PROCEEDINGS OF THE INTERNATIONAL CONFERENCE
ON FIFTH GENERATION COMPUTER SYSTEMS 1992,
edited by ICOT. © ICOT, 1992

Abduction in Logic Programming with Equality

P.T. Cox, E. Knill, T. Pietrzykowski

Technical University of Nova Scotia, School of Computer Science
P.O. Box 1000, Halifax, Nova Scotia
Canada B3J 2X4

Abstract

Equality can be added to logic programming by using surface deduction. Surface deduction yields interpretations of unification failures in terms of residual hypotheses needed for unification to succeed. It can therefore be used for abductive reasoning with equality. In surface deduction the input clauses are first transformed to flat form (involving no nested terms) and symmetrized (if necessary). They are then manipulated by binary resolution, a restricted version of factoring and compression. The theoretical properties of surface deduction, including refutation completeness and a weak deductive completeness (relative to equality), are established in [Cox *et al.* 1991]. In this paper we show that weak deductive completeness implies that surface deduction will yield essentially all hypotheses when used as an abductive inference engine. The full characterization of equational implication for goal clauses given in [Cox *et al.* 1991] is shown to yield a uniquely defined equationally equivalent residuum for every goal clause. The residuum naturally represents the corresponding abductive hypothesis. An example illustrating the use of surface deduction in abductive reasoning is presented.

1 Introduction

In abductive reasoning, the task is to explain a given observation by introducing appropriate hypotheses ([Cox and Pietrzykowski 1987], [Goebel 1990]). Most presentations of abduction do not include reasoning with equality, nor do they allow the introduction of equality assumptions to explain an observation. A notable exception is E. Charniak's work on motivation analysis [Charniak 1988]. Charniak allows the introduction of certain restricted equality assumptions to determine motivations for observed actions. He shows that the introduction of such equality assumptions is required to successfully abduce motivations. In this paper we consider the problem of abductive reasoning with Horn clauses in the presence of equality. We show that surface deduction has the necessary properties for use in an abductive inference system.

In the presence of equality, an abduction problem consists of a theory $\mathcal{T}$ and a formula O (the *observation*). An *explanation* of $(O, \mathcal{T})$ is a formula E consistent with $\mathcal{T}$ such that E together with $\mathcal{T}$ equationally implies O. We will assume that O and E are existentially quantified conjunctions of facts and that $\mathcal{T}$ is a Horn clause theory.

One way to obtain an explanation E, given an observation O and a theory $\mathcal{T}$, is to deduce $\neg E$ from $\mathcal{T}$ and $\neg O$. Since explanations with less irrelevant information are preferred, it is sufficient to deduce a clause $\neg E'$ such that $\neg E'$ implies $\neg E$. Intuitively, E' is at least as good an explanation as E (see Section 4). It follows that a deduction system adequate for abductive reasoning should satisfy a weak deductive completeness: If the theory $\mathcal{T}$ implies a non-tautological clause $\neg E$, then we must be able to deduce a clause $\neg E'$ from $\mathcal{T}$ such that $\neg E'$ implies $\neg E$. In the absence of equality, SLD-resolution (see [Lloyd 1984]) satisfies this condition.

The problem of introducing equality to Horn clause logic has been well-studied, see [Hõlldobler 1989] for an excellent overview. The simplest approach to this problem involves adding the equality axioms (which are Horn clauses) to the set of input clauses. However, unrestricted use of these axioms results in inefficiency. Furthermore, this approach does not yield any insights into the degree to which the equality axioms are needed. Paramodulation and other term rewriting systems do not explicitly introduce new equality assumptions into derivations and therefore do not satisfy the weak deductive completeness condition. Other approaches, such as the ones in [van Emden and Lloyd 1984] and extended in [Hoddinott and Elcock 1986] using the homogeneous form of clauses, require restricting the form of the input theory. Here, we use the results of [Cox *et al.* 1991] to show that if equality is introduced to Horn clause logic via surface deduction, all explanations for an abduction problem can be obtained.

In surface deduction, a set of input clauses is first transformed to a flat form and symmetrized. The deduction then proceeds using linear input resolution for Horn clauses (see [Lloyd 1984]) together with a limited use of factoring and a new rule called compression. The additional deduction rules are equivalent to those restricted uses of the reflexivity axiom ($x \doteq x$:-) which preserve flatness. They are required only at the end of a deduction.

A clause is flat if it has no nested functional expressions, and every variable which appears immediately to the right of an equality symbol ($\doteq$) appears only in such positions. A stronger version of flatness requires that in addition the clause is separated. This means that every variable appears at most once in any given literal and has only one occurrence inside a functional or relational expression. Symmetrization affects only those clauses with equalities in their heads (see Section 3).

The idea of using flattening to add equality to theorem proving is due to [Brand 1975] and is applied to logic prgramming in [Cox and Pietrzykowski 1986] where surface deduction is defined. Flattening is closely related to narrowing. In narrowing the process of flattening is implicit in the deduction rules. The relationship between the two methods is examined in [Bosco *et al.* 1988]. Separation of terms is implicit in the transformations to homogeneous forms of [Hoddinott and Elcock 1986]. The symmetrization method used here is similar to the one introduced in [Chan 1986] and does not increase the number of clauses in the theory.

In [Cox *et al.* 1991] it is shown that surface deduction satisfies a weak deductive completeness provided that the input clauses are first transformed to separated form. As an application of this result, equational implication for goal clauses is found to have a simple syntactic characterization analogous to subsumption.

Once an explanation E is obtained by surface deduction, in what form should E be presented? For example if $\neg E$ (the actual clause deduced) is given by

$$:- x \doteq a, y \doteq b, y \doteq c,$$

then $:- y \doteq b, y \doteq c$ is equationally equivalent to $\neg E$. Therefore the atom $x \doteq a$ is irrelevant and should be removed. In Section 4 it is shown that the characterization of equational implication for goal clauses given in [Cox *et al.* 1991] implies that for every goal clause C there is a uniquely defined equational residuum RES(C) which cannot be further reduced without weakening the corresponding explanation. The notion of equational residuum is related to that of prime implicates used in switching theory [Kohavi 1978], truth maintenance systems [Reiter and de Kleer 1987] and diagnoses [de Kleer *et al.* 1988]. RES(C) is an equational prime implicate of a flattening of C.

In Section 2 the terminology is established; in Section 3 surface deduction is defined and the completeness results needed for abductive reasoning are given. In Section 4 the formalism of abductive reasoning with surface deduction is discussed; and finally in Section 5 an example is presented of an abductive problem solved by using surface deduction.

2 Preliminaries

Familiarity with logic programming is assumed (see e.g. [Lloyd 1984]). As in [Hölldobler 1990], let $\doteq$ denote the equality predicate symbol. The usual equality symbol = is used exclusively for syntactic equality. If L is an atom and $C = \{M_1, \ldots, M_n\}$ is a set of atoms, then $L :- C$ denotes the Horn clause $L \vee \neg M_1 \vee \ldots \neg M_n$. In this expression, L is the *head* and C is the *body* of the clause. A clause of the form $:- C$ is a *goal* clause. The atoms of C are the subgoals of $:- C$. A clause of the form $L :-$ is a *fact*. If $C_1, \ldots, C_n$ are sets of atoms and C is the union of the C_i, then $L :- C_1, \ldots, C_n$ means $L :- C$. When possible, set notation is omitted for one-element sets.

If OP is an operation which maps clauses to clauses and $\mathcal{A}$ is a set of clauses, then $\mathrm{OP}(\mathcal{A}) = \{\mathrm{OP}(C) \mid C \in \mathcal{A}\}$. Let σ be a substitution. If $x_i\sigma = t_i$ for $i = 1, \ldots, n$ and $x\sigma = x$ for all other variables, then σ is denoted by $\{x_1 \leftarrow t_1, \ldots x_n \leftarrow t_n\}$. A substitution σ is *variable-pure* iff $x\sigma$ is a variable for every variable x.

The expression 'most general unifier' is abbreviated by 'mgu'. An equality is an atom of the form $s \doteq t$. Let $\mathcal{E}$ be the set of equality axioms other than $x \doteq x$:- . If $\mathcal{A}$ and $\mathcal{B}$ are sets of clauses, then $\mathcal{A}$ satisfies (or implies) $\mathcal{B}$ iff every model of $\mathcal{A}$ is a model of $\mathcal{B}$. $\mathcal{A}$ *equationally* satisfies (or implies) $\mathcal{B}$ iff $\mathcal{A} \cup \mathcal{E} \cup \{x \doteq x :- \}$ satisfies $\mathcal{B}$. $\mathcal{A}$ and $\mathcal{B}$ are (equationally) equivalent iff each (equationally) satisfies the other. $\mathcal{A}$ is equationally inconsistent iff $\mathcal{A}$ equationally implies the empty clause.

3 Surface Deduction

In surface deduction, a refutation of a set of input clauses procedes by first transforming the input clauses to a flat form and then refuting the result using resolution, factoring and compression. The transformation subsumes the equality axioms other than reflexivity. The rules of factoring and compression subsume reflexivity.

Definition. Let C be a clause and t a term. An occurrence of t on the left-hand side (right-hand side) of an equality $t \doteq s$ ($s \doteq t$) in C is a *root* (*surface*) occurrence of t in C. Every other occurrence of t is an *internal* occurrence of t. The term t is a *root term* of C iff it has a root occurrence in C. *Surface* and *internal terms* are defined analogously.

Definition. A clause C is *flat* iff

(i) every atom of C is of the form $P(x_1, \ldots, x_n)$, $x \doteq f(x_1, \ldots, x_n)$ or $x \doteq y$, and

(ii) no surface variable of C is a root or internal variable of C.

Definition. Let C be a Horn clause. An *elementary flattening* of C is obtained by either

(i) replacing some of the non-surface occurrences of a non-variable term t by a new variable y and adding the equality $y \doteq t$ to the body,

or

(ii) replacing some of the surface occurrences of a root or internal variable x of C by a new variable y and adding the equality $x \doteq y$ to the body.

An elementary flattening of the set of clauses $\mathcal{A}$ is obtained by replacing a clause in $\mathcal{A}$ by an elementary flattening of that clause.

Modifying a clause C by successive elementary flattenings eventually results in a flat clause (a *flattening* of C) which cannot be flattened any further (Theorem 2 of [Cox and Pietrzykowski 1986]).

Definition. Let C be a clause. Then FLAT(C) denotes a (arbitrary but fixed) flattening of C.

For any set of clauses $\mathcal{A}$, $FLAT(\mathcal{A})$ is equationally equivalent to $\mathcal{A}$. In [Cox *et al.* 1991] it is shown that the transformation FLAT subsumes the replacement axioms but not transitivity and symmetry.

In order to subsume transitivity and symmetry, we need another transformation.

Definition. Let C be a clause with an equality in its head. Then C is *symmetric* iff C is of the form

$$x \doteq u \text{ :- } x \doteq v, s \doteq v, y \doteq u, y \doteq t, M$$

for some terms s and t and set of atoms M, where x, y, u and v do not occur in M, s or t. The set of clauses $\mathcal{A}$ is *symmetrized* iff every clause C of $\mathcal{A}$ with an equality in its head is symmetric.

Definition. Let C be a Horn clause. If C does not have an equality in its head or if C is symmetric, then the *symmetrization* SYM(C) of C is C. If C is not symmetric and of the form $s \doteq t$:- M, then SYM(C) is given by

$$x \doteq u \text{ :- } x \doteq v, s \doteq v, y \doteq u, y \doteq t, M.$$

Note that if $\mathcal{A}$ is a set of Horn clauses, then SYM($\mathcal{A}$) is equationally equivalent to $\mathcal{A}$, and if $\mathcal{A}$ is flat, then SYM($\mathcal{A}$) is flat. In [Cox *et al.* 1991] it is shown that the transformation SYM subsumes transitivity and symmetry. In order to subsume replacement, transitivity and symmetry, the transformations SYM and FLAT are composed.

Flattening and symmetrization followed by SLD-resolution using resolution with $x \doteq x$:- as an additional deduction rule is refutation complete for logic programming with equality. However, weak deductive completeness is not satisfied [Cox *et al.* 1991]. In order to obtain weak deductive completeness an additional transformation is required.

Definition. A *positive (negative) root* occurrence of the term t in the clause C is a root occurrence in the head (body) of C.

Definition. The flat clause C is *separated* in the variable x iff

(i) every literal of C has at most one occurrence of x,

(ii) C has at most one internal occurrence of x, and

(iii) if x has an internal occurrence in C, then x has a negative root occurrence in C.

The clause C is separated iff C is separated in all its variables.

If $\mathcal{A}$ is a set of separated flat Horn clauses, then SYM($\mathcal{A}$) is separated. Separated clauses can be obtained from a given flat clause by using the transformation SEP:

Definition. Let C be a flat clause and x a variable. The clause SEP(C) is the separated flat clause obtained by applying the following transformation to C: For every variable x such that C is not separated in x, replace each internal occurrence of x by a new variable x_i and add the equalities $x \doteq y, x_1 \doteq y, x_2 \doteq y, \ldots$ to the body of C (where y is a new surface variable).

The rules of factoring and compression used in surface deduction are:

(i) *Root factoring*. The clause C' is a root factor of C iff C' is obtained by factoring two equalities of C with the same root variable.

(ii) *Surface factoring*. The clause C' is a surface factor of C iff C' is obtained by factoring two equalities of C with the same surface term.

(iii) *Root compression*. The clause C' is a root compression of C iff C' is obtained by removing an equality $x \doteq t$ from the body of C, where x has only one occurrence in C.

(iv) *Surface compression*. The clause C' is a surface compression of C iff C' is obtained by removing an equality $x \doteq y$ from the body of C, where y has only one occurrence in C.

A *compression* is a root or surface compression. A *compression* of a clause C is a clause C' obtained from C by a sequence of applications of compression rules.

The soundness of root and surface factoring and compression (in the presence of equality) is shown in [Cox and Pietrzykowski 1986]. Observe that binary resolution, surface and root factoring and compression preserve flatness. The relationship between factoring, compression and resolution with the reflexivity axiom is determined by the following result (proved implicitly in [Cox and Pietrzykowski 1986] and explicitly in [Cox *et al.* 1991]; see also [Hoddinott and Elcock 1986]):

Theorem 3.1 *Let* $:- C$ *be a flat goal clause. If* $:- C'$ *is a flat goal clause obtained from* $:- C$ *by a sequence of binary resolutions with* $u \doteq u :-$, *then* $:- C'$ *can be obtained from* $:- C$ *by a sequence of root and surface factorings and compressions.*

Definition. Let $\mathcal{A}$ be a set of flat Horn clauses. The flat goal clause C is *S-deducible* from $\mathcal{A}$ iff C can be obtained from $\mathcal{A}$ by a sequence of binary resolutions, surface and root factorings and compressions. Note that we can assume that the deduction is linear. $\mathcal{A}$ is *S-refutable* iff the empty clause is S-deducible from $\mathcal{A}$.

To state the weak deductive completeness result for flat, separated and symmetrized clauses, we need the transformation defined next.

Definition. Let $:- C$ be a flat goal clause. Then $:- C$ is *reduced* iff $:- C$ has no surface variables and no two equalities of $:- C$ have the same right-hand sides. A flat reduced clause REDU($:- C$) is obtained from $:- C$ by factoring equalities with identical right-hand sides until all right-hand sides are distinct, and by removing all remaining equalities with surface variables by surface compression. Note that for every flat goal clause $:- C$, REDU($:- C$) is equationally equivalent to $:- C$.

Theorem 3.2 [Cox *et al.* 1991] *Let* $\mathcal{A}$ *be a set of Horn clauses and* $:- C$ *a goal clause. Then* $\mathcal{A}$ *equationally implies* $:- C$ *iff there is a flat goal clause* $:- C'$ *such that for some variable-pure substitution* σ, $:- C'\sigma \subseteq$ REDU(FLAT($:- C$)) *and* $:- C'$ *is S-deducible from* SYM(SEP(FLAT($\mathcal{A}$))).

As an application of this result, the following theorem is proved in [Cox *et al.* 1991]:

Theorem 3.3 *Let* $:- A$ *and* $:- B$ *be goal clauses. Then* $:- A$ *equationally implies* $:- B$ *iff there is a variable-pure substitution* σ *such that a compression of* FLAT($:- A$)σ *is included in* REDU(FLAT($:- B$)).

Definition. Let $:- C$ be a goal clause. An *equational residuum* of $:- C$ is a minimal subclause of REDU(FLAT($:- C$)) which is equationally equivalent to $:- C$.

Every equational residuum of $:- C$ is equationally equivalent to $:- C$. The fact that every subclause of a reduced clause is reduced implies that if $:- C'$ is a equational residuum of $:- C$, then $:- C'$ is reduced. The next theorem shows that the equational residuum is unique.

Theorem 3.4 [Cox *et al.* 1991] *Let* $:- A'$ *and* $:- B'$ *be equational residua of the goal clauses* $:- A$ *and* $:- B$ *respectively. Then* $:- A$ *is equationally equivalent to* $:- B$ *iff* $:- A'$ *is a variant of* $:- B'$.

4 Abduction using Surface Deduction

An *existential conjunction of facts* is a conjunction of facts with all its free variables quantified existentially. The abduction problem for Horn clause logic with equality can be stated as follows:

Abduction Problem: An abduction problem is a pair $(\mathcal{A}, O)$, where $\mathcal{A}$ is a *theory* of Horn clauses and O (the *observation*) is an existential conjunction of facts. An *explanation* of the abduction problem $(\mathcal{A}, O)$ is an existential conjunction of facts E consistent with $\mathcal{A}$ such that E and $\mathcal{A}$ equationally imply O.

Let $\neg O$ and $\neg E$ denote the disjunctions of the negations of the constituent facts of O and E respectively. Since E and $\mathcal{A}$ equationally imply O iff $\neg O$ and $\mathcal{A}$ equationally imply $\neg E$, a solution to an abduction problem can be obtained by deducing a clause C from $\mathcal{A}$ and $\neg O$, and negating C to obtain E.

In general, it is desirable for an explanation E of an abductive problem $(\mathcal{A}, O)$ to have certain additional properties (see [Cox and Pietrzykowski 1987]). For example, an explanation E should not contain any facts not required to yield the observation from $\mathcal{A}$. Thus if E and E' are explanations of $(\mathcal{A}, O)$ and E equationally implies E', E' is preferred over E. (Here 'preferred' is to be understood as 'at least as good as'.)

For abduction, a desirable property of a deduction system is that for every explanation E of an abductive problem $(\mathcal{A}, O)$, one can obtain an explanation preferred over E. The weak completeness result of Theorem 3.2 implies that surface deduction with separated clauses has this property.

Theorem 4.1 *Let* $(\mathcal{A}, O)$ *be an abductive problem. Then for every explanation* E *of* $(\mathcal{A}, O)$, *there is an explanation* E' *preferred over* E *such that* E' *is S-deducible from* SYM(SEP(FLAT($\mathcal{A}$))) $\cup$ {SEP(FLAT($\cup\{\neg O\}$))}.

Proof. This follows by Theorem 3.2 and the fact that $\neg O$ is a goal clause, so that it does not need to be symmetrized. ∎

Flattenings of a clause can be viewed as alternate representations of the clause's term structure and are therefore essentially equivalent. Without loss of generality we restrict our attention to explanations E such that $\neg E$ is flat (*flat* explanations).

If E and E' are explanations of $(\mathcal{A}, O)$ such that E equationally implies E' but is not equationally equivalent to E', then E' is strictly preferred over E. Given an explanation E of $(\mathcal{A}, O)$ there are many equationally equivalent existential conjunctions of facts, all of which are also explanations of $(\mathcal{A}, O)$. The preference criteria introduced so far do not distinguish among equationally equivalent explanations. Following the intuition that a "simpler" explanation should be preferred, we give a stronger definition of preference:

Definition. Let E and E' be flat explanations. Then E' is strictly preferred over E iff either E equationally implies E' but is not equivalent to E', or E is equationally equivalent to E' and E' has fewer atoms.

Given these preference criteria, we have the following theorem which determines the most preferred flat explanation among equationally equivalent ones:

Theorem 4.2 *For any explanation E, if E' is the negation of the equational residuum of $\neg E$, then E' is the unique most preferred flat explanation among flat explanations equationally equivalent to E.*

Proof. Let $:- A$ be a flat clause equationally equivalent to $\neg E$. If $:- A$ is not reduced, then REDU($:- A$) has fewer atoms than $:- A$ and the corresponding explanation is therefore strictly preferred. Assume that $:- A$ is reduced. If the equational residuum of $:- A$ is not given by $:- A$, then the equational residuum of $:- A$ has fewer atoms than $:- A$, so that the corresponding explanation is strictly preferred. The result now follows by the uniqueness theorem for equational residua, Theorem 3.4. ∎

5 An Application

Examples from the domain of story comprehension and motivation analysis which demonstrate the need for the inclusion of equality in abductive reasoning are given in [Charniak 1988]. Here we give an example from a different domain.

Consider the following (imaginary, but realistic) situation. A researcher X experimentally determines the value of a quantity associated with a physical object (e.g. the mass of an isotope of an element) and sends us the result. We have independently obtained a value for the same quantity (by theory and/or experiment) and our value differs from X's value. We believe our value to be correct and we would like to explain the discrepancy. We do not know the exact means by which X's value was obtained, but we know what kinds of experimental apparatus X might have used. One kind of apparatus (type A) is notorious for a hard-to-control drift in the settings which results in a systematic bias in the readings. Thus we can explain the discrepancy between our and X's values by hypothesizing that X used apparatus of type A with a systematic bias equal to the difference between the two values.

The situation is formalized as follows: Let $TA(x)$ mean that x is an apparatus of type A. Let $Vt(y)$ be the true value of quantity y, $Vm(z,y)$ the value of quantity y measured in experiment z, $A(u)$ the apparatus used in experiment u and $B(x)$ the systematic bias of apparatus x. The quantity measured by X is q, and the experiment performed by X is given the name e. With these definitions, our knowledge $\mathcal{T}$ consists of the clauses

T1: $Vt(q) \doteq 0 :-$

T2: $Vm(x_1,x_2) \doteq Vt(x_2) + B(A(x_1)) :- TA(A(x_1))$.

T3: $x_1 \doteq 0 + x_1 :-$

⋮

where knowledge about other types of apparatus and theorems about real numbers other than T3 have been omitted. The observation O is given by

O: $Vm(e,q) \doteq 2 :-$

The first task is to obtain a flattening of $\mathcal{T}$ and the negation of the observation:

fT1: $x_1 \doteq 0 :- x_1 \doteq Vt(x_2), x_2 \doteq q.$

fT2: $x_4 \doteq x_5 + x_6 :- TA(x_3),\ x_6 \doteq B(x_3),\ x_4 \doteq Vm(x_1,x_2),\ x_5 \doteq Vt(x_2),\ x_3 \doteq A(x_1).$

fT3: $x_1 \doteq x_2 + x_1 :- x_2 \doteq 0.$

fO: $:- x_1 \doteq 2,\ x_1 \doteq Vm(x_2,x_3),\ x_2 \doteq e,\ x_3 \doteq q.$

The clauses fT1 and fO are separated. Separated clauses for fT2 and fT3 are given by

sfT2: $x_4 \doteq x_5 + x_6 :- TA(x_3),\ x_6 \doteq B(x_7),\ x_3 \doteq x_8,\ x_7 \doteq x_8,\ x_4 \doteq Vm(x_1,x_2),\ x_5 \doteq Vt(x_{10}),\ x_2 \doteq x_9,\ x_{10} \doteq x_9,\ x_3 \doteq A(x_{11}),\ x_1 \doteq x_{12},\ x_{11} \doteq x_{12}.$

sfT3: $x_1 \doteq x_2 + x_3 :- x_3 \doteq x_4,\ x_1 \doteq x_4,\ x_2 \doteq 0.$

All clauses of $\mathcal{T}$ have equalities in their heads and need to be symmetrized. The fully transformed set of clauses is given by

T1′: $x_3 \doteq x_4 :- x_3 \doteq x_5,\ x_1 \doteq x_5,\ x_6 \doteq x_4,\ x_6 \doteq 0,\ x_1 \doteq Vt(x_2),\ x_2 \doteq q.$

T2′: $x_{13} \doteq x_{14} :- x_{13} \doteq x_{15},\ x_4 \doteq x_{15},\ x_{16} \doteq x_{14},\ x_{16} \doteq x_5 + x_6,\ TA(x_3),\ x_6 \doteq B(x_7),\ x_3 \doteq x_8,\ x_7 \doteq x_8,\ x_4 \doteq Vm(x_1,x_2),\ x_5 \doteq Vt(x_{10}),\ x_2 \doteq x_9,\ x_{10} \doteq x_9,\ x_3 \doteq A(x_{11}),\ x_1 \doteq x_{12},\ x_{11} \doteq x_{12}.$

T3′: $x_5 \doteq x_6 :- x_5 \doteq x_7,\ x_1 \doteq x_7,\ x_8 \doteq x_6,\ x_8 \doteq x_2 + x_3,\ x_3 \doteq x_4,\ x_1 \doteq x_4,\ x_2 \doteq 0.$

O′: $:- x_1 \doteq 2,\ x_1 \doteq Vm(x_2,x_3),\ x_2 \doteq e,\ x_3 \doteq q.$

The negation of the desired explanation can now be deduced from O′. In the deduction below, the literals involved in each step are underlined.

O′	:- $\underline{x_1 \doteq 2}, x_1 \doteq Vm(x_2, x_3), x_2 \doteq e, x_3 \doteq q.$
res. with T2′	:- $\underline{x_1 \doteq x_{18}}, \underline{x_7 \doteq x_{18}}, x_{19} \doteq 2, x_{19} \doteq x_8 + x_9, TA(x_6), x_9 \doteq B(x_{10}), x_6 \doteq x_{11}, x_{10} \doteq x_{11}, \underline{x_7 \doteq Vm(x_4, x_5)}, x_8 \doteq Vt(x_{13}), x_5 \doteq x_{12}, x_{13} \doteq x_{12}, x_6 \doteq A(x_{14}), x_4 \doteq x_{15}, x_{14} \doteq x_{15}, \underline{x_1 \doteq Vm(x_2, x_3)}, x_2 \doteq e, x_3 \doteq q.$
surf. fact. followed by root fact. and compr.	:- $x_{19} \doteq 2, x_{19} \doteq x_8 + x_9, TA(x_6), x_9 \doteq B(x_{10}), x_6 \doteq x_{11}, x_{10} \doteq x_{11}, \underline{x_8 \doteq Vt(x_{13})}, x_3 \doteq x_{12}, x_{13} \doteq x_{12}, x_6 \doteq A(x_{14}), x_2 \doteq x_{15}, x_{14} \doteq x_{15}, x_2 \doteq e, x_3 \doteq q.$
res. with T1′	:- $x_{19} \doteq 2, x_{19} \doteq x_8 + x_9, TA(x_6), x_9 \doteq B(x_{10}), x_6 \doteq x_{11}, x_{10} \doteq x_{11}, x_8 \doteq x_{24}, x_{20} \doteq x_{24}, \underline{x_{25} \doteq Vt(x_{13})}, x_{25} \doteq 0, \underline{x_{20} \doteq Vt(x_{21})}, \underline{x_{21} \doteq q}, \underline{x_3 \doteq x_{12}}, \underline{x_{13} \doteq x_{12}}, x_6 \doteq A(x_{14}), x_2 \doteq x_{15}, x_{14} \doteq x_{15}, x_2 \doteq e, \underline{x_3 \doteq q}.$
surf. fact., and compr.	:- $x_{19} \doteq 2, \underline{x_{19} \doteq x_8 + x_9}, TA(x_6), x_9 \doteq B(x_{10}), x_6 \doteq x_{11}, x_{10} \doteq x_{11}, x_8 \doteq x_{24}, x_{20} \doteq x_{24}, x_{20} \doteq 0, x_{20} \doteq Vt(x_3), x_6 \doteq A(x_{14}), x_2 \doteq x_{15}, x_{14} \doteq x_{15}, x_2 \doteq e, x_3 \doteq q.$
res. with T3′	:- $x_{19} \doteq 2, x_{19} \doteq x_{31}, x_{25} \doteq x_{31}, \underline{x_{32} \doteq x_8 + x_9}, \underline{x_{32} \doteq x_{26} + x_{27}}, x_{27} \doteq x_{28}, x_{25} \doteq x_{28}, \underline{x_{26} \doteq 0}, TA(x_6), x_9 \doteq B(x_{10}), x_6 \doteq x_{11}, x_{10} \doteq x_{11}, \underline{x_8 \doteq x_{24}}, \underline{x_{20} \doteq x_{24}}, \underline{x_{20} \doteq 0}, x_{20} \doteq Vt(x_3), x_6 \doteq A(x_{14}), x_2 \doteq x_{15}, x_{14} \doteq x_{15}, x_2 \doteq e, x_3 \doteq q.$
root fact., surf. fact. and compr.	:- $\underline{x_{19} \doteq 2}, \underline{x_{19} \doteq x_{31}}, \underline{x_{25} \doteq x_{31}}, \underline{x_9 \doteq x_{28}}, \underline{x_{25} \doteq x_{28}}, TA(x_6), x_9 \doteq B(x_{10}), x_6 \doteq x_{11}, x_{10} \doteq x_{11}, x_8 \doteq 0, x_8 \doteq Vt(x_3), x_6 \doteq A(x_{14}), x_2 \doteq x_{15}, x_{14} \doteq x_{15}, x_2 \doteq e, x_3 \doteq q.$
root fact., surf. fact., compr.	:- $x_9 \doteq 2, TA(x_6), x_9 \doteq B(x_{10}), x_6 \doteq x_{11}, x_{10} \doteq x_{11}, \underline{x_8 \doteq 0}, x_8 \doteq Vt(x_3), x_6 \doteq A(x_{14}), x_2 \doteq x_{15}, x_{14} \doteq x_{15}, x_2 \doteq e, x_3 \doteq q.$
res. with T1′	:- $x_9 \doteq 2, TA(x_6), x_9 \doteq B(x_{10}), x_6 \doteq x_{11}, x_{10} \doteq x_{11}, \underline{x_8 \doteq x_{21}}, \underline{x_{17} \doteq x_{21}}, \underline{x_{22} \doteq 0}, \underline{x_{17} \doteq Vt(x_{18})}, \underline{x_{18} \doteq q}, \underline{x_8 \doteq Vt(x_3)}, x_6 \doteq A(x_{14}), x_2 \doteq x_{15}, x_{14} \doteq x_{15}, x_2 \doteq e, x_3 \doteq q.$
surf. fact., root fact. and compr.	:- $x_9 \doteq 2, TA(x_6), x_9 \doteq B(x_{10}), x_6 \doteq x_{11}, x_{10} \doteq x_{11}, x_6 \doteq A(x_{14}), x_2 \doteq x_{15}, x_{14} \doteq x_{15}, x_2 \doteq e, x_3 \doteq q.$
reduction to the min. residuum	:- $x_6 \doteq A(x_2), x_2 \doteq e, TA(x_6), x_9 \doteq B(x_6), x_9 \doteq 2.$

The last clause is the negation of the desired explanation. Note how two resolutions with T1′ were used to simulate symmetry.

6 Conclusion

From a theoretical perspective, surface deduction is very appealing in its simplicity. We have seen how (at least in theory), surface deduction can be applied in situations such as abductive reasoning where deduction rather than refutation is the primary goal.

The preference criteria for explanations given in Section 4 are very weak. However, we believe that no matter what preference criteria are used, RES(C) is at least as good an explanation as C. One of the most important problems in abductive reasoning is to determine stronger preference criteria to avoid combinatorial explosion. These issues are discussed in [Poole and Provan 1990].

From a practical point of view, one of the frequently recognized problems with flattening the clauses of the input theory is that one loses most of the advantages of unification, particularly if the input theory contains few equalities. One can regain some of these advantages in practice by interpreting the set of equalities in the body of a clause as a directed graph or hypergraph (with arcs from the root variables to the surface terms) which defines the set of possible definitions of the main terms and variables of the clause. Such a directed graph generalizes the usual tree representation of terms. Unification and more generally term rewriting can then be replaced by (hyper)graph rewriting rules. To implement this idea, the deduction procedures must be substantially enhanced. The types of graph rewriting rules and graph representations needed require further research.

Many of the results used in this paper can be generalized to arbitrary clauses so that the restriction of abductive reasoning to Horn clause theories can be removed. These generalizations will be the topic of a forthcoming paper.

References

[Baxter 1976] L. D. Baxter. *The Complexity of Unification.* Ph.D. Thesis, University of Waterloo, 1976.

[Bosco *et al.* 1988] P. G. Bosco, E. Giovannetti, and C. Moiso. Narrowing vs. SLD-Resolution. *Theoretical Computer Science* Vol. 59 (1988), pp. 3–23.

[Brand 1975] D. Brand. Proving Theorems with the Modification Method. *SIAM J. Comput.* Vol. 4 (1975), pp. 412–430.

[Chan 1986] K. H. Chan. Equivalent Logic Programs and Symmetric Homogeneous Forms of Logic Programs with Equality. Technical Report 86, Dept. of Computer Science, Univ. of Western Ontario, London, Ont., Canada, 1986.

[Charniak 1988] E. Charniak. Motivation Analysis, Abductive Unification, and Nonmonotonic Equality. *Artificial Intelligence* Vol. 34 (1988), pp. 275–295.

[Colmerauer *et al.* 1982] A. Colmerauer *et al.*. *Prolog II: Reference Manual and Theoretical Model.* Groupe d'Intelligence Artificielle, Faculte des Sciences de Luminy, Marseilles, 1982.

[Cox and Pietrzykowski 1985] P. T. Cox and T. Pietrzykowski. Surface Deduction: a Uniform Mechanism for Logic Programming. In *Proc. Symp. On Logic Programming*, IEEE Press, Washington, 1985. pp. 220–227.

[Cox and Pietrzykowski 1986] P. T. Cox and T. Pietrzykowski. Incorporating Equality into Logic Programming via Surface Deduction. *Ann. Pure Appl. Logic*, Vol. 31 (1986), pp. 177–189.

[Cox and Pietrzykowski 1987] P. T. Cox and T. Pietrzykowski. General Diagnosis by Abductive Inference. In *Proceedings of the Symposium on Logic Programming*, IEEE Press, Washington, 1987. pp. 183–189.

[Cox *et al.* 1991] P.T. Cox, E. Knill, and T. Pietrzykowski. Equality and Abductive Residua for Horn Clauses. Technical Report TR-8-1991, School of Computer Science, Technical University of Nova Scotia, Halifax, NS, Canada, 1991.

[van Emden and Lloyd 1984] M.H. van Emden and J.W. Lloyd. A Logical Reconstruction of Prolog II. In *Proc. 2nd Intl. Conf. on Logic Prog.* Uppsala, 1984. pp. 35–40.

[Goebel 1990] R. Goebel. A Quick Review of Hypothetical Reasoning Based on Abduction. In *AAAI Spring Symposium on Automated Abduction*, Stanford University, 1990. pp. 145–149.

[Hoddinott and Elcock 1986] P. Hoddinott and E.W. Elcock. PROLOG: Subsumption of Equality Axioms by the Homogeneous Form. In *Proceedings of the Symposium on Logic Programming*, 1986. pp. 115–126.

[Hölldobler 1989] S. Hölldobler. *Foundations of Equational Logic Programming*. Lecture Notes in Computer Science 353, Springer Verlag, Berlin, 1989.

[Hölldobler 1990] S. Hölldobler. Conditional Equational Theories and Complete Sets of Transformations. *Theoretical Computer Science*, Vol. 75 (1990), pp. 85–110.

[de Kleer *et al.* 1988] J. de Kleer, A. K. Mackworth, and R. Reiter. Characterizing Diagnoses. In *Proceedings Eighth National Conference on Artificial Intelligence*, 1990. pp. 324–330.

[Kohavi 1978] Z. Kohavi. *Switching and Finite Automata Theory*. McGraw-Hill, 1978.

[Lloyd 1984] J. W. Lloyd. *Foundations of Logic Programming*. Springer Verlag, Berlin, 1984.

[Paterson and Wegman 1978] M. S. Paterson and M. N. Wegman. Linear Unification. *J. Comput. Syst. Sci.* Vol. 16 (1978), pp. 158–167.

[Poole 1988] D. Poole. A Logical Framework for Default Reasoning. *Artificial Intelligence* Vol. 36 (1988), pp. 27–47.

[Poole and Provan 1990] D. Poole and G. M. Provan. What is an Optimal Diagnosis? In *Conference on Uncertainty in AI*, Boston, 1990.

[Reiter and de Kleer 1987] R. Reiter and J. de Kleer. Foundations of Assumption-Based Truth Maintenance Systems: Preliminary Report. In *Proceedings of the National Conference on Artificial Intelligence*, 1987. pp. 183–188.