

DEFEASIBLE INHERITANCE AND REFERENCE CLASSES

Laurent Audibert

Karl Schlechta

Laboratoire d'Informatique de Marseille, CNRS ESA 6077

CMI, Technopôle de Château-Gombert

F-13453 Marseille Cedex 13, France

ks@gyptis.univ-mrs.fr

<http://protis.univ-mrs.fr/> ~ ks

January 9, 1999

Abstract

We formalize how information from a reference class is used to augment the information of a base class. While theory revision operates on theories and formulas of the same language, the languages of the base and the reference class might and will be different in our approach. The information we consider is defeasible, and we examine two semantical approaches, one working on preferential structures expressing this information, the other working on the partial orders defined by the information. We show that our two approaches are equivalent. We finally apply these ideas to elucidate defeasible inheritance, choosing the reference classes via valid paths, and, conversely, we motivate the definition of valid paths with the reference class concept.

1 INTRODUCTION

Throughout, we work in a defeasible reasoning framework, and in finite propositional languages.

1.1 Motivation

This paper has two aims: First, to formalize, in a defeasible setting, the use of a reference class to augment information of a base class (Section 2). Second, to argue that there is

an important interplay between the concept of a reference class and defeasible inheritance formalisms. On the one hand, the simple framework of inheritance diagrams allows us to find the “right” reference classes and to decide which information shall be inherited from the reference classes (via valid paths, see the end of Section 3). On the other hand, our concept allows us to argue in favour of an upward chaining, on-path formalism (see the end of Section 3, too). Even if there are still many existing and imaginable inheritance formalisms which are upward chaining and on-path, we think that any systematic argument in favour of one or the other formalism is a welcome contribution to clarification. The lack of a convincing semantics has allowed a multitude of formalisms, which are often distinguished essentially by the more or less satisfactory way in which some examples are solved, and not by systematic arguments.

We use the concept of a reference class in the following sense. Let X be a set, and ϕ be some information that holds defeasibly for X , in the sense that “most $x \in X$ satisfy ϕ ”, or “normally, $\phi(x)$ for $x \in X$ ” etc. Sometimes, we shall then say “in X , things satisfy normally ϕ ” etc. “Most” is used in this article in a qualitative, not necessarily probabilistic sense, and only for the intuition. Let Y be another set, and most $y \in Y$ satisfy ψ . Suppose further that the language of ϕ is disjoint from the language of ψ . We use Y as a reference class for the base class X by augmenting the information about the normal state of affairs of X by the information about the normal state of affairs of Y (giving priority to the base class information). Thus, in our example, we will now conclude that most $x \in X$ satisfy ψ , too. Note that X might be a singleton $\{x\}$, and $x \in Y$, this is a special case.

We thus depart somewhat from the philosophical tradition, see e.g. the work of Reichenbach, [Rei49], and also Kyburg and Pollock, [Kyb74], [Kyb83], [Pol90] - we have neither the intention nor the competence to give a detailed comparison here. We just want to give a specific formal interpretation of the notion of reference class.

We keep the languages of base and reference class disjoint for four reasons: First, this seems to suffice in the context of defeasible inheritance. Second, this seems to correspond to the organization of human knowledge bases at least in some cases. Third, keeping languages separate favours small databases. Fourth, it seems a justifiable initial simplification.

This might be the place to compare our approach informally with theory revision. Revising T with ϕ , $T * \phi$, presupposes that both T and ϕ are formulated in the same language. In general, however, the language of A used as reference class for B , is different from - in our approach even disjoint to - the language of B . On the other hand, both approaches treat the in some sense minimal modification of one information by another one. The information imported from the reference class will usually be considered less reliable, because subject to some reasoning by analogy, than direct information, i.e. information of the base class. Thus, the reference class information can be compared to the old theory T in theory revision, and the direct information to ϕ .

As said above, one of the aims of this paper is to formalize how reference class information is used to augment the base information. We want to “add” reference class information

formulated in a reference class language \mathcal{L}_R to base information in a base language \mathcal{L}_B , resulting in some theory in a language \mathcal{L}_{B+R} . As we work in a defeasible setting, there are two natural semantical approaches. If both theories are expressed by preferential structures (over different languages) - see e.g. [KLM90], [LM92], [Sch92], [Sch97] for motivation, discussion and results for preferential structures - we will construct a preferential structure in the augmented language \mathcal{L}_{B+R} which in a reasonable way expresses the joint information. (Definitions 2.1, 2.2), In a more general case, defeasible information corresponds to coherent systems of filters or orders over formulas (= sets of models) - essentially by a definition like $\alpha < \beta$ iff $Con(\beta)$ and $\alpha \vee \beta \sim \neg\alpha$, where \sim expresses the defeasible information - see [BB94], [FH98], [Sch97-4]. In that case, we have to construct in a reasonable way an order (or coherent system of filters) on the augmented language from the orders (or systems) of the base and the reference languages. (Definition 2.5). In both cases, augmentation corresponds to the construction of a product structure from the individual structures representing the base and the reference class information. Abstractly, we take two nonmonotonic databases, transform them into semantical form (as preferential structures or partial orders), compute the product structure, and transform it back into the resulting joint nonmonotonic database.

As a preferential structure generates a partial order on formulas, it is a natural and basic formal question whether passing from preferential structures to the product structure and then to the partial order on the product gives the same result as going first from the preferential structures to the partial orders and then to the partial order on the product. This is answered positively in Proposition 2.8 (2). Thus, our definitions are in a certain way natural, as they seem well-behaved in elementary algebraic operations.

An important problem of reference classes is to determine which classes are the reference classes for a given base class. If we know that Tweety is a bird and a penguin, shall we take the candidate “birds” or the candidate “penguins” as reference class for Tweety’s flying abilities? We give an answer in the simple case of defeasible inheritance diagrams: Y is a reference class for X iff there is a valid positive path from X to Y .

A defeasible inheritance diagram Γ is a finite directed acyclic graph whose nodes stand (roughly) for sets, with two types of arrows $X \rightarrow Y$ and $X \not\rightarrow Y$, where $X \rightarrow Y$ ($X \not\rightarrow Y$) expresses roughly “elements of X are normally (not) elements of Y ”, or “most X ’s are (not) Y ’s”. We can reformulate this as “in the class X , things are normally (not) in Y ”, or “in the class X , most things are (not) Y ’s”. Recall that our “most” is a qualitative and not necessarily probabilistic one, which is not made precise and is there for intuitive purposes only.

The problem of inheritance diagrams is to find the “valid paths” in the diagram, and their conclusions. If, for instance, Γ is the diagram $X \rightarrow Y \rightarrow Z$, then it is universally accepted that the paths $X \rightarrow Y$, $Y \rightarrow Z$, $X \rightarrow Y \rightarrow Z$ are valid in Γ , and we conclude from the third path that in X , things are normally in Z . We justify this reasoning as follows: As the path $X \rightarrow Y$ is valid, not only “most X ’s are Y ’s”, but also Y is a reference class for X . Otherwise, why should X inherit from Y that things are normally

in Z ? Transitivity of reasoning is not admissible in a defeasible setting, and we see no other possible justification. Thus, we think that a suitable concept of reference class has to be at the basis of reasoning in inheritance diagrams, and we hope that our approach is a first step in this direction. On the other hand, in reasoning in inheritance diagrams - at least in the upward chaining variants - information “flows” only via valid (positive) paths, so reference classes seem to be found via such valid paths.

Consider now the diagram Γ' consisting of the arrows $X \rightarrow Y \rightarrow Z$, $Y \rightarrow Z'$, $X \not\rightarrow Z'$. Even if there is valid path $X \rightarrow Y$, there is no valid path $X \rightarrow Y \rightarrow Z'$ in Γ' , as the arrow $X \not\rightarrow Z'$ precludes it. So, not all information is inherited from a reference class, but only information corresponding again to valid paths.

In this interpretation, we need a valid initial segment ($X \rightarrow Y$) to inherit “most things are in Z ” from Y by the valid path $X \rightarrow Y \rightarrow Z$, so we are led to an upward chaining formalism.

In Γ , the base class X has the information that things are normally in Y , the reference class Y has the information that things are normally in Z , the language of the class X is (determined by) $Y(x)$, the language of Y is $Z(x)$, and the languages are disjoint, so we can apply our formalism. In Γ' , the languages of X and Y are not disjoint, but we can easily make them so.

1.2 Overview of the paper

In the remainder of this Section 1 we will introduce the basic definitions of preferential structures and of inheritance diagrams.

In Section 2, we will define the augmented preferential structure (Definitions 2.1 and 2.2), and the order on the augmented language (Definition 2.5), examine some formal properties of our definitions (Facts 2.1 - 2.7) and show their equivalence when appropriate (Proposition 2.8). Proposition 2.8 contains our main formal results. Section 2 is written almost entirely in an abstract algebraic style, we do not speak about sets of models there, but about arbitrary sets. Thus, our results are valid independent of logic and models. Note also that we restrict our discussion there to the case of one reference class, which is already sufficiently nasty, as can be seen in the proof of Proposition 2.8.

In Section 3, we will look at defeasible inheritance diagrams and analyse them using the concept of a reference class. In particular, this will allow us to decide in favour of an upward chaining formalism, our cautious approach will lead to on-path preclusion only. We will thus have good reasons to decide two questions in the multitude of inheritance formalisms.

1.3 Basic definitions

We will work in a finite situation, the base logic will be classical propositional logic (in a language with finitely many propositional variables).

Notation 1.1

We use \mathcal{P} to denote the power set operator.

Let \mathcal{L} be a propositional language, we denote by $v(\mathcal{L})$ the set of its variables, by $M_{\mathcal{L}}$ the set of its classical models, ϕ etc. shall denote formulas, T etc. theories (sets of formulas) in \mathcal{L} and $M(T) \subseteq M_{\mathcal{L}}$ the models of T , likewise $M(\phi)$. $Con(\cdot)$ stands for (classical) consistency.

Definition 1.1

(Preferential structures, simplified: the labelling function is the identity, or, equivalently, no multiple copies of models.)

(1) $\mathcal{Z} = \langle X, \prec \rangle$ is called a preferential structure iff X is a set, and \prec a binary relation on X .

(2) If $\mathcal{Z} = \langle X, \prec \rangle$ is a preferential structure, then $x \in X$ is called a minimal element of X iff $x \in X$ and there is no $x' \in X$ s.t. $x' \prec x$.

(3) If $\mathcal{Z} = \langle X, \prec \rangle$ is a preferential structure, and X a set, then $\mu_{\mathcal{Z}}(X) := \{x \in X : x \text{ is a minimal element of } X\}$. When the context is clear, we write μ for $\mu_{\mathcal{Z}}$.

Note: If A is finite, \prec acyclic, $A \neq \emptyset$, then $\mu(A) \neq \emptyset$.

(4) A relation \prec on X is called ranked iff there is a set O totally ordered by a relation $<$ and a function $f : X \rightarrow O$ s.t. $x \prec y$ iff $f(x) < f(y)$ for all $x, y \in X$.

(5) A preferential structure $\mathcal{Z} = \langle X, \prec \rangle$ s.t. all $x \in X$ are classical models for some (propositional) language \mathcal{L} is called a classical preferential structure. It defines a (in general nonmonotonic) logic $\models_{\mathcal{Z}}$ for \mathcal{L} by $T \models_{\mathcal{Z}} \phi$ iff $\mu(M(T)) \subseteq M(\phi)$.

(6) A structure $\langle X, \prec \rangle$ is called transitive etc. iff \prec is transitive etc.

Note:

We will work on finite sets here, and our relations will be transitive, so the structures will be smooth (see e.g. [KLM90]). Moreover, the relations will be acyclic.

We now turn to defeasible inheritance diagrams. For more details and motivation the reader is referred e.g. to [HTT87], [Sch97], [SL89], [THT87].

Definition 1.2

(1) A defeasible inheritance diagram is a finite directed acyclic graph with two types of arrows or links, \rightarrow and $\not\rightarrow$, and no parallel arrows, i.e. there is at most one arrow from any X to any Y . Such diagrams will be denoted Γ etc.

(2) \mapsto will stand for \rightarrow or $\not\rightarrow$.

Let now Γ be fixed, subsequent definitions are relative to Γ .

(3) A (generalized) path in Γ is a sequence of concatenated arrows pointing the same way. So $X \rightarrow Y \not\rightarrow Z$ is a generalized path, neither $X \rightarrow Y \leftarrow Z$ nor $X \rightarrow Y Z \rightarrow U$ is one.

(4) A positive (generalized) path ends with \rightarrow , a negative one with $\not\rightarrow$.

- (5) An arrow is called an atomic path, other paths are called composite.
- (6) Potential paths in Γ are defined inductively:
- An arrow is a potential path.
 - If $X \dots \rightarrow Y$ is a positive potential path, then $X \dots \rightarrow Y \mapsto Z$ is a potential path.
- Thus $X \rightarrow Y \not\rightarrow Z$ is a potential path, $X \not\rightarrow Y \rightarrow Z$ is not. This is intuitively justified, from the latter we cannot conclude anything about X and Z .
- (7) Two paths $\sigma : X \dots \mapsto Y$, $\sigma' : X \dots \mapsto Y$ are called contradictory iff one is positive, and the other negative.
- (8) A subpath of a path σ with the same beginning as σ is called an initial segment of σ . Thus, $X \rightarrow Y$ and $X \rightarrow Y \rightarrow Z$ are initial segments of $X \rightarrow Y \rightarrow Z \rightarrow U$.

The main problem of defeasible inheritance diagrams is to define in a coherent and intuitively appealing way the set of “valid” paths and their results. If a path $X \dots \rightarrow Y$ ($X \dots \not\rightarrow Y$) is accepted as valid, we say the result “most X 's are Y 's” (“most X 's are not Y 's”) is valid in the diagram. There is a multitude of definitions of validity of paths, with sometimes very subtle differences.

We will now present some of the basic ideas and alternatives.
Let again a diagram Γ be given.

(1) Concatenation.

All potential paths are candidates for valid paths.

(2) The concept of preclusion (= preemption).

Some paths are considered “stronger” than others, and if a stronger path contradicts a weaker one, the first one might be accepted as valid, but not the second. Example: In the diagram $X \rightarrow Y \rightarrow Z$, $X \not\rightarrow Z$, the direct arrow $X \not\rightarrow Z$ is considered stronger than the composite one $X \rightarrow Y \rightarrow Z$, the latter will not be valid (and the former will, all arrows will be valid paths).

(3) Extension based versus directly sceptical approach.

There are, however, conflicts which cannot be resolved this way. Example: In the diagram $X \rightarrow Y \rightarrow Z$, $X \rightarrow U \not\rightarrow Z$, both paths from X to Z have equal strength, but they contradict each other. The extension based approach makes two extensions, in one extension $X \rightarrow Y \rightarrow Z$ is valid, in the other one $X \rightarrow U \not\rightarrow Z$ is valid. Usually, one will then take the intersection of extensions, formed either over the set of valid paths, or of accepted results (“most X 's are Z 's” in the first one, “most X 's are not Z 's” in the second one etc.). A directly sceptical approach will not accept any of $X \rightarrow Y \rightarrow Z$, $X \rightarrow U \not\rightarrow Z$ as valid, as they are in unresolvable conflict (and neither the result “most X 's are Z 's” nor “most X 's are not Z 's”).

(4) Upward versus downward chaining.

Upward chaining formalisms require that initial segments of valid paths be valid, downward chaining formalisms that terminal segments of valid paths be valid. For example, in the diagram $A \rightarrow B \rightarrow C \rightarrow D$, if the path $A \rightarrow B \rightarrow C \rightarrow D$ is to be valid, the subpath $A \rightarrow B \rightarrow C$ ($B \rightarrow C \rightarrow D$) has to be valid in upward (downward) chaining formalisms. This leads naturally to an inductive definition of validity.

(5) On-path versus off-path preclusion.

A path $\sigma : X \dots \rightarrow Y \dots \rightarrow Z$ and an arrow $Y \not\rightarrow U$ is an off-path preclusion of the path $\tau : X \dots \rightarrow Z \dots \rightarrow U$, but an on-path preclusion only iff all nodes on τ between X and Z are on σ . A preclusion will be called valid if the precluding path σ is valid.

Example: In the classical Tweety diagram $Tweety \rightarrow birds \rightarrow flying\text{-things}$, $Tweety \rightarrow penguins \not\rightarrow flying\text{-things}$, $penguins \rightarrow birds$, $\sigma : Tweety \rightarrow penguins$, $penguins \not\rightarrow flying\text{-things}$ is an on-path preclusion of $Tweety \rightarrow penguins \rightarrow birds \rightarrow flying\text{-things}$, but $\sigma' : Tweety \rightarrow penguins \rightarrow birds$, $penguins \not\rightarrow flying\text{-things}$ is only an off-path preclusion of $Tweety \rightarrow birds \rightarrow flying\text{-things}$. (The node *Tweety* stands here for an element, not a set, but this is no serious problem.)

Our formal definition of valid paths will be given below in Definition 3.2.

2 THE FORMALISM OF REFERENCE CLASSES

Recall that our formal approach is semantical, we work either with the preferential semantics or the abstract partial order semantics of nonmonotonic logics - and abstract even further by working with arbitrary sets. This liberates us immediately from all considerations about equivalence of formulas, and, in general, we consider the semantical approach the better one. Recall further that we take two nonmonotonic databases, transform them into semantical form (as preferential structures or partial orders), compute the product structure, and transform it back into the resulting joint nonmonotonic database. This section deals with the computation of the product structure. Some basic and natural formal properties are developed in Proposition 2.8.

2.1 Reference classes and preferential structures

We formalize now the importation of information from a reference class in the preferential structure approach. For illustration, we first treat a simple example.

Let $v(\mathcal{L}_b) := \{p\}$, $v(\mathcal{L}_r) := \{q\}$, and the defeasible base theory be given by the preferential structure $m' \models p \prec m \models \neg p$, the reference class theory by $n' \models q \prec n \models \neg q$. The joint language is then defined by $\{p, q\}$, and we have to define a preferential structure on the four models of p and q . It seems clear that $m_0 \models p \wedge q \prec m_1 \models \neg p \wedge q$ and that $m_2 \models p \wedge \neg q \prec m_3 \models \neg p \wedge \neg q$, as by the base information p is preferred to $\neg p$. Likewise, we should

have $m_0 \models p \wedge q \prec m_2 \models p \wedge \neg q$ and $m_1 \models \neg p \wedge q \prec m_3 \models \neg p \wedge \neg q$, as the reference class information prefers q to $\neg q$. But, we will also have $m_2 \models p \wedge \neg q \prec m_1 \models \neg p \wedge q$, even though the reference class prefers q to $\neg q$, as the information (and preferences) of the base class - which prefers p to $\neg p$ - are considered stronger.

This leads to the following basic definition:

Definition 2.1

Let \mathcal{L}_b and \mathcal{L}_r be two disjoint (propositional) languages. A model m of the joint language can be uniquely decomposed into m_b and m_r , the parts of the base and the reference class language. Let further \prec_b and \prec_r be preferential relations on the base and reference class models. We then define a preference relation on the models of the joint language by $m \prec m'$ iff $m_b \prec_b m'_b$ or $(m_b = m'_b \text{ and } m_r \prec_r m'_r)$.

In the abstract algebraic setting of Section 2.2, we will use the following notation:

If \prec_X is a binary relation on X , \prec_Y a binary relation on Y , we define $\prec_{X \times Y}$ on $X \times Y$ by $(x, y) \prec_{X \times Y} (x', y')$ iff $x \prec_X x'$ or $(x = x' \text{ and } y \prec_Y y')$.

Note that our definition encodes priority of the base class.

We generalize this definition to the case with multiple reference classes:

Definition 2.2

Let \mathcal{L}_b and \mathcal{L}_{r_i} be pairwise disjoint (propositional) languages, and m_b (m_{r_i}) the components of a model m of the joint language as above, let \prec_b (\prec_{r_i}) be preferential relations on the base and reference class models. We then define a preference relation on the models of the joint language by

$m \prec m'$ iff $m_b \prec_b m'_b$ or $(m_b = m'_b \text{ and there is } i \text{ s.t. } m_{r_i} \prec_{r_i} m'_{r_i}, \text{ and for all } j \text{ } m_{r_j} \prec_{r_j} m'_{r_j} \text{ or } m_{r_j} = m'_{r_j})$.

We note the following properties:

Fact 2.1

If the relations \prec_b and \prec_{r_i} are irreflexive (transitive, acyclic), then so is \prec as defined above. \square

Rankedness is not preserved, however, as the following example shows:

Example 2.1

Let \mathcal{L}_b be defined by p, q , \mathcal{L}_r by r , \prec_b be the transitive closure of $p \wedge q \prec_b p \wedge \neg q \prec_b \neg p \wedge \neg q$, $p \wedge q \prec_b \neg p \wedge q \prec_b \neg p \wedge \neg q$, and \prec_r by $r \prec_r \neg r$ (both structures are obviously ranked), then the resulting \prec is the transitive closure of $p \wedge q \wedge r \prec p \wedge q \wedge \neg r \prec p \wedge \neg q \wedge r \prec p \wedge \neg q \wedge \neg r \prec \neg p \wedge \neg q \wedge r \prec \neg p \wedge \neg q \wedge \neg r$ and $p \wedge q \wedge \neg r \prec \neg p \wedge q \wedge r \prec \neg p \wedge q \wedge \neg r \prec \neg p \wedge \neg q \wedge r$. So $p \wedge \neg q \wedge r$ and $\neg p \wedge q \wedge r$ are incomparable, $p \wedge \neg q \wedge r \prec p \wedge \neg q \wedge \neg r$, but not $\neg p \wedge q \wedge r \prec p \wedge \neg q \wedge \neg r$.

Definition 2.3

Let $A \subseteq X \times Y$.

- (1) $A^X := \{x \in X : \exists y \in Y. (x, y) \in A\}$,
- (2) $A^Y := \{y \in Y : \exists x \in X. (x, y) \in A\}$,
- (3) for $y \in Y$, $A^X(y) := \{x \in X : (x, y) \in A\}$,
- (4) for $x \in X$, $A^Y(x) := \{y \in Y : (x, y) \in A\}$.

Fact 2.2

- (1) $A \subseteq B \rightarrow A^X \subseteq B^X$,
 - (2) $A \subseteq B \rightarrow A^X(y) \subseteq B^X(y)$,
 - (3) $(A \cup B)^X = A^X \cup B^X$,
 - (4) $(A \cup B)^X(y) = A^X(y) \cup B^X(y)$,
 - (5) $(A \cap B)^X(y) = A^X(y) \cap B^X(y)$,
 - (6) $(A - B)^X(y) = A^X(y) - B^X(y)$.
- (likewise, of course, for A^Y etc.) \square

Fact 2.3

Let $A \subseteq X \times Y$, $\prec_{X \times Y}$ be defined from \prec_X , \prec_Y as in Definition 2.1. Then $\mu_{X \times Y}(A) = \{(x, y) : x \in \mu_X(A^X), y \in \mu_Y(A^Y(x))\}$ - where μ_X stands for μ_{\prec_X, \prec_X} etc.

Proof:

Let $B := \{(x, y) : x \in \mu(A^X), y \in \mu(A^Y(x))\}$.

" \subseteq ": Suppose $(x, y) \notin B$, we show $(x, y) \notin \mu(A)$. Case 1: $x \notin \mu(A^X)$. If $x \notin A^X$, then $(x, y) \notin A$, so $(x, y) \notin \mu(A)$. Suppose $x \in A^X$, but $x \notin \mu(A^X)$, so there is $x' \in A^X$, $x' \prec x$. Thus there is y' s.t. $(x', y') \in A$, but then $(x', y') \prec (x, y)$. Case 2: $y \notin \mu(A^Y(x))$. If $y \notin A^Y(x)$, then $(x, y) \notin A$. If $y \in A^Y(x)$, but $y \notin \mu(A^Y(x))$, then there is $y' \prec y$, $y' \in A^Y(x)$, so $(x, y') \in A$ and $(x, y') \prec (x, y)$.

" \supseteq ": Let $(x, y) \in B$. (1) $(x, y) \in A : y \in \mu(A^Y(x)) \subseteq A^Y(x) \rightarrow (x, y) \in A$. (2) Suppose $(x, y) \in A$, but there is $(x', y') \in A$, $(x', y') \prec (x, y)$. Consider the cases of Definition 2.3. Case 1: $x' \prec x$. Then $x \notin \mu(A^X)$. Case 2: $x = x'$, but $y' \prec y$. Then $y \notin \mu(A^Y(x))$. \square

Fact 2.4

If $A = X' \times Y'$, then $\mu(A) = \mu(X') \times \mu(Y')$. \square

2.2 Reference classes and orders between formulas

Recall that we work here in a purely algebraic setting over some finite sets X, Y, Z , which are to be understood as the sets of models of some languages $\mathcal{L}_X, \mathcal{L}_Y, \mathcal{L}_Z$. The relations \prec will be transitive and acyclic.

Definition 2.4

Define from a relation \prec on Z a relation $<_{\prec}$ on $\mathcal{P}(Z)$ by $A <_{\prec} B :\Leftrightarrow B \neq \emptyset$ and $\mu(A \cup B) \cap A = \emptyset$.

Fact 2.5

$<_{\prec}$ as defined in Definition 2.4 is equivalent to the (model-variant of the) definition $\alpha <' \beta :\Leftrightarrow \text{Con}(\beta)$ and $\alpha \vee \beta \models_Z \neg\alpha$. \square

Condition 2.1

We recall the usual filter (or ideal) and coherence properties:

- (1) $\emptyset < A$ if $A \neq \emptyset$,
- (2) $A, B < C \rightarrow A \cup B < C$,
- (3) $A \subseteq B < C \subseteq D \rightarrow A < D$,
- (4) $A, B < C \rightarrow A < C - B$,
- (5) $A < B \rightarrow B \neq \emptyset$.

Fact 2.6

$<_{\prec}$ as defined in Definition 2.4 satisfies the properties of Condition 2.1.

Proof:

- (1) Trivial.
- (2) We show $\mu(A \cup B \cup C) \cap (A \cup B) = \emptyset$. Suppose there is $x \in (A \cup B)$ s.t. there is no $x' \prec x, x' \in A \cup B \cup C$, but this contradicts the hypotheses $\mu(A \cup C) \cap A = \emptyset$ and $\mu(B \cup C) \cap B = \emptyset$.
- (3) We show $\mu(A \cup D) \cap A = \emptyset$. But if $x \in A \subseteq B$, then by $\mu(B \cup C) \cap B = \emptyset$ there is (by finiteness of Z and transitivity of \prec) $x'' \in \mu(B \cup C) \subseteq C \subseteq D, x'' \prec x$, so $x \notin \mu(A \cup D)$.
- (4) We first show $C - B \neq \emptyset$. As $B < C, C \neq \emptyset$, and $\mu(B \cup C) \neq \emptyset$. But $\mu(B \cup C) \subseteq C - B$. It remains to show $\mu((C - B) \cup A) \cap A = \emptyset$. Let $x \in A$. By $\mu(C \cup A) \cap A = \emptyset$, there is $x' \in C, x' \prec x$. If $x' \notin B$, we are done, suppose $x' \in B$. As $\mu(B \cup C) \cap B = \emptyset$, there must be $x'' \in C - B, x'' \prec x'$. So $x'' \prec x$.
- (5) Trivial by definition. \square

Fact 2.7

If $<$ on $\mathcal{P}(Z)$ satisfies the properties (2) – (4) of Condition 2.1, then the following hold:

- (1) If $A, B \subseteq C, C - A < A, C - B < B$, then $C - (A \cap B) < A \cap B$,
- (2) If $A < C - A, B < C - B$, then $A \cup B < C - (A \cup B)$.

Proof:

- (1) $C - A < A \rightarrow$ (by property (3)) $C - A < C$, likewise $C - B < C$, so by (2) $C - (A \cap B) = ((C - A) \cup (C - B)) < C$, so by (4) $C - (A \cap B) < C - (C - (A \cap B)) = A \cap B$.
(2) $A < C - A \subseteq C$, $B < C - B \subseteq C$, so by (2) $A \cup B < C$, so by (4) $A \cup B < C - (A \cup B)$. \square

The following definition seems at first sight somewhat complicated, but we do not see a simpler one which gives the same Proposition 2.8. Moreover, intuitively it seems to be the right one. Roughly, $A < B$ iff there is $W \subseteq B - A$ which is a “big” subset of $A \cup B$ in both coordinates.

Definition 2.5

Let $<_X$ be a binary relation on $\mathcal{P}(X)$, $<_Y$ a binary relation on $\mathcal{P}(Y)$. We then define a binary relation $<_{X \times Y}$ on $X \times Y$ by
 $A < B$ iff there is $W \subseteq B - A$ s.t.

$$(A \cup B)^X - W^X <_X W^X \text{ and for all } x \in W^X. (A \cup B)^Y(x) - W^Y(x) <_Y W^Y(x).$$

We call such W a witness for $A < B$.

Proposition 2.8

- (1) If $<_X$ and $<_Y$ satisfy the properties of Condition 2.1, then so does $<_{X \times Y}$.
(2) If $<_{\prec_X}$ is generated from \prec_X , $<_{\prec_Y}$ from \prec_Y as in Definition 2.4, $\prec_{X \times Y}$ from \prec_X and \prec_Y as in Definition 2.1, then $<_{X \times Y}$ as defined in Definition 2.5 from $<_{\prec_X}$ and $<_{\prec_Y}$ is the relation $<_{\prec_{X \times Y}}$ on $\mathcal{P}(X \times Y)$ generated by $\prec_{X \times Y}$ as in Definition 2.4 (commutativity of definitions).
(3) If $A \cup B = X' \times Y'$ and $B \neq \emptyset$, then $A <_{X \times Y} B$ iff $(\mu(X') \times \mu(Y')) \cap A = \emptyset$.

Proof:

(1) The proof is surprisingly tedious - but perhaps there is a much nicer one we just did not see!

(1.1) Let $A \neq \emptyset$, we have to show $\emptyset < A$. $W := A$ witnesses $\emptyset < A$.

(1.2) Let $A < C$, $B < C$, we have to show $A \cup B < C$. Let W_A witness $A < C$, W_B witness $B < C$. Define $W_{A \cup B} := \{(x, y) : x \in W_A^X \cap W_B^X, (x, y) \in C - (A \cup B)\}$. $W_{A \cup B}$ witnesses $A \cup B < C$:

Note $W_{A \cup B} \subseteq C - (A \cup B)$.

We first show $\forall x \in W_A^X \cap W_B^X. W_{A \cup B}^Y(x) > (A \cup B \cup C)^Y(x) - W_{A \cup B}^Y(x)$. By $W_A \subseteq C - A$, we have $C^Y(x) - A^Y(x) = (C - A)^Y(x) \supseteq W_A^Y(x) > (A \cup C)^Y(x) - W_A^Y(x) \supseteq (A \cup C)^Y(x) - (C - A)^Y(x) = A^Y(x)$, likewise $C^Y(x) - B^Y(x) > B^Y(x)$. So by Fact 2.7, (2) $W_{A \cup B}^Y(x) = C^Y(x) - (A \cup B)^Y(x) > (A \cup B)^Y(x) = (A \cup B \cup C)^Y(x) - W_{A \cup B}^Y(x)$.

Consequently, by property (5), $\forall x \in W_A^X \cap W_B^X. W_{A \cup B}^Y(x) \neq \emptyset$, so $W_{A \cup B}^X = W_A^X \cap W_B^X$.

We now show $W_{A \cup B}^X > (A \cup B \cup C)^X - W_{A \cup B}^X$. By prerequisite, $W_B^X > (C \cup B)^X - W_B^X$. Note that $W_B \subseteq C$, so $C^X \supseteq W_B^X > (C \cup B)^X - W_B^X = (C^X \cup B^X) - W_B^X \supseteq B^X - C^X \supseteq B^X - (C^X \cup A^X)$, so $C^X > B^X - (C^X \cup A^X)$. Likewise, $C^X \supseteq W_A^X > (A^X \cup C^X) - W_A^X \supseteq C^X -$

W_A^X , so $C^X > C^X - W_A^X$. So by property (4) of Condition 2.1 $W_A^X = C^X - (C^X - W_A^X) > B^X - (C^X \cup A^X)$. So by $W_A^X > (A^X \cup C^X) - W_A^X$, $W_A^X > B^X - (C^X \cup A^X)$, and by property (2) of Condition 2.1, $W_A^X > ((A^X \cup C^X) - W_A^X) \cup (B^X - (C^X \cup A^X)) = (A^X \cup B^X \cup C^X) - W_A^X$. For the latter equality: " \subseteq ": $((A^X \cup C^X) - W_A^X) \subseteq (A^X \cup B^X \cup C^X) - W_A^X$, and $B^X - (C^X \cup A^X) \subseteq (A^X \cup B^X \cup C^X) - W_A^X$ by $W_A \subseteq C$. " \supseteq ": If $x \in (A^X \cup C^X) - W_A^X$, this is ok. If $x \in B^X - (A^X \cup C^X \cup W_A^X) = B^X - (A^X \cup C^X)$, this is ok, too. Thus, $W_A^X > (A \cup B \cup C)^X - W_A^X$. Likewise, $W_B^X > (A \cup B \cup C)^X - W_B^X$, so by Fact 2.7, (1) $W_A^X \cap W_B^X > (A \cup B \cup C)^X - (W_A^X \cap W_B^X)$.

(1.3) We first show $A \subseteq B < C \rightarrow A < C$. Let W_B witness $B < C$. Then $W_B \subseteq C - B$, so $W_B \subseteq C - A$. By property (3), the other properties of W_B hold to witness $A < C$.

We now show $A < B \subseteq C \rightarrow A < C$. Let W_A witness $A < B$, and define $W := \{(x, y) : x \in W_A^X, (x, y) \in W_A \cup (C - (A \cup B))\} \cup \{(x, y) \in C : x \in C^X - (A \cup B)^X\}$. Thus, $W^X = W_A^X \cup (C^X - (A \cup B)^X)$ and $W \subseteq C - A$. We now show $(A \cup C)^X - W^X = (A \cup B)^X - W_A^X$, and conclude that $W^X > (A \cup C)^X - W^X : (C^X - (A \cup B)^X) \cup (A \cup B)^X = C^X \cup A^X \cup B^X = C^X \cup A^X = (A \cup C)^X$, so $(A \cup C)^X - W^X = ((C^X - (A \cup B)^X) \cup (A \cup B)^X) - (W_A^X \cup (C^X - (A \cup B)^X)) = (A \cup B)^X - W_A^X$, as $(A \cup B)^X \cap (C^X - (A \cup B)^X) = \emptyset$. But now $W^X \supseteq W_A^X > (A \cup B)^X - W_A^X = (A \cup C)^X - W^X$. It remains to show that $W^Y(x) > (A \cup C)^Y - W^Y(x)$ for $x \in W^X$. Case 1: $x \in C^X - (A \cup B)^X$, then $(A \cup C)^Y(x) = C^Y(x)$, so $(A \cup C)^Y(x) - W^Y(x) = C^Y(x) - C^Y(x) = \emptyset < C^Y(x) = W^Y(x)$ (as $x \in C^X$, $C^Y(x) \neq \emptyset$). Case 2: $x \in W_A^X$. By prerequisite, $W_A^Y(x) > (A \cup B)^Y(x) - W_A^Y(x)$. $W^Y(x) = W_A^Y(x) \cup (C - (A \cup B))^Y(x) \supseteq W_A^Y(x) > (A \cup B)^Y(x) - W_A^Y(x) =$ (as $(A \cup B)^Y(x) \cap (C - (A \cup B))^Y(x) = \emptyset$) $(A \cup B)^Y(x) - (W_A^Y(x) \cup (C - (A \cup B))^Y(x)) =$ (as $(A \cup B) - (C - (A \cup B)) = (A \cup C) - (C - (A \cup B))$) $(A \cup C)^Y(x) - (W_A^Y(x) \cup (C - (A \cup B))^Y(x)) = (A \cup C)^Y(x) - W^Y(x)$.

(1.4) We have to show $A, B < C \rightarrow A < C - B$. Choose $W_{A \cup B}$ as in the construction for (1.2). Then $W_{A \cup B} \subseteq (C - B) - A$. As shown above, $W_{A \cup B}^X > (A \cup B \cup C)^X - W_{A \cup B}^X \supseteq (A \cup (C - B))^X - W_{A \cup B}^X$. Likewise for $x \in W_{A \cup B}^X$ $W_{A \cup B}^Y(x) > (A \cup B \cup C)^Y(x) - W_{A \cup B}^Y(x) \supseteq (A \cup (C - B))^Y(x) - W_{A \cup B}^Y(x)$.

(1.5) If W witnesses $A < B$, then $W \subseteq B - A$, $W^X > (A \cup B)^X - W^X$, so $W^X \neq \emptyset$, so $B \supseteq W \neq \emptyset$.

(2) We have to show for $A, B \subseteq X \times Y$ $A <_{X \times Y} B$ iff $A <_{X \times Y} B$. Now, $A <_{X \times Y} B$ iff $B \neq \emptyset$ and $\mu_{X \times Y}(A \cup B) \cap A = \emptyset$, where by Fact 2.3 $\mu_{X \times Y}(A \cup B) = \{(x, y) : x \in \mu_X((A \cup B)^X), y \in \mu_Y((A \cup B)^Y(x))\}$. Moreover, $A <_{X \times Y} B$ iff there is $W \subseteq B - A$ s.t. $(A \cup B)^X - W^X <_{X \times Y} W^X$ and for all $x \in W^X$ $(A \cup B)^Y(x) - W^Y(x) <_{Y \times Y} W^Y(x)$.

Note that by $B \neq \emptyset$, $A \cup B \neq \emptyset$, $\mu(A \cup B) \neq \emptyset$ etc.

" \rightarrow ": Let $W := \mu_{X \times Y}(A \cup B)$. Then $W \subseteq B - A$, and $W = \{(x, y) : x \in \mu_X((A \cup B)^X), y \in \mu_Y((A \cup B)^Y(x))\}$, so $W^X = \mu_X((A \cup B)^X)$, and for $x \in W^X$ $W^Y(x) = \mu_Y((A \cup B)^Y(x))$. Thus by $W^X \neq \emptyset$, $W^X >_{X \times Y} (A \cup B)^X - W^X$. Moreover, if $x \in W^X$, then $W^Y(x) \neq \emptyset$, and thus $W^Y(x) >_{Y \times Y} (A \cup B)^Y(x) - W^Y(x)$. So W witnesses $A <_{X \times Y} B$.

" \leftarrow ": Let $A <_{X \times Y} B$, we have to show $B \neq \emptyset$ and $\mu_{X \times Y}(A \cup B) \cap A = \emptyset$. By Fact 2.6, $<_{X \times Y}$ and $<_{Y \times Y}$ satisfy the properties of Condition 2.1, thus so does $<_{X \times Y}$ (by (1)),

so if $A <_{X \times Y} B$, then $B \neq \emptyset$. Let now W witness $A <_{X \times Y} B$. Thus $W \subseteq B - A$ and $(A \cup B)^X - W^X <_{\prec_X} W^X$, so $\mu_X((A \cup B)^X) \subseteq W^X$. Likewise, for all $x \in W^X$ $(A \cup B)^Y(x) - W^Y(x) <_{\prec_Y} W^Y(x)$, so $\mu_Y((A \cup B)^Y(x)) \subseteq W^Y(x)$. By Fact 2.3, $\mu_{X \times Y}(A \cup B) = \{(x, y) : x \in \mu_X((A \cup B)^X), y \in \mu_Y((A \cup B)^Y(x))\}$, so $\mu_{X \times Y}(A \cup B) \subseteq W$. Thus by $W \cap A = \emptyset$, $\mu_{X \times Y}(A \cup B) \cap A = \emptyset$.

(3) Trivial by Fact 2.4. \square

3 REFERENCE CLASSES AND DEFEASIBLE INHERITANCE

As said above, one of the problems with reference classes is how to find them. Recall that we will say in an inheritance diagram that Y is a reference class for X iff there is a good argument (in the diagram) that X 's behave like Y 's, or, that “most X 's are Y 's”, or, that there is valid positive path (argument) from X to Y . (A negative valid path gives no direct information about reference classes.)

Direct versus indirect inheritance, direct versus indirect information

In the simple diagram $A \rightarrow B \rightarrow C \rightarrow D$ (in which all potential paths will be valid in all inheritance diagram formalisms), e.g. A has the direct information from the direct arrow $A \rightarrow B$ that A 's are B 's, and the indirect information that A 's are C 's (from the composite valid path $A \rightarrow B \rightarrow C$). On the other hand, A inherits directly from B via the direct arrow $A \rightarrow B$ that B 's are C 's, and indirectly from C via the valid path $A \rightarrow B \rightarrow C$ that C 's are D 's.

Direct inheritance of direct information is too weak to account for such diagrams, as we cannot conclude this way that A 's are D 's; with this reasoning, we cannot go beyond paths of length 2.

Add now to the above diagram the path $A \rightarrow E \not\rightarrow C$. If we accept the idea that reference classes for X are those to which there is a valid positive path, in a directly sceptical framework, B and E are the only reference classes for A . On the other hand, C is a reference class for B . Thus, B has the indirect information that B 's are D 's, or, that things are normally in D . So, if we inherit indirect information, A will inherit from B that things are normally D 's and thus conclude that A 's are D 's too, which is not acceptable. Moreover, if we accept to inherit indirect information, the base class has no control where the information comes from, which seems contradictory to the spirit of inheritance networks.

Thus, the only correct way seems to have indirect inheritance of direct information. This leads naturally to upward chaining, as discussed at the end of this Section.

Valid paths

The main rule of preference is that direct links have priority over composite paths, as a direct link gives more reliable information than a composite path. Warning: The word “preference” is used here in colloquial style, and not in the sense of preferential structures. We now give an auxiliary definition, followed by the formal definition of validity. Again, we fix a diagram Γ , definitions will be relative to Γ .

Definition 3.1

$deg(X, Y)$ is the maximal length of all generalized paths from X to Y in Γ .

Fix now an arbitrary enumeration ρ_X of the nodes $\neq X$ in Γ to which there is a generalized path from X , and which respects deg , i.e. $deg(X, Y) < deg(X, Z) \rightarrow \rho_X(Y) < \rho_X(Z)$.

Definition 3.2

(Inductive definition of valid paths in Γ .) Let Y be the ρ_X – *first* node not yet treated. Let Φ_{XY}^0 be the set of potential paths from X to Y . Let Φ_{XY}^1 be the set of paths in Φ_{XY}^0 from X to Y which are direct arrows or whose initial segments are valid (this has already been decided by induction, as ρ_X respects deg). Let Φ_{XY}^2 be the set of paths in Φ_{XY}^1 from X to Y for which there is no valid on-path preclusion in Γ (again, this has already been decided by induction). Let Φ_{XY} be the set of paths in Φ_{XY}^2 from X to Y for which there is no contradictory path in Φ_{XY}^2 . Φ_{XY} is the set of valid paths from X to Y .

We note:

Our approach is directly sceptical (by the final step of the definition), and upward chaining (but in general not downward chaining) by the definition of Φ_{XY}^1 . An example which shows that downward chaining need not hold, is: $A \rightarrow B \rightarrow C \rightarrow E$, $B \rightarrow D \not\rightarrow E$, $A \not\rightarrow D$. $A \rightarrow B \rightarrow C \rightarrow E$ is valid, $B \rightarrow C \rightarrow E$ is not.

We do not necessarily consider more specific information to be more reliable - our only preference is for direct links over composite paths. That’s why we did not adopt off-path preclusion.

We now define formally:

Definition 3.3

Y is a reference class for X in the diagram Γ , iff there is a valid positive path from X to Y in Γ .

We have not yet specified which information shall be inherited from the reference classes of a given class X . To see the problem, consider the following diagram: $A \rightarrow B \rightarrow C \rightarrow E$, $B \not\rightarrow D$, $C \rightarrow D$. The defeasible information of node B , “most things are in C , but not in D ”, in partial order representation $C \wedge \neg D > C \wedge D > \neg C \wedge D$, $C \wedge \neg D > \neg C \wedge \neg D >$

$\neg C \wedge D$, that of node C “most things are in D and E ”, or $D \wedge E > D \wedge \neg E > \neg D \wedge \neg E$, $D \wedge E > \neg D \wedge E > \neg D \wedge \neg E$. So B prefers $\neg D$, and C prefers D , obviously, we want to inherit the information that $\neg D$ is preferred.

Thus, to solve the problem which information will be inherited from a reference class Y , we will inherit the information which corresponds to a valid path from X via Y to the node which corresponds to that information. In our example, there is a valid path $A \rightarrow B \not\rightarrow D$, but no valid path $A \rightarrow B \rightarrow C \rightarrow D$, and the resulting languages can be chosen disjoint.

It is now evident that we have to take an upward chaining approach. We first need a valid path $\sigma : X \dots \rightarrow Y$ to the reference class, which is contained in a valid path $\sigma' : X \dots \rightarrow Y \rightarrow Z$ to the node corresponding to the information. The fact that we showed that Y is a reference class for X (via validity of σ) justified the result that X 's behave like Y 's.

4 CONCLUSION

We have clarified the use of reference classes (in our sense) in a defeasible setting, examined two formal definitions of this use, one in the framework of preferential structures, the other in the framework of orders defined by a nonmonotonic database. We have shown that both approaches are equivalent, in the sense that the preferential approach is the same as the one which first transforms the preference relations into orders, and then operates on the orders.

We have, we think, clarified the concepts of defeasible inheritance by analyzing such diagrams using the notion of reference class, which led us to adopt an upward chaining, on-path preclusion approach. The simple setting of inheritance diagrams permits a clear choice of the reference classes and the information inherited from such classes.

Our approach to the use of reference classes seems broad enough to treat other situations than simple inheritance diagrams, for instance those where other defeasible information, not only class-inclusion, is appended to an inheritance diagram, similar to KL-ONE situations.

We have not addressed the first order problem, so this is a subject of further research. We did not investigate either whether our formalism - though different - can help to clarify the use of reference classes in philosophy of science.

5 ACKNOWLEDGEMENTS

Three anonymous referees have helped to make the paper much more readable (we hope).

References

- [BB94] Shai Ben-David, R.Ben-Eliyahu, “A modal logic for subjective default reasoning”, Proceedings LICS-94, 1994
- [FH98] N.Friedman, J.Halpern, “Plausibility Measures and Default Reasoning”, IBM Almaden Research Center Tech.Rept. 1995, to appear in Journal of the ACM
- [HTT87] J.F.Horty, D.S.Touretzky, R.H.Thomason: “A sceptical theory of inheritance in nonmonotonic semantic networks”, Proceedings AAAI-87, pp. 358-363
- [KLM90] S.Kraus, D.Lehmann, M.Magidor, “Nonmonotonic reasoning, preferential models and cumulative logics”, Artificial Intelligence, 44 (1-2), p.167-207, July 1990
- [LM92] D.Lehmann, M.Magidor, “What does a conditional knowledge base entail?”, Artificial Intelligence, 55(1), p. 1-60, May 1992
- [Kyb74] H.E.Kyburg: “The logical foundations of statistical inference”, Reidel, 1974
- [Kyb83] H.E.Kyburg: “The reference class”, Philos. Sci. 50, 1983, pp. 374-397
- [Pol90] J.L.Pollock: “Nomic probabilities and the foundations of induction”, Oxford Univ. Press, 1990
- [Rei49] H.Reichenbach: “Theory of probability”, Univ. of California Press, Berkeley, 1949
- [Sch92] K.Schlechta: “Some Results on Classical Preferential Models”, Journal of Logic and Computation, Oxford, Vol.2, No.6 (1992), p. 675-686
- [Sch97] K.Schlechta: “Nonmonotonic logics: Basic concepts, results, and techniques”, Springer, 1997
- [Sch97-4] K.Schlechta: “Filters and Partial Orders”, Journal of the Interest Group in Pure and Applied Logics, Vol. 5, No. 5, p. 753-772, 1997
- [Sho87] Yoav Shoham: “A semantical approach to nonmonotonic logics”. In Proc. Logics in Computer Science, p.275-279, Ithaca, N.Y., 1987
- [SL89] B.Selman, H.Levesque: “The tractability of path-based inheritance”, Proceedings IJCAI 1989, pp. 1140-1145
- [THT87] D.S.Touretzky, J.F.Horty, R.H.Thomason: “A clash of intuition: The current state of nonmonotonic multiple inheritance systems”, Proceedings IJCAI 1987, pp. 476-482