Philosophy 220

Trees for PL 2

Old (still applicable) notions concerning branches:

- Closed: A branch is closed for a tree in PL if it contains an atomic sentence AND the negation of that atomic sentence. A tree is closed when all branches of the tree are closed.
- Open: a branch is open if it is not closed, and a tree is open if at least one branch is completed AND open.

New notions:

- Completed: A branch is completed if one of the following is true:
  - It is closed.
  - It is open and every sentence on the branch is one of the following:
    - An atomic sentence
    - The negation of an atomic sentence
    - A decomposed sentence
    - A universal sentence of which the following conditions ALL apply:
      - It has been decomposed at least once
      - It has been decomposed once for every constant on the branch

Infinite Trees

- Unfortunately, in PL we have the possibility of generating trees whose branches *cannot* satisfy the conditions for completion previously specified.
- For example: (next slide)

An infinite tree

1.	(∀x)(∃y)Pxy		SM
2.	(∃y)Pay √		1, ∀D
3.	Pab		2, ∃D
4.	(∃y)Pby √		1, ∀D
5.	Pbc		2, ∃D

- Note that we must keep re-decomposing 1 every time a new constant is added to the branch, and since we must introduce a foreign constant when we do existential decomposition, this process will go on forever and the tree will never be completed because 1 will never be decomposed once for every constant on the branch.

The solution:

- We introduce a new way to decompose existential statements to deal with this called ∃D2:
- When there exists an x with some property, that x could be one of the things already mentioned (a constant on the branch) or it could be another thing (a constant foreign to the branch)

1. (∃x)Gx
2. Ga
3. Ga   Gb   1, ∃D2

- ∃D2 can branch an indefinite number of times. It branches once for every constant on the branch and one additional time to introduce a constant foreign to the branch.

Solution to the infinite tree:

1.	(∀x)(∃y)Pxy			SM
2.	(∃y)Pay √			1, ∀D
3.	Pab			2, ∃D
4.	(∃y)Pby √			1, ∀D
5.	Pba	Pbb	Pbc	2, ∃D2

- Now notice that the ‘Pbc’ branch is still infinite, but at least the ‘Pba’ and ‘Pbb’ branches are completed and open because 1 on those branches is decomposed once for every constant on the tree.

Systematic trees:

- A systematic tree is a tree that is guaranteed to reach a result so long as the system is followed (and so long as such a result is possible).
- Note that a systematic tree is often not the shortest or simplest tree, but it always gets a result (again, whenever such a result is possible).
- A tree is systematic if and only of it follows "The System"

- "The System" (Paraphrased from text p. 491)
- Stop if (because you have a result):
  - The tree closes
  - You get a completed open branch
- Do the tree in the following order:
  1. Decompose truth functional sentences and existentially quantified sentences (with ∃D2)
  2. Decompose universal sentences once, and once for every constant on the branch.
  - Every time a step 1 action is available, do it before doing any available step 2 action.

Limits to the system:

- There are still some infinite trees, but it usually becomes clear which those are. Generally, whenever the only branches that continue are the branches of ∃D2 that introduce foreign constants, there is no finite solution, but alas, there is no way to prove that.