Logics for Computation

Lecture #6: No Way to Say Warm in French

Carlos Areces and Patrick Blackburn

{carlos.areces,patrick.blackburn}@loria.fr

INRIA Nancy Grand Est
Nancy, France

ESSLLI 2008 - Hamburg - Germany

The Story so Far

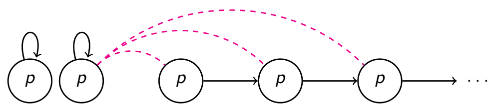
- We are working with the $\langle R \rangle$ language
- We saw that the lenguage cannot say
  - "I am not a tree"
  - "I am infinite"
- On the positive side
  - It has a simple reasoning calculus: labelled tableaux
  - It is decidable.

What do we do Today

- We will extend the expressive power of the language. . .
- . . . and explore quite a number of possibilities.
- We will learn to count till n.
- We will learn to name nodes.
- We will learn to say "everywhere".
- We will learn to say "it won't take forever".

One is the Same as Infinite

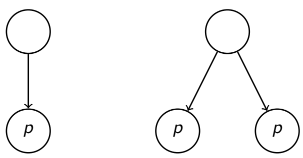
- We said in the previos lecture that we cannot say infinite in the $\langle R \rangle$ language.
- Let's see this in more detail. Consider the model:



- This is not a tree. Hence, there should be a tree like structure which should be the same as this one for the $\langle R \rangle$ language.

One is the Same as Two

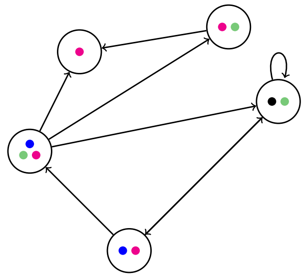
- But let's consider a simpler example (after all, infinite is quite a big number).
- We saw that the $\langle R \rangle$ language cannot distinguish between one and two!!!



Learning to Count

Suppose we want to say that two green nodes are accessible . . .

$\langle R \rangle$green $\wedge$ $\langle R \rangle$green
$\langle R \rangle$(green $\wedge$ pink) $\wedge$
$\langle R \rangle$(green $\wedge \neg$ pink)
$\langle = 2\ R \rangle$ green



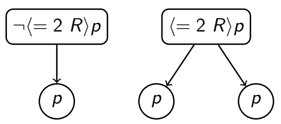
Alice in Wonderland

> **Humpty Dumpty**: When I use a word, it means just what I choose it to mean – neither more nor less.
> **Alice**: The question is, whether you can make words mean so many different things.
> **Humpty Dumpty**: The question is: which is to be master – that's all.

- If the language cannot express something we are interested in, we just extend the language!
- Counting successors:

$$\mathcal{M}, w \models \langle = n\ R \rangle \varphi \text{ iff } |\{w' \mid wRw' \text{ and } \mathcal{M}, w' \models \varphi\}| = n$$

- Clearly:



The models are not the same for the $\langle = n\ R \rangle$ language.

Extending the Language

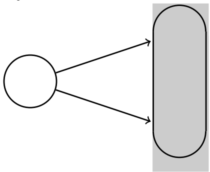
- What other things we cannot say in the $\langle R \rangle$ language?
- Plenty:
  1. In that particular node.
  2. Everywhere in the model.
  3. In a finite number of steps.

  . . .
- Luckily, as Humpty Dumpty says, we are the masters, and we can desing the language that better pleases us.
- Let's get to work. . .

Names for Points

Suppose that n is a name for a point. That is, it can label a unique point in any relational structure.

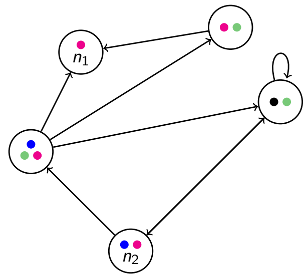
$\langle R\rangle$(pink $\wedge$ n)



- Looks useful...
- We can introduce names into our language (you probably know them as constants).

Names for Points

- When we allow names in our language, our models will look like this:



- We have already used something like names. Anybody remembers when?
- Tableaux for the $\langle R\rangle$ language!
- If we also introduce the :-operator we can write things like

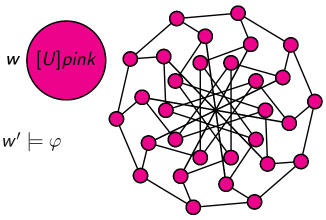
  n_1:pink

  n_2:$\langle R\rangle$pink

  n_2:$\langle R\rangle\langle R\rangle\langle R\rangle n_2$

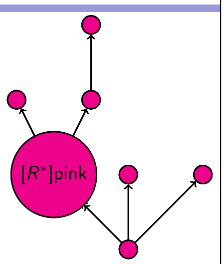
Everywhere in the model

- Suppose we want to paint everything pink (we love pink).
- Can we do it? Let's see and example, consider this model:



- Is there a formula of the $\langle R\rangle$ language, that we can make true at w, so that pink is true everywhere in the model?
- Define the $[U]$ operator as $\mathcal{M}, w \models [U]\varphi$ iff for all w', $\mathcal{M}, w' \models \varphi$
- Then $\mathcal{M}, w \models [U]$pink if the whole model is pink.
- Jah!

In a Finite Number of Steps



- In the $\langle R\rangle$ language we can say
  - In one step p: $\langle R\rangle p$
  - In two steps p: $\langle R\rangle\langle R\rangle p$
  - ...
  - But we cannot say, in a finite (zero or more, but unspecified) number of steps p:
    $p \vee \langle R\rangle p \vee \langle R\rangle\langle R\rangle p \vee \ldots$
- Define the $\langle R^*\rangle$ operator as
  $\mathcal{M}, w \models \langle R^*\rangle\varphi$ iff there is w' s.t. wR^*w' and $\mathcal{M}, w' \models \varphi$
  for R^* is the reflexive and transitive closure of R.
  (Let's write $[R^*]\varphi$ for $\neg\langle R^*\rangle\neg\varphi$)

- Pretty choosy! (ok, let's say selective)

The Complete Menu

Our models are structures like $\mathcal{M} = \langle W, \{R_i\}, \{P_i\}, \{N_i\}\rangle$

- Counting successors:
  $\mathcal{M}, w \models \langle = n\ R\rangle\varphi$ iff $|\{w' \mid wRw' \text{ and } \mathcal{M}, w' \models \varphi\}| = n$.
  (compare with
  $\mathcal{M}, w \models \langle R\rangle\varphi$ iff there is w' s.t. wRw' and $\mathcal{M}, w' \models \varphi$.)
- Names and the :-operator:
  $$\mathcal{M}, w \models n_i \text{ iff } w = N_i$$
  (compare with
  $$\mathcal{M}, w \models p_i \text{ iff } w \in P_i)$$
  $$\mathcal{M}, w \models n_1{:}\varphi \text{ iff } \mathcal{M}, N_i \models \varphi$$
  (compare with
  $\mathcal{M}, w \models \langle R\rangle\varphi$ iff there is w' s.t. wRw' and $\mathcal{M}, w' \models \varphi$.)
- The $[U]$-operator:
  $$\mathcal{M}, w \models [U]\varphi \text{ iff for all } w', \mathcal{M}, w' \models \varphi$$
  (compare with
  $\mathcal{M}, w \models \langle R\rangle\varphi$ iff there is w' s.t. wRw' and $\mathcal{M}, w' \models \varphi$.)

Back to the Intuitions!

The main idea we want to get across is:

There are plenty of options, go and chose what you need!

Even more, if it is not there, then define it yourself

Remember what Humpty Dumpty said:
"The question is: Which is to be master"

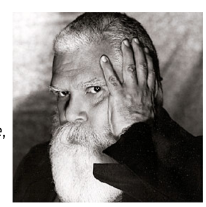
- By combining the operators we have been discusing we obtain a wide variety of languages.
  - We go from languages of low expressivity (PL) to languages of high expressivity (the selective $[R^*]$).
  - We go from languages of 'low' complexity (NP-complete) to languages of hight complexity (EXPTIME-complete).
- By chosing the right expressivity for a given application we will pay the exact price required.

What we Covered Today

- We discussed the polytheistic approach in full glory:
  - counting
  - constants
  - universal quantification
  - reflexive and transitive closure
- By combining all these operators we obtain very diverse logics.
- But, they all share the same semantics: relational structures!
- They are just different ways of talking about something.

Relevant Bibliography I

[...] No way to say *warm* in French. There was only *hot* and *tepid*. If there's no word for it, how do you think about it? [...] Imagine, in Spanish having to assign a gender to every object: dog, table, tree, can-opener. Imagine, in Hungarian, not being able to assign a gender to anything: *he*, *she*, *it* all the same word.



- My French is not good enough to say if it's true...
- But it's definitely a great science fiction book!

Delany, Samuel (1966). *Babel-17*. Ace Books.

Relevant Bibliography II

- Many of the languages that we have been discussing are investigated in detail in the area known as Modal Logics.
- The name 'modal' (in many cases as opposed to 'classical') doesn't make much sense.
- Some of these languages have been extensively studied by somebody you know quite well by now.
  Blackburn's Web page: `http://www.loria.fr/~blackbur`
- M. de Rijke also pushed the idea of working with modal logics extending the $\langle R \rangle$ language.
  de Rijke's Web page: `http://staff.science.uva.nl/~mdr/`

Blackburn, Patrick and van Benthem, J (2006). *Chapter 1 of the Handbook of Modal Logics*, Blackburn, P.; Wolter, F.; and van Benthem, J., editors, Elsevier.

de Rijke, Maarten (1993). *Extending Modal Logic* PhD Thesis. Institute for Logic, Language and Computation, Unviersity of Amsterdam.

The Next Lecture

DIY First Order Logic