# Variables in Hypotheses

David Poole

Research Report CS-86-44
September 1986

# Variables in Hypotheses

David Poole
Logic Programming and Artificial Intelligence Group,
University of Waterloo,
Waterloo, Ontario, Canada, N2L3G1
dlpoole@waterloo.csnet

September 25, 1986

**Abstract**

In many applications we want to build systems which must test the consistency of some theory (or set of axioms). There is a problem which arises when we are generating theories that contain variables. This problem is general to many applications, for example abduction, learning, default reasoning, diagnosis, and is examined here in the context of theory formation from a fixed set of possible hypotheses. Two solutions are examined, the first where we are only allowed to have ground instances in theories formed, and the second where we may have universally quantified variables in the theory. It is shown that for the second case that the solution of reverse Skolemisation is not adequate to solve the problem, nor is any naive pattern matcher. A general solution for both cases is presented.

## 1 Introduction

There are many applications in which one wants to test consistency (or failure to prove the negation) of some formula which is generated by some program. For example:

- default reasoning, where we want to be able to use some instance of a formula if it is consistent

- learning, where we want to be able to hypothesis some causal rule if it is consistent

- negation as failure, where we want to conclude an atom if we can't prove its negation

We consider the problem in terms of a theory formation system which has a fixed set of possible hypotheses (i.e. where we assume that some other system is supplying the general forms we want to be able to assume).

The problem is how to test the consistency of a theory which is generated by some program. This is a problem because variables in a generated theory somehow need to have their quantification reversed when checking consistency. This paper shows that some proposed solutions do not work, and gives a solution to the problems where we don't allow variables in our hypotheses, and the general case where we have a single hypothesis.

## 2 Formal Framework

We use the standard syntax of the first order predicate calculus, with variables in upper case.

$F$ is a set of closed formulae (called *facts*), which we are giving as true

$\Delta$ is a set of formulae, each instance of which can be used as a possible hypothesis

We say formula $g$ is explainable if there is some $D$, a set of instances of elements of $\Delta$, such that

$$F \cup D \models g$$

$$F \cup D \text{ is consistent}$$

$D$ is said to be the theory that explains $g$.

Without loss of generality, we assume that $g$ is variable free, and is an atom. If $w$ is a wff which we want to explain, we can add $w \Rightarrow g$ to $F$, and try to explain $g$, where $g$ is a unique predicate not appearing elsewhere. $g$ is explainable if and only if $w$ is.

Without loss of generality we also assume that elements of $\Delta$ do not contain bound variables. If $w$ is an element of $\Delta$ with free variables $X_1, ..., X_n$, we can replace $w$ with $d(X_1, ..., X_n)$ in $\Delta$, where $d$ is a predicate symbol not appearing elsewhere, and add $\forall X_1 ... \forall X_n \; d(X_1, ..., X_n) \Rightarrow w$ to $F$. $d(X_1, ..., X_n)$ is assumable and consistent in the resulting system if and only if $w$ was assumable and consistent in the original system. This will simplify the analysis which follows.

N.B. $w \in \Delta$ is equivalent to [Reiter80]'s normal default : $Mw/w$ [Poole86]

## 3 Implementation

The obvious way to implement explainability [Reiter80,PGA86] is to note that both proving the observations, and testing consistency are both the role of a theorem prover. Intuitively the idea is to try to prove the goal from $F$ and $\Delta$, and make $D$ the set of instances of $\Delta$ used in the proof. Again a theorem prover is the appropriate tool to check whether $F \cup D$ is consistent.

In this paper I assume that we are using some sort of resolution theorem prover to generate the instances of hypotheses which imply the goal. The results, however, do not seem to be restricted to such systems.

There is a problem which arises when there are variables in the $D$ generated. Consider the following example:

**Example 1** Let $\Delta = \{p(X)\}$. That is, any instance of $p$ can be used if it is consistent. Let $F = \{\forall Y(p(Y) \Rightarrow g), \neg p(a)\}$.

$g$ is explainable with the theory $\{p(b)\}$, which is consistent with $F$ (consider the interpretation $I = \{\neg p(a), p(b)\}$ on the domain $\{a, b\}$), and implies $g$. So according to our semantics above, $g$ is explainable.

However, if we try to prove $g$, we generate $D = \{p(Y)\}$ where $Y$ is free (implicitly a universally quantified variable). The existence of the fact $\neg p(a)$ should not make it inconsistent, as we want $g$ to be explainable.

**Theorem 1** *It is not adequate to only consider interpretations in the Herbrand universe of some set of formulae.*

**Proof** consider the example above; the Herbrand universe is just the set $\{a\}$. Within this domain there is no consistent theory to explain $g$. □

This shows that Herbrand's theorem is not applicable to the whole system. It is, however, applicable to each of the deduction steps [Chang73].

We now proceed to show how explainability can be computed. This is done in two stages. First we examine the case where only ground instances of defaults are allowed. This is then expanded to allowing general instances of hypotheses in a theory.

## 4 Ground Instances of Defaults

Consider the case where we are only allowing ground instances of possible hypotheses in a theory. A ground instance is defined to be one without variables or Skolem constants.

The ground procedure[1] to compute explainability becomes

1. Skolemise $F$, and treat free variables as universally quantified (as in Resolution theorem proving);
2. try to prove $g$ using elements of $F$ and $\Delta$ as axioms. Make $D$ the set of instances of $\Delta$ used in the proof;
3. remove any $D$ which contains a Skolem function;
4. replace free variables in $D$ with unique constants;
5. add the $D$ to $F$ and try to prove an inconsistency. If complete search for a proof fails, $g$ is explainable.

**Example 2** consider $F$ and $\Delta$ as in example 1 above. If we try to prove $g$, we use the hypothesis instance $p(Y)$. This means that $g$ is provable from any instance of $p(Y)$. To show $g$ cannot be explained, we must show that all of the instances are inconsistent. The above algorithm says we replace $Y$ with a constant $\beta$. $p(\beta)$ is consistent with the facts. So that we can show $g$ is explainable.

Let us first try to justify this procedure.

**Lemma 2** *If $D$ is some consistent theory which predicts $g$, then some more general set $B$ of instances of defaults can be generated in the manner described above such that $D = B\theta$ for some $\theta$.*

**Proof** If there is a theory $D$, then there is a ground instance which is also consistent and proves $g$ (as any proof can be converted into a ground proof, and if a theory is inconsistent then so is any ground instance). The fact that some more general instance will be found is a direct corollary of the lifting lemma [Chang73, page 84]. □

**Theorem 3** *The above procedure is correct.*

**Proof** The third step of the procedure just enforces the groundness of defaults found. The fourth and fifth steps follow from checking if $\exists\overline{X}D$ is consistent by Skolemising the $\overline{X}$ (the free variables in $D$). □.

[1] This problem is, in general, undecidable; if it halts, it has computed a correct answer, and if a provable answer exists this (nondeterministic) algorithm can compute it.

## 5 Arbitrary Instances of Defaults

Sometimes we don't want to be restricted to just ground instances of defaults. Consider the following examples:

**Example 3** Consider the blocks world, where we only want positive knowledge about which blocks are on each other, and we want the closed world assumption for "on". This is done by having the defaults: $\Delta = \{\neg on(X,Y)\}$.

If we have

$$\begin{aligned} F = \{ & \forall X((\neg \exists Y \ \mathrm{on}(Y,X)) \Rightarrow \mathrm{cleartop}(X)), \\ & \mathrm{on}(a,b)\} \end{aligned}$$

This says that a block has a clear top if there is nothing on it, and that block $a$ is on block $b$. Intuitively, we want to conclude that $b$ does not have a clear top, and all other blocks have a clear top. The theory used to explain block $c$ having a clear top is $\{\neg on(f(c),c)\}$ where $f(c)$ is the individual said to exist in the first fact.

**Example 4** Let $\Delta = \{ontable(X)\}$. That is we may assume that any block is on the table. Let

$$\begin{aligned} F = \{ & (\exists Y \ \mathrm{red}(Y) \wedge ontable(Y)) \Rightarrow g \\ & \exists X \ \mathrm{red}(X)\} \end{aligned}$$

Intuitively we want to say that $g$ is explainable as there is a red thing on the table, namely the object that we know is red (but do not know its name). The ground procedure would reject such an answer, as it must know the name of the individual said to exist.

If we want to expand the procedure given in the previous section, we have to consider how to handle Skolem functions in the theories generated. One way to try to do this is pattern match the instance of the hypotheses generated with the instances that lead to inconsistencies.

It has been suggested that we "reverse Skolemise" [Bledsoe78,Cox80] the the generated hypotheses and try to prove their negation. If we can prove their negation, we have shown the theory inconsistent; if a complete theorem prover fails to prove their negation the theory is consistent. This is equivalent to unifying the reverse Skolemised form with the inconsistent instances of possible hypotheses.

**Theorem 4** *No such pattern matching program will work in general. That is there can be no algorithm which does pattern matching on the instances which lead to the goal to be explained, and the instances which are inconsistent such that the goal is explained if and only if the pattern matcher fails.*

**Proof:** To prove this it is adequate to show two examples which have identical inconsistent hypotheses and syntactically identical instances which can prove the goal, but have opposite answers.

The examples we use are based on the cleartop example. Consider $\Delta = \{\neg on(X,Y)\}$ and

$$\begin{aligned} F = \{\ & \forall X((\neg\exists Y \ \mathrm{on}(Y,X)) \Rightarrow \mathrm{cleartop}(X)),\\ & \mathrm{on}(a,b)\\ & \mathrm{red}(b)\\ & (\forall X \ \mathrm{cleartop}(X)) \Rightarrow g_1,\\ & (\forall X \ \neg\mathrm{red}(X) \Rightarrow \mathrm{cleartop}(X)) \Rightarrow g_2\} \end{aligned}$$

That is, there is one block ($a$) on a red block ($b$). $g_1$ is explainable if all blocks have a clear top. $g_2$ is explainable if all non-red blocks have a clear top. According to our semantics $g_1$ should not be explainable, but $g_2$ should be explainable.

When attempting to compute their explanations, we note that exactly the same instances of hypotheses lead to each goal, and exactly the same instances are inconsistent. Put into Skolem normal form this becomes:

$$\begin{aligned} F = \{\ & ((\neg\mathrm{on}(f(X),X)) \Rightarrow \mathrm{cleartop}(X)),\\ & \mathrm{on}(a,b)\\ & \mathrm{red}(b)\\ & \mathrm{cleartop}(c_1) \Rightarrow g_1,\\ & \mathrm{cleartop}(c_2) \Rightarrow g_2,\\ & \mathrm{red}(c_2) \Rightarrow g_2\} \end{aligned}$$

To prove each $g_i$ we generate the theory $\{\neg on(f(c_i),c_i)\}$, and the only inconsistent instance of hypotheses is $\neg on(a,b)$. Note that the last clause is not used in either the proof of $g_2$ nor in proof of inconsistency. □

The problem we have is that we have lost the context of what the Skolem constants represent.

# 6 Computing Inconsistencies

## 6.1 Skolemisation

Skolemisation is the replacement of

$$\forall X_1 ... \forall X_n \exists y \; w[X_1, ..., X_n, Y]$$

(where $w[X_1, ..., X_n, Y]$ is a well formed formula with free variables $X_1, ..., X_n, Y$) with

$$\forall X_1 ... \forall X_n \; w[X_1, ..., X_n, f(X_1, ..., X_n)]$$

If we replace all existential variables by their corresponding Skolem function, the system is in Skolem normal form. All variables remaining are universally quantified, and so explicit scoping is redundant and can be removed.

**Theorem 5 (Skolem)** *A set of formulae is unsatisfiable iff the Skolemised form is unsatisfiable.*

This is proven in [Chang73, theorem 4.1]. For more details see [Chang73].

## 6.2 Hilbert's $\varepsilon$-symbol

Hilbert's $\varepsilon$-symbol is a notational device to implicitly describe an individual said to exist. $\varepsilon x.P(x)$ is, intuitively "an $x$ such that $P(x)$ is true". This was designed to eliminate existential variables through the equivalence:

$$\exists X \; w[X] \equiv w[\varepsilon X.w[X]]$$

where $w[X]$ is any well formed formula parameterised by $X$. See [Leisenring69] for a detailed description of Hilbert's $\varepsilon$-symbol.

## 6.3 Building the Knowledge Base

The problem that arose before is that we did not know the context of the Skolem functions used. In this section we show how Hilbert's $\varepsilon$-symbol can be used to keep track of which functions the Skolem functions denote.

When Skolemising, we replace

$$\forall X_1 ... \forall X_n \exists y \; w[X_1, ..., X_n, Y]$$

with

$$\forall X_1 ... \forall X_n \; w[X_1, ..., X_n, f(X_1, ..., X_n)]$$

We should also define what $f$ is. We can use Hilbert's $\varepsilon$-symbol to define $f$:

$$f = \lambda X_1, ..., \lambda X_n.\varepsilon y.w[X_1, ..., X_n, y]$$

that is

$$f(X_1, ..., X_n) = \varepsilon y.w[X_1, ..., X_n, y]$$

**Example 5** The fact $\exists X\ q(X)$, when Skolemised, becomes $q(c)$ where $c$ is some unique constant symbol. $c$ is defined as $\varepsilon X.q(X)$.

To build the knowledge base, Skolemise all variables, and record the definitions of all Skolem functions and constants.

## 6.4 The Prover

The prover presented here is an extension of the one presented in section 4. The proof of correctness follows directly from the proof in that section. Once the knowledge base has been built, we try to prove the observation to be explained, using the facts in $F$ and the hypotheses in $\Delta$. Let $D$ be the set of instances of elements of $\Delta$ used in the proof. $D$ may contain free variables and Skolem functions. We then replace all Skolem functions in $D$ with their definition. If $X_1, ..., X_n$ are the free variables in $D$, we have proven

$$\forall X_1, ..., \forall X_n(D \Rightarrow g)$$

that is

$$(\exists X_1, ..., \exists X_n D) \Rightarrow g$$

The aim is to prove that $\exists X_1, ..., \exists X_n D$ is consistent with the facts. Notice that we do not have to worry about Skolem functions in $D$, as we removed all of these. We can use the procedure presented in section 4 as it is now applicable. However, we must be concerned about Hilbert's $\varepsilon$-symbol appearing in the generated theory to be proven inconsistent.

If we have the generated theory $p(\varepsilon x.q(x))$ we are assuming $p(Y)$ for all instances that we need to. We have to assume $p(Y)$ for all $Y$ such that

$$F \wedge q(Y) \not\models g$$

to prove inconsistent with $F$, we try to prove for some $Y$

$$F \models \neg p(Y)$$

$$F \wedge q(Y) \not\models g$$

If we cannot prove this, we assume all of the instances of $p(X)$ which are needed to prove $g$. That is, we are assuming $p(X)$ for all the individuals which do not otherwise lead to $g$.

If we can find such an individual, there is no instance of $p(X)$ which can be consistently assumed (for this proof), and can be used to prove $g$ (as all we know is that one $Y$ such that $q(Y)$ is true).

**Example 6** Consider the blocks world of example 3. Let $\Delta = \{\neg on(X,Y)\}$.

$F = \{\ \forall X(\neg\exists Y \text{ on}(Y,X)) \Rightarrow \text{cleartop}(X),$
$\text{on}(a,b),$
$\text{cleartop}(b) \Rightarrow g_b,$
$\text{cleartop}(c) \Rightarrow g_c\}$

That is, we can explain $g_b$ if $b$ has a clear top, and explain $g_c$ if $c$ has a clear top.

Skolemising the facts gives,

$F_s = \{\ \neg\text{on}(f(X),X) \Rightarrow \text{cleartop}(X),$
$\text{on}(a,b),$
$\text{cleartop}(b) \Rightarrow g_b,$
$\text{cleartop}(c) \Rightarrow g_c\}$

where $f(X) = \varepsilon Y.\neg\text{on}(Y,X) \Rightarrow \text{cleartop}(X)$.

We can prove $g_b$ by generating the theory $\{\neg on(f(b),b)\}$. This can be proven inconsistent if we can prove $on(Y,b)$ for some $Y$ such that

$$F \wedge (\neg\text{on}(Y,b) \Rightarrow \text{cleartop}(b)) \not\models g_b$$

which is provable ($Y = a$).

The corresponding theory generated to explain $g_c$ cannot be proven inconsistent as we cannot prove $\exists Y\ on(Y,c)$.

**Example 7** Consider example 4. $\Delta = \{ontable(X)\}$.

$$F = \{\ (\exists Y \text{ red}(Y) \wedge ontable(Y)) \Rightarrow g \\ \exists X \text{ red}(X)\}$$

Skolemised, the facts become,

$$F = \{\ (\text{red}(Y) \wedge ontable(Y)) \Rightarrow g \\ \text{red}(c)\}$$

where $c = \varepsilon X.\text{red}(X)$. We can explain $g$, generating the potential theory $\{ontable(c)\}$ which is, when $c$ is replaced by its definition, $\{ontable(\varepsilon X.\text{red}(X))\}$. We cannot prove $\neg ontable(X)$ for any $X$, so that $g$ is explained.

**Example 7A** Let $F_1 = F \cup \{red(a), \neg ontable(a)\}$. $g$ should not be explainable from $F_1$, as there is no reason to assume that there is another individual which is also red. We can prove, $\neg ontable(X)$ for $X = a$ and can prove $F \wedge \mathrm{red}(a) \not\models g$.

**Example 7B** Let $F_2 = F \cup \{\neg red(a), \neg ontable(a)\}$. $g$ should be explainable from $F_2$, as we know there is another individual which is red which we can assume is on the table. We can explain $g$ with the same theory, we can prove $\neg ontable(X)$ for $X = a$, but can prove $F \wedge \mathrm{red}(a) \models g$.

**Example 8** Consider the example in theorem 4 above.

$$f = \lambda X.\varepsilon Y.\neg \mathrm{on}(Y, X) \Rightarrow \mathrm{cleartop}(X)$$

The Skolem constants, $c_i$ have different definitions,

$$c_1 = \varepsilon X.\mathrm{cleartop}(X) \Rightarrow g_1$$

$$c_2 = \varepsilon X.(\neg \mathrm{red}(X) \Rightarrow \mathrm{cleartop}(X)) \Rightarrow g_1$$

In each case the generated theory is $\{\neg on(f(c_i), c_i)\}$. So for each of these we have to prove

$$on(Y, X)$$

for some $Y$ and $X$. This can be proven for $Y = a$ and $X = b$. In the first case, we also have to prove

$$F \wedge (\mathrm{cleartop}(b) \Rightarrow g_1) \wedge (\neg \mathrm{on}(a, b) \Rightarrow \mathrm{cleartop}(b)) \not\models g_1$$

which can be proven. Hence $g_1$ cannot be explained.

To prove the second case inconsistent, we have to prove

$$F \wedge ((\neg \mathrm{red}(b) \Rightarrow \mathrm{cleartop}(b)) \Rightarrow g_2) \wedge (\neg \mathrm{on}(a, b) \Rightarrow \mathrm{cleartop}(b)) \not\models g_2$$

so we have to check

$$F \wedge (\mathrm{red}(b) \Rightarrow g_2) \wedge (\mathrm{cleartop}(b)) \Rightarrow g_2) \wedge (\neg \mathrm{on}(a, b) \Rightarrow \mathrm{cleartop}(b)) \not\models g_2$$

Which is not true (as red$(b)$ is in $F$). Thus $g_2$ can be explained.

## 7 Conclusion

There are many areas in which this problem arises. Some people have assumed that it is sufficient to consider the Herbrand Universe [Reiter80]. Others have tried to define a "reverse Skolemisation" algorithm which can be applied to the hypotheses generated, and unified with the instances leading to inconsistencies [Cox80,Bledsoe78]. We have shown that both of these ideas cannot work.

We have shown that we need to keep track of the context in which Skolem functions are defined, and have shown how this can be done by using Hilbert's $\varepsilon$-symbol. An procedure is given which solves this problem for the case of a single hypothesis and a single theory.

## Acknowledgements

This research was supported under NSERC grant A6260. Thanks to Eric Neufeld, Paul Van Arragon and Randy Goebel for valuable comments on an earlier draft of this paper.

## References

[Bledsoe78] Bledsoe,W.W. and Ballantyne,A.M., *Unskolemizing,* University of Texas at Austin, Math Dept Memo ATP-41A, July 1978.

[Cox80] Cox,P.T. and Pietrzykowski,T., "A Complete, Nonredundant Algorithm for Reverse Skolemisation", in Bibel,W. and Kowalski,R. (Eds) *Fifth Conference on Automated Deduction*, Springer-Verlag, Lecture Notes in Computer Science 87, Heidelburg, Germany, pp 374-385.

[Chang73] C. Chang and R. Lee, *Symbolic Logic and Mechanical Theorem Proving*, Academic Press, 1973.

[Leisenring69] A. C. Leisenring, *Mathematical Logic and Hilbert's $\varepsilon$-symbol,* MacDonald Technical and Scientific, London, 1969.

[Poole85] D. Poole, "On the Comparison of Theories: Preferring the Most Specific Explanation", *Proceedings Ninth International Joint Conference on Artificial Intelligence*, Los Angeles, August 1985, pp. 144-147.

[Poole86] D. Poole, ***Default Reasoning and Diagnosis as Theory Formation***, Technical Report CS-86-08, Department of Computer Science, University of Waterloo, March 1986, 19 pages.

[PGA86] D. Poole, R. Goebel and R. Aleluinas, "Theorist: a logical reasoning system for defaults and diagnosis", to appear in N.Cercone and G.McCalla (Eds.) *Knowledge Representation.*

[Reiter80] R. Reiter, "A Logic for Default Reasoning", *Artificial Intelligence*, Vol 13, pp. 81–132.