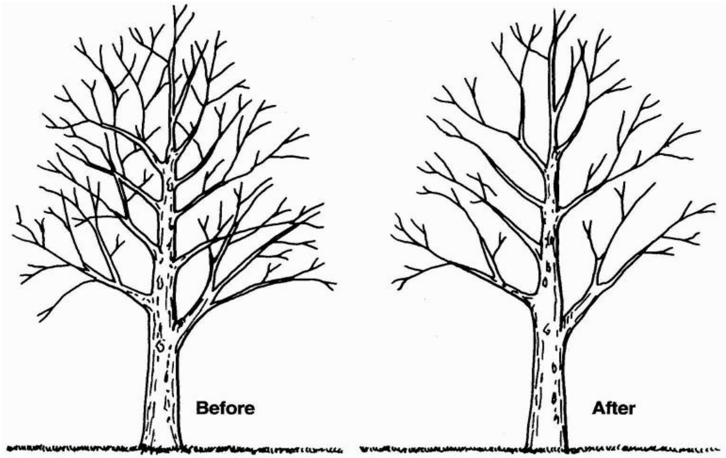
Tree Trimming



Marilynn Johnson
February 27, 2018

Phi Sigma Tau
The Philosophy Club

The Aim

In this talk today I aim to introduce you to two systems of logics and two rules of inference I proposed for a book by Graham Priest

This talk was originally presented to a logic audience but I hope that even those without a logic background will get the gist of some of the broad points

The bigger issues here are about the notion of existence and logical truth in general

The finer details are about two specific universal instantiation rules that I will present here

The Puzzle

Is the following proposition true? Why?

‘Barack Obama lives at 1600 Pennsylvania Avenue’

The Puzzle

Is the following proposition true? Why?

‘Sherlock Holmes lives at 221B Baker Street’

The Puzzle

Is the following proposition true? Why?

‘If superman is exposed to kryptonite then he will grow weak’

The Puzzle

‘If superman is exposed to kryptonite (and that kryptonite is not kryptonite-x) then superman will grow weak’

The Puzzle

‘If superman is exposed to kryptonite (and that kryptonite is not kryptonite-x) then superman will grow weak and if superman is weak then Lois Lane will not be rescued’

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

What I Will Argue

In *An Introduction to Non-Classical Logic: From If to Is* Graham Priest (2008) presents branching rules in Free Logic, Variable Domain Modal Logic, and Intuitionist Logic.

I propose:

1. a simpler, non-branching rule to replace Priest's rule for universal instantiation in Free Logic

2. a second, slightly modified version of this rule to replace Priest's rule for universal instantiation in Variable Domain Modal Logic

3. third and fourth rules, further modifying the second rule, to replace Priest's branching universal and particular instantiation rules in Intuitionist Logic (although I will not discuss these in this talk).

What I Will Argue

In each of these logics the proposed rule(s) lead(s) to tableaux with fewer branches.

In Intuitionist logic, the proposed rules allow for the resolution of a particular problem Priest is grappling with throughout the chapter.

I demonstrate that the proposed rules can greatly simplify tableaux and argue that they should be used in place of the rules given by Priest.

Outline

- I. Introduction
- II. Universal Instantiation in Free Logic

  Closed Tableaux
  Open Tableaux

- III. Universal Instantiation in Variable Domain Modal Logic

  Closed Tableaux
  Open Tableaux

I. Introduction

The tableau procedure is essentially a test to see whether or not the premises and negated conclusion of some inference leads to a contradiction (1.5).

If the premises and negated conclusion do lead to a contradiction this indicates that the inference is valid;

if the premises and negated conclusion do not lead to a contradiction this indicates that the inference is not valid.

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

I. Introduction

A tableau is complete when every rule that can be applied has been applied (1.4.5).

A branch of a tableau closes as soon as there is a contradiction, formulas A and ~A, on the branch.

Because subsequent steps will not undo the contradiction, the branch may be closed as soon as the contradiction appears, whether or not all the rules on that branch have been applied.

An ‘X’ at the bottom of a branch is used to indicate closure.

A tableau is closed when every branch on the tableau has been closed (1.4.6).

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

I. Introduction

Most of the rules of Classical Logic tableaux carry over to Free Logic tableaux. They are the following (1.4.4):

$$\begin{array}{ccc} & A \supset B & \\ \swarrow & & \searrow \\ \sim A & & B \end{array}$$

$$\begin{array}{c} \sim(A \supset B) \\ \downarrow \\ A \\ \downarrow \\ B \end{array}$$

$$\begin{array}{ccc} & A \vee B & \\ \swarrow & & \searrow \\ A & & B \end{array}$$

$$\begin{array}{c} \sim(A \vee B) \\ \downarrow \\ \sim A \\ \downarrow \\ \sim B \end{array}$$

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

I. Introduction

$$\begin{array}{ccc} & {\sim}(A \wedge B) & \\ \swarrow & & \searrow \\ {\sim}A & & {\sim}B \end{array}$$

$$\begin{array}{c} A \wedge B \\ \downarrow \\ A \\ \downarrow \\ B \end{array}$$

$$\begin{array}{ccc} & A \equiv B & \\ \swarrow & & \searrow \\ A & & {\sim}A \\ \downarrow & & \downarrow \\ B & & {\sim}B \end{array}$$

$$\begin{array}{ccc} & {\sim}(A \equiv B) & \\ \swarrow & & \searrow \\ A & & {\sim}A \\ \downarrow & & \downarrow \\ {\sim}B & & B \end{array}$$

$$\begin{array}{c} {\sim}{\sim}A \\ \downarrow \\ A \end{array}$$

I. Introduction

$$\begin{array}{c} {\sim}\forall xA \\ \downarrow \\ \exists x{\sim}A \end{array} \qquad\qquad \begin{array}{c} {\sim}\exists xA \\ \downarrow \\ \forall x{\sim}A \end{array}$$

I. Introduction

The systems of Free Logic dispense with the assumption of Classical Logic that every object in the domain exists.

This means that constants can refer to non-existent objects.

This allows us to distinguish between something "existing" in the sense of being within the domain, D, of objects we wish to reason over, and of *those* objects, "existence proper" in the sense of being within the sub-domain of objects, E, that we are to say exist.

I. Introduction

In Chapter 13 of *An Introduction to Non-Classical Logic: From If to Is* Graham Priest (2008) introduces the following rule for universal instantiation in Free Logic:

$$
\begin{array}{ccc}
 & \forall x A & \\
\swarrow & & \searrow \\
\sim \mathcal{E}a & & A_x(a)
\end{array}
$$

In this rule, a is any constant already on the branch; if there is no constant already on the branch, one is introduced.

In essence, each of the rules I introduce replaces a branching rule that indicates a quantifier applied to a predicate, A, is true when either ~Ɛa is true or A is true of all existent objects with a non-branching quantifier rule that is applied only when there is an existent object on the branch.

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

II. Free Logic

The systems of Free Logic dispense with the assumption of classical logic that every object in the domain exists.

To depict existence and nonexistence, an existence predicate, Ɛ, is introduced to the machinery of classical logic.

Ɛa denotes 'a exists'; ~Ɛa denotes 'a does not exist'.

Nonexistent objects can be thought of as fictions, such as Sherlock Holmes, or abstract objects, such as freedom.

II. Free Logic

The rules for free logic are the same as those of classical logic, except that the rules of Universal Instantiation and Particular Instantiation are modified to the following (13.10.2):

Universal Instantiation

$$\begin{array}{ccc} & \forall x A & \\ \swarrow & & \searrow \\ \sim \mathcal{E} a & & A_x(a) \end{array}$$

Particular Instantiation

$$\begin{array}{c} \exists x A \\ \downarrow \\ \mathcal{E} c \\ A_x(c) \end{array}$$

The Universal Instantiation Rule is applied for all constants, a, on the branch. If there is no constant on the branch when the rule is to be instantiated, one is introduced.

II. Free Logic

In cases where there is a constant, a, predicated by the existence predicate, Ɛ, on the branch before the Universal Instantiation Rule is applied, the left branch of the rule always closes because the Ɛa already on the branch contradicts with the introduced ~Ɛa.

In certain tableaux where this occurs several times, the left branch repeatedly is created only to be subsequently closed by the contradiction presented.

II. Closed Tableaux in Free Logic

A. Closed tableau completed using Priest's Universal Instantiation Rule:

$$\forall x(Px \supset Qx),\ \exists xPx \models \exists xQx$$

1. $\forall x(Px \supset Qx)$
2. $\exists xPx$
3. $\sim\exists xQx$
4. $\forall x\sim Qx$
5. $\mathcal{E}c$
6. Pc

↙ ↘

7. $\sim \mathcal{E}c$ $\sim Qc$

X   ↙ ↘

8. $\sim \mathcal{E}c$ $Pc \supset Qc$

X   ↙ ↘

9. $\sim Pc$ Qc

X   X

II. Closed Tableaux in Free Logic

I propose that a way to avoid such series of branches and closures is a simplification of the rule: only instantiate $\forall xA$ if there is already an existent object on the branch.

In reading an interpretation off the branch, if it is not explicitly stated in the tableau that an object exists, it is understood that no object exists.

The Proposed Universal Instantiation Rule can be stated as such:

$$\forall xA$$
$$\mathcal{E}a$$
$$\downarrow$$
$$A_x(a)$$

a is a constant that stands for an object that exists on the branch. In the absence of a constant predicated by $\mathcal{E}$ on the branch, $\forall xA$ is not instantiated.

II. Closed Tableaux in Free Logic

B. Closed tableau from above completed using Proposed Universal Instantiation Rule:

$$\forall x(Px \supset Qx), \exists xPx \models \exists xQx$$

1. $\forall x(Px \supset Qx)$
2. $\exists xPx$
3. $\sim\exists xQx$
4. $\forall x{\sim}Qx$
5. $\mathcal{E}a$
6. Pa
7. $\sim Qa$
8. $Pa \supset Qa$

$$\swarrow \quad \searrow$$

9. $\sim Pa$	Qa
X	X

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

II. Open Tableaux in Free Logic

C. Open tableau completed using Priest's Universal Instantiation Rule:

$$\forall x(Px \supset Qx),\ \forall x Px \models \exists x Qx$$

1. $\forall x(Px \supset Qx)$

2. $\forall x Px$

3. $\sim\exists x Qx$

4. $\forall x \sim Qx$

$\swarrow \quad \searrow$

5. $\sim \mathcal{E}a \qquad\qquad Pa$

$\swarrow \quad \searrow \qquad\qquad \swarrow \quad \searrow$

6. $\sim \mathcal{E}a \qquad \sim Qa \qquad \sim \mathcal{E}a \qquad \sim Qa$

$\swarrow \quad \searrow \qquad \swarrow \quad \searrow \qquad \swarrow \quad \searrow \qquad \swarrow \quad \searrow$

7. $\sim \mathcal{E}a \quad Pa \supset Qa \qquad \sim \mathcal{E}a \quad Pa \supset Qa \qquad \sim \mathcal{E}a \quad Pa \supset Qa \qquad \sim \mathcal{E}a \quad Pa \supset Qa$

$\uparrow \qquad \swarrow \quad \searrow \qquad \swarrow \quad \searrow \qquad \swarrow \quad \searrow \qquad \swarrow \quad \searrow$

8. $\sim Pa \quad Qa \qquad \sim Pa \quad Qa \qquad \sim Pa \quad Qa \qquad \sim Pa \quad Qa$

$\qquad\qquad\qquad X \qquad\qquad\qquad\qquad X$

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

II. Open Tableaux in Free Logic

A countermodel is read off an open branch of the tableau following the basic procedure of Classical First-Order Logic (12.4.8).

On the Priest model of reading a countermodel off a tableau, if a constant or predicate is not on the branch, it may be said to exist or not exist, to be satisfied or not satisfied, respectively. He calls this the "*don't care* condition" (12.4.8).

II. Open Tableaux in Free Logic

For the countermodel determined by the open branch of tableau C there is a constant, a, on the branch, so one need not be introduced. The countermodel generated by the tableau is: D = { δ_a }, E = { $\emptyset$ }, and v(a) = δ_a, v(P) = $\emptyset$, v(Q) = $\emptyset$. This can be depicted as:

Countermodel:	δ_a
$\mathcal{E}$	X
P	X
Q	X

II. Open Tableaux in Free Logic

D. Open tableau completed using Proposed Universal Instantiation Rule:

$$\forall x(Px \supset Qx), \forall xPx \mid= \exists xQx$$

1. $\forall x(Px \supset Qx)$
2. $\forall xPx$
3. $\sim\exists xQx$
4. $\forall x{\sim}Qx$

$\uparrow$

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

II. Open Tableaux in Free Logic

The countermodel is generated following a similar procedure as above, except that here the denotation of a is in the extension of E if and only if Ɛa is on the branch. On the model I propose here, predicates other than the existence predicate continue to fall within the “don't care” condition, but the existence predicate does not.

That is, an object is said to exist only if it follows the existence predicate on the open branch. If it is not explicitly stated on the branch that an object is an existent object, the set of existent objects is taken to be empty.

An arbitrary object, δ_a, is introduced so that the domain is non-empty. δ_a is not within the extension of Ɛ, P or Q.

Countermodel:	δ_a
Ɛ	X
P	X
Q	X

II. Tableaux in Free Logic

Tableau A (closed), completed with Priest's rule, has 9 lines and branches **3** times.

Tableau B (closed), completed with the proposed rule, has 9 lines and branches **1** time.

Tableau C (open), completed with Priest's rule, has 8 lines and branches **11** times.

Tableau D (open), completed with the proposed rule, has 4 lines and **does not** branch.

Tableau C and Tableau D arrived at the same countermodel.

As is demonstrated by this example executed two different ways, the proposed rule can simplify tableaux in Free Logic.

(I can go through the soundness & completeness proofs for any of the logics in the Q&A if people want to see them)

III. Variable Domain Modal Logic

With a slight modification, the proposed Universal Instantiation Rule for Free Logic may be carried over to Variable Domain Modal Logic.

Variable Domain Modal Logic is Free Logic supplemented with the possibility and necessity operators from Constant Domain Modal Logic. As such, the notion of possible worlds is reflected in Priest's slight adaptation of his Free Logic Rules. A world is given in the rules as *i.*

Universal Instantiation

$$\begin{array}{ccc} & \forall x A, i & \\ \swarrow & & \searrow \\ \sim \mathcal{E}a, i & & A_x(a), i \end{array}$$

Particular Instantiation

$$\begin{array}{c} \exists x A, i \\ \downarrow \\ \mathcal{E}c, i \\ A_x(c), i \end{array}$$

III. Variable Domain Modal Logic

Application of the branching rule for Universal Instantiation in Variable Domain Modal Logic often leads to a series of branches and immediate closures if the object, a, is already on the branch for the world, *i,* similar to the branching demonstrated above in Free Logic tableaux A and C.

The difference is that in Free Logic, the left branch would close if the object were anywhere on the branch; whereas with Variable Domain Modal Logic, the left branch will only close if the object is on the branch *and* in the same world as that for which the rule is applied.

III. Closed Tableaux in Variable Domain Modal Logic

E. Closed tableau completed using Priest's Universal Instantiation Rule:

$$\Box\forall x\,(A \supset B) \supset (\Box\forall x A \supset \Box\forall x B)$$

1. $\sim(\Box\forall x\,(A \supset B) \supset (\Box\forall x A \supset \Box\forall x B)),\ 0$
2. $\Box\forall x\,(A \supset B),\ 0$
3. $\sim(\Box\forall x A \supset \Box\forall x B),\ 0$
4. $\Box\forall x A,\ 0$
5. $\sim\Box\forall x B,\ 0$
6. $\Diamond\sim\forall x B,\ 0$
7. $0r1$
8. $\sim\forall x B,\ 1$
9. $\exists x\sim B,\ 1$
10. $\varepsilon a,\ 1$
11. $\sim B_x(a),\ 1$
12. $\forall x A,\ 1$
13. $\forall x\,(A \supset B),\ 1$

↙ ↘

14. $\sim\ \varepsilon a,\ 1$ X | $A_x(a),\ 1$

↙ ↘

15. $\sim\ \varepsilon a,\ 1$ X | $A_x(a) \supset B_x(a),\ 1$

↙ ↘

16. $\sim A_x(a),\ 1$ X | $B_x(a),\ 1$ X

III. Closed Tableaux in Variable Domain Modal Logic

Echoing the Free Logic Rule presented above, I propose the following rule for Universal Instantiation in Variable Domain Modal Logic:

Universal Instantiation

$\forall x A,\ i$

$\mathcal{E}a,\ i$

$\downarrow$

$A_x(a),\ i$

III. Closed Tableaux in Variable Domain Modal Logic

F. Closed tableau completed using Proposed Universal Instantiation Rule:

$$\Box\forall x\,(A \supset B) \supset (\Box\forall xA \supset \Box\forall xB)$$

1. $\sim(\Box\forall x\,(A \supset B) \supset (\Box\forall xA \supset \Box\forall xB)),\ 0$
2. $\Box\forall x\,(A \supset B),\ 0$
3. $\sim(\Box\forall xA \supset \Box\forall xB),\ 0$
4. $\Box\forall xA,\ 0$
5. $\sim\Box\forall xB,\ 0$
6. $\Diamond\sim\forall xB,\ 0$
7. $0r1$
8. $\sim\forall xB,\ 1$
9. $\exists x\sim B,\ 1$
10. $\mathcal{E}a,\ 1$
11. $\sim B_x(a),\ 1$
12. $\forall xA,\ 1$
13. $\forall x\,(A \supset B),\ 1$
14. $A_x(a),\ 1$
15. $A_x(a) \supset B_x(a),\ 1$

↙ ↘

16. $\sim A_x(a),\ 1$ X | $B_x(a),\ 1$ X

THE GRADUATE CENTER CITY UNIVERSITY OF NEW YORK

III. Open Tableaux in Variable Domain Modal Logic

I will now apply Priest's Universal Instantiation Rule and the Proposed Universal Instantiation Rule to a formula that generates a countermodel in Variable Domain Modal Logic.

This example is chosen because it demonstrates how the modification to the Free Logic Rule affects the application of the rule here for Variable Domain Modal Logic.

III. Open Tableaux in Variable Domain Modal Logic

G. Open tableau (of the Barcan Formula) completed using Priest's Universal Instantiation Rule:

$\forall x\Box Px \supset \Box\forall xPx$

1. $\sim(\forall x\Box Px \supset \Box\forall xPx), 0$
2. $\forall x\Box Px, 0$
3. $\sim\Box\forall xPx, 0$
4. $\Diamond\sim\forall xPx, 0$
5. $0r1$
6. $\sim\forall xPx, 1$
7. $\exists x\sim Px, 1$
8. $\mathcal{E}a, 1$
9. $\sim Pa, 1$

↙ ↘

10. $\sim\mathcal{E}a, 0$	$\Box Pa, 0$
↑	11. $Pa, 1$
	X

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

III. Open Tableaux in Variable Domain Modal Logic

A countermodel is read off an open branch of the tableau following the same basic procedure as above. In Variable Domain Modal Logic, an interpretation is the quadruple <D, W, R, *v*> where the domain, D is the non-empty domain, W is a non-empty set world, R is a binary accessibility relation on W, and *v* assigns every formula a truth value.

	δ_a					δ_a
ε	X	w_0	$\rightarrow$	w_1	ε	√
P	X				P	X

III. Open Tableaux in Variable Domain Modal Logic

H. Open tableau (of the Barcan Formula) completed using Proposed Universal Instantiation Rule:

$$\forall x \Box Px \supset \Box \forall x Px$$

1. $\sim(\forall x \Box Px \supset \Box \forall x Px), 0$
2. $\forall x \Box Px, 0$
3. $\sim\Box \forall x Px, 0$
4. $\Diamond\sim\forall x Px, 0$
5. $0 r 1$
6. $\sim\forall x Px, 1$
7. $\exists x \sim Px, 1$
8. $\mathcal{E}a, 1$
9. $\sim Pa, 1$

$\uparrow$

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

III. Open Tableaux in Variable Domain Modal Logic

Although the constant a is on the branch, the Proposed Universal Instantiation Rule is not applied because it is not in the same world as the universal quantifier. (The existent object is in world 1 but $\forall x \Box Px$ is at world 0.)

If there had been an existent object at world 0, line 2 could have been instantiated.

This aspect of the example highlights this difference between the rule I introduced above for Free Logic, and the rule I introduce here for Variable Domain Modal Logic.

III. Open Tableaux in Variable Domain Modal Logic

The countermodel is generated following the same procedure as above and may be verified in the same way.

	δ_a					δ_a
ε	X	w_0	$\longrightarrow$	w_1	ε	√
P	X				P	X

III. Tableaux in Variable Domain Modal Logic

Tableau E (closed), completed with Priest's rule, has 16 lines and branches **3** times.

Tableau F (closed), completed with the proposed rule, has 16 lines and branches **1** time.

Tableau G (open), completed with Priest's rule, has 11 lines and branches **1** time.

Tableau H (open), completed with the proposed rule, has 9 lines and **does not** branch.

Tableau G & Tableau H arrived at the same countermodel.

As is demonstrated by these examples executed two different ways, the proposed rule can greatly simplify tableaux in Variable Domain Modal Logic.

Discussion Questions

What makes things true in fiction?

Do we need two notions of existence?

Should we resist the conclusion that certain types of inference are valid in one logic and not another?

Does this undermine our notion of truth?

Should we prefer one system of logic?

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

End

Free Logic - Soundness

The Locality Lemma (13.7.2), Denotation Lemma (13.7.3), and Corollary (13.7.4) are unchanged.

Theorem: The tableaux of free logic are sound with respect to their semantics.

The proof is as in the classical case (12.8.5-12.8.7) and the particular instantiation case (13.7.5). The only difference in the Soundness Lemma is in the case of universal instantiation. For universal instantiation, we have the following:

In Free Logic, soundness is proven by demonstrating that, if we assume that on some interpretation, I, everything on the branch thus far is true, application of the rule in question maintains truth on at least one of the branches, on some interpretation I'. The proposed Universal Instantiation Rule is applied when two conditions are met: $\forall xA$ and Ɛa are both on the branch. Let's consider an interpretation, I, on which, by the Inductive Hypothesis, these two conditions are met. The truth of $\forall xA$ means that for every object in the inner domain (every existent object) that object has the property A. That is, for all d Є E Axkd is true. Let the constant *a* refer to *d*. Then since Ɛa is on the branch, *d* is in the inner domain. By the Denotation Lemma, Axkd is true in I if and only Axa is. So Axa is true in I. Thus, we can take I' to be I.

Free Logic - Completeness

Theorem: The tableaux of free logic are complete with respect to their semantics.

In Free Logic, completeness is proven by demonstrating that for every formula, A, on an open branch, B

If A is on B then v(A) = 1, and
If ~A is on B then v(A) = 0.

The open branch, B, induces an interpretation that is defined as follows. Let C be the set of all constants on B. There are two domains at play: D and E. Let D be the domain of B. D consists of every object that has been named on the branch. If D is empty we must introduce an arbitrary constant to the branch that denotes an arbitrary object. The second domain, E, is the inner domain or class of existent objects. E consists of only those objects on the branch which are named by some constant, c, that is preceded by the existence predicate, Ɛ. For the extension of the predicates, if it is not stated on the branch that some predicate extends to some object, then the object is not in the extension of that predicate.

The only cases that involve the modified rule of universal instantiation are as the cases in the induction for the truth of a universally quantified sentence, and falsity for an existentially quantified sentence. These are as follows.

Suppose that ∀xA is on the branch. We must show that ∀xA is true in the induced interpretation. Take some object in the domain, d. Call it c. If d Є E, Ɛc is on the branch. Since ∀xA and Ɛc are on the branch, Ac will also be on the branch. By the Inductive Hypothesis, v(Ac)=1. By the Denotation Lemma v(Axkd)=1.Thus v(∀xA) = 1.

Suppose that ~∃xAx is on the branch. This means that ∀x~A is also on the branch. Reasoning exactly as before, it follows that v(∀x~A)=1. And since ∀x~A is logically equivalent to ~∃xAx, v(~∃xA)=1.

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK

Variable Domain Modal Logic - Soundness

The Locality Lemma (15.9.3) and Denotation Lemma (15.9.4) are unchanged.

Theorem: The tableaux of variable domain K are sound with respect to their semantics.

The proof is as in (15.9.5). The only difference in the Soundness Lemma is in the case of universal instantiation. For universal instantiation, we have the following:

In Variable Domain Modal Logic, soundness is proven by demonstrating that, if we assume that on some interpretation, I, everything on the branch thus far is true, application of the rule in question maintains truth on at least one of the branches, on some interpretation, I'. Let f be a function that shows interpretation I to be faithful to the branch. Consider an application of the rule:

$$\begin{array}{l} \forall xA,\ i \\ \mathcal{E}a,\ i \\ \downarrow \\ A_x(a),\ I \end{array}$$

Assume that in I $\forall$xA is true at f(i) and Ɛa true at f(i). This means that $\forall$d ϵ D f(i) and Ax(kd) true at f(i). Let d be such that v(a)=v(kd). Because Ɛa true at f(i) this means d ϵ D f(i). This means Ax(kd) is true at f(i). By the Denotation Lemma (15.9.4) Ax(a) is true at f(i). Hence, I is faithful to the branch, and we can take I' to be I.

Variable Domain Modal Logic - Completeness

Theorem: The tableaux of variable domain K are complete with respect to their semantics. (The proof is as in 15.9.6 with a small modification.)

In variable domain K, completeness is proven by demonstrating that for every formula, A, on an open branch, B

If A, i is on B then $v_{w_i}(A) = 1$, and
If ~A, i is on B then $v_{w_i}(A) = 0$.

The open branch, B, induces an interpretation <D, W, R, v> that is defined as follows. Let C be the set of all constants on the branch. $W = \{w_i: i \text{ occurs on } B\}$. $w_i R w_j$ if and only if irj occurs on B. $D = \{\delta a: a \in C\}$ (or if C is empty, $D = \{\delta\}$, for some arbitrary δ). For all constants, a, on B, $v(a) = \delta a$. For every n-place predicate on B (including $\mathfrak{E}$), $\langle \delta a_1, \ldots, \delta a_n \rangle \in v_{w_i}$ if and only if $Pa_1 \ldots a_n$, i is on B. $D_{w_i} = v(w_i) = v_{w_i}(\mathfrak{E}) = \{\delta a: \mathfrak{E}a, i \text{ occurs on } B\}$. This means $\delta a \in D_{w_i}$ if and only if $\mathfrak{E}a$, i on the branch.

The only cases that involve the modified universal instantiation rule are the cases in the induction for the truth of the universally quantified sentence, and falsity for an existentially quantified sentence. These are as follows.

Suppose that $\forall xA$, *i* is on the branch. We must show that $\forall xA$, *i* is true in the induced interpretation. This means for $\forall d \in D_{w_i}$ $A_x(k_d)$ is true at w_i. Suppose that $d \in D_{w_i}$. Let c denote d. Because $d \in D_{w_i}$ this means that $\mathfrak{E}c$, *i*, is on B. So, we have applied the modified universal instantiation rule and $A_x(c)$, is on B. By Induction Hypothesis, $A_x(c)$ is true at w_i and $A_x(k_d)$ is true at w_i by the Denotation Lemma. Thus $v_{w_i}(\forall xA) = 1$.

Suppose that $\sim\exists xA$, *i* is on the branch. This means that $\forall x{\sim}A$, *i* is also on the branch. Reasoning exactly as before, it follows that $v_{w_i}(\forall x{\sim}A) = 1$. And since $\forall x{\sim}A$, *i* is logically equivalent to $\sim\exists xA$, *i* $v_{w_i}(\sim\exists xA) = 1$.

Intuitionist Logic - Soundness

Theorem: The tableaux of Intuitionist Logic are sound with respect to their semantics.

The proof is as in 20.9.5. The only differences in the Soundness Lemma are in case for truth in Universal Instantiation and case for falsity in Particular Instantiation.

In Intuitionist Logic, soundness is proven by demonstrating that, if we assume that on some interpretation, I, everything on the branch thus far is true, application of the rule in question maintains truth on at least one of the branches, on some interpretation, I'.

We have two rules to consider. For truth case of Universal Instantiation we have the following:
Let f be a function that shows interpretation I to be faithful to the branch. Consider an application of the rule:

$$
\begin{array}{ll}
\text{(i)} & \forall xA, +i \\
& irj \\
& \mathfrak{E}a, +j \\
& \downarrow \\
& A_x(a), +j
\end{array}
$$

Assume that in I $\forall$xA is true at f(i) and Ɛa true at f(j) when *f(i)Rf(j)*. This means that for $\forall$ d ∈ D f(j) Ax(kd) is true at f(j). Let d be such that v(a)=v(kd). Because Ɛa true at f(j) this means d ∈ D f(j). This means Ax(kd) is true at f(j). By the Denotation Lemma (20.9.3) Ax(a) is true at f(j). Hence, I is faithful to the branch, and we can take I' to be I.

For the falsity case of Particular Instantiation we have the following:
Let f be a function that shows interpretation I to be faithful to the branch. Consider an application of the rule:

$$
\begin{array}{ll}
\text{(ii)} & \exists xA, -i \\
& \mathfrak{E}a, +i \\
& \downarrow \\
& A_x(a), -i
\end{array}
$$

Assume that in I $\exists$xA is false at f(i) and Ɛa true at f(i). This means that for $\forall$ d ∈ D f(i) Ax(kd) is false at f(i). Let d be such that v(a)=v(kd). Because Ɛa true at f(i) this means d ∈ D f(i). This means Ax(kd) is false at f(i). By the Denotation Lemma (20.9.3) Ax(a) is false at f(i). Hence, I is faithful to the branch, and we can take I' to be I.

Intuitionist Logic - Completeness

Theorem: The tableaux of variable domain K are complete with respect to their semantics.

In Intuitionist Logic, completeness is proven by demonstrating that for every formula, A, on an open branch, B
If A, +i is on B then $vw_i(A) = 1$, and
If A, –i is on B then $vw_i(A) = 0$.

The open branch, B, induces an interpretation <D, W, R, v> that is defined as follows. Let C be the set of all constants on the branch. $W = \{w_i$: i occurs on B$\}$. w_iRw_j if and only if irj occurs on B. $D = \{\delta a: a \in C\}$. For all constants, a, on B, $v(a) = \delta a$. For every n-place predicate on B (including Ɛ), $<\delta a_1,\dots, \delta a_n> \in vw_i$ if and only if $Pa_1\dots a_n$, +i is on B. $Dw_i = v(w_i) = vw_i(\mathcal{E}) = \{\delta a$: Ɛa, i occurs on B$\}$. This means $\delta a \in Dw_i$ if and only if Ɛa, i on the branch.

The argument is as in 20.9.8. There are only two differences. The plus case for Universal Instantiation and minus case for Existential Instantiation.

For the first of these, we have the following: Suppose that $\forall xA$, +*i* is on the branch. We must show that $\forall xA$ is true at w_i in the induced interpretation. This means that for all j such that irj $\forall d \in Dw_j$ Ax(kd) is true at w_j. Suppose that $d \in Dw_j$. Let c denote d. Because $d \in Dw_j$ this means that Ɛc, +j, is on B. So, we have applied the modified universal instantiation rule and Ax(c) +j, is on B. By Induction Hypothesis, Ax(c) is true at w_j and Ax(kd) is true at w_j by the Denotation Lemma. Thus $vw_i(\forall xA) = 1$.

For the second case: Suppose that $\exists xA$, –*i* is on the branch. We must show that $\exists xA$ is false at w_i in the induced interpretation. This means that for every c such that $d \in Dw_i$ Ax(kd) is false at w_i. Suppose that $d \in Dw_i$. Let c denote d. Because $d \in Dw_i$ this means that Ɛc, +*i*, is on B. So, we have applied the modified particular instantiation rule and Ax(c) –*i*, is on B. By Induction Hypothesis, Ax(c) is false at w_i and Ax(kd) is false at w_i by the Denotation Lemma. Thus $vw_i(\exists xA) = 0$.

THE GRADUATE CENTER
CITY UNIVERSITY OF NEW YORK