28/04/2006, Dr Marcus Rossberg, mr30@st-and.ac.uk
office hours: Thursdays, 10-12 (*Arché* building, College St, 19.4)

PY1003

Lecture 19

Revision I

(1) Trees and their uses

The tree method can be used to demonstrate a number of properties of sentences and relations between sentences:

What is tested:	How it is tested:
Consistency of a sentence	Tree has at least one open branch
Contradictory sentence	Tree has no open branches
Tautological sentence	Tree for the *negation* of the sentence has no open branch
Consistency of a *set* of sentences	Tree for all sentences has at least one open branch
Validity of an argument	Tree starting with all the premises and the *negation* of the conclusion has no open branches
Equivalence of two sentences	Tree for the negation of the bi-conditional between the two sentences has no open branches

(2) Truth tree rules for sentential connectives

$\neg\neg A$ ✓	$A \wedge B$ ✓	$A \vee B$ ✓	$A \rightarrow B$ ✓	$A \leftrightarrow B$ ✓
|	|	/ \	/ \	/ \
A	A	A B	$\neg A$ B	A $\neg A$
	B			B $\neg B$

A	$\neg(A \wedge B)$ ✓	$\neg(A \vee B)$ ✓	$\neg(A \rightarrow B)$ ✓	$\neg(A \leftrightarrow B)$ ✓
$\underline{\neg A}$	/ \	|	|	/ \
	$\neg A$ $\neg B$	$\neg A$	A	A $\neg A$
		$\neg B$	$\neg B$	$\neg B$ B

Remember:

- Tick a sentence after you have applied a rule for it
- Do not skip steps – apply one rule after the other
- A tree rule must be apply to *every* open branch on which the sentence appears
- Always apply tree rules to whole sentences, not to portions of them
- A tree must never fork into more than two branches
- It is usually of advantage to apply non-branching rules first
- A branch of tree closes only if that branch contains a sentence, and its negation as *isolated sentences*. It is not sufficient that the negation of a sentence present on the tree occurs embedded within another sentence.
- The sentence and its negation must occur on *the same branch*; otherwise the branch cannot be closed

(3) Interpretations, models and counterexamples for sentential (propositional) logic

- Within sentential (or propositional) logic, an *interpretation* of a set of sentences is an assignment of truth-values to the atomic sentences that occur within the sentences of the set.
- A *model* of a set of sentences is an interpretation that makes all of the sentences in that set true.
- A *counterexample* to an argument is an interpretation that makes the premises true and the conclusion false. Thus a counterexample for an argument is a model for the set of sentences consisting of the premises of the argument, and the negation of its conclusion.

We can use truth-trees to construct models and counterexamples.

Example:

```
       P → Q ✓
     ¬(R ∨ Q) ✓
   (P∨S) ∨ ¬Q ✓
         |
        ¬R
        ¬Q
        / \
     ¬P    Q
     / \
  P∨S   ¬Q
   / \
  P   S
      ☝
```

In order to construct a model using a tree, we select (arbitrarily) an open branch of the tree. If an atomic sentence occurs on that branch, we assign the sentence the truth-value True (or T). If the negation of an atomic sentence occurs on that branch, we assign the sentence the truth-value False (or F). If an atomic sentence occurs in the set of starting sentences, but not on the selected branch, we assign it arbitrarily one of the two truth-values (T or F).

I(P) = F
I(Q) = F
I(R) = F
I(S) = T

(4) Tree rules for predicate logic

Universal instantiation:

$\forall x\, Fx$ ✓a
|
Fa

(where ‘a’ is a name occurring on the branch)

Existential instantiation:

$\exists x\, Fx$ ✓
|
Fa

(where ‘a’ is a “new” name)

<u>The rules for negated quantifiers:</u>

$\neg\forall x\, Fx$ ✓
|
$\exists x \neg Fx$

$\neg\exists x\, Fx$ ✓
|
$\forall x \neg Fx$

<u>Remember:</u>

- A sentence does *not* become "used up" after you apply the rule for the universal quantifier
- Apply the rules for negated quantifiers first, followed by the rule for the existential quantifers, followed by the rule for universal instantiation
- *Always* introduce a "new" name when you apply the rule for the existential quantifier, a name that does not yet occur on the branch
- Only introduce a new name when applying the rule for the universal quantifiers if there is no name at all on the branch yet

(5) Models and counterexamples for predicate logic

- In predicate logic, an interpretation of a set of sentences is an assignment of predicates to objects.
- A model of a set of sentences consists of a *domain* together with an interpretation that makes the sentences in the set true together.
- A counterexample (or countermodel) is an interpretation of an argument that makes the premises true and the conclusion false.

To construct a model for a set of sentences, select an open branch of its tree. The domain of the model is represented as a set that contains all and only the things whose names (or individual letters) occur on the selected branch. For instance, our domain might look like this:

$$\mathrm{D} = \{a, b, c\}$$

The interpretation assigns every predicate in occurring on the branch an extension. The extension of the predicate contains the things that the predicate is true of. In an interpretation, extensions are represented by sets. What these sets contain can be read off the branch. If an atomic, not negated sentence occurs on the branch, e.g. '*Fa*', write down the name occurring in that sentence, '*a*' in this case, between the set brackets:

$$\mathrm{I}(F) = \{a\}$$

If the open branch contains, say, '*Ga*', '¬*Gb*', '*Gd*', include between the set brackets all and only the names occurring in the *non-negated* sentences, and just leave the negated ones out. Do not include names not occurring in connection with the predicate in question:

$$\mathrm{I}(G) = \{a, d\}$$

If predicates do not occur in atomic sentences on that branch, nor only in negation of atomic sentences, assign the empty set to the predicate, either by writing *nothing* between the set brackets (resist the temptation to put a dash or slash in it!), or by using the symbol for the empty set: '$\varnothing$'.

$\mathrm{I}(H) = \{\ \}$ or $\mathrm{I}(H) = \varnothing$

<u>Example:</u> $\forall x(Gx \rightarrow Fx), \exists xGx \vdash \forall xFx$

$\forall x(Fx \rightarrow Gx)$ ✓a ✓b
$\exists xGx$ ✓
$\neg\forall xFx$ ✓
|
$\exists x\neg Fx$ ✓
|
Ga
|
$\neg Fb$
|
$Fa \rightarrow Ga$ ✓
/ \
$\neg Fa$ Ga
| |
$Fb \rightarrow Gb$ ✓ $Fb \rightarrow Gb$ ✓
/ \ / \
$\neg Fb$ Gb $\neg Fb$ Gb
☝

Countermodel constructed using the branch pointed out by '☝':

Domain:

D= $\{a, b\}$

Interpretation:

$\mathrm{I}(F) = \varnothing$
$\mathrm{I}(G) = \{a, b\}$