Abstract

We present a method for validating abstract data type specifications. The method takes as input a set of ground terms \( L_0 \) and a set of conditional equations \( A_0 \) over \( L_0 \). The object of this method is to find a normal form function, \( \text{Norm} \), for the pair \( < L_0, A_0 > \). The function \( \text{Norm} \) is computed as a sequence of step functions \( S_1, S_2, ..., S_n \).

Each step function \( S_i \), \( 0 \leq i \leq n \), takes as input a pair \( < L_{i-1}, A_{i-1} > \), where \( L_{i-1} \) is a set of ground terms and \( A_i \) is a set of conditional equations over the set of terms \( L_{i-1} \). At each step \( i \), a set of equations \( E_i \) is selected from the set of theorems of the pair \( < L_{i-1}, A_{i-1} > \). The set of equations \( E_i \) is transformed into a set of reductions \( R_i \). The step function \( S_i \) is defined as the top-down reduction extension of \( R_i \) to \( L_{i-1} \). The output of \( S_i \) is the pair \( < L_i, A_i > \), where \( L_i \) is the set of normal forms of \( L_{i-1} \) under the set of reductions \( R_i \) and \( A_i \) is the set of normal forms of the equations in \( A_{i-1} \) under the same set of reductions. This way, a theorem in the system \( < L_{i-1}, A_{i-1} > \) becomes a theorem in the system \( < L_i, A_i > \). The last step, \( S_n \), has as output the pair \( < L_n, \phi > \). The only theorems in \( < L_n, \phi > \) are the identities. This way the sequence \( < L_0, A_0 > \rightarrow S_1 \rightarrow S_2 \rightarrow ... \rightarrow S_{n-1} \) \( A_{n-1} > \rightarrow S_n \rightarrow < A_n, \phi > \) gives us a procedure to compute the normal form of the terms in \( < L_0, A_0 > \).

In this paper we present criteria for choosing the sets of equations \( E_i \) which simplify the pair \( < L_{i-1}, A_{i-1} > \). We also present results that characterize the output set \( < L_i, A_i > \) of \( S_i \) as a function of the set \( < L_{i-1}, A_{i-1} > \) and of the set of reductions \( R_i \). If the sets of reductions \( R_i \) are confluent and terminating, then they can be combined, by using a priority system similar to the one developed by Baeten, Bergstra and Klop, to form a confluent and terminating set of reductions on the set \( < L_0, A_0 > \).

1. Preliminaries

The purpose of this research is to define a method for validating specifications of abstract data types. We use initial algebra semantics as models for our abstract data types ([2],[3]). An initial algebra is the quotient of a free algebra under a set of conditional equations. The free algebra is obtained from a set of operators \( F \). Each operator in \( F \) has a fixed arity; we define the arity to be a function from the set of operators \( F \) to the set \( S^+ \), where \( S \) is the set of sorts. The set \( S^+ \) contains all the nonempty strings over \( S \). If \( w_1w_2...w_n\epsilon S^+ \) is an arity, we write it as \( w_1...w_{n-1} \rightarrow w_n \). The operators which have arity \( s \) are called constants of sort \( s \); we shall call an operator which is not a constant a constructor. The free algebra generated by \( F \) is obtained in the usual way, by starting with constants and repeatedly applying constructors to obtain new terms. We use \( T(F)_s \) to denote the set of terms of sort \( s \) generated by the set of operators \( F \); \( T(F) \) denotes the union of the sets \( T(F)_s \). The elements of \( T(F) \) are called ground terms.
Let $X$ be an $S$-sorted set. The set $X$ is disjoint from $F$ and each element $x$ in $X$ has arity $\rightarrow s$, $s$ being an element in $S$. We call $x$ a variable of sort $s$. We choose $X$ large enough to include a denumerable set of variables for each sort. We can add the set of variables $X$ to the set of operators $F$ and form the free algebra $T(F, X)$ corresponding to the set of operators $X \cup F$. $T(F, X)$ is the set of terms with variables.

The notions of occurrence and subterm are assumed to be known; they can be found in Huet and Oppen ([5]). We use the notation $Var(t)$ to denote the set of variables in the term $t \in T(F, X)$. An equation in $T(F, X)$ is a pair $< M, N >$, where $M$ and $N$ are terms in $T(F, X)_s$, for some sort $s \in S$. We write it $M = N$; we call it a pure equation. We use the notation $Var(M = N)$ for $Var(M) \cup Var(N)$. A conditional equation in $T(F, X)$ is an implication $e_1, e_2, \ldots, e_n \implies e$, where $e_1, e_2, \ldots, e_n, e$ are pure equations in $T(F, X)$.

A substitution is an $S$-sorted map $s : X \rightarrow T(F, X)$ in which $s(x) \neq x$ for finitely many variables $x$. The set of variables $x$ for which $s(x) \neq x$ is called the domain of the substitution $s$. The substitution $s$ can be extended to a map $\bar{s} : T(F, X) \rightarrow T(F, X)$ by specifying that: $\bar{s}(t) = t$ if $t$ is constant, $\bar{s}(t) = s(t)$ if $t$ is a variable and $\bar{s}(f(t_1, \ldots, t_n)) = f(\bar{s}(t_1), \ldots, \bar{s}(t_n))$ for terms of the form $f(t_1, t_2, \ldots, t_n)$. The term $\bar{s}(t)$ is called an instance of $t$. If $s$ is a substitution with domain $Var(M = N)$ and range $L$, where $L$ is a subset of $T(F, X)$, the equation $\bar{s}(M) = \bar{s}(N)$ is called an instance of the equation $M = N$ over the set $L$; if $L = T(F, X)$ we simply call it an instance of $M = N$. In a similar way we define the notion of instance of a conditional equation. A ground substitution is a substitution that has $T(F)$ as its range. We will assume that the the sets of ground terms $T(F)_s$ are not empty; this condition is useful in defining the quotient algebras. We will work with subsets of $T(F)$.

**Definition 1.1**
1. A subset $L$ of $T(F)$ is stable if for all sorts $s$, $L_s$, the set of subterms of sort $s$, is not empty.
2. Let $t$ be a term in $T(F, X)$, $V$ be the set of variables occurring in $t$, and $L$ be a stable subset of $T(F)$. We say that $L$ is closed under $t$ if for all ground substitutions $s$ with domain $V$ and range $L$, $s(t)$ is also a member of $L$.
3. We say that $L$ is closed under a pure equation $M = N$ if $L$ is closed under $M$ and $L$ is closed under $N$.
4. We say that $L$ is closed under a conditional equation $e_1, e_2, \ldots, e_n \implies e$ if $L$ is closed under all the equations $e_1, e_2, \ldots, e_n, e$.
5. We say that $L$ is closed under a set of equations if $L$ is closed under each equation in the set.

For example the set $\{a, f(a), \ldots, f^n(a), \ldots\}$ is closed under the equation $f(f(x)) = x$, since it is closed under the set $\{f(f(x)), x\}$. The notation $f^n(a)$ stands for $f(\ldots f(a)\ldots)$, where the constructor $f$ occurs $n$ times.

Further on we will assume that all sets $L$ are stable.

**Definition 1.2**
1. Let $L$ be a subset of $T(F)$ and $t$ a term in $T(F, X)$. We say that $t$ has the subterm property for $L$, if for all instances $t'$, of $t$ over $L$, the following property holds: if $t'$ is in $L$ then all its subterms are in $L$.
2. We say that $L$ is full for a set of terms if each term in the set has the subterm property for $L$. 