32002: AI (Unification Algorithm) **Spring 2015**

Lecture 6: January 20, 2015

Lecturer: K.R. Chowdhary *: Professor of CS (VF)*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the Instructor.*

6.0.1 Unification Algorithm

The unification algorithm recursively compares the structures of the clauses to be matched, working across element by element. The criteria is, that the matching individuals, functions, and predicates must have the same names; the matching functions and predicates must have the same number of arguments, and all bindings of variables to values must be consistent throughout the whole match.

To unify two atomic formulas in an expression A, we need to understand the *disagreement* set.

Definition 6.1 *Disagreement Set.*

If A is any set of well-formed expressions, we call the set D the disagreement set of A, whenever D is the set of all well-formed subexpressions of the well-formed expressions in A, which begin at the first symbol position at which not all well-formed expressions in A have the same symbol. □

Example 6.2 *Find out the disagreement set for given set of atoms.*

Solution. Let the string is, $A = \{P(x, h(x, y), y),\ P(x, k(y), y),\ P(x, a, b)\}$, having three predicate expressions. The disagreement set for A is,

$$D = \{h(x, y), k(y), a\}. \quad (6.1)$$

Once the disagreement is resolved through unification for this this symbol position, there is no disagreement at this position. The process is repeated for the new first symbol position at which all wffs in A do not have same symbol, and so on, until A becomes a singleton.

For A to be a finite nonempty set of well-formed expressions for which the substitution algorithm (1) terminates with "return σ_A", the substitution σ_A available as output of the unification algorithm is called the most general unifier (mgu) of A, and A is said to be most generally unifiable.

Through manually running the algorithm 1 for the disagreement set in (6.1), stepwise computation for σ_k is as follows:

For $k = 0$, and $b_0 = \epsilon$,

$$\sigma_{k+1} = \sigma_k\{k(y)/h(x, y)\}$$
$$\Rightarrow \sigma_1 = \{k(y)/h(x, y)\}$$

Algorithm 1 Algorithm-Unification(Input: A, Output: σ_A)

Set $\sigma_0 = \epsilon, k = 0$
while true **do**
  if $A\sigma_k$ is a singleton **then**
    Set $\sigma_A = \sigma_k$
    terminate
  end if
  Let U_k be the earliest and V_k be the next earliest element, in the disagreement set D_k of $A\sigma_k$ (see equation 6.1)
  if U_k is a variable, and does not occur in V_k **then**
    set $\sigma_{k+1} = \sigma_k\{V_k/U_k\}$,
    $k = k + 1$
  else
    (A is not unifiable)
    exit.
  end if
end while

which, in the next iteration becomes,

$$\begin{aligned}\sigma_2 &= \sigma_1\{a/k(y)\}\\ &= \{k(y)/h(x,y)\}\{a/k(y)\}\end{aligned}$$

The same process is repeated for the disagreement set of 3rd argument in A, which results to substitution set as $\{b/y\}$.

$$\begin{aligned}\sigma_3 &= \sigma_2\{b/y\}\\ &= \{k(y)/k(x,y)\}\{a/k(y)\}\{b/y\}\end{aligned}$$

On substituting these, we have,

$$A = \{p(x,a,b), p(x,a,b), p(x,a,b\}$$

which is a singleton, and σ_3 is mgu. For obtaining the unifier σ_k, the necessary relation required between U_k and V_k is, U_k has to be a variable, and V_k can be a constant, variable, function, or predicate. U_k may even be a predicate or function with variable.

The algorithm 1 always terminates for finite nonempty set of well-formed expressions, for otherwise there would be generated an infinite sequence of $A, A\sigma_1, A\sigma_2, \ldots,$ of finite nonempty sets of well-formed expressions with the property that each successive set contains one less variable than its predecessor. But this is impossible since A contains only finitely many distinct variables.