Free and bound variables

We'd better investigate how variables can arise in formulas.

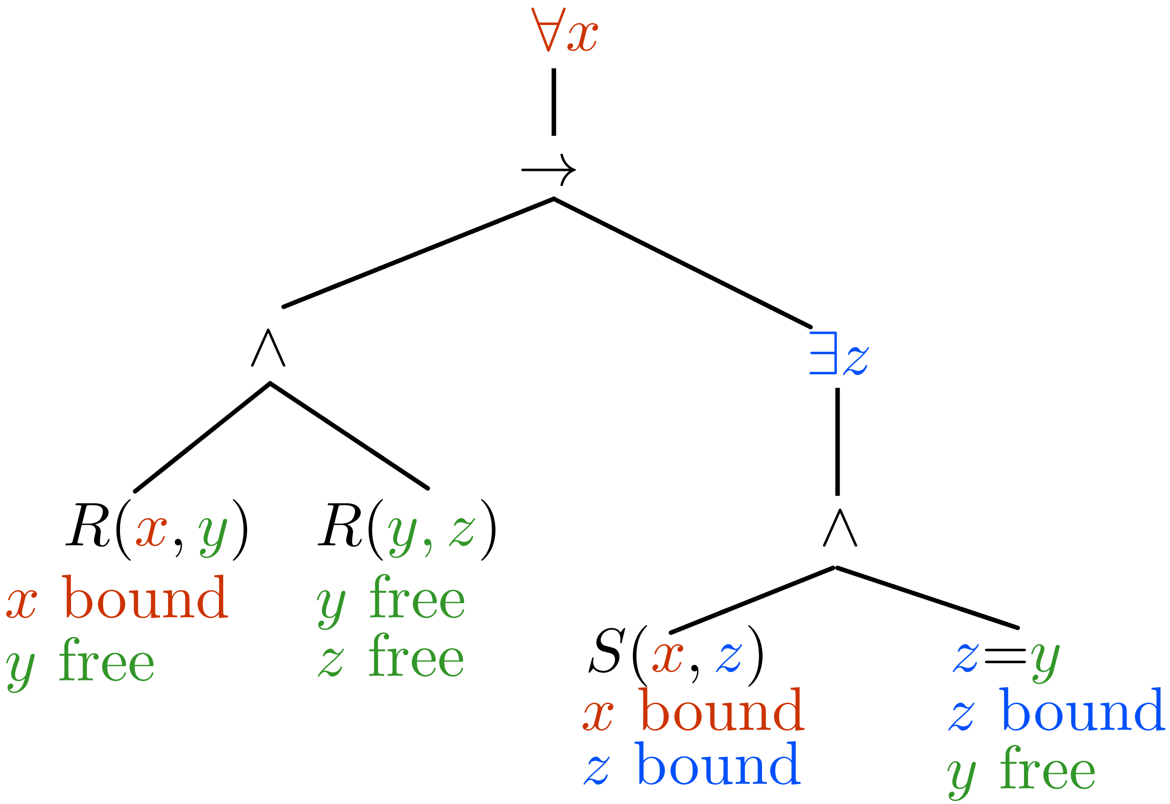
Definition 1.7

Let A be a formula.

1. An occurrence of a variable x in an atomic subformula of A is said to be bound if it lies under a quantifier $\forall x$ or $\exists x$ in the formation tree of A.
2. If not, the occurrence is said to be free.
3. The free variables of A are those variables with free occurrences in A.

Example

$$\forall x(R(x,y) \land R(y,z) \to \exists z(S(x,z) \land z = y))$$



The free variables of the formula are y, z.
Note: z has both free and bound occurrences.

Sentences

Definition 1.8

A sentence is a formula with no free variables.

Example 1.9

- $\forall x(\texttt{bought}(\texttt{Tony}, x) \to \texttt{PC}(x))$ is a sentence.
- Its subformulas are not sentences:

$$\texttt{bought}(\texttt{Tony}, x) \to \texttt{PC}(x)$$
$$\texttt{bought}(\texttt{Tony}, x)$$
$$\texttt{PC}(x)$$

Which are sentences?

- $\texttt{bought}(\texttt{Frank}, \texttt{Texel})$
- $x = x$
- $\texttt{bought}(\texttt{Susan}, x)$
- $\forall x(\exists y(y = x) \to x = y)$
- $\forall x \forall y(x = y \to \forall z(R(x, z) \to R(y, z)))$

Problem 1: free variables

Sentences are true or false in a structure.

But non-sentences are not!

A formula with free variables is neither true nor false in a structure M, because the free variables have no meaning in M.

It's like asking 'is $x = 7$ true?'

So, the structure is not a 'complete' situation — it doesn't fix the meanings of free variables. (They are variables, after all!)

Handling values of free variables

So we must specify values for free variables, before evaluating a formula to true or false.

This is so even if it turns out that the values do not affect the answer (like $x = x$).

Assignments to variables

We supply the missing values of free variables using something called an assignment.

What a structure does for constants, an assignment does for variables.

Definition 1.10 (assignment)

Let M be a structure. An assignment (or 'valuation') into M is something that allocates an object in $\mathrm{dom}(M)$ to each variable.

For an assignment h and a variable x, we write $h(x)$ for the object assigned to x by h.

[Formally, $h : V \to \mathrm{dom}(M)$ is a function.]

Evaluating terms

Given an L-structure M plus an assignment h into M, we have a 'complete situation'. We can then evaluate:

- any L-term, to an object in $\mathrm{dom}(M)$,
- any L-formula with no quantifiers, to true or false.

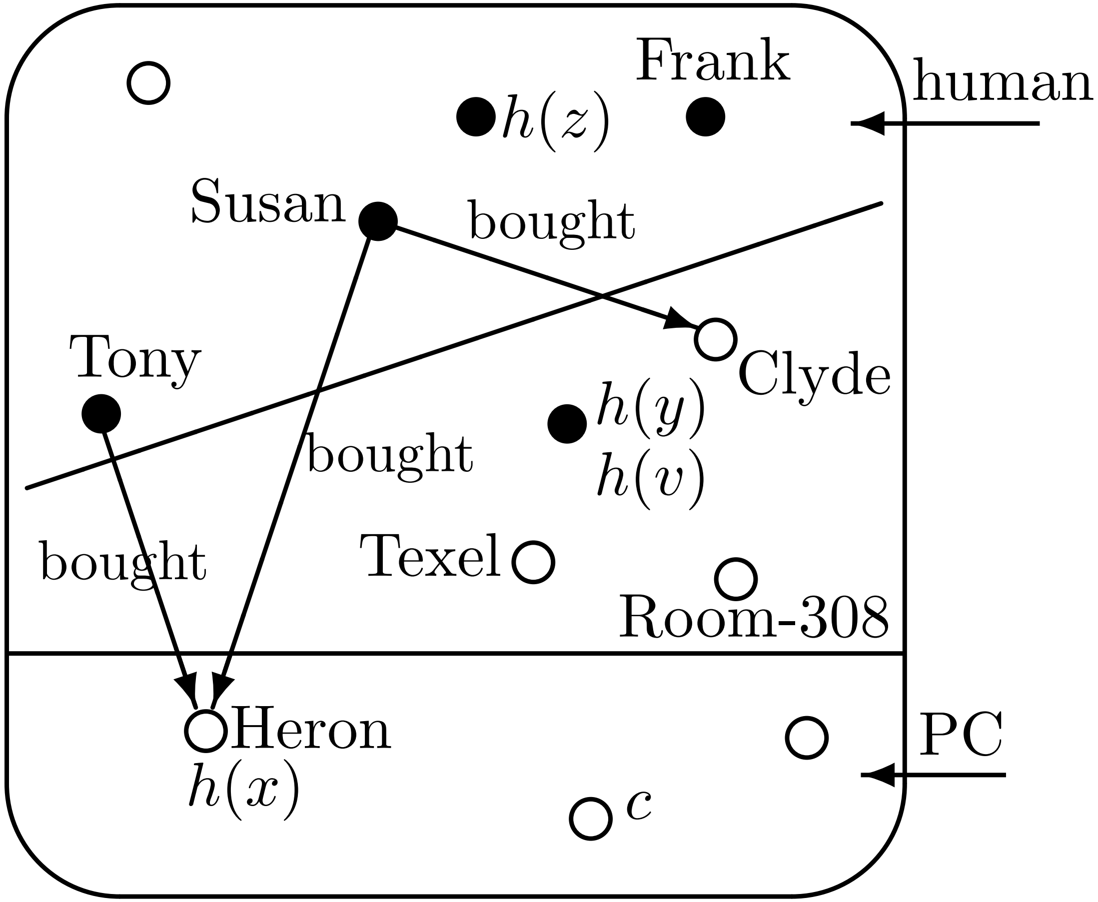
We do the evaluation in two stages: first terms, then formulas.

Definition 1.11 (value of term)

Let L be a signature, M an L-structure, and h an assignment into M. Then for any L-term t, the *value of t in M under h* is the object in M allocated to t by:

- M, if t is a constant — that is, t^M,
- h, if t is a variable — that is, $h(t)$.

Evaluating terms: example



M, h

The value in M under h of the term `Tony` is (the ● marked) 'Tony'.
The value in M under h of the term x is Heron.

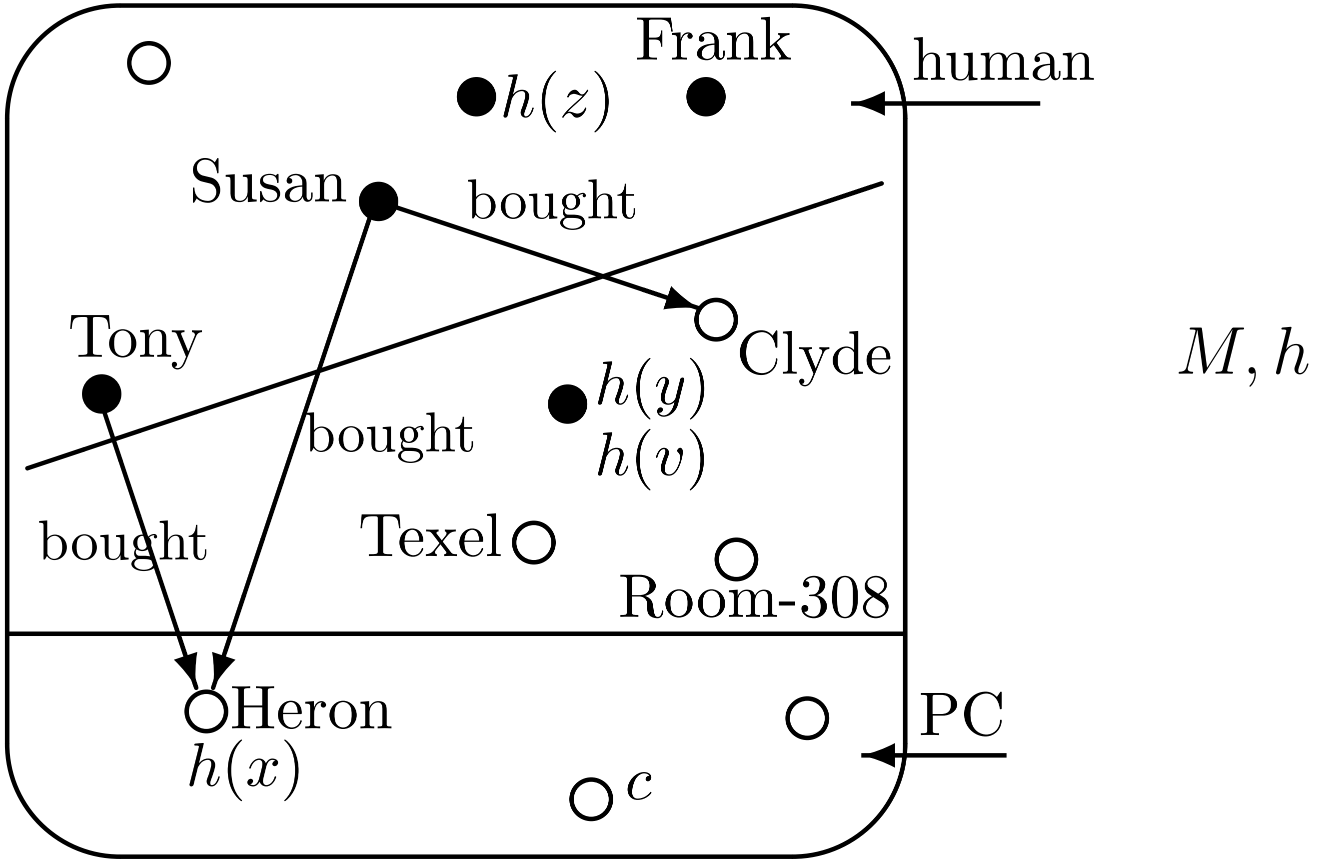
Semantics of quantifier-free formulas

We can now evaluate any formula without quantifiers.
Fix an L-structure M and an assignment h.
We write $M, h \models A$ if A is true in M under h, and $M, h \not\models A$ if not.

Definition 1.12

1. Let R be an n-ary relation symbol in L, and $t_1, \ldots, t_n$ be L-terms. Suppose that the value of t_i in M under h is a_i, for each $i = 1, \ldots, n$ (see Def. 1.3). Then $M, h \models R(t_1, \ldots, t_n)$ if M says that the sequence $(a_1, \ldots, a_n)$ is in the relation R. If not, then $M, h \not\models R(t_1, \ldots, t_n)$.
2. If t, t' are terms, then $M, h \models t = t'$ if t and t' have the same value in M under h. If they don't, then $M, h \not\models t = t'$.
3. $M, h \models \top$, and $M, h \not\models \bot$.
4. $M, h \models A \wedge B$ if $M, h \models A$ and $M, h \models B$. Otherwise, $M, h \not\models A \wedge B$.
5. $\neg A$, $A \vee B$, $A \to B$, $A \leftrightarrow B$ — similar: as in propositional logic.

Evaluating quantifier-free formulas: example



- $M, h \models \texttt{human}(z)$
- $M, h \models x = \texttt{Heron}$
- $M, h \not\models \texttt{bought}(\texttt{Susan}, v) \vee z = \texttt{Frank}$

Problem 2: bound variables

We now know how to specify values for free variables: with an assignment. This allowed us to evaluate all quantifier-free formulas. But most formulas involve quantifiers and bound variables. Values of bound variables are not — and should not be — given by the situation, as they are controlled by quantifiers.

How do we handle this?

Answer:

We let the assignment vary. Rough idea:

- for $\exists$, want some assignment to make the formula true;
- for $\forall$, demand that all assignments make it true.