A Presumptive System of Defeasible Inference

Ronald P. Loui
Department of Computer Science and
Department of Philosophy
The University of Rochester
Rochester, NY 14627

TR 190
May 1986

Abstract

This paper presents a system of non-monotonic reasoning with defeasible rules that is as presumptive as possible (as bold as possible), while still being warranted. The advantage of such a system is that many multiple extension problems can be solved without additional explicit knowledge; ordering competing extensions can be done in a natural and defensible way, with mere implicit knowledge. The objectives closely resemble Poole's objectives.

But the logic is different from Poole's. The most important difference is that this system allows the kind of chaining that many other non-monotonic systems allow. Also, the form in which the inference system is presented is quite novel for an AI system. It mimics an established system of inductive logic, and it treats defeat in the way of the epistemologist-philosophers. The focus is syntactic, and the limitations of resource-bounded theorem-proving can be treated formally.

The contributions are both of content and form: (content) the kinds of defeat that are considered, and (form) the way in which defeat is treated in the rules of inference.

The author is supported by a grant from the U.S. Army Signals Warfare Center.

REPORT DOCUMENTATION PAGE

Form Approved
OPM No. 0704-0188

1. AGENCY USE ONLY *(Leave Blank)*

2. REPORT DATE: 1986

3. REPORT TYPE AND DATES COVERED: Unknown

4. TITLE AND SUBTITLE: A Presumptive System of Defeasible Inference

5. FUNDING NUMBERS: DAAB10-86-C-0567

6. AUTHOR(S): Ronald P. Loui

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES):
University of Rochester
Department of Philosophy
Rochester, NY 14627

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES):
U.S. Army CECOM Signals Warfare Directorate
Vint Hill Farms Station
Warrenton, VA 22186-5100

10. SPONSORING/MONITORING AGENCY REPORT NUMBER: 92-TRF-0020

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT: Statement A; Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)* This paper presents a system on non-monotonic reasoning with defeasible rules that is as presumptive as possible (as bold as possible), while still being warranted. The advantage of such a system is that many multiple extension problems can be solved without additional explicit knowledge; ordering competing extensions can be done in a natural and defensible way, with mere implicit knowledge. The objectives closely resemble Poole's objectives.

But the logic is different from Poole's. The most important difference is that this system allows the kind of chaining that many other non-monotonic systems allow. Also, the form in which the inference system is presented is quite novel for an AI system. It mimics an established system of indictive logic, and it treats defeat in the way of the epistemologist-philosophers. The focus is syntactic, and the limitation of resource-bounded theorem-proving can be treated formally.

14. SUBJECT TERMS: Artificial Intelligence, Data Fusion, Defeasibe Inference, Presumptive System, Truth Maintenance, Real-World Knowledge

15. NUMBER OF PAGES: 20

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

A Presumptive System of Defeasible Inference

R. P. Loui
University of Rochester
Depts. of Computer Science and Philosophy
May 1986

Abstract.

This paper presents a system of non-monotonic reasoning with defeasible rules that is as presumptive as possible (as bold as possible), while still being warranted. The advantage of such a system is that many multiple extension problems can be solved without additional explicit knowledge; ordering competing extensions can be done in a natural and defensible way, with mere implicit knowledge. The objectives closely resemble Poole's objectives.

But the logic is different from Poole's. The most important difference is that this system allows the kind of chaining that many other non-monotonic systems allow. Also, the form in which the inference system is presented is quite novel. It mimics an established system of inductive logic, and it treats defeat in the way of the epistemologist-philosophers.

The contributions are both of content and of form: (content) the kinds of defeat that are considered, and (form) the way in which defeat is treated in the rules of inference.

keywords. defeat, defeasible, non-monotonic, multiple extension, inference.

subject category. knowledge representation.

1.1. Taxonomies, Motivations, and Apologies.

The traditional taxonomy of non-monotonic reasoning systems separates closed world assumptions and circumscription from default logics, NML, and conditional logic. I have no problem with this simple bifurcation. But there's another way of separating the systems and approaches. This separation is based on how the various systems relate the evidence, or suppositions, to the non-monotonic inferences.

First, there are the TMS-style systems, e.g. Doyle's and Goodwin's systems, and McDermott and Doyle's NML, and the philosophers' standard conditional logic. They do not distinguish between the basic evidence and the inferred propositions. Sentences belong simply to the unstratified mire of ins and outs.

Next, there are systems like Fahlman's and Poole's and Nute's which distinguish the evidence from that which is subsequently made evident. Philosophers' inductive logics do the same. Default logic talks as if there is a distinction. When theories are extended, there is a clear distinction between theory and extension, i.e., between evidence and conclusion. In circumscription, too, we talk as if it is easy to distinguish that which was circumscribed from that which was implied by circumscription. Poole and Fahlman, and the conditional logics, try to enforce a further distinction between necessary and contingent facts. Hanks and McDermott enforce temporal distinctions among facts about situations.

Finally, there are systems like the Glymour and Thomason theory revision system. Their system orders a subset of the *evidence* temporally. There is an initial theory, T. It is revised first by A. Then the resultant theory is revised by B. And so on. The epistemological building blocks here have even more structure than before. I include in this third approach systems that do not order the evidence, but instead use additional information to order the *defeasible rules*. Rich's likelihood approach and Shastri's evidential approach ask for explicit orderings.

Multiple extension problems are handled with varying effectiveness among these approaches. Not surprisingly, the latter approaches afford the most leverage for dealing with such problems. Separating evidence from conclusions allows competing extensions to be judged on their closeness to the evidence. Ordering evidence or rules allows further discrimination of competing extensions.

This paper is concerned with multiple extension problems and how to deal with them effectively. It presents new inference rules, for a system of defeasible inference. More importantly, it points out some new ways in which non-monotonic systems can be strengthened.

The third approach in my taxonomy is theoretically the most effective: methodologically presupposing that evidence or rules can be ordered by recentness or significance. However, I have used simpler epistemological building blocks, as much for reasons of practicality as for aesthetics. The present approach will consequently fall in the second class of approaches. It will nevertheless push the paradigm as fas as possible, with respect to solving multiple extension problems. In some places, goes even beyond Poole's work, solving multiple extension problems when he can't. In other places, my system is more cautious than Poole's. In the end, I'll compare my system to Poole's system, since his is the closest to mine, in terms of goals and intuitions.

Strictly speaking, multiple extensions belong in the province of default logic. I have used the phrase generally, and also said that multiple extensions are problems. I mean to refer to a general situation that commonly challenges the knowledge engineer. Now is the time to be more precise.

Suppose some subset of the evidence is prima facie evidence for a conclusion, and independently, some other subset is prima facie evidence for one of the conclusion's contraries. I don't care how prima facie warrant is formalized, so long as there is some way of expressing this contention.

In an adequately defined non-monotonic formal system, such a situation won't be a problem. The legitimate inferences will be defined with respect to whatever knowledge has been provided (i.e., evidence and inferential connections). From the point of view of the formal system, there is no choice, so there is no problem. But there is still the problem for the person who wants to define such a system conscionably. How should such a situation be resolved consistently? It is the problem of answering this question that I refer to, with concern.

There's an easy out. Choose any arbitrary way to handle such situations, then leave it to the manager of the knowledge base to tailor her rules to the analytic rules of the system. Tailor the knowledge to fit the inference engine. Provide the knowledge that will combine with the system designer's chosen solution to produce the desired inferences. It is a defensible position, with this exception: I'm betting that the inference engine so conceived will miss some of the regularities of the desired inferential behavior. Perfectly good inference rules will be missing. I'm betting that what will happen will be like omitting modus ponens. Figuratively, knowledge engineers will write "a" and "if a then b" and will want "b" automatically, but won't get it.

I admit that the issues must be settled by experience. And there are too many logics like this one that need to be judged. Still, the regularities I have in mind are not born of a lark. They owe their origins to inductive logic, suitably abstracted so that they will yield a qualitative system. This is some a priori justification. I will thus be able to point out why I think I will win the above bet.

I'll still use the phrase "multiple extensions," to refer to inferences that would be possible prior to adopting some policy that decides among them.

1.2. Directives.

The first interesting aspect of this system is the form in which its (defeasible) inference rules are presented. They explicitly mention defeat and enumerate the various kinds of defeat. Defeat is much more complicated than being a member of an exception list. So the form of presentation will already be something new to A.I., borrowing from an established style among epistemologists. It is closely modeled after the inference rules described by Kyburg for inductive logic.

The inferences depend on the form in which non-monotonic inferential connections are encoded; the treatment is syntactic. Moreover, what can be inferred depends on the *form* in which a *subset* of the knowledge is encoded. This subset is the set of defeasible rules. With apologies to Pat Hayes, that's just the way it is. In fact, part of the ploy is systematically extracting information about the choice of extensions from the form of the rules provided.

Poole says his treatment is semantic. The virtues, he says, are that (1) it can be understood and justified *independently* of particular implementations, (2) it does not fall into the problems of shortest paths and the redundancy of the object language. My treatment shares these virtues, though is syntactic. In fact, nothing in principle prevents a semantic treatment; I just think the intuitions are better laid bare syntactically. One could freely replace my use of $\vdash$ with $\models$. In the end, our semantics would look very much alike; both would depend on the monotonic knowledge and the set of non-monotonic rules, though not on the form of the monotonic knowledge, or the form of the non-monotonic rules' antecedents or consequents.

The next interesting things about the system have to do with the varieties of defeat.

Let a >-- b mean b is inferrable when a is established, unless there is interference; i.e., that a is prima facie evidence for b. I'll use the >-- symbol when I mean to refer to some abstract non-monotonic rule, whether it is a connective or a meta-linguistic relation, and whatever its particular formal behavior turns out to be.

Most have noticed that something like a "specificity defeater" is needed for non-monotonic inference systems. Specificity says that when e_1 >-- h and $e_1 \wedge e_2$ >-- not-h, then on evidence $e_1 \wedge e_2$, infer not-h. The latter inferential connection defeats the former.

Specificity has not yet been distinguished from superior evidence. One advance of the logic of my system is this distinction.

Superior evidence can hold when there is not superior specificity. Let

$e_1 \wedge e_2$ >-- g_1; g_1 >-- h;

e_1 >-- $g_2 \wedge g_3$; $g_2 \wedge g_3$ >-- not-h;

and let e1 $\wedge$ e2 be evidence. The choice to chain through the first two non-monotonic rules relies on more evidence. But on the respective last steps of the chains, where the chains are led to contrary conclusions, the rule in the first chain does not have a more specific antecedent than the rule in the second chain. We don't know that g_1 entails $g_2 \wedge g_3$.

Conversely, superior specificity can hold when there is not superior evidence. Let

e >-- $g_1 \wedge g_2$; g_1 >-- h;

e >-- $g_1 \wedge g_2$; $g_1 \wedge g_2$ >-- not-h;

where e is the evidence. In the chain that leads to not-h, there is more specificity in the antecedent of the critical inference. But of the two chains, neither uses more evidence.

A rule that is embodied in the logic is that using more evidence is better than having more specificity. When one chain uses more evidence and another has more specificity, the one with more evidence prevails. There will be another defeater, discussed below, based on "directness." When directness and specificity compete, neither prevails; the inference chains interfere with each other. When either directness or specificity stand toe-to-toe with superior evidence, evidence always emerges victorious.

The directness defeater is the second gem hidden in this logic. It, too, is new to these kinds of systems. It is implied by a shortest path rule. But it is substantially different. It is arguably implicit in some breadth-first extension strategies. Among the two chains below, where the evidence is e, the former is more direct.

e >-- g_1; g_1 >-- h;

$e >\!\!-\!- g_1; g_1 >\!\!-\!- g_2; g_2 >\!\!-\!-$ not-h.

Among the next two chains, neither is more direct, unless $g1 \vdash g2$ or $g2 \vdash g1$:

$e >\!\!-\!- g_1; g_1 >\!\!-\!- h;$

$e >\!\!-\!- g_2; g_2 >\!\!-\!- g_3; g_3 >\!\!-\!-$ not-h.

Directness relies on there being a subset of intermediary conclusions. Unlike a "shortest path" rule, directness does not hold just because the intermediary conclusions are fewer.

Poole's generality and specificity conditions are like my specificity and directness conditions. The big difference will be in how the defeaters are combined when there is conflict or chaining. I have postponed detailed comparison until (section 3) after my formalism's presentation.

Finally, in my formalism, I have written defeasible rules as assertions in the meta-language. $\blacktriangleright\!\!-$ is going to be an infix, two-place, meta-linguistic relation. Symbolically, it has the same status as $\vdash$. Meta-linguistic assertions involving this relation are supposed to be supplied by the user. It's possible to make it a connective in the object language, but the treatment is better if it is a meta-linguistic relation. So this system requires that knowledge be supplied in both an object language and a meta-language.

2. Formalism.

L, a language is defined as usual. So there are $Pred_L$, $Term_L$, Var_L, Cn_L, AF_L, OF_L, etc., and Sn_L, the sentences of L.
Neg("P") is "$\neg$P"; Neg("$\neg$P") is "P".

$\Phi \blacktriangleright\!\!- \Psi$ reads "Φ in the absence of defeaters, is reason for Ψ,"
or just "Ψ why? Φ".
$\Psi \in Sn_L$. Φ can be $\in Sn_L$, or $\subseteq Sn_L$, which is notational convenience, just like $\vdash$.
In the meta-language, sentences of this form are D-$rules_{ML}$.
Φ is the antecedent of the rule. Ψ is the consequent of the rule.

Do not suppose that there are any interesting rules that govern $\blacktriangleright\!\!-$. It won't be closed under chaining, it won't be left-adjunctive or right-disjunctive. In fact, none of the assertions involving this relation can be synthesized, i.e., none follow from inference; all are supplied externally.

The logic of sentences with the new relation is not claimed to analyze an existing concept, such as "if A, then subjunctively conditionally, B", or "if A, evidently B", or "A is a prima facie reason for B". Rather, the new relation and its logic are axiomatic and are supposed to be useful for knowledge representat.on, and non-monotonic inference therefrom.

A database is any pair <EK, R> where
EK $\subseteq Sn_L$, the "evidential knowledge," is supplied;
and R $\subseteq$ D-rules$_{ML}$, the set of "defeasible rules," which must also be supplied.

For each database, we define a defeasible extension,
DK $_{(<EK, R>)} \subseteq Sn_L$, the "defeasible knowledge."
We leave off the subscript when it is unambiguous what database it extends.

I'll use single quotes when asserting that a sentence belongs to R, e.g., ' "a $\wedge$ b" $\blacktriangleright\!\!-$ "c $\vee$ d" ' $\in$ R. That just says that "a $\wedge$ b" $\blacktriangleright\!\!-$ "c $\vee$ d" is a defeasible rule supplied. I'll also try to avoid quoting meta-linguistic sentences whenever possible! My meta-linguistic quinean quotes are the same as for the object language.

$\Phi \; {}_1\!\blacktriangleright\!\!-_{<EK,R>} \Psi$

reads "Φ in the absence of defeaters entails Ψ, *with exactly one defeasible step*" and holds just in case
for some Γ, $\Lambda \in Sn_L$,
1. $\ulcorner\Gamma \blacktriangleright\!\!- \Lambda\urcorner \in$ R and
2. Φ, EK $\vdash \Gamma$ and
3. Φ, EK, $\Lambda \vdash \Psi$ and
4. (no monotonic redundancy) for all ξ, a proper subset of Φ:
   it is not the case that [ξ, EK $\vdash \Gamma$ and ξ, EK, $\Lambda \vdash \Psi$]

The relativization to <EK, R> is omitted when unambiguous, i.e. $\Phi \; {}_1\!\blacktriangleright\!\!- \Psi$.

This relation is just like links between steps in non-monotonic proofs.

Note that ${}_1\!\blacktriangleright\!\!-$ is left-adjunctive: i.e., "A" ${}_1\!\blacktriangleright\!\!-$ "C" does guarantee "A $\wedge$ B" ${}_1\!\blacktriangleright\!\!-$ "C". It is also right-disjunctive: i.e., "A" ${}_1\!\blacktriangleright\!\!-$ "B" does guarantee "A" ${}_1\!\blacktriangleright\!\!-$ "B $\vee$ C".

The system allows defeasible inferences from EK and R into DK. We could define "entailment in the absence of defeaters, *with multiple steps*," $|\!\blacktriangleright\!\!-$, but that will just be membership in DK. Eventually, we'll define the membership of DK in terms of EK and R. DK will not be monotonic with repect to monotonic growth of EK or R. DK and EK taken together are supposed to contain knowledge for subsequent action and practical deliberation.

EK is assumed consistent with respect to $\vdash$. It will be a theorem that DK is closed under $\vdash$.

All defeasible evidential connections are represented using the new relation. At the moment we can't say:

"A" ▸— not ("B" ▸— "C"),

or *if* ⊢ "A", *then remove* '"B" ▸— "C"' *from* R. Such assertions, explicit rule defeaters, are addressed later.

Open sentences in R are instantiable, and R is closed under instantiation. This is just a representational shortcut. If R contains ⌜Px⌝ ▸— ⌜Qx⌝, where x ∈ Var_L, then R contains ⌜Pa⌝ ▸— ⌜Qa⌝, where *a* ∈ $Term_L$.

The eventual goal is to define

A ∈ $DK_{(<EK, R>)}$ iff
  A defeats ⌜¬A⌝ in <EK, R> and . . .,

where A is in Sn_L. The present concern, therefore, is defining what kinds of defeat there will be. It will be done in terms of what kinds of support there is.

Consider digraphs with nodes labeled by sentences,
  where no two nodes have the same label, and
  the graph has a unique sink.
  Let nl(P) be the *node labeled* P, P ∈ Sn_L, and
  let Label(n) ∈ Sn_L, be the label of node n.
  The support of a node n of a graph G, support(n), is the set
    {Label(m) : <m, n> is an edge in G}.
    In figure 1, support(nl("A")) = { "B", "C" }.

G , e.g. figure 1, is a support graph of A (in <EK, R>) iff G
  is such a graph, with sink labeled A, and
  support corresponds to one-step defeasible entailment without redundancy, i.e.,
  for all nodes X,
    support(X) ı▸— Label(X),
  and sources correspond to evidence, i.e.,
  if s is a source of G, then EK ⊢ Label(s).

For any A ∈ Sn_L, there are many support graphs of A in <EK, R>.

It will be possible to distinguish canonical forms, but I won't do it formally; the definitions needn't make any distinctions. It is useful conceptually to think of canonical, or maximally supportive graphs of propositions. Canonical forms will also be useful during computation.

For instance, it is possible to conjoin onto any node label the entirety of the

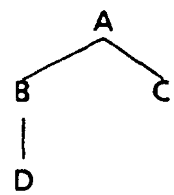

fig. 1. A support graph of "A".
Edges are directed upwards.
Quotes on labels are omitted.

This support graph corresponds to
"D ▸— B";
"B & C ▸— A";
EK ⊢ "D".

node's support (recursively). So the top node in figure 1 could have been "A & B & C & D". Because of the wording of the subsequent definitions, the recursive carrying of supports is uninteresting.

If there is support for contrary propositions, the support graphs will compete. In order to win the competition, every one of the losing proposition's graphs must be beatable by some graph of the winning proposition.

G_1 reflects G_2 iff

1. G_1 uses more evidence than G_2; or
2. It is not the case that G_1 uses more evidence than G_2 and
   a. G_1 is strictly more specific than G_2; or
   b. G_1 is strictly more direct than G_2; or
   c. G_1 has strictly preferred nodes to G_2.

Reflection is just the first step towards defeat.

Having more evidence corresponds to the total evidence requirement in inductive logics. It overrides all other considerations, so is treated specially. Specificity is just like the specificity defeater in other defeasible logics. It's like the subset defeater in Kyburg's system, and its use is implicit in conditioning probability measures. Directness arguably captures the intuition that warrant falls off with chaining through defeasible rules. Defeat can't be claimed just on the basis of how many such rules are used, however; directness here is more cautious and applies only when one chain is a subchain of the other. The last condition on preferred nodes is just a recursion step, which makes the other conditions like base cases. It turns out to be redundant, because of (*), below. It remains because it makes reflection more intuitive.

Here are some motivating examples, to which the reader is supposed to assent easily. Slightly more bothersome examples are presented with the definitions.

More evidence: EK = {"A", "B"};

R = { "A" ▸— C""; "A ∧ B" ▸— "D"; "D" ▸— " ¬C"};

" ¬C", "D" ∈ DK.

The chain through "D" just uses more of the evidence; it is better grounded in fact, so it is inferrable, and "C" is not.

A favorable example is A = "Amy is a new-waver"; B = "Amy is a college student"; C = "Amy is counter-cultural"; D = "Amy is fashionable".

More specificity: EK = {"A"};

R = { "A" ▸— "B ∧ C"; "B ∧ C" ▸— " ¬D"; "B" ▸— "D"};

"B ∧ C", " ¬D" ∈ DK.

Although "B" and "C" are not strictly classified as evidence, they are more specific than "B" alone. They act like evidence for the later conclusions. And rules with more specific antecedents are always preferred when there is conflict. An example is A = "Cheryl is a dancer"; B = "Cheryl is an active, physical person"; C = "Cheryl avoids risking knee injury"; D = "Cheryl will go skiing".

More directness: EK = {"A"};

R = { "A" ▸— "B"; "A" ▸— "C"; "B" ▸— " ¬C"};

"B", C" ∈ DK.

Although "B" is inferrable, it does not enjoy the same evidential status as "A". Inferences directly from "A" are preferred. Of the various kinds of defeat, this is probably the most sensitive to the language used. Still, whenever intervening sentences would seem to unnecessarily dilute an evidential connection obtained through chaining, e.g., "A" ▸— "B_1"; "B_1" ▸— "B_2"; "B_2" ▸— "B_3"; . . .; "B_{10}" ▸— "C", it should be asserted that "A" ▸— "C", directly. "C" is now strongly connected to "A", as desired.

A good example is A = "Donna is a senior"; B = "Donna studies"; C = "Donna parties".

Preferred nodes: EK = {"A", "B", "D"};

R = { "A" $\blacktriangleright\!\!-$ "C"; "A $\wedge$ B" $\blacktriangleright\!\!-$ "$\neg$C"; "C $\wedge$ D" $\blacktriangleright\!\!-$ "$\neg$E"; "$\neg$C $\wedge$ D" $\blacktriangleright\!\!-$ "E"};

"$\neg$C", "E" $\in$ DK.

Since "$\neg$C" defeats "C", "E" should defeat "$\neg$E". There is no other way in which "$\neg$C"'s preferability can be propagated, since nothing in DK can be used in determining $_1\blacktriangleright\!\!-$.

A plausible example of node preference is A = "Joanne works in a well-trafficked bar"; B = "Joanne is a nursing student"; C = "Joanne is a sexual health risk"; D = "Joanne is attractive"; E = "Rudolph will make a play for Joanne".

G_1 uses more evidence than G_2 iff
for every P in the sources of G_2,
the sources of $G_1 \vdash$ P.

The sources of a graph G is the set
{P : nl(P) is a source of G}.

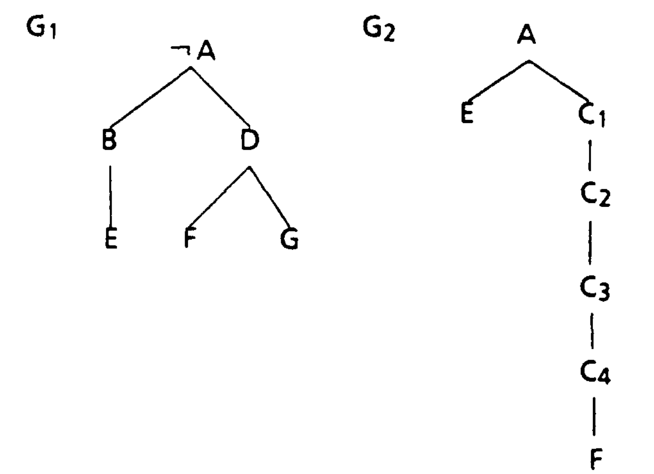

fig. 2. G_1 reflects G_2 because of more evidence despite the directness of E in G_2, and irrespective of the greater number of nodes in G_2.

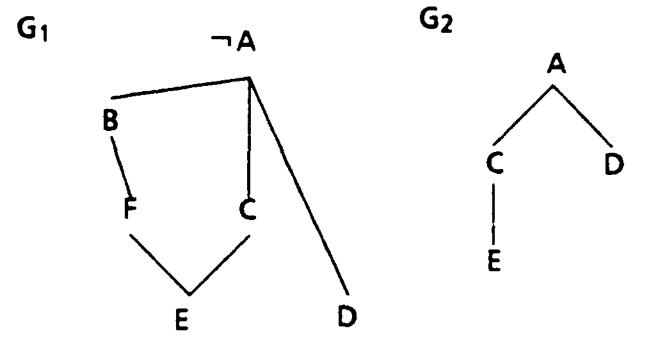

fig. 3. G_1 reflects G_2 because of strict specificity.

G_1 is strictly more specific than G_2 iff
for every P in the support of the sink of G_2,
the support of the sink of G1 $\vdash$ P.

Note that the recursion step having to do with preferred nodes simplifies this definition. We don't have to worry about specificity for the internal nodes.

G_1 is strictly more direct than G_2 iff
for every path, l, from source to sink in G_2, there is an m-short-cut of l in G_1.

l_1 is an m-short-cut of l_2 iff
if $<nl(A), nl(B)>$ is an edge in l_1, then
there is a sub-path of l_2 from some n_1 to some n_2 s.t.
$Label(n_1) = A$ and
$Label(n_2) = B$;
and $Label(source(l_1)) = Label(source(l_2))$
and $Neg(Label(dest(l_1))) = Label(dest(l_2))$.

There is no mention of $\vdash$ here because when searching for a reflecting graph, one is free to vary the labels with respect to $\vdash$. Recall that one-step entailment, 1▸—, allows monotonic deduction applied before the antecedent and applied after the consequent of the defeasible rule used.

G_1 has strictly preferred nodes to G_2 iff
for every node, n_2, of G_2, there is a node, n_1, of G_1 s.t.
a. $Label(n_1) = Label(n_2)$; or

b. Label(n_1) defeats Label(n_2);
and for some node, n_2*, of G_2, there is a node, n_1*, of G_1, s.t.
Label(n_1*) defeats Label(n_2*).

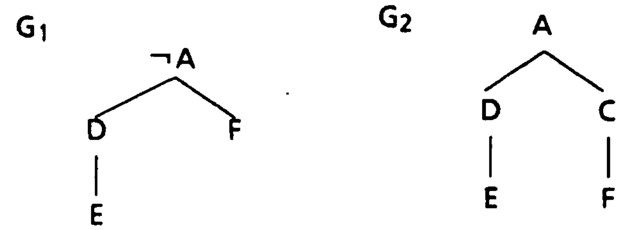

fig. 4. G_1 reflects G_2 because of strict directness.

fig. 5. G_1 reflects G_2 because of strictly preferred nodes.

G_1 is a reflecting support graph of G_2 iff
G_1 reflects G_2 and it is not the case that G_2 reflects G_1.

The relation of being a reflecting support graph is the natural anti-symmetric version of the reflection relation. This weakens the logic. It prevents defeat when there is one reason to defeat in one direction, but a different reason to defeat in the other. So in figure 6, neither "A" nor "¬A" is in DK, if G_1 and G_2 are the only support graphs.

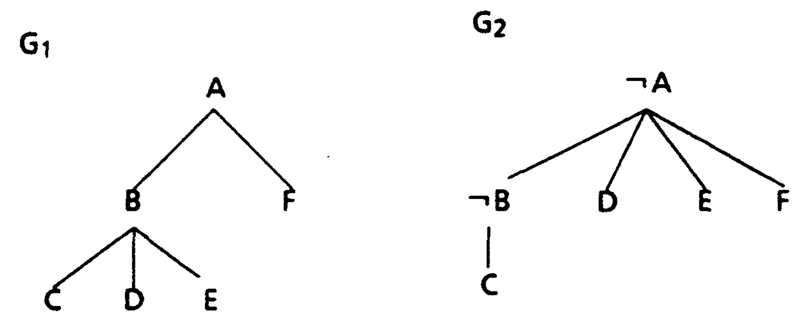

fig. 6. Neither "A" nor "¬A" is in DK, if G_1 and G_2 are the only support graphs. G_1 reflects G_2 because of the preferred node, (preferred because of more evidence). G_2 reflects G_1 because of strict specificity. So neither is a reflecting support graph of the other.

A defeats B (in <EK, R>) iff

A, B $\in Sn_L$,

A, EK $\vdash \ulcorner \neg B \urcorner$, and

for every support graph of B (in <EK, R>), G, there is a support graph of A (in <EK, R>), G', s.t. G' is a reflecting support graph of G.

A $\in$ DK iff

A defeats $\ulcorner \neg A \urcorner$ and

(*) there is some support graph of A (in <EK, R>), G, that is "grounded" in DK: i.e., for all n, if n is an interior node of G, then Label(n) $\in$ DK.

(*) is a recursion step. It prevents A from entering DK when it is based on chains, none of whose intermediary conclusions are in DK.

So much for the logic of ▸—.

It may look like a plumber's helper, but it's not; and it's not used for the same thing, either.

3. Comparison with Poole.

Poole's rules look like this:

1. Find two non-monotonic conclusions to choose between: g_1 and g_2.

e.g., g1 = "flies(edna)"; g2 = "$\neg$flies(edna)".

2. Find two sets of defeasible rules, which, together with the evidence, *allow* g_1 and g_2 respectively, via (just what anyone would expect) monotonic inference and non-monotonic modus ponens.

e.g., D_1 = {bird(x) >-- flies(x)}; D_2 = {emu(x) >-- $\neg$flies(x)}, given "emu(edna)" and "bird(x) $\vee$ $\neg$emu(x)".

3. Find an assertion, p_1, s.t. D_1 is solely applicable, i.e.,

a. D_1, the "necessary" facts, and p_1 allow g_1.

b. D_2, the "necessary" facts, and p_1 do not allow g_1 and do not allow g_2.

e.g., p_1 = "bird(edna)".

4. Fail to find an assertion, p_2, s.t. D_2 is solely applicable, i.e.,

a. D_2, the "necessary" facts, and p_2 allow g_2.

b. D_1, the "necessary" facts, and p_2 do not allow g_2 and do not allow g_1.

e.g., the only p_2 satisfying 4a is "emu(edna)". But "bird(x) $\vee$ $\neg$emu(x)" is a necessary fact. Thus, D_1, p_2, and the necessary facts allow "bird(edna)", hence "flies(edna)". 4b is violated. Therefore, 4 is satisfied.

5. Declare that D_2 is better than D_1 and therefore g_2 is preferred to g_1.

My basic intutions of more evidence, directness, and specificity are all implied by Poole's rules, for simple cases.

More evidence: Contingently, "A" and "B". D_1 = {A >-- C}. D_2 = {A $\wedge$ B >-- $\neg$C}. "A" makes D_1 applicable, but not D_2. Anything entailing "A $\wedge$ B" makes D_2 applicable, must also entail "A", thus make D_1 applicable. So "$\neg$C" is preferred.

Directness: Contingently, "A". D_1 = {A >-- B; B >-- C}. D_2 = {A >-- $\neg$C}. "B" makes D_1 applicable, but not D_2. Anything that makes D_2 applicable must entail "A", which makes D_1 applicable. So "$\neg$C" is preferred.

Specificity: Contingently, "A". D_1 = {A >-- B; B >-- D}. D_2 = {A >-- B; A >-- C; B $\wedge$ C >-- $\neg$D}. "B" makes D_1 applicable, but not D_2. Anything that makes D_2 applicable entails "A", which makes D_1 applicable, or entails "B $\wedge$ C", which also makes D_1 applicable. So "$\neg$C" is preferred.

But my system diverges from Poole when there is chaining. I have the preferred sub-node defeater. Poole is more cautious.

Chaining. Contingently, "A". D_1 = {A >-- B; B >-- C; C >-- D}. D_2 = {A >-- ¬C; ¬C >-- ¬D}. "C" makes D_1 solely applicable. "¬C" makes D_2 solely applicable. I will allow "¬D", but Poole will abstain. Both Poole and I prefer "¬C" to "C".

Poole and I also diverge over conflict between evidence and directness, or evidence and specificity.

Evidence versus Directness. Contingently, "A" and "B". D_1 = {A >-- C; C >-- E}. D_2 = {A ∧ B >-- D; D >-- ¬E}. "C" makes D_1 applicable, but not D_2. "D" makes D_2 applicable but not D_1. Poole won't choose between these, but I'll take D_2 and "¬E".

Evidence versus Specificity. Contingently, "A" and "B". D_1 = {A >-- C ∧ D ; C ∧ D >-- E}. D_2 = {A ∧ B >-- D; D >-- ¬E}. "A" makes D_1 applicable, but not D_2. "D" makes D_2 applicable but not D_1. Poole again won't choose between these; again, I'll take D_2 and "¬E".

There are also situations in which Poole chooses one theory over another, reaching a non-monotonic conclusion, where mine abstains. Consider a case wherein my specificity and directness defeaters compete without resolution, but nevertheless one set of reasons is more specific than the other in Poole's sense.

Poole's Specificity without Defeat in My Sense.Contingently, "D ∧ C ∧ E". D_1 = {D ∧ C ∧ E >-- A}. D_2 = {D >-- B; B ∧ C ∧ E >-- ¬A}. "B ∧ C ∧ E" makes D_1 applicable, but not D_2. Anything that makes D_1 applicable entails "B", which makes D_2 applicable. So "A" is preferred. But looking at figure 7, there is a support graph of A that is more direct than any support graph of ¬A. And there is a support graph of ¬A that is more specific than any support graph of A. So neither is preferred.

Again, the point is not that one choice is right or wrong, but that relative to Poole's, my system is sometimes too bold and at other times too cautious; and relative to mine, Poole's system is sometimes too bold and at other times, too cautious. His is more elegantly stated, but mine has been given more motivation. Mine allows natural chaining. Mine also takes superior evidence more seriously. His takes seriously the diversity of evidence and of intermediate conclusions.

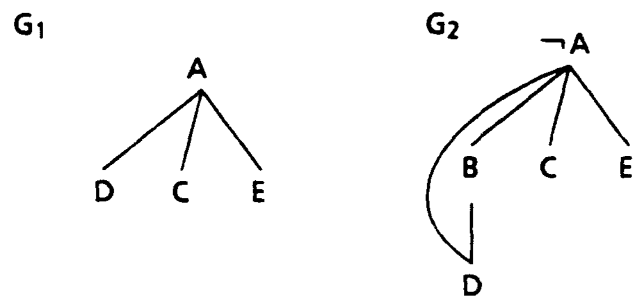

fig. 7. G_1 reflects G_2 because of directness; G_2 reflects G_1 because of specificity. A does not defeat ¬A in my system. But in Poole's system, {B, C, E} makes the G_2 rules applicable, but not the G_1 rules. So the G_1 theory is more specific; A is inferrable.

Appendix 1. Relativization to Limited Computation.

It's very easy to relativize this logic to some set of support graphs that are computable, noticeable, or apparent. Let $\Omega(A)$ be the computed support graphs of $A \in Sn_L$. Then change the definition of defeat:

A defeats B (in <EK, R>) iff
  $A, B \in Sn_L$,
  A, EK $\vdash$ ⌜¬B⌝, and
  for every support graph of B in $\Omega(B)$, G, there is a support graph of A in $\Omega(A)$, G', s.t. G' is a reflecting support graph of G.

The recursion step (*) in the definition of DK's membership should also be relativized to Ω.

Another natural relativization is to the set of inferences that are performed, rather than to the full obligations of $\vdash$. We can define)▸—, instead of ı▸—, which allows for habits of limited inference.

These relativizations make inferences non-monotonic not only in evidence and rules, but also in computation.

Appendix 2. Explicit Defeat of Defeasible Rules.

Until now, all defeat has come from conflicting consequents. It was not possible to write explicit defeat of defeasible rules. If ▸— had been a connective, we could have written "A ▸— ¬(B ▸— C)" or "A —▸ ¬(B ▸— C)" directly in Sn_L (the latter uses the standard material, truth-functional connective —▸). At present, the only way "A" can defeat "B" ▸— "C" is if A is a reason to infer "¬C", defeasibly or otherwise.

But sometimes in the presence of "A", the connection between "B" and "C" is simply defeated, and no conflicting alternative is suggested.

McCarthy's way of solving this is to tag rules with their exceptions, as conjuncts in the consequents. Instead of "B" ▸— "C", write "B" ▸— "C ∧ ¬X_A"; then write "A —▸ X_A", which now conflicts with the defeasible rule. The drawback of McCarthy's method is that rules in R will have to be modified every time an exception is added. If one of the above forms could be used, these exceptions could simply be added to EK, monotonically, without threat of subsequent revision.

The easiest way to add this feature to the system is to allow more complex assertions about membership in R. For A to defeat the connection between B and C, write

if '⊢ "A" ' is true *then* '"B" ▸— "C"' ∉ R.

This poor sentence must use the material conditional connective and the predicate "is true" from the meta-meta-language! This is because membership and non-membership in R already require the naming of sentences in the meta-language.

These explicit defeaters are not defeasible.

But if there is inference going on at the meta-meta-level, such knowledge can be stated and used with no problem. Then if ⊢ "A", i.e., if '⊢ "A" ' is true, and "B" ▸— "C" is found in R, there is an inconsistency (unless membership in R is defeasible, i.e., ▸▸— '"B" ▸— "C"' ∈ R, using the relation ▸▸— described below).

Appendix 3. Explicit Defeasible Defeat of Defeasible Rules.

What about something like "A ▸— ¬(B ▸— C)"? This would be simple to write if ▸— had been a connective. But it's easier to see what's going on meta-linguistically.

Consider a new relation in the meta-meta-language, which is to ▸— as ∧ is to &,

i.e., it's just the meta-meta-linguistic analogue of the meta-lingusitic ▸—. Let's use the symbol ▸▸— for this relation. Then we write

['⊢ "A" ' is true] ▸▸— ['"B" ▸— "C" ' ∉ R],

and include it in R, or some higher-meta-analogue of R. If A, (i.e., "A" is in EK, i.e., ⊢ "A", i.e., '⊢ "A" ' is true) and if there is no undefeated way of getting "B" ▸— "C" into R, this latest rule says that it isn't in R. We can define reflection and defeat to govern ▸▸—, just as it was done at the level below.

I suppose there would be problems if someone insisted that "A" in DK led to '⊢ "A" ' is true. Don't let him insist.

Acknowledgements.

Henry Kautz was most useful for discussing these ideas, which were also discussed with Rich Pelavin and Henry Kyburg. Pelavin's comments significantly improved the presentation, as did Kyburg's. I personally prefer approaches to defeasible reasoning that explicitly account for risk. Kautz, Pelavin, Kyburg, and David Israel provided encouragement.

Bibliography.

Doyle, J. "A Truth Maintenance System," *Artificial Intelligence 12*, 1979.

Etherington, D. and Reiter, R. "On Inheritance Hierarchies with Exceptions," Proc. AAAI-83, 1983.

Fahlman, S. *NETL: A System for Representing and Using Real-World Knowledge*, MIT Press, 1979.

Glymour, C. and Thomason, R. "Default Reasoning and the Logic of Theory Perturbation," Proc. AAAI Workshop on Non-Monotonic Reasoning, New Paltz, NY, 1984.

Goodwin, J. "WATSON: A Dependency Directed Inference System," Proc. AAAI Workshop on Non-Monotonic Reasoning, New Paltz, NY, 1984.

Hanks, S. and McDermott, J. "Temporal Reasoning and Default Logics", Yale Computer Science Research Report 430, 1985.

Harper, W., Stalnaker, R., and Pearce, G. *Ifs*, Reidel, 1981.

Kyburg, H. *Logical Foundations of Statistical Inference*, Reidel, 1974.

McCarthy, J. "Applications of Circumscription to Formalising Common Sense Knowledge," Proc. AAAI Workshop on Non-Monotonic Reasoning, New Paltz, NY, 1984.

McDermott, J. and Doyle, J. "Non-monotonic Logic I," *Artificial Intelligence* 13, 1980.

Nute, D. "A Non-monotonic Logic Based on Conditional Logic," working paper, Advanced Computational Methods Center, Atlanta, 1985.

Poole, D. "On the Comparison of Theories: Preferring the Most Specific Explanation," Proc. IJCAI-85, Los Angeles, 1985.

Shastri, L. "Evidential Reasoning in Semantic Networks," Technical Report, Dept. of Computer Science, U. Rochester, 1984.

Reiter, R. "A Logic for Default Reasoning," *Artificial Intelligence* 13, 1980.

Rich, E. "Default Reasoning as Likelihood Reasoning," Proc. AAAI-83, 1983.

Touretzky, D. "Implicit Orderings of Defaults in Inheritance Systems," Proc. AAAI-84, 1984.