# Induction and Synthesis by Simplification

Idriss Bengeloune
FB14 Informatik
Universität des Saarlandes
Postfach 151150 D–66041 Saarbrücken
e-mail: idriss@cs.uni-sb.de

## Abstract

We present a method for proving that existentially quantified formulas are valid in the initial model of a given set of equations. This approach avoids completion and explicit induction and relies on the notion of cover set. Attemping to prove a formula, we instantiate it with terms from the cover set and a simplification strategy is started. The simplification is based upon an ordered term rewriting system and may use previously proved conjectures. We show how to use the proof obtained from our method to generate a recursive definition of a skolem function for an existentially quantified formula.

**keywords:** Existence proofs, induction, term rewriting, cover sets, program synthesis.

## Résumé

Les formules existentielles jouent un rôle important dans le domaine de la synthèse déductive de programmes. Nous présentons une méthode permettant la construction (automatique) de preuves de validité de formules existentielles dans le modèle initial d'un ensemble d'équations. Cette méthode est basée sur une notion de *cover set* et de système de réécriture ordonné. Nous proposons également un algorithme de synthèse de définitions récursives de fonction de Skolem pour une formule existentielle à partir de sa preuve de validité.

# 1 Introduction

## 1.1 Motivation

Existence formulas are frequently used in computer science to describe a relation between the input and the output of a desired program. In the field of program synthesis inductive existence proofs are used to compute algorithmic definitions for the skolem functions under consideration [15]. Inductive theorems are usually valid only in particular models of a given set of axioms, for instance Herbrand models or the initial model. We propose a method for existence proofs in the initial model of an equational variety. The basic idea was inspired by the way Chazarain and Kounalis [8] use (purely) algebraic simplifications to mechanize inductive reasoning. We consider first-order existentially quantified formulas of the form $\forall\, x\; \exists\, y\; \Phi(x, y)$, where $x = x_1 \ldots, x_n$ and $y = y_1, \ldots, y_r$ are variables and $\Phi$ is any quantifier-free formula with equality as only predicate symbol.

## 1.2 An informal overview of our approach

McCarthy [7] and Burstall [6] recognized the importance of inductive reasoning in program verification and computer science. Burstall proposed a structural induction principle for recursively defined data structures. Since then, work has been intensively done in this field. Musser [14] used the Knuth-Bendix completion procedure to prove equations by induction from an equational specification of data types. This method has been called the inductionless induction, since it tries to get rid of induction.
Our approach relies on the notion of cover sets and uses axioms (stated as ordered rewrite rules) and previously proved conjectures to reduce the formula to be proved to a tautology or to a smaller (w.r.t a well-founded ordering) instance of the formula itself. Our method differs from the one proposed in [8] in the fact that we use cover sets and ordered rewriting instead of test sets and traditional term rewriting. The contributions of this paper are threefold: First, we can now deal with interesting equational theories for which terminating term rewriting systems do not exist and where the method proposed in [8] fails. Second, we enhance the flexibility of the method by allowing the user to choose the kind of induction principle he wants to apply. Depending on the choice of the cover set he can perform either structural induction or term rewriting induction [16]. Third, we present an algorithm to synthesize a recursive definition of a skolem function for an existentially quantified formula.

Let us describe in a few words how we proceed in attempting to prove a existentially quantified formula in the initial model of a given set E of equations. E will be divided into two parts: a set of rewrite rules $R$ and a set of equations $E_1$. The rewrite rules

serve to define functions and are used in the evaluation of terms. The equations represent additional knowledge about the problem domain which may be used for the proof and the program synthesis. Ordered rewriting allows us to use such equations, whenever the term obtained after the application of the considered equation is smaller (w.r.t a well-founded ordering) than the one before application.
The universally quantified variables of the formula to be proved are replaced with the terms of the cover set and the obtained formulas are then reduced using the rewrite rules from $R$ or/and by equations from $E_1$. If the formulas are in "solved" form (e.g. tautology or smaller instance) then we are done, else we proceed instantiating either universally or existentially quantified variables with elements of the cover set. The simplification process is then repeated till all relevant formulas are transformed into formulas in "solved" form.

The remainder of the paper is organized as follows. Section 2 introduces the essential notions used throughout this paper. Section 3 discusses two theorems for inductive existence proofs based on different notions of cover sets. In section 4, we consider an example and compare our method with other well-known inductive proof methods. Section 5 presents an algorithm for generating a recursive definition of a skolem function for an existentially quantified formula. We also give an example to illustrate the way we synthesize recursive function definition from existence proofs.

## 2 Basic notions

We assume that the reader is familiar with the definitions used in the field of term rewriting. We adopt the notations used in [8].
Let $X$ be a set of variables and $F$ a signature of function symbols. Let $T(F,X)$ denote the set of terms built from $F$ and $X$ and $GT(F)$ the set of ground terms. An equation is a pair written as $l = r$. A specification is a triple $SP = (F_k, F_d, E)$, where $F_k$ and $F_d$ are two disjoint sets of function symbols and $E$ a set of equations over $T(F_k \cup F_d, X)$. $F_k$ is called the set of constructors of $SP$ and $F_d$ the set of defined function symbols. $E$ will be divided in two sets $E_1$ and $E_2$, where the equations of $E_1$ serve to define the functions symbols of $F_d$ and can be transformed into a terminating term rewriting system $R$. $E_2$ describes additional knowledge about the problem domain. A term rewriting system $R$ associated with an equational theory $E$ is a finite set of rewrite rules $R = \{l_i \to r_i\}_{i=1}^{n}$, such that $\{l_i = r_i\}_{i=1}^{n}$ and $E$ are equivalent (i.e, $s = t$ is true in $\{l_i \to r_i\}_{i=1}^{n}$ if and only if $s =_E t$). Given a set of equations $E$, we recall that a Herbrand model of $E$ is a model of $E$ whose domain is the set of ground terms. An equation $s = t$ is a logical consequence of $E$ iff it is valid in all models of $E$. This will be denoted by $E \models s = t$. We denote by $=_E$ the smallest monotonic congruence that contains $E$ ($s =_E t \Leftrightarrow E \models s = t$). From the well

known soundness and completeness theorem of Birkhoff we get: $E \vdash s = t \Leftrightarrow E \models s = t$. The initial model $I(E)$ is defined to be the quotient of the term algebra by the congruence $=_E$, restricted to ground terms. A reduction ordering is a transitive monotonic relation on terms which is terminating and stable under substitutions. Two well-founded orderings $>_1$ and $>_2$ are compatible if $(>_1 \cup >_2)^+$ is well-founded, or, equivalently, if they are both included in a well-founded order.

## 2.1 Ordered Term rewriting systems

Ordered term rewriting systems allow us to overcome the limitations that are traditionally caused by termination criteria. They are based on the idea that unoriented equations can also be used for rewriting, provided that they are used along a reducing direction. As a consequence, we can now deal with symmetric equations, which are not orientable as rewrite rules. Commutativity equations are well-known examples of this.

**Definition 1** A term t[u] is reducible by an equation $l = r$ if there is a substitution $\sigma$ such than

- $\sigma(l) = u$, and
- $\sigma(l) > \sigma(r)$

where $<$ is a ground-total simplification ordering on $T(F, X)$ (i.e., a monotonic and stable ordering, which has the subterm property and is total on ground terms).

An ordering $>$ has the subterm property, if $t > s$ whenever s is a proper subterm of t.

**Definition 2** An ordered term rewriting system consists of a terminating term rewriting system (w.r.t a ground-total simplification ordering) and a possibly empty set of equations.

An ordered term rewriting sequence is then a sequence of traditional and ordered term rewriting steps. Since the ordering is well-founded, no term can be reduced infinitely. The totality condition implies ground confluence of the ordered term rewriting system and can be dropped when the set $E$ of equations is empty, since ground confluence is not necessary for the correctness of our proof method. Ground confluence will be necessary if we want to synthesize programs with deterministic results.

## 2.2 Inductive theorems and cover sets

An equation $l = r$ is an inductive theorem of a given set of axioms $E$, written $E \models_{ind} l = r$, if and only if, for every ground substitution $\sigma$, the equation $\sigma(l) = \sigma(r)$ is a logical consequence of $E$.

We take the following to be our operational definition of inductive existence theorems (see [8]).

**Definition 3** A formula of the form $\forall x \exists y \Phi(x, y)$ is an inductive theorem of a given set of equations $E$ if for all ground substitutions $\sigma$ there exists a ground substitution $\nu$ such that $\Phi(\sigma(x), \nu(y))$ is a logical consequence of $E$.

The above definition is not effective because it involves testing all ground substitutions. In order to develop an effective method, we consider finite sets of nonground terms that can cover all ground cases. This is formalized in the following definition.

**Definition 4** Let $E$ be a set of equations and $>$ a stable, well-founded order. A set of terms $\{t_i\}$ of sort s is a $>$-cover set for s (with respect to $E$) if, for every ground term g of sort s, there is a $t_i$ and a substitution $\sigma$ such that $g =_E \sigma(t_i)$ and $g \geq \sigma(t_i)$.

We can easily generalize this notion for n-tuples of terms of possibly different sorts. A cover set $M$ for $s_1 \times \ldots \times s_n$ is then a set of n-tuples of terms from $s_1, \ldots, s_n$, such that for every n-tuple $< t_1, \ldots, t_n >$ of ground terms there exists a n-tuple $< g_1, \ldots, g_n >$ from $M$ and a substitution $\sigma$, such that $t_i =_E \sigma(g_i)$, and $t_i \geq g_i$, $1 \leq i \leq n$.

**Example 1** Let $>_{rpo}$ be the recursive path ordering [10] generated by the precedence order $s > + > 0$. Then $M = \{0, s(0), x + y\}$ is a $>_{rpo}$-cover set for natural numbers with the following equations defining the addition.

$(N_1)$ $x + 0 = 0$
$(N_2)$ $x + s(y) = s(x + y)$

Several notions of cover sets have been proposed [2], [16], [4]. We will see another definition of cover sets, which is based on the so-called "cover functions" and was first introduced in [17]. It may be hard sometimes to find a suitable cover set. As proposed in [4], the function symbols in the conjecture and their definitions offer an insight into the problem.

## 3 Two main theorems

We now formulate two theorems, which show how to prove existentially quantified formulas in the initial model of a set of equations. The two theorems are based on different notion of cover sets.

**THEOREM 1**: Let $>_1$ and $>_2$ be two well-founded and compatible orders on $T(F,X)$. Let $E$ be a set of equations and $T$ an ordered term rewriting system associated to $E$, ($T$ consists of a terminating term rewriting system $R$ and a possibly empty set $E_1$ of equations). Let $M$ be a $>_1$-cover set for $s_1 \times \ldots \times s_n$. The following formulas:

$$\forall x_1, \ldots, x_n \exists y_1, \ldots, y_r \Phi_i(x_1, \ldots, x_n, y_1, \ldots, y_r) \ (i = 1, \ldots, l)$$

are inductive theorems, if

- For all $\Phi_i$ and for all substitutions $\sigma$ with $\sigma(x) \in M$, there exists a substitution $\mu$ such that $\mu(y_j)$, is either a element from the cover set for the sort of $y_j$ or a variable of the same sort, $(1 \leq j \leq r)$, and such that

$$\Phi_i(\sigma(x), \mu(y)) \longrightarrow_T^* \Psi_i, \text{ where } \Psi_i \text{ is either}$$

1. an equational tautology (say $z = z$), or
2. a formula of the form $\forall x' \exists y' . y' = H(x')$ (explicit form)
3. a formula $\Phi_k(\alpha(x), y')$, with a substitution $\alpha$ such that $\alpha(x) \ll_2 \sigma(x)$, where $\ll_2$ is the multiset extension of $<_2$.

We should mention that the above theorem remains sound even if the term rewriting system $R$ does not terminate. For the sake of simplicity we prove the theorem with formulas containing only one universal and one existential variable.

**Proof**: We show that for any ground term s the following holds:

$$\forall i \exists t \in GT(F) \ E \vdash \Phi_i(s,t) \ (*)$$

We prove (*) by induction on ground terms over the well-founded order $<$, in which the orders $<_1$ and $<_2$ are both included (we know that such an order exists, since $<_1$ and $<_2$ are compatible). We now assume that (*) holds for any $s'$ such that $s' < s$ and prove that

(*) holds for $s$.
Since $M$ is a $>_1$-cover set for the sort of $s$, there is a cover set term $s_1 \in M$ and a ground substitution $\sigma$ such that $\sigma(s_1) =_E s$ and $s \geq_1 \sigma(s_1)$. By hypothesis there exists for all $i$ a term $T$ such that

$$(^{**})\ \Phi_i(s_1, T) \longrightarrow^* \Psi_i$$

and after application of $\sigma$

$$(^{***})\ \Phi_i(\sigma(s_1), T) \longrightarrow^* \sigma(\Psi_i)$$

*Consider now the following cases:*

1. $\Psi_i$ is an equational tautology. Then $\Phi_i(\sigma(s_1), T)$ is also an equational tautology and $\Phi_i(s, T)$ too. Hence, (*) holds by taking for t any ground instance of $T$.

2. $\Psi_i$ is of the form $\forall x' \exists y'.y' = H(x')$, which is an equational theorem and consequently $\Phi_i(\sigma(s_1), T)$ and $\Phi_i(s, T)$ too. We can then prove (*) by taking for $t$ the term $T[y'/H(\sigma(x'))]$.

3. $\Psi_i$ is of the form $\Phi_k(w, y')$ and $w <_2 s_1$. The stability of $<_2$ implies that $s' = \sigma(w) <_2 \sigma(s_1)$ and since $<_1$ and $<_2$ are both included in $<$, we have: $s' = \sigma(w) < \sigma(s_1) < s$. From the induction hypothesis we know that there exists a term $t'$, such that $E \vdash \Phi_k(s', t')$. Hence, after application of the ground substitution $\eta : y' \mapsto t'$ to (***) we get:

$$\Phi_i(\sigma(s_1), \eta(T)) \longrightarrow^* \Phi_k(\sigma(w), \eta(y')) = \Phi_k(s', t')$$

Hence $\Phi_i(\sigma(s_1), \eta(T))$ is an equational theorem and $\Phi_i(s, \eta(T))$ too. Therefore we can prove (*) by taking $\eta(T)$ for $t$. □

The first two cases of the theorem can be seen as the base cases and the last one as the counterpart of the induction hypothesis in a traditional proof by induction. If the set of equations $E_1$ is empty and if we choose (for $<_2$) the decreasing order generated by the term rewriting relation then, the above theorem is reduced to term rewriting induction [16]. The decreasing order generated by a reduction order $>$ is $(> \cup \rhd)^+$, where $\rhd$ is the strict super term order.

The next example illustrates the importance of ordered term rewriting within our proof method. Let us first give an informal description of the way the above theorem can be used to find proofs. First, we replace the universal variables of the formula to be proved with the terms of the cover set. Then, we use the rewrite rules from $R$ or/and the equations from $E_1$ to simplifiy the instantiated formulas. If the simplified formulas are of one of the forms mentioned in the theorem (e.g., tautology or a smaller instance of the conjecture itself) we are done. If not, we replace the existential variable with cover set terms and try to simplify the obtained formulas with the ordered term rewriting system. If one of these formulas is already in one of the cases 1., 2. or 3. of the theorem, we are done. Otherwise, we apply again the same procedure to the obtained formulas. The termination of the process is not guaranteed, but whenever it terminates our theorem ensures the validity of all lemmas $\Phi_i$ obtained during the proof.

**Example 2** let $>_{lpo}$ be the lexicographic path order generated by the precedence order $0 < s < +$. Then $M = \{0, s(x)\}$ is a $>_{lpo}$-cover set for nat. Consider now the following ordered term rewriting system $E$, which consists of $R = \{0+x \to x,\ S(x)+y \to S(x+y)\}$ and the equation $E_1 = \{x + y = y + x\}$. We assume that the conjecture C: $\forall x, y\ S(x) = S(y) \Rightarrow x = y$ has been already proved.

Let P(x): $\forall x \exists y. S(x) = S(y) + 0$ be the formula to be proved.
We replace the universal variable with each term of the cover set and get:
$P(0) : \exists y. S(0) = S(y) + 0$
$P(S(x_1)) : \forall x_1 \exists y. S(S(x_1)) = S(y) + 0$
Since $0 + S(y) <_{lpo} S(y) + 0$ we can use $E_1$ to get:
$P'(0) : \exists y. S(0) = 0 + S(y)$
$P'(S(x_1)) : \forall x_1 \exists y. S(S(x_1)) = 0 + S(y)$
Using $R$ and C these two formulas are reduced to:
$P''(0) : \exists y. 0 = y$
$P''(S(x_1)) : \forall x_1 \exists y. S(x_1) = y$
which satisfy respectively the cases 2. and 3. of the theorem. Therefore, $P(x)$ is true in the initial model of $E$.

The above example is trivial, but it cannot be proved without using the commutativity of the addition. Since commutativity equations are symmetric, orienting them in either direction results in infinite rewrite sequences. Ordered rewriting allows us to get rid of this situation without being forced to reflect the commutativity property of operators in all rewrite rules where they occur.
We now introduce another definition of cover sets, which incorporates additional information via the so-called "cover functions". The following cover set definition generalizes

the one in [17] by allowing the description of ground terms instead of ground constructor terms of a sort s.

**Definition 5** Let $GT(F)_s$ be the set of all ground terms of sort s and $M$ be a finite set of n-tuples of terms (taken from $(T(F,X)_{s_1} \times \ldots \times T(F,X)_{s_n})$)

- We say a mapping $\Psi : M \longrightarrow \mathcal{P}(GT(F)_{s_1} \times \ldots \times GT(F)_{s_n})$ is a cover function for $s_1 \times \ldots \times s_n$ upon $M$ if
  1. **Completeness** for any n-tuple of ground terms $T_1 =< t_1, \ldots, t_n >$ there is an n-tuple $T_2 =< m_1, \ldots, m_n > \in M$, such that $T_1 \in \Psi(T_2)$ and a substitution $\sigma$ such that $t_i =_E \sigma(m_i) \quad i = 1, \ldots, n$
  2. **Minimality** $\Psi(T_2) \neq \emptyset$ for $T_2 \in M$
  3. **Uniqueness** $T_2$ in (1) is unique.
- $M$ is said to be a cover set for $s_1 \times \ldots \times s_n$ if $M$ possesses a cover function defined above.

A comparison of the different notions of cover sets and their respective induction principles is beyond the scope of this paper.
Let us now introduce a relation on terms which is necessary for the soundness of the cover set (as defined above) induction principle.

**Definition 6** Let $>$ be a reduction ordering, E a set of equations and G a set of ground terms. A term $t_1$ is said to be inductively greater than $t_2$ with respect to G if for any ground substitution $\sigma$ and for any term $t \in$ G,

$$t =_E \sigma(t_1) \Rightarrow t > \sigma(t_2).$$

This is denoted by $t_1 >^i t_2$ with respect to G.

Let $SubT(t =< t_1, \ldots, t_n >, \ \Psi(t))$ denote the n-tuples of terms such that for any term $s =< s_1, \ldots, s_n > \in SubT(t =< t_1, \ldots, t_n >, \ \Psi(t))$ there is $j$, $1 \leq j \leq n$, such that $s_i = t_i$, $1 \leq i < j$ and $t_j >^i s_j$ with respect to $\Psi(t)$ and $s_j$ is a strict subterm of $t_j$.

We now formulate another theorem which relies on the last definition of cover sets.

**THEOREM 2**: Let $>$ be a reduction order on $T(F, X)$. Let $E$ be a set of equations and $T$ the ordered term rewriting system associated to $E$. Let $M$ be a cover set for $s_1, \ldots, s_n$ with cover function $\Pi$. The following formulas:

$$\forall x \exists y \Phi_i(x,y) \quad (x = x_1 \ldots x_n \ , y = y_1 \ldots y_m) \ and \ i = 1, \ldots, l$$

are inductive theorems, if

- For all $\Phi_i$ and for all substitutions $\sigma$ with $\sigma(x) \in M$, there exists a substitution $\mu$ such that $\mu(y_j)$ is either a element from the cover set for the sort of $y_j$ or a variable of the same sort, $(1 \leq j \leq r)$ and such that

  $$\Phi_i(\sigma(x), \mu(y)) \longrightarrow_T^* \psi_i, \text{ where } \Psi_i \text{ is either}$$

  1. an equational tautology (say z = z), or
  2. a formula of the form $\forall x' \exists y' . y' = H(x')$ (explicit form)
  3. a formula $\Phi_k(\alpha(x), y')$, with a substitution $\alpha$ such that $\alpha(x) \in SubT(\sigma(x), \Pi(\sigma(x)))$

*Proof*: analogous to the proof of theorem 1.

The above theorem can be seen as a generalized version of the structural induction principle. If the cover set contains only constructor terms and if there is no relation between the constructors then, we get the structural induction principle as a special case.

## 4 An example

Consider the following rewrite rules for the definition of append and reverse functions for lists:
$R = \{ \ append(nil, \ l) \rightarrow l$
$append(cons(a,l), m) \rightarrow cons(a, append(l,m))$
$reverse(nil) \rightarrow nil$
$reverse(cons(a,m)) \rightarrow append(reverse(m), cons(a, nil))\}$
Let $>_{rpo}$ the recursive path ordering generated by the precedence order $reverse > append > cons > nil$. $R$ is obviously terminating (w.r.t $>_{rpo}$). We also assume the following lemmas:
$(L1) : \forall a : El, m : list.reverse(append(m, cons(a, nil))) = cons(a, reverse(m))$
$(L2) : \forall a, b, c, d : El, l, r : list.cons(a, append(l, cons(b, nil))) = cons(c, append(r, cons(d, nil)))$
$\Longrightarrow a = c \wedge l = r) \wedge b = d$
Consider the following theorem:

$$(*) \quad \forall l_1 : list \ \exists l_2 : list.reverse(reverse(l_1)) = reverse(l_2)$$

where the witness for $l_2$ is clearly going to be $reverse(l_1)$, but we have to find a suitable cover set in order to prove (*). Consider the following cover set

$M = \{nil, cons(a, nil), cons(x_1, append(l', cons(x_2, nil)))\}$ with the following cover function:

$\Psi : M \longrightarrow \mathcal{P}(T_\Sigma)$
$\Psi(nil) = nil,$
$\Psi(cons(a, nil)) = \{l \mid |l| = 1\},$
$\Psi(cons(x_1, append(l_1, cons(x_2, nil)))) = \{l \mid |l| > 1\}$
Wir first replace the universal variable $l_1$ with the cover set elements:

- $\exists l_2 : list.reverse(reverse(nil)) = reverse(l_2)$
- $\forall x : El\ \exists l_2 : list.reverse(reverse(cons(x, nil))) = reverse(l_2)$
- $\forall x_1, x_2 : El, l : list\ \exists l_2 : list.reverse(reverse(cons(x_1, append(l, cons(x_2, nil))))) = reverse(l_2)$

These formulas are reduced (using the rules in R and $(L1)$ ) to:

- $\exists l_2 : list.nil = reverse(l_2)$ (I)

- $\forall x : El,\ \exists l_2.cons(x, nil) = reverse(l_2)$ (II)

- $\forall x_1, x_2 : El, l : list.cons(x_1, append(reverse(reverse(l)), cons(x_2, nil))) = reverse(l_2)$ (III)

Since (I), (II) and (III) cannot be reduced, we substitute the cover set values $nil, cons(y, nil)$ and $cons(y_1, append(l', cons(y_2, nil)))$ for the existential variable $l_2$ respectively in (I), (II) and (III). After simplification we get:

- $nil = nil$
- $\forall x : El\ \exists y : El.cons(x, nil) = cons(y, nil)$
- $\forall x_1, x_2 : El, l : list\ \exists y_1, y_2 : El, l' : list.cons(x_1, append(reverse(reverse(l)), cons(x_2, nil))) = cons(y_2, append(reverse(l'), cons(y_1, nil)))$.

The first formula is an equational tautology. The second is also a tautology if we take x for y. After applying $(L2)$ to the third formula, we get:

$\forall x_1, x_2 : El, l : list\ \exists y_1, y_2 : El, l' : list.\ x_1 = y_2 \wedge reverse(reverse(l)) = reverse(l') \wedge x_2 = y_1$

which consists of two formulas in the explicit form mentioned by case 2 of theorem 2 and a formula which is a smaller instance of (*), since $l$ is a proper subterm of $l_1$ and $l_1 >^i l$ holds. Therefore, (*) is true in the initial model of $R$.

**Remark:** The above formula cannot be proved by taking $\{nil, cons(x, l)\}$ as cover set. This illustrates the fact that our proof method is a generalized version of the structural induction principle (see also [17]). We should also mention that the test set method would fail trying to prove (*), since $\{nil, cons(x, l)\}$ is an inappropriate test set for $R$.

# 5 Generating a skolem function

In this section we show how to generate recursive definitions of skolem functions for existentially quantified formulas. These generated functions (or programs) are in fact by-products of existence proofs constructed using the proof method proposed in this paper. Before introducing the algorithm, we first restrict the definitions of cover sets in order to avoid possible inconsistencies in the definition of the functions we will be generate. The following restrictions are now placed on the definitions of cover sets:

1. The given well-founded ordering must be total on ground terms.
2. For any ground term $t$ there exists a unique cover set term $t_1$ and a substitution $\sigma$, such that $t =_E \sigma(t_1)$ and $\sigma(t_1)$ is ground and minimal by the given ordering.

These restrictions guarantee that the generated programs are ground confluent. Ground confluence means that the results of the programs are deterministic.

## 5.1 The algorithm

Let $M = \{c_1, \ldots, c_m\}$ be a cover set of $s_1 \times, \ldots, \times s_k$. Out of a proof of an existentially quantified formula $\forall x \exists y \; \Phi(x, y)$ (where $x = x_1, \ldots, x_k$ and $y = y_1, \ldots, y_n$) by means of theorem 1 or theorem 2 we can take n-tuples of terms $t_1, \ldots, t_m$ such that,

$$\Phi(c_j, t_j) \longrightarrow^* \Psi_j, \; j = 1, \ldots, m$$

Consider now each $\Psi_j$ and let $f : s_1 \times \ldots \times s_k \longrightarrow s$ (where $s$ is the sort of $t_{j_i}$ and $i = 1, \ldots, n$) be the following function:

1. $f(c_j) = t_{j_i}$, if $\Psi_j$ is an equational tautology (say z=z) and where $t_{j_i}$ must be closed, since we want to synthesize functions and not relations.

2. $f(c_j) = \gamma(t_{j_i})$, if $\Psi_j$ is of the form $\forall x^{'} \exists y^{'}.y^{'} = H(x^{'})$ and where $\gamma$ is the substitution $\gamma : y^{'} \longrightarrow H(x^{'})$.

3. $f(c_j) = \gamma(t_{j_i})$, if $\Psi_j$ is of the form $\Phi(\alpha(x_1, \dots, x_k), y^{'})$ where $\gamma : y^{'} \longrightarrow f(\alpha(x_1, \dots, x_k))$, $\alpha(x_1, \dots, x_k) \ll c_j$ and where $\ll$ is the multiset extension of the given well-founded ordering.

1 and 2 are the base cases, whereas 3 represents the recursion case of the definition of the generated skolem function. Functions which are constructed by means of the above algorithm represent the computational content of existence proofs derived using the proof method proposed in this paper.

**THEOREM 3**: If we can prove the validity of an existentially quantified formula by means of theorem 1 or theorem 2 then, the function which is derived using the above algorithm satisfies the existence formula.

**Proof**: Let $E$ be the given set of equations and $f$ the function, which was derived from $\forall x \exists y\, \Phi(x, y)$ (*) by means of the above algorithm. The function $f$ satisfies (*) if $\Phi(x, f(x))$ is an inductive theorem of $E$. i.e., $\forall t \in GT(F)\ E \vdash \Phi(t, f(t))$. From the proof of (*) we can take a term $T \in GT(F)$ such that, $E \vdash \Phi(t, T)$. Therefore, $\Phi(t, f(t))$ is an equational theorem, since $f(t)$ is an instance of $T$ for any $t \in GT(F)$. □

– The synthesized function is completely defined, since the completeness of the cover sets is guaranteed by definition.

– The function is terminating, since her arguments are smaller (w.r.t the given well-founded ordering) after each recursive call.

– The function is well defined, since the cover set is "non-overlapping".

**Example 3** Consider the example of section 4. Using the above algorithm we can generate the following function $f : list \longrightarrow list$:

- $f(nil) = nil$
- $f(cons(x, nil)) = cons(x, nil)$
- $f(cons(x_1, append(l, cons(x_2, nil)))) = cons(x_2, append(f(l), cons(x_1, nil)))$

Through our proof and synthesis method we have generated a new version of the reverse function, which is more efficient than the initial one. Depending on the choice of the cover set, we can synthesize efficient programs. An important aspect of our method is that it can be used to transform programs by transforming (using another cover set) their respective proofs.

## 6 Conclusion

We have presented a method for proving the validity of existence formulas in the initial model of a given set of equations. This method generalizes the one developed in [8] by using ordered term rewriting and cover sets. As a consequence, we can deal now with interesting equational theories for which terminating term rewriting systems do not exist. Cover sets, as opposed to test sets, improve the flexibility and the power of the method. Suitable cover sets may be (automatically) constructed from completely defined functions. We have also proposed in this paper an algorithm which synthesizes a recursive function out of an existence proof. We plan to investigate the relationship between the different notions of cover sets, test sets and their respective induction principles. We are also insterested in finding methods to automate the determination of cover sets.

## References

[1] Aubin, J. (1976) Mechanizing structural induction. Ph.D Thesis, University of Edinburgh, Edinburgh.

[2] Bachmair, L. (1988) Proof by consistency in equational theories. Proc. 3th Symposium on Logic in Computer Science, Edinburgh.

[3] Basin, D.A., and Walsh, T. (1994) Termination Orderings for Rippling Proc. 12th CADE, LNCS vol.814

[4] Boyer, R.S. and Moore, J S. (1979) A computational logic. Academic Press, New York

[5] Bundy, A., Stevens, A., van Harmelen, F., Ireland, A., and Smaill, A. Rippling: A heuristic for guiding inductive proofs. Artificial Intelligence, 62:185-253.

[6] Burstall, R. (1969) Proving properties of programs by structural induction Computer Journal 12(1), 41-48.

[7] McCarthy, J. (1963) A Basis for a mathematical theory of computation Computer Programming and formal systems, P. Braffort and d. Hirschberg [ed.], North-Holland, Amsterdam.

[8] Chazarain, J., Kounalis, E. (1994). Mechanizable Inductive Proofs for a Class of ∀∃ formulas Proc. 12th CADE, LNCS vol. 814.

[9] Dershowitz, N. (1983) Applications of the Knuth-Bendix Completion Procedure. Laboratory Operation, Aerospace Corporation, Aerospace report No. ATR-83(8478)-2.

[10] Dershowitz, N. (1985) Termination, In Proc. of the International Conference on rewriting techniques and Applications, Dijon, France.

[11] Hsiang, J., Rusinowitch, M. (1987) On word problems in equational theories. In Ottmann, T. ed.,14th Intern. Colloq. Automata, Languages and Programming , volume 267 of Lect. Notes in Comp. Science, 54-71. Springer-Verlag.

[12] Huet, G., and Hullot, J.M.(1982) Proofs by induction in equational theories with constructors. Journal of Comp. and System Sciences, 25(2), 1982.

[13] Hutter, D. (1989) Guiding inductive proofs. In M.E. Stickel, editor, 10th International Conference of Automated Deduction, LNCS vol. 489.

[14] Musser, D. 1980. On proving inductive properties of abstract data types. In Proc. 7th ACM Symp. on Principles of Programming Languages, 154-162. Las Vegas, Nevada.

[15] Manna, Z., Waldinger, R. (1980) A deductive Approach to Program synthesis ACM Trans. Program. Lang. Systems, 2(1), 90-121

[16] Reddy, U. (1989) Term rewriting induction. Proc. 10th CADE, LNCS vol. 489.

[17] Zhang, H., Kapur, D., and Krishnamoorthy, M.S. (1988) A Mechanizable Induction Principle for equational specifications In E.Lusk and R. Overbeek, editors, 9th CADE Conf.,LNCS vol 310.