Proceedings of AAAI-90
Boston, Massachusetts
July, 1990

Boolean Extensions of Inheritance Networks

John F. Horty
Philosophy Department and
Institute for Advanced Computer Studies
University of Maryland
College Park, MD 20742
horty@umiacs.umd.edu

Richmond H. Thomason
Intelligent Systems Program
University of Pittsburgh
Pittsburgh, PA 15260
thomason+@cad.cs.cmu.edu

Abstract

Much of the theoretical research on nonmonotonic inheritance has concentrated on formalisms involving only IS-A links between primitive nodes. However, it is hard to imagine a useful network representation of commonsense or expert knowledge that would not involve nodes representing negative, conjunctive, or disjunctive properties. Certain nodes of this kind were included in some of the earliest formalisms for defeasible inheritance, but were omitted in later work, either to secure tractability or to simplify the task of theoretical analysis. The purpose of the present paper is to extend the theoretical analysis of defeasible inheritance to networks incorporating these expressive enhancements.

1 Introduction

Much of the theoretical research on nonmonotonic inheritance has concentrated on formalisms involving only positive and negative IS-A links between primitive nodes. However, even setting aside the need for relational reasoning, many knowledge representation applications call for extensions of this austere collection of network primitives. The following are typical instances of mechanisms that provide useful expressive power.

- *Negative antecedents.* We may wish to say that if a seat is unoccupied it is available.
- *Conjunction.* We may wish to say that purple mushrooms are poisonous.
- *Disjunction.* We may wish to say that members of congress are (by definition) either senators or representatives, and (by default) either democrats or republicans.

It is hard to imagine a useful representation of commonsense or expert knowledge that would not call for such definitions and assertions. Also, vital tasks like recognition of concept instances require expressive extensions. If, for instance, we enter three subtypes of sentences—declarative, imperative, and interrogative—along with information about the constituent structure of these three types, a recognizer must somehow know that a string of words that is not classifiable under any of these three types is not a sentence.

What all these expressive extensions have in common is the introduction of boolean node types into network formalisms. Such types were included in some of the earliest formalisms, such as that of Fahlman [4], but were omitted or restricted in later work, either to secure tractability or to simplify the task of theoretical analysis. If extensions of this sort are to be developed piecemeal, there would inevitably be much duplication of effort. In a system providing for negative antecedents, for instance, as well as in one introducing limited disjunctions, it would be necessary to provide separate inheritance definitions—and this would require, for example, a separate account in each case of when one reason for a conclusion should preempt another.

The purpose of the present paper is to present a framework providing criteria that apply generally to the design of boolean extensions of nonmonotonic inheritance networks: we develop a theory of defeasible inheritance for networks containing, in addition to nodes representing primitive kinds or properties, also nodes representing arbitrary boolean combinations of properties. As in [9], we allow for both strict and defeasible links. However, the treatment of reasoning conflicts here is credulous, rather than skeptical. This is for the sake of presentation only: the credulous theory is somewhat simpler, and the other ideas involved in boolean inheritance are already complicated enough.

It is hoped that the theory presented here will help to meet the challenges posed by Brachman [2] and Israel [10] for defeasible inheritance reasoners. These papers raise two serious problems for knowledge representation services employing defeasible inheritance reasoning: illicit cancellation and the need for classification. The former problem, which is simply that definitions create strict connections that should not be subject to exceptions, is solved by providing a mixed theory like that of [9], in which some conclusions cannot be canceled without introducing inconsistency into the network. The second problem is that, when defeasible links are present, it is not clear how a defined concept should be placed in an inheritance hierarchy. In view of the usefulness of the classifier in KL-ONE style systems, this challenge has to be taken very seriously.

The theory of [9] does not deal with this classification issue, because the mixed system of that paper does not provide for complex concepts. Since the present work, however, gives an account of the conclusions derivable from a defeasible network containing boolean-defined concepts, it provides also a specification for a classification algorithm in this boolean case. Admittedly, implementing such an algorithm would entangle us in intractability problems; but such entanglements also arise in purely monotonic settings. As far as we know at present, intractability problems are not necessarily worsened by the presence of defeasible links.

2 Basic concepts

2.1 Nodes and arguments

Nodes representing objects or individuals are depicted, as usual, by lowercase letters from the beginning of the alphabet (a through d). Nodes representing kinds or properties of individuals, however, may now be compound. Simple nodes, representing primitive properties, are depicted by lowercase letters from the middle of the alphabet (f through t). Compound property nodes are then obtained by closing the simple property nodes under the boolean operations of $\wedge$, $\vee$, and $\neg$; the nodes belonging to this closure, both simple and compound, will be depicted by uppercase letters from the middle of the alphabet (F through T). Capital letters from the end of the alphabet (U through Z) range over nodes in general, both individual nodes and property nodes.

Although we generally use infix notation in examples, we take the boolean operations of $\wedge$ and $\vee$ officially to operate on sets of nodes; thus, for instance, $F \wedge G \wedge H$ is the result of applying conjunction to the set $\{F, G, H\}$. (It will be convenient to take the conjunction and disjunction of a unit set to refer to that set's only member.) The compound property nodes should be interpreted in the obvious way: if F and G are properties, then the nodes the nodes $F \wedge G$, $F \vee G$, and $\neg F$ represent the property of being both an F and a G, the property of being either an F or a G, and the property of not being an F, respectively.

We allow for both strict and defeasible links. A *strict* link has the form $X \Rightarrow F$. If X is itself a property node, such a link is equivalent in meaning to a universally quantified material conditional; for example, the link $p \wedge q \Rightarrow r$ might represent the statement 'All purple mushrooms are poisonous'. If X is an individual node, a strict link will mean that this individual possesses the property F; the link $a \Rightarrow \neg(p \wedge q)$, for example, might represent the statement 'Albert is not a purple mushroom'. A *defeasible* link has the form $X \rightarrow F$. If X is a property node, such a link is to be interpreted as a defeasible generic statement; the link $p \wedge q \rightarrow r$ might represent the statement 'Purple mushrooms tend to be poisonous'. It is harder to find a natural reading for these defeasible links when X is an object, but we can assume that they represent defeasible assertions about the properties of individuals; the link $a \rightarrow \neg(p \wedge q)$, then, might represent a statement like 'It is most natural to suppose that Albert is not a purple mushroom'.

In theories of inheritance that allow only primitive nodes, reasoning processes can be represented by paths, or linear sequences of links. With the addition of compound nodes, however, inheritance networks are able to support more complicated patterns of reasoning. We represent these patterns using certain kinds of proof trees, called *arguments*; we refer to them by lowercase Greek letters (α, β, γ, ...).

Arguments will be classified as strict or defeasible, simple or compound; and with each argument α there is associated a premise node $\mathcal{P}(\alpha)$ and a conclusion node $\mathcal{C}(\alpha)$. These are analogous to the two nodes, in ordinary inheritance, standing at the beginning and the end of a reasoning path. An argument *supports* a conditional statement constructed out of its premise and conclusion. If α is a strict argument, it supports the statement $\mathcal{P}(\alpha) \Rightarrow \mathcal{C}(\alpha)$; if α is a defeasible argument, it supports the statement $\mathcal{P}(\alpha) \rightarrow \mathcal{C}(\alpha)$.

The simple arguments are those containing only one inference. If $X_1 \wedge \cdots \wedge X_n$ is a node, then the tree

$$\alpha = \frac{X_1 \quad \cdots \quad X_n}{F}$$

is a simple strict argument, with $\mathcal{P}(\alpha) = X_1 \wedge \cdots \wedge X_n$ and $\mathcal{C}(\alpha) = F$. Likewise, for any node X, the tree

$$\alpha = \frac{X}{F}\ .$$

is a simple defeasible argument, with $\mathcal{P}(\alpha) = X$ and $\mathcal{C}(\alpha) = F$. Compound arguments are defined inductively by the following clauses, in which the bracketed components are optional.

1. If $\alpha_1, \ldots, \alpha_n$ are strict arguments and $\bigwedge\{\mathcal{P}(\alpha_1), \ldots, \mathcal{P}(\alpha_n), [X_1, \ldots, X_m]\}$ is a node, then

$$\alpha = \frac{\alpha_1 \quad \cdots \quad \alpha_n \quad [X_1 \quad \cdots \quad X_n]}{F}$$

is a compound strict argument, with $\mathcal{P}(\alpha) = \bigwedge\{\mathcal{P}(\alpha_1), \ldots, \mathcal{P}(\alpha_n), [X_1, \ldots, X_m]\}$ and $\mathcal{C}(\alpha) = F$.

2. If $\alpha_1, \ldots, \alpha_n$ are arguments at least one of which is defeasible, and $\mathcal{P}(\alpha_i) = X$ for each α_i, then

$$\alpha = \frac{\alpha_1 \quad \ldots \quad \alpha_n \quad [X]}{F}$$

is a compound defeasible argument, with $\mathcal{P}(\alpha) = X$ and $\mathcal{C}(\alpha) = F$.[1]

3. If α_1 is any argument, strict or defeasible, with $\mathcal{P}(\alpha_1) = X$, then

$$\alpha = \frac{\alpha_1}{F}$$

is a compound defeasible argument, with $\mathcal{P}(\alpha) = X$ and $\mathcal{C}(\alpha) = F$.

In addition to specifying the arguments themselves, this definition allows us also to classify inferences contained in an argument as strict or defeasible: a double inference bar, analogous to the double-arrow link, indicates that the inference is strict; a single inference bar, analogous to the single-arrow link, indicates that it is defeasible.

The arguments defined here are intended as a generalization of the standard inheritance paths; a path is simply an argument that does not branch. To save vertical space, we will use ordinary path notation from [9] to refer to those arguments that can be identified with paths. For the same reason, we will occasionally write those compound arguments defined by the third clause above horizontally, as (α_1/F).

2.2 Nets, theories, and extensions

Capital Greek letters from the beginning of the alphabet $(\Gamma, \Delta, \Theta, \ldots)$ stand for *networks*, which are finite sets of links; those from the end of the alphabet $(\Phi, \Xi, \Psi, \ldots)$ stand for sets of arguments. Intuitively, the statements belonging to a network are supposed to represent the information provided as hypotheses to some reasoning agent. We imagine this agent developing a body of accepted arguments in stages, by a process of argument formation and ratification. Since arguments are a means of constructing other arguments, at any stage in this process there will be an *argument set*, consisting of patterns of reasoning that have been explicitly carried out and accepted.

The relation of support already defined between arguments and statements can be extended in the obvious way to a relation between argument sets and statement sets: an argument set Φ will be said to *support* a statement set Δ just in case Δ is the set of statements supported by the arguments in Φ.

The primary task for a proof-theoretic account of inheritance networks is to specify the *theories* associated with each network—the statement sets that an ideal reasoner could arrive at, given the information in that network as hypotheses. Following the strategy developed in previous work on inheritance networks, we approach this task here in a roundabout way. We first define the relation between a network and certain argument sets known as the *extensions* of that network; intuitively, these represent alternative argument sets that an ideal reasoner would be able to accept, based on the initial information contained in the network. Once this relation has been defined, it is then a simple matter to specify the theories associated with a network: Δ is a theory of the net Γ just in case there is an extension of Γ that supports Δ.

3 Inheritability

If Γ is a network and Φ is some set of arguments, we describe the pair $\langle\Gamma, \Phi\rangle$ as an *epistemic context*. Although, formally, any such pairing of a net and an argument set counts as a context, it is part of the intuitive picture that the argument set should arise out of the net. In any given context, certain arguments can be classified as *inheritable*—forcible or persuasive. We use the symbol '$\mid\!\sim$' to stand for this relation of inheritability, so that '$\langle\Gamma, \Phi\rangle \mid\!\sim \alpha$' means that the argument α is inheritable in the context $\langle\Gamma, \Phi\rangle$.

This notion of inheritability is the central concept in our proof-theoretic account of inheritance networks. In the present section, we set out an appropriate notion of inheritability for arguments. This notion will then be used in the following section to provide a definition of credulous extensions for inheritance networks containing compound nodes.

3.1 Motivation

By appealing to logic, it is easy to specify the conditions under which strict arguments, at least, should be classified as inheritable in a context. Suppose we have chosen some background logic for the boolean connectives; this logic will determine a consequence relation $\vdash$. In fact, the account of inheritance presented in this paper can be cast against a number of background logics, but for reasons described in Thomason et al. [12], the most promising candidate seems to be the four-valued logic of Belnap [1]. Now imagine, for a moment, that we interpret the nodes as propositions rather than properties; and that we supplement our background logic with the strict statements belonging to some net Γ as addi-

[1] The reason why a common premise is required when defeasible arguments are combined is that defeasible arguments can be compared and combined only with respect to the same total background evidence. This common evidence is represented here by the shared premise.

tional rules of inference. This leads to a new logic, with a new consequence relation $\vdash_\Gamma$. We say that a strict argument is Γ-*valid* if the conclusion of each inference in the argument, regarded as a proposition, follows from its premises in the logic $\vdash_\Gamma$; and we will say that such an argument is inheritable in the context $\langle \Gamma, \Phi \rangle$ if it is Γ-valid.

The intuitive idea behind this treatment of the strict arguments can be described as follows. Initially, we construe all nodes as properties (this includes individuals, since we can interpret the node a as the property of being a). Each strict link in the net then induces a new logical rule of inference involving a single free variable, say x: the link $p \Rightarrow q$, for instance, leads to the rule of inference $p(x)/q(x)$; the link $a \Rightarrow p$ leads to $x = a/p(x)$. We want to evaluate strict arguments against this new predicate logic determined by the net. However, since we are reasoning about statements containing only a single free variable, it suffices to treat the logic as propositional, and the nodes as propositions. Notice that, for the four-valued logic, and for any other reasonable logic of the boolean connectives, this treatment already provides, even for entirely strict arguments, a polynomial reduction of inheritability to an NP-complete problem.

We turn now to the matter of inheritability for defeasible arguments; and here, we focus on the special case of compound defeasible arguments ending in a defeasible inference. This is really the most interesting case, since it forces us to isolate the conditions under which an inference can be drawn using defeasible information. Once the case is understood, it is then a simple matter to embed it in a general definition of inheritability. For arguments of this kind, the account we provide is modeled on that of Touretzky [13]: such an argument will be classified as inheritable in a context if it is *constructible*, but neither *conflicted* nor *preempted*. These key concepts—constructibility, conflict, and preemption—are defined by Touretzky for paths. Our task here is to generalize them in such a way that they will apply to arguments as well.

The generalization is unproblematic in the case of constructibility. We adopt the following definition, which captures the idea that the arguments constructible in a context are those that can be assembled by chaining together, in a certain way, the arguments and links already present in that context.

Definition 1 (Constructibility) *Let α be an argument of the form (α_1/Y), with $C(\alpha_1) = U$. Then α* is constructible *in the context $\langle \Gamma, \Phi \rangle$ iff $\alpha_1 \in \Phi$ and $U \rightarrow Y \in \Gamma$.*

It is helpful to think of the final inference, or link, in a constructible argument as a reason for accepting that argument. Suppose, for example, that the agent

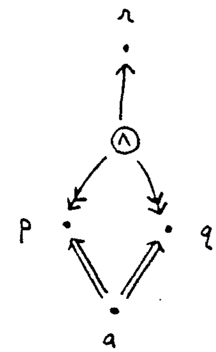


Figure 1: Γ_1

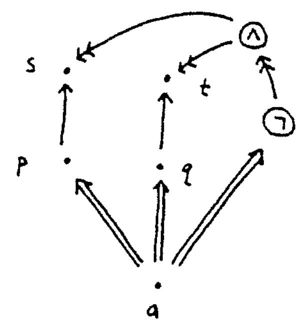


Figure 2: Γ_2

is given the net $\Gamma_1 = \{a \Rightarrow p, a \Rightarrow q, p \wedge q \rightarrow r\}$ as his initial information, where p = purple things, q = mushrooms, and r = poisonous things. This net is shown in Figure 1, in which the compound conjunctive node $p \wedge q$ is depicted as a conjunction sign with double-barbed arrows pointing at its conjuncts (disjunctive and negative nodes will be depicted in a similar fashion). Now let

$$\alpha = \frac{\begin{array}{cc} a & a \\ = & = \\ p & q \end{array}}{p \wedge q}$$

and $\beta = (\alpha/r)$; and suppose that the reasoning agent has already accepted the argument α, so that he is in the epistemic context $\langle \Gamma, \Phi \rangle$ with $\Phi = \Gamma \cup \{\alpha\}$. In this new context, the argument β is constructible. Since the agent has already accepted the argument that a is a purple mushroom, the link $p \wedge q \rightarrow r$, telling him that purple mushrooms are poisonous, gives him a reason to accept the argument that a is poisonous.

Constructibility is a necessary condition that an argument of this kind must satisfy in order to be classified as inheritable, but it is not sufficient. Even if an argument is constructible in some context, it will not be classified as inheritable if it is either conflicted or preempted.

The intuitive force of the requirement that a conflicted argument should not be inheritable is that the agent must check for consistency before accepting arguments. An argument cannot be classified as persuasive—even if there is some reason for accepting it—whenever the adoption of that argument would introduce a conflict into an epistemic context. The need for this requirement is usually illustrated by the familiar Nixon Diamond. Once an agent has embraced, say, the argument that Nixon is a pacifist, he can no longer accept the conflicting argument that Nixon is not a pacifist.

The generalization of the notion of conflictedness to nets with compound nodes is nontrivial, because the more powerful strict consequence relation provided by boolean nodes can make conflicts indirect. To illustrate this possibility, suppose that the reasoner is given as his initial information the net $\Gamma_2 = \{a \Rightarrow p, p \rightarrow s, a \Rightarrow q, q \rightarrow t, a \Rightarrow \neg(s \wedge t)\}$, depicted in Figure 2; and suppose also that he has already accepted the argument $a \Rightarrow q \rightarrow t$, so that his epistemic context is $\langle \Gamma_2, \Phi \rangle$ with $\Phi = \Gamma_2 \cup \{a \Rightarrow q \rightarrow t\}$. Here, the argument $a \Rightarrow p \rightarrow s$ would introduce a conflict into the context, even though the context does not contain an explicit argument to the contrary.

In order to generalize the notion of conflict to handle cases like this, we need to introduce a few preliminary ideas. First, where Φ is an argument set and X is a node, we let $\Phi(X) = \{\mathcal{C}(\alpha) : \alpha \in \Phi \text{ and } \mathcal{P}(\alpha) = X\}$. This set can be thought of as the projection of Φ onto X; it tells us what an agent who has accepted all the arguments in Φ has concluded in particular about X. Next, we say in the obvious way that a set of nodes $\mathcal{F}$ is a *conflict set in* Γ iff $\mathcal{F} \vdash_\Gamma Z$ and $\mathcal{F} \vdash_\Gamma \neg Z$, for some node Z; and that $\mathcal{F}$ is a *minimal* conflict set in Γ (a Γ-*mcs*) iff $\mathcal{F}$ but none of its proper subsets is a conflict set in Γ. Using these ideas, the notion of a conflicted argument can be defined as follows.

Definition 2 (Conflict) *Let α be an argument of the form (α_1/Y), with $\mathcal{P}(\alpha_1) = X$. Then α is* conflicted *in the context $\langle \Gamma, \Phi \rangle$ iff there is a Γ-mcs $\mathcal{F}$ such that $Y \in \mathcal{F}$ and $\mathcal{F} - \{Y\} \subseteq \Phi(X)$.*

In the example above, it can now be seen that the argument $a \Rightarrow p \rightarrow s$ is conflicted in the context $\langle \Gamma_2, \Phi \rangle$, since $\mathcal{F} = \{s, t, \neg(s \wedge t)\}$ is a Γ_2-mcs and $\mathcal{F} - \{s\} \subseteq \Phi(a)$.

The appeal to *minimal* conflict sets in the definition above is one way of capturing the idea that a path is conflicted only if it is somehow responsible for a conflict.[2] Why not simply say in the definition above that (α_1/Y) is conflicted in $\langle \Gamma, \Phi \rangle$ if $\Phi(X) \cup \{Y\}$ is a conflict set in Γ? The problem with this simpler formulation is that it does not properly capture the idea that a path is conflicted only if it *introduces* a conflict into a context; and as a result, it classifies too many paths as conflicted. Consider, for example, the net $\Gamma_3 = \{a \Rightarrow p, a \Rightarrow \neg p, a \Rightarrow q, q \rightarrow r\}$, depicted in Figure 3. Suppose the agent has drawn no inferences, so that he is in the context $\langle \Gamma, \Phi \rangle$, with $\Phi = \Gamma$. According to the simpler analysis, the argument $a \Rightarrow q \rightarrow r$ would have to count as conflicted, since $\Phi(a) \cup \{r\}$ is a conflict set. But according to the official analysis contained in our definition, $a \Rightarrow q \rightarrow r$ does not count as conflicted, since it does not introduce a conflict into the context. It seems that this official analysis of conflictedness is preferable: Γ_3 contains a little bit of inconsistency elsewhere, but it gives us no real reason not to conclude that a is a r.

[2] There may be other, more natural mechanisms for capturing this notion of blame for a conflict.

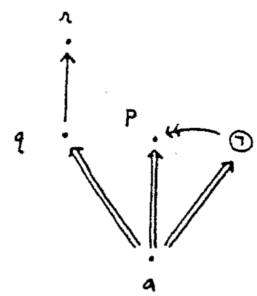


Figure 3: Γ_3

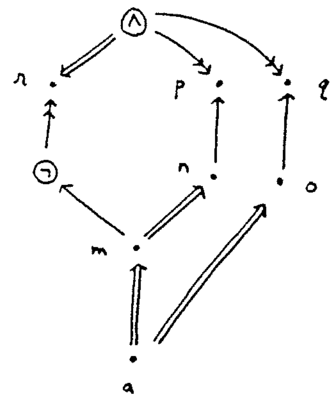


Figure 4: Γ_4

Like a conflicted argument, an argument that is preempted cannot be classified as inheritable. An ideal reasoner should not view an argument as persuasive whenever his context provides him with a more specific reason for accepting a conflicting argument. The notion of specificity appealed to here is carried over directly from simple inheritance: we say that $V \leq_\Phi U$ iff $U \in \Phi(V)$; and that $V <_\Phi U$, meaning that V is a specific kind of U, iff $V \leq_\Phi U$ but it is not the case that $U \leq_\Phi V$. Using this idea of specificity, the notion of preemption can be defined as follows.

Definition 3 (Preemption) *Let α be an argument of the form (α_1/Y), with $\mathcal{P}(\alpha_1) = X$ and $\mathcal{C}(\alpha_1) = U$. Then α is* preempted *in the context $\langle \Gamma, \Phi \rangle$ iff there are nodes V and W such that*

1. *$X \leq_\Phi V$ and $V <_\Phi U$,*
2. *$V \rightarrow W \in \Gamma$ and there is a Γ-mcs $\mathcal{F}$ such that $W, Y \in \mathcal{F}$ and $\mathcal{F} - \{W, Y\} \subseteq \Phi(X)$.*

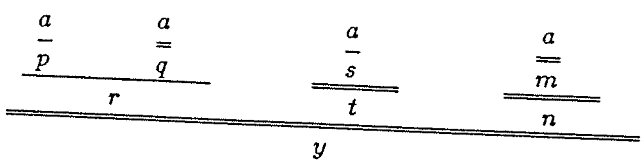
The first clause of this definition tells us that V represents a better reason than U for drawing conclusions about X; the second clause tells us that V suggests a conclusion about X that conflicts in the context with the conclusion suggested by U. Again, it would be simpler to replace the second clause with a stipulation to the effect that $V \rightarrow W \in \Gamma$ and $\Phi(X) \cup \{W, Y\}$ is a conflict set; but this would often lead to the wrong results. For example, with the simpler stipulation, the path $x \Rightarrow u \rightarrow y$ would be improperly preempted in the context $\langle \Gamma, \Gamma \rangle$, where $\Gamma = \{x \Rightarrow u, u \rightarrow y, x \Rightarrow v, v \Rightarrow u, v \rightarrow w, x \Rightarrow \neg w\}$.

To illustrate the notion of preemption, let us suppose that the reasoner is given as his initial information the net $\Gamma_4 = \{a \Rightarrow m, m \Rightarrow n, a \Rightarrow o, n \rightarrow p, o \rightarrow q, (p \wedge q) \Rightarrow r, m \rightarrow \neg r\}$, illustrated in Figure 4. Just to give the net some concreteness, we adopt the following interpretation: p = people with Disease-1, q = people with Disease-2, r = people exhibiting Symptom-3,

n = people who have spent time on Island-1, o = people who have spent time on Island-2, m = people who have spent time in some particular swamp on Island-1. Under this interpretation, what Γ_4 tells us is that the Island-1 people tend to acquire Disease-1, that the Island-2 people tend to acquire Disease-2, that anyone with both of these diseases must exhibit Symptom-3, that the swamp people tend not to exhibit this symptom, and that the individual a has spent time both on Island-2 and in the Island-1 swamp. Now suppose that the agent has reasoned his way to the epistemic context $\langle \Gamma_4, \Phi \rangle$, where $\Phi = \Gamma_4 \cup \{a \Rightarrow m \Rightarrow n, a \Rightarrow o \rightarrow q\}$; that is, he has already decided that a has Disease-2. In this context, he will find that the argument $a \Rightarrow m \Rightarrow n \rightarrow p$ is preempted by the more specific argument $a \Rightarrow m \rightarrow \neg r$.

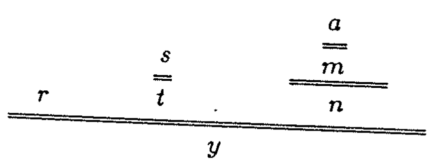
3.2 The definition

At this point, we can embed the special cases of inheritability that we have considered into a general definition. It is convenient to begin by introducing some notation, analogous to that of [9], for analyzing the structure of arguments. Where α is an argument tree, we let $\sigma(\alpha)$ be the maximal strict subtree of α beginning with its root inference; and we let $\delta(\alpha)$ be the set of argument trees that remains when $\sigma(\alpha)$ is truncated from α. These concepts are easier to illustrate than to define precisely; so suppose that α is the argument



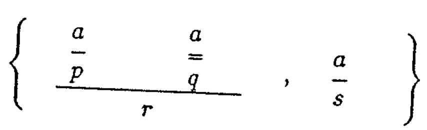
In that case, we would have

$\sigma(\alpha)$ =



and

$\delta(\alpha)$ =



Using this notation, we can classify the arguments according to their structure as follows. An argument α might end in a strict inference without without being entirely strict, in which case we would have $\sigma(\alpha) \neq \alpha$ and $\delta(\alpha) \neq \{\alpha\}$. Alternatively, the argument might be entirely strict, in which case we would have $\sigma(\alpha) = \alpha$, or it might end with a defeasible inference, in which case we would have $\delta(\alpha) = \{\alpha\}$. We use this classification of arguments to define inheritability for argument trees.

Definition 4 (Inheritability)

Case A: $\sigma(\alpha) \neq \alpha$ *and* $\delta(\alpha) \neq \{\alpha\}$. *Then* $\langle \Gamma, \Phi \rangle \mathrel{|\!\sim} \alpha$ *iff* $\sigma(\alpha) \in \Phi$ *and* $\delta(\alpha) \subseteq \Phi$.

Case B: $\sigma(\alpha) = \alpha$. *Then* $\langle \Gamma, \Phi \rangle \mathrel{|\!\sim} \alpha$ *iff* α *is* Γ*-valid.*

Case C-I: $\delta(\alpha) = \{\alpha\}$ *and* α *is a direct link. Then* $\langle \Gamma, \Phi \rangle \mathrel{|\!\sim} \alpha$ *iff* $\alpha \in \Gamma$.

Case C-II: $\delta(\alpha) = \{\alpha\}$ *and* α *is a compound path. Then* $\langle \Gamma, \Phi \rangle \mathrel{|\!\sim} \alpha$ *iff*

1. *α is constructible in* $\langle \Gamma, \Phi \rangle$,
2. *α is not conflicted in* $\langle \Gamma, \Phi \rangle$,
3. *α is not preempted in* $\langle \Gamma, \Phi \rangle$.

Here, Case A reduces the question of inheritability for a defeasible argument ending in a strict inference to two questions: inheritability for a strict argument and inheritability for arguments ending in a defeasible inference. These are treated in Cases B and C.

4 Credulous extensions

With the relation of defeasible inheritability secured, it is straightforward to define the credulous extensions of the inheritance networks containing compound nodes. Intuitively, an extension is supposed to represent some total set of arguments that an ideal reasoner would be able to accept, based on the initial information in some network. Therefore, we need only ask: what could prevent a path set Φ from representing such an ideal set of arguments determined by the net Γ? There are two possibilities. First, Φ might contain too few arguments; there might be some argument inheritable in the context $\langle \Gamma, \Phi \rangle$ that does not actually belong to Φ. Or second, Φ might contain too many arguments; some argument actually belonging to Φ might turn out not to be inheritable in the context. The credulous extensions of a net Γ can be defined as those path sets exhibiting neither of these defects.

Definition 5 *The argument set* Φ *is a* credulous extension *of the theory* Γ *iff* $\Phi = \{\alpha : \langle \Gamma, \Phi \rangle \mathrel{|\!\sim} \alpha\}$.

One requirement of the theory of skeptical mixed inheritance from [9] was that it should specialize to previously formulated skeptical theories of strict and defeasible inheritance. In the same way, it can be shown that the theory of boolean inheritance presented here specializes to the mixed theory of [7], a forward-chaining variant of the original credulous theory of inheritance from Touretzky [13]. Criteria guaranteeing the existence of extensions for boolean nets, and also the properties of soundness and stability, are established in [8], which describes a defeasible logic from which the present treatment of inheritance is descended.

5 Conclusion

The main result of the paper is a generalization Touretzky's familiar inheritance definition to networks containing boolean-defined nodes. We have concentrated on motivating this definition, and have not discussed the independent problem of providing a model theoretic semantics for the theory; this problem is complex and somewhat problematic even in the case of simple inheritance. It seems however, that certain model theoretic studies of simple inheritance networks, such as those of Gelfond and Przymusinska [5] or Ginsberg [6], could be extended to the boolean system of this paper.

In obtaining generality, our inheritance formalism also becomes intractable; as we have pointed out, there is a polynomial reduction of inheritance in our system to the problem of boolean satisfiability. We have tried to keep an open mind on whether it is best to secure a polynomial inheritance algorithm at all costs, or to provide expressive adequacy even if this requires intractable algorithms. It remains to be established that special-purpose representation problems can be fully supported by tractable algorithms, so that the representation tasks can be served without user-defined calls to the programming language that underlies the representation system. In the case of medical applications, for example, Doyle and Patil [3] argue forcefully that expressive extensions are needed for KL-ONE style systems. The special-purpose problem of representing syntactic structures in natural languages provides another case in point. So far, the formalisms that have been developed for this purpose have been highly intractable, though in this application one would hope for tractability (see Shieber [11] for background).

Whether we seek tractable fragments of the full theory or begin with a powerful theorem-proving implementation of the full system, it seems impossible to avoid the need for careful testing of the system under field conditions. Both sorts of systems need to be tested for expressive adequacy and for practical efficiency, but with different emphases in the testing. The theory of this paper serves as a first step in designing implementations of both sorts.

Acknowledgments

This material is based on work supported by the National Science Foundation under Grant No. IRI-8700705, and by the Army Research Office under Grant No. DAAL-03-88-K0087.

References

[1] N. Belnap. A useful four-valued logic. In *Modern Uses of Multiple-valued Logic*, J. Dunn and G. Epstein (eds.), D. Reidel (1977), pp. 8–37.

[2] R. Brachman. "I lied about the trees" or, defaults and definitions in knowledge representation. *The AI Magazine*, vol. 6 (Fall, 1985), pp. 80–93.

[3] J. Doyle and R. Patil. Two dogmas of knowledge representation: language restrictions, taxonomic classification, and the utility of representation services. MIT/LCS Technical Report 387.b (1989).

[4] S. Fahlman. *NETL: a System for Representing and Using Real-world Knowledge*. The MIT Press (1979).

[5] M. Gelfond and H. Przymusinska. Formalization of inheritance reasoning in autoepistemic logic. Unpublished manuscript, University of Texas at El Paso (1990).

[6] M. Ginsberg. A local formalization of inheritance: preliminary report. Unpublished manuscript, Stanford University (1990).

[7] J. Horty. A credulous theory of mixed inheritance. Forthcoming in *Inheritance Hierarchies in Knowledge Representation*, M. Lenzerini, D. Nardi, and M. Simi (eds.), John Wiley and Sons (1990).

[8] J. Horty. Defeasible arguments: an extension of path-based reasoning. Technical Report, Institute for Advanced Computer Studies, University of Maryland (1990).

[9] J. Horty and R. Thomason. Mixing strict and defeasible inheritance. *Proceedings of AAAI-88*, Morgan Kaufmann (1988), pp. 427–432.

[10] D. Israel and R. Brachman. Some remarks on the semantics of representation languages. In *On Conceptual Modeling: Perspectives from Artificial Intelligence, Databases, and Programming Languages*, M. Brodie, J. Mylopoulos, and J. Schmidt (eds.), Springer-Verlag (1984).

[11] S. Shieber. *An Introduction to Unification-based Approached to Grammar*. Center for the Study of Language and Information, Stanford, CA, (1986).

[12] R. Thomason, J. Horty, and D. Touretzky. A calculus for inheritance in monotonic semantic nets. In *Methodologies for intelligent Systems*, Z. Ras and M. Zemankova (eds.), North-Holland (1987), pp. 280–287.

[13] D. Touretzky. *The Mathematics of Inheritance Systems*. Morgan Kaufmann (1986), 220 pp.