Expressing Preferences in Default Logic

James P. Delgrande
School of Computing Science
Simon Fraser University
Burnaby, B.C.
Canada V5A 1S6
jim@cs.sfu.ca

Torsten Schaub*
Institut für Informatik
Universität Potsdam
Postfach 60 15 53, D–14415 Potsdam
Germany
torsten@cs.uni-potsdam.de

March 8, 2001

Abstract

We address the problem of reasoning about preferences among properties (outcomes, desiderata, etc.) in Reiter's default logic. Preferences are expressed using an *ordered default theory*, consisting of default rules, world knowledge, and an ordering, reflecting preference, on the default rules. In contrast with previous work in the area, we do not rely on prioritised versions of default logic, but rather we transform an ordered default theory into a second, standard default theory wherein the preferences are respected, in that defaults are applied in the prescribed order. This translation is accomplished via the naming of defaults, so that reference may be made to a default rule from within a theory. In an elaboration of the approach, we allow an ordered default theory where preference information is specified within a default theory. Here one may specify preferences that hold by default, in a particular context, or give preferences among preferences. In the approach, one essentially axiomatises how different orderings interact within a theory and need not rely on metatheoretic characterisations. As well, we can immediately use existing default logic theorem provers for an implementation. From a theoretical point of view, this shows that the explicit representation of priorities among defaults adds nothing to the overall expressibility of default logic.

Keywords: Nonmonotonic reasoning, default logic, knowledge representation, preference handling

* Affiliated with Simon Fraser University, Burnaby, Canada.

1 Introduction

The notion of *preference* or *priority* in commonsense reasoning is pervasive. For example, in scheduling not all deadlines may be simultaneously satisfiable, and in configuration various goals may not be simultaneously met. Preferences among deadlines and goals may allow for an acceptable, non-optimal, solution. In decision making, preferences clearly play a major role. In buying a car for example, one may have various criteria in mind (inexpensive, safe, fast, etc.); given such desiderata, preferences allow us to come to an appropriate compromise solution.

It is not difficult to envisage situations going beyond such simple preferences. Thus, there may be preferences among preferences. For example, in legal reasoning, laws may apply by default but the laws themselves may conflict. For instance, newer laws will usually have priority over less recent ones, and laws of a higher authority have priority over laws of a lower authority. In case of conflict, the "authority" preference takes priority over the "recency" preference. This also illustrates that one may have several preference orderings, where the orderings are by different criteria (recency, authority, specificity, etc.) and where one will need to adjudicate among these preferences to come up with a "global" preferred outcome.

We have two goals in this paper. First, we present a general framework based on default logic [Reiter, 1980] in which preferences may be expressed. Given that there has been a wide variety of approaches proposed for dealing with preference (Sections 3 and 7), this framework provides a uniform setting in which preference orderings can be expressed and compared. Second, we present a number of approaches to preference in this framework. In considering how preference orderings may be encoded in default logic, we address first the case where a default theory consists of world knowledge and a set of default rules together with (external) preference information between default rules. We show how such a default theory can be translated into a second theory where preference information is now incorporated in the theory. With this translation we obtain a theory in standard default logic, rather than requiring machinery external to default logic, as is found in previous approaches. We next generalise this approach so that preferences may appear arbitrarily as part of a default theory and, specifically, preferences among default rules may (via the naming of default rules) themselves be part of a default rule. This allows the specification of preferences among preferences, preferences holding in a particular context, or preferences holding by default. This allows one to axiomatise within a theory how different preferences orderings interact. As well we consider elaborations to these approaches. In these approaches, we formalise a *prescriptive* notion of preference, wherein the ordering specifies the order in which default rules are to be applied. This is in contrast to a *descriptive* notion of preference, where the order reflects the "desirability" that a rule be applied.

Previous approaches have generally added machinery to an extant approach to nonmonotonic reasoning. In contrast, we remain within the framework of standard default logic, rather than building a scheme on top of default logic. This has several advantages. Foremost, the approach is *flexible*. As stated above, we can axiomatise how a preference order interacts with other knowledge, including other default information and preference orders. Thus we can integrate different orderings in the same setting, with arbitrary relationships (or metaorderings) among them. Second, it is easier to compare differing approaches to handling such orderings. Third, by "compiling" preferences into default logic, and in using the standard machinery of default logic, we obtain insight into the notion of preference orderings. So, for instance, if someone doesn't like our notion of preference given here, they are free to axiomatise their own within this framework. Also, for example, we implicitly show that explicit priorities provide no real increase in the expressibility of default logic. This final point is particularly important given that nonmonotonic reasoning systems are now beginning to find application in practical reasoning systems; hence explicitly dealing with preferences may be seen as a step in developing knowledge engineering methods for applying default reasoning technologies in reasoning systems. Lastly, there exist theorem provers for default logic. Consequently our approach can be immediately incorporated in such a prover. To this end, our approach has been implemented under the syntactic restriction of extended logic programming; this implementation serves as a front-end to the logic programming systems `dlv` and `smodels`.

2 Default Logic and Ordered Default Logic

Default logic [Reiter, 1980] augments classical logic by *default rules* of the form $\frac{\alpha:\beta_1,\dots,\beta_n}{\gamma}$. For the most part we deal with *singular* defaults for which $n = 1$. [Marek and Truszczyński, 1993] show that any default rule can be transformed into a set of defaults with $n = 1$ and $n = 0$; hence our one use of a non-singular rule in Section 5 is for notational convenience only. A singular rule is *normal* if β is equivalent to γ; it is *semi-normal* if β implies γ. We sometimes denote the *prerequisite* α of a default δ by $Prereq(\delta)$, its *justification* β by $Justif(\delta)$, and its *consequent* γ by $Conseq(\delta)$. Accordingly, $Prereq(D)$ is the set of prerequisites of all default rules in D; $Justif(D)$ and $Conseq(D)$ are defined analogously. Empty components, such as no prerequisite or even no justifications, are assumed to be tautological. Defaults with unbound variables are taken to stand for all corresponding instances. A set of default rules D and a set of formulas W form a *default theory* (D, W) that may induce a single or multiple *extensions* in the following way.

Definition 2.1 *Let (D, W) be a default theory and let E be a set of formulas. Define $E_0 = W$ and for $i \geq 0$:*

$$\begin{aligned} GD_i &= \left\{ \tfrac{\alpha:\beta_1,\dots,\beta_n}{\gamma} \in D \,\middle|\, \alpha \in E_i, \neg\beta_1 \notin E, \dots, \neg\beta_n \notin E \right\} \\ E_{i+1} &= Th(E_i) \cup \{Conseq(\delta) \mid \delta \in GD_i\} \end{aligned}$$

Then E is an extension for (D, W) if $E = \bigcup_{i=0}^{\infty} E_i$.

Any such extension represents a possible set of beliefs about the world at hand. The above procedure is not constructive since E appears in the specification of GD_i. We define $GD(D, E) = \bigcup_{i=0}^{\infty} GD_i$ as the set of default rules *generating* extension E. An enumeration $\langle\delta_i\rangle_{i\in I}$ of default rules is *grounded* in a set of formulas W, if we have for every $i \in I$ that $W \cup Conseq(\{\delta_0, \dots, \delta_{i-1}\}) \vdash Prereq(\delta_i)$.

For adding preferences among default rules, a default theory is usually extended with an ordering on the set of default rules. In analogy to [Baader and Hollunder, 1993a; Brewka, 1994a], an *ordered default theory* $(D, W, <)$ is a finite set D of default rules, a finite set W of formulas, and a strict partial order $< \subseteq D \times D$ on the default rules. That is, $<$ is a binary irreflexive and transitive relation on D. For simplicity in the following development we assume the existence of a default $\delta_\top = \frac{\top : \top}{\top} \in D$ where for every rule $\delta \in D$, we have $\delta < \delta_\top$ if $\delta \neq \delta_\top$. This gives us a (trivial) maximally preferred default that is always applicable.

3 What's a Default Preference?

This section discusses preference orderings in general. While we employ default logic, the discussion is independent of any particular approach to nonmonotonic reasoning. Assume that we have an *ordered default theory*. We can write $\frac{\alpha_1 : \beta_1}{\gamma_1} < \frac{\alpha_2 : \beta_2}{\gamma_2}$ to express a preference between two defaults. Informally, the intent is that a higher-ranked default should be applied or considered before a lower-ranked default.

The notion of *preference* among defaults, broadly construed, is very general, in that there are few restrictions that one would place on default rules in a preference ordering. Consider for example, the defaults "Canadians speak English", "Québécois speak French", "residents of the north of Québec speak Cree". A preference ordering can be expressed as follows:

$$\frac{Can : English}{English} < \frac{Que : French}{French} < \frac{NQue : Cree}{Cree}. \tag{1}$$

So if a resident of the north of Québec didn't speak Cree, it would be reasonable to assume that that person spoke French, and if they didn't speak French, then English. Here we have a relation of specificity (or subsumption) among the default rule prerequisites. Consider though a variation on (1) where in the north of Québec the first language is French, then English, then Cree: The resulting preference ordering is as follows.

$$\frac{NQue : Cree}{Cree} < \frac{Can : English}{English} < \frac{Que : French}{French}. \tag{2}$$

So here there is no specificity order implied by $<$ among rules prerequisites. Indeed for preferences, one need not have any antecedent information. That one prefers something (say, a car) that is red, then green might be expressed as $\frac{: Green}{Green} < \frac{: Red}{Red}$. In the most general case, we might have two defaults, with no relation between them except for a given a priori preference relation.

Preferences may also apply to other preferences. The legal reasoning example given in the introduction would be such an instance. Finally one may have different preferences in different contexts, and as well as preferences by default. So, all in all, one may encounter quite a variety of different preferences in the reasoning process. We address these possibilities here using default logic. The novel feature of our approach is that preferences are dealt with within the extant framework of default logic. We do this by introducing machinery whereby the application of default rules may be very tightly controlled. Given this machinery, we show how a given preference ordering may be "compiled" into a "standard" default theory in which defaults are applied according to this ordering. Consequently one has the freedom and flexibility to axiomatise within a theory how different orderings interact, when they apply, etc.

We have argued elsewhere [Delgrande and Schaub, 2000] that the notion of *inheritance of properties* is distinct from that of preference. For default property inheritance, the ordering on defaults reflects a relation of *specificity* among the default rule prerequisites. Informally, for adjudicating among conflicting defaults, one determines the most specific (with respect to rule antecedents) defaults as candidates for application. Consider for example defaults concerning primary means of locomotion: "animals normally walk", "birds normally fly", "penguins normally swim":

$$\frac{Animal : Walk}{Walk} < \frac{Bird : Fly}{Fly} < \frac{Penguin : Swim}{Swim}. \tag{3}$$

If we learn that some thing is a penguin (and so a bird and animal), then we would want to apply the highest-ranked default, if possible, and only the highest-ranked default. Significantly, if the penguins-swim default is blocked (say the penguin in question has a fear of water) we *don't* try to apply the next default to see if it might fly. Our interests in this paper lie solely with *preference*; see [Delgrande and Schaub, 2000] for an encoding of inheritance of properties.

Of approaches dealing with inheritance of properties, in [Touretzky *et al.*, 1987; Pearl, 1990; Geffner and Pearl, 1992] (among many others) specificity is determined implicitly, emerging as a property of an underlying formal system. [Reiter and Criscuolo, 1981; Etherington and Reiter, 1983; Delgrande and Schaub, 1994] have addressed adding specificity information in default logic. [Boutilier, 1992; Brewka, 1994a; Baader and Hollunder, 1993a] consider adding preferences in default logic while [McCarthy, 1986; Lifschitz, 1985; Grosof, 1991] and [Brewka, 1996; Zhang and Foo, 1997; Brewka and Eiter, 1998] do the same in circumscription and logic programming, respectively.[1] We return to these approaches in Section 7, once we have presented and developed our framework.

3.1 Prescriptive and descriptive preference

There are (at least) two ways that a preference order may be interpreted. For a *prescriptive* interpretation, the idea is that an order on defaults specifies the order in which the defaults are to be applied. Thus one applies (if possible) the most preferred default(s), the next most preferred, and so on. This approach then has a somewhat "algorithmic" feel to it. In a *descriptive* interpretation, the preference order represents a ranking on desired outcomes: the desirable (or: preferred) situation is one where the most preferred default(s) are applied.[2] The distinction between these interpretations is illustrated in the following example [Brewka and Eiter, 2000]:

$$\frac{: A}{A} < \frac{: \neg B}{\neg B} < \frac{A : B}{B}. \tag{4}$$

Assume that there is no initial world knowledge. In a prescriptive interpretation, one would fail to apply the most preferred default (viz. $\frac{A : B}{B}$) since the antecedent isn't provable. However, one might expect to apply the two lesser-preferred defaults, giving an extension containing $\{A, \neg B\}$.[3] In a descriptive interpretation one might observe that by applying the

[1]Although these latter papers include examples best interpreted as dealing with property inheritance, arguably they in fact implement the (distinct) notion of preference, described following.

[2]This isn't intended as a cut-and-dried distinction, but rather as an often useful classification. For example, [Brewka and Eiter, 2000] contains elements of both.

[3]This is for instance obtained in [Baader and Hollunder, 1993a; Brewka, 1994a; Marek and Truszczyński, 1993]; the approach presented in Section 4 yields no "preferred" extension.

least-preferred default, the most preferred default can be applied; this yields an extension containing $\{A, B\}$. This has led some researchers to advocate systems based on the descriptive interpretation.

In contrast, we advocate a prescriptive interpretation. We elaborate on this in Section 7, but it is worth summarising our reasons for favouring this approach here. First, a descriptive interpretation seems to require (at least in its obvious implementation) a meta-level approach, or failing that, an expensive encoding at the object level (Section 6). This is due to the fact that one wants to find a scenario (i.e. extension) in which the most preferred default(s) are applied, enabled perhaps via the application of *other*, arbitrary, defaults. In contrast, in the prescriptive approach, one may generate an extension, and be guaranteed that it represents a scenario in which the most preferred default(s) that can be applied are applied. Second, there are interesting ordered default theories where, in a prescriptive interpretation, one can guarantee the existence of a most-preferred extension, generated by a strictly iterative process (see Theorem 4.6). Hence there is reason to believe that a prescriptive interpretation will generally be more efficient than a descriptive interpretation (even though the respective complexity classes may be the same) and specific instances in which it is guaranteed to be much more efficient. In addition, if a descriptive interpretation uses a meta-level approach then adjudicating among different preference orderings, choosing preferences by default, and all the generalities discussed above must be determined at the meta-level. In contrast, with our prescriptive approach, we can axiomatise within our theory how we want different preference orders to interact.

Lastly, a prescriptive interpretation arguably comes with more representational "force" and allows a "tighter" characterisation of a domain. This is illustrated by the example (4). Here the prescriptive interpretation appears to give a curious result. However, we argue the problem is not with a prescriptive interpretation per se, but rather with the encoding of the example. The default $\frac{A:B}{B}$ has highest priority, but this default can only be applied if the prerequisite is proved; one way that this can come about is by applying the default $\frac{:A}{A}$. But then it would seem that $\frac{:A}{A}$ should be considered first and thus have higher priority than $\frac{A:B}{B}$, since it enables the application of this default. Second, there is no situation in which $\frac{A:B}{B}$ can be applied and $\frac{:A}{A}$ cannot. Thus, while the default $\frac{:A}{A}$ may be pragmatically less "important" than $\frac{A:B}{B}$ in a theory, the inference structure of default logic is such that $\frac{:A}{A}$ cannot be applied after $\frac{A:B}{B}$.[4] Yet this is what the order $<$ in (4) stipulates. An analogy may be made with proving a theorem: a theorem (by analogy: $\frac{A:B}{B}$) may be "important" and lemmas (by analogy: $\frac{:A}{A}$) may be less "important", but one way or another the lemmas are proved before the theorem can be proved. Hence we argue that (4), while syntactically well-formed, is of questionable meaning. More generally, a prescriptive interpretation forces a knowledge base designer to be explicit about what things should be applied in what order. A descriptive interpretation on the other hand simply gives a "wish list" of preferences which may or may not be meaningful. We return to and elaborate on these points at the end of the paper in Section 7, where we compare our approach with others.

[4] That is, one cannot have a grounded enumeration of the generating defaults (Definition 2.1) in which $\frac{A:B}{B}$ is applied before $\frac{:A}{A}$.

4 Static Preferences on Defaults

We show here how ordered default theories can be translated into standard default theories. Our strategy is to add sufficient "tags" to a default rule in a theory to enable the control of rule application. This is comparable to the usage of abnormality predicates in circumscription [McCarthy, 1986]. We are given an ordered default theory $(D, W, <)$ which is then translated into a regular default theory (D', W') such that the explicit preferences in $<$ are "compiled" into D' and W'.

We begin by associating a unique name with each default rule. This is done by extending the original language by a set of constants[5] N such that there is a bijective mapping $n : D \to N$. We write n_δ instead of $n(\delta)$ (and we often abbreviate n_{δ_i} by n_i to ease notation). Also, for default rule δ along with its name n, we sometimes write $n : \delta$ to render naming explicit. To encode the fact that we deal with a finite set of distinct default rules, we adopt a unique names assumption (UNA_N) and domain closure assumption (DCA_N) with respect to N. That is, for a name set $N = \{n_1, \ldots, n_m\}$, we add axioms

$$\begin{array}{ll} \mathrm{UNA}_N: & (n_i \neq n_j) \ \text{ for all } \ n_i, n_j \in N \ \text{ with } \ i \neq j \\ \mathrm{DCA}_N: & \forall x.\ name(x) \equiv (x = n_1 \vee \ldots \vee x = n_m). \end{array}$$

For convenience, we write $\forall x \in N.\ P(x)$ instead of $\forall x.\ name(x) \supset P(x)$.

The use of names allows the expression of preference relations between default rules in the object language. So we assert that default $n_j : \frac{\alpha_j : \beta_j}{\gamma_j}$ is preferred to $n_i : \frac{\alpha_i : \beta_i}{\gamma_i}$ by $n_i \prec n_j$, where $\prec$ is a (new) predicate in the object language. Finally, in discussions of preference relations we sometimes write $\delta_i < \delta_j$ or $\frac{\alpha_i : \beta_i}{\gamma_i} < \frac{\alpha_j : \beta_j}{\gamma_j}$ to show a preference between two defaults; however it should be kept in mind that these latter expressions are not expressions *within* a default theory (as are given by $\prec$), but expressions *about* a default theory.

Given $\delta_i < \delta_j$, we want to ensure that before δ_i is applied, that δ_j be applied or found to be inapplicable.[6] We do this by first translating default rules so that rule application can be explicitly controlled. For this purpose, we need to be able to, first, detect when a rule has been applied or when a rule is blocked, and, second, control the application of a rule based on other antecedent conditions. For a default rule $\frac{\alpha : \beta}{\gamma}$, there are two cases for it to not be applied: it may be that the antecedent is not known to be true (and so its negation is consistent), or it may be that the justification is not consistent (and so its negation is known to be true). For detecting this case, we introduce a new, special-purpose predicate $\mathsf{bl}(\cdot)$. Similarly we introduce a special-purpose predicate $\mathsf{ap}(\cdot)$ to detect the case where a rule has been applied. For controlling application of a rule we introduce predicate $\mathsf{ok}(\cdot)$. Then, a default rule $\delta = \frac{\alpha : \beta}{\gamma}$ is mapped to

$$\frac{\alpha \wedge \mathsf{ok}(n_\delta)\ :\ \beta}{\gamma \wedge \mathsf{ap}(n_\delta)}, \quad \frac{\mathsf{ok}(n_\delta)\ :\ \neg\alpha}{\mathsf{bl}(n_\delta)}, \quad \frac{\neg\beta \wedge \mathsf{ok}(n_\delta)\ :}{\mathsf{bl}(n_\delta)}\ . \tag{5}$$

These rules are sometimes abbreviated by $\delta_a, \delta_{b_1}, \delta_{b_2}$, respectively. While δ_a is more or less the image of the original rule δ, rules δ_{b_1} and δ_{b_2} capture the aforementioned situation of non-applicability.

[5][McCarthy, 1986] first suggested naming defaults using a set of *aspect* functions. See also [Brewka, 1994b]. Theorist [Poole, 1988] uses atomic propositions to name defaults.

[6]That is, we wish to exclude the case where $\delta_i \in GD_n$ and $\delta_j \in GD_m$ for $n \leq m$ in Definition 2.1.

None of the three rules in (5) can be applied unless $\mathsf{ok}(n_\delta)$ is true. Since $\mathsf{ok}(\cdot)$ is a new predicate symbol, it can be expressly made true in order to potentially enable the application of the three rules in the image of the translation. If $\mathsf{ok}(n_\delta)$ is true, the first rule of the translation may potentially be applied. If a rule has been applied, then this is indicated by assertion $\mathsf{ap}(n_\delta)$. The last two rules give conditions under which the original rule is inapplicable: either the negation of the original antecedent α is consistent (with the extension) or the justification β is known to be false; in either such case $\mathsf{bl}(n_\delta)$ is concluded.

This translation says nothing about which defaults are "considered" in an ordering before others. However, for $\delta_i < \delta_j$ we can now fully control the order of rule application: if δ_j has been applied (and so $\mathsf{ap}(n_j)$ is true), or known to be inapplicable (and so $\mathsf{bl}(n_j)$ is true), then it's ok to apply δ_i. So we would have something like $(\mathsf{ap}(n_j) \vee \mathsf{bl}(n_j)) \supset \mathsf{ok}(n_i)$, but adjusted to allow for the fact that there might be other rules with higher priority than δ_i. The idea is thus to *delay* the consideration of less preferred rules until the applicability question has been settled for the respective higher ranked rules.

Taking all this into account, we obtain the following translation, mapping ordered default theories in some language $\mathcal{L}$ onto standard default theories in the language $\mathcal{L}^+$ obtained by extending $\mathcal{L}$ by new predicates symbols $(\cdot \prec \cdot)$, $\mathsf{ok}(\cdot)$, $\mathsf{bl}(\cdot)$, and $\mathsf{ap}(\cdot)$, and a set of associated default names:

Definition 4.1 *Given an ordered default theory $(D, W, <)$ over $\mathcal{L}$ and its set of default names $N = \{n_\delta \mid \delta \in D\}$, define $\mathcal{T}((D, W, <)) = (D', W')$ over $\mathcal{L}^+$ by*

$$\begin{aligned} D' &= \left\{ \tfrac{\alpha \wedge \mathsf{ok}(n) : \beta}{\gamma \wedge \mathsf{ap}(n)},\ \tfrac{\mathsf{ok}(n) : \neg\alpha}{\mathsf{bl}(n)},\ \tfrac{\neg\beta \wedge \mathsf{ok}(n) :}{\mathsf{bl}(n)} \;\middle|\; n : \tfrac{\alpha : \beta}{\gamma} \in D \right\} \cup D_\prec \\ W' &= W \cup W_\prec \cup \{DCA_N, UNA_N\} \end{aligned}$$

where

$$\begin{aligned} D_\prec &= \left\{ \tfrac{: \neg(x \prec y)}{\neg(x \prec y)} \right\} \\ W_\prec &= \{ n_\delta \prec n_{\delta'} \mid (\delta, \delta') \in < \} \\ &\cup \ \{\mathsf{ok}(n_\top)\} \\ &\cup \ \{\forall x \in N. [\forall y \in N. (x \prec y) \supset (\mathsf{bl}(y) \vee \mathsf{ap}(y))] \supset \mathsf{ok}(x)\}. \end{aligned}$$

W' contains prior world knowledge W, together with assertions for managing the priority order $<$ on defaults. The first part of $W_\prec$ specifies that $\prec$ is a predicate whose positive instances mirror those of the strict partial order $<$. $\mathsf{ok}(n_\top)$ asserts that it is ok to apply the maximally preferred (trivial) default. The third formula in $W_\prec$ controls the application of defaults: for every n_i, we derive $\mathsf{ok}(n_i)$ whenever for every n_j with $n_i \prec n_j$, either $\mathsf{ap}(n_j)$ or $\mathsf{bl}(n_j)$ is true. This axiom allows us to derive $\mathsf{ok}(n_i)$, indicating that δ_i may potentially be applied whenever we have for all δ_j with $\delta_i < \delta_j$ that δ_j has been applied or cannot be applied.

This alone gives necessary but not sufficient conditions for rendering δ_i potentially applicable. If $(\delta_i, \delta_j) \notin <$ then $(n_i \prec n_j) \notin W_\prec$; however, for the last formula in $W_\prec$ to work properly we must be able to conclude (in the extension) that $\neg(n_i \prec n_j)$. This is addressed by adding the default rule in $D_\prec$ that renders the resulting theory complete with respect to priority statements. That is, for all resulting extensions E we have that $(n_i \prec n_j) \in E$

or $\neg(n_i \prec n_j) \in E$. We also have $(n_\delta \prec n_\top) \in W'$ for every rule $\delta \neq \delta_\top$ by the definition of ordered default theories. Since $<$ is a strict partial order, W' also includes the transitive closure of $\prec$ and no reflexivities such as $n \prec n$.

Note that the translation results in a manageable increase in the size of the default theory. For ordered theory $(D, W, <)$, the translation $\mathcal{T}((D, W, <))$ is only a constant factor larger than $(D, W, <)$.[7]

As an example, consider the defaults:

$$n_1 : \frac{A_1 : B_1}{C_1}, \quad n_2 : \frac{A_2 : B_2}{C_2}, \quad n_3 : \frac{A_3 : B_3}{C_3}, \quad n_\top : \frac{\top : \top}{\top}.$$

We obtain for $i = 1, 2, 3$:

$$\frac{A_i \wedge \mathsf{ok}(n_i) : B_i}{C_i \wedge \mathsf{ap}(n_i)}, \qquad \frac{\mathsf{ok}(n_i) : \neg A_i}{\mathsf{bl}(n_i)}, \qquad \frac{\neg B_i \wedge \mathsf{ok}(n_i) :}{\mathsf{bl}(n_i)},$$

and analogously for $\delta_\top$ where A_i, B_i, C_i are $\top$. Given $\delta_1 < \delta_2 < \delta_3$, we obtain $n_1 \prec n_2$, $n_2 \prec n_3$, $n_1 \prec n_3$ along with $n_k \prec n_\top$ for $k \in \{1, 2, 3\}$ as part of $W_\prec$. From $D_\prec$ we get $\neg(n_i \prec n_j)$ for all remaining combinations of $i, j \in \{1, 2, 3, \top\}$. It is instructive to verify that $\mathsf{ok}(n_3)$, along with

$$(\mathsf{ap}(n_3) \vee \mathsf{bl}(n_3)) \supset \mathsf{ok}(n_2), \quad \text{and} \quad ((\mathsf{ap}(n_2) \vee \mathsf{bl}(n_2)) \wedge (\mathsf{ap}(n_3) \vee \mathsf{bl}(n_3))) \supset \mathsf{ok}(n_1)$$

are obtained after a few iterations in Definition 2.1 (see below); from this we get that n_3 must be taken into account first, followed by n_2 and then n_1.

For illustration, we provide in Figure 1 traces of extension constructions based on the pseudo-iterative specification given in Definition 2.1: Given $W = \{A_1, A_2, A_3\}$, we obtain the trace of conclusions given in the left column of Figure 1. The trace demonstrates how the successive introduction of ok-literals allows for navigating the consecutive consideration of rules along the given preferences. The delay between the application of the individual rules is due to the fact that in Definition 2.1 the deductive closure of E_i is determined at E_{i+1}.

Next, consider where W also contains $C_3 \supset \neg B_2$ and $C_2 \supset \neg B_3$. The corresponding trace is given in the middle column of Figure 1. We get $(\delta_2)_{b_2} \in GD_5$ instead of $(\delta_2)_a \in GD_5$; therefore, also $\mathsf{bl}(n_2) \in E_6$ instead of $C_2 \wedge \mathsf{ap}(n_2) \in E_6$. This is because $\neg B_2 \in E$. Suppose there is an extension containing $C_2 \wedge \neg B_3$ as opposed to $C_3 \wedge \neg B_2$. As before, we obtain $\mathsf{ok}(n_3) \in E_3$. Since we have $\neg B_3$ in our putative extension, against which we check consistency, $(\delta_3)_a$ is inapplicable. Also, $(\delta_3)_{b_1}$ is inapplicable, since A_3 belongs to the given facts. Finally, not even $(\delta_3)_{b_2}$ is applicable since $\neg B_3$ is not *derivable*, although it belongs to the putative extension. So, since we can neither derive $\mathsf{ap}(n_3)$ nor $\mathsf{bl}(n_3)$, the pseudo-iterative process is interrupted; we cannot derive $\mathsf{ok}(n_2)$ and thus $\neg B_3$ cannot belong to any E_i. This behaviour nicely reflects the fact that rules (to be more precise, their justifications) can only be blocked by higher-ranked rules, since the application of lower-ranked rules is delayed until applicability has been settled for their predecessors.

Finally, consider the case where the prerequisite of the second default rule, A_2, is missing. The corresponding trace is given in the right column of Figure 1. As opposed to the two previous scenarios, we now get $(\delta_2)_{b_1}$ in GD_5.

[7] This assumes we count the default in $D_\prec$ as a single default.

i	E_i	GD_i
0	$\mathsf{ok}(n_\top)$	
	A_1, A_2, A_3	
1	$\top \wedge \mathsf{ok}(n_\top)$	$(\delta_\top)_a$
2	$\top \wedge \mathsf{ap}(n_\top)$	
3	$\mathsf{ap}(n_\top)$	
	$\mathsf{ok}(n_3)$	
	$A_3 \wedge \mathsf{ok}(n_3)$	$(\delta_3)_a$
4	$C_3 \wedge \mathsf{ap}(n_3)$	
5	$C_3,\ \mathsf{ap}(n_3)$	
	$\mathsf{ok}(n_2)$	
	$A_2 \wedge \mathsf{ok}(n_2)$	$(\delta_2)_a$
6	$C_2 \wedge \mathsf{ap}(n_2)$	
7	$C_2,\ \mathsf{ap}(n_2)$	
	$\mathsf{ok}(n_1)$	
	$A_1 \wedge \mathsf{ok}(n_1)$	$(\delta_1)_a$
8	$C_1 \wedge \mathsf{ap}(n_1)$	
9	$C_1,\ \mathsf{ap}(n_1)$	

i	E_i	GD_i
0	$\mathsf{ok}(n_\top)$	
	A_1, A_2, A_3	
	$C_3 \supset \neg B_2$	
	$C_2 \supset \neg B_3$	
1	$\top \wedge \mathsf{ok}(n_\top)$	$(\delta_\top)_a$
2	$\top \wedge \mathsf{ap}(n_\top)$	
3	$\mathsf{ap}(n_\top)$	
	$\mathsf{ok}(n_3)$	
	$A_3 \wedge \mathsf{ok}(n_3)$	$(\delta_3)_a$
4	$C_3 \wedge \mathsf{ap}(n_3)$	
5	$C_3,\ \mathsf{ap}(n_3)$	
	$\mathsf{ok}(n_2),\ \neg B_2$	
	$\neg B_2 \wedge \mathsf{ok}(n_2)$	$(\delta_2)_{b_2}$
6	$\mathsf{bl}(n_2)$	
7		
	$\mathsf{ok}(n_1)$	
	$A_1 \wedge \mathsf{ok}(n_1)$	$(\delta_1)_a$
8	$C_1 \wedge \mathsf{ap}(n_1)$	
9	$C_1,\ \mathsf{ap}(n_1)$	

i	E_i	GD_i
0	$\mathsf{ok}(n_\top)$	
	$A_1, \qquad A_3$	
1	$\top \wedge \mathsf{ok}(n_\top)$	$(\delta_\top)_a$
2	$\top \wedge \mathsf{ap}(n_\top)$	
3	$\mathsf{ap}(n_\top)$	
	$\mathsf{ok}(n_3)$	
	$A_3 \wedge \mathsf{ok}(n_3)$	$(\delta_3)_a$
4	$C_3 \wedge \mathsf{ap}(n_3)$	
5	$C_3,\ \mathsf{ap}(n_3)$	
	$\mathsf{ok}(n_2)$	$(\delta_2)_{b_1}$
6	$\mathsf{bl}(n_2)$	
7		
	$\mathsf{ok}(n_1)$	
	$A_1 \wedge \mathsf{ok}(n_1)$	$(\delta_1)_a$
12	$C_1 \wedge \mathsf{ap}(n_1)$	
13	$C_1,\ \mathsf{ap}(n_1)$	

Figure 1: Tracing the pseudo-iterative definition.

The following theorem summarises the major technical properties of our approach, and demonstrate that rules are applied in the desired order:

Theorem 4.1 *Let E be a consistent extension of $\mathcal{T}((D,W,<))$ for ordered default theory $(D,W,<)$. We have for all $\delta,\delta' \in D$ that*

1. $n_\delta \prec n_{\delta'} \in E$ *iff* $\neg(n_\delta \prec n_{\delta'}) \notin E$
2. $\mathsf{ok}(n_\delta) \in E$
3. $\mathsf{ap}(n_\delta) \in E$ *iff* $\mathsf{bl}(n_\delta) \notin E$
4. $\mathsf{ok}(n_\delta) \in E_i$ *and* $\mathit{Prereq}(\delta) \in E_j$ *and* $\neg \mathit{Justif}(\delta) \notin E$ *implies* $\mathsf{ap}(n_\delta) \in E_{\max(i,j)+3}$
5. $\mathsf{ok}(n_\delta) \in E_i$ *and* $\mathit{Prereq}(\delta) \notin E$ *implies* $\mathsf{bl}(n_\delta) \in E_{i+1}$
6. $\mathsf{ok}(n_\delta) \in E_i$ *and* $\neg \mathit{Justif}(\delta) \in E$ *implies* $\mathsf{bl}(n_\delta) \in E_j$ *for some* $j > i+1$
7. $\mathsf{ok}(n_\delta) \notin E_{i-1}$ *and* $\mathsf{ok}(n_\delta) \in E_i$ *implies* $\mathsf{ap}(n_\delta) \notin E_j$ *for* $j < i+2$ *and* $\mathsf{bl}(n_\delta) \notin E_j$ *for* $j < i$

While Theorem 4.1.2 guarantees that we consider all default rules in D, Theorem 4.1.3 reflects the fact that extensions contain complete knowledge about the application of the rules in D. Among the more procedural propositions, Theorem 4.1.5 shows that our approach allows us to detect blockage due to non-derivability of the prerequisite immediately after having the "ok" for the default at hand. The remaining properties provide a detailed account of the technical intuitions underlying our approach: as with all default logics, we consider all default rules during the consideration of rules with lower preference.

This is made more precise in the following theorem, by adopting a rule-based perspective. For an extension E and its generating default rules $GD(D,E)$, we trivially have $\delta_a \in GD(D,E)$ iff $\mathsf{ap}(n_\delta) \in E$, and $\delta_{b_1} \in GD(D,E)$ or $\delta_{b_2} \in GD(D,E)$ iff $\mathsf{bl}(n_\delta) \in E$.

Theorem 4.2 *Let E be a consistent extension of $\mathcal{T}((D,W,<)) = (D',W')$ for ordered default theory $(D,W,<)$ and GD_i be defined wrt E and (D',W'). Then, we have for all $\delta \in D$*

8. $\delta_a \in GD(D',E)$ *iff* $(\delta_{b_1} \notin GD(D',E)$ *and* $\delta_{b_2} \notin GD(D',E))$

For all default rules $\delta,\delta' \in D$ such that $\delta < \delta'$, we have

9. $\delta'_a, \delta'_{b_1}, \delta'_{b_2} \notin GD_i$ *implies* $\delta_a, \delta_{b_1}, \delta_{b_2} \notin GD_j$ *for* $j < i+3$
10. $\delta'_a \in GD_i$ *or* $\delta'_{b_1} \in GD_i$ *or* $\delta'_{b_2} \in GD_i$ *implies* $\delta_a \in GD_j$ *or* $\delta_{b_1} \in GD_j$ *or* $\delta_{b_2} \in GD_j$ *for some* $j > i+2$
11. $\delta_a \in GD_i$ *or* $\delta_{b_1} \in GD_i$ *or* $\delta_{b_2} \in GD_i$ *implies* $\delta'_a \in GD_j$ *or* $\delta'_{b_1} \in GD_j$ *or* $\delta'_{b_2} \in GD_j$ *for some* $j < i-2$.

Unlike the above, the last series of results focuses on the successive application of lower and higher ranked rules. As already observable in the traces in Figure 1, the minimum three-step delay between rules stemming from δ and those originated by δ' is due to the belated formation of deductive closure in Definition 2.1. The important overall consequence of this series of results is that our translation provides full control over default application.

4.1 Semantical underpinnings

So far, we have described our approach in rather specific, technical terms, so that the question about more general properties arises. To begin with, a more global and systematic view of the machinery put forward by our translation is obtainable by combining the previously given results:

Theorem 4.3 *Let $(D, W, <)$ be an ordered default theory and let E be a set of formulas.*

If E is a consistent extension of $\mathcal{T}((D, W, <)) = (D', W')$ then we have for all grounded enumerations $\langle \zeta_i \rangle_{i \in I}$ of $GD(D', E)$ and for all $\delta, \delta' \in D$:

$$\text{If } \delta < \delta', \text{ then } j < i \qquad \text{for all } \zeta_i = \delta_t \text{ and some}^8\ \zeta_j = (\delta')_{t'} \text{ with } t, t' \in \{a, b_1, b_2\}\ .$$

Thus a grounded enumeration for an extension of the translated defaults conforms to the ordering given on the initial set of defaults.

Moreover, it turns out that our translation $\mathcal{T}$ amounts to selecting those extensions of the original default theory that are in accord with the provided ordering. This can be expressed in the following way.

Definition 4.2 *Let (D, W) be a default theory and let $< \subseteq D \times D$ be a strict partial order.*

An extension E of (D, W) is $<$-preserving if there exists a grounded enumeration $\langle \delta_i \rangle_{i \in I}$ of $GD(D, E)$ such that for all $i, j \in I$ and $\delta \in D \setminus GD(D, E)$, we have that

1. *if $\delta_i < \delta_j$ then $j < i$ and*
2. *if $\delta_i < \delta$ then $Prereq(\delta) \notin E$ or $W \cup Conseq(\{\delta_0, \ldots, \delta_{i-1}\}) \vdash \neg Justif(\delta)$.*

From the perspective of δ_i, the two conditions distinguish between present and absent higher-ranked rules. In the former case, an ordering $<$ prescribes that the preferred rule δ_j must be applied before δ_i, while its absence is only tolerable if either its prerequisite is not derivable (at all) or its justification is refuted by other, even higher-ranked rules. In any case, the applicability issue must first be settled for higher-ranked default rules before it is addressable for lower-ranked rules. This conception of prescriptive preference is mirrored by our translation of ordered theories into regular theories.

Theorem 4.4 *Let (D, W) be a default theory and let $< \subseteq D \times D$ be a strict partial order. Let E be a set of formulas.*

We have that E is a $<$-preserving extension of (D, W) iff $E = E' \cap \mathcal{L}$ for some extension E' of $\mathcal{T}((D, W, <))$.

The notion of $<$-preservation not only provides a semantics for our approach, but it may also be seen as a general semantical account for prescriptive preferences among defeasible rules.

By the above theorem, it is clear that any extension of a translated default theory is a regular extension of the underlying *unordered* default theory:

[8] This is because there are cases, where both δ_{b_1} and δ_{b_2} apply.

Corollary 4.1 *Let $(D, W, <)$ be an ordered default theory over $\mathcal{L}$.*
If E is an extension of $\mathcal{T}((D, W, <))$ then $E \cap \mathcal{L}$ is an extension of (D, W).

Also, the approach is equivalent (modulo the original language) to standard default logic if there are no preferences:

Corollary 4.2 *For a default theory (D, W) over $\mathcal{L}$ and a set of formulas E, we have that E is an extension of (D, W) iff $E = E' \cap \mathcal{L}$ for some extension E' of $\mathcal{T}((D, W, \emptyset))$.*

4.2 The Existence of Extensions

Given the above theorems, one might expect that ordered default theories would enjoy the same properties as standard default logic. This is indeed the case, but with one important exception: normal ordered default theories do not guarantee the existence of extensions. For example, the image of the ordered default theory (under our translation)

$$\left(\{n_1 : \tfrac{:B}{B},\ n_2 : \tfrac{B:C}{C}\}, \emptyset, \{\delta_1 < \delta_2\}\right) \tag{6}$$

has no extension. Informally the problem is that we have a preference $\delta_1 < \delta_2$. However, if $W = \emptyset$, only default δ_1 is applicable, and once it has applied, δ_2 becomes applicable. Thus we have an ordering implicit in the form of the defaults and world knowledge, but where this implicit ordering is contradicted by the assertion $\delta_1 < \delta_2$. Not surprisingly then there is no extension.

In more detail, our translation gives us (among other things) the following information:

$$\tfrac{\mathsf{ok}(n_1):B}{B\wedge \mathsf{ap}(n_1)}, \quad \tfrac{B\wedge \mathsf{ok}(n_2):C}{C\wedge \mathsf{ap}(n_2)}, \quad \tfrac{\mathsf{ok}(n_2):\neg B}{\mathsf{bl}(n_2)}, \quad (\mathsf{ap}(n_2) \vee \mathsf{bl}(n_2)) \supset \mathsf{ok}(n_1). \tag{7}$$

If E is an extension of the translated theory, then in Definition 4.1 we will have for some i (in fact $i = 3$) that $\mathsf{ok}(n_2) \in E_i$ and $\mathsf{ok}(n_1) \notin E_i$. Only the third default in (7) may potentially be applied. The assumption that $B \in E$ leads to an immediate contradiction: this would block application of the third default, no default is applicable, and so $B \notin E$. Since $B \notin E$ the third default is applicable and we conclude $\mathsf{bl}(n_2) \in E_4$, hence $\mathsf{ok}(n_1) \in E_5$. But now the first default in (7) is applicable; we derive B, hence $B \in E$, contradiction.

From a declarative point of view, we see that the single regular extension of (6), viz. $Th(\{B, C\})$, is generated by the grounded sequence $\langle \delta_1, \delta_2 \rangle$. This extension however is not $<$-preserving, since its sequence of generating default rules violates the given preference ordering.[9] Conversely, the only sequence compatible with $<$, namely $\langle \delta_2, \delta_1 \rangle$, is not grounded and it is therefore not legitimate. So, we are faced with an explicit preference order that is incompatible with the implicit application order. In the example, this implicit order was induced by the inferential relation between one default rule's consequent and another's prerequisite. There is, however, a second source for such an implicit ordering, given by the blocking of a rule. That is, one rule's consequent may imply the negation of another's justification. To see this, consider the following theory:

$$\left(\{n_1 : \tfrac{:B}{B},\ n_2 : \tfrac{:C\wedge\neg B}{C}\}, \emptyset, \{\delta_1 < \delta_2\}\right) \tag{8}$$

[9] To be precise, it violates Condition *1* in Definition 4.2.

As above, this theory has no $<$-preserving extension. To see this, note that its single regular extension, viz. $Th(\{B\})$, is generated by $\langle\delta_1\rangle$. This sequence violates Condition *2* of Definition 4.2, since the inapplicability of the higher-ranked rule δ_2 is not justifiable in the absence of the lower-ranked rule δ_1. One way to address this problem is to replace rigid preferences by preferences that apply by default only. An approach which would permit this is detailed in Section 5. However we suggest that (6) and (8) are incoherent in some sense, and the lack of extension indicates a problem in the specification of the original theory.

We can make this notion more precise by relating our approach to that of [Papadimitriou and Sideri, 1994], which characterises a class of default theories that always have extensions. In [Papadimitriou and Sideri, 1994], a graph $G((D,W))$ is constructed from a default theory where default rules are the nodes, and directed edges reflect "positive" and "negative" influences of one rule on another. Our main result is that if a default theory is of this class that is guaranteed to have an extension, and if our ordering $<$ does not conflict with an edge in $G((D,W))$, then the ordered default theory has an extension.

We assume for the remainder of this section that we are dealing with propositional default theories only. That is, in both an original theory and the translated theory, we have propositional formulas only. Hence (former) atomic formulas such as $\mathsf{ok}(n)$ and $n_i \prec n_j$ are regarded as propositional atoms. We can do this since we assume that we have a finite set of default rules with a finite set of instances. Assume further that formulas are expressed in conjunctive normal form. To be sure, our axiomatisation of Definition 4.1 loses much of its interest (and so for instance the partial order $\prec$ becomes a set of unrelated atomic sentences) but the point remains that we can use this to characterise ordered theories having extensions.

[Papadimitriou and Sideri, 1994] present the following results. For semi-normal default theory (D,W) define its *literal graph* $L((D,W))$ as the graph with nodes consisting of literals in the theory, and where (x,y) is a directed edge if both $\neg x$ and y appear in the same clause of W or of a consequent of D. $L^*(x,y)$ indicates that there is a path from x to y in $L((D,W))$. Intuitively the graph indicates what literals may contribute to the proof of others in the context of the default theory.

From this another graph $G((D,W)) = (D,\mathcal{E})$, which we will call the *dependency graph*, is defined where the vertices are default rules from D, $\mathcal{E} = \mathcal{E}_0 \cup \mathcal{E}_1$, and $\mathcal{E}_0$ and $\mathcal{E}_1$ are disjoint. Informally $(\delta,\delta') \in \mathcal{E}_0$ if the application of δ may help bring about the application of δ', and $(\delta,\delta') \in \mathcal{E}_1$ if the application of δ may help prevent the application of δ'. Formally:

1. $(\delta,\delta') \in \mathcal{E}_0$ if there is a literal x appearing positively in the consequent of δ, a literal y appearing positively in the prerequisite of δ', and $L^*(x,y)$.

2. $(\delta,\delta') \in \mathcal{E}_1$ if there is a literal x appearing positively in the consequent of δ, a literal y appearing negatively in the justification or in the consequent of δ', and $L^*(x,y)$.

Observe that the information provided by $\mathcal{E}_0$ and $\mathcal{E}_1$ (along with the underlying distinction) amounts to a formalisation of the implicit application orderings discussed above (along with their two different sources).

Edges in $\mathcal{E}_0$ are assigned 0 weight and edges in $\mathcal{E}_1$ are assigned a weight of 1. Default theory (D,W) is *even* if all cycles in $G((D,W))$ have total weight that is even (i.e. there are no cycles with an odd number of edges from $\mathcal{E}_1$). Their main result says that every even default theory has an extension. Given this result, we obtain:

Theorem 4.5 *Let $(D, W, <)$ be a propositional, semi-normal, ordered default theory such that (D, W) is even and for the associated dependency graph $G((D, W)) = (D, \mathcal{E})$, we have that if $\mathcal{C}$ is a cycle of $(D, \mathcal{E} \cup \{(\delta', \delta) \mid \delta < \delta'\})$ then $\mathcal{C}$ is a cycle of $(D, \mathcal{E})$.*

Then $\mathcal{T}((D, W, <))$ has an extension.

Thus $(D, \mathcal{E} \cup \{(\delta', \delta) \mid \delta < \delta'\})$ has no new cycles incorporating elements of $<$. The dependency graph reflects dependencies among the default rules, specifically which rules may help block or activate others. [Papadimitriou and Sideri, 1994] show that theories obeying certain constraints have an extension; we show that an ordered theory where the ordering doesn't contradict one of these constraints also has an extension.

Now we can make precise why we consider theories (6) and (8) to be incoherent. First, we observe that both underlying default theories are even. For the standard theory underlying (6), we get a dependency graph with arc set $\mathcal{E} = \mathcal{E}_0 \cup \mathcal{E}_1 = \{(\delta_1, \delta_2)\} \cup \emptyset$. Together with the preference $\delta_1 < \delta_2$ from (6), we obtain arc set $\mathcal{E} \cup \{(\delta_2, \delta_1)\}$ that includes the new cycle formed by (δ_1, δ_2) and (δ_2, δ_1). Consider how this is reflected in the translated theory: Let $\mathcal{E}' = \mathcal{E}'_0 \cup \mathcal{E}'_1$ be the arc set of the dependency graph in the translated default theory $\mathcal{T}((D, W, <))$. As the proof of Theorem 4.5 shows, there are many arcs in $\mathcal{E}'_0$, most of them in a sense redundant, in that they cannot contribute to a cycle. However, two arcs are of crucial importance in the translated theory of (6). We get

$$\left(\frac{\mathsf{ok}(n_2):\neg B}{\mathsf{bl}(n_2)}, \frac{\mathsf{ok}(n_1):B}{B \wedge \mathsf{ap}(n_1)}\right) \in \mathcal{E}'_0$$

resulting from $(\mathsf{bl}(n_2), \mathsf{ok}(n_1))$ being in the literal graph of the translated theory, reflecting the preference of δ_2 over δ_1. We also get

$$\left(\frac{\mathsf{ok}(n_1):B}{B \wedge \mathsf{ap}(n_1)}, \frac{\mathsf{ok}(n_2):\neg B}{\mathsf{bl}(n_2)}\right) \in \mathcal{E}'_1$$

by virtue of the occurrences of B in the consequent and its negation in the justification of the rules. These two edges form an odd cycle (that is, a cycle with weight 1) and so the applicability condition of [Papadimitriou and Sideri, 1994] fails here. Note how this reflects our earlier intuitions. The first arc is a result of the preference of δ_2 over δ_1; the second results from the possibility of δ_1 activating δ_2.

In (8) something similar happens. In the underlying standard theory we get a dependency graph with arc set $\mathcal{E} = \mathcal{E}_0 \cup \mathcal{E}_1 = \emptyset \cup \{(\delta_1, \delta_2)\}$. Incorporating the arc (δ_2, δ_1) induced by $\delta_1 < \delta_2$ gives an extended arc set possessing a new cycle. In fact, in the translated theory we get edges

$$\left(\frac{\mathsf{ok}(n_1):B}{B \wedge \mathsf{ap}(n_1)}, \frac{\mathsf{ok}(n_2):C\wedge\neg B}{C \wedge \mathsf{ap}(n_2)}\right) \in \mathcal{E}'_1 \qquad \text{and} \qquad \left(\frac{\mathsf{ok}(n_2):C\wedge\neg B}{C \wedge \mathsf{ap}(n_2)}, \frac{\mathsf{ok}(n_1):B}{B \wedge \mathsf{ap}(n_1)}\right) \in \mathcal{E}'_0$$

again yielding an odd cycle. The first edge is a result of δ_1 "canceling" δ_2; the second reflects the preference of δ_2 over δ_1.

A further useful consequence follows. Its proof is independent of the Papadimitriou and Sideri result, and instead relies on the fact that normal default theories have extensions. Let $\mathit{Pred}(S)$ be the set of predicate symbols occurring in a set of formulas S.

Theorem 4.6 *Let $(D, W, <)$ be a normal, ordered default theory such that*

$$Pred(Prereq(D)) \cap Pred(Conseq(D)) = \emptyset \quad \text{and} \quad Pred(W) \cap Pred(Conseq(D)) = \emptyset.$$

Then $\mathcal{T}((D, W, <))$ has an extension.

Here we have a class of default theories where the default conclusions are independent of the rule prerequisites, and so represent new default information. The second condition excludes the expression of such "dependencies" via W. Hence the theory with $\frac{:q}{q} < \frac{:p}{p}$ for atomic sentences p, q, and with $W = \emptyset$ is acceptable, whereas the same preference with $W = \{p \equiv q\}$ is not.

Note that the proofs of the above theorems show that, as a corollary, an extension E is found by a purely iterative process. Hence, in select cases, extensions of ordered default theories may be (relatively) efficiently generated.

5 Dynamic Preferences on Defaults

We now consider situations where the presence of preferences is context-dependent. We deal with standard default theories (D, W) over a language already including a predicate $\prec$ expressing a preference relation by means of default names. In order to keep a finite domain closure axiom, we restrict ourselves to a finite set of default rules D which is in a 1–1 correspondence with a finite name set N.

Since preferences are now available dynamically by inferences from W and D, we lack a priori complete information about the ordering predicate $\prec$. This a priori complete information was available in the rigid case since we were given all positive instances of the explicit order $<$ between rules and the "closed world default" $\frac{:\neg(x\prec y)}{\neg(x\prec y)}$ for negative instances. However this leads to a problem if we allow positive preferences by default. Consider where our only preference is given by $\frac{:n\prec m}{n\prec m}$. Intuitively if we have no other preference information, this default should be applicable. However we also have the "closed world" default for preferences, given in $D_\prec$ above, that asserts that if there is no known or derived preference between rules, then no preference exists. An instance of $D_\prec$ is $\frac{:\neg(n\prec m)}{\neg(n\prec m)}$. So if we simply have these two defaults then we run the risk of potentially having an unwanted extension where $\frac{:\neg(n\prec m)}{\neg(n\prec m)}$ applies over $\frac{:n\prec m}{n\prec m}$. Obviously we cannot solve the problem by asserting that $\frac{:\neg(n\prec m)}{\neg(n\prec m)} < \frac{:\, n\prec m}{n\prec m}$ since our approach would now be circular. We address this issue by adding a new binary predicate $\not\prec$ indicating that for defaults δ and δ' neither $(n_\delta \prec n_{\delta'}) \in E$ nor $\neg(n_\delta \prec n_{\delta'}) \in E$ for a given extension E. We add the following rule, where x, y are variables ranging over default names:

$$\frac{:\ \neg(x \prec y), (x \prec y)}{(x \not\prec y)}. \tag{9}$$

This rule accounts for situations where neither $(x \prec y)$ nor $\neg(x \prec y)$ is derivable. That is for names n and m, the only time this rule will apply is when $n \prec m \notin E$ and $\neg(n \prec m) \notin E$. So, since $\not\prec$ is an introduced predicate, the only time we have $n \not\prec m \in E$ is when the default theory has no information on whether the two defaults are in a preference relation or not.

We now consider standard default theories in a language $\mathcal{L}$ including the set of default names and propositions formed by binary predicate $\prec$ applied to variables and default names only; these are mapped onto theories in the language $\mathcal{L}^\star$ obtained by extending $\mathcal{L}$ with new predicate symbols $(\cdot \not\prec \cdot)$, $\mathsf{ok}(\cdot)$, $\mathsf{bl}(\cdot)$, and $\mathsf{ap}(\cdot)$:

Definition 5.1 *Given a default theory (D,W) over $\mathcal{L}$ and its set of default names $N = \{n_\delta \mid \delta \in D\}$, we define $\mathcal{D}((D,W)) = (D',W')$ over $\mathcal{L}^\star$ by*

$$\begin{aligned} D' &= \left\{ \tfrac{\alpha \wedge \mathsf{ok}(n) : \beta}{\gamma \wedge \mathsf{ap}(n)}, \ \tfrac{\mathsf{ok}(n) : \neg\alpha}{\mathsf{bl}(n)}, \ \tfrac{\neg\beta \wedge \mathsf{ok}(n) :}{\mathsf{bl}(n)} \;\middle|\; n : \tfrac{\alpha : \beta}{\gamma} \in D \right\} \cup D_\prec \\ W' &= W \cup W_\prec \cup \{DCA_N, UNA_N\} \end{aligned}$$

where

$$\begin{aligned} D_\prec &= \left\{ \tfrac{: \neg(x \prec y), (x \prec y)}{(x \not\prec y)} \right\} \\ W_\prec &= \{\forall x \in N.\, \neg(x \prec x)\} \\ &\cup \ \{\forall xyz \in N.\, ((x \prec y) \wedge (y \prec z)) \supset (x \prec z)\} \\ &\cup \ \{\forall x \in N.\, (x \neq n_\top) \supset (x \prec n_\top)\} \\ &\cup \ \{\mathsf{ok}(n_\top)\} \\ &\cup \ \left\{\forall x \in N.\ \left(\forall y \in N.\, (x \not\prec y) \ \vee \ [(x \prec y) \supset (\mathsf{bl}(y) \vee \mathsf{ap}(y))]\right) \supset \mathsf{ok}(x)\right\} \end{aligned}$$

In contrast to Definition 4.1, D and W now may contain preference information expressed by $\prec$ applied to default names. The first three axioms in $W_\prec$ account for information that was implicitly provided by ordered default theories in the rigid case. The last axiom is a straightforward extension of that found in the rigid case, now also accounting for the information provided by the default rule in $D_\prec$.

Again, we observe that for ordered theory (D,W), the translation $\mathcal{D}((D,W))$ is only a constant factor larger than (D,W). We note that Theorem 4.1 and Theorem 4.2 carry over to the general case except for Theorem 4.1.1. We get instead

1$'$. either $n_\delta \prec n_{\delta'} \in E$ or $\neg(n_\delta \prec n_{\delta'}) \in E$ or $n_\delta \not\prec n_{\delta'} \in E$

In fact, ordered default theories are treated in the same way by our basic and general approach, except for different augmented languages:

Theorem 5.1 *Let $(D,W,<)$ be an ordered default theory over $\mathcal{L}$.*

For each extension E of $\mathcal{T}((D,W,<))$ there is an extension E' of $\mathcal{D}((D, W \cup \{n_\delta \prec n_{\delta'} \mid (\delta,\delta') \in <\}))$ such that $E \cap \mathcal{L} = E' \cap \mathcal{L}$ and vice versa.

Together with Corollary 4.2, this result implies that our dynamic approach yields all regular extensions (modulo the original language) if (D,W) does not contain an occurrence of $\prec$:

Theorem 5.2 *For a default theory (D,W) over a language $\mathcal{L}$ excluding $\prec$-symbols, and a set of formulas E, we have that*

E is an extension of (D,W) iff $E = E' \cap \mathcal{L}$ for some extension E' of $\mathcal{D}((D,W))$.

The notion of $<$-preservation is not directly applicable to the dynamic case. This is because there is no adequate counterpart of a 'regular extension of the original theory' since the preference information is only fully developed in the extensions of the image of the translation. Nonetheless, we can provide an analogous criterion being invariantly satisfied by all extensions obtained after our translation:

Theorem 5.3 *Let (D, W) be an ordered default theory and let E be a set of formulas.*

If E is a consistent extension of $\mathcal{D}((D, W))$ then we have for all grounded enumerations $\langle \zeta_i \rangle_{i \in I}$ of $GD(D, E)$ and for all $\delta, \delta' \in D$:

$$\textit{If } (n_\delta \prec n_{\delta'}) \in E \textit{ then } j < i \textit{ for all } \zeta_i = \delta_t \textit{ and some } \zeta_j = (\delta')_{t'} \textit{ with } t, t' \in \{a, b_1, b_2\}\ .$$

As argued above, there is no sensible correspondence to regular extensions in a dynamic setting since preferences are present in both the original and resulting theory. Thus, there is no counterpart for Corollary 4.1 in the dynamic case.

The advantage of dynamic preferences over static ones is that they allow for specifying *context-sensitive* preferences, where the context is spanned by the encompassing extension. In fact, the dynamic setting does not necessarily furnish "softer" preferences than obtainable in the static case. Dynamic preferences do rather provide additional means for canceling preferences in certain cases. Clearly, a theory like

$$\left(\{n_1 : \tfrac{:B}{B},\ n_2 : \tfrac{:\neg B}{\neg B},\ n_{21} : \tfrac{:n_1 \prec n_2}{n_1 \prec n_2}\}, \emptyset\right)$$

yields a single extension containing $\neg B$. Adding $n_2 \prec n_1$ as a fact results in a single extension with B. Adding instead a default rule such as $n_{12} : \frac{:n_2 \prec n_1}{n_2 \prec n_1}$ gives rise to two extensions, one with B and one with $\neg B$. The important thing to note is that the two last additions gave rise to alternative contexts, that is, in both cases $\neg(n_1 \prec n_2)$ was derivable, first in an overriding way (in W), then as included in an alternative extension.

If no such default alternative is provided, dynamic preferences are as rigid as static ones. To see this, let us reconsider theories (6) and (8) in a dynamic setting:

$$\left(\{n_1 : \tfrac{:B}{B},\ n_2 : \tfrac{B:C}{C},\ n_{21} : \tfrac{:n_1 \prec n_2}{n_1 \prec n_2}\}, \emptyset\right) \tag{10}$$

$$\left(\{n_1 : \tfrac{:B}{B},\ n_2 : \tfrac{:C \wedge \neg B}{C},\ n_{21} : \tfrac{:n_1 \prec n_2}{n_1 \prec n_2}\}, \emptyset\right) \tag{11}$$

As in the static case, the images of both theories do not possess any extensions. This is because there is simply no way to refute the preference imposed by the third default rule. If this was possible, we would get unwanted preferences, as argued at the start of this section when motivating $D_\prec$.

So, it should be clear that if one wants to suspend dynamic preferences in certain contexts, one has to provide a specification for these contexts. One way of doing this is to equip rules with dynamic preferences with predicates playing the same role as abnormality predicates in circumscription [McCarthy, 1986]. For implementing this option in the above examples, we could replace

$$n_{yx} : \frac{:n_x \prec n_y}{n_x \prec n_y} \qquad \text{by} \qquad p(y, x) : \frac{:(n_x \prec n_y) \wedge \neg \mathsf{ko}(p(y,x))}{n_x \prec n_y}$$

and supply a corresponding blocking policy, comparable to circumscription policies. In the case of (10), one could fix the problem by means of $\forall x, y \in N.\ \mathsf{ap}(n_x) \wedge \mathsf{ap}(n_y) \supset \mathsf{ko}(p(y,x))$, that is, by canceling preferences between non-conflicting rules. Alternately one could adopt a different strategy by putting:

$$\forall x, y \in N.\ (\mathit{Conseq}(\delta_x) \supset \mathit{Prereq}(\delta_y)) \supset \mathsf{ko}(p(y,x))\ .$$

This axiom ranks derivability over preference (see Section 4.2). One could similarly rank blockage over preference in addressing the lack of extensions in (11):

$$\forall x, y \in N.\ (\mathit{Conseq}(\delta_x) \supset \neg \mathit{Justif}(\delta_y)) \supset \mathsf{ko}(p(y,x))\ .$$

Of course, our axiomatic approach leaves room for many other policies.

5.1 Examples

To further illustrate our approach, we consider two extended examples, taken from the literature. The first is given by Junker in [1997]: [10]

> *"Jim and Jane have the following habits:*
>
> 1. *Normally, Jim and Jane go to at most one attraction when they go out in an evening.*
> 2. *Jim prefers the theatre to the night club.*
> 3. *Jane prefers the night club to the theatre.*
> 4. *If Jim invites Jane then he respects her preferences (and vice versa).*
> 5. *Normally Jim invites Jane.*
> 6. *An exception to 1. is Saturday.*
> 7. *An exception to 5. is Jim's birthday, where Jane invites Jim.*
>
> *If no further information is given we conclude that Jim and Jane will go to the night club. When we learn that Jim has birthday we revise this and conclude that they go to the theatre. However, the day in question is a Saturday. Hence, they should go to both attractions. Finally the news tells that the theatre is closed for work. Thus we again conclude that they go to the night club."*

We adapt Junker's modelling as follows:

$$n_{\mathit{theatre}} : \frac{:\ \mathit{goto}(\mathit{theatre}) \wedge \neg \mathit{closed}(\mathit{theatre})}{\mathit{goto}(\mathit{theatre})} \tag{12}$$

$$n_{\mathit{night\text{-}club}} : \frac{:\ \mathit{goto}(\mathit{night\text{-}club}) \wedge \neg \mathit{closed}(\mathit{night\text{-}club})}{\mathit{goto}(\mathit{night\text{-}club})} \tag{13}$$

$$n_{\mathit{evenings}} : \frac{:\ \neg(\mathit{goto}(\mathit{theatre}) \wedge \mathit{goto}(\mathit{night\text{-}club})) \wedge \neg \mathit{party\text{-}night}}{\neg(\mathit{goto}(\mathit{theatre}) \wedge \mathit{goto}(\mathit{night\text{-}club}))} \tag{14}$$

[10] A resemblance to existing persons is accidental and not in accord with the intention of the authors.

$$n_{jane} : \frac{invites(jim, jane) \ : \ n_{theatre} \prec n_{night\text{-}club}}{n_{theatre} \prec n_{night\text{-}club}} \tag{15}$$

$$n_{jim} : \frac{invites(jane, jim) \ : \ n_{night\text{-}club} \prec n_{theatre}}{n_{night\text{-}club} \prec n_{theatre}} \tag{16}$$

$$n_{invitation} : \frac{: \ invites(jim, jane)}{invites(jim, jane)} \tag{17}$$

$$saturday \supset party\text{-}night \tag{18}$$

$$birthday(jim) \supset invites(jane, jim) \tag{19}$$

$$invites(jim, jane) \supset \neg invites(jane, jim) \tag{20}$$

$$closed(theatre) \supset \neg goto(theatre) \tag{21}$$

$$\forall x \in N. \ (goto(x) \supset (n_x \prec n_{evenings})) \tag{22}$$

$$\forall x \in N. \ (goto(x) \supset (n_x \prec n_{invitation})) \tag{23}$$

Recall that n_x is simply an abbreviation for $n(x)$. The choice where to go, is formalised by means of default rules $\delta_{theatre}$ and $\delta_{night\text{-}club}$. Both spots are supposed to be open, unless they are known to be closed. Observe that both δ_{jane} and δ_{jim} model a combination of above statements 2. and 3. with 4., respectively. So, δ_{jane} tells us that Jane's preferences hold since Jim is inviting her. Finally, following [Junker, 1997], we give in (22/23) preference to rules $\delta_{evenings}$ and $\delta_{invitation}$, talking *about* where to go, over $\delta_{theatre}$ and $\delta_{night\text{-}club}$ simply indicating where to go.[11]

First of all, it is easy to see that default theory

$$(D, W) = (\ \{(12) - (17)\} \ , \ \{(18) - (23)\} \ ) \tag{24}$$

has three extension in standard default logic, one containing $goto(theatre)$, one containing $goto(night\text{-}club)$, and one including both.

Clearly, the application of the rules in (14)–(17) is not subject to any preferences, so that any extension must contain:

$$\mathsf{ap}(n_{evenings}), \ \mathsf{ap}(n_{jane}), \ \mathsf{ap}(n_{jim}), \text{ and } \mathsf{ap}(n_{invitation}) \ .$$

This is different for the rules in (12) and (13), whose application depends on the context spanned by the encompassing extension. Because of (22) and (23), the rules $\delta_{theatre}$ and $\delta_{night\text{-}club}$ depend both on the applicability of $\delta_{evenings}$ and $\delta_{invitation}$. Their interdependency, however, is subject to the presence of corresponding preference literals, like $n_{theatre} \not\prec n_{night\text{-}club}$ and $n_{theatre} \prec n_{night\text{-}club}$ in the case of $\delta_{theatre}$. This is reflected by the

[11] This is because the former influence the latter but not vice versa.

following two formulas, common to all extension construction processes:

$$
\begin{array}{lllll}
(& [\,(n_{\mathit{theatre}} \not\prec n_{\mathit{night\text{-}club}}) & & & \\
& \quad\vee & & & \\
& (n_{\mathit{theatre}} \prec n_{\mathit{night\text{-}club}} & \supset & \mathsf{ap}(n_{\mathit{night\text{-}club}}) \vee \mathsf{bl}(n_{\mathit{night\text{-}club}}))\,] & \\
\wedge & & & [\,\mathsf{ap}(n_{\mathit{evenings}}) \vee \mathsf{bl}(n_{\mathit{evenings}})\,] & \\
\wedge & & & [\mathsf{ap}(n_{\mathit{invitation}}) \vee \mathsf{bl}(n_{\mathit{invitation}})]\ \ ) & \supset\ \ \mathsf{ok}(n_{\mathit{theatre}})
\end{array}
$$

$$
\begin{array}{lllll}
(& [\,(n_{\mathit{night\text{-}club}} \not\prec n_{\mathit{theatre}}) & & & \\
& \quad\vee & & & \\
& (n_{\mathit{night\text{-}club}} \prec n_{\mathit{theatre}} & \supset & \mathsf{ap}(n_{\mathit{theatre}}) \vee \mathsf{bl}(n_{\mathit{theatre}}))\,] & \\
\wedge & & & [\,\mathsf{ap}(n_{\mathit{evenings}}) \vee \mathsf{bl}(n_{\mathit{evenings}})\,] & \\
\wedge & & & [\mathsf{ap}(n_{\mathit{invitation}}) \vee \mathsf{bl}(n_{\mathit{invitation}})]\ \ ) & \supset\ \ \mathsf{ok}(n_{\mathit{night\text{-}club}})
\end{array}
$$

The image $\mathcal{D}((D,W))$ of theory (D,W), given in (24), leads to an extension containing $\mathit{invites}(\mathit{jim},\mathit{jane})$ and therefore also $n_{\mathit{theatre}} \prec n_{\mathit{night\text{-}club}}$ and $\neg(n_{\mathit{night\text{-}club}} \prec n_{\mathit{theatre}})$. Because of $\mathsf{ap}(n_{\mathit{invitation}})$ and $\mathsf{ap}(n_{\mathit{evenings}})$, we then get $\mathsf{ok}(n_{\mathit{night\text{-}club}})$ and $(\mathsf{ap}(n_{\mathit{night\text{-}club}}) \vee \mathsf{bl}(n_{\mathit{night\text{-}club}})) \supset \mathsf{ok}(n_{\mathit{theatre}})$. As a result, we obtain $\mathit{goto}(\mathit{night\text{-}club})$ and $\neg\mathit{goto}(\mathit{theatre})$.

When we learn that it happens to be Jim's birthday, thus adding $\mathit{birthday}(\mathit{jim})$, we get an extension containing $\mathit{invites}(\mathit{jane},\mathit{jim})$ and therefore also $n_{\mathit{night\text{-}club}} \prec n_{\mathit{theatre}}$ and $\neg(n_{\mathit{theatre}} \prec n_{\mathit{night\text{-}club}})$. Analogously, we then obtain $\mathit{goto}(\mathit{theatre})$ and $\neg\mathit{goto}(\mathit{night\text{-}club})$.

Learning furthermore that Jim's birthday falls on a Saturday makes it a real party-night. In fact, we conclude from *saturday* that it's *party-night*, which blocks default rule $\delta_{\mathit{evenings}}$, and we end up with an extension containing both $\mathit{goto}(\mathit{theatre})$ and $\mathit{goto}(\mathit{night\text{-}club})$. Note that this is concluded in the presence of $\mathit{invites}(\mathit{jane},\mathit{jim})$ and $n_{\mathit{night\text{-}club}} \prec n_{\mathit{theatre}}$.

Finally, the news tells us that the theatre is closed, $\mathit{closed}(\mathit{theatre})$; this blocks default rule $\delta_{\mathit{theatre}}$ and we only conclude $\mathit{goto}(\mathit{night\text{-}club})$, as above, despite the presence of $n_{\mathit{night\text{-}club}} \prec n_{\mathit{theatre}}$.

Next, we consider an example from [Gordon, 1993], discussed in [Brewka, 1994b]:

> *"A person wants to find out if her security interest in a certain ship is 'perfected', or legally valid. This person has possession of the ship, but has not filed a financing statement. According to the code UCC, a security interest can be perfected by taking possession of the ship. However, the federal Ship Mortgage Act (SMA) states that a security interest in a ship may only be perfected by filing a financing statement. Both UCC and SMA are applicable; the question is which takes precedence here. There are two legal principles for resolving such conflicts.* Lex Posterior *gives precedence to newer laws; here we have that UCC is more recent than SMA. But* Lex Superior *gives precedence to laws supported by the higher authority; here SMA has higher authority since it is federal law."*

Apart from $\delta_\top$, we obtain the following default rules:

$$
\mathit{ucc} : \frac{\mathit{possession}\ :\ \mathit{perfected}}{\mathit{perfected}}\ , \qquad \mathit{sma} : \frac{\mathit{ship} \wedge \neg\mathit{finstmt}\ :\ \neg\mathit{perfected}}{\neg\mathit{perfected}}\ ,
$$

$$
\mathit{lp}(x,y) : \frac{\mathit{newer}(y,x)\ :\ x \prec y}{x \prec y}\ , \qquad \mathit{ls}(x,y) : \frac{\mathit{statelaw}(x) \wedge \mathit{fedlaw}(y)\ :\ x \prec y}{x \prec y}\ .
$$

To preserve finiteness, we restrict our attention to name set $N = \{n_\top\} \cup N_0 \cup N_1$ where $N_0 = \{ucc, sma\}$ and $N_1 = \{lp(x,y), ls(x,y) \mid x,y \in N_0\}$, and the corresponding default instances. We have the facts:

$$\begin{array}{l} possession,\ ship,\ \neg finstmt, \\ newer(ucc, sma),\ fedlaw(sma),\ statelaw(ucc), \\ \forall x,y,u,v \in N_0.\ lp(x,y) \prec ls(u,v)\ . \end{array}$$

From this specification, we obtain a single extension, $E \supseteq \{\neg perfected, ucc \prec sma\}$. We obtain $\forall xy \in N_0.\ \mathsf{ok}(ls(x,y))$. In E we get $\mathsf{ok}(ls(ucc,sma))$ while

$$\mathsf{bl}(ls(ucc,sma)) \vee \mathsf{ap}(ls(ucc,sma)) \supset \mathsf{ok}(lp(sma,ucc)).$$

(All other instances of these axioms are eliminated by deriving $x \not\prec y$.) We then conclude by $ls(ucc,sma)$ that $ucc \prec sma$. This blocks $lp(sma,ucc)$ since its justification $sma \prec ucc$ has become refuted. Thus, $sma \not\prec ucc \in E$ yielding $\mathsf{ok}(sma)$ and subsequently $\neg perfected$.

[Brewka, 1994b] solves this problem by first generating 4 entire extensions, where $E_1 \supseteq \{perfected, sma \prec ucc\}$, $E_2 \supseteq \{\neg perfected, sma \prec ucc\}$, $E_3 \supseteq \{perfected, ucc \prec sma\}$, $E_4 \supseteq \{\neg perfected, ucc \prec sma\}$. In a second step he rules out E_1, E_2, E_3 since they do not verify a certain priority criterion. The remaining extension, E_4 is after all the one obtained in our approach.

6 Further extensions

An axiomatic approach to preferences offers a highly flexible framework for specifying preferences. For instance, a more fine-grained approach is to distinguish the source of blockage by replacing δ_{b_1} and δ_{b_2} by

$$\frac{\mathsf{ok}(n_\delta)\ :\ \neg\alpha}{\mathsf{bl}_p(n_\delta)} \quad \text{and} \quad \frac{\neg\beta \wedge \mathsf{ok}(n_\delta)\ :\ }{\mathsf{bl}_j(n_\delta)}, \quad \text{respectively.}$$

Accordingly, we would obtain in $W_\prec$ the axiom

$$\forall x \in N.\, [\forall y \in N.\, (x \prec y) \supset (\mathsf{bl}_p(y) \vee \mathsf{bl}_j(y) \vee \mathsf{ap}(y))] \supset \mathsf{ok}(x)\ .$$

We use such an encoding in [Delgrande and Schaub, 2000] where we argue that property inheritance comprises a mechanism distinct from preference.[12]

Two different substantive extensions are discussed in the remainder of this section.

6.1 Expressing generalised preferences

An important generalisation of our notion of *preference*, expressed in Section 3 is the following.

[12] Thus one would encode that it is ok to apply a rule just if all $<$-greater rules are blocked via failure to prove the antecedent. We do not develop this mechanism here since it would take us too far from our primary interest, preference.

Generalised Preference For preferences $\delta_1 < \ldots < \delta_m$, apply δ_m if possible; apply δ_{m-1} if possible, continue in this fashion until no more than k (for fixed k where $1 \leq k \leq m$) defaults have been applied.

An example is where a student wishes to take $k = 3$ courses out of $m = 10$ possible courses, and so provides a list of preferences over the courses. There are two important subcases corresponding to $k = 1$ and $k = m$. In the first case a maximum of one default is applied. In the second case one attempts to apply every default.

Given the predicates $\mathsf{bl}(\cdot)$, $\mathsf{ap}(\cdot)$, and notably $\mathsf{ko}(\cdot)$, it is a straightforward matter to assert that a maximum of k default rules in a priority order can be applied. We modify the definition of δ_a (cf. (5)), by setting

$$\delta_a = \frac{\alpha \wedge \mathsf{ok}(n_\delta) \;:\; \beta \wedge \neg\mathsf{ko}(n_\delta)}{\gamma \wedge \mathsf{ap}(n_\delta)}\;.$$

In addition we add the following axiom to the initial set of facts:

$$\forall x_1, \ldots, x_k \in N.\ \Big([\bigwedge_{i \neq j; i,j=1..k} (x_i \neq x_j) \supset \bigwedge_{i=1..k} \mathsf{ap}(x_i)] \supset \forall x \in N.\ [\bigwedge_{i=1..k} (x \neq x_i) \supset \mathsf{ko}(x)]\Big)$$

We abbreviate $\neg(x = y)$ by $x \neq y$. This axiom states that if k distinct rules are applied, then all remaining rules are ko'ed. For coherence, it is furthermore convenient to supply a statement to the effect that all rules are considered in turn, although some of them have been "knocked out". This can be done with $\forall x \in N.\ \mathsf{ko}(x) \supset \mathsf{bl}(x)$.

For illustration, consider a student who wishes to take three courses, and has a preference ordering on the ten available courses. So depending on the prerequisites and what courses are still open (i.e. aren't fully subscribed) the student's preferences are satisfied as far as possible. Taking a variable s for students and another c_i for courses, we get:

$$\frac{\mathit{prerequisite}(s, c_i) \;:\; \mathit{open}(c_i) \wedge \neg\mathit{canceled}(c_i)}{\mathit{subscribe}(s, c_i)}$$

Our student has the following preferences: c_5 is her most preferred course; she wants to avoid c_4; she prefers c_6 over c_2, if c_5 isn't open anymore; and finally she prefers c_6 over c_3 and c_5, unless *Smith* is giving the course:

$$\begin{array}{lll} c_i \prec c_5,\ i = 1..4, 6..10, & \neg\mathit{open}(c_5) \supset c_6 \prec c_2, & \frac{:\, \mathit{lecturer}(c_6) \neq \mathit{Smith}}{(c_3 \prec c_6) \wedge (c_7 \prec c_6)} \\ c_4 \prec c_j,\ j = 1..3, 5..10, & & \end{array}$$

Assume that our student fulfills the prerequisites for all but course 9 and 10, and that courses 1,3, and 8 are already fully subscribed. Fortunately, *Smith* is on sabbatical.

For $k = 3$ and $C = \{c_i \mid i = 1..10\}$, we get

$$\forall x_1, x_2, x_3 \in C.\ \Big([(x_1 \neq x_2) \wedge (x_1 \neq x_3) \wedge (x_2 \neq x_3) \supset \mathsf{ap}(x_1) \wedge \mathsf{ap}(x_2) \wedge \mathsf{ap}(x_3)] \\ \supset \forall y \in C.\ [(y \neq x_1) \wedge (y \neq x_2) \wedge (y \neq x_3) \supset \mathsf{ko}(y)]\Big)$$

Applying our dynamic translation $\mathcal{D}$ along with the above modifications to the corresponding default theory yields, after a few iterations in Definition 2.1, among others:

$$
\begin{array}{rcl}
 & & \mathsf{ok}(n_5), \\
(\mathsf{ap}(n_5) \vee \mathsf{bl}(n_5)) & \supset & \mathsf{ok}(n_i), \\
[(\mathsf{ap}(n_6) \vee \mathsf{bl}(n_6)) \wedge (\mathsf{ap}(n_5) \vee \mathsf{bl}(n_5))] & \supset & \mathsf{ok}(n_j), \\
[(\bigwedge_{j\in J} \mathsf{ap}(n_j) \vee \mathsf{bl}(n_j)) \wedge (\bigwedge_{i\in I} \mathsf{ap}(n_i) \vee \mathsf{bl}(n_i)) \wedge (\mathsf{ap}(n_5) \vee \mathsf{bl}(n_5))] & \supset & \mathsf{ok}(n_4),
\end{array}
$$

for $i \in I = \{1, 2, 6, 8, 9, 10\}$ and $j \in J = \{3, 7\}$. Taking into account the above constraints, we get that c_5 must be taken first, followed by c_2 or c_6 or c_7 (but c_6 before c_7) and finally c_4.

As a result, we obtain two extensions: one containing c_5, c_6, and c_2 and another one containing c_5, c_6, and c_7. Neither of them contains c_4 nor is there an extension containing c_5, c_2, and c_7 due to $c_7 \prec c_6$. If c_5 turned out to be overbooked, we would get a single extension containing c_2, c_6, and c_7. If additionally c_7 is canceled, we get c_2, c_6, and c_4. That is, our student is finally obliged to take c_4. Otherwise, if it turns out that *Smith* keeps lecturing c_6 despite his sabbatical, she is faced with three alternatives: c_5, c_6, c_2 or c_5, c_6, c_7 or c_5, c_2, c_7.

6.2 Preferences among Sets of Defaults

Preferences also may apply to sets of defaults. Consider for example preferences in buying a car, specifically a situation in which one ranks the price (E) over safety features (S), and safety features (S) over power (P), but safety features together with power is ranked over price. In an obvious extension of preference to sets of defaults, we can write this as:[13]

$$\left\{\tfrac{:P}{P}\right\} < \left\{\tfrac{:S}{S}\right\} < \left\{\tfrac{:E}{E}\right\} < \left\{\tfrac{:P}{P}, \tfrac{:S}{S}\right\}. \tag{25}$$

Intuitively, if we were given only that not all desiderata can be satisfied (i.e. $W = \{\neg(P \wedge E \wedge S)\}$) then we could apply the defaults in the highest-ranked set and conclude that P and S can be met.[14] This approach is described in [Delgrande and Schaub, 1998], and a related approach and implementation in the framework of extended logic programs is described in [Delgrande *et al.*, 2000a].

We omit the details here, but the overall methodology is the same as for static and dynamic preferences. In the set-based approach, a default theory now has an ordering given on sets of defaults. As before, we take an ordered theory and translate it into a standard default theory. Consider a general assertion $D' < D''$ where $D', D'' \subseteq D$. Informally we prefer the application of the set D'' to that of D'. We can say that D'' is applicable if all its member defaults are, and inapplicable if one of its members is inapplicable. Consequently we consider D' after *all* defaults in D'' are found to be applicable, or *some* default in D'' is found to be inapplicable.

To do this, we extend our set of names so that, in addition to names for individual default rules, we also have names for the sets of rules mentioned in the ordering over sets,

[13] To be sure, this is a naïve encoding; see [Brewka and Gordon, 1994] for a more realistic formalisation.

[14] Note that we cannot simply replace a set of defaults with a default consisting of the conjunction of the respective prerequisites, justifications, and consequents of defaults in the set, since this doesn't allow for rule interactions. For example by this scheme the set $\left\{\frac{:P}{P}, \frac{P:S}{S}\right\}$ would be replaced by the meaningless default $\frac{P\,:\,P\wedge S}{P\wedge S}$.

$<$. Roughly speaking, for a set with name m, if it is $\mathsf{ok}(m)$ to apply a set of rules then it is ok to apply the individual rules in the set. If $\mathsf{ap}(n_i)$ is true for every rule δ_i in a set, then the set is flagged as applied by $\mathsf{ap}(m)$. If $\mathsf{bl}(n_i)$ is true for some rule δ_i in a set, then the set is so flagged by $\mathsf{bl}(m)$. Finally, for a given set, if every $<$-greater set is applied or blocked then it is ok to apply the set.

However, there is a problem with "side-effects" in a naïve implementation of this approach. Assume in the example (25) that P and S cannot be jointly met (i.e. $W = \{\neg(P \wedge S)\}$). We would expect that there would be a single extension containing E and S. In a naïve implementation, one would try to apply the defaults in the highest-ranked set. On applying the default $\frac{:P}{P}$ it would prove to be the case that $\frac{:S}{S}$ could not be applied, or vice versa. So we would find that the topmost nontrivial set isn't applicable. However, in finding that defaults in the highest ranked set cannot all be applied, we do not want to actually apply the default $\frac{:P}{P}$, since this will lead to an incorrect result: if the instance of $\frac{:P}{P}$ in this set were applied, then we would next try to apply the default in the next set (viz. $\frac{:E}{E}$), which would be successful, but then, given P, we couldn't apply the default in the following set, viz. $\frac{:S}{S}$. We would then (incorrectly) obtain an extension containing $\{P, E\}$. So in determining that the defaults in a set cannot all be applied, we must avoid the side-effect where some of these defaults are in fact applied. We do this by detecting when a set of defaults is going to be blocked. This will occur just when the negation of a prerequisite is consistent (with the final extension), or if the set of consequents denies the justification of a default in a set. In either of these cases, the set as a whole is blocked, and there are no side effects that propagate to lower ranked sets. Consult [Delgrande and Schaub, 1998] for details.

7 Related Work

In Section 3 we argued that there is no single notion for treating prioritised information. Of work in default logic, we have argued that [Reiter and Criscuolo, 1981; Etherington and Reiter, 1983; Delgrande and Schaub, 1994] address property inheritance (as exemplified by Example (3)). In particular, these approaches are based on the idea of resolving conflicts by appeal to specificity information. Hence, for non-conflicting rules they may produce inappropriate results when used for preferences. For instance, take $\frac{A:B}{B} < \frac{C:D}{D}$ along with $A, C, \neg D$. While one expects a single extension containing B, the aforecited approaches would replace the first rule either by $\frac{A:B\wedge\neg C}{B}$ or $\frac{A:B\wedge(C\supset D)}{B}$, respectively, neither of which would be applicable for providing B.

For preference, there are *descriptive* and *prescriptive* interpretations. In the former case, one has a "wish list" where the intent is that one way or another the highest-ranked defaults be applied. In the latter case the ordering reflects the order in which defaults should be applied.

[Rintanen, 1995] addresses descriptive preference orders in normal default theories (this despite the paper's title and examples, which would indicate that the paper deals with property inheritance). An order on extensions is defined as follows. A default rule $\frac{A:B}{B}$ is *applied* in extension E just if $A, B \in E$. Extension E is preferred over E' iff there is $\delta \in D$ applied in E but not in E' such that if δ' is preferred over δ and δ' is applied in E' then it is also applied in E. While this paper addresses a different notion of preference than ours, it is worth noting two sources of "inefficiency" in Rintanen's approach not present in ours. First, preference on extensions is given in terms of a total order on preferences among rules; consequently, given a partial order on rules, all total orders that preserve the original partial order must be generated. Second, all extensions are generated, and then the preferred extension is found via the "filtering" mechanism. Consider an ordered theory where we just have the preference $\frac{:B(x)}{B(x)} < \frac{:A(x)}{A(x)}$ along with $\forall x(\neg A(x) \vee \neg B(x))$. Clearly the number of total orders resulting from this partial order will be exponential in the number of instances of A and B. Similarly the number of extensions in the unordered theory will be exponential in the number of instances. In contrast, in our approach a single extension is produced, in a translated theory that is only a constant factor larger than the original.

For prescriptive approaches, [Baader and Hollunder, 1993a] and [Brewka, 1994a] present prioritised variants of default logic in which the iterative specification of an extension is modified. A default is only applicable at an iteration step (cf. Definition 2.1) if no $<$-greater default is applicable.[15] The primary difference between these approaches rests on the number of defaults applicable at each step. While Brewka allows only for applying a single default that is maximal with respect to a total extension of $<$, Baader and Hollunder allow for applying all $<$-maximal defaults at each step. In contrast we translate priorities into standard default theories.

As a first distinguishing example, consider the normal default rules

$$\delta_1 : \frac{:A}{A}, \quad \delta_2 : \frac{:B}{B}, \quad \delta_3 : \frac{B:C}{C}, \quad \delta_4 : \frac{A:\neg C}{\neg C}$$

along with $\delta_4 < \delta_3$, taken from [Baader and Hollunder, 1993b]. With no facts Baader and Hollunder obtain one extension containing $\{A, B, C\}$. Curiously, Brewka obtains an additional extension containing $\{A, B, \neg C\}$. In our approach, the resultant default theory yields only the first extension containing $\{A, B, C\}$. So here our approach yields the same result as Baader and Hollunder's.

As a second example, again from [Baader and Hollunder, 1993b], consider the rules

$$\delta_1 : \frac{:A}{A}, \quad \delta_2 : \frac{B:\neg A}{\neg A}, \quad \delta_3 : \frac{:B}{B}, \quad \delta_4 : \frac{A:\neg B}{\neg B}$$

along with $\delta_1 < \delta_2$, $\delta_3 < \delta_4$. Baader and Hollunder show that in Brewka's approach two extensions are obtained, one containing $\{A, \neg B\}$ and another containing $\{\neg A, B\}$. However an additional extension is obtained in Baader and Hollunder's approach, containing $\{A, B\}$. Our approach yields only the first two extensions. So, as opposed to the previous example, our approach yields here the same result as Brewka's approach. In all, we observe that in both examples our approach yields the fewer and arguably more intuitive extensions.

Neither Brewka nor Baader and Hollunder deal with context-sensitive preferences or sets of preferences. In addition, in our approach we translate preferences into standard default theories. [Junker, 1997] addresses static and dynamic preferences in a more restricted framework that roughly corresponds to ordered prerequisite-free normal default theories. In a preliminary report, [Brewka and Gordon, 1994] consider sets of preferences, but not in default logic.

[15] These authors use $<$ in the reverse order from us.

[Brewka and Eiter, 1998] address (static) preference in extended logic programs; this is extended to default logic in [Brewka and Eiter, 2000] . In common with previous work, Brewka and Eiter begin with a partial order on a rule base, but define preference with respect to total orders that conform to the original partial order. As well, answer sets or extensions, respectively, are first generated and the "preferred" answer sets (extensions) are selected subsequently. In contrast, in our approach, we deal only with the original partial order, which is translated into the object theory. As well, only "preferred" extensions are produced in our approach; there is no need for meta-level filtering of extensions.

We have the following result relating the approach of [Brewka and Eiter, 2000] with the instance of our framework captured by translation $\mathcal{T}$:

Theorem 7.1 *Let $(D, W, <)$ a normal, prerequisite-free, ordered default theory over $\mathcal{L}$.*

For each extension E of $\mathcal{T}((D, W, <))$, there is an extension E' of $(D, W, <)$ according to [Brewka and Eiter, 2000] such that $E = E' \cap \mathcal{L}$ and vice versa.

To see that this result does not extend to prerequisite-free theories, consider

$$\left(\left\{\ n_1 : \tfrac{:B}{B}, n_2 : \tfrac{:A}{A}, n_3 : \tfrac{:\neg B}{A}\ \right\}, \emptyset, \{\delta_i < \delta_j \mid i < j\}\right) \tag{26}$$

This theory has a single regular extension, containing A and B. Observe that the theory obtained by applying translation $\mathcal{T}$ yields no preferred extension due to the interaction among blockage and preference, described in Section 4.2. This differs from the approach of Brewka and Eiter in [2000], who accept the above extension as a preferred one.[16] This is done by imposing an additional fixed-point condition: Given an ordered default theory $(D, W, <)$ and a regular extension E, the preferredness of E is established with respect to the theory

$$(D \setminus \{\delta \mid \mathit{Conseq}(\delta) \in E \text{ and } \neg \mathit{Justif}(\delta) \in E\}, W, <).$$

This turns theory (26) into $\left(\left\{\ n_1 : \tfrac{:B}{B}, n_2 : \tfrac{:A}{A}\ \right\}, \emptyset, \{\delta_1 < \delta_2\}\right)$, whose only preferred extension contains A and B (see Theorem 7.1).

Theorem 7.1 does not extend to normal default theories, either. Observe that (6) yields a preferred extension, $Th(\{A, B\})$, in the approach of Brewka and Eiter, while ours does not due to the interaction among groundedness and preference. Such an interaction is avoided in [Brewka and Eiter, 2000] by another fixed-point construction eliminating prerequisites that belong to the extension and eliminating rules whose prerequisites do not belong to the extension.

In all, our initial approach captured by $\mathcal{T}$ can be seen as refining the set of extensions obtained in [Brewka and Eiter, 2000]:

Theorem 7.2 *Let $(D, W, <)$ be an ordered default theory over $\mathcal{L}$.*

For each extension E' of $\mathcal{T}((D, W, <))$ there is a preferred extension E of $(D, W, <)$ according to [Brewka and Eiter, 2000] such that $E = E' \cap \mathcal{L}$.

[16] We note that theory (8), used to illustrate the interaction between preferences and blockage relations, yields also no preferred extension in the approach of [Brewka and Eiter, 2000].

To see that such a refinement makes sense, consider the following example due to Baader and Hollunder [1993a]:

$$\left(\left\{\ n_1 : \frac{Penguin : \neg Fly}{\neg Fly}, n_2 : \frac{Bird : Winged}{Winged}, n_3 : \frac{Winged : Fly}{Fly}, n_4 : \frac{Penguin : Bird}{Bird}\ \right\}, \{Penguin\}, \{\delta_2 < \delta_1\}\right)$$

While this theory has a single extension in our approach, containing $\neg Fly$, $Bird$, and $Winged$, the approach of Brewka and Eiter yields an additional extension with Fly.

The above discussion was dominated by the comparison between our specific translation $\mathcal{T}$ and the approach of Brewka and Eiter. In fact, our overall framework is general enough to express the strategy for preference handling proposed in [Brewka and Eiter, 2000]. This instance of our framework is described in [Delgrande *et al.*, 2000b] and is omitted here for brevity. Lastly we note that Brewka and Eiter begin with two "principles" that in their view provide meaning postulates for the term "preference", and so should be satisfied by any approach dealing with preference. We show in [Delgrande *et al.*, 2000a] that our approach also satisfies both principles.[17] [18]

Among the various approaches to preferences in (extended) logic programming, a central role is played by the approach in [Gelfond and Son, 1997] because unlike others it avoids defining a new nonmonotonic formalism in order to cope with preference information. Instead, Gelfond and Son introduce a special-purpose language for directly encoding preferences in a logic programming setting.[19] As with our framework, this approach offers a variety of different preference handling instances. Unlike us, however, Gelfond and Son pursue a "two-level" approach in reifying rules and preferences. For example, a rule like $\frac{r \wedge \neg s : \neg q}{p}$, or $p \leftarrow r, \neg s, not\ q$ in terms of logic programming, is expressed by the formula (or after reification by the corresponding *term* inside a *holds*-predicate, respectively) $default(n, p, [r, \neg s], [q])$ where n is the name of the rule. The semantics of a domain description is defined in terms of a set of domain-independent rules for predicates like *holds*. These rules can be regarded as a meta-interpreter for the domain description. Interestingly, the approach is based on the notion of "defeat" (of justifications) in contrast to an order-preserving consideration of rules, as found in our approach. Also, the specific strategy elaborated upon in [Gelfond and Son, 1997] differs from the ones considered in this paper in that it *"stops the application of default d_2 if defaults d_1 and d_2 are in conflict with each other and the default d_1 is applicable."* [Gelfond and Son, 1997]. (We consider such strategies in a companion paper [Delgrande and Schaub, 2000].) For detecting such conflicts, however, the approach necessities an extra conflict-indicating predicate. That is, one must state explicitly $conflict(d_1, d_2)$ to indicate that d_1 and d_2 conflict.

[17]To be precise, we do this in the framework of extended logic programming. The underlying proofs lift to default logic in a straightforward way.

[18][Brewka and Eiter, 1998] claim, erroneously, that the present approach violates their Principle I, based on their claim that we have extension $Th(\{A, \neg B\})$ for Example (4). However our approach in this case has no extension; see the discussion preceding Theorem 4.5.

[19]The chosen setting, viz. answer set semantics, corresponds to default logic on the fragment of extended logic programs [Gelfond and Lifschitz, 1991].

8 Discussion

We have described a very general framework for incorporating preferences into default logic. Via the naming of defaults we allow preferences to appear arbitrarily in D and W in a default theory. This allows preferences among preferences, preferences by default, preferences holding only in certain contexts, and so on. Given that such preferences are axiomatised using standard default logic, this approach may be regarded as providing a formalisation of a notion of "preference". The intuition on which the approach is based is clear: that a preference order specifies the order in which defaults are to be taken into account to see whether they are applicable. We argue that such a prescriptive approach is preferable to a descriptive approach, both from the point of view of representational force, as well as (pragmatic) computational considerations.

In the base approach, we are given a standard Reiter default theory together with a strict partial order on the defaults. We also allow preference information to be expressed within a default theory, so that preference information can appear in the world knowledge W or in the defaults D. In [Delgrande and Schaub, 1998] we show how preferences on sets of defaults can be similarly handled. In all cases we translate the default theory into second, standard default theory without explicit preferences, but in which defaults are applied according to the given ordering. Notably in all cases the translated theory is larger than the original by only a constant factor.

We prove that the defaults are indeed applied in the appropriate order. As well, we show that we have developed a set of strict generalisations in each of the elaborations to the basic approach, in that in a preference-based theory with no preferences, the translated theory (modulo the language) gives the same result as in classical default logic. In addition to the formal results, we illustrate the generality of the approach by formalising examples due to Junker and Gordon, of context-based preference, canceling preferences, preferences among preferences, and preferences by default. As well, elsewhere we show that we can capture the approach of [Brewka and Eiter, 2000].

It might be argued that, given complete information about preferences, such generality may not be required: [Doyle and Wellman, 1991], building on work by Arrow, argue that in any preference-based default theory, for coherence, one requires a "dictator" to adjudicate preferences. That is, in a complete system there must be, essentially, some way of determining a unique, complete, priority ordering. So in this sense, all one needs is what we have called the rigid approach of Section 4. We provide the more general framework of Section 5 for several reasons. First, in most realistic scenarios, one may not have complete information regarding preferences, and so a complete adjudication of preferences may not be possible. In this case one would obtain multiple extensions, representing possible "completions" of an incomplete ordering. Second, it allows the more flexible specification of preferences, leaving it up to the user to ensure that there is no ambiguity in preferences. This is well illustrated by Gordon's legal example (see below also) where a natural expression of interacting types of laws uses preferences among preferences.

Our approach of translating priorities into standard default theories has several advantages over previous work, which is phrased at the meta-level or alters the machinery of standard default logic. First, for a translated default theory, in our approach any extension is "preferred", in the sense that *only* "preferred" extensions (as specified by the ordering on rules) are produced. In contrast, previous approaches, in one fashion or another, must select among extensions for the most preferred. Hence one could expect the present approach to be (pragmatically) more efficient, since it avoids the generation of unnecessary extensions.

Another advantage of this axiomatic approach is that it allows us to formalise preference within the object theory. In particular this allows one to combine several orderings inside the same framework, and to specify in the theory how they interact. For instance, in legal reasoning different principles prefer different laws. We saw how *Lex Posterior* gives precedence to newer laws, while *Lex Superior* gives precedence to laws supported by the higher authority and finally *Lex Specialis* gives precedence to more specific laws. For reasoning with and about such principles, an explicit representation seems to be advantageous. For instance, we can state that *Lex Specialis* applies unless *Lex Posterior* or *Lex Superior* denies the precedence:

$$\frac{(x \prec_{specialis} y) \;:\; \neg(y \prec_{posterior} x), \neg(y \prec_{superior} x)}{x \prec y}$$

Also, generally, authority takes precedence over time. However, here too there may be exceptions, so that we must account for this by a default rule:

$$\frac{(x \prec_{superior} y) \wedge (y \prec_{posterior} x) \;:\; x \prec y}{x \prec y}.$$

The axiomatic approach makes it easier to compare differing approaches to handling different preference orderings or types of orderings, since we remain within the same "base" framework. That is, default logic provides a powerful tool with which to express such orderings. As a result also, our approach can be immediately implemented by making use of existing default logic theorem provers, since a preference based theory can be translated in a straightforward and efficient way into default logic. Lastly, by "compiling" preferences into default logic, and in using the standard machinery of default logic, we obtain insight into the notion of preference orderings. Thus, and as a point of theoretical interest, we show that incorporating explicit priorities among sets of rules in default logic in fact provides no real increase in the expressibility of default logic.

Lastly, our approach has been implemented under the syntactic restriction of extended logic programming and serves as a front-end to the logic programming systems `dlv` [Eiter *et al.*, 1997] and `smodels` [Niemelä and Simons, 1997]. The current prototype is available at

`http://www.cs.uni-potsdam.de/~torsten/plp/` .

This URL also contains examples taken from the literature, including those discussed in this paper. Both the dynamic approach to (single) preferences and a set-based approach have been implemented. Details on this implementation can be found in [Delgrande *et al.*, 2000a].

A Proofs of Theorems

The following definition is drawn upon in the following proofs.

Definition A.1 *Let (D, W) be a default theory. For any set of formulas S, let $\Gamma(S)$ be the smallest set of formulas S' such that*

1. *$W \subseteq S'$,*
2. *$Th(S') = S'$,*
3. *For any $\frac{\alpha:\beta}{\gamma} \in D$, if $\alpha \in S'$ and $\neg\beta \notin S$ then $\gamma \in S'$.*

A set of formulas E is an extension of (D, W) if $\Gamma(E) = E$.

Proof 4.1

1. By consistency of E, we cannot have both $n_\delta \prec n_{\delta'} \in E$ and $\neg(n_\delta \prec n_{\delta'}) \in E$.

   Assume that for some $\delta, \delta' \in D$, we have neither $n_\delta \prec n_{\delta'} \in E$ nor $\neg(n_\delta \prec n_{\delta'}) \in E$. Then, however, the default rule $\frac{:\neg(n_\delta \prec n_{\delta'})}{\neg(n_\delta \prec n_{\delta'})}$ in $D_\prec$ is applicable and we obtain $\neg(n_\delta \prec n_{\delta'}) \in E$, which contradicts our assumption.

   We have thus shown that $n_\delta \prec n_{\delta'} \in E$ iff $\neg(n_\delta \prec n_{\delta'}) \notin E$.

   For further proofs, we observe moreover the following complementary proposition.

   Lemma 1 *Let E be a consistent extension of $\mathcal{T}((D, W, <)) = (D', W')$ for ordered default theory $(D, W, <)$. We have for all $\delta, \delta' \in D$ that $(n_\delta \prec n_{\delta'}) \in E$ iff $(n_\delta \prec n_{\delta'}) \in W'$.*

   Proof 1 Clearly, we have $(n_\delta \prec n_{\delta'}) \in E$ if $(n_\delta \prec n_{\delta'}) \in W'$.

   Assume we have $(n_\delta \prec n_{\delta'}) \in E$ and $(n_\delta \prec n_{\delta'}) \notin W'$. Since $(n_\delta \prec n_{\delta'}) \notin W' = E_0$, there must exist (according to Definition 2.1) some $i \geq 0$ with $(n_\delta \prec n_{\delta'}) \notin E_i$ but $(n_\delta \prec n_{\delta'}) \in E_{i+1}$. Since there are no default rules with consequents containing positive occurrences of $\prec$-literals, we must have $(n_\delta \prec n_{\delta'}) \in Th(E_i)$. Due to the same reason, all positive occurrences in E_i must stem from $W_\prec$. In fact, all positive occurrences of $\prec$-literals in $W_\prec$ are connected disjunctively with a positive ok-literal. That is, they are of the form $((n_\delta \prec n_{\delta'}) \wedge \phi) \vee \varphi \vee \mathsf{ok}(n_\delta)$ for some formulas ϕ, φ. A proof for $E_i \vdash (n_\delta \prec n_{\delta'})$ must thus contain the negative ok-literal $\mathsf{ok}(n_\delta)$. There are however no negative occurrences of ok-literals in $\mathcal{T}((D, W, <))$, neither in D' nor in W', a contraction. ■

2+3. We show by induction on $<$ that $\mathsf{ok}(n_\delta) \in E$ and either $\mathsf{ap}(n_\delta) \in E$ or $\mathsf{bl}(n_\delta) \in E$ for all $\delta \in D$.

   Base By definition, $\mathsf{ok}(n_\top) \in W_\prec \subseteq E$. Also, $\perp \notin E$, since E is consistent. This implies that $\frac{\top \wedge \mathsf{ok}(n_\top):\top}{\top \wedge \mathsf{ap}(n_\top)} \in GD(D', E)$ along with $\frac{\mathsf{ok}(n_\top) \wedge \perp:}{\mathsf{bl}(n_\top)} \notin GD(D', E)$ and $\frac{\mathsf{ok}(n_\top):\perp}{\mathsf{bl}(n_\top)} \notin GD(D', E)$. That is, $\mathsf{ap}(n_\top) \in E$ and $\mathsf{bl}(n_\top) \notin E$.

   Step Consider δ and assume that for all δ' with $\delta < \delta'$ we have $\mathsf{ok}(n_{\delta'}) \in E$ and either $\mathsf{ap}(n_{\delta'}) \in E$ or $\mathsf{bl}(n_{\delta'}) \in E$.

   First, we prove the following lemma.

Lemma 2 *Given the induction hypothesis, we have* $\mathsf{ok}(n_\delta) \in E$.
Proof 2 By the induction hypothesis, we have either $\mathsf{ap}(n_{\delta'}) \in E$ or $\mathsf{bl}(n_{\delta'}) \in E$ for all δ' with $\delta < \delta'$.
By definition of $W_\prec$ and Lemma 1, we have $n_\delta \prec n_{\delta'} \in E$ for all δ, δ' with $\delta < \delta'$.
Analogously, we get $(n_\delta \prec n_{\delta'}) \notin E$ for all δ, δ' with $\delta \not< \delta'$. From this, we get by means of $D_\prec$ that $\neg(n_\delta \prec n_{\delta'}) \in E$ for all δ, δ' with $\delta < \delta'$.
Because E is deductively closed and contains

$$\forall x \in N. [\forall y \in N. (x \prec y) \supset (\mathsf{bl}(y) \vee \mathsf{ap}(y))] \supset \mathsf{ok}(x) ,$$

we therefore deduce that $\mathsf{ok}(n_\delta) \in E$. ■

Consider $n : \frac{\alpha : \beta}{\gamma} \in D$. We distinguish the following two cases.

- $\frac{\alpha \wedge \mathsf{ok}(n) : \beta}{\gamma \wedge \mathsf{ap}(n)} \in GD(D', E)$. Consequently, we have $\mathsf{ok}(n), \mathsf{ap}(n) \in E$.
- $\frac{\alpha \wedge \mathsf{ok}(n) : \beta}{\gamma \wedge \mathsf{ap}(n)} \notin GD(D', E)$. Then, we have one of the following cases.
  * $\alpha \wedge \mathsf{ok}(n) \notin E$. By Lemma 2 and the fact that E is deductively closed, we get $\alpha \notin E$. Again, by Lemma 2, this implies $\frac{\mathsf{ok}(n) : \neg\alpha}{\mathsf{bl}(n)} \in GD(D', E)$. That is, $\mathsf{bl}(n) \in E$.
  * $\neg\beta \in E$. By Lemma 2, this implies $\frac{\neg\beta \wedge \mathsf{ok}(n) :}{\mathsf{bl}(n)} \in GD(D', E)$. That is, $\mathsf{bl}(n) \in E$.

This demonstrates that either $\mathsf{ap}(n_\delta) \in E$ or $\mathsf{bl}(n_\delta) \in E$ for all $\delta \in D$. That is, $\mathsf{ap}(n_\delta) \in E$ iff $\mathsf{bl}(n_\delta) \notin E$.

4. Given $\mathsf{ok}(n_\delta) \in E_i$ and $\mathit{Prereq}(\delta) \in E_j$, we get $\mathsf{ok}(n_\delta) \wedge \mathit{Prereq}(\delta) \in E_{\max(i,j)+1}$. With $\neg\mathit{Justif}(\delta) \notin E$, this implies $\delta_a \in GD_{\max(i,j)+1}$, and furthermore $\mathit{Conseq}(\delta) \wedge \mathsf{ap}(n_\delta) \in E_{\max(i,j)+2}$. Hence, $\mathsf{ap}(n_\delta) \in E_{\max(i,j)+3}$.

5. Given $\mathsf{ok}(n_\delta) \in E_i$ and $\mathit{Prereq}(\delta) \notin E$, we get $\delta_{b_1} \in GD_i$, and furthermore $\mathsf{bl}(n_\delta) \in E_{i+1}$.

6. We have $\mathsf{ok}(n_\delta) \in E_i$ and $\neg\mathit{Justif}(\delta) \in E$. Assume $\neg\mathit{Justif}(\delta) \in E_k$ for some minimal k. Then, we get $\mathsf{ok}(n_\delta) \wedge \neg\mathit{Justif}(\delta) \in E_{\max(i,k)+1}$. This implies $\delta_{b_2} \in GD_{\max(i,k)+1}$, and furthermore $\mathsf{bl}(n_\delta) \in E_{\max(i,k)+2}$. That is, $\mathsf{bl}(n_\delta) \in E_j$ for some $j > i+1$.

7. Let $\mathsf{ok}(n_\delta) \notin E_{i-1}$ and $\mathsf{ok}(n_\delta) \in E_i$. We distinguish two cases.

   - $\mathsf{ap}(n_\delta) \in E$. Consequently, $\mathit{Prereq}(\delta) \wedge \mathsf{ok}(n_\delta) \in E$ and $\neg\mathit{Justif}(\delta) \notin E$.
     Assume that $\mathit{Prereq}(\delta) \in E_j$ for some $j < i$ (otherwise the claim follows trivially). We then have $\mathit{Prereq}(\delta) \wedge \mathsf{ok}(n_\delta) \in E_i$, implying that $\delta_a \in GD_i$. Hence, $\mathit{Conseq}(\delta) \wedge \mathsf{ap}(n_\delta) \in E_{i+1}$ and $\mathsf{ap}(n_\delta) \in E_{i+2}$. Accordingly, $\mathsf{ap}(n_\delta) \notin E_k$ for $k < i+2$.
   - $\mathsf{bl}(n_\delta) \in E$. We may distinguish the following cases.
     * $\mathit{Prereq}(\delta) \notin E$. This along with $\mathsf{ok}(n_\delta) \in E_i$ implies that $\delta_{b_1} \in GD_i$. Hence, $\mathsf{bl}(n_\delta) \in E_{i+1}$ but $\mathsf{bl}(n_\delta) \notin E_k$ for $k < i+1$.

* $\neg \mathit{Justif}(\delta) \in E$. Assume that $\neg \mathit{Justif}(\delta) \in E_j$ for some $j < i$. We then have $\neg \mathit{Justif}(\delta) \wedge \mathsf{ok}(n_\delta) \in E_i$, implying that $\delta_{b_2} \in GD_i$. Hence, $\mathsf{bl}(n_\delta) \in E_i$ but $\mathsf{bl}(n_\delta) \notin E_k$ for $k < i$.

Considering both last cases, we obtain that $\mathsf{bl}(n_\delta) \notin E_k$ for $k < i$.

■

Proof 4.2

8. This is a corollary to Theorem 4.1.3.

9. We have $\delta < \delta'$ and $\delta'_a, \delta'_{b_1}, \delta'_{b_2} \notin GD_i$ for some $\delta, \delta' \in D$.

   Assume that one of $\delta'_a \in GD_{i+1}$ or $\delta'_{b_1} \in GD_{i+1}$ or $\delta'_{b_2} \in GD_{i+1}$ holds for all δ' with $\delta < \delta'$. Thus, we have $\mathsf{bl}(n_{\delta'}) \in E_{i+2}$ or $\mathit{Conseq}(\delta') \wedge \mathsf{ap}(n_{\delta'}) \in E_{i+2}$ — and subsequently $\mathsf{ap}(n_{\delta'}) \in E_{i+3}$ — for all such δ'.

   Consider E_{i+2}. By definition of $W_\prec$, we have $n_\delta \prec n_{\delta'} \in E_{i+2}$ for all δ, δ' with $\delta < \delta'$ and by applying the argument of Lemma 1, we obtain $\neg(n_\delta \prec n_{\delta'}) \in E_{i+2}$ for all δ, δ' with $\delta \not< \delta'$. Also, we have

   $$\Big(\forall x \in N. [\forall y \in N. (x \prec y) \supset (\mathsf{bl}(y) \vee \mathsf{ap}(y))] \supset \mathsf{ok}(x)\Big) \in E_{i+2} \ .$$

   Because E_{i+3} contains the deductive closure of E_{i+2}, we deduce that $\mathsf{ok}(n_\delta) \in E_{i+3}$ but $\mathsf{ok}(n_\delta) \notin E_j$ for $j < i+3$. We thus have $\delta_a, \delta_{b_1}, \delta_{b_2} \notin GD_j$ for $j < i+3$.

10. Assume we have $\delta'_a \in GD_i$ or $\delta'_{b_1} \in GD_i$ or $\delta'_{b_2} \in GD_i$ for all δ' with $\delta < \delta'$ but $\delta'_a \notin GD_{i-1}$ and $\delta'_{b_1} \notin GD_{i-1}$ and $\delta'_{b_2} \notin GD_{i-1}$ for some such δ'.

    We then get either $\mathsf{bl}(n_{\delta'}) \in E_{i+1}$ or $\mathit{Conseq}(\delta') \wedge \mathsf{ap}(n_{\delta'}) \in E_{i+1}$ — and subsequently $\mathsf{ap}(n_{\delta'}) \in E_{i+2}$ — for all such δ'.

    By the reasoning employed in 9, this implies one of the following: $\delta_{b_1} \in GD_{i+2}$ or $\delta_a \in GD_{i+3}$ or $\delta_{b_2} \in GD_{i+3}$. In all, we thus have $\delta_a \in GD_j$ or $\delta_{b_1} \in GD_j$ or $\delta_{b_2} \in GD_j$ for some $j > i+2$.

11. Assume we have $\delta_a \in GD_i$ or $\delta_{b_1} \in GD_i$ or $\delta_{b_2} \in GD_i$ for all δ with $\delta < \delta'$ but $\delta_a \notin GD_{i-1}$ and $\delta_{b_1} \notin GD_{i-1}$ and $\delta_{b_2} \notin GD_{i-1}$ for some such δ.

    By reasoning backwards along the lines of 9, we get that either $\mathsf{ok}(n_\delta) \in E_{i-1}$ or $\mathsf{ok}(n_\delta) \in E_{i-2}$. Assume $\mathsf{ok}(n_\delta) \in E_{i-1}$. Continuing reasoning backwards in this way yields $\mathsf{bl}(n_{\delta'}) \in E_{i-2}$ or $\mathsf{ap}(n_{\delta'}) \in E_{i-2}$ for all δ' with $\delta < \delta'$.

    - If $\mathsf{bl}(n_{\delta'}) \in E_{i-2}$, we have either $\delta_{b_1} \in GD_{i-3}$ or $\delta_{b_2} \in GD_{i-3}$.
    - If $\mathsf{ap}(n_{\delta'}) \in E_{i-2}$, then $\mathit{Conseq}(\delta') \wedge \mathsf{ap}(n_{\delta'}) \in E_{i-3}$, that is, $\delta_a \in GD_{i-4}$.

    In all, we thus have $\delta'_a \in GD_j$ or $\delta'_{b_1} \in GD_j$ or $\delta'_{b_2} \in GD_j$ for some $j < i-2$.

■

Proof 4.3 Let E be a consistent extension of $\mathcal{T}((D,W,<)) = (D',W')$.

Assume there is an enumeration of $\langle\delta_i\rangle_{i\in I}$ of $GD(D',E)$ and some $\delta,\delta' \in D$ with $\delta < \delta'$ such that $i < j$ for some $\delta_i = \delta_t$ and all $\delta_j = (\delta')_{t'}$ with $t,t' \in \{a,b_1,b_2\}$. (Note that $\delta_i,\delta_j \in D'$.)

Since $\langle\delta_i\rangle_{i\in I}$ is grounded, we have that $W' \cup \mathit{Conseq}(\{\delta_0,\ldots,\delta_{i-1}\}) \vdash \mathit{Prereq}(\delta_i)$, which implies that $W' \cup \mathit{Conseq}(\{\delta_0,\ldots,\delta_{i-1}\}) \vdash \mathsf{ok}(n_\delta)$.

Without loss of generality, assume that we have for all $\delta'' \in D$ with $\delta'' \neq \delta'$ and $\delta'' \neq \delta$ that either $\delta \not< \delta''$, and so $\neg(n_\delta \prec n_{\delta''}) \in E$ by Lemma 1, or if $\delta < \delta''$, that is $(n_\delta \prec n_{\delta''}) \in E$ by Lemma 1, then $W' \cup \mathit{Conseq}(\{\delta_0,\ldots,\delta_{i-1}\}) \vdash \mathsf{ap}(n_{\delta''}) \vee \mathsf{bl}(n_{\delta''})$. That is, we suppose that δ' is at i the only default preferred to δ whose application has not yet been settled. By means of Lemma 1, we also have $(n_\delta \prec n_{\delta'}) \in E$ since $\delta < \delta'$.

Now, by definition of $\mathcal{T}((D,W,<))$, literal $\mathsf{ok}(n_\delta)$ is only derivable by means of

$$\forall x \in N.\, [\forall y \in N.\, (x \prec y) \supset (\mathsf{bl}(y) \vee \mathsf{ap}(y))] \supset \mathsf{ok}(x)\ ,$$

By what we have just supposed, this is reducible at i to

$$W' \cup \mathit{Conseq}(\{\delta_0,\ldots,\delta_{i-1}\}) \vdash (\mathsf{ap}(n_{\delta'}) \vee \mathsf{bl}(n_{\delta'})) \supset \mathsf{ok}(n_\delta)\ .$$

However, we have that $W' \cup \mathit{Conseq}(\{\delta_0,\ldots,\delta_{i-1}\}) \not\vdash \mathsf{ap}(n_{\delta'}) \vee \mathsf{bl}(n_{\delta'})$, since $j > i$ for all $\delta_j = (\delta')_{t'}$ with $t' \in \{a,b_1,b_2\}$ (and the fact that E is consistent), a contradiction. ■

Proof 4.4 Let (D,W) be a default theory and let $< \subseteq D \times D$ be some strict partial order on the default rules.

if-part Let E' be an extension of $(D',W') = \mathcal{T}((D,W,<))$.

Define

$$E = \mathit{Th}(W \cup \{\mathit{Conseq}(\delta) \mid \delta'_a \in GD(D',E')\})$$

Obviously, we have $E = E' \cap \mathcal{L}$ and for all $\varphi \in \mathcal{L}$ that $\varphi \in E$ iff $\varphi \in E'$. We show first that E is an extension of (D,W) and second that E is $<$-preserving.

By construction of E, we have the following.

1. $W \subseteq E$

2. $E = \mathit{Th}(E)$

To see that also

3. For any $\delta \in D$, if $\mathit{Prereq}(\delta) \in E$ and $\neg\mathit{Justif}(\delta) \notin E$ then $\mathit{Conseq}(\delta) \in E$.

is true, assume $\mathit{Prereq}(\delta) \in E$ and $\neg\mathit{Justif}(\delta) \notin E$. We then also have $\mathit{Prereq}(\delta) \in E'$ and $\neg\mathit{Justif}(\delta) \notin E'$. Since also $\mathsf{ok}(n_\delta) \in E'$ by Theorem 4.1.2, we obtain $\delta'_a \in GD(D',E')$, which implies $\mathit{Conseq}(\delta) \in E$ by definition of E. According to A.1, we have $\Gamma(E) \subseteq E$ by minimality of $\Gamma(E)$.

To show the reversal, assume that $E \not\subseteq \Gamma(E)$. Since $W \subseteq \Gamma(E)$ and both E and $\Gamma(E)$ are deductively closed, there must be some $\delta \in D$ such that $\mathit{Conseq}(\delta) \in E$ but

$Conseq(\delta) \notin \Gamma(E)$. By definition of E, $Conseq(\delta) \in E$ implies $\delta'_a \in GD(D', E')$. Hence, $\neg Justif(\delta) \notin E'$, which is equivalent to $\neg Justif(\delta) \notin E$. An induction on the grounded enumeration of $GD(D', E')$ shows that $\delta'_a \in GD(D', E')$ implies $Prereq(\delta) \in \Gamma(E)$. According to Definition A.1, we thus have $Conseq(\delta) \in \Gamma(E)$, a contradiction.

We have therefore shown that E is an extension of (D, W).

Finally, we must show that E preserves $<$:

Since E' is an extension of (D', W'), there is a grounded enumeration $\langle \delta'_k \rangle_{k \in K}$ of $GD(D', E')$. Let $\langle \delta_i \rangle_{i \in I}$ be the enumeration obtained from $\langle \delta'_k \rangle_{k \in K}$ by removing all default rules of form δ_{b_1} and δ_{b_2} and by replacing all default rules of form δ_a by their original rules $\delta \in D$. We note that $E = Th(W \cup \{Conseq(\delta_i) \mid i \in I\})$, since $\{\delta \mid \delta_a \in GD(D', E')\} = \{\delta_i \mid i \in I\}$. Furthermore, $\{\delta_i \mid i \in I\}$ equals $GD(D, E)$.

Consider δ_i for some $i \in I$ along with some $\delta \in D$ such that $\delta_i < \delta$. According to Definition 4.2, we distinguish the following two cases.

- If $\delta \in GD(D, E)$, then $\delta = \delta_j$ for some $j \in I$.

  Moreover, there are $k_i, k_j \in K$ such that $\delta'_{k_i} = (\delta_i)_a$ and $\delta'_{k_j} = (\delta_j)_a$. Since $\delta_i < \delta_j$, we get by Theorem 4.1.11, that $k_j < k_i$.

  By construction of $\langle \delta_i \rangle_{i \in I}$, this implies $j < i$.

- If $\delta \notin GD(D, E)$, then we also have $\delta_a \notin GD(D', E')$.

  By Theorem 4.1.8, this implies that $\delta_{b_1} \in GD(D', E')$ or $\delta_{b_2} \in GD(D', E')$:

  - If $\delta_{b_1} \in GD(D', E')$, then $\neg Justif(\delta) \in E'$. This is however equivalent to $\neg Justif(\delta) \in E$.
  - If $\delta_{b_2} \in GD(D', E')$, then $Prereq(\delta) \notin E'$.

    More precisely, let $\delta_{b_2} = \delta'_k$ for some $k \in K$. Then, we have

    $$W \cup \{Conseq(\delta'_l) \mid l < k\} \vdash \neg Prereq(\delta) \ .$$

    With $\delta'_{k_i} = (\delta_i)_a$, Theorem 4.1.10, implies moreover that $k < k_i$. By construction of $\langle \delta_i \rangle_{i \in I}$, we thus obtain $W \cup \{Conseq(\delta_l) \mid l < i\} \vdash \neg Prereq(\delta)$.

only-if-part Let E be a $<$-preserving extension of (D, W). That is, there exists a grounded enumeration $\langle \delta_i \rangle_{i \in I}$ of $GD(D, E)$ satisfying Conditions *1.* and *2.* in Definition 4.2.

Define

$$E' = Th\big(E \cup W_{\prec} \cup \{\mathrm{DCA}_N, \mathrm{UNA}_N\} \cup E^{\mathcal{L}}\big)$$

where $W_{\prec}$ is as defined in Definition 4.1 and

$$\begin{aligned} E^{\mathcal{L}} \quad = \quad & \{\mathsf{ok}(n_\delta) \mid \delta \in D\} \cup \{\mathsf{ap}(n_\delta) \mid \delta \in GD(D, E)\} \cup \{\mathsf{bl}(n_\delta) \mid \delta \notin GD(D, E)\} \\ & \cup \{\neg(n_\delta \prec n_{\delta'}) \mid (\delta, \delta') \notin <\} \end{aligned}$$

Clearly, we have $E = E' \cap \mathcal{L}$ and for all $\varphi \in \mathcal{L}$ that $\varphi \in E$ iff $\varphi \in E'$ We show that E' is an extension of $(D', W') = \mathcal{T}((D, W, <))$.

For this, define $E'_0 = W'$ and otherwise let E'_i and GD'_i be defined as in Definition 2.1 but here with respect to E'.

"$\subseteq$"-part We first show that $E' \subseteq \bigcup_{i=0}^{\infty} E'_i$. To begin with, we note that

$$W \cup W_{\prec} \cup \{\mathrm{DCA}_N, \mathrm{UNA}_N\} \subseteq E'_0 \quad \subseteq \quad \bigcup_{i=0}^{\infty} E'{}_i \tag{27}$$

by definition of $\mathcal{T}((D, W, <))$.

By definition of E', we have $\frac{:\neg(n_\delta \prec n_{\delta'})}{\neg(n_\delta \prec n_{\delta'})} \in GD'_0$ whenever $(\delta, \delta') \notin <$, which implies

$$\{\neg(n_\delta \prec n_{\delta'}) \mid (\delta, \delta') \notin <\} \subseteq E'_1 \quad \subseteq \quad \bigcup_{i=0}^{\infty} E'{}_i \ . \tag{28}$$

In what follows, we show the following inclusions.

$$\{\mathsf{ok}(n_\delta) \mid \delta \in D\} \quad \subseteq \quad \bigcup_{i=0}^{\infty} E'{}_i \tag{29}$$

$$\{\mathit{Conseq}(\delta), \mathsf{ap}(n_\delta) \mid \delta \in GD(D, E)\} \quad \subseteq \quad \bigcup_{i=0}^{\infty} E'{}_i \tag{30}$$

$$\{\mathsf{bl}(n_\delta) \mid \delta \notin GD(D, E)\} \quad \subseteq \quad \bigcup_{i=0}^{\infty} E'{}_i \tag{31}$$

First, we draw the reader's attention to the fact that the individual membership of the aforementioned sets in $\bigcup_{i=0}^{\infty} E'{}_i$ implies also that their deductive closure, given by E', is in $\bigcup_{i=0}^{\infty} E'{}_i$, since $\bigcup_{i=0}^{\infty} E'{}_i$ is a deductively closed set.

For proving inclusions (29) to (31), we define from the grounded enumeration $\langle \delta_i \rangle_{i \in I}$ of $GD(D, E)$, the enumeration $\langle \delta_{i,j} \rangle_{i \in I, j \in J}$, where $\delta_{i,0} = \delta_i$ and for $j > 0$, we let $\delta_{i,j}$ denote the default rules, say δ, in $D \setminus GD(D, E)$ for which either $\mathit{Prereq}(\delta) \notin E$ or $W \cup \mathit{Conseq}(\{\delta_0, \ldots, \delta_{i-1}\}) \vdash \neg \mathit{Justif}(\delta)$. The enumeration, or better its underlying lexicographic order on $I \times J$, is subject to the following constraint:

If $\delta_{i,j} < \delta_{k,l}$, then $k, l < i, j$, that is, $k < i$ or $k = i$ and $l < j$,

stipulating compatibility with $< \subseteq D \times D$. This is a feasible condition because (i) it is true for all $\delta \in GD(D, E)$, and (ii) all default rules in $D \setminus GD(D, E)$ can be arranged accordingly. Also, note that the enumeration encompasses all default rules in D, that is, $D = \{\delta_{i,j} \mid i \in I, j \in J\}$.

In concrete terms, we show by induction on the lexicographic order induced by $I \times J$ that

1. $\mathsf{ok}(n_\delta) \in \bigcup_{i=0}^{\infty} E'{}_i$ for all $\delta \in D$,
2. $\delta_a \in \bigcup_{i=0}^{\infty} GD_i$ or $\delta_{b_1} \in \bigcup_{i=0}^{\infty} GD_i$ or $\delta_{b_2} \in \bigcup_{i=0}^{\infty} GD_i$ for all $\delta \in D$.

The latter is clearly equivalent to proving inclusions (30) and (31).

Base By definition, we have $\delta_{0,0} = \delta_\top$. Also, by definition, $\mathsf{ok}(n_\top) \in E_0$. Clearly, we have $\delta_\top \in GD_1$. The argument for default rules of form $\delta_{0,j}$ is analogous to that given below.

Step Consider $\delta_{i,j}$ and assume that 1. and 2. hold for all $\delta_{i',j'}$ with $i', j' < i, j$.

We first show the following lemma:

Lemma 3 *Given the induction hypothesis, we have* $\mathsf{ok}(n_{i,j}) \in \bigcup_{i=0}^{\infty} E'{}_i$.

Proof 3 By the induction hypothesis, we have either $\mathsf{ap}(n_{\delta_{i',j'}}) \in \bigcup_{i=0}^{\infty} E'{}_i$ or $\mathsf{bl}(n_{\delta_{i',j'}}) \in \bigcup_{i=0}^{\infty} E'{}_i$ for all $\delta_{i',j'}$ with $i', j' < i, j$. By construction, this implies that either $\mathsf{ap}(n_{\delta_{i',j'}}) \in \bigcup_{i=0}^{\infty} E'_i$ or $\mathsf{bl}(n_{\delta_{i',j'}}) \in \bigcup_{i=0}^{\infty} E'_i$ for all $\delta_{i,j}$ with $\delta_{i,j} < \delta_{i',j'}$.

By definition of $W_{\prec}$ and the fact that $W_{\prec} \subseteq E'_0$, we have $n_\delta \prec n_{\delta'} \in \bigcup_{i=0}^{\infty} E'_i$ for all δ, δ' with $\delta < \delta'$. Together with (28) and the fact that

$$\Big(\forall x \in N.\, [\forall y \in N.\, (x \prec y) \supset (\mathsf{bl}(y) \vee \mathsf{ap}(y))] \supset \mathsf{ok}(x)\Big) \in \bigcup_{i=0}^{\infty} E'_i \ ,$$

we deduce that $\mathsf{ok}(n_\delta) \in \bigcup_{i=0}^{\infty} E'_i$, because $\bigcup_{i=0}^{\infty} E'_i$ is deductively closed. ■

For $\delta_{i,j} \in D$, we distinguish the following two cases.

$j = 0$ Consider $\delta_{i,0} \in GD(D, E)$. Since $\langle \delta_i \rangle_{i \in I}$ is grounded, we have $W \cup Conseq(\{\delta_0, \ldots, \delta_{i-1}\}) \vdash Prereq(\delta_{i,0})$. By the induction hypothesis, assuring that $\{(\delta_0)_a, \ldots, (\delta_{i-1})_a\} \subseteq \bigcup_{i=0}^{\infty} GD'_i$ holds, we get that $Prereq(\delta_{i,0}) \in E'_{j'}$ for some $j' \geq i$. In addition, we have $\mathsf{ok}(n_i) \in E'_{j''}$ for some $j'' \geq i$, by Lemma 3. Therefore, $Prereq(\delta_{i,0}) \wedge \mathsf{ok}(n_i) \in E'_j$ for some $j \geq i$.

Also, $\delta_{i,0} \in GD(D, E)$ implies that $\neg Justif(\delta_{i,0}) \notin E$, which is equivalent to $\neg Justif(\delta_{i,0}) \notin E'$ by definition of E'.

As a consequence, we obtain that $(\delta_{i,0})_a \in GD'_j$, that is, $(\delta_{i,0})_a \in \bigcup_{i=0}^{\infty} GD'_i$.

$j \neq 0$ Otherwise, we have $\delta_{i,j} \notin GD(D, E)$, which makes us distinguish the following cases:

- If $Prereq(\delta_{i,j}) \notin E$, then $Prereq(\delta_{i,j}) \notin E'$ by definition of E'. By Lemma 3, we get $\mathsf{ok}(n_{i,j}) \in E'_m$ for some m; hence $(\delta_{i,j})_{b_1} \in GD'_m$. That is, $(\delta_{i,j})_{b_1} \in \bigcup_{i=0}^{\infty} GD_i$.
- If $W \cup Conseq(\{\delta_0, \ldots, \delta_{i-1}\}) \vdash \neg Justif(\delta_{i,j})$, then the induction hypothesis, assuring that $\{(\delta_0)_a, \ldots, (\delta_{i-1})_a\} \subseteq \bigcup_{i=0}^{\infty} GD'_i$ holds, implies that $\neg Justif(\delta_{i,j}) \in E'_p$ for some p. In addition, we get by the induction hypothesis that $\mathsf{ok}(n_{i,j}) \in E'_m$ holds for some m. Hence $\neg Justif(\delta_{i,j}) \wedge \mathsf{ok}(n_{i,j}) \in E'_{\max(p,m)+1}$; whence $(\delta_{i,j})_{b_2} \in GD'_{\max(p,m)+1}$. That is, $(\delta_{i,j})_{b_2} \in \bigcup_{i=0}^{\infty} GD_i$.

"$\supseteq$"-part Next, we show that $\bigcup_{i=0}^{\infty} E'_i \subseteq E'$. That is, we prove by induction that $E'_i \subseteq E'$ for all i.

Base We have $E'_0 = W \cup W_{\prec} \cup \{\mathrm{DCA}_N, \mathrm{UNA}_N\} \subseteq E'$ by definition of E'.

Step Assume $E'_i \subseteq E'$ and consider $v \in E'_{i+1}$.

- If $v \in Th(E'_i)$, we also get $v \in E'_{i+1}$ by the induction hypothesis and the fact that E' is deductively closed.
- If $v \in \{Conseq(\delta') \mid \delta' \in GD'_i\}$, then we must distinguish the following cases:
  - If $\delta' = \frac{:\neg(n \prec m)}{\neg(n \prec m)}$, then $(n \prec m) \notin E'$. By definition of E', we then have $\neg(n \prec m) \in E'$; therefore, we also have $v \in E'$.

– If $\delta' = \frac{\alpha\wedge\mathsf{ok}(n_\delta):\beta}{\gamma\wedge\mathsf{ap}(n_\delta)}$, then $\alpha \wedge \mathsf{ok}(n_\delta) \in E'_i$ and $\neg\beta \notin E'$.
By the induction hypothesis and the fact that E' is deductively closed, we get $\alpha \in E'$. This is however equivalent to $\alpha \in E$. Correspondingly, we have $\neg\beta \notin E$. This implies $\delta \in GD(D,E)$, that is, $\gamma \in E$; hence $\gamma \in E'$ because $E \subseteq E'$. Also, $\mathsf{ap}(n_\delta) \in E'$ because $\{\mathsf{ap}(n_\delta) \mid \delta \in GD(D,E)\} \subseteq E'$. Therefore, we obtain $\alpha \wedge \mathsf{ok}(n_\delta) \in E'$, that is, $v \in E'$.

– If $\delta' = \frac{\mathsf{ok}(n_\delta):\neg\alpha}{\mathsf{bl}(n_\delta)}$), then $\neg\alpha \notin E'$, whence $\neg\alpha \notin E$. Therefore, $\delta \notin GD(D,E)$, which implies $\mathsf{bl}(n_\delta) \in E'$ because $\{\mathsf{bl}(n_\delta) \mid \delta \notin GD(D,E)\} \subseteq E'$. That is, $v \in E'$.

– If $\delta' = \frac{\neg\beta\wedge\mathsf{ok}(n_\delta):}{\mathsf{bl}(n_\delta)}$, then $\neg\beta \wedge \mathsf{ok}(n_\delta) \in E'_i$. By the induction hypothesis and the fact that E' is deductively closed, we get $\neg\beta \in E'$. This is however equivalent to $\neg\beta \in E$. Therefore, $\delta \notin GD(D,E)$, which implies $\mathsf{bl}(n_\delta) \in E'$ because $\{\mathsf{bl}(n_\delta) \mid \delta \notin GD(D,E)\} \subseteq E'$. That is, $v \in E'$.

Accordingly, $E'_{i+1} \subseteq E'$.

Therefore, we have shown that $\bigcup_{i=0}^{\infty} E'_i \subseteq E'$

■

Proof 4.1 This is an immediate consequence of Theorem 4.4. ■

Proof 4.2 For $< = \emptyset$, the two conditions of Definition 4.2 are trivially true for any enumeration of default rules. In such a case, all extension of a default theory are $<$-preserving.

The actual result is then an immediate consequence of Theorem 4.4. ■

Proof 4.5 Let $(D, W, <)$ be a propositional, semi-normal, ordered default theory such that (D, W) is even. Consequently (D, W) has an extension [Papadimitriou and Sideri, 1994, Theorem 5]. Let the associated literal graph be $L((D, W))$ and let the associated dependency graph be $G((D, W)) = (D, \mathcal{E})$.

Since (D, W) is propositional, we can simplify the translation given in Definition 4.1.

We define the translation of ordered default theory $(D, W, <)$ over a propositional language based on a set of atomic sentences $\mathbf{P}$ as follows. For each $n \in N$, corresponding to the ground instances $\mathsf{ok}(n)$, $\mathsf{ap}(n)$, $\mathsf{bl}(n)$ we will have additional atomic sentences $\mathsf{ok.n}$, $\mathsf{ap.n}$, $\mathsf{bl.n}$ respectively. For each $n, m \in N$, corresponding to the ground instance $n \prec m$ we will have additional atomic sentence $\mathsf{n.{\prec}.m}$. For sentence α of FOL, we will let $\mathbf{Pr}(\alpha)$ be the sentence where each ground instance of the form $\mathsf{ok}(n)$, $\mathsf{ap}(n)$, $\mathsf{bl}(n)$, $n \prec m$ is replaced by its corresponding atomic sentence in $\mathbf{P}$.

Our translation is given as follows: $\mathcal{T}((D, W, <)) = (D', W')$ where

$$\begin{aligned} D' &= \left\{ \frac{\alpha\wedge\mathsf{ok.n}:\beta}{\gamma\wedge\mathsf{ap.n}}, \frac{\mathsf{ok.n}:\neg\alpha}{\mathsf{bl.n}}, \frac{\neg\beta\wedge\mathsf{ok.n}:}{\mathsf{bl.n}} \;\middle|\; n : \frac{\alpha:\beta}{\gamma} \in D \right\} \\ W' &= W \cup W_\prec \end{aligned}$$

and where

$$\begin{aligned} W_\prec &= \{\mathbf{Pr}(n_\delta \prec n_{\delta'}) \mid (\delta, \delta') \in <\} \cup \{\neg\mathbf{Pr}(n_\delta \prec n_{\delta'}) \mid (\delta, \delta') \notin <\} \\ &\cup \{\mathsf{ok.n}_\top\} \\ &\cup \{\mathbf{Pr}((\wedge_{y\in N}[(x \prec y) \supset (\mathsf{bl}(y) \vee \mathsf{ap}(y))]) \supset \mathsf{ok}(x)) \mid x \in N\}. \end{aligned}$$

For $\delta \in D$ we denote the defaults in the image of the translation by δ_a, δ_{b_1}, δ_{b_2} respectively.

The translation given in Definition 4.1 is simplified by removing the default in $D_\prec$ and instead listing explicitly the occurrences and non-occurrences of $\prec$ (again, represented by $|D|^2$ atomic sentences) about which we have complete knowledge. We no longer require DCA_N and UNA_N. The last formula in $W_\prec$ in Definition 4.1 is replaced by $|D|$ explicit conjunctions.

Consider the image of $\mathcal{T}((D, W, <))$, viz. (D', W').

Clearly, for the respective literal graphs we have $L((D, W)) \subseteq L((D', W'))$.

Observe that the only addition to the literal graph comes from the final formula in $W_\prec$, viz. the $m = |D|$ instances of

$$\mathbf{Pr}((\wedge_{y \in N}[(n \prec y) \supset (\mathsf{bl}(y) \vee \mathsf{ap}(y))]) \supset \mathsf{ok}(n)) \qquad \text{for each } n \in N.$$

For $\delta_i \in D$, we have the associated formula

$$\begin{aligned}
&([\mathsf{n_i}.\prec.\mathsf{n_1} \supset (\mathsf{bl.n_1} \vee \mathsf{ap.n_1})] \wedge \\
&\ [\mathsf{n_i}.\prec.\mathsf{n_2} \supset (\mathsf{bl.n_2} \vee \mathsf{ap.n_2})] \wedge \\
&\ \ldots \wedge \\
&\ [\mathsf{n_i}.\prec.\mathsf{n_m} \supset (\mathsf{bl.n_m} \vee \mathsf{ap.n_m})]) \supset \mathsf{ok.n_i}.
\end{aligned}$$

This can be written in DNF as

$$\begin{aligned}
&[\mathsf{n_i}.\prec.\mathsf{n_1} \wedge \neg\mathsf{bl.n_1} \wedge \neg\mathsf{ap.n_1}] \vee \\
&[\mathsf{n_i}.\prec.\mathsf{n_2} \wedge \neg\mathsf{bl.n_2} \wedge \neg\mathsf{ap.n_2}] \vee \\
&\ \ldots \vee \\
&[\mathsf{n_i}.\prec.\mathsf{n_m} \wedge \neg\mathsf{bl.n_m} \wedge \neg\mathsf{ap.n_m}] \vee \mathsf{ok.n_i}.
\end{aligned} \tag{32}$$

Since we have complete knowledge about our partial order on defaults, Equation (32) is logically equivalent (with respect to $W_\prec$) to

$$\begin{aligned}
&[\mathsf{n_i}.\prec.\mathsf{n_{i_1}} \wedge \neg\mathsf{bl.n_{i_1}} \wedge \neg\mathsf{ap.n_{i_1}}] \vee \\
&[\mathsf{n_i}.\prec.\mathsf{n_{i_2}} \wedge \neg\mathsf{bl.n_{i_2}} \wedge \neg\mathsf{ap.n_{i_2}}] \vee \\
&\ \ldots \vee \\
&[\mathsf{n_i}.\prec.\mathsf{n_{i_j}} \wedge \neg\mathsf{bl.n_{i_j}} \wedge \neg\mathsf{ap.n_{i_j}}] \vee \mathsf{ok.n_i} \\
&\qquad \text{where, for } \delta_i \text{, exactly } \mathsf{n_i}.\prec.\mathsf{n_{i_1}}, \mathsf{n_i}.\prec.\mathsf{n_{i_2}}, \ldots, \mathsf{n_i}.\prec.\mathsf{n_{i_j}} \in W_\prec.
\end{aligned}$$

Since $\mathsf{n_i}.\prec.\mathsf{n_{i_1}}, \mathsf{n_i}.\prec.\mathsf{n_{i_2}}, \ldots, \mathsf{n_i}.\prec.\mathsf{n_{i_j}} \in W_\prec$, this in turn is equivalent (with respect to $W_\prec$) to

$$\begin{aligned}
&[\neg\mathsf{bl.n_{i_1}} \wedge \neg\mathsf{ap.n_{i_1}}] \vee \\
&[\neg\mathsf{bl.n_{i_2}} \wedge \neg\mathsf{ap.n_{i_2}}] \vee \\
&\ \ldots \vee \\
&[\neg\mathsf{bl.n_{i_j}} \wedge \neg\mathsf{ap.n_{i_j}}] \vee \mathsf{ok.n_i} \\
&\qquad \text{where, for } \delta_i \text{, exactly } \mathsf{n_i}.\prec.\mathsf{n_{i_1}}, \mathsf{n_i}.\prec.\mathsf{n_{i_2}}, \ldots, \mathsf{n_i}.\prec.\mathsf{n_{i_j}} \in W_\prec.
\end{aligned}$$

So these formulas (one for each default) are the only formulas that add edges to the literal graph. Since these formulas are in DNF, the edges added to arrive at $L((D', W'))$ will consist of pairs of literals drawn from distinct disjuncts in these formulas.

We obtain the following edges. For default δ_i we obtain the following three sets of edges in $L((D', W'))$:

$$\begin{aligned}&(\mathsf{bl.n_i}, \neg\mathsf{bl.n_j})\\&(\mathsf{bl.n_i}, \neg\mathsf{ap.n_j})\\&(\mathsf{ap.n_i}, \neg\mathsf{bl.n_j})\\&(\mathsf{ap.n_i}, \neg\mathsf{ap.n_j}) \qquad \text{for various } 1 \leq j \leq |D|, i \neq j\end{aligned} \tag{33}$$

$$\begin{aligned}&(\neg\mathsf{ok.n_i}, \neg\mathsf{bl.n_j})\\&(\neg\mathsf{ok.n_i}, \neg\mathsf{ap.n_j}) \qquad \text{for } \mathsf{n_i}.{\prec}.\mathsf{n_j} \in W_{\prec}\end{aligned} \tag{34}$$

$$\begin{aligned}&(\mathsf{bl.n_i}, \mathsf{ok.n_j})\\&(\mathsf{ap.n_i}, \mathsf{ok.n_j}) \qquad \text{for } \mathsf{n_i}.{\prec}.\mathsf{n_j} \in W_{\prec}\end{aligned} \tag{35}$$

Consider the dependency graph $G((D', W')) = (D', \mathcal{E}')$ where $\mathcal{E}' = \mathcal{E}'_0 \cup \mathcal{E}'_1$.

1. The four edge types in the first group (33), contribute no edges to $\mathcal{E}'_0$ or $\mathcal{E}'_1$. This is because, for any default rule with $\mathsf{bl.n_i}$ or $\mathsf{ap.n_i}$ in its consequent, there is no default rule with $\mathsf{bl.n_i}$ or $\mathsf{ap.n_i}$ in its prerequisite or $\neg\mathsf{bl.n_i}$ or $\neg\mathsf{ap.n_i}$ in its justification.

2. For the second set of edge types, group (34), $\neg\mathsf{ok.n_i}$ does not occur in the consequent of a default in D', and so these edges do not contribute any edges to $\mathcal{E}'_0$ or $\mathcal{E}'_1$.

3. This leaves the group (35). $\mathsf{ok.n_j}$ does not appear negatively in the justification or consequent of a rule in D', and so does not contribute to $\mathcal{E}'_1$. However, for every rule with literals $\mathsf{bl.n_i}$ or $\mathsf{ap.n_i}$ in its consequent, there are rules with literal $\mathsf{ok.n_j}$ in the antecedent, namely the image of every rule δ_j (in D) where $\delta_j < \delta_i$.

So what this means is that, for W', we have that

$$\text{If } (\delta', \delta) \in < \text{ then } (\delta_x, \delta'_y) \in \mathcal{E}'_0 \text{ for } x, y \in \{a, b_1, b_2\}. \tag{36}$$

Moreover, as argued above, there are no other additions to $\mathcal{E}'_0$ resulting from W' that weren't already present from (D, W).

For the defaults in D', we have the following.

1. If $(\delta, \delta') \in \mathcal{E}_0$ then there is a literal x appearing positively in $\mathit{Conseq}(\delta)$, a literal appearing positively in $\mathit{Prereq}(\delta')$ and $L^*(x, y)$.

   But any literal in $\mathit{Conseq}(\delta)$ is a literal in $\mathit{Conseq}(\delta_a)$ and any literal in $\mathit{Prereq}(\delta')$ is a literal in $\mathit{Prereq}(\delta'_a)$.

   As well, a literal in $\mathit{Prereq}(\delta')$ appearing positively also appears negatively in $\mathit{Prereq}(\delta'_{b1})$.

$$\text{Thus if } (\delta, \delta') \in \mathcal{E}_0 \text{ then } (\delta_a, \delta'_a) \in \mathcal{E}'_0 \text{ and } (\delta_a, \delta'_{b_1}) \in \mathcal{E}'_1. \tag{37}$$

2. If $(\delta', \delta) \in \mathcal{E}_1$, a similar argument shows that

$$(\delta'_a, \delta_a) \in \mathcal{E}'_1 \text{ and } (\delta'_a, \delta_{b_2}) \in \mathcal{E}'_0. \tag{38}$$

Since there are no other cases, $\mathcal{E}' = \mathcal{E}'_0 \cup \mathcal{E}'_1$ is completely specified by (36)–(38).

Observe that if $G((D, W))$ is even, then the graph with vertices in D' and edges given in (37) and (38) is even. (That is, the subgraph with vertices $\{\delta_a | \delta \in D\}$ is isomorphic to $G((D, W))$. Otherwise for $\delta, \delta' \in D$ the only other edges are of the form $(\delta'_a, \delta_{b_1})$ (from (37)) and $(\delta'_a, \delta_{b_2})$ (from (38)). But the vertices δ_{b_1}, δ_{b_2} have no outgoing edges and so cannot be part of any cycle.) We will use this observation to conclude the proof.

Our result now follows easily: we are given that $(D, W, <)$, a propositional, semi-normal, ordered default theory where (D, W) is even and for the associated dependency graph $G((D, W)) = (D, \mathcal{E})$, we have that

$$(D, \mathcal{E} \cup \{\mathsf{n}_\delta.\prec.\mathsf{n}_{\delta'} \mid (\delta', \delta) \in <\})$$

has no cycles incorporating elements of $<$. (That is, if $\mathcal{C}$ is a cycle in $(D, \mathcal{E} \cup \{\mathsf{n}_\delta.\prec.\mathsf{n}_{\delta'} \mid (\delta', \delta) \in <\})$ then $\mathcal{C}$ is a cycle in $(D, \mathcal{E})$.)

For the dependency graph $G((D', W')) = (D', \mathcal{E}')$, let

$$\mathcal{E}_C = \{(\delta_x, \delta'_y) \mid x, y \in \{a, b_1, b_2\} \text{ and: } (\delta', \delta) \in < \text{ or } (\delta, \delta') \in \mathcal{E}'\}.$$

We have that

$$\mathcal{E}' = \mathcal{E}'_0 \cup \mathcal{E}'_1 \subseteq \mathcal{E}_C.$$

and so $(D', \mathcal{E}')$ is a subgraph of $(D', \mathcal{E}_C)$.

By assumption, in $G((D, W))$ the edge set $\mathcal{E} \cup \{\mathsf{n}_\delta.\prec.\mathsf{n}_{\delta'} \mid (\delta', \delta) \in <\}$ has no cycles incorporating elements of $<$.

So in $\mathcal{E}_C$ there are no cycles incorporating elements in the image of $<$.

So in $\mathcal{E}' \subseteq \mathcal{E}_C$ there are no cycles incorporating elements in the image of $<$.

So from the above observation, the only cycles in $\mathcal{E}'$ are even. ■

Proof 4.6 Let $(D, W, <)$ be a normal, ordered default theory where

$$Pred(Prereq(D)) \cap Pred(Conseq(D)) = \emptyset \quad \text{and} \quad Pred(W) \cap Pred(Conseq(D)) = \emptyset.$$

Consider the default theory (D', W) where D' is defined by:

$$D' = \left\{ \tfrac{\top : \beta}{\beta} \,\middle|\, \tfrac{\alpha : \beta}{\beta} \in D \text{ and } W \vdash \alpha \right\}.$$

We make use of the following lemma:

Lemma 4 *Given our assumptions above, E is an extension of (D, W) iff E is an extension of (D', W).*

Proof 4 Clearly it is harmless replacing the prerequisite of a default by $\top$ if the prerequisite is implied by W.

Since $Pred(Prereq(D)) \cap Pred(Conseq(D)) = \emptyset$, for any default δ where $W \not\vdash Prereq(\delta)$, we have that for extension E of (D, W), $\delta \notin GD(D, E)$.

If this were not the case, then we would have $W \not\vdash Prereq(\delta)$ but $E \vdash Prereq(\delta)$ or equivalently $W \cup Conseq(GD(D, E)) \vdash Prereq(\delta)$.

However this is impossible since $W \cup Conseq(GD(D, E))$ is consistent by assumption, and we have that $Pred(Prereq(\delta)) \cap Pred(Conseq(GD(D, E))) = \emptyset$ and $Pred(W) \cap Pred(Conseq(GD(D, E))) = \emptyset$.

Consequently $\delta \notin GD(D, E)$ and the lemma follows immediately. ■

Given Lemma 4 we can assume without loss of generality that for $\delta \in D$ we have $Prereq(\delta) \equiv \top$.

Definition 4.1 simplifies considerably. We obtain

$$\begin{aligned} D' &= \left\{ \frac{\top \wedge \mathsf{ok}(n) : \beta}{\beta \wedge \mathsf{ap}(n)}, \frac{\mathsf{ok}(n) : \neg\top}{\mathsf{bl}(n)}, \frac{\neg\beta \wedge \mathsf{ok}(n) :}{\mathsf{bl}(n)} \;\middle|\; n : \frac{\top : \beta}{\beta} \in D \right\} \\ &= \left\{ \frac{\mathsf{ok}(n) : \beta}{\beta \wedge \mathsf{ap}(n)}, \frac{\neg\beta \wedge \mathsf{ok}(n) :}{\mathsf{bl}(n)} \;\middle|\; n : \frac{\top : \beta}{\beta} \in D \right\} \end{aligned}$$

We conclude by noting that since it is impossible to derive $\neg\mathsf{ok}(n)$ for any $\delta_n \in D$, so it is impossible to derive $\neg\mathsf{ap}(n)$, $\neg\mathsf{bl}(n)$. Hence $\frac{\mathsf{ok}(n) : \beta \wedge \mathsf{ap}(n)}{\beta \wedge \mathsf{ap}(n)}$ has precisely the same effect as $\frac{\mathsf{ok}(n) : \beta}{\beta \wedge \mathsf{ap}(n)}$, and $\frac{\neg\beta \wedge \mathsf{ok}(n) : \mathsf{bl}(n)}{\mathsf{bl}(n)}$ has precisely the same effect as $\frac{\neg\beta \wedge \mathsf{ok}(n) :}{\mathsf{bl}(n)}$. Consequently our translated default theory is equivalent to a normal default theory, which is guaranteed to have an extension. ■

Proof 5.1 We start by fixing the components of the different default theories:

Define $\mathcal{T}((D, W, <)) = (D^t, W^t)$,

$$D^t = \{\delta_a, \delta_{b_1}, \delta_{b_2} \mid \delta \in D\} \cup D^t_{\prec} \tag{39}$$
$$W^t = W \cup W^t_{\prec} \cup \{\mathrm{DCA}_N, \mathrm{UNA}_N\} \tag{40}$$

where $W^t_{\prec}$ and $D^t_{\prec}$ are defined as their unindexed counterparts $W_{\prec}$ and $D_{\prec}$, respectively, in Definition 4.1.

Accordingly, define $\mathcal{D}((D, W \cup \{n_\delta \prec n_{\delta'} \mid (\delta, \delta') \in <\})) = (D^d, W^d)$,

$$D^d = \{\delta_a, \delta_{b_1}, \delta_{b_2} \mid \delta \in D\} \cup D^d_{\prec} \tag{41}$$
$$W^d = W \cup \{n_\delta \prec n_{\delta'} \mid (\delta, \delta') \in <\} \cup W^d_{\prec} \cup \{\mathrm{DCA}_N, \mathrm{UNA}_N\} \tag{42}$$

where $W^d_{\prec}$ and $D^d_{\prec}$ are defined as their unindexed counterparts $W_{\prec}$ and $D_{\prec}$, respectively, in Definition 5.1.

only-if part Let E be an extension of $\mathcal{T}((D, W, <))$. Define

$$\begin{aligned} E' = Th(W^d \;&\cup\; \{\mathsf{ok}(n_\delta) \mid \delta \in D\} \\ &\cup\; \{Conseq(\delta) \wedge \mathsf{ap}(n_\delta) \mid \delta_a \in GD(D^t, E)\} \\ &\cup\; \{\mathsf{bl}(n_\delta) \mid \delta_{b_i} \in GD(D^t, E), i = 1, 2\} \\ &\cup\; \left\{ (n_\delta \not\prec n_{\delta'}) \mid \frac{: \neg(n_\delta \prec n_{\delta'})}{\neg(n_\delta \prec n_{\delta'})} \in GD(D^t, E) \right\}) \end{aligned}$$

First, we show that $E \cap \mathcal{L} = E' \cap \mathcal{L}$. We distinguish three cases, while abbreviating $\{Conseq(\delta) \mid \delta \in GD(D^t, E)\}$ by $Conseq(GD(D^t, E))$:

- Consider $v \in W$.

  Since $W \subseteq E \cap \mathcal{L}$ and $W \subseteq E' \cap \mathcal{L}$, this implies $v \in E \cap \mathcal{L}$ iff $v \in E' \cap \mathcal{L}$.

- Consider $v \in \mathit{Conseq}(GD(D^t, E))$.

  As before, $\mathit{Conseq}(GD(D^t, E)) \subseteq E \cap \mathcal{L}$ and $\mathit{Conseq}(GD(D^t, E)) \subseteq E' \cap \mathcal{L}$, implies $v \in E \cap \mathcal{L}$ iff $v \in E' \cap \mathcal{L}$.

- Consider $v \in \mathit{Th}(W \cup \mathit{Conseq}(GD(D^t, E)))$.

  We have $W \cup \mathit{Conseq}(GD(D^t, E)) \subseteq E \cap \mathcal{L}$ and $W \cup \mathit{Conseq}(GD(D^t, E)) \subseteq E' \cap \mathcal{L}$. The fact that E as well as E' are deductively closed imply furthermore that $v \in \mathit{Th}(W \cup \mathit{Conseq}(GD(D^t, E))) \subseteq E \cap \mathcal{L}$ iff $v \in \mathit{Th}(W \cup \mathit{Conseq}(GD(D^t, E))) \subseteq E' \cap \mathcal{L}$.

We have thus shown that for all $v \in \mathcal{L}$ that

$$v \in E \cap \mathcal{L} \text{ iff } v \in E' \cap \mathcal{L}. \tag{43}$$

We draw on this fact in the sequel.
Second, we show that E' is an extension of $\mathcal{D}((D, W \cup \{n_\delta \prec n_{\delta'} \mid (\delta, \delta') \in <\}))$.

For this, we first show the following three propositions:

1. $W^d \subseteq E'$. This is true by definition.

2. $\mathit{Th}(E') = E'$. This is true by definition.

3. For any $\delta \in D^d$, if $\mathit{Prereq}(\delta) \in E'$ and $\neg \mathit{Justif}(\delta) \notin E'$ then $\mathit{Conseq}(\delta) \in E'$.

   To show this, suppose $\mathit{Prereq}(\delta) \in E'$ and $\neg \mathit{Justif}(\delta) \notin E'$.

   - If $\delta = \frac{:\neg(n_\delta \prec n_{\delta'}),(n_\delta \prec n_{\delta'})}{(n_\delta \not\prec n_{\delta'})}$, then we have $\neg(n_\delta \prec n_{\delta'}) \notin E'$ as well as $(n_\delta \prec n_{\delta'}) \notin E'$. The definition of E' and the fact that $(n_\delta \prec n_{\delta'}) \notin E'$ imply that $(n_\delta \prec n_{\delta'}) \notin W^d$, that is, $(\delta, \delta') \notin <$. This implies that $(n_\delta \prec n_{\delta'}) \notin W^t_\prec$, that is, $(n_\delta \prec n_{\delta'}) \notin W^t$. This implies by Lemma 1 that $(n_\delta \prec n_{\delta'}) \notin E$. Hence, $\frac{:\neg(n_\delta \prec n_{\delta'})}{\neg(n_\delta \prec n_{\delta'})} \in GD(D^t, E)$, which finally implies $(n_\delta \not\prec n_{\delta'}) \in E'$ by definition of E'.
   - If $\delta = \delta_a$, then $\mathit{Prereq}(\delta) \wedge \mathsf{ok}(n_\delta) \in E'$ and $\neg \mathit{Justif}(\delta) \notin E'$. Since E' is deductively closed we have $\mathit{Prereq}(\delta) \in E'$.

     With (43), we obtain $\mathit{Prereq}(\delta) \in E$ and $\neg \mathit{Justif}(\delta) \notin E$. Since E is an extension of $\mathcal{T}((D, W, <))$, we obtain by Theorem 4.1.1 that $\mathsf{ok}(n_\delta) \in E$. Since E is deductively closed we have moreover $\mathit{Prereq}(\delta) \wedge \mathsf{ok}(n_\delta) \in E$. We thus get $\delta_a \in GD(D^t, E)$, which implies $\mathit{Conseq}(\delta) \wedge \mathsf{ap}(n_\delta) \in E'$ by definition of E'.
   - If $\delta = \delta_{b_1}$, then $\mathsf{ok}(n_\delta) \in E'$ and $\neg \mathit{Prereq}(\delta) \notin E'$.

     With (43), we obtain $\neg \mathit{Prereq}(\delta) \notin E$. Since E is an extension of $\mathcal{T}((D, W, <))$, we obtain by Theorem 4.1.1 that $\mathsf{ok}(n_\delta) \in E$. We thus get $\delta_{b_1} \in GD(D^t, E)$, which implies $\mathsf{bl}(n_\delta) \in E'$ by definition of E'.

- If $\delta = \delta_{b_2}$, then $\neg \mathit{Justif}(\delta) \wedge \mathsf{ok}(n_\delta) \in E'$. Since E' is deductively closed we have $\neg \mathit{Justif}(\delta) \in E'$.

  With (43), we obtain $\neg \mathit{Justif}(\delta) \in E$. Since E is an extension of $\mathcal{T}((D, W, <))$, we obtain by Theorem 4.1.1 that $\mathsf{ok}(n_\delta) \in E$. Since E is deductively closed we have moreover $\mathit{Prereq}(\delta) \wedge \mathsf{ok}(n_\delta) \in E$. We thus get $\delta_{b_2} \in \mathit{GD}(D^t, E)$, which implies $\mathsf{bl}(n_\delta) \in E'$ by definition of E'.

We have thus shown that $\mathit{Conseq}(\delta) \in E'$.

According to Definition A.1, we get $\Gamma(E') \subseteq E'$ by minimality of $\Gamma(E')$.

To show the reverse, assume that $E' \not\subseteq \Gamma(E')$. Consider $v \in E'$ and assume $v \notin \Gamma(E')$. We distinguish the following cases.

- If $v \in W^d$ then $v \in \Gamma(E')$, since $W^d \subseteq \Gamma(E')$, a contradiction.

- If $v \in \left\{ (n_\delta \not\prec n_{\delta'}) \mid \frac{:\neg(n_\delta \prec n_{\delta'})}{\neg(n_\delta \prec n_{\delta'})} \in \mathit{GD}(D^t, E) \right\}$ then $(n_\delta \prec n_{\delta'}) \notin E$. This implies by Lemma 1 that $(n_\delta \prec n_{\delta'}) \notin W^t$, that is, $(\delta, \delta') \notin <$.

  By definition of E', we thus have $\neg(n_\delta \prec n_{\delta'}) \notin E'$ as well as $(n_\delta \prec n_{\delta'}) \notin E'$. Since $\frac{:\neg(n_\delta \prec n_{\delta'}), (n_\delta \prec n_{\delta'})}{(n_\delta \not\prec n_{\delta'})} \in D^d$, we have by definition of Γ that $(n_\delta \not\prec n_{\delta'}) \in \Gamma(E')$, a contradiction.

- We proceed by induction on the grounded enumeration $\langle \delta_i \rangle_{i \in I}$ of $\mathit{GD}(D^t, E)$ for

  $$v \in \{\mathsf{bl}(n_\delta) \mid \delta_{b_i} \in \mathit{GD}(D^t, E), i = 1, 2\} \cup \{\mathit{Conseq}(\delta) \wedge \mathsf{ap}(n_\delta) \mid \delta_a \in \mathit{GD}(D^t, E)\} .$$

  We show that for all $\delta_i, i \in I$ such that $\delta_i = (\delta)_a$ or $\delta_i = (\delta)_{b_j}$ for some $j = 1, 2$ and some $\delta \in D$ that either $\mathit{Conseq}(\delta) \wedge \mathsf{ap}(n_\delta) \in \Gamma(E')$ or $\mathsf{bl}(n_\delta) \in \Gamma(E')$.

  First, we have the following lemma.

  Lemma 5 *Given the induction hypothesis, we have* $\mathsf{ok}(n_\delta) \in \Gamma(E')$.

  Proof 5 Analogous to Proof 2. ■

  Base By definition, $\mathsf{ok}(n_\top) \in \Gamma(E')$. Clearly, we have $\delta_0 = (\delta_\top)_a$, which implies that $\mathit{Conseq}(\delta_\top) \wedge \mathsf{ap}(n_\top) \in \Gamma(E')$.

  Step Consider δ_i.

  Since $\langle \delta_i \rangle_{i \in I}$ is grounded in W^t, we obtain $W^t \cup \mathit{Conseq}(\{\delta_0, \ldots, \delta_{i-1}\}) \vdash \mathit{Prereq}(\delta_i)$. Let $\mathit{Prereq}(\delta_i) = \phi \wedge \mathsf{ok}(n_\delta)$ For $\phi \in \{\mathit{Prereq}(\delta), \top, \neg \mathit{Justif}(\delta)\}$ where $\delta_i = (\delta)_a$ or $\delta_i = (\delta)_{b_j}$ for some $j = 1, 2$ and some $\delta \in D$. Then, we clearly have $W \cup \mathit{Conseq}(\{\delta_0, \ldots, \delta_{i-1}\}) \vdash \phi$ and by monotonicity $W^d \cup \mathit{Conseq}(\{\delta_0, \ldots, \delta_{i-1}\}) \vdash \phi$. By definition, $W^d \subseteq \Gamma(E')$. Furthermore, we have $\mathit{Conseq}(\delta_j) \in \Gamma(E')$ for $j < i$ by the induction hypothesis. This implies that $\phi \in \Gamma(E')$ because $\Gamma(E')$ is deductively closed. By Lemma 5, we have furthermore $\mathsf{ok}(n_\delta) \in \Gamma(E')$ which gives us, again by appeal to $\Gamma(E')$'s deductive closure, that $\phi \wedge \mathsf{ok}(n_\delta) \in \Gamma(E')$. As a consequence, $\mathit{Prereq}(\delta_i) \in \Gamma(E')$.

Since $\delta_i \in GD(D^t, E)$ for all $i \in I$, we have $\neg Justif(\delta_i) \notin E$. By (43), this implies $\neg Justif(\delta_i) \notin E'$.

By Definition A.1, we get $Conseq(\delta_i) \in \Gamma(E')$. That is, either $Conseq(\delta) \wedge \mathsf{ap}(n_\delta) \in \Gamma(E')$ or $\mathsf{bl}(n_\delta) \in \Gamma(E')$.

Thus, we obtain $v \in \Gamma(E')$, the desired contradiction.

- For $v \in \{\mathsf{ok}(n_\delta) \mid \delta \in D\}$, we draw on what we have just shown in Lemma 5 and Theorem 4.2.8. This gives $v \in \Gamma(E')$, a contradiction.

Since both E' and $\Gamma(E')$ are deductively closed, we get that $E' \subseteq \Gamma(E')$.

This completes the proof showing that E' is an extension of $\mathcal{D}((D, W \cup \{n_\delta \prec n_{\delta'} \mid (\delta, \delta') \in <\}))$.

if part Let E be an extension of $\mathcal{D}((D, W \cup \{n_\delta \prec n_{\delta'} \mid (\delta, \delta') \in <\}))$. Define

$$\begin{array}{rcl} E' = Th(W^t & \cup & \{\mathsf{ok}(n_\delta) \mid \delta \in D\} \\ & \cup & \{Conseq(\delta) \wedge \mathsf{ap}(n_\delta) \mid \delta_a \in GD(D^d, E)\} \\ & \cup & \{\mathsf{bl}(n_\delta) \mid \delta_{b_i} \in GD(D^d, E), i = 1, 2\} \\ & \cup & \left\{\neg(n_\delta \prec n_{\delta'}) \mid \frac{:\neg(n_\delta \prec n_{\delta'}), (n_\delta \prec n_{\delta'})}{(n_\delta \not\prec n_{\delta'})} \in GD(D^d, E)\right\} \Big) \end{array}$$

The rest of the proof continues in analogy to that given in the only-if-part; it is therefore omitted for brevity. ∎

Proof 5.2 The result follows from Theorem 5.1 and Corollary 4.2. ∎

Proof 5.3 Analogous to Proof 4.3. ∎

Proof 7.1

if part Let E be an extension of $(D, W, <)$ according to [Brewka and Eiter, 2000].

Wlog. we stipulate for every $\delta \in D$ that $\delta \in GD(D, E)$ iff $Conseq(\delta) \in E$. This condition is easily enforced by substituting each default $\delta = \frac{\alpha:\beta}{\gamma} \in D$ by $\frac{\alpha:\beta}{\gamma \wedge p_\delta}$, where p_δ is a new unique atom. This facilitates the treatment of generating defaults $\delta \in GD(D, E)$ that are "inactive" in some set $E_i \subseteq E$ because their consequent already belongs to E_i, viz. $Conseq(\delta) \in E_i$. We thus ensure that $GD(D, E)$ is identical to the set of actually applied rules in [Brewka and Eiter, 2000, Definition 4].

Since E is by definition also a regular extension of (D, W), we show next that E is $<$-preserving. To this end, consider the application sequence $\langle \delta_i \rangle_{i \in I}$ of $GD(D, E)$ induced by [Brewka and Eiter, 2000, Definition 4]. We must distinguish the following two cases:

1. Consider $\delta_i < \delta_j$. Assume $i < j$.

   Since $\delta_i \in GD(D, E)$ and $\delta_j \in GD(D, E)$, we have $\neg Justif(\delta_i) \notin E$ and $\neg Justif(\delta_j) \notin E$. By monotonicity, we get for $E_{i-1} \subseteq E$ that $\neg Justif(\delta_i) \notin E_{i-1}$ and $\neg Justif(\delta_j) \notin E_{i-1}$.

   By what we assume wlog we moreover have $Conseq(\delta_i) \notin E_{i-1}$ and $Conseq(\delta_j) \notin E_{i-1}$.

Hence both δ_i and δ_j are active in E_i according to [Brewka and Eiter, 2000].

Applying δ_i at this stage is a contradiction to $\delta_i < \delta_j$. Hence, $j < i$

2. Consider $\delta_i < \delta$, where $\delta \in D \setminus GD(D, E)$.

   By [Brewka and Eiter, 2000, Proposition 2], there is a set $K_\delta \subseteq \{\delta' \in GD(D, E) \mid \delta' <' \delta\}$ (where $<'$ is a total extension of $<$) such that

   $$W \cup Conseq(K_\delta) \models \neg Justif(\delta)$$

   Clearly $k < i$ for all $\delta_k \in K_\delta$. Hence, we obtain $W \cup Conseq(\{\delta_0, \ldots, \delta_{i-1}\}) \vdash \neg Justif(\delta)$.

By Theorem 4.4, there is then an extension E' of $\mathcal{T}((D, W, <))$ such that $E = E' \cap \mathcal{L}$.

only-if part This direction is a special case of Theorem 7.2. ∎

Proof 7.2 Let E' be a preferred extension of $\mathcal{T}((D, W, <)) = (D', W')$. Then, according to Theorem 4.4, $E = E' \cap \mathcal{L}$ is a $<$-preserving extension of (D, W). That is, there exists a grounded enumeration $\langle \delta_i \rangle_{i \in I}$ of $GD(D, E)$ such that for all $i, j \in I$ and $\delta \in D \setminus GD(D, E)$, we have that

1. if $\delta_i < \delta_j$ then $j < i$ and
2. if $\delta_i < \delta$ then $Prereq(\delta) \notin E$ or $W \cup Conseq(\{\delta_0, \ldots, \delta_{i-1}\}) \vdash \neg Justif(\delta)$.

Consider $\delta \in D$ with $Prereq(\delta) \in E$ and $Conseq(\delta) \notin E$. Clearly, $\delta \in D \setminus GD(D, E)$. Hence, there is some minimal $k \in I$ such that $W \cup Conseq(\{\delta_0, \ldots, \delta_k\}) \vdash \neg Justif(\delta)$ and $\delta_l \not< \delta$ for $l \in \{1, \ldots, k\}$. Now, consider a total extension $<'$ of $<$, where $\delta <' \delta_l$ for $l \in \{1, \ldots, k\}$. Given that E is a regular extension of (D, W) and that for any $\delta \in D$ with $Prereq(\delta) \in E$ and $Conseq(\delta) \notin E$ there is some $k \in I$ such that $W \cup Conseq(\{\delta_0, \ldots, \delta_k\}) \vdash \neg Justif(\delta)$ where $\delta <' \delta_l$ for $l \in \{1, \ldots, k\}$, we get according to [Brewka and Eiter, 2000, Proposition 2] that E is a preferred extension of $(D, W, <')$ according to [Brewka and Eiter, 2000]. Since, $<'$ is a total extension of $<$, E is furthermore a preferred extension of $(D, W, <)$ according to [Brewka and Eiter, 2000]. ∎

References

[Baader and Hollunder, 1993a] F. Baader and B. Hollunder. How to prefer more specific defaults in terminological default logic. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 669–674, 1993.

[Baader and Hollunder, 1993b] F. Baader and B. Hollunder. How to prefer more specific defaults in terminological default logic. Technical Report RR-92-58, DFKI, December 1993.

[Boutilier, 1992] C. Boutilier. What is a default priority? In J. Glasgow and B. Hadley, editors, *Proceedings of the Canadian Artificial Intelligence Conference*, pages 148–155. Morgan Kaufmann Publishers, 1992.

[Brewka and Eiter, 1998] G. Brewka and T. Eiter. Preferred answer sets for extended logic programs. In A. Cohn, L. Schubert, and S. Shapiro, editors, *Proceedings of the Sixth International Conference on the Principles of Knowledge Representation and Reasoning*, pages 86–97. Morgan Kaufmann Publishers, 1998.

[Brewka and Eiter, 2000] G. Brewka and T. Eiter. Prioritizing default logic. In St. Hölldobler, editor, *Intellectics and Computational Logic — Papers in Honour of Wolfgang Bibel*. Kluwer Academic Publishers, 2000. To appear.

[Brewka and Gordon, 1994] G. Brewka and T. Gordon. How to buy a porsche: An approach to defeasible decision making. In *AAAI-94 Workshop on Computational Dialectics*, pages 28–38, Seattle, WA, July 1994. AAAI Press.

[Brewka, 1994a] G. Brewka. Adding priorities and specificity to default logic. In L. Pereira and D. Pearce, editors, *European Workshop on Logics in Artificial Intelligence (JELIA'94)*, Lecture Notes in Artificial Intelligence, pages 247–260. Springer Verlag, 1994.

[Brewka, 1994b] G. Brewka. Reasoning about priorities in default logic. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, volume 2, pages 940–945. The AAAI Press/The MIT Press, 1994.

[Brewka, 1996] G. Brewka. Well-founded semantics for extended logic programs with dynamic preferences. *Journal of Artificial Intelligence Research*, 4:19–36, 1996.

[Delgrande and Schaub, 1994] J. Delgrande and T. Schaub. A general approach to specificity in default reasoning. In J. Doyle, P. Torasso, and E. Sandewall, editors, *Proceedings of the Fourth International Conference on the Principles of Knowledge Representation and Reasoning*, pages 146–157. Morgan Kaufmann Publishers, 1994.

[Delgrande and Schaub, 1998] J. Delgrande and T. Schaub. Reasoning with sets of preferences in default logic. In H.-Y. Lee and H. Motoda, editors, *Pacific Rim International Conference on Artificial Intelligence*, volume 1531 of *Lecture Notes in Artificial Intelligence*, pages 134–145. Springer Verlag, 1998.

[Delgrande and Schaub, 2000] J.P. Delgrande and T. Schaub. The role of default logic in knowledge representation. In J. Minker, editor, *Logic-Based Artificial Intelligence*. Kluwer Academic Publishers, 2000.

[Delgrande *et al.*, 2000a] J. Delgrande, T. Schaub, and H. Tompits. Logic programs with compiled preferences. In C. Baral and M. Truszczyński, editors, *Proceedings of the Eighth International Workshop on Non-Monotonic Reasoning*. `arXiv.org` e-Print archive, 2000.

[Delgrande *et al.*, 2000b] J. Delgrande, T. Schaub, and H. Tompits. Logic programs with compiled preferences. In W. Horn, editor, *Proceedings of the European Conference on Artificial Intelligence*, pages 392–398. IOS Press, 2000.

[Doyle and Wellman, 1991] J. Doyle and M.P. Wellman. Impediments to universal preference-based default theories. *Artificial Intelligence*, 49(1-3):97–128, 1991.

[Eiter *et al.*, 1997] T. Eiter, N. Leone, C. Mateis, G. Pfeifer, and F. Scarcello. A deductive system for nonmonotonic reasoning. In J. Dix, U. Furbach, and A. Nerode, editors, *Proceedings of the Fourth International Conference on Logic Programming and Non-Monotonic Reasoning*, volume 1265 of *Lecture Notes in Artificial Intelligence*, pages 363–374. Springer Verlag, 1997.

[Etherington and Reiter, 1983] D.W. Etherington and R. Reiter. On inheritance hierarchies with exceptions. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, pages 104–108. Morgan Kaufmann Publishers, 1983.

[Geffner and Pearl, 1992] H. Geffner and J. Pearl. Conditional entailment: Bridging two approaches to default reasoning. *Artificial Intelligence*, 53(2-3):209–244, 1992.

[Gelfond and Lifschitz, 1991] M. Gelfond and V. Lifschitz. Classical negation in logic programs and deductive databases. *New Generation Computing*, 9:365–385, 1991.

[Gelfond and Son, 1997] M. Gelfond and T. Son. Reasoning with prioritized defaults. In J. Dix, L. Pereira, and T. Przymusinski, editors, *Third International Workshop on Logic Programming and Knowledge Representation*, volume 1471 of *Lecture Notes in Computer Science*, pages 164–223. Springer Verlag, 1997.

[Gordon, 1993] T. Gordon. *The pleading game: An Artificial Intelligence Model of Procedural Justice*. Dissertation, Technische Hochschule Darmstadt, Alexanderstraße 10, D-64283 Darmstadt, Germany, 1993.

[Grosof, 1991] B. Grosof. Generalizing prioritization. In J. Allen, R. Fikes, and E. Sandewall, editors, *Proceedings of the Second International Conference on the Principles of Knowledge Representation and Reasoning*. Morgan Kaufmann Publishers, 1991.

[Junker, 1997] U. Junker. A cumulative model semantics for dynamic preferences on assumptions. In M. Pollack, editor, *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 162–167. Morgan Kaufmann Publishers, 1997.

[Lifschitz, 1985] V. Lifschitz. Closed-world databases and circumscription. *Artificial Intelligence*, 27:229–235, 1985.

[Marek and Truszczyński, 1993] V. Marek and M. Truszczyński. *Nonmonotonic logic: context-dependent reasoning*. Artifical Intelligence. Springer Verlag, 1993.

[McCarthy, 1986] J. McCarthy. Applications of circumscription to formalizing commonsense knowledge. *Artificial Intelligence*, 28:89–116, 1986.

[Niemelä and Simons, 1997] I. Niemelä and P. Simons. Smodels: An implementation of the stable model and well-founded semantics for normal logic programs. In J. Dix, U. Furbach, and A. Nerode, editors, *Proceedings of the Fourth International Conference on Logic Programing and Nonmonotonic Reasoning*, pages 420–429. Springer, 1997.

[Papadimitriou and Sideri, 1994] C. Papadimitriou and M. Sideri. Default theories that always have extensions. *Artificial Intelligence*, 69:347–357, 1994.

[Pearl, 1990] J. Pearl. System Z: A natural ordering of defaults with tractable applications to nonmonotonic reasoning. In R. Parikh, editor, *Proc. of the Third Conference on Theoretical Aspects of Reasoning About Knowledge*, pages 121–135, Pacific Grove, Ca., 1990. Morgan Kaufmann Publishers.

[Poole, 1988] D. Poole. A logical framework for default reasoning. *Artificial Intelligence*, 36:27–47, 1988.

[Reiter and Criscuolo, 1981] R. Reiter and G. Criscuolo. On interacting defaults. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 270–276, Vancouver, B.C., 1981.

[Reiter, 1980] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13(1-2):81–132, 1980.

[Rintanen, 1995] J. Rintanen. On specificity in default logic. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1474–1479. Morgan Kaufmann Publishers, 1995.

[Touretzky *et al.*, 1987] D.S. Touretzky, J.F. Horty, and R.H. Thomason. A clash of intuitions: The current state of nonmonotonic multiple inheritance systems. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 476–482, Milan, 1987.

[Zhang and Foo, 1997] Y. Zhang and N. Foo. Answer sets for prioritized logic programs. In J. Maluszynski, editor, *Proceedings of the International Symposium on Logic Programming (ILPS-97)*, pages 69–84. MIT Press, 1997.