A Skeptic's Menagerie: Confictors, Preemptors, Reinstaters, and Zombies in Nonmonotonic Inheritance

David S. Touretzky
School of Computer Science
Carnegie Mellon
Pittsburgh, PA 15213

Richard H. Thomason
Intelligent Systems Program
University of Pittsburgh
Pittsburgh, PA 15260

John F. Horty
Institute for Advanced Computer Studies
University of Maryland
College Park, MD 20742

Abstract

Subtle differences in the method of constructing arguments in inheritance systems can result in profound differences in both the results reached and the efficiency of inference. This paper focuses on issues surrounding the defeat of arguments in nonmonotonic inheritance. Looking primarily at skeptical reasoners, we analyze several types of defeat that may be encountered, especially the defeat of defeaters. Finally, we raise some questions specific to networks that mix strict and defeasible links.

1 Introduction

In earlier work we presented a skeptical approach to nonmonotonic inheritance reasoning [4] that differed in several respects from Touretzky's original credulous approach [15]. The main difference is that conflicting paths such as the well-known Nixon diamond generate multiple extensions in a credulous reasoner, while a skeptical reasoner produces a single extension in which all conflicted paths are excluded. As discussed in our "Clash of Intuitions" paper [16], inheritance systems of either type may differ in several other technical respects, such as the direction in which arguments are extended in computing inheritance (upward vs. downward reasoning), the precise definition of the preemption relation, the treatment of negative information, and the admission of strict (as opposed to defeasible) links.

In this paper we analyze another major point of difference among nonmonotonic reasoners: the treatment of defeated paths, primarily in skeptical systems. We define several types of possible interactions among paths according to the types of defeat involved. This "skeptic's menagerie" provides new insights into inheritance reasoning, and helps us to evaluate the computational consequences of alternative axiomatizations.

2 Defeat in Inheritance Systems

Let $\Gamma$ be an inheritance network containing defeasible positive or negative links of form $x \rightarrow y$ or $x \nrightarrow y$. A path $\sigma$ through a network is a sequence of links, e.g., the path $x \rightarrow y \nrightarrow z$ is composed of the links $x \rightarrow y$ and $y \nrightarrow z$. Let $\Phi$ be the extension of $\Gamma$ containing all paths through $\Gamma$ permitted by some upward skeptical inheritance rule, such as the one in [4]. By definition, $\Gamma \subseteq \Phi$. We say that $\Gamma$ "permits" $\sigma$ (written $\Gamma \triangleright \sigma$) iff $\sigma \in \Phi$.

Definition 1 A path $\rho$ is a situator of $x \rightarrow \tau \rightarrow w \nrightarrow y$ with respect to $x \rightarrow \sigma \rightarrow z \nrightarrow y$ iff $\rho$ has the form $x \rightarrow \gamma_1 \rightarrow w \rightarrow \gamma_2 \rightarrow z$. (Similarly for paths of opposite polarity.)

A situator establishes the "betweenness" required by the inferential distance metric for preemption to hold. The above definition produces "off-path preemption" [16], originally proposed by Sandewall [11]. For "on-path preemption" we require a closer relationship between the subject path and the situator: $x \rightarrow \sigma \rightarrow z$ must be a subpath of $\rho$.

Definition 2 A path $x \rightarrow \sigma \rightarrow z \rightarrow y$ is preempted in $\Gamma$ iff $\Gamma \triangleright x \rightarrow \tau \rightarrow w \nrightarrow y$ with a permitted situator. (Similarly for paths of opposite polarity.)

In the famous "Clyde the elephant" example, the path Clyde $\rightarrow$ Elephant $\rightarrow$ Gray is defeated by the preemper Clyde $\rightarrow$ Royal-Elephant $\nrightarrow$ Gray, since $\Gamma$ permits the situator Clyde $\rightarrow$ Royal-Elephant $\rightarrow$ Elephant. In our definition the situator must be a permitted path, but some inheritance schemes only require the two subpaths $x \rightarrow \gamma_1 \rightarrow w$ and $w \rightarrow \gamma_2 \rightarrow z$ to be permitted [2; 6]. Inheritance systems also disagree about whether the preemper must be permitted; some require only the initial segment $x \rightarrow \tau \rightarrow w$ to be permitted.

Definition 3 A conflictor of a path $x \rightarrow \sigma \rightarrow y$ is a path $x \rightarrow \tau \nrightarrow y$ having no permitted situator with respect to the former path. (Similarly for paths of opposite polarity.)

Definition 4 A path $x \rightarrow \sigma \rightarrow y$ is conflicted in $\Gamma$ iff $\Gamma \triangleright x \rightarrow \sigma$ and there is a conflictor $x \rightarrow \tau \nrightarrow y$ such that $\Gamma \triangleright x \rightarrow \tau$. (Similarly for paths of opposite polarity.)

In the classic Nixon diamond example, the conflicted paths are Nixon $\rightarrow$ Quaker $\rightarrow$ Pacifist and Nixon $\rightarrow$ Republican $\nrightarrow$ Pacifist. Each is a conflictor of the other because neither is situated with respect to the other.

Deflectionors, being unsituated, always generate multiple extensions in a credulous theory, or no conclusion in a skeptical theory, unless they are preempted by a direct link. We view direct links as preemptors and not deflectionors because a direct link $x \nrightarrow y$ is always situated with respect to any path from $x$ to $y$, therefore it cannot generate multiple extensions or no conclusion the way a true deflectionor can. Returning to the elephant example, the direct link Royal-Elephant $\nrightarrow$ Gray preempts the path Royal-Elephant $\rightarrow$ Elephant $\rightarrow$ Gray, but does not conflict with it.

Definition 5 An inheritance defeater is either a conflictor or a situated preemper.
The concept of a defeater as a general category of paths that interfere with a particular argument is due originally to Pollock [9, 10]. Since his overall theoretical framework is so different from ours, however, it is hard to find any exact correspondence between his notion of a defeater and the one presented here.

The definition of inheritability varies depending on whether one is using upward or downward reasoning. For upward reasoning we have:

Definition 6 A path $x \rightarrow \sigma \rightarrow y \rightarrow z$ is inheritable iff $x \rightarrow \sigma \rightarrow y$ is permitted and $y \rightarrow z \in \Gamma$. (Similarly for negative paths.)

All links in $\Gamma$ are permitted. Permission of longer paths is defined in terms of inheritability and defeat:

Definition 7 A path is permitted if it is inheritable and has no defeaters.

Finally, we need to define the conclusions an inheritance theory allows one to draw from a network:

Definition 8 $\Gamma$ supports the conclusion $x \rightarrow y$ iff its extension contains some path $x \rightarrow \sigma \rightarrow y$. (Similarly for negative paths.)

3 Defeater Defeaters

Some inheritance systems prohibit defeated paths from themselves acting as defeaters. In such a system, the ability of a path to act as a defeater may itself be defeated by some other path. Thus, these systems are said to admit 'defeater-defeaters,' another term due to Pollock, and later used by Loui [7]. Since in these systems defeating a path prevents it from defeating other paths, adding a defeater-defeater to a nonmonotonic network can reinstate a previously defeated path. Given the two types of inheritance defeat defined in the preceding section, there are five types of defeater-defeater in an inheritance system:

- preemptor-preemptor
- preemptor-confictor
- conflictor-preemptor
- conflictor-confictor
- situator-preemptor

These different types of defeater-defeater can have different effects in inheritance networks, and in some cases there is disagreement about the best way to handle them. We shall consider several of these issues in the following sections.

4 Reinstatement

Consider the network in Figure 1. The path $X \rightarrow \text{Chicken} \not\rightleftharpoons \text{Flies}$ is a preemptor of $X \rightarrow \text{Chicken} \rightarrow \text{Bird} \rightarrow \text{Flies}$; it is situated by $X \rightarrow \text{Chicken} \rightarrow \text{Bird}$. It is not necessary that $X \rightarrow \text{Chicken} \not\rightleftharpoons \text{Flies}$ actually be a permitted path for it to be a preemptor. (In fact, $X \rightarrow \text{Chicken} \not\rightleftharpoons \text{Flies}$ is defeated by the direct link $X \rightarrow \text{Flies}$.) But suppose we decided that preemptors must be permitted. This would cause the longer path to go through, since it has no permitted preemptor. In this case the path $X \rightarrow \text{Flies}$ would be acting as a "reinstater" of $X \rightarrow \text{Chicken} \rightarrow \text{Bird} \rightarrow \text{Flies}.

1This missing type, situator-confictor, will be discussed in section 5.

Figure 1: The intuitiveness of reinstatement seems to depend on the identity of the node X.

Figure 2: Conflictores can be reinstaters only in credulous theories.

Definition 9 A reinstater is a path whose permission defeats preemptors of other paths to the same conclusion, thereby allowing them to also go through.

Reinstaters are a type of defeater-defeater; specifically, they are preemption defeaters. In skeptical systems, reinstaters must be preemptor-preemptors, not preemptor-conflictores, since a reinstating path by definition must be permitted; in a skeptical system conflictors are never permitted. But in credulous reasoners conflictors will be permitted in some extensions, and can therefore act as reinstaters there. In Figure 2, for example, note that $x \rightarrow p \rightarrow y$ is a conflictor of $x \rightarrow q \not\rightleftharpoons y$, which is in turn a preemptor of $x \rightarrow q \rightarrow r \rightarrow y$. The path $x \rightarrow p \rightarrow y$ will be permitted in one of the two cedulous extensions, and in that extension it can reinstate $x \rightarrow q \rightarrow r \rightarrow y$ if the inheritance axioms permit reinstatement.

Reinstatement may not be desirable in inheritance reasoning. Suppose node X in Figure 1 stands for the concept "chicken with a jet pack." We have directly asserted that chickens with jet packs can fly, but the reason for their flying has nothing to do with their being birds. Therefore the
direct link does not logically support the argument the rein-
statement path represents, which is that X can fly because birds
fly. In general, the problem with reinstaters is that they al-
low any defeated argument to go through if its conclusion
holds, even if the argument’s reasons are undercut by more
specific information. Therefore we conclude that reinstaters are
undesirable.

To test our example further, suppose that X stands for
the concept “wild chicken,” and that wild chickens, having
stronger wings than their domestic cousins, can fly. In this
case the wild chicken’s flying ability really is a consequence
of its being a bird. This may seem to be evidence in favor
of reinstatement, but what it really shows, we believe, is that
the network of Figure 1 doesn’t capture all of our knowledge
about the relationship of wild chickens to ordinary chickens.
In particular, it doesn’t express the fact that the reason why
wild chickens fly is that they cancel precisely those excep-
tional properties of chickens that prevent them from flying.
Figure 3 shows one way to express this knowledge. The
network mixes strict and defeasible links, and under the def-
inition of inheritance given in [3], the path Wild-Chicken ⇒
Chicken ⇒ Bird ⇒ Strong-Wings ⇒ Flies is permitted. It is
not, however, “reinstated” (as we have defined the term), as
there is no preemtior that could have defeated it. This fol-
Ews from the fact that no positive path from Wild-Chicken
can reach Weak-Wings.

Reinstatement can affect not just the arguments a network
permits, but also the conclusions it reaches. In Figure 4, if
we do not allow reinstatement, the link x → o does not cause
Γ to permit the situator path x → m → n → o. Therefore
the two paths x → m → y and x → o → y conflict, and so
a skeptical reasoner will draw no conclusion about whether x
is a y. But if reinstatement is allowed, the permitted situator
x → m → n → o causes x → m → y to preempt x → o → y,
causing us to conclude x ∴ y.

Although reinstatement may be semantically undesirable,
the parallel marker propagation algorithms described in [4]
for defeasible nets and [17] for mixed nets depend on this
property in order to compute preemption efficiently. One of
the lemmas in the correctness proof of the algorithm in [4]
states (roughly) that if x → z_i ∈ Γ_i, z_i → z_{i+1} ∈ Γ_i for i from
1 to n − 1, and Γ_i supports x → z_i for i from 1 to n, then
Γ ∼ x → z_1 → y. It marks all nodes with M_T
having a direct positive or negative link to y with
mark M_{dir}; both m and o are so marked. Next, any nodes
on preempted paths must be eliminated. A preemption mark,
M_pre, is used to kill off nodes whose link to y is preempted
by some lower node. This is done in a very simple way: Ev-
every node with M_{dir} sends M_pre up “→” links, and M_pre is
propagated up “→” links into nodes with M_T
as far as it will go. So node m first sends M_pre to node n,
and node o sends it nowhere because no node above o has M_T.
Propagation then causes M_pre to spread upward from n to o,
killing off. Finally, only those nodes that have M_{dir} and not
M_pre are allowed to send M_T
up “→” links or M_F
up “∼” links to y. Since m is the only node meeting these conditions,
y is marked with M_F,
meaning Γ supports x ∼ y.

The important thing to note here is that the preemption
marker M_pre that originates at m flows from n to o precisely
because m, n, and o are all marked with M_T,
meaning Γ supports x → m, x → n, and x → o. But the preemption
is only valid because, as the lemma tells us, this guarantees
that Γ ∼ x → m → n → o. The lemma holds for the sys-

tem in [4] because its definition of inheritance requires that
preemptors not have preemptors of their own; this is what
produces reinstatement.

In more recent work, Hory has developed an alternative
axiomatization of skeptical preemption that does not admit
reinstaters [5]. In this scheme preemptors need not be permit-
ted; they can defeat other paths even if they themselves are
preempted by more specific paths.
It is an open question whether preemption without reinstatement can be computed efficiently without enumerating all possible situator paths. It cannot be done using just parallel marker propagation with a bounded number of markers. The reason is that, referring to Figure 4 again, there can be an arbitrary number of paths between m and α. Only one of these need go through in order to situate the preemperor \( x \rightarrow m \not\rightarrow y \) with respect to \( x \rightarrow \alpha \rightarrow y \), and each situator must be examined independently.

Thus, we see the possibility for an interesting tradeoff between inheritance definitions: some give the most “correct” results, while others have efficient algorithms that are correct in most cases, but will produce different results in certain situations.  

5 Defeat of Situators

The previous section looked at defeat of preemption by defeating the preemperor. It’s also possible to defeat preemption by defeating the situator. This form of defeat can result in a conflict (skepticism or multiple extensions) rather than the replacement of a conclusion with its opposite.

Figure 5 shows an example of a situator-preemperor. The path \( a \rightarrow d \rightarrow e \) has preemperor \( a \rightarrow b \not\rightarrow e \), with situator \( a \rightarrow b \rightarrow c \rightarrow d \). But the preemperor isn’t situated because the situator isn’t permitted; the link \( a \not\rightarrow c \) preempts its initial segment \( a \rightarrow b \). Unsituated preemperors are conflictors, so we must be skeptical about whether \( a \) is an \( e \).

Situator-conflictors are excluded from our list of defeater-defeaters because they have no independent effect. Let \( \rho \) be a path of form \( x \rightarrow \tau_1 \rightarrow w \rightarrow \tau_2 \rightarrow z \). Suppose \( \rho \) is a situator of \( x \rightarrow \tau \rightarrow w \not\rightarrow y \), a potential preemperor of the subject path, \( x \rightarrow \sigma \rightarrow z \rightarrow y \). Then a conflictor \( \xi \) of \( \rho \) must be of form \( x \rightarrow \mu \not\rightarrow z \), which also makes it a conflictor of \( x \rightarrow \sigma \rightarrow z \), an initial segment of the subject path. In all skeptical definitions, if an initial segment of a subject path is conflicted, the subject path is not even potentially inheritable; it will not be permitted. It therefore does not matter whether the subject path’s preemperors have permitted situators.

In credulous theories there are two cases to consider. Let \( \xi \) be a conflictor of the situator \( \rho \). In extensions that support \( x \not\rightarrow z \) (because \( \xi \) is permitted), the subject path has a conflicted initial segment and will not be inheritable. In extensions that support \( x \rightarrow z \), \( \rho \) will be a permitted path, and hence the situator will not be conflicted. So in credulous as well as skeptical theories, situator-conflictors do not function as defeater-defeaters.

6 Conflicted Paths

An important difference between defeat by preemperors vs. defeat by conflictors is that preempted paths cannot be extended further; they are dead. Conflicted paths will hold in some credulous extension, and thus can be extended there, perhaps giving rise to other instances of defeat.

Since skeptical reasoning as defined in [4] does not allow conflicted paths to be extended, one way of making a path go through is to conflict some initial segment of its defeater. In the double-diamond example reproduced in Figure 6, because Nixon is conflicted about Pacifist, the negative path to Anti-Military goes through unopposed. Thus, the path Nixon \( \rightarrow \) Republican \( \not\rightarrow \) Pacifist is acting as a conflictor-conflictor, another type of defeater-defeater.

An alternative notion of skepticism to the one presented here is one where the extension is the intersection of all credulous extensions. In [16] we called this “ambiguity propagation,” and Stein and others view it as a more rational or pure form of skepticism [14; 8]. Conflicted paths in such a system cannot be salvaged by conflictor-conflictors. This observation led Makinson and Schlechta to propose the notion of “zombies,” conflicted paths that are dead but can still kill other paths [8]. In their proposal, conflictor-conflictors would not be defeater-defeaters. Thus, even though we are conflicted about Nixon’s pacifism, we would still go on to form the zombie path Nixon \( \rightarrow \) Quaker \( \rightarrow \) Pacifist \( \rightarrow \) Anti-Military.
to conflict with Nixon → Football-Fan ≠ Anti-Military, causing us to be skeptical about the upper diamond in Figure 6 as well as the lower.

Stein [13; 14] and Makinson and Schlechta [8] have shown that the intersection of all credulous extensions may not support some of the conclusions that all extensions support. This occurs when a conclusion is reached via different paths in two extensions. (Makinson and Schlechta call this a “floating conclusion.”) Thus, marker propagation algorithms (which by our definition are limited to a constant number of markers) cannot support this ideal form of skepticism, since there is no way to keep track of which conclusions were reached in which extensions.

Recently, Schlechta has shown that no path-based approach to skeptical reasoning (for any “reasonable” form of skepticism) can produce the intersection of credulous extensions [12]. Hence, we cannot even axiomatize ideal skepticism in our purely path-based formalism. However, the analysis of various defeater-defeater situations reported here applies to other formulations of inheritance as well.

7 Defeat in Mixed Nets

Networks that mix strict and defeasible links, as in [3] and [5], use similar definitions of defeat to the ones presented here, except that strict extensions of paths must be taken into account. The strict extension of a path is the set of nodes reachable by purely strict links from the path’s conclusion. For example, a reinstarter need not have the same conclusion as the reinstated path; the reinstated path’s conclusion simply needs to be in the reinstater’s strict extension.

One very interesting idea for mixed nets is to require that situators be strict paths. This is in accordance with Brachman’s observation [1] that inclusion in natural hierarchies is strict; only properties are defeasible. Note that our proposal does not reduce the network to a simple class/property system as defined in [15], since it is still possible to chain off of defeasible links, defeasible inferences can have strict extensions, and they can generate conflicts. However, the preemption relation is simplified by requiring situators to be strict, since only defeasible paths have the potential to be reinstated, and these can never be situators.

Unfortunately, we still cannot allow reinstatement without affecting the conclusions the system will reach. Figure 7 shows an example of why this is true. The path $x \Rightarrow v \Rightarrow m \Leftarrow y$ preempts $x \Rightarrow v \Rightarrow u \Rightarrow y$, with a strict situator $x \Rightarrow v \Rightarrow u$. The preemptor is itself preempted by the direct link $x \Leftarrow m$. But with no reinstatement, this does not restore any path from $x$ to $y$. Therefore we should reach no conclusion about $y$. An axiomatization in which preemptors must be preempted (leading to reinstatement) would conclude that $x$ is a $y$.

8 Conclusions

Inheritance theory has a richer structure than we previously imagined. Complex patterns of defeat and reinstatement were known to exist, but had not been systematically analyzed. Even within a narrow family of reasoners, such as skeptical, upward, purely defeasible systems, we find that differences in axiomatization can affect both the results produced and the efficiency of inference.

Figure 7: A mixed network whose conclusions are affected by reinstatement.

There are two immediate observations to be drawn from our investigations. First, axiomatizations of inheritance should avoid reinstatement, because it is semantically undesirable. But, second, marker propagation systems$^3$ have implicitly relied on reinstatement, by assuming that preempts will be permitted paths, in order to compute permission efficiently. Since they also cannot implement ideal skepticism, we conclude that simple marker propagation architectures may not be as well-suited to inheritance reasoning as previously thought. They may still be useful as fast query/retrieval devices, provided that the correctness of the query algorithm is enforced by other means, such as the conditioning algorithms of [15], or is shown experimentally to provide correct results for most naturally occurring networks.

Acknowledgements

This work was funded by National Science Foundation grant IRI-9003165. We thank Bart Selman and one of the anonymous referees for helpful discussions.

References


$^3$As we have defined them, using a small, fixed number of markers with no internal structure.


