PHIL1012 Lecture 15: Trees for MPL, Pt. 1

Trees!

Trees test for satisfiability.

- Is there a model in which all of the propositions at the top of the tree are true?

We can test for other logical concepts by reducing them to satisfiability.

<u>Example</u> An argument is invalid iff the set containing its premises and the negation of its conclusion is satisfiable.

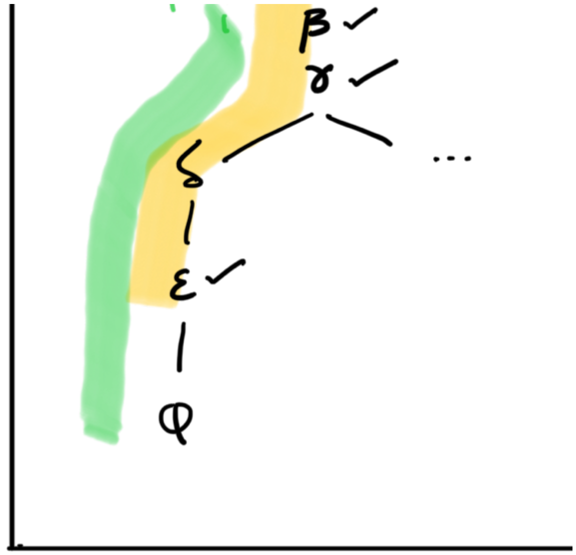
Why do trees work?

The basic idea is that (non-branching) rules have the following desired property ...

Suppose we are extending a branch p by applying a non-branching rule. This will result in a longer, extended branch p'.

Desired property

If there is a model on which every proposition on p is true, then there is a model in which every proposition on p' is true.

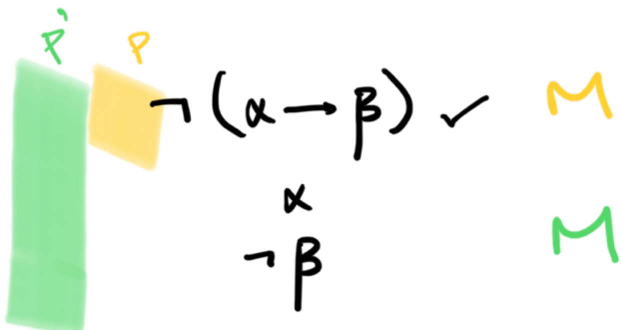


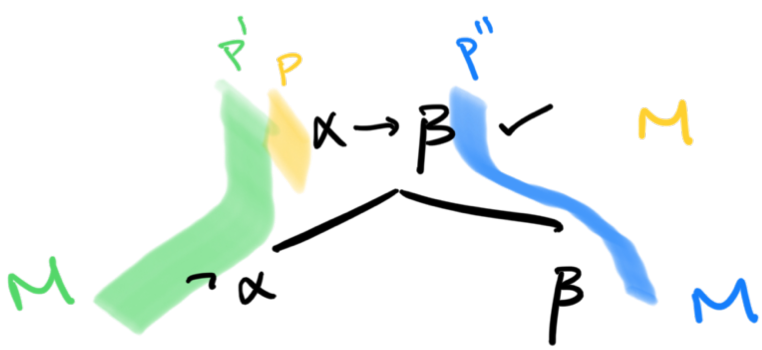
Similarly, for branching rules:

Suppose we are branching p, creating two extended paths p' and p''.

If there is a model in which every proposition on p is true, then either there is a model in which every proposition on p' is true or there is a model in which every proposition on p'' is true.

Examples





New tree rules

① $\neg \exists \underline{x}\, \alpha(\underline{x})$ ✓ M

$\forall \underline{x}\, \neg \alpha(\underline{x})$ M

$\neg \exists x\, Fx$ ✓ M

$\forall x\, \neg Fx$ M

- Suppose $\neg \exists \underline{x}\, \alpha(\underline{x})$ is true in M.
- Then $\exists \underline{x}\, \alpha(\underline{x})$ is false in M.

This means: there is no model like M that assigns a referent to new $\underline{a}$ and makes $\alpha(\underline{a}/\underline{x})$ true.

In other words: $\neg\alpha(a/x)$ is true in every M^{a} i.e.

In other words: $\neg\alpha(\underline{a}/\underline{x})$ is true in every M_a, i.e.

$\forall \underline{x} \neg \alpha(\underline{x})$ is true in M.

② $\neg \forall \underline{x}\, \alpha(\underline{x})$ ✓

$\exists \underline{x} \neg \alpha(\underline{x})$

$\neg \forall x\, A x$ ✓

$\exists x \neg A x$

③ $\exists \underline{x}\, \alpha(\underline{x})$ ✓ $\underline{a}$

$\alpha(\underline{a}/\underline{x})$

where $\underline{a}$ is a name that is new to the path

Examples

① $\exists x\, Gx$ ✓a

Ga

② $\exists x\, Gx$ ✓b

Gb

③ $\exists x\,(Gx \wedge Fx)$ ✓d

$Gd \wedge Fd$

④ $\exists x\,(Gx \wedge Fx)$ ✓c

$Gc \wedge Fx$ WRONG!

⑤ $\exists x\,(Da \vee Gx)$ ✓ b

$Da \vee Gb$

⑥ $\exists x\,(Db \vee Fx)$ ✓ c

$Db \vee Fc$ WRONG!

Why does it have the desired property?

$\exists \underline{x}\ \alpha(\underline{x})$ ✓ $\underline{a}$

$\alpha(\underline{a}/\underline{x})$

Why must we use a new name?

Consider...

p' $\quad$ p

Fa

$\exists x \neg Fx \checkmark a$

$\neg Fa$

$\times$

Note The name must be new to the path, not necessarily the entire tree, e.g.

$\exists x Fx \lor \exists x Gx \checkmark$

$\exists x Fx \checkmark a \qquad\qquad \exists x Gx \checkmark a$

$Fa \qquad\qquad Ga$

(4)

$$\forall \underline{x}\, \alpha(\underline{x}) \;\backslash\; \underline{a}$$
$$\alpha(\underline{a}/\underline{x})$$

where $\underline{a}$ can be any name at all.

Note No tick !

Examples

$$\forall x\, Gx \;\backslash\; a$$
$$Ga$$

$\forall x\, Gx \quad \backslash f$

Gf

$\forall x\, Gx \quad \backslash a b c d \ldots$

Ga

Gb

Gc

Gd

$\vdots$

$\forall x\, (Fx \to Gx) \quad \backslash a$

$Fa \to Ga$

$\forall x(Fx \vee Gb) \setminus b$ ✓

$Fb \vee Gb$

$\forall x(Fx \to Gx) \setminus ab$

$Fa \to Gb$ WRONG!

$\forall x(Fx \to Gx) \setminus ab$

$Fa \to Ga$ ✓

$Fb \to Gb$

$Fa \vee \forall x Gx \setminus b$

$Fa \vee Gb$ WRONG!

$Fa \vee \forall x Gx$ ✓

$\forall x Gx \setminus b$

Fa Gb

Why does it have the desired property?

$\forall \underline{x}\ \alpha(\underline{x}) \setminus \underline{a}$

$\alpha(\underline{a} / \underline{x})$

(Exercise!)

Finished trees and saturated paths

A tree is finished when every path on it is either closed or saturated.

α

$\neg\alpha$

x

Roughly, a path is saturated if all rules that can be applied to it have been applied.

But we need to limit this, e.g.

$$\begin{array}{l} \forall x\, Fx \quad \backslash abc \ldots \\ \hline Fa \\ Fb \\ Fc \\ \vdots \end{array}$$

If a path contains a universal proposition α, then the universal rule must be applied at least once to α, and it must be applied using every name on the path.

Example

$Fa \wedge Gb$ ✓

$\exists x\, Fx$ ✓c

$\forall x\, Gx$ \ abc

Fa

Gb

Fc

Ga

Gb

Gc

$\uparrow$

A path is saturated iff

(i) every wff on it — apart from (negations of) atomi[c]
and universals — has had the relevant rule applied
to it, and

(ii) every universal wff has had the universal rule
applied to it

- at least once, and

- once for each name on the path.

Heuristic Apply tree rules in this order:

- connectives
- negated quantifiers
- $\exists$
- $\forall$

Example

$\neg(Fa \rightarrow \neg\exists x(Fx \wedge Gx))$ ✓

$\neg\forall y\, Fy$ ✓

Fa

$\neg\neg\exists x(Fx \wedge Gx)$ ✓

$\exists x(Fx \wedge Gx)$ ✓ b

$\exists y\, \neg Fy$ ✓ c

$Fb \wedge Gb$ ✓

Fb

Gb