

lecture 17, trees for MPL

phil1012 introductory logic

overview

this lecture

- an introduction to trees for MPL
- motivation for MPL truth trees
- motivation for new rules
- constructing MPL truth trees
- finished trees, closed paths, and saturation

learning outcomes

- after doing the relevant reading for this lecture, listening to the lecture, and attending the relevant tutorial, you will be able to:
 - explain the general rationale behind trees for MPL
 - explain the particular rationale behind each tree rule for MPL
 - explain what it means to say that a path is saturated
 - construct trees using the tree rules for MPL
 - use trees to test for various logical properties of MPL formulas
 - read off (counter)models from open paths of MPL trees

required reading

- section 10.1 of chapter 10

tables and trees

tables and trees

- recall that in PL we had two methods of proof: truth tables and truth trees
- in MPL we have only one: truth trees
- the tree rules for MPL are similar to those for PL
- but there are additional rules for the quantifiers
- and the rationale appeals to models
- let's start by running through the rationale for the rules

rationale for tree rules

rationale for tree rules

- the general rationale for the tree rules:
 - the rules prescribe propositions that must be true, given that what we have already written down is true

-
- in the non-branching case:

- if there is a model in which every proposition on some old path is true, then there is a model in which every proposition on the new path is true.

-
- in the branching case:
 - if there is a model in which every proposition on some old path is true, then there is a model in which every proposition on at least one new path is true.
-

- all of the rules from PL have the relevant property. Take the rules for negated disjunction (a non-branching rule) and disjunction (a branching rule) for example.

the rationale for the rule for negated disjunction

- rule for negated disjunction:

$$\begin{array}{l} \neg(\alpha \vee \beta) \checkmark \\ \neg\alpha \\ \neg\beta \end{array}$$

- rationale:
 - suppose there is some model in which $\neg(\alpha \vee \beta) \not\models (\alpha \vee \beta)$ is true
 - it follows by rule (2) that $(\alpha \vee \beta) \models (\alpha \vee \beta)$ is false on this model
 - it follows by rule (4) that $\alpha \models \alpha$ and $\beta \models \beta$ are both false on this model
 - it follows by rule (2) that $\neg\alpha \models \neg\alpha$ and $\neg\beta \models \neg\beta$ are both true on this model
 - so if there is a model in which every proposition on some old path is true *before* this rule is applied, then there is a model in which every proposition on the new path is true *after* this rule has been applied

the rationale for the rule for disjunction

- rule for disjunction:

$$\begin{array}{c} (\alpha \vee \beta) \checkmark \\ \swarrow \quad \searrow \\ \alpha \quad \beta \end{array}$$

- rationale:
 - suppose there is a model in which $(\alpha \vee \beta) \models (\alpha \vee \beta)$ is true
 - it follows by rule (4) that either $\alpha \models \alpha$ is true or $\beta \models \beta$ is true in this model
 - so if there is a model in which every proposition on some old path is true before this rule is applied, then there is a model in which every proposition on at least one new path is true after this rule has been applied

rules for quantifiers

rules for quantifiers

- we now introduce four new rules for the quantifiers.
 - the rule for negated existential quantifier.
 - the rule for negated universal quantifier.
 - the rule for existential quantifier.
 - the rule for universal quantifier.
- the application of the rules for the negated quantifiers is the same as for the other rules.
- the unnegated quantifiers require special treatment.

rule for negated existential quantifier

$$\neg \exists x \alpha(x) \checkmark$$
$$\forall x \neg \alpha(x)$$

- rationale:
 - suppose that $\neg \exists x \alpha(x) \not\equiv \exists x \neg \alpha(x)$ is true in some model \mathcal{M}
 - it follows from this that $\exists x \alpha(x) \equiv \exists x \neg \alpha(x)$ is false in \mathcal{M}
 - it follows from this that there is no model just like \mathcal{M} except that it also assigns a referent to a —where a is some name to which \mathcal{M} assigns no referent—in which $\alpha(a/x) \equiv \alpha(a/\underline{x})$ is true
 - in other words, $\alpha(a/x) \equiv \alpha(a/\underline{x})$ is false in every model just like \mathcal{M} except that it also assigns a referent to a
 - it follows from this that $\neg \alpha(a/x) \equiv \neg \alpha(a/\underline{x})$ is true on every model just like \mathcal{M} except that it also assigns a referent to a
 - it follows from this that $\forall x \neg \alpha(x) \equiv \forall x \neg \alpha(\underline{x})$ is true in \mathcal{M}

rule for negated universal quantifier

$$\neg \forall x \alpha(x) \checkmark$$
$$\exists x \neg \alpha(x)$$

- rationale:
 - suppose that $\neg \forall x \alpha(x) \not\equiv \exists x \neg \alpha(x)$ is true in some model \mathcal{M}
 - it follows from this that $\forall x \alpha(x) \equiv \exists x \neg \alpha(x)$ is false in \mathcal{M}
 - it follows from this that there is some model just like \mathcal{M} except that it also assigns a referent a —where a is some name to which \mathcal{M} assigns no referent—in which $\alpha(a/x) \equiv \alpha(a/\underline{x})$ is false
 - it follows from this that $\neg \alpha(a/x) \equiv \neg \alpha(a/\underline{x})$

\underline{x} is true in \mathcal{M} . It follows that $\exists \underline{x} \neg \alpha(\underline{x})$ is true in \mathcal{M} .

rule for existential quantifier

$$\frac{\exists x \alpha(\underline{x}) \quad \checkmark \underline{a}}{\alpha(\underline{a})}$$

- note 1: at the time of applying the rule at the bottom of some path, the name \underline{a} used in applying the rule must be one that has not yet appeared anywhere on the path.
- note 2: when applying the rule and checking off the formula, we write the name we have used in applying the rule next to the check mark.
- let's look at a couple of examples . . .

- suppose we have:

$$\begin{array}{l} Fa \\ Gb \\ \exists x Hx \end{array}$$

- the following is a correct application of the rule for the existential quantifier:

$$\begin{array}{l} Fa \\ Gb \\ \exists x Hx \quad \checkmark c \\ Hc \end{array}$$

- the name c has not been used on the path

- the following is an incorrect application of the rule for the existential quantifier:

$$\begin{array}{l} Fa \\ Gb \\ \exists x Hx \quad \checkmark b \\ Hb \end{array}$$

- the name b has already been used on the path

- okay, now consider another example

- suppose we have:

$$\begin{array}{c} (\exists x Fx \vee \exists x Gx) \quad \checkmark \\ \swarrow \quad \searrow \\ \exists x Fx \quad \exists x Gx \end{array}$$

- we apply the rule for the existential quantifier once:

$$\begin{array}{c}
 (\exists x Fx \vee \exists x Gx) \checkmark \\
 \swarrow \quad \searrow \\
 \exists x Fx \checkmark a \quad \exists x Gx \\
 Fa
 \end{array}$$

- we apply the rule again:

$$\begin{array}{c}
 (\exists x Fx \vee \exists x Gx) \checkmark \\
 \swarrow \quad \searrow \\
 \exists x Fx \checkmark a \quad \exists x Gx \checkmark a \\
 Fa \qquad \qquad Ga
 \end{array}$$

- this is okay because the name has not been used *on this path*.

- okay, now let's look at the rationale for the rule for the existential quantifier

- rationale:
 - suppose there is a model \mathcal{M} in which $\exists x \alpha(x)$ is true.
 - it follows from this that there is at least one object o in the domain of \mathcal{M} such that $\alpha(a/x)$ is true in $\mathcal{M}^{\underline{a}_o}$, where \underline{a} is some name not assigned a referent in \mathcal{M} , and $\mathcal{M}^{\underline{a}_o}$ is a model that is just like \mathcal{M} except that in it the name \underline{a} is assigned the referent o .
 - so, if there is a model \mathcal{M} in which $\exists x \alpha(x)$ is true, then there is a different model $\mathcal{M}^{\underline{a}_o}$ in which $\alpha(a/x)$ is true $\exists x \alpha(x)$ will be true on $\mathcal{M}^{\underline{a}_o}$.
 - so, if there is a model \mathcal{M} in which $\exists x \alpha(x)$ is true, then there is a different model $\mathcal{M}^{\underline{a}_o}$ in which $\alpha(a/x)$ is true, and $\exists x \alpha(x)$ is true.

rule for universal quantifier:

$$\begin{array}{c}
 \forall x \alpha(x) \backslash \underline{a} \\
 \alpha(\underline{a}/x)
 \end{array}$$

- note 1: the name \underline{a} does not have to be new on the path
- note 2: when applying the rule, we write a backslash, not a check mark—and we write the name used in applying the rule next to the backslash

- note 3: we can apply the rule multiple times

$$\begin{aligned} & \forall x \alpha(x) \quad \underline{a}, \underline{b}, \underline{c} \\ & \alpha(\underline{a}/\underline{x}) \\ & \alpha(\underline{b}/\underline{x}) \\ & \alpha(\underline{c}/\underline{x}) \end{aligned}$$

-
- okay, let's go over the rationale for the rule for the universal quantifier
-

- rationale (case 1):
 - suppose that the name \underline{a} used in applying the rule is new to the path.
 - if there is a model \mathcal{M} in which $\forall x \alpha(x)$ is true, then for every object oo in the domain of \mathcal{M} , $\alpha(\underline{a}/\underline{x})$ is true in $\mathcal{M}^{\underline{a}_o}$ where \underline{a} is our new name (which is not assigned a referent in \mathcal{M}), and $\mathcal{M}^{\underline{a}_o}$ is a model that is just like \mathcal{M} except that in it the name \underline{a} is assigned the referent oo
 - if $\forall x \alpha(x)$ is true in \mathcal{M} , then $\forall x \alpha(x)$ is true in $\mathcal{M}^{\underline{a}_o}$.
 - it follows that if there is a model in which $\forall x \alpha(x)$ is true, then there is a model on which $\forall x \alpha(x)$ is true and $\alpha(\underline{a}/\underline{x})$ is true.
-

- rationale (case 2):
 - suppose that the name \underline{a} used in applying the rule is not new to the path.
 - if there is a model \mathcal{M} in which $\forall x \alpha(x)$ is true, then for every object oo in the domain of \mathcal{M} , $\alpha(\underline{d}/\underline{x})$ is true in $\mathcal{M}^{\underline{d}_o}$ where \underline{d} is a new name (which is not assigned a referent in \mathcal{M}), and $\mathcal{M}^{\underline{d}_o}$ is a model that is just like \mathcal{M} except that in it the name \underline{d} is assigned the referent oo
 - in this case, \underline{a} is (already) assigned to a referent in \mathcal{M} —suppose it is assigned to the object k
 - we have just seen that $\alpha(\underline{d}/\underline{x})$ is true in every model just like \mathcal{M} except that it assigns a referent to \underline{d}
 - so $\alpha(\underline{d}/\underline{x})$ is true in $\mathcal{M}^{\underline{d}_k}$, the model that assigns \underline{d} 's referent to k
 - but then $\alpha(\underline{a}/\underline{x})$ must be true in \mathcal{M} , because \underline{a} 's referent is the same object k
-

- okay, so much for the motivation for the rules for the existential

quantifier and the universal quantifier, now let's take a look at trees for MPL in their entirety

tree rules for MPL

tree rules for MPL

- the tree rules for MPL consist of:
 - the tree rules from PL.
 - tree rules for disjunction, negated disjunction, conjunction, negated conjunction, conditional, negated conditional, biconditional, negated biconditional, and double negation are the same as in PL
- and the tree rules for the quantifiers:

-
- rule for existential quantifier

$$\frac{\exists x \alpha(x) \quad \checkmark \quad \text{(new } \underline{a})}{\alpha(\underline{a}/x)}$$

-
- rule for negated existential quantifier

$$\frac{\neg \exists x \alpha(x) \quad \checkmark}{\forall x \neg \alpha(x)}$$

-
- rule for universal quantifier

$$\frac{\forall x \alpha(x) \quad \backslash \underline{a} \quad \text{(any } \underline{a})}{\alpha(\underline{a}/x)}$$

-
- rule for negated universal quantifier

$$\frac{\neg \forall x \alpha(x) \quad \checkmark}{\exists x \neg \alpha(x)}$$

recommended order of application

- there's a recommended order of application for the rules:
 - rules from PL. non-branching first
 - rules for negated quantifiers
 - rule for (unnegated) existential quantifier
 - rule for (unnegated) universal quantifier

-
- that's it. those are the four new rules, and notes on their application.
 - now we've got to say something about finished trees

finished trees, saturated paths

finished trees, saturated paths

- because we can go on applying the rule for the universal quantifier as many times as we like, we need to say something about when our trees are finished
- to do so we appeal to the idea of a saturated path
- a tree is **finished** if each of its paths is either **closed** or **saturated**.

- a path is **saturated** if and only if
 - every formula on it—apart from atomic formulas, negations of atomic formulas, and formulas whose main operator is a universal quantifier—has had the relevant rule applied; and
 - every formula on it whose main operator is a universal quantifier
 - has had the universal quantifier rule applied to it at least once, and
 - has had the rule applied to it once for each name that appears on the path.

- let's consider an example

- suppose we have:

$$\begin{array}{l} \exists xFx \\ \forall xHx \\ Gb \end{array}$$

- we apply the existential quantifier rule to get:

$$\begin{array}{l} \exists xFx \checkmark a \\ \forall xHx \\ Gb \\ Fa \end{array}$$

- then we apply the universal quantifier rule to get:

$$\begin{array}{l} \exists xFx \checkmark a \\ \forall xHx \setminus a \\ Gb \\ Fa \\ Ha \end{array}$$

- this tree is unfinished!
- why? because there is a name on the path, bb in this case, which we have not yet used in applying the universal quantifier rule

- so we apply the universal quantifier rule using the name bb to get:

$$\begin{array}{l} \exists xFx \checkmark a \\ \forall xHx \setminus a, b \\ Gb \\ Fa \\ Ha \\ Hb \end{array}$$

- this tree is finished!
 - why? because we've applied the universal quantifier rule at least once, and we have applied it using every name on the path
-

- don't worry if you haven't fully understood the idea of saturating a path
- we'll do plenty of practice in the live lecture and in the tutorials

wrapping up

this lecture

- introducing the tree rules for MPL
- the rationale for the rules
- finished trees, saturated paths

next lecture

- lecture 18, uses of trees for MPL