On the Comparison of Theories: Preferring the Most Specific Explanation

David L Poole

Logic Programming and Artificial Intelligence Group
Department of Computer Science
University of Waterloo
Waterloo, Ontario, Canada N2L 3G1

Abstract

Default reasoning is reasoning with generalised knowledge which we want to use if there is no more specific knowledge also applicable. This paper presents a formal, model-theoretic characterisation of default reasoning. Defaults are treated as possible hypotheses in a "scientific" theory to explain the results. One of the problems with systems that reason with defaults occurs when two answers can be produced, and one is preferred. In terms of our default logic, we define a semantic characterisation of the notion of the more specific theory. This overcomes many of the problems which motivated non-normal defaults, and provides a semantics for correct inheritance in inheritance systems, where we want choose the result supported by the most specific knowledge. We also show how to produce a general computational mechanism in terms of normal first order predicate calculus deduction systems.

1. Introduction

Various formalisms have been proposed to give a semantics to systems with knowledge with exceptions (Reiter[80], McDermott and Doyle[80], McCarthy[80,84], Moore(83], Poole[84b,c]). One problem which arises, particularly with respect to inheritance systems (Etherington and Reiter[83], Touretzky[84]), is how to resolve ambiguity when different defaults can be used to produce different answers.

Poole [84b] gives a model theoretic definition of default reasoning with defaults as possible hypotheses in a scientific theory. An answer is explainable if it logically follows from some consistent set of default instances (a theory) together with the facts. In this paper we propose, in the case of ambiguous answers, a theory comparator to find the answer supported by the best explanation. This is applied to the problem of default inheritances in semantic networks, where the most specific theory is preferred. If there is general knowledge applicable to an answer, and more specific knowledge also applicable, then the most specific knowledge is preferred.

2. Default Logic

2.1. Syntax

The syntax is an extension of the first order predicate calculus. A *wff* is a well formed formula of the first order predicate calculus. The input of the system is defined as follows:

<input>::= <wff> |
<list-of-variables> ASSUME <wff>

Inputs of the first type are called facts, and inputs of the second type are called defaults. An instance of a default is the wff with values substituted for the variables appearing before the ASSUME.

The facts correspond to the axioms in a normal logic, and the defaults are knowledge which can be used as long as they are consistent.

2.2. Formal Semantics

If F is the set of facts, and Δ the set of defaults, we say that g is explainable if there is some D, a set of instances of elements of A such that

$$F \cup D \vdash g$$

and $F \cup D$ is consistent.

D is said to be the theory that explains g. D *is* like a "scientific" theory. *Theorems*$(F \cup D)$ is the corresponding "logical" theory.

That is, g is explainable if there is a theory which explains it. The instances of defaults are the possible hypotheses in this theory. This theory must be consistent with all of the facts.

2.3. Example 1.

Consider the knowledge that "birds fly" This is knowledge we may want to use although it has exceptions (eg. emus, penguins, roast ducks, little chicks etc.). The default "a bird can fly unless it can be shown not to" is given by:

$$(x)\ ASSUME\ bird(x) \supset flies(x)$$

If we are also given the knowledge that emus are birds that don't fly, and that Tweety is a bird, and Edna is an emu, we input

$$\forall x\ emu(x) \supset \neg flies(x)$$
$$\forall x\ emu(x) \supset bird(x)$$
$$bird(tweety)$$
$$emu(edna)$$

From this knowledge we can explain *flien (tweety)* with the theory

$$\{bird\ (tweety) \supset flies\ (tweety)\}$$

which is consistent (its negation cannot be proven), and together with the facts can be used to prove *flies (tweety).*

We cannot explain *flies(edna)* as the corresponding theory for edna is not consistent, as its negation can be proven from the facts.

2.4. Example 2

Suppose that we modify the first example to have "emus don't fly" as a default rather than a fact. We get the following system:

$$(x)\ ASSUME\ bird(x) \supset flies(x)$$
$$(x)\ ASSUME\ emu(x) \supset \neg flies(x)$$
$$\forall x\ emu(x) \supset bird(x)$$
$$bird(tweety)$$
$$emu(edna)$$

We can now explain everything we could in example 1, and can also explain *flies (edna)* with the theory $D_1 = \{bird(edna) \supset flies(edna)\}$, as well as $\neg flies(edna)$ with the theory $D_2 = \{emu(edna) \supset \neg flies(edna)\}$.

This is an example where there are two different things which can be explained, and where one is preferred. We prefer to say that edna does not fly, as our knowledge that emus don't fly is more specific than our knowledge about birds flying.

8. A Theory Comparator

So far we have given the semantics of a default logic, closely related to the normal defaults of Reiter [80] (see section 5.2). Our theories differ from his extensions, as our theories are intended to be minimal (the simplest theory to explain the results), whereas his extensions are the theorems following from maximal sets of instances of defaults which are consistent.

As in example 2 above, a problem arises with normal defaults when two different answers can be produced in two extensions, but where one is preferable over the other (Reiter and Criscuolo[8l]).

Within inheritance hierarchies we want default links, but want to use the most specific knowledge to deduce an answer (Etherington and Reiter[83], Touretzky[84]). Within our system this is done by preferring the most specific theory to explain the results. We want our definition of more specific to be a semantic rather than a syntactic definition so that it can be understood and justified independently of any implementation, and so that it does not fall into the problems of shortest paths and redundant knowledge which arise with particular syntactic definitions of best inheritance (Touretzky[84]).

To formalise the notion of one theory being applicable whenever another one is, we need to be able to talk about, "what if something else were the case" To do this we divide the facts into two classes:

F_n those facts necessarily true in any world in our domain.

F_e those facts which happen to be true in the case we are considering.

In the *birds fly* example above, "$emu(x) \supset bird(x)$" is a necessary part of our domain. We do not want to consider the case "what if emus were not birds?". This should be contrasted with the fact *emu(edna).* In this we want to say the theory which explains *flies(edna)* is more general as it is applicable even if we only knew Edna was a bird. Thus we do want to consider the case where Edna is not an emu.

F_n and F_c can be seen to correspond to the network links and the starting nodes in NETL (Fahlman[79]), or to the necessary and contingent facts in modal logic (Hughes and Cresswell[68]).

Suppose we have a system consisting of necessary facts F_n, contingent facts F_c and defaults Δ. A **solution** S is a pair $<D,g>$, where D is a theory to explain g. That is, $F_n \cup F_c \cup D \vdash g$.

For some possible fact F_p, we say solution $S=<D,g>$ is **applicable** if $F_p \cup D \cup F_n \vdash g$. That is, if D can be used to prove g given F_p and F_n.

We say F_p is **adequate** to make D explain g if $F_n \cup D \cup F_p \vdash g$.

Given two solutions $S_1=<D_1,g_1>$ and $S_2=<D_2,g_2>$, we say S_1 is **more general** than S_2 (written $S_1 \leq S_2$), if there is possible fact F_p for which S_1 is applicable, S_2 is not applicable and F_p is not adequate to make D_2 explain g_1.

That is $<D_1,g_1> \leq <D_2,g_2>$ if there is an F_p, such that

$F_p \cup D_1 \cup F_n \vdash g_1$ and
$F_p \cup D_2 \cup F_n \nvdash g_2$ and
$F_p \cup D_2 \cup F_n \nvdash g_1$

The first two conditions say there is a fact that makes D_1 applicable and D_2 not applicable. The third condition puts a restriction on the F_p to make sure F_p does not make g_1 follow independently of D_1. We want F_p to actually need D_1 to prove g_1, rather than, for example, being g_1 itself.

Another way of looking at it is to define more specific, which is to mean that one is applicable whenever another is. Define S_1 is **more specific** than S_2 (written $S_1 \geq S_2$) if S_1 is not more general than S_2. That is if, for every F_p which is not adequate to make D_2 explain g_1, if S_1 is applicable then S_2 is applicable. More formally, if for every F_p

if $F_p \cup D_1 \cup F_n \vdash g_1$ and
$F_p \cup D_2 \cup F_n \nvdash g_1$ then
$F_p \cup D_2 \cup F_n \vdash g_2$

We say S_1 is strictly more specific than S_2, written $S_1 > S_2$, if S_1 is more specific than S_2 and S_2 is not more specific that S_1. Thus strictly more specific is the irreflexive version of more specific.

The preferred solution is the most specific solution. We prefer the most specific knowledge to be used in an explanation. If there is no most specific solution, because two solutions are not comparable, or because they are both more specific than each other (i.e. there is no strictly more specific solution) then an ambiguous answer is produced.

3.1. Example 2 Revisited

Within example 2, divide the facts into the necessary facts $F_n = \{emu(x) \supset bird(x)\}$ and the contingent facts $F_c = \{bird(tweety), emu(edna)\}$

To show D_1 is more general than D_2, choose F_p to be $bird(edna)$. In this case, $<D_1, flies(edna)>$ is applicable, but $<D_2, \neg flies(edna)>$ is not applicable. F_p is not adequate to make D_2 explain $flies(edna)$.

We cannot show D_2 is more general than D_1, as the only facts which are adequate to make D_2 explain $\neg flies(edna)$, are $emu(edna)$ and $\neg flies(edna)$ (or anything which logically implies either). For the first, $<D_2, flies(edna)>$ is also applicable. For the second, $\neg flies(edna)$ is adequate to make D_1 explain $\neg flies(edna)$. So the third condition in the definition of more general is used to reject this.

Thus we have shown that the solution $<\neg flies(edna), D_2>$ is strictly more specific than $<flies(edna), D_2>$. Our definition produces the intuitive answer for this case

3.2. Example 3

Consider the example of university students being typically adults, adults being typically employed, and university students being typically unemployed This is expressed with

(x) *ASSUME* $university\_student(x) \supset adult(x)$
(x) *ASSUME* $adult(x) \supset employed(x)$
(x) *ASSUME* $university\_student(x) \supset \neg employed(x)$

From this knowledge, given the contingent fact that Fred is a university student, there is a theory to explain her being unemployed, namely the theory $D_1 = \{university\_student(Fred) \supset \neg employed(Fred)\}$. There is a theory to explain her being employed, namely the theory $D_2 = \{university\text{-}student(Fred) \supset adult(Fred), adult(Fred) \supset employed(Fred)\}$. The reason that we prefer the first theory is that we have more explicit knowledge about university students than about adults in general. Using our theory comparator defined above we have $D_1 \geq D_2$, as the only fact that makes D_1 applicable to $\neg employed(Fred)$, that is not the goal itself is $university\_student(Fred)$ If this is the case then theory D_2 is also applicable. However we can not show $D_2 \geq D_1$ as $F_p = adult(Fred)$ together with D_2 produces the goal $employed(Fred)$. With $adult(Fred)$, D_1 does not give the goal $\neg employed(Fred)$. The theory comparator produces the desired result, namely $D_1 > D_2$.

Note that the same result is produced if university students being adults is given as a fact. The same contingent fact will produce the same comparisons. In this case, the above description holds. In fact any set of defaults and facts that makes $university\_student(Fred)$ imply $adult(Fred)$ will produce the same results. However, the same results are not produced if adults being employed is given as a fact instead. This is because the corresponding D_1 is adequate to prove $employed(Fred)$ given $adult(Fred)$. There is then a symmetry between adulthood and unemployment. Both theories are more specific than each other (neither is more general).

3.3. Example 4.

For examples where there is no preferable theory an ambiguous result is still obtained. Consider the republican-quaker-pacifist example. In this example the defaults are

(x) *ASSUME* $republican(x) \supset pro\text{-}defence(x)$
(x) *ASSUME* $pro\text{-}defence(x) \supset \neg pacifist(x)$
(x) *ASSUME* $quaker(x) \supset pacifist(x)$

If we are given the world fact that *Dick* is a republican and a quaker then we can explain both his pacifism and his non-pacifism.

The two corresponding theories are both more general than each other. This can be shown by considering the possible facts *republican (dick),* which makes the theory explaining his non-pacifism more general, and the fact *quaker(dick)* making the alternate theory more general. Thus we prefer neither explanation.

3.4. Example 5

To show that we can handle more complex cases, even where no contradictory results are obtained, consider the following example:

$$(x)\ ASSUME\ bird(x) \supset ans(x, flies)$$
$$(x)\ ASSUME\ emu(x) \supset ans(x, runs)$$
$$(x)\ ASSUME\ at\_emu\_farm(x) \supset emu(x)$$
$$ASSUME\ soothsaying$$
$$\forall x\ emu(x) \supset bird(x)$$
$$\forall x\ \forall y\ ans(x,y) \wedge soothsaying \supset say(x,y)$$

From the contingent fact, *at_emu_farm(randy),* we can explain *say(randy , runs)* and *say (randy , flies).* The solution of *say (randy, flies)* is more general, as it is also applicable for the fact *bird(randy).* However, we cannot show that the solution explaining *say (randy ,runs)* is more general. Thus we prefer the solution of *say(randy ,runs).*

3.5. Example 6.

The following example shows how the comparitor may produce a seemingly unintuitive answer. Consider the defaults $\Delta = \{a \supset b, b \supset c, a \supset \neg c, d \supset e, e \supset f\ d \supset \neg f\}$, $F_n = \{\neg c \wedge \neg f \supset g_1, c \wedge f \supset g_2\}$, $F_c = \{a \wedge d\}$. Although the theories to explain -c and -d are strictly more specific than the corresponding theories to explain c and *d,* and the theory to explain g_2 is more general than the theory to explain g_1 the theory to explain g_1 is more general than the theory to explain g_2. This can be shown by considering $F_p = a \wedge \neg f$. This problem cannot be fixed up by simply making F_p be compatible with D_1

4. Implementation

For the general case of finding the best explanation a standard first order predicate calculus deduction system can be used (eg. Poole[82,84a)). To prove that a goal is explainable, we use the deduction system in two phases. In the first we try to prove the goal using the facts and the defaults as the axioms in the deduction system. The defaults used in the proof of the goal can be collected into a potentially explaining theory. This theory can be checked for consistency by using the same deduction system to try to prove the theory and the facts are inconsistent. This is, of course, undecidable, in general. See Poole[84c] for a provably correct procedure for such a system, together with a description of the reverse Skolemisation needed to match the instances giving an inconsistency with those explaining the goal.

The general proof for more specific is also a set of standard first order proofs. In general, standard first order proof techniques are adequate for handling the default system. There will, of course, need to be ways to improve the efficiency of such a system by allowing the proofs to interact to prune each other's search spaces, and by exploiting the structure of specific domains (see for example Poole[84c]).

If we have a restricted language, for example in inheritance systems where logical entailment can be implemented by looking up the lattice, the corresponding efficiency can be transferred to handling the problem of finding the most specific theory. For the case of inheritance in semantic networks, Touretzky[84) has provided an efficient way to implement our preference of the most specific theory by conditioning an inheritance system so that a shortest path reasoner will work.

6. A Comparison With Other Systems

5.1. Semantic Networks

An attempt to characterise the notion of using the most specific knowledge is incorporated in preferring the shortest inference path in NETL (Fahlman[79j). As pointed out in Touretiky[84], this causes problems when there are redundant links and ambiguous extensions. As our system uses a semantic notion of more specific rather than some syntactic notion, it does not fall into these problems of shortest paths.

When everything is in the form of a semantic network, our logic provides a semantics for the inferential distance of Touretzky[84]. The inferential distance is whether there is an inheritance path from the subclass (the more specific) to the superclass. The superclass then provides the fact which makes it more general. Our logic is not, however, restricted to the simple implications found in semantic networks, but can use arbitrary wffs as both defaults and facts. It also handles a mixture of defaults and facts rather than treating all arcs as defaults. As shown in example 5, our system can find more specific theories when there are different, but not necessarily contradictory answers.

5.2. Reiter's Default Logic

This work can be most directly compared with the defaults of Reiter[80]. The definition of default above corresponds to his definition of a normal default. His normal defaults of the form $\alpha(\bar{x}):Mw(\bar{x})/w(\bar{x})$ are expressed as $(\bar{x})\ ASSUME\ \alpha(\bar{x}) \supset w(\bar{x})$, the difference being that we allow the contrapositive of our defaults. We have a simple model-theoretic semantics as well as a proof theory (see Poole[84c] for a detailed comparison).

The problems which gave rise to non-normal defaults, have however been solved in a very different manner. We have more modular statements than the corresponding semi-normal defaults (Reiter and Criscuolo[81], Etherington and Reiter[83]), as we do not. need to change the more general rules to add in exceptions to them, or know all of the more specific rules that may override a general rule we may have. We can add both more specific and less specific knowledge in a modular way without needing to change the other knowledge, or even needing to know that there are exceptions. The use of abnormality conditions (McCarthy[84]) still does not give the modularity of our system, but rather allows the exceptions to a general rule to be stated in a more modular way, but it must still be changed. In summary we have the notion that more specific knowledge is preferred as an integral part of the logic, rather than needing to be explicitly given.

6. Conclusion

We have outlined how defaults can be treated as possible hypotheses in a scientific theory used to explain the results. The problem of multiple answers can be overcome by allowing a comparison of theories to choose the answer which is supported by the "best" theory. When we have a set of knowledge which consists of general knowledge and more specific knowledge we want to choose the more specific knowledge in preference to the more general knowledge when there is a conflict.

This paper gives a formal account of a system for such reasoning. This is defined in terms of normal first-order model-theoretic semantics, and is defined so a normal first-order predicate calculus deduction system can be used to derive results.

We also provide an alternate semantics, and motivation for the "correct" inheritance in inheritance systems (Touretzky[84)). There are considerable advantages over non-normal defaults in terms of modularity for solving the same problem.

This is not an attempt to solve all of the problems of when one theory is better than another. This may change from domain to domain, for example in the diagnosis domain (Jones and Poole[85]) where it is the theory, which is the diagnosis, that is important, and for example in learning systems where we want the most general theory (within constraints) which explains the observations. There are examples one can imagine where there is more domain-specific knowledge about which explanation is better.

Acknowledgements

Thank to Eric Neufeld, Paul Van Arragon, Randy Goebel and Sumo Kindersley for valuable comments on an earlier draft of this paper. This research was sponsored by no defence department.

References

Etherington,D.W. and Reiter,R [83], "On Inheritance Hierarchies With Exceptions", Proc AAAI-88, August 1983, pp. 104-108.

Fahlman,S.E. [79], NETL: A System for Representing and Using Real-World Knowledge, MIT Press, 1979.

Hughes,G.E. and Cresswell,M.J.[68], An Introduction to Modal Logic, Methuen, London.

Jones,M. and Poole,D. |85], "An Expert System for Educational Diagnosis Based on Default Logic", Proc. Fifth International Workshop on Expert Systems and Their Applications, Avignon, France, May 1985.

McCarthy,J.[84], "Applications of Circumscription to Formalising Common Sense Knowledge", Proc. AAAI Workshop on Non-monotonic Reasoning, NY, Oct, 295-324.

McDermott,D.V. and Doyle,J.[80], "Non Monotonic Logic 1", Artificial Intelligence, Vol 13, pp 41-72.

Moore,R.C. [83], "Semantical Considerations on Nonmonotonic Logic", IJCAI-88pp 272-279.

Poole,D.L. [82], The Theory of CES: A Complete Expert System, Ph.D. Dissertation, Department of Computer Science, Australian National University, October 1982.

Poole,D.L [84a], The Logical Definition of Deduction Systems, Computer Science Research Report CS-84-12, University of Waterloo.

Poole,D.L. [84b], "A Logical System for Default Reasoning" AAAI Workshop on Non-monotonic Reasoning, NY, Oct, 373-384

Poole,D.L. [84c], A Computational Logic of Default Reasoning, Computer Science Research Report, University of Waterloo.

Reiter,R. [80], "A Logic for Default Reasoning", Artificial Intelligence, Vol 13, pp 81-132.

Reiter,R. and Criscuolo,G. [81], "On Interacting Defaults", IJCAI-81, pp. 270-276.

Touretzky,D.S.[84], "Implicit Orderings of Defaults in Inheritance Systems", Proc. AAAI-84, August 1984, pp 322-325.