Splitting an operator: An algebraic modularity result and its application to auto-epistemic logic

Joost Vennekens and **David Gilis** and **Marc Denecker**
Department of Computer Science, K.U. Leuven
Celestijnenlaan 200A
B-3001 Leuven, Belgium

Abstract

It is well known that it is possible to *split* certain auto-epistemic theories under the semantics of expansions, i.e. to divide such a theory into a number of different "levels", such that the models of the entire theory can be constructed by incrementally constructing models for each level. Similar results exist for other non-monotonic formalisms, such as logic programming and default logic. In this work, we present a general, algebraic theory of splitting under a £xpoint semantics. Together with the framework of *approximation theory*, a general £xpoint theory for arbitrary operators, this gives us a uniform and powerful way of deriving splitting results for each logic with a £xpoint semantics. We demonstrate the usefulness of this approach, by applying our results to auto-epistemic logic.

Introduction

An important aspect of human reasoning is that it is often incremental in nature. When dealing with a complex domain, we tend to initially restrict ourselves to a small subset of all relevant concepts. Once these "basic" concepts have been £gured out, we then build another, more "advanced", layer of concepts on this knowledge. A quite illustrative example of this can be found in most textbooks on computer networking. These typically present a seven-layered model of the way in which computers communicate. First, in the so-called physical layer, there is only talk of hardware and concepts such as wires, cables and electronic pulses. Once these low-level issues have been dealt with, the resulting knowledge becomes a *£xed* base, upon which a new layer, the data-link layer, is built. This no longer considers wires and cables and so on, but rather talks about packages of information travelling from one computer to another. Once again, after the workings of this layer have been £gured out, this information is "taken for granted" and becomes part of the foundation upon which a new layer is built. This process continues all the way up to a seventh layer, the application layer, and together all of these layers describe the operation of the entire system.

In this paper, we investigate a formal equivalent of this method. More speci£cally, we address the question of whether a formal theory in some non-monotonic language can be *split* into a number of different levels or *strata*, such that the formal semantics of the entire theory can be constructed by succesively constructing the semantics of the various strata. (We use the terms "strati£cation" and "splitting" interchangeably to denote a division into a number of different levels. This is a more general use of both these terms, than in literature such as (Apt, Blair, & Walker 1988) and (Gelfond 1987).) Such strati£cations are interesting from both a practical and a more theoretical, knowledge representational point of view. For instance, computing models of a strati£ed version of a theory is often signi£cantly faster than computing models of the original theory. Furthermore, in order to be able to build and maintain large knowledge bases, it is crucial to know which parts of a theory can be analysed or constructed independently and, conversely, whether combining several correct theories will have any unexpected side-effects.

It is therefore not surprising that this issue has already been intensively studied. Indeed, splitting results have been proven for auto-epistemic logic under the semantics of expansions (Gelfond & Przymusinska 1992), default logic under the semantics of extensions (Turner 1996) and various kinds of logic programs under the stable model semantics (Lifschitz & Turner 1994; Erdoğan & Lifschitz 2004). In all of these works, strati£ability is seen as a syntactical property of a theory in a certain language under a certain formal semantics.

In our work, we take a different approach to studying this topic. The semantics of several (non-monotonic) logics can be expressed through £xpoint characterizations in some lattice of semantic structures. In such a semantics, the meaning of a theory is given by the way in which it revises a proposed "state of affairs"; its models are those states which no longer have to be revised. Knowing such a revision operator for a theory should suf£ce to know whether it is strati£able: this will be the case if and only if no higher levels are ever used to revise the state of affairs concerning lower-level concepts. This motivates us to study the strati£cation of these revision operators themselves. As such, we are able to develop a general theory of strati£cation at an abstract, algebraic level and apply its results to each formalism which has a £xpoint semantics.

This approach is especially powerful when combined with the framework of *approximation theory*, a general £xpoint theory for arbitrary operators, which has already proved highly useful in the study of non-monotonic reasoning. It naturally captures, for instance, (most of) the common semantics of logic programming (Denecker, Marek, & Truszczynski 2000), auto-epistemic logic (Denecker, Marek, & Truszczynski 2003) and default logic (Denecker, Marek, & Truszczynski 2003). As such, studying strati£cation within this framework, allows our abstract results to be directly and easily applicable to logic programming, auto-epistemic logic and default logic.

Studying strati£cation at this more *semantical* level has three distinct advantages. First of all, it avoids duplication of effort, as the same algebraic theory takes care of strati£cation in logic programming, auto-epistemic logic, default logic and indeed any logic with a £xpoint semantics. Secondly, our results can be used to easily extend existing results to other (£xpoint) semantics of the aforementioned languages. Finally, our work also offers greater insight into the general principles underlying various known strati£cation results, as we are able to study this issue in itself, free of being restricted to a particular syntax or semantics.

In (Vennekens, Gilis, & Denecker 2004), we presented our basic theory of strati£able operators and used these results to derive splitting results for logic programming. In this work, we deal with auto-epistemic logic. In doing so, we also extend the algebriac results from (Vennekens, Gilis, & Denecker 2004) to a larger class of operators.

Preliminaries: Approximation theory

This presentation of approximation theory is based on (Denecker, Marek, & Truszczynski 2000), but introduces a slightly more general de£nition of approximations. Let $\langle L, \leq \rangle$ be a lattice. An element (x, y) of the square L^2 of the domain of such a lattice, can be seen as denoting an interval $[x, y] = \{z \in L \mid x \leq z \leq y\}$. Using this intuition, we can de£ne a *precision* order $\leq_p$ on the set L^2 from the order $\leq$ on L: for each $x, y, x', y' \in L$,

$$(x, y) \leq_p (x', y') \text{ iff } x \leq x' \text{ and } y' \leq y.$$

Indeed, if $(x, y) \leq_p (x', y')$, then $[x, y] \supseteq [x', y']$. It can easily be shown that $\langle L^2, \leq_p \rangle$ is also a lattice, which we will call the *bilattice* corresponding to L. Moreover, if L is complete, then so is L^2. As an interval $[x, x]$ contains precisely one element, elements (x, x) of L^2 are called *exact*.

Approximation theory is based on the study of operators on bilattices L^2 which are monotone w.r.t. the precision order $\leq_p$. Such operators are called *approximations*. For an approximation A and elements x, y of L, we denote by $A^1(x, y)$ and $A^2(x, y)$ the unique elements of L, for which $A(x, y) = (A^1(x, y), A^2(x, y))$. An approximation *approximates* an operator O on L if for each $x \in L$, $A(x, x)$ contains $O(x)$, i.e. $A^1(x, x) \leq O(x) \leq A^2(x, x)$. An *exact* approximation is one which maps exact elements to exact elements, i.e. $A^1(x, x) = A^2(x, x)$ for all $x \in L$. Each exact approximation approximates a unique operator O on L, namely that which maps each $x \in L$ to $A^1(x, x)$. An approximation is *symmetric* if for each pair $(x, y) \in L^2$, if $A(x, y) = (x', y')$ then $A(y, x) = (y', x')$. Each symmetric approximation is also exact.

For an approximation A on L^2, the following two operators on L can be de£ned: the operator $A^1(\cdot, y)$ maps an element $x \in L$ to $A^1(x, y)$, i.e. $A^1(\cdot, y) = \lambda x.A^1(x, y)$, and the function $A^2(x, \cdot)$ maps an element $y \in L$ to $A^1(x, y)$, i.e. $A^2(x, \cdot) = \lambda y.A^2(x, y)$. As all such operators are monotone, they all have a unique least £xpoint. We de£ne an operator $C_A^{\downarrow}$ on L, which maps each $y \in L$ to $lfp(A^1(\cdot, y))$; similarly the operator $C_A^{\uparrow}$ maps each $x \in L$ to $lfp(A^2(x, \cdot))$. $C_A^{\downarrow}$ and $C_A^{\uparrow}$ are called, respectively, the *lower stable operator* and *upper stable operator* of A. They are both anti-monotone. Combining these two operators, the operator $\mathcal{C}_A$ on L^2 maps each pair (x, y) to $(C_A^{\downarrow}(y), C_A^{\uparrow}(x))$. This operator is called the *partial stable operator* of A. Because the lower and upper partial stable operators are anti-monotone, the partial stable operator $\mathcal{C}_A$ is monotone. Note that if an approximation A is symmetric, its lower and upper partial stable operators are equal, i.e. $C_A^{\downarrow} = C_A^{\uparrow}$. An approximation A de£nes a number of different £xpoints: the least £xpoint of an approximation A is called its *Kripke-Kleene £xpoint*, £xpoints of its partial stable operator $\mathcal{C}_A$ are *stable £xpoints* and the least £xpoint of $\mathcal{C}_A$ is called the *well-founded £xpoint* of A. As shown in (Denecker, Marek, & Truszczynski 2000) and (Denecker, Marek, & Truszczynski 2003), these £xpoints correspond to various semantics of logic programming, auto-epistemic logic and default logic.

Finally, it should be noted that an approximation as de£ned in (Denecker, Marek, & Truszczynski 2000) corresponds to our de£nition of a *symmetric* approximation.

Strati£cation of operators

In this section, we present an algebraic theory of strati£cation. First, we brie¤y restate results from (Vennekens, Gilis, & Denecker 2004), dealing with strati£cation of operators and approximations on product lattices. These results are then extended to a larger class of operators.

Product lattices

We begin by de£ning the notion of a *product set*, which is a generalization of the well-known concept of cartesian products. Let I be a set, which we will call the *index set* of the product set, and for each $i \in I$, let $\langle S_i, \leq_i \rangle$ be a partially ordered set. The *product set* $S = \bigotimes_{i \in I} S_i$ is the following set of functions, selecting one element from each set S_i:

$$\bigotimes_{i \in I} S_i = \{f \mid f : I \to \bigcup_{i \in I} S_i \text{ such that } \forall i \in I : f(i) \in S_i\}$$

The *product order* $\leq_\otimes$ on S is de£ned by $\forall x, y \in S : x \leq_\otimes y$ iff $\forall i \in I : x(i) \leq_i y(i)$. It can easily be shown that if all of the partially ordered sets S_i are (complete) lattices, the product set S, together with its product order $\leq_\otimes$, is also a (complete) lattice. The pair $\langle S, \leq_\otimes \rangle$ is called the *product lattice* of lattices S_i. We will only consider product lattices with a *well-founded* index set, i.e. index sets I with a partial order $\preceq$ such that each non-empty subset of I has a $\preceq$-minimal element. This allows us to use inductive arguments in dealing with elements of product lattices.

In the next sections, the following notations will be used. For a function $f : A \to B$ and a subset A' of A, we denote by $f|_{A'}$ the restriction of f to A', i.e. $f|_{A'} : A' \to B : a' \mapsto$

$f(a')$. For an element x of a product lattice $\otimes_{i\in I} L_i$ and an $i \in I$, we abbreviate $x|_{\{j\in I | j \preceq i\}}$ by $x|_{\preceq i}$. We also use abbreviations $x|_{\prec i}$, $x|_{\not\preceq i}$ and $x|_i$. If i is a minimal element of the well-founded set I, $x|_{\prec i}$ is the empty function. A set $\{x|_{\preceq i} \mid x \in L\}$, ordered by the appropriate restriction $\leq_\otimes|_{\preceq i}$ of the product order, is also a lattice. Clearly, this sublattice of L is isomorphic to the product lattice $\otimes_{j\preceq i} L_i$. We denote this sublattice by $L|_{\preceq i}$ and use a similar notation $L|_{\prec i}$ for $\otimes_{j \prec i} L_i$. If f, g are functions $f : A \to B$, $g : C \to D$ and the domains A and C are disjoint, we denote by $f \sqcup g$ the function from $A \cup C$ to $B \cup D$, such that for all $a \in A$, $(f \sqcup g)(a) = f(a)$ and for all $c \in C$, $(f \sqcup g)(c) = g(c)$. We call $f \sqcup g$ an *extension* of f. For each element x of a product lattice L and each index $i \in I$, the extension $x|_{\prec i} \sqcup x|_i$ of $x|_{\prec i}$ is clearly equal to $x|_{\preceq i}$. For ease of notation, we will sometimes simply write $x(i)$ instead of $x|_i$ in such expressions, i.e. we will identify an element a of the ith lattice L_i with the function from $\{i\}$ to L_i which maps i to a. Similarly, $x|_{\prec i} \sqcup x(i) \sqcup x|_{\not\preceq i} = x$.

Operators on product lattices

In this section, we de£ne and discuss strati£able operators on product lattices. For proof of the results presented in this section, we refer to (Vennekens, Gilis, & Denecker 2004).

Intuitively, an operator O on a product lattice $L = \otimes_{i\in I} L_i$ is strati£able over the order $\preceq$ of its index set I, if the value $(O(x))(i)$ of $O(x)$ in the ith stratum only depends on values $x(j)$ for which $j \preceq i$.

De£nition 1. An operator O on a product lattice L is *strati£able* iff $\forall x, y \in L, \forall i \in I$: if $x|_{\preceq i} = y|_{\preceq i}$ then $O(x)|_{\preceq i} = O(y)|_{\preceq i}$.

It is possible to characterize strati£ablity in a more constructive way. The following proposition shows that strati£ablity of an operator O on a product lattice L is equivalent to the existence of a family of operators on each lattice L_i (one for each partial element u of $L|_{\prec i}$), which mimics the behaviour of O on this lattice.

Proposition 1. *Let O be an operator on a product lattice L. O is strati£able iff for each $i \in I$ and $u \in L|_{\prec i}$ there exists a unique operator O_i^u on L_i, such that for all $x \in L$: if $x|_{\prec i} = u$ then $(O(x))(i) = O_i^u(x(i))$.*

These operators O_i^u are called the *components* of O. As shown by the following theorem, their existence allows us to incrementally (w.r.t. the well-founded order $\preceq$ on the index set I) construct the £xpoints and least £xpoint of a strati£able operator.

Theorem 1. *Let O be a strati£able operator on a complete product lattice L. Then for each $x \in L$:*

$$x \textit{ is a £xpoint of } O$$
$$iff$$
$$\forall i \in I : x(i) \textit{ is a £xpoint of } O_i^{x|_{\prec i}}.$$

Moreover, if O is monotone, then the components of O are also monotone and

$$x \textit{ is the least £xpoint of } O$$
$$iff$$
$$\forall i \in I : x(i) \textit{ is the least £xpoint of } O_i^{x|_{\prec i}}.$$

It is worth noting that — as shown in (Vennekens, Gilis, & Denecker 2004) — the well-foundedness of the order $\preceq$ on the index set I is a necessary condition for this last equivalence to hold.

Using this result, it is possible to construct the £xpoints (least £xpoint) of a strati£able operator O, by means of the following procedure. First, we construct the £xpoints (least £xpoints) of the components O_i, for all minimal elements i of the index set I. These (least) £xpoints then allow the construction of operators O_j^u for all indices j on the "next level" (i.e. such that all indices $k \prec j$ are minimal), using the previously construced (least) £xpoints as $u \in L|_{\prec j}$. This process can be repeated until all indices have been dealt with. As the previous theorem shows, combining the (least) £xpoints of all levels then yields the £xpoints (least £xpoint) of O itself.

In the preliminaries, we introduced the Kripke-Kleene, stable and well-founded £xpoints of an approximation. We will now investigate the relation between these various £xpoints of an approximation and its components. In doing so, it will be convenient to switch to an alternative representation of the bilattice L^2 of a product lattice $L = \otimes_{i\in I} L_i$. Indeed, this bilattice is clearly isomorphic to the structure $\otimes_{i\in I} L_i^2$, i.e. to a product lattice of bilattices. From now on, we will not distinguish between these two representations.

Because of this isomorphism, we can view an approximation as a monotone operator on a product lattice. Therefore, theorem 1 already provides a way of constructing the £xpoints and the least (Kripke-Kleene) £xpoint of a strati£able approximation A, by means of its components $A_i^{(u,v)}$. As mentioned in the preliminaries, an exact approximation A approximates a unique operator O. Because $A(x, x) = (O(x), O(x))$ for each $x \in L$, it is clear that if A is strati£able, then O is strati£able as well. This leaves only the stable and well-founded £xpoints of A to be investigated. It turns out that these can be incrementally constructed from the partial stable operators $\mathcal{C}_{A_i^{(u,v)}}$ of the components $A_i^{(u,v)}$ of A.

Theorem 2. *Let L be a product lattice and let $A : L^2 \to L^2$ be a strati£able approximation. Then for each element (x, y) of L^2: (x, y) is a (least) £xpoint of $\mathcal{C}_A$ iff $\forall i \in I$: $(x, y)(i)$ is a (least) £xpoint of $\mathcal{C}_{A_i^{(x,y)|_{\prec i}}}$.*

Putting all of this together, the main results of this section can be summarized as follows. If A is a strati£able approximation on a product lattice L, then a pair (x, y) is a £xpoint, Kripke-Kleene £xpoint, stable £xpoint or well-founded £xpoint of A iff for each $i \in I$, $(x(i), y(i))$ is a £xpoint, Kripke-Kleene £xpoint, stable £xpoint or well-founded £xpoint of the component $A_i^{(x,y)|_{\prec i}}$ of A. Moreover, if A is exact then an element $x \in L$ is a £xpoint of the unique operator O approximated by A iff for each $i \in I$, $(x(i), x(i))$

is a £xpoint of the component $A_i^{(x,x)|_{\prec i}}$ of A. These characterizations give us a way of incrementally constructing each of these £xpoints.

Operators on other lattices

The theory developed in (Vennekens, Gilis, & Denecker 2004) and summarized in the previous sections allows us to incrementally construct £xpoints of operators on product lattices. However, not every operator is (isomorphic to) an operator on a (non-trivial) product lattice. So, the question arises whether it is also possible to incrementally construct the £xpoints of an operator O on a lattice L, which is *not* a product lattice. Clearly, this could be done if the £xpoints of O were uniquely determined by the £xpoints of some other operator $\tilde{O}$ on a lattice $\tilde{L}$, where $\tilde{L}$ *is* a product lattice.

This section will investigate similarity conditions which suf£ce to ensure the existence of such a correspondence between the £xpoints $fp(\tilde{O})$ of an operator $\tilde{O}$ on $\tilde{L}$ and those of an operator O on L. More precisely, we will determine a number of properties for a function k from $\tilde{L}$ to L, such that applying k to the £xpoints of $\tilde{O}$ yields the £xpoints of O. For a set S, function $f : S \to T$ and $t \in T$, we denote the set $\{f(s) \mid s \in S\}$ by $f(S)$ and the set $\{s \in S \mid f(s) = t\}$ by $f^{-1}(t)$. Using these notations, we can rephrase our goal as determining conditions which ensure that $k(fp(\tilde{O})) = fp(O)$.

De£nition 2. Let $\tilde{L}$ and L be lattices, $\tilde{O}$ an operator on $\tilde{L}$ and O an operator on L. If there exists a function k from $\tilde{L}$ to L, such that $k \circ \tilde{O} = O \circ k$ and $O(L) \subseteq k(\tilde{L})$, $\tilde{O}$ *k-mimics* O. This is denoted by $\tilde{O} \overset{k}{\propto} O$.

Clearly, if $\tilde{O} \overset{k}{\propto} O$, then $k(fp(\tilde{O})) \subseteq fp(O)$. However, in general the reverse inclusion does not hold. Indeed, all that can be shown, is that for each £xpoint x of O, $k^{-1}(x)$ is not empty and closed under application of $\tilde{O}$. To ensure that $k(fp(\tilde{O})) \supseteq fp(O)$ holds as well, we will use an additional similarity condition between the lattices $\tilde{L}$ and L, such that successive applications of a monotone $\tilde{O}$ to a certain element of $k^{-1}(x)$ are sure to reach a £xpoint which is still in $k^{-1}(x)$.

De£nition 3. Let $\tilde{L}$ and L be complete lattices. If there exists a function k from $\tilde{L}$ to L, such that each non-empty set $k^{-1}(x)$, with $x \in L$, has a central element (i.e. one which is comparable to all other elements of $k^{-1}(x)$) and k is chain-continuous[1] then $\tilde{L}$ is *k-similar* to L. This is denoted by $\tilde{L} \overset{k}{\rightsquigarrow} L$.

Theorem 3. *Let $\tilde{L}$ and L be complete lattices, $\tilde{O}$ a monotone operator on $\tilde{L}$ and O a monotone operator on L, such that $\tilde{L} \overset{k}{\rightsquigarrow} L$ and $\tilde{O} \overset{k}{\propto} O$. Then*

1. $k(fp(\tilde{O})) = fp(O)$, *and*
2. $k(lfp(\tilde{O})) = lfp(O)$.

Proof. Let $\tilde{L}, \tilde{O}, L$ and O be as above.

1. Let $x \in fp(O)$ and let $\tilde{c}$ be a central element of $k^{-1}(x)$. As $\tilde{O}$ is monotone and $\tilde{c}$ is a central element, the elements $(O^{\beta}(\tilde{c}))_{\beta<\alpha}$ form a chain for each ordinal α. As such, for some ordinal γ, $\tilde{O}^{\gamma}(\tilde{c})$ is a £xpoint of $\tilde{O}$. Moreover, because $O(\tilde{c}) \in k^{-1}(x)$ and k is chain continuous, by induction, for each ordinal α, $\tilde{O}^{\alpha}(\tilde{c}) \in k^{-1}(x)$. Therefore $k(\tilde{O}^{\gamma}(\tilde{c})) = x$.
2. Because each pair of comparable elements of $\tilde{L}$ forms a chain, chain continuity of k implies that k is order-preserving. This proves the correspondence between least £xpoints.

□

So far, we have shown that if a monotone operator $\tilde{O}$ k-mimics O, the £xpoints and the least £xpoint of O can be found by simply applying k to each £xpoint or, respectively, the least £xpoint of $\tilde{O}$. If the operator O is an approximation, however, we are also often also interested in its stable and well-founded £xpoints.

Proposition 2. *Let $\tilde{L}, L$ be lattices, such that $\tilde{L} \overset{k}{\rightsquigarrow} L$. Let $\overline{k}$ be the function from $\tilde{L}^2$ to L^2 which maps each pair $(\tilde{x}, \tilde{y})$ to $(k(\tilde{x}), k(\tilde{y}))$. Let $\tilde{A}$ be an approximation on $\tilde{L}^2$ and A an approximation on L^2, such that $\tilde{A} \overset{\overline{k}}{\propto} A$. Then $\mathcal{C}_{\tilde{A}} \overset{\overline{k}}{\propto} \mathcal{C}_A$.*

Proof. Let $\tilde{L}, L, \tilde{A}, A, \overline{k}, k$ be as above. We will show that $C^{\downarrow}_{\tilde{A}} \overset{k}{\propto} C^{\downarrow}_{A}$. The proof of $C^{\uparrow}_{\tilde{A}} \overset{k}{\propto} C^{\uparrow}_{A}$ is similar. Because $C^{\downarrow}_{A}(L^2) \subseteq A^1(L^2)$, $C^{\downarrow}_{A}(L^2) \subseteq k(\tilde{L})$. By de£nition, for each $\tilde{y} \in \tilde{L}$, $k(C^{\downarrow}_{\tilde{A}}(\tilde{y})) = k(lfp(\tilde{A}^1(\cdot, \tilde{y})))$, which because of theorem 3 equals $lfp(k(\tilde{A}^1(\cdot, \tilde{y})) = lfp(A^1(\cdot, k(\tilde{y})))$. □

As each partial stable operator $\mathcal{C}_A$ is by de£nition monotone, this proposition shows that if an approximation A is $\overline{k}$-mimicked by an approximation $\tilde{A}$, its stable and well-founded £xpoints can be found by applying $\overline{k}$ to the stable £xpoints or, respectively, the well-founded £xpoint of $\tilde{A}$. Moreover, it is easy to see that if $\tilde{A}$ and A are exact approximations of, respectively, operators $\tilde{O}$ and O, then $\tilde{O} \overset{k}{\propto} O$ as well. Therefore, if $\tilde{O}$ is also monotone, then $fp(\tilde{O}) = k(fp(O))$.

Using the results

Our goal is to use the abstract results presented in this section to derive concrete splitting results for non-monotonic reasoning formalisms such as logic programming, autoepistemic logic and default logic. As noted earlier, for each of these formalisms there exists a class of approximations, such that the various kinds of £xpoints of the approximation A_T associated with a theory T correspond to the models of T under various semantics for this formalism.

[1] A totally ordered subset of a lattice is called a *chain*. A function k from a lattice $\tilde{L}$ to a lattice L is *chain-continuous* iff for each chain $(\tilde{x}_i)_{i<\alpha} \subseteq \tilde{L}$, $k(glb(\{\tilde{x}_i \mid i < \alpha\})) = glb(\{k(\tilde{x}_i) \mid i < \alpha\})$ and $k(lub(\{\tilde{x}_i \mid i < \alpha\})) = lub(\{k(\tilde{x}_i) \mid i < \alpha\})$.

As we showed in (Vennekens, Gilis, & Denecker 2004), in the case of logic programming these approximations operate on a (lattice isomorphic to) product lattice. Therefore, is was possible to derive splitting results for logic programming by the following method: First, we identi£ed syntactical conditions, such that for each logic program P satisfying these conditions, the corresponding approximation A_P is strati£able. Our results then show that the £xpoints, least £xpoint, stable £xpoints and well-founded £xpoint of such an approximation A_P (which correspond to the models of P under various semantics for logic programming) can be incrementally constructed from the components of A_P. Of course, in order for this result to be of any practical use, one also needs to be able to actually construct these components. Therefore, we also presented a procedure of deriving new programs P' from the original program P, such that the approximations associated with these programs P' are precisely those components.

For auto-epistemic logic and default logic the situation is, however, slightly more complicated, because the approximations which de£ne the semantics of these formalism do not operate on a product lattice. Therefore, in these cases, we cannot simply follow the above procedure. Instead, we need to preform the additional step of £rst £nding approximations $\tilde{A}_T$ on a product lattice "similar" to the original lattice, which "mimic" the original approximations A_T of theories T. The results of this section then show that the various kinds of £xpoints of A_T can be found from the corresponding £xpoints of $\tilde{A}_T$. Therefore, we can split theories T by stratifying these new approximations $\tilde{A}_T$ and, as these $\tilde{A}_T$ *are* operators on a product lattice, this can be done by the above procedure. In other words, we then just need to determine syntactical conditions which suf£ce to ensure that $\tilde{A}_T$ is strati£able and present a way of constructing new theories T' from T, such that the components of $\tilde{A}_T$ correspond to approximations associated with these new theories.

Application to auto-epistemic logic

In this section, we will £rst describe the syntax of auto-epistemic logic and give a brief overview, based on (Denecker, Marek, & Truszczynski 2003), of how a number of different semantics for this logic can be de£ned using concepts from approximation theory. Then, we will follow the methodology outlined in the £nal paragraph of the previous section to prove concrete splitting results for this logic.

Syntax and Semantics

Let $\mathcal{L}$ be the language of propositional logic based on a set of atoms Σ. Extending this language with a modal operator K, gives a language $\mathcal{L}_K$ of modal propositional logic. An auto-epistemic theory is a set of fomulas in this language $\mathcal{L}_K$. For such a formula φ, the subset of Σ containing all atoms which appear in φ, is denoted by $At(\varphi)$; atoms which appear in φ at least once outside the scope of the model operator K are called *objective* atoms of φ and the set of all objective atoms of φ is denoted by $At_O(\varphi)$. A *modal subformula* is a formula of the form $K(\psi)$, with ψ a formula.

To illustrate, consider the following example:

$$F = \{\varphi_1 = p \vee \neg Kp\ ;\ \ \varphi_2 = K(p \vee q) \vee q\}$$

The objective atoms $At_O(\varphi_2)$ of φ_2 are $\{q\}$, while the atoms $At(\varphi_2)$ are $\{p, q\}$. The formula $K(p \vee q)$ is a modal subformula of φ_2.

An *interpretation* is a subset of the alphabet Σ. The set of all interpretations of Σ is denoted by $\mathcal{I}_\Sigma$, i.e. $\mathcal{I}_\Sigma = 2^\Sigma$. A *possible world structure* is a set of interpretations, i.e. the set of all possible world structures $\mathcal{W}_\Sigma$ is de£ned as $2^{\mathcal{I}_\Sigma}$. Intuitively, a possible world structure sums up all "situations" which are possible. It therefore makes sense to order these according to inverse set inclusion to get a *knowledge order* $\leq_k$, i.e. for two possible world stuctures Q, Q', $Q \leq_k Q'$ iff $Q \supseteq Q'$. Indeed, if a possible world structure contains *more* possibilities, it actually contains *less* knowledge.

Following (Denecker, Marek, & Truszczynski 2003), we will de£ne the semantics of an auto-epistemic theory by an operator on the bilattice $\mathcal{B}_\Sigma = \mathcal{W}_\Sigma^2$. An element (P, S) of $\mathcal{B}_\Sigma$ is known as a *belief pair* and is called *consistent* iff $P \leq_k S$. In a consistent belief pair (P, S), P can be viewed as describing what must *certainly* be known, i.e. as giving an *under*estimate of what is known, while S can be viewed as denoting what might *possibly* be known, i.e. as giving an *over*estimate. Based on this intuition, there are two ways of estimating the truth of modal formulas according to (P, S): we can either be *conservative*, i.e. assume a formula is false unless we are sure it must be true, or we can be *liberal*, i.e. assume it is true unless we are sure it must be false. To conservatively estimate the truth of a formula $K\varphi$ according to (P, S), we simply have to check whether φ is surely known, i.e. whether φ is known in the underestimate P. To conservatively estimate the truth of a formula $\neg K\varphi$, on the other hand, we need to determine whether φ is de£nitely unknown; this will be the case if φ cannot possibly be known, i.e. if φ is not known in the overestimate S. The following de£nition extends these intuitions to reach a conservative estimate of the truth of arbitrary formulas. Note that the objective atoms of such a formula are simply interpreted by an interpretation $X \in \mathcal{I}_\Sigma$.

De£nition 4. For each $(P, S) \in \mathcal{B}_\Sigma, X \in \mathcal{I}_\Sigma$, $a \in \Sigma$ and formulas φ, φ_1 and φ_2, we inductively de£ne $\mathcal{H}_{(P,S),X}$ as:

- $\mathcal{H}_{(P,S),X}(a) = \mathbf{t}$ iff $a \in X$ for each atom a;
- $\mathcal{H}_{(P,S),X}(\varphi_1 \wedge \varphi_2) = \mathbf{t}$, iff $\mathcal{H}_{(P,S),X}(\varphi_1) = \mathbf{t}$ and $\mathcal{H}_{(P,S),X}(\varphi_2) = \mathbf{t}$;
- $\mathcal{H}_{(P,S),X}(\varphi_1 \vee \varphi_2) = \mathbf{t}$, iff $\mathcal{H}_{(P,S),X}(\varphi_1) = \mathbf{t}$ or $\mathcal{H}_{(P,S),X}(\varphi_2) = \mathbf{t}$;
- $\mathcal{H}_{(P,S),X}(\neg\varphi) = \neg\mathcal{H}_{(S,P),X}(\varphi)$;
- $\mathcal{H}_{(P,S),X}(K\varphi) = \mathbf{t}$ iff $\mathcal{H}_{(P,S),Y}(\varphi) = \mathbf{t}$ for all $Y \in P$;

It is worth noting that an evaluation $\mathcal{H}_{(Q,Q),X}(\varphi)$, i.e. one in which all that might possibly be known is also surely known, corresponds to the standard S_5 evaluation (Ch & van der Hoek 1995). Note also that the evaluation $\mathcal{H}_{(P,S),X}(K\varphi)$ of a modal subformula $K\varphi$ depends only on (P, S) and not on X. We sometimes emphasize this by writing $\mathcal{H}_{(P,S),\cdot}(K\varphi)$. Similarly, $\mathcal{H}_{(P,S),X}(\varphi)$ of an objective formula φ depends only on X and we sometimes write $\mathcal{H}_{(\cdot,\cdot),X}(\varphi)$.

As mentioned above, it is also possible to liberally estimate the truth of modal subformulas. Intuitively, we can do this by assuming that everything which might be known, *is* in fact known and that everything which might be unknown, i.e. which is not surely known, is in fact unknown. As such, to liberally estimate the truth of a formula according to a pair (P, S), it suf£ces to treat S as though it were describing what we surely know and P as though it were describing what we might know. In others words, it suf£ces to simply switch the roles of P and S, i.e. $\mathcal{H}_{(S,P),\cdot}$ provides a liberal estimate of the truth of modal formulas.

These two ways of evaluating formulas can be used to derive a new, more precise belief pair (P', S') from an original pair (P, S). First, we will focus on constructing the new overestimate S'. As S' needs to overestimate knowledge, it needs to contain as few interpretations as possible. This means that S' should consist of only those interpretations, which manage to satisfy the theory even if the truth of its modal subformulas is underestimated. So, $S' = \{X \in \mathcal{I}_\Sigma \mid \forall\varphi \in T : \mathcal{H}_{(P,S),X}(\varphi) = \mathbf{t}\}$. Conversly, to construct the new underestimate P', we need as many interpretations as possible. This means that P' should contain all interpretations which satisfy the theory, when liberally evaluating its modal subformulas. So, $P' = \{X \in \mathcal{I}_\Sigma \mid \forall\varphi \in T : \mathcal{H}_{(S,P),X}(\varphi) = \mathbf{t}\}$. These intuitions motivate the following de£nition of the operator $\mathcal{D}_T$ on $\mathcal{B}_\Sigma$:

$$\mathcal{D}_T(P, S) = (\mathcal{D}^u_T(S, P), \mathcal{D}^u_T(P, S))$$

with $\mathcal{D}^u_T(P, S) = \{X \in \mathcal{I}_\Sigma \mid \forall\varphi \in T : \mathcal{H}_{(P,S),X}(\varphi) = \mathbf{t}\}$.

This operator is an approximation (Denecker, Marek, & Truszczynski 2003). Moreover, since $\mathcal{D}^1_T(P, S) = \mathcal{D}^u_T(S, P) = \mathcal{D}^2_T(S, P)$ and $\mathcal{D}^2_T(P, S) = \mathcal{D}^u_T(P, S) = \mathcal{D}^1_T(S, P)$, it is by de£nition symmetric and therefore approximates a unique operator on $\mathcal{W}_\Sigma$, namely the operator D_T (Moore 1984), which maps each Q to $\mathcal{D}^u_T(Q, Q)$. As shown in (Denecker, Marek, & Truszczynski 2003), these operators de£ne a family of semantics for a theory T:

- £xpoints of D_T are *expansions* of T (Moore 1984),
- £xpoints of $\mathcal{D}_T$ are *partial expansions* of T (Denecker, Marek, & Truszczynski 1998),
- the least £xpoint of $\mathcal{D}_T$ is the *Kripke-Kleene £xpoint* of T (Denecker, Marek, & Truszczynski 1998),
- £xpoints of $C^{\downarrow}_{\mathcal{D}_T}$ are *extensions* (Denecker, Marek, & Truszczynski 2003) of T,
- £xpoints of $\mathcal{C}_{\mathcal{D}_T}$ are *partial extensions* (Denecker, Marek, & Truszczynski 2003) of T and
- the least £xpoint of $\mathcal{C}_{\mathcal{D}_T}$ is the *well-founded model* of T (Denecker, Marek, & Truszczynski 2003).

These various dialects of auto-epistemic logic differ in their treatment of "ungrounded" expansions (Konolige 1987), i.e. expansions which arize from cyclicities such as $Kp \rightarrow p$.

When calculating the models of a theory, it is often useful to split the calculation of $\mathcal{D}^u_T(P, S)$ into two separate steps: In a £rst step, we evaluate each modal subformula of T according to (P, S) and in a second step we then compute all models of the resulting propositional theory. To formalize this, we introduce the following notation: for each formula φ and $(P, S) \in \mathcal{B}_\Sigma$, the formula $\varphi\langle P, S\rangle$ is inductively de£ned as:

- $a\langle P, S\rangle = a$ for each atom a;
- $(\varphi_1 \wedge \varphi_2)\langle P, S\rangle = \varphi_1\langle P, S\rangle \wedge \varphi_2\langle P, S\rangle$;
- $(\varphi_1 \vee \varphi_2)\langle P, S\rangle = \varphi_1\langle P, S\rangle \vee \varphi_2\langle P, S\rangle$;
- $(\neg\varphi)\langle P, S\rangle = \neg(\varphi\langle S, P\rangle)$;
- $(K\varphi)\langle P, S\rangle = \mathcal{H}_{(P,S),\cdot}(K\varphi)$.

For a theory T, we denote $\{\varphi\langle P, S\rangle \mid \varphi \in T\}$ by $T\langle P, S\rangle$. Because clearly $\mathcal{H}_{(\cdot,\cdot),X}(T\langle P, S\rangle) = \mathbf{t}$ iff $\mathcal{H}_{(P,S),X}(T) = \mathbf{t}$, it is the case that for each $(P, S) \in \mathcal{B}_\Sigma$:

$$\mathcal{D}^u_T(P, S) = \{X \in \mathcal{I}_\Sigma \mid \mathcal{H}_{(\cdot,\cdot),X}(T\langle P, S\rangle)\}.$$

To illustrate, we will construct the Kripke-Kleene model of our example theory $F = \{p \vee \neg Kp;\ K(p \vee q) \vee q\}$. This computation starts at the least precise element $(\mathcal{I}_{\{p,q\}}, \{\})$ of $\mathcal{B}_{\{p,q\}}$. We £rst construct the new underestimate $\mathcal{D}^u_F(\{\}, \mathcal{I}_{\{p,q\}})$. It is easy to see that

$$\mathcal{H}_{(\{\},\mathcal{I}_{\{p,q\}}),\cdot}(\neg Kp) = \neg\mathcal{H}_{(\mathcal{I}_{\{p,q\}},\{\}),\cdot}(Kp) = \neg\mathbf{f} = \mathbf{t},$$

and

$$\mathcal{H}_{(\{\},\mathcal{I}_{\{p,q\}}),\cdot}(K(p \vee q)) = \mathbf{t}.$$

Therefore, $F\langle\{\}, \mathcal{I}_{\{p,q\}}\rangle = \{p \vee \mathbf{t};\ q \vee \mathbf{t}\}$ and $\mathcal{D}^u_F(\{\}, \mathcal{I}_{\{p,q\}}) = \mathcal{I}_{\{p,q\}}$. Now, to compute the new overestimate $\mathcal{D}^u_F(\mathcal{I}_{\{p,q\}}, \{\})$, we note that

$$\mathcal{H}_{(\mathcal{I}_{\{p,q\}},\{\}),\cdot}(\neg Kp) = \neg\mathcal{H}_{(\{\},\mathcal{I}_{\{p,q\}}),\cdot}(Kp) = \neg\mathbf{t} = \mathbf{f},$$

and

$$\mathcal{H}_{(\mathcal{I}_{\{p,q\}},\cdot),\cdot}(K(p \vee q)) = \mathbf{f}.$$

Therefore, $F\langle(\mathcal{I}_{\{p,q\}}, \{\}\rangle = \{p \vee \mathbf{f};\ q \vee \mathbf{f}\}$ and $\mathcal{D}^u_F(\mathcal{I}_{\{p,q\},\{\}} = \{\{p, q\}\}$. So, $\mathcal{D}_T(\mathcal{I}_{\{p,q\}}, \{\}) = (\mathcal{I}_{\{p,q\}}, \{\{p, q\}\})$.

To compute $\mathcal{D}^u_F(\{p, q\}, \mathcal{I}_{\{p,q\}})$, we note that $\mathcal{H}_{(\cdot,\mathcal{I}_{\{p,q\}}),\cdot}(\neg Kp)$ and $\mathcal{H}_{(\{\{p,q\}\},\cdot),\cdot}(K(p \vee q)$ are still $\mathbf{t}$. So, $\mathcal{D}^u_F(\{\{p, q\}\}, \mathcal{I}_{\{p,q\}}) = \mathcal{I}_{\{p,q\}}$. Similary, both $\mathcal{H}_{(\cdot,\{\{p,q\}\}),\cdot}(\neg Kp)$ and $\mathcal{H}_{(\mathcal{I}_{\{p,q\}},\cdot),\cdot}(\neg Kp)$ are still $\mathbf{f}$. So, $\mathcal{D}^u_F(\mathcal{I}_{\{p,q\}}, \{\{p, q\}\}) = \{\{p, q\}\}$. Therefore, $(\mathcal{I}_{\{p,q\}}, \{\{p, q\}\})$ is the least £xpoint of $\mathcal{D}_F$, i.e. the Kripke-Kleene model of F.

Strati£cation

Let $(\Sigma_i)_{i \in I}$ be a partition of the alphabet Σ, with $\langle I, \preceq\rangle$ a well-founded index set. For an interpretation $X \in \mathcal{I}_\Sigma$, we denote the intersection $X \cap \Sigma_i$ by $X|_{\Sigma_i}$. For a possible world structure Q, $\{X|_{\Sigma_i} \mid X \in Q\}$ is denoted by $Q|_{\Sigma_i}$.

In the previous section, we de£ned the semantics of auto-epistemic logic in terms of an operator on the bilattice $\mathcal{B}_\Sigma = \mathcal{W}^2_\Sigma$. However, for our purpose of stratifying auto-epistemic theories, we are interested in the bilattice $\tilde{\mathcal{B}}_\Sigma$ of the product lattice $\tilde{\mathcal{W}}_\Sigma = \bigotimes_{i\in I} \mathcal{W}_{\Sigma_i}$. An element of this product lattice consists of a number of possible interpretations for each level Σ_i. As such, if we choose for each Σ_i one of its interpretations, the union of these "chosen" interpretations interprets the entire alphabet Σ. Therefore, the set of all possible ways of choosing one interpretation for each Σ_i, determines a set of possible interpretations for Σ, i.e. an element of $\mathcal{W}_\Sigma$.

$$\kappa : \tilde{\mathcal{W}}_\Sigma \to \mathcal{W}_\Sigma : \tilde{Q} \mapsto \{\bigcup_{i\in I} S(i) \mid S \in \otimes_{i\in I}\tilde{Q}(i)\}.$$

Similary, $\tilde{\mathcal{B}}_\Sigma$ can be mapped to $\mathcal{B}_\Sigma$ by the function $\overline{\kappa}$, which maps each $(\tilde{P}, \tilde{S}) \in \tilde{\mathcal{B}}_\Sigma$ to $(\kappa(\tilde{P}), \kappa(\tilde{S}))$.

This function κ is, however, not an isomorphism. Indeed, unlike $\mathcal{W}_\Sigma$, elements of $\tilde{\mathcal{W}}_\Sigma$ cannot express that an interpretation for a level Σ_i is possible in combination with a *certain* interpretation for another level Σ_j, but *not* with a different interpretation for Σ_j. For instance, if we split the alphabet $\{p, q\}$ of our example F into $\Sigma_0 = \{p\}$ and $\Sigma_1 = \{q\}$, the element $\{\{p, q\}, \{\}\}$ of $\mathcal{W}_\Sigma$ is not in $\kappa(\tilde{\mathcal{W}}_\Sigma)$, because it expresses that $\{p\}$ is only a possible interpretation for Σ_0 when Σ_1 is interpreted by $\{q\}$ and not when Σ_1 is interpreted by $\{\}$. For this reason, elements of $\kappa(\tilde{\mathcal{W}}_\Sigma)$ and $\overline{\kappa}(\tilde{\mathcal{B}}_\Sigma)$ are called *disconnected*.

Because $\tilde{\mathcal{B}}_\Sigma$ and $\mathcal{B}_\Sigma$ are not isomorphic, we cannot directly stratify the operator $\mathcal{D}_T$. Instead, we need to follow the previously outlined methodology for dealing with "operators on other lattices". As a £rst step, we show that $\tilde{\mathcal{W}}_\Sigma$ is κ-similar to $\mathcal{W}_\Sigma$. Recall that a lattice $\tilde{L}$ is k-similar to a lattice L iff there exists a $k : \tilde{L} \to L$, such that k is chain-continuous and each non empty set $k^{-1}(x)$ has a central element.

Proposition 3. *$\tilde{\mathcal{W}}_\Sigma$ is κ-similar to $\mathcal{W}_\Sigma$ and $\tilde{\mathcal{B}}_\Sigma$ is $\overline{\kappa}$-similar to $\mathcal{B}_\Sigma$.*

Proof. As κ is clearly order preserving and $\tilde{\mathcal{W}}_\Sigma$ is £nite, κ is also chain continuous. Therefore, it suf£ces to show that for each $Q \in k(\mathcal{W}_\Sigma)$, $k^{-1}(Q)$ has a central element. If $Q \neq \{\}$, then $k^{-1}(Q)$ is a singleton. Furthermore, $\kappa^{-1}(\{\}) = \{\tilde{Q} \in \tilde{\mathcal{W}}_\Sigma \mid \exists i \in I : \tilde{Q}(i) = \{\}\}$. The element $\tilde{\top}_{\{\}}$ of $\tilde{\mathcal{W}}_\Sigma$ which maps each $i \in I$ to $\{\}$ is a $\leq_k$-largest and therefore central element of this set. □

In order to achieve our goal of being able to incrementally construct the models of a theory by means of the components of some operator on $\tilde{\mathcal{B}}_\Sigma$, we need to restrict our attention to a class of theories whose models are disconnected.

De£nition 5. An auto-epistemic theory T is *strati£able* w.r.t. a partition $(\Sigma_i)_{i\in I}$ of its alphabet, if there exists a partition $\{T_i\}_{i\in I}$ of T such that for each $i \in I$ and $\varphi \in T_i$: $At_O(\varphi) \subseteq \Sigma_i$ and $At(\varphi) \subseteq \bigcup_{j\preceq i} \Sigma_j$.

Clearly, for a strati£able theory, the evaluation $\mathcal{H}_{(P,S),X}(\varphi)$ of a formula $\varphi \in T_i$ only depends on the value of (P, S) in strata $j \preceq i$ and that of X in stratum i.

Proposition 4. *Let T be a strati£able theory. Let $i \in I$ and $\varphi \in T_i$. Then for each (P, S), $(P', S') \in \mathcal{B}_\Sigma$ and $X, X' \in \mathcal{I}_\Sigma$, such that $X|_{\Sigma_i} = X'|_{\Sigma_i}$ and $\forall j \preceq i$, $P|_{\Sigma_j} = P'|_{\Sigma_j}$ and $S|_{\Sigma_j} = S'|_{\Sigma_j}$, $\mathcal{H}_{(P,S),X}(\varphi) = \mathcal{H}_{(P',S'),X'}(\varphi)$.*

This proposition allows us to construct an operator on $\tilde{\mathcal{B}}_\Sigma$ which mimics the $\mathcal{D}_T$-operator of a strati£able theory T.

De£nition 6. Let T be a strati£able theory. Let $(\tilde{P}, \tilde{S})$ be in $\tilde{\mathcal{B}}_\Sigma$. We de£ne $\tilde{\mathcal{D}}^u_T(\tilde{P}, \tilde{S}) = \tilde{Q}$, with, for each $i \in I$, $\tilde{Q}(i) = \{X \in \mathcal{I}_{\Sigma_i} \mid \forall\varphi \in T_i : \mathcal{H}_{\overline{\kappa}(\tilde{P},\tilde{S}),X}(\varphi) = \mathbf{t}\}$. Furthermore, $\tilde{\mathcal{D}}_T(\tilde{P}, \tilde{S}) = (\tilde{\mathcal{D}}^u_T(\tilde{S}, \tilde{P}), \tilde{\mathcal{D}}^u_T(\tilde{P}, \tilde{S}))$ and $\tilde{D}_T(\tilde{Q}) = \tilde{\mathcal{D}}^u_T(\tilde{Q}, \tilde{Q})$.

Proposition 5. *Let T be a strati£able theory. Then $\tilde{\mathcal{D}}_T$ $\overline{\kappa}$-mimics $\mathcal{D}_T$, i.e. each $\mathcal{D}_T(P, S)$ is disconnected and $\overline{\kappa}\circ\tilde{\mathcal{D}}_T = \mathcal{D}_T \circ \overline{\kappa}$.*

Proof. Let $(P, S) \in \mathcal{B}_\Sigma$ and $X, Y \in \mathcal{D}^u_T(P, S)$. Let $Z \in \mathcal{I}_\Sigma$ be such that for some $i \in I$, $Z|_{\Sigma_i} = X|_{\Sigma_i}$ and, $\forall j \neq i$, $Z|_{\Sigma_j} = Y|_{\Sigma_j}$. Then, by proposition 4, for all formulae $\varphi \in T_i$, $\mathcal{H}_{(P,S),Z}(\varphi) = \mathcal{H}_{(P,S),X}(\varphi)$ and for all formulae $\varphi \in T_j$ with $j \neq i$, $\mathcal{H}_{(P,S),Z}(\varphi) = \mathcal{H}_{(P,S),Y}(\varphi)$. Therefore Z is also in $\mathcal{D}^u_T(P, S)$. This shows that $\mathcal{D}^u_T(P, S)$, and therefore also $\mathcal{D}_T(P, S)$, is disconnected. To show that $\overline{\kappa} \circ \tilde{\mathcal{D}}_T = \mathcal{D}_T \circ \overline{\kappa}$, it suf£ces to prove that for each $(\tilde{P}, \tilde{S}) \in \tilde{\mathcal{B}}_\Sigma$, $\mathcal{D}^u_T(\overline{\kappa}(\tilde{P}, \tilde{S})) = \kappa(\tilde{\mathcal{D}}^u_T(\tilde{P}, \tilde{S}))$. Let X be an element of $\mathcal{I}_\Sigma$. Then, by de£nition, $X \in \kappa(\tilde{\mathcal{D}}^u_T(\tilde{P}, \tilde{S}))$ is equivalent to $\forall i \in I, \forall\varphi \in T_i : \mathcal{H}_{\overline{\kappa}(\tilde{P},\tilde{S}),X|_{\Sigma_i}}(\varphi) = \mathbf{t}$, which, by proposition 4, is equivalent to $\forall\varphi \in T : \mathcal{H}_{\overline{\kappa}(\tilde{P},\tilde{S}),X}(\varphi) = \mathbf{t}$. □

Therefore, it suf£ces to show that $\tilde{\mathcal{D}}_T$ is monotone in order to prove a correspondence between £xpoints and least £xpoints of $\tilde{\mathcal{D}}_T$ and $\mathcal{D}_T$ (theorem 3).

Proposition 6. *Let T be a strati£able theory. Then $\tilde{\mathcal{D}}_T$ is an approximation.*

Proof. Let $(\tilde{P}, \tilde{S}), (\tilde{P}', \tilde{S}') \in \tilde{\mathcal{B}}_\Sigma$, such that $(\tilde{P}, \tilde{S}) \leq_p (\tilde{P}', \tilde{S}')$. Because then $(\tilde{S}, \tilde{P}) \geq (\tilde{S}', \tilde{P}')$, we only need to show that $\mathcal{D}^u_T(\tilde{P}, \tilde{S}) \geq_\otimes \mathcal{D}^u_T(\tilde{P}', \tilde{S}')$. As κ is order-preserving, $\overline{\kappa}(\tilde{P}, \tilde{S}) \leq_p \overline{\kappa}(\tilde{P}', \tilde{S}')$. From (Denecker, Marek, & Truszczynski 2003), we know this implies that for each φ of T and X in $\mathcal{I}_\Sigma$, if $\mathcal{H}_{\overline{\kappa}(\tilde{P},\tilde{S}),X} = \mathbf{t}$ then $\mathcal{H}_{\overline{\kappa}(\tilde{P}',\tilde{S}'),X} = \mathbf{t}$. Hence, $\tilde{\mathcal{D}}^u_T(\tilde{P}, \tilde{S})(i) \subseteq \mathcal{D}^u_T(\tilde{P}', \tilde{S}')(i)$. □

By theorem 3, these two propositions show that $\overline{\kappa}(fp(\tilde{\mathcal{D}}_T)) = fp(\mathcal{D}_T)$ and $\overline{\kappa}(lfp(\tilde{\mathcal{D}}_T)) = lfp(\mathcal{D}_T)$. Moreover, by proposition 2, the £xpoints and the least £xpoint of $\mathcal{C}_{\tilde{\mathcal{D}}_T}$, i.e. the stable £xpoints and well-founded £xpoint of $\tilde{\mathcal{D}}_T$, correspond to the £xpoints and least £xpoints of $\mathcal{C}_{\mathcal{D}_T}$, i.e. the stable £xpoints and well-founded £xpoint of $\mathcal{D}_T$. As mentioned at the end of section , $\tilde{D}_T$ also κ-mimics D_T. However, because $\tilde{D}_T$ is not monotone, it does not satisfy the conditions of theorem 3. As such, the best result which can be obtained for this operator, is that $\{Q \in fp(D_T) \mid Q \neq \{\}\} = \kappa(fp(\tilde{D}_T))$. This follows from κ being injective on the subset $\{\tilde{Q} \in \tilde{\mathcal{W}}_\Sigma \mid$

$\kappa(\tilde{Q}) \neq \{\}\}$ of its domain. To see that a stronger proposition does not hold, consider the theory $T = T_1 \cup T_2$, with $T_1 = \{Kp \to \neg p, \neg Kp \to p\}$ and $T_2 = \{q \wedge \neg q\}$. Clearly, D_T has $\{\}$ as a £xpoint, but $\tilde{D}_T$ has no £xpoints, as it oscillates between $\{\{p\}\} \sqcup \{\}$ and $\{\{\}\} \sqcup \{\}$.

As each operator $\tilde{\mathcal{D}}_T$ is strati£able by construction, these results allow us to incrementally construct the various models of a strati£able theory from the components of $\tilde{\mathcal{D}}_T$. These components themselves can in turn be constructed by replacing certain parts of T by their truth value according to a partial pair of interpretations $(\tilde{U}, \tilde{V}) \in \tilde{\mathcal{B}}_\Sigma|_{\prec i}$. Before showing this for all strati£able theories, we will £rst deal only with the following, more restricted class of theories.

De£nition 7. A theory T is *modally separated* w.r.t. to a partition $(\Sigma_i)_{i\in I}$ of its alphabet iff there exists a corresponding partition $(T_i)_{i\in I}$ of T, such that for each $i \in I$ and $\varphi \in T_i$

- $At_O(\varphi) \subseteq \Sigma_i$,
- for each modal subformula $K\psi$ of φ, either $At(\psi) \subseteq \Sigma_i$ or $At(\psi) \subseteq \bigcup_{j\prec i} \Sigma_j$.

Clearly, modally separated theories are by de£nition strati£able. The fact that each modal subformula of a level T_i of a modally separated theory T contains either only atoms from Σ_i or only atoms from a strictly lower level, makes it easy to construct the components of its $\tilde{\mathcal{D}}_T$-operator. Replacing all modal subformulae of a level T_i which contain only atoms from a strictly lower level $j \prec i$, by their truth-value according to a partial belief pair $(\tilde{U}, \tilde{V}) \in \mathcal{B}_\Sigma|_{\prec i}$ results in a "conservative theory" T^c, while replacing these subformulae by their truth-value according to $(\tilde{V}, \tilde{U})$ yields a "liberal theory" T^l. The pair $(\mathcal{D}^u_{T^l}, \mathcal{D}^u_{T^c})$ is then precisely the component $(\tilde{\mathcal{D}}_T)_i^{(\tilde{U},\tilde{V})}$ of $\tilde{\mathcal{D}}_T$.

To make this more precise, we inductively de£ne the following transformation $\varphi\langle U, V\rangle_i$ of a formula $\varphi \in T_i$, given a partial belief pair $(\tilde{U}, \tilde{V}) \in \tilde{\mathcal{B}}_\Sigma|_{\prec i}$:

- $a\langle\tilde{U}, \tilde{V}\rangle_i = a$ for each atom a;
- $(\varphi_1 \wedge \varphi_2)\langle\tilde{U}, \tilde{V}\rangle_i = \varphi_1\langle\tilde{U}, \tilde{V}\rangle_i \wedge \varphi_2\langle\tilde{U}, \tilde{V}\rangle_i$;
- $(\varphi_1 \vee \varphi_2)\langle\tilde{U}, \tilde{V}\rangle_i = \varphi_1\langle\tilde{U}, \tilde{V}\rangle_i \vee \varphi_2\langle\tilde{U}, \tilde{V}\rangle_i$;
- $(\neg\varphi)\langle\tilde{U}, \tilde{V}\rangle_i = \neg(\varphi\langle\tilde{V}, \tilde{U}\rangle_i)$;
- $(K\varphi)\langle\tilde{U}, \tilde{V}\rangle_i = \begin{cases} \mathcal{H}_{(\tilde{U},\tilde{V}),\cdot}(K\varphi) & \text{if } At(\varphi) \subseteq \bigcup_{j\prec i} \Sigma_j \\ K(\varphi) & \text{if } At(\varphi) \subseteq \Sigma_i. \end{cases}$

Note that this transformation $\varphi\langle\tilde{U}, \tilde{V}\rangle_i$ is identical to the transformation $\varphi\langle P, S\rangle$ de£ned earlier, except for the fact that in this case, we only replace modal subformulae with atoms from $\bigcup_{j\prec i}$ and leave modal subformulae with atoms from Σ_i untouched.

From the various de£nitions, it is now clear that the components $(\tilde{\mathcal{D}}_T)_i^{(\tilde{U},\tilde{V})}$ of the $\tilde{\mathcal{D}}_T$-operator of a modally separated theory T can be constructed as follows:

Proposition 7. *Let T be a modally separated theory. Let $i \in I$, $(\tilde{U}, \tilde{V}) \in \tilde{\mathcal{B}}_\Sigma|_{\prec i}$ and $(\tilde{P}_i, \tilde{S}_i) \in \tilde{\mathcal{B}}_{\Sigma_i}$. Then:*

$$(\tilde{\mathcal{D}}_T)_i^{(\tilde{U},\tilde{V})}(\tilde{P}_i, \tilde{S}_i) = (\mathcal{D}^u_{T_i\langle\tilde{V},\tilde{U}\rangle_i}(\tilde{S}_i, \tilde{P}_i), \mathcal{D}^u_{T_i\langle\tilde{U},\tilde{V}\rangle_i}(\tilde{P}_i, \tilde{S}_i)).$$

Now, all that remains is to characterize the components of strati£able theories which are not modally separated. It turns out that for each strati£able theory T, there exists a modally separated theory T', which is equivalent to φ w.r.t. evaluation in disconnected possible world structures. To simplify the proof of this statement, we recall that each formula φ can be written in an equivalent form φ' such that each modal subformula of φ' is of the form $K(a_1 \vee \cdots \vee a_m)$, with each a_i a literal. This result is well-known for S_5 semantics and can — using the same transformation — be shown to also hold for all semantics considered here[2].

Proposition 8. *Let (P, S) be a disconnected element of $\mathcal{B}_\Sigma$. Let $i \in I$, $b_1, \ldots, b_n$ literals with atoms from Σ_i and $c_1, \ldots, c_m$ literals with atoms from $\bigcup_{j\prec i} \Sigma_j$. Then*

$$\mathcal{H}_{(P,S),\cdot}(K(\bigvee_{j=1..n} b_j \vee \bigvee_{j=1..m} c_j)) = \mathcal{H}_{(P,S),\cdot}(K(\bigvee_{j=1..n} b_j) \vee K(\bigvee_{j=1..m} c_j)).$$

Proof. By de£nition,

$$\mathcal{H}_{(P,S),\cdot}(K(\bigvee_{j=1..n} b_j \vee \bigvee_{j=1..m} c_j)) = \mathbf{t}$$

iff

$$\forall X \in P : \mathcal{H}_{(\cdot,\cdot),X}(\bigvee_{j=1..n} b_j \vee \bigvee_{j=1..m} c_j) = \mathbf{t}.$$

This is equivalent to $\forall X \in P$, $\mathcal{H}_{(\cdot,\cdot),X}(\bigvee_{j=1..n} b_j) = \mathbf{t}$ or $\mathcal{H}_{(\cdot,\cdot),X}(\bigvee_{j=1..m} c_j) = \mathbf{t}$. Because P is disconnected, it contains all possible combinations $X|_{\cup_{j\prec i}\Sigma_j} \cup Y|_{\Sigma_i} \cup Z|_{\cup_{j\not\preceq i}\Sigma_j}$, with $X, Y, Z \in P$. Therefore the previous statement is in turn equivalent to for each $X, Y \in P$, $\mathcal{H}_{(\cdot,\cdot),X|_{\Sigma_i}}(\bigvee_{j=1..n} b_j) = \mathbf{t}$ or $\mathcal{H}_{(\cdot,\cdot),Y|_{\cup_{j\prec i}\Sigma_j}}(\bigvee_{j=1..m} c_j)$, which proves the result. □

The modally separated formula corresponding to a formula φ will be denoted by $[\varphi]$. In the case of our example $F = \{p \vee \neg Kp;\ K(p \vee q) \vee q\}$, the modally separated theory $[F] = \{p \vee \neg Kp;\ K(p) \vee K(q) \vee q\}$ is equivalent to F w.r.t. evaluation in $\overline{\kappa}(\tilde{\mathcal{W}}_{\{p,q\}})$. This results now allows us to characterize the components of all strati£able theories.

[2]To show this, it suf£ces to show that each step of this transformation preserves the value of the evalation $\mathcal{H}_{(P,S),X}(\varphi)$. For all steps corresponding to properties of (three-valued) propositional logic, this is trivial. The step of transforming a formula $K(K(\varphi))$ to $K(\varphi)$ also trivially satis£es this requirement. All that remains to be shown, therefore, is that $\mathcal{H}_{(P,\cdot),\cdot}(K(\varphi \wedge \psi)) = \mathcal{H}_{(P,\cdot),\cdot}(K(\varphi) \wedge K(\psi))$. By de£nition, $\mathcal{H}_{(P,\cdot),\cdot}(K(\varphi \wedge \psi)) = \mathbf{t}$ iff $\forall X \in P : \mathcal{H}_{(\cdot,\cdot),X}(\varphi) = \mathbf{t}$ and $\mathcal{H}_{(\cdot,\cdot),X}(\varphi) = \mathbf{t}$, which in turn is equivalent to $\forall X \in P : \mathcal{H}_{(\cdot,\cdot),X}(K(\varphi)) = \mathbf{t}$ and $\forall X \in P : \mathcal{H}_{(\cdot,\cdot),X}(K(\psi)) = \mathbf{t}$.

Theorem 4. *Let $i \in I$, $(\tilde{U},\tilde{V}) \in \tilde{\mathcal{B}}_\Sigma|_{\prec i}$ and $(\tilde{P}_i,\tilde{S}_i) \in \tilde{\mathcal{B}}_{\Sigma_i}$. Then:*

$$(\tilde{\mathcal{D}}_T)_i^{(\tilde{U},\tilde{V})}(\tilde{P}_i,\tilde{S}_i) = (\mathcal{D}^u_{[T_i]\langle(\tilde{V},\tilde{U})\rangle}(\tilde{S}_i,\tilde{P}_i), \mathcal{D}^u_{[T_i]\langle(\tilde{U},\tilde{V})\rangle}(\tilde{P}_i,\tilde{S}_i)).$$

Putting all of this together, we arrive at the following results: an element (P,S) of $\mathcal{B}_\Sigma$ is an expansion apart from $\{\}$, partial expansion, (partial) extension, Kripke-Kleene £xpoint or well-founded model of a strati£able theory T iff $\exists(\tilde{P},\tilde{S}) \in \tilde{\mathcal{B}}_\Sigma$, such that $\overline{\kappa}(\tilde{P},\tilde{S}) = (P,S)$ and for all $i \in I$, $(\tilde{P},\tilde{S})(i)$ is, respectively, an expansion, partial expansion, (partial) extension, Kripke-Kleene £xpoint or well-founded model of $[T_i]\langle(\tilde{P},\tilde{S})|_{\prec i}\rangle$. Using these results, we can therefore incrementally construct the models of a strati£able theory under each of these semantics.

To illustrate, we will use these results to incrementally compute the Kripke-Kleene model of our example F, which we previously partioned into $F_0 = \{p \vee \neg Kp\}$ and $F_1 = \{K(p \vee q) \vee q\}$. The Kripke-Kleene model of F_0 is $(\{\{\},\{p\}\},\{\{p\}\})$. Let F_1' be $[F_1]\langle(\{\{p\}\},\{\{\},\{p\}\})\rangle = \{\mathbf{t} \vee K(q) \vee q\}$ and let F_1'' be $[F_1]\langle(\{\{\},\{p\}\},\{\{p\}\})\rangle = \{\mathbf{f} \vee K(q) \vee q\}$. The least £xpoint of $(\mathcal{D}^u_{F_1'},\mathcal{D}^u_{F_1''})$ is $(\{\{\},\{q\}\},\{\{q\}\})$. Therefore, the Kripke-Kleene £xpoint of F is $(\mathcal{I}_{\{p,q\}},\{\{p,q\}\})$. Of course, (partial) expansions, (partial) extensions and the well-founded model of F can be computed in a similar manner.

Related work

In (Gelfond & Przymusinska 1992), it was shown that modally separated auto-epistemic theories can be split under the semantics of expansions. We have extended this result to a larger class of theories and to other semantics for auto-epistemic logic. (Turner 1996) proves a splitting theorem for default logic under Reiter's semantics of extensions (Reiter 1980). In [DMT03], approximation theory was used to show the equivalence (under the Konolige transformation (Konolige 1987)) between members of the family of semantics for default logic and the corresponding members of the family of semantics for autoepistemic logic. As such, the results derived here for auto-epistemic logic could also be applied to default logic. This would extend Turner's results to other semantics for default logic and to a larger class of theories, as (Turner 1996) considers only default theories which correspond to modally separated auto-epistemic theories.

Conclusion

We have studied the issue of strati£cation at the general, algebraic level of *operators* and *approximations* (section). This gave us a useful set of theorems, which enabled us to easily prove splitting results for all £xpoint semantics of auto-epistemic logic (section), thus generalizing existing results. Similar results can be obtained for logic programming (Vennekens, Gilis, & Denecker 2004) and, in future work, for default logic. As such, the importance of the work presented here is threefold. Firstly, there are the concrete, applied splitting results themselves. Secondly, there is the general, algebraic framework for studying strati£cation, which can be applied to each formalism with a £xpoint semantics. Finally, on a more abstract level, our work offers greater insight into the principles underlying existing results, as we are able to "look beyond" purely syntactical properties of a certain formalism.

References

Apt, K.; Blair, H.; and Walker, A. 1988. Towards a theory of Declarative Knowledge. In Minker, J., ed., *Foundations of Deductive Databases and Logic Programming*. Morgan Kaufmann.

Ch, J., and van der Hoek, M. 1995. *Epistemic Logic for Computer Science and Arti£cial Intelligence*. Cambridge University Press.

Denecker, M.; Marek, V.; and Truszczynski, M. 1998. Fixpoint 3-valued semantics for autoepistemic logic. In *Proceedings of the Fifteenth National Conference on Arti£cial Intelligence*, 840–845. MIT Press / AAAI-Press.

Denecker, M.; Marek, V.; and Truszczynski, M. 2000. Approximating operators, stable operators, well-founded £xpoints and applications in non-monotonic reasoning. In *Logic-based Arti£cial Intelligence*, The Kluwer International Series in Engineering and Computer Science. Kluwer Academic Publishers, Boston. 127–144.

Denecker, M.; Marek, V.; and Truszczynski, M. 2003. Uniform semantic treatment of default and autoepistemic logics. *Arti£cial Intelligence* 143(1):79–122.

Erdoğan, S., and Lifschitz, V. 2004. De£nitions in Answer Set Programming. In *Proc. Logic Programming and Non Monotonic Reasoning, LPNMR'04*, volume 2923 of *LNAI*, 185–197. Springer-Verlag.

Gelfond, M., and Przymusinska, H. 1992. On consistency and completeness of autoepistemic theories. *Fundamenta Informaticae* 16(1):59–92.

Gelfond, M. 1987. On Strati£ed Autoepistemic Theories. In *Proc. of AAAI87*, 207–211. Morgan Kaufman.

Konolige, K. 1987. On the relation between default and autoepistemic logic. In Ginsberg, M. L., ed., *Readings in Nonmonotonic Reasoning*. Los Altos, CA: Kaufmann. 195–226.

Lifschitz, V., and Turner, H. 1994. Splitting a logic program. In *International Conference on Logic Programming*, 23–37.

Moore, R. C. 1984. Possible-world semantics for autoepistemic logic. In *Proc. of the Non-Monotonic Reasoning Workshop*, 344–354.

Reiter, R. 1980. A logic for default reasoning. *Arti£cial Intelligence* 13(1–2):81–132.

Turner, H. 1996. Splitting a default theory. In *Proc. Thirteenth National Conference on Arti£cial Intelligence and the Eighth Innovative Applications of Arti£cial Intelligence Conference*, 645–651. AAAI Press.

Vennekens, J.; Gilis, D.; and Denecker, M. 2004. Splitting an operator: an algebriac modularity result and its application to logic programming. Accepted for ICLP'04.