Example 22 Consider the clause

$$C = \{Q(g(y, x)), \neg P(f(x))\}$$

Replacing x by $f(z)$ in C results in the instance

$$C' = \{Q(g(y, f(z))), \neg P(f(f(z)))\}$$

Replacing y by $j(a)$ and z by b in C' results in the instance

$$C'' = \{Q(g(j(a), f(b))), \neg P(f(f(b)))\}$$

Assuming that a and b are constants, C'' is a ground instance of C.

Theorem 16 *A set S of clauses is unsatisfiable if and only if there is a finite unsatisfiable set S' of ground instances of clauses of S.*

The proof is rather involved; see Chang and Lee, pages 56–61, for details. The ($\Longrightarrow$) direction is the interesting one. It uses a non-constructive argument to show that if there is no finite unsatisfiable set S', then there must be a model of S.

The ($\Longleftarrow$) direction simply says that if S' is unsatisfiable then so is S. This is straightforward since every clause in S' is a logical consequence of some clause in S. Thus if S' is inconsistent, the inconsistency is already present in S.

Question: how do we discover *which* ground instances? Answer: by *unification*.

Exercise 29 Consider a first-order language with 0 and 1 as constant symbols, with $-$ as a 1-place function symbol and $+$ as a 2-place function symbol, and with $<$ as a 2-place predicate symbol.

(a) Describe the Herbrand Universe for this language.

(b) The language can be interpreted by taking the integers for the universe and giving 0 1, $-$, $+$, and $<$ their usual meanings over the integers. What do those symbols denote in the corresponding Herbrand model?

9 Unification

Unification is the operation of finding a common instance of two terms. Though the concept is simple, it involves a complicated theory. Proving the unification algorithm's correctness (especially termination) is difficult.

To introduce the idea of unification, consider a few examples. The terms $f(x, b)$ and $f(a, y)$ have the common instance $f(a, b)$, replacing x by a and y

by b. The terms $f(x, x)$ and $f(a, b)$ have no common instance, assuming that a and b are distinct constants. The terms $f(x, x)$ and $f(y, g(y))$ have no common instance, since there is no way that x can have the form y and $g(y)$ at the same time — unless we admit the infinite term $g(g(g(\cdots)))$.

Only variables may be replaced by other terms. Constants are not affected (they remain constant!). If a term has the form $f(t, u)$ then instances of that term must have the form $f(t', u')$, where t' is an instance of t and u' is an instance of u.

9.1 Substitutions

We have already seen substitutions informally. It is now time for a more detailed treatment.

Definition 17 A *substitution* is a finite set of replacements

$$[t_1/x_1, \ldots, t_k/x_k]$$

where $x_1, \ldots, x_k$ are distinct variables such that $t_i \neq x_i$ for all $i = 1, \ldots, k$. We use Greek letters ϕ, θ, σ to stand for substitutions.

The finite set $\{x_1, \ldots, x_k\}$ is called the *domain* of the substitution. The domain of a substitution θ is written $\mathrm{dom}(\theta)$.

A substitution $\theta = [t_1/x_1, \ldots, t_k/x_k]$ defines a function from the variables $\{x_1, \ldots, x_k\}$ to terms. Postfix notation is usual for applying a substitution; thus, for example, $x_i\theta = t_i$. Substitutions may be applied to terms, not just to variables. Substitution on terms is defined recursively as follows:

$$\begin{aligned} f(t_1, \ldots, t_n)\theta &= f(t_1\theta, \ldots, t_n\theta) \\ x\theta &= x \qquad \text{for all } x \notin \mathrm{dom}(\theta) \end{aligned}$$

Here f is an n-place function symbol. The operation substitutes in the arguments of functions, and leaves unchanged any variables outside of the domain of θ.

Substitution may be extended to literals and clauses as follows:

$$\begin{aligned} P(t_1, \ldots, t_n)\theta &= P(t_1\theta, \ldots, t_n\theta) \\ \{L_1, \ldots, L_m\}\theta &= \{L_1\theta, \ldots, L_m\theta\} \end{aligned}$$

Here P is an n-place predicate symbol (or its negation), while $L_1, \ldots, L_m$ are the literals in a clause.

Example 23 The substitution $\theta = [h(y)/x, b/y]$ says to replace x by $h(y)$ and y by b. The replacements occur simultaneously; it does *not* have the effect of replacing x by $h(b)$. Its domain is $\mathrm{dom}(\theta) = \{x, y\}$. Applying this substitution gives

$$\begin{aligned} f(x, g(u), y)\theta &= f(h(y), g(u), b) \\ R(h(x), z)\theta &= R(h(h(y)), z) \\ \{P(x), \neg Q(y))\}\theta &= \{P(h(y)), \neg Q(b))\} \end{aligned}$$

9.2 Composition of substitutions

If ϕ and θ are substitutions then so is their *composition* $\phi \circ \theta$, which satisfies

$$t(\phi \circ \theta) = (t\phi)\theta \qquad \text{for all terms } t$$

Can we write $\phi \circ \theta$ as a set of replacements? It has to satisfy the above for all relevant variables:

$$x(\phi \circ \theta) = (x\phi)\theta \qquad \text{for all } x \in \mathrm{dom}(\phi) \cup \mathrm{dom}(\theta)$$

Thus it must be the set consisting of the replacements

$$(x\phi)\theta \;/\; x \qquad \text{for all } x \in \mathrm{dom}(\phi) \cup \mathrm{dom}(\theta)$$

Equality of substitutions ϕ and θ is defined as follows: $\phi = \theta$ if $x\phi = x\theta$ for all variables x. Under these definitions composition enjoys an associative law. It also has an identity element, namely [], the empty substitution.

$$\begin{aligned} (\phi \circ \theta) \circ \sigma &= \phi \circ (\theta \circ \sigma) \\ \phi \circ [] &= \phi \\ [] \circ \phi &= \phi \end{aligned}$$

Example 24 Let $\phi = [j(x)/u, 0/y]$ and $\theta = [h(z)/x, g(3)/y]$. Then $\mathrm{dom}(\phi) = \{u, y\}$ and $\mathrm{dom}(\theta) = \{x, y\}$, so $\mathrm{dom}(\phi) \cup \mathrm{dom}(\theta) = \{u, x, y\}$. Thus

$$\phi \circ \theta = [j(h(z))/u, h(z)/x, 0/y]$$

Notice that $y(\phi \circ \theta) = (y\phi)\theta = 0\theta = 0$; the replacement $g(3)/y$ has disappeared.

Exercise 30 Verify that $\circ$ is associative and has [] for an identity.

9.3 Unifiers

Definition 18 A substitution θ is a *unifier* of terms t_1 and t_2 if $t_1\theta = t_2\theta$. More generally, θ is a unifier of terms $t_1, t_2, \ldots, t_m$ if $t_1\theta = t_2\theta = \cdots = t_m\theta$. The term $t_1\theta$ is called the *common instance* of the unified terms. A unifier of two or more literals is defined similarly.

Two terms can only be unified if they have similar structure apart from variables. The terms $f(x)$ and $h(y, z)$ are clearly non-unifiable since no substitution can do anything about the differing function symbols. It is easy to see that θ unifies $f(t_1, \ldots, t_n)$ and $f(u_1, \ldots, u_n)$ if and only if θ unifies t_i and u_i for all $i = 1, \ldots, n$.

Example 25 The substitution $[3/x, g(3)/y]$ unifies the terms $g(g(x))$ and $g(y)$. The common instance is $g(g(3))$. These terms have many other unifiers, including the following:

unifying substitution	common instance
$[f(u)/x, g(f(u))/y]$	$g(g(f(u)))$
$[z/x, g(z)/y]$	$g(g(z))$
$[g(x)/y]$	$g(g(x))$

Note that $g(g(3))$ and $g(g(f(u)))$ are instances of $g(g(x))$. Thus $g(g(x))$ is more general than $g(g(3))$ and $g(g(f(u)))$; it admits many other instances. Certainly $g(g(3))$ seems to be arbitrary — neither of the original terms mentions 3! A separate point worth noting is that $g(g(x))$ is equivalent to $g(g(z))$, apart from the name of the variable. Let us formalize these intuitions.

9.4 Most general unifiers

Definition 19 The substitution θ is *more general* than ϕ if $\phi = \theta \circ \sigma$ for some substitution σ.

Example 26 Recall the unifiers of $g(g(x))$ and $g(y)$. The unifier $[g(x)/y]$ is more general than the others listed, for

$$
\begin{aligned}
[3/x, g(3)/y] &= [g(x)/y] \circ [3/x] \\
[f(u)/x, g(f(u))/y] &= [g(x)/y] \circ [f(u)/x] \\
[z/x, g(z)/y] &= [g(x)/y] \circ [z/x] \\
[g(x)/y] &= [g(x)/y] \circ []
\end{aligned}
$$