Non-Monotonic Systems: A Framework for Analyzing Semantics and Structural Properties of Non-Monotonic Reasoning

Tomi Janhunen*
Helsinki University of Technology,
Digital Systems Laboratory

March, 1998

Doctoral Thesis

Dissertation for the degree of Doctor of Technology to be presented with due permission for public examination and debate in Auditorium E at Helsinki University of Technology (Espoo, Finland) on the 14th of March, 1998, at 12 o'clock noon.

*Contact information of the author: Tomi.Janhunen@hut.fi (email), P.O. Box 1100, FIN-02015 HUT, FINLAND (mail), +358-9-451-3255 (telephone), +358-9-451-3369 (fax).

HELSINKI UNIVERSITY OF TECHNOLOGY

ABSTRACT OF THE DOCTORAL THESIS

Author: Tomi Janhunen

Title of the Thesis: Non-Monotonic Systems: A Framework for Analyzing Semantics and Structural Properties of Non-Monotonic Reasoning

Translation: Epämonotoniset järjestelmät: lähestymistapa epämonotonisen päättelyn semantiikkojen ja rakenteisten ominaisuuksien analysointiin

Date: March, 1998

Number of pages: vi + 211

Department: Computer Science and Engineering

Professorship: Tik-79

Supervisor: Professor Leo Ojala

Instructor: Docent Ilkka Niemelä

Non-monotonic systems are introduced as a framework for analyzing non-monotonic reasoning. Such systems are defined as parametrized inference operators in a way that is compatible with Tarski's characterization of classical reasoning. The generality of the framework is demonstrated by presenting non-monotonic systems for the leading non-monotonic logics such as circumscription, default logic and autoepistemic logic. Various translations between non-monotonic logics are studied within the framework. Non-monotonic systems associated with the non-monotonic logics under consideration are shown to be preserved by these translations – including new translations between default logic and bimodal autoepistemic logic presented in the work. The standard semantics of non-monotonic reasoning – stable and stationary semantics – are generalized for all non-monotonic systems. Various properties of the semantics such as cumulativity and computational complexity are addressed. Moreover, it is established that the translation functions presented preserve these semantics. Finally, a new semantics called cautious semantics is obtained as a systematic refinement of stationary semantics. Cautious semantics is analyzed using the methodology developed in the work. A comparison is made with similar approaches in the literature. Applications of cautious semantics are considered in the fields of logic programming and consistency-based diagnosis.

Keywords: Non-monotonic reasoning, semantics, cumulativity, computational complexity

Preface

I have prepared this thesis while holding a research position in Digital Systems Laboratory at Helsinki University of Technology from January 1995 to December 1997. I am very grateful to the head of the laboratory, professor Leo Ojala, for the opportunity of preparing this thesis under his supervision.

I am greatly indebted to the instructor of this thesis, docent Ilkka Niemelä, for offering me challenging research problems in the field of non-monotonic reasoning. His advice and insight into the research field have been indispensable. I thank all my colleagues in Digital Systems Laboratory for the inspiring working atmosphere in the laboratory. I also thank Elizabeth Heap-Talvela for her help in checking the English language of the thesis.

The research reported in the thesis has been funded since January 1995 by Academy of Finland through Helsinki Graduate School in Computer Science and Engineering. The financial support from Alfred Kordelin Foundation, Emil Aaltonen Foundation, Foundation of Technology and Jenny and Antti Wihuri Foundation is also acknowledged with gratitude.

This thesis is dedicated to my family and friends for their continuous support and patience during these years.

Otaniemi, March 1998

Tomi Janhunen

Contents

List of Tables

List of Figures

1 Introduction

A fundamental property of classical reasoning is *monotonicity*: the more premises are given, the more conclusions can be drawn. In contrast to this, *non-monotonic reasoning* (NMR) lacks this property and earlier conclusions may be withdrawn on the basis of additional premises. This kind of non-monotonicity phenomenon appears frequently in problem domains where inferences are based on *incomplete information* or *default conclusions* and thus conclusions are naturally defeated. There is a variety of applications in *artificial intelligence* (AI) and related fields which are subject to non-monotonicity: logic programming [61], truth maintenance [22], inheritance [111], diagnosis [95] and reasoning about actions [35, 108]. To model these kinds of systems, a number of *non-monotonic logics* have been proposed as formalizations of non-monotonic reasoning. Among the leading formalizations are circumscription [68], default logic [94] and autoepistemic logic [72] which characterize the sets of conclusions associated with a set of premises under a non-monotonic inference mechanism in a declarative, implementation-independent way.

Since the introduction of non-monotonic logics, their mutual relationship has been intensively investigated. Basically, two approaches have been used. Firstly, numerous *translation functions* [26, 39, 46, 47, 52, 53, 55, 66, 78, 112, 120] have been proposed to transform theories of one logic into theories of another. An essential requirement for such a transformation is that the sets of conclusions associated with a theory and its translation match. Once established, these kinds of relationships form the basis of ranking logics in generality, i.e. their knowledge representation capability. For instance, it has been established that default reasoning is a special case of autoepistemic reasoning [52, 55, 66, 78, 112]. The second approach for studying the interconnections of non-monotonic logics stems from the obvious question about what properties these logics have in common (in addition to non-monotonicity). In other words, a meta-level definition or characterization of non-monotonic reasoning is searched for. This matter has already attracted many researchers [19, 31, 56, 63, 110] and abstract properties of non-monotonic inference have been introduced. Subsequently, the above mentioned non-monotonic logics have been analyzed with respect to these properties.

The study of non-monotonic logics has concentrated much on their *semantics*. Traditionally, one defines a logic in two phases by first specifying the syntax (the structure of the *sentences*) and then assigning the semantics (i.e. meaning) for the syntactic constructs. However, the semantics of non-monotonic logics has turned out to be a non-trivial matter and several semantic definitions have been proposed. For instance, default theories have been considered under Reiter's original semantics [94] as well as numerous variants [11, 89, 96, 110] of that. The field of logic programming has contributed a variety of semantics (see, e.g., [3, 19, 32, 44, 115, 118]) which have been influenced by the semantics introduced for non-monotonic logics, and vice versa. For example, the *stable semantics* [32] of normal logic programs is a derivative of Moore's semantics for autoepistemic logic [72] whereas the *stationary semantics* [89] of default logic was obtained from the *well-founded semantics* [115] of normal logic programs. The relationship of the various semantics of non-monotonic logics as well as their abstract properties have received much attention in the literature. In particular, the *cumulativity* property [11, 19, 63, 96, 99, 110] and *computational complexity* [25, 37, 38, 77, 106] have been intensively analyzed. The former has been considered as a fundamental property that non-monotonic reasoning should obey while the latter issue is essential when efficient implementations of non-monotonic reasoning are gradually developed [84, 85].

Although the formalizations of non-monotonic reasoning proposed in the literature vary much in syntax, there are considerable resemblances in their semantic definitions. For instance, very similar fixed point conditions are involved in Reiter's default logic [94] and in autoepistemic logic proposed by Moore [72]. Further similarities have been noticed when the properties of non-monotonic logics have been analyzed. For instance, the above mentioned complexity results obtained for circumscription, default logic and autoepistemic logic indicate that the major decision problems of these logics are all complete problems on the second level of the polynomial time hierarchy [2]. Moreover, the properties of semantics seem to transfer between the various formalizations. For example, the well-founded semantics of normal logic programs and the stationary semantics of default logic are both cumulative. These observations on the formalizations of non-monotonic reasoning give rise to the following topics and research problems to be addressed in this work:

- *Common structure in non-monotonic logics*

  Non-monotonic logics have all been proposed as formalizations of a particular kind of reasoning, where classical monotonic reasoning is supported with inference mechanisms to deal with incomplete information. This suggests that non-monotonic reasoning could be seen as a form of *parametrized* monotonic reasoning. If this view is adopted, then the problem is identifying the parameters involved and their effect on reasoning. To address this problem in this work, we introduce a notion of a *non-monotonic system* which is defined as a parameterized inference operator. The aim in this definition is to extend Tarski's characterization [109] of monotonic inference operators to the parametrized case and thus establish the minimal requirements for inference operators associated with non-monotonic logics. The generality of the notion is demonstrated by extracting non-monotonic systems for circumscription, default logic and autoepistemic logic.

- *Abstract semantics*

  A further question is whether the semantic definitions of non-monotonic logics can be lifted to the level of inference operators. For instance, principles similar to that of the stable semantics of normal logic programs [32] have appeared in various contexts such as circumscription [68], default logic [94] and autoepistemic logic [72, 66]. Also the well-founded semantics of normal logic programs [115] led to the introduction of a stationary semantics for default logic [89]. This suggests that we could abstract on the semantic principles behind the stable semantics and the well-founded semantics of normal logic programs and formulate the respective notions of *stable semantics* and *stationary semantics* independently of a particular formalization of non-monotonic reasoning. Such an abstraction can be useful when a particular semantics is generalized for further forms of non-monotonic reasoning. Another issue is whether the applicability of a particular semantics can be addressed directly from its general formulation. As a response to these needs, it is investigated how various semantic notions associated with non-monotonic logics can be lifted to the level of non-monotonic systems.

- *Properties shared by the semantics*

  Once the abstract formulations of stable and stationary semantics (or any other semantics) are available, it is natural to ask what properties of the semantics can be established on this level of abstraction. The question is approached in this work by taking properties such as *strength*, *cumulativity*, *preservation under translations* and *computational complexity* into consideration. Let us briefly discuss these properties. (i) Traditionally, the semantics have been ordered in strength on the basis of the sets of conclusions they provide. For instance, it has been shown that stable models of a normal logic program extend its well-founded model [115]. It seems that such comparisons can also be made on a higher level of abstraction. (ii) As mentioned above, the cumulativity property of well-founded semantics is preserved when the semantics is generalized for default logic [89]. This suggests that cumulativity and other structural properties of the semantics are also independent of the particular formalization and thus analyzable on an abstract level. (iii) It is interesting to know whether the semantics under consideration are preserved under the translations proposed in the literature. In this respect, the interconnection of Reiter's default logic [94] and Marek and Truszczyński's strong autoepistemic logic [66] is not straightforward. For instance, the translation function proposed by Konolige [52, 53] fails to capture the stable semantics of default logic on the autoepistemic side. Moreover, Gottlob [39] argues essentially that *modular* translations in the other direction are impossible. (iv) Recent results suggest that a particular semantics has sources of computational complexity that are *inherent* in the semantics. For instance, the computational complexity of default logic is different under stable and stationary semantics [37, 38]. Therefore, it is worth studying these issues by starting from the abstract formulations.

- *Alternative semantics*

  As established in the field of logic programming, stationary semantics is considerably weaker semantics than stable semantics. On the other hand, the existence of stable models is not guaranteed while a normal logic program has always a well-founded model (that determines the stationary semantics of the program). This setting has attracted attention and a number of semantics that reside in the gap between stationary semantics and stable semantics have been proposed. Among these approaches are [3, 19, 44, 118] which are restricted to normal logic programs. However, these semantics are not satisfactory in all respects (see, e.g., [19] for an extensive study): some of them are still too weak and some of them lead to unwanted conclusions in particular benchmark problems. This work addresses the possibilities to define reasonable semantics in the above mentioned gap on a higher level of abstraction, i.e using non-monotonic systems. Expected advantages of such an abstract approach are that (i) the properties of semantics are inherited for all formalizations of non-monotonic reasoning that meet the criteria of non-monotonic systems and (ii) semantics become independent of syntax.

1.1 Structure of the Work

The contents of this work are organized in eleven sections that are briefly previewed in the following. For the reader's convenience, the rough dependencies of the sections have been depicted in Figure 1. In **Section 2**, we look back to the theory of inference operators to set up the notion of non-monotonic systems as parametrized inference operators. The major non-monotonic logics are then shown to conform to the specification by identifying the inference operators associated with these logics. Moreover, the relationship of the default logic and autoepistemic logic is addressed in detail. A novel variant of autoepistemic logic is proposed to make this relationship straightforward and tight. **Section 3** analyzes the interconnections of non-monotonic systems. Three basic relations among non-monotonic systems are introduced: subsystems, functional equivalences and embeddings. Using these notions, it is then shown that a variety of translation functions preserve the non-monotonic systems associated with the theories under translation. In particular, these results are extended for a completely new translation function which is introduced to map autoepistemic theories into default ones. In **Section 4**, two leading semantics – namely stable semantics and stationary semantics – are presented. These semantics are directly built upon notions of consistency and completeness introduced for non-monotonic systems. This extends the two semantics to

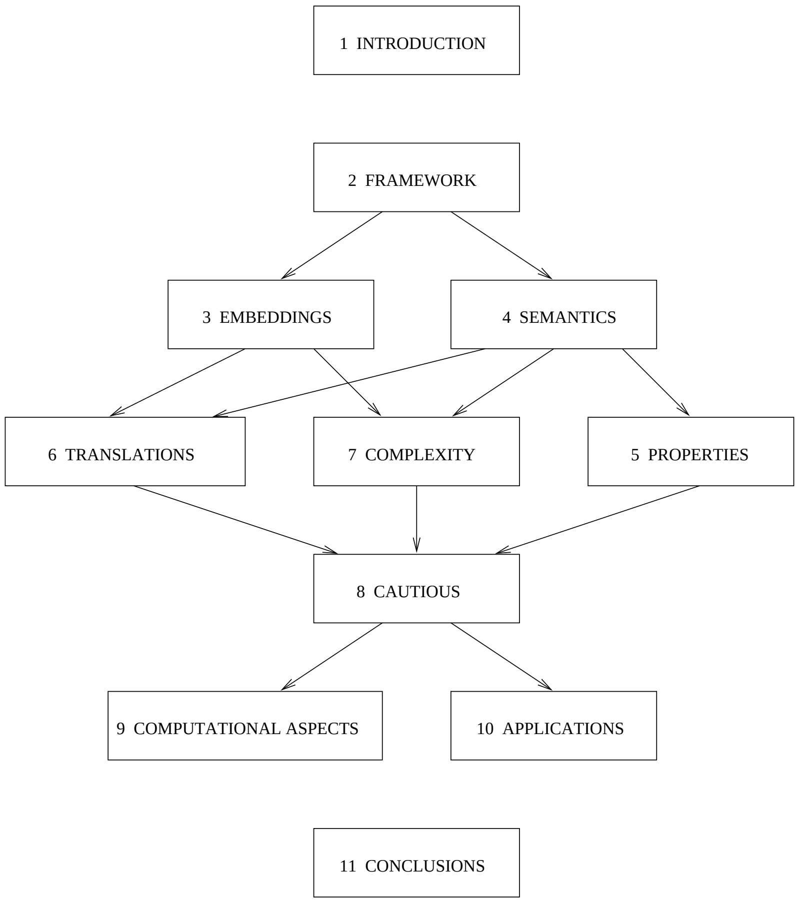


Figure 1: Dependencies of the Sections

a much broader context, including the bimodal variant of autoepistemic logic proposed in this work. The basic properties of the two semantics as well as their mutual relationship is addressed. **Section 5** concentrates on the abstract properties of the semantics. The most important of these is the cumulativity property, but other properties are also addressed. In particular, the properties analyzed in conjunction with stable and stationary semantics in order to demonstrate that such analysis can be performed on this level of abstraction. In **Section 6**, we further analyze the three basic relations (introduced in Section 3) in conjunction with the semantics considered so far. The translation functions introduced in Section 3 are shown to preserve both stable semantics and stationary semantics. In **Section 7**, the methods of complexity theory are applied to analyzing the time complexity of the decision procedures for stable semantics and stationary semantics and to classifying the corresponding decision problems in the polynomial time hierarchy. In **Section 8**, the machinery introduced in the preceding sections is used to analyze a novel semantics – cautious semantics – which is a derived from the stationary semantics by a systematic refinement technique. Cautious semantics is shown to be a semantics in the gap between stationary and stable semantics. The abstract properties of the semantics are then analyzed, e.g. the cumulativity property is addressed. **Section 9** concentrates on the computational aspects of cautious semantics. Algorithms for the refinement technique are presented and decision procedures are devised for cautious semantics. The complexity of the major decision problems is analyzed. A range of gradually refining approximations of cautious semantics is also considered. The weakest and the strongest approximations in the range are identified. In **Section 10**, some applications of cautious semantics are addressed. First of all, we analyze different classes of logic programs under cautious semantics determined by an autoepistemic interpretation of the program rules. As the second application, the fault diagnosis of digital circuits is considered. It is demonstrated how such reasoning is performed under cautious semantics. The conclusions of the work are presented in **Section 11**. Some preliminary notions and definitions related to propositional logic, ordinal and cardinal numbers as well as transfinite induction are collected in **Appendix A**. Also, an index of the major concepts is included to help the reader.

1.2 Scientific Contributions

This work is a synthesis and extension of the ideas published by the author elsewhere [43, 44, 45, 46, 47, 48]. In the following, the scientific contributions of this work are briefly assessed section by section (excluding the introduction and the conclusions of the work).

2. Thiele [110] analyzes the abstract properties of a default reasoning system. This work extends Thiele's work by giving the first class status for these properties as a specification of a non-monotonic system and showing that the inference operators associated with circumscription and autoepistemic logic fit this specification in a natural way. A new variant of autoepistemic logic is devised by introducing separate modalities for beliefs and disbeliefs. The resulting inference operator is also shown to be a non-monotonic system.

3. Three kinds of relations among non-monotonic systems are proposed in this work: subsystems, functional equivalences and embeddings. The usefulness of these notions is demonstrated, e.g., by showing how translation functions proposed for non-monotonic logics give rise to natural embeddings. In particular, a novel translation of an autoepistemic theory into a default theory is proposed and the problems associated with the converse translation proposed by Konolige [52] are shown to disappear with the separation of beliefs and disbeliefs. All of the translation functions presented are shown to preserve the associated non-monotonic systems.

4. The relationship of well-founded semantics and stable semantics are well-understood in the context of normal logic programs [32, 115]. Neverthe-

less, this work demonstrates how these semantics are concisely definable by starting from the specification of non-monotonic systems. This makes these semantics applicable in much broader context. A particular novelty of the work is that the separate modalities for belief and disbelief allow a reasonable notion of stationary semantics for autoepistemic theories – in contrast to theories in Moore's autoepistemic logic [72] as well as Marek and Truszczyński's strong autoepistemic logic [66] for which stationary semantics collapses to the modal logic of necessitation [29].

5. Further properties of stable semantics and stationary semantics are lifted to the level of non-monotonic systems. In particular, stationary semantics is shown to be cumulative for all non-monotonic systems satisfying the specification. Also, a sufficient condition is provided for the cumulativity of stable semantics. Circumscription and syntactically restricted default logic are shown to share this property.

6. The conditions on which embeddings preserve stable semantics and stationary semantics are analyzed. The main result is that the translation functions considered in this work preserve stable and stationary semantics. In particular, this holds for the new translation function Tr_{J} proposed by the author [46, 47] – in contrast to the translation functions proposed by Konolige [52] and Truszczyński [113] for which counter-examples are obtained already on the level of non-monotonic systems (as presented in Section 3). The translation technique behind Tr_{J} sheds also new light on the relationship of default logic and standard autoepistemic logic under stable expansions, since it leads to a modular translation which captures stable expansions of conventional autoepistemic theories in terms of default extensions [46]. This result differs in an interesting way from Gottlob's result [39]. He shows that a modular translation is not possible if the goal is to capture the propositional parts of stable expansions in terms of default extensions. Our result becomes possible, because we capture entire stable expansions.

7. The work demonstrates that complexity results as well as algorithms can also be abstracted for the non-monotonic logics under consideration. New complexity results are obtained for the bimodal variant of autoepistemic logic proposed in this work. First of all, the time complexity of the decision problem related to stable semantics does not change by the introduction of the separate modalities for beliefs and disbeliefs, i.e. it remains $\mathbf{\Pi}_2^{\mathrm{p}}$-complete. Moreover, the decision problem of stationary semantics is $\mathbf{\Delta}_2^{\mathrm{p}}$-complete and thus less complex than that of stable semantics.

8. A completely new semantics is also proposed for non-monotonic systems. This semantics is closely related to the semantics of normal logic programs proposed by the author elsewhere [44]. The results concerning cautious semantics are new. Most importantly, it is shown that (i) cautious semantics

is a semantics in the gap between stationary semantics and stable semantics and (ii) cautious semantics is shown to be cumulatively monotonic but not cumulatively closed – in contrast to stable semantics.

9. The main result is that the decision problem of cautious semantics has the same time complexity as the decision problem of stable semantics, i.e. it is $\mathbf{\Pi}_2^{\mathrm{p}}$-complete. New algorithms for reasoning under cautious semantics are presented. Because these algorithms are formulated at the level of non-monotonic systems, they can be applied directly in conjunction with circumscription, default logic and autoepistemic logic.

10. It is shown that cautious semantics leads to a new semantics for normal logic programs under the Gelfond-Lifschitz-style interpretation of program rules as autoepistemic sentences. This semantics is shown to be invariant under various program transformations considered, e.g., by Dix [20]. The disjunctive case is also considered: a special novelty in this case is that the resulting notion of the stationary semantics of disjunctive logic programs has a lower time complexity than stable semantics. As a second application, the diagnosis problem of digital circuits is considered. It is demonstrated that (i) cautious semantics is strong enough to solve this problem in contrast to stationary semantics and (ii) cautious semantics provides a new structure for the problem compared to stable semantics.

2 A Framework of Non-Monotonic Systems

The overall goal of this section is identifying the fundamental properties of *non-monotonic reasoning* using the theory of inference operators that goes back to Tarski [109]. These properties are then considered as an abstract specification for systems performing/modeling non-monotonic reasoning. Such systems will be called *non-monotonic systems* in the sequel.

The framework of non-monotonic systems is established in two phases. Firstly, we introduce *monotonic systems* as a characterization of classical monotonic reasoning in Section 2.1. The characterization is based on *inference operators* that are effectively *closure operators* on sets of sentences in some logical language. Then we extend the characterization of monotonic reasoning to cover non-monotonic reasoning in Section 2.2. This leads to a notion of a *non-monotonic system* and a specification of non-monotonic reasoning in terms of inference operators. The presented specification provides the basis for the analysis of non-monotonic reasoning systems in sections to come.

The rationality of the specification is established in Section 2.3 where various logical approaches to non-monotonic reasoning such as circumscription [68], default logic [94] and autoepistemic logic [72, 73] including the variant proposed in this work are shown to conform to the specification. Inference operators associated these formalizations of non-monotonic reasoning are identified. This supports the view that the theory of non-monotonic systems serves as a meta-theory for these logics. On the other hand, a more practical view is also justifiable. The above mentioned formalizations of non-monotonic reasoning cover a wide range of applications of non-monotonic reasoning such as systems for *logic programming* [61], *truth maintenance* [22], *inheritance* [111], *diagnosis* [95] and *reasoning about action* [35, 108]. In this way, various applications of non-monotonic reasoning come into the scope of our framework, but on a relatively high level of abstraction.

2.1 Monotonic Systems

To provide an immediate link to practical reasoning systems, we start with an example of monotonic reasoning. We will consider a simple rule-based system which manipulates *atoms* denoting certain properties of the components of a digital circuit. Given a component `x`, we represent its inputs and outputs using atoms like `In1(x,y)`, `In2(x,y)` and `Out(x,y)` where `y` is one of the wire states `0` and `1`. The behaviour of the components is represented in terms of *rules* which are then used for reasoning about the component state given input values. In Example 2.1 below, we have listed three rules that allow a reasoning system to infer the output values of an and-gate given its input values. The variables in the rules are universally quantified over and-gates, so that we consider the *substitution instances* of the rules below for each and-gate `G` in the circuit (obtained by substituting systematically `G` for `x` in the rules).

Example 2.1 Consider a system based on the following rules. By the first two rules the output of an and-gate is `0` if either one of the inputs is `0`. The last rule provides the condition by which an output of an and-gate is `1`.

If $\mathtt{In1}(\mathtt{x},0)$, then $\mathtt{Out}(\mathtt{x},0)$.
If $\mathtt{In2}(\mathtt{x},0)$, then $\mathtt{Out}(\mathtt{x},0)$.
If $\mathtt{In1}(\mathtt{x},1)$ and $\mathtt{In2}(\mathtt{x},1)$, then $\mathtt{Out}(\mathtt{x},1)$.

The system reasons about an and-gate `G` as follows. Given an atom $\mathtt{In2}(\mathtt{G},0)$ as an assumption, the system applies the second rule and concludes that $\mathtt{Out}(\mathtt{G},0)$. The other rules are not applicable. Hence the system provides us the conclusions $\mathtt{In2}(\mathtt{G},0)$ and $\mathtt{Out}(\mathtt{G},0)$[1]. Reasoning with these kinds of rules is inherently *monotonic* in the following sense: the more atoms are given as assumptions for the system, the more atoms can be inferred using the rules. In our example, the system infers also $\mathtt{Out}(\mathtt{G},1)$ if the system is provided with additional atoms $\mathtt{In1}(\mathtt{G},1)$ and $\mathtt{In2}(\mathtt{G},1)$. As a result of adding these assumptions, our set of conclusions is extended by three more atoms. For this reason, we consider the system of our example to be a *monotonic system.* □

Motivated by our example, we proceed to a formal definition of a *reasoning system* $\mathcal{S}$. The logical language associated with $\mathcal{S}$ is denoted by $\mathcal{L}(\mathcal{S})$. For the moment, it is sufficient to consider $\mathcal{L}(\mathcal{S})$ as an arbitrary, but *denumerable* set of sentences[2]. We take the input/output behavior demonstrated in Example 2.1 as a key feature of a system $\mathcal{S}$. Therefore we formalize $\mathcal{S}$ as an *inference operator* in the sense of Tarski [109]. Such operators are simply functions $\mathcal{S} : \mathbf{2}^{\mathcal{L}(\mathcal{S})} \rightarrow \mathbf{2}^{\mathcal{L}(\mathcal{S})}$ on the language $\mathcal{L}(\mathcal{S})$ associated with the system $\mathcal{S}$[3]. Under this abstract interpretation, the reasoning system $\mathcal{S}$ maps a set of sentences $T \subseteq \mathcal{L}(\mathcal{S})$ given as *premises* or *assumptions* to a set of *conclusions* $\mathcal{S}(T) \subseteq \mathcal{L}(\mathcal{S})$. A sentence $\phi \in \mathcal{L}(\mathcal{S})$ is called a *theorem* if it is one of the conclusions provided by the system $\mathcal{S}$. Thus $\mathcal{S}(T)$ is the set of theorems provided by the system $\mathcal{S}$. A *theory* is defined as a set of theorems.[4]

Functionality of the above kind is all we require from a reasoning system $\mathcal{S}$ in general, but a further restriction is necessary to make $\mathcal{S}$ *monotonic.* Monotonic reasoning has its theoretical foundations in classical monotonic logics. In fact,

[1]Here we are interested in the total set of conclusions provided by the system.
[2]It seems that restriction to denumerable languages is unnecessary, but all systems considered in this work have this property.
[3]The symbol $\mathcal{S}$ denotes the system as well as the associated inference operator.
[4]See also the discussion in conjunction with the reflexivity axiom REF below.

Axiom	For every $T' \subseteq T \subseteq \mathcal{L}(\mathcal{S})$:
(REF)	$T \subseteq \mathcal{S}(T)$,
(CLO)	$\mathcal{S}(\mathcal{S}(T)) = \mathcal{S}(T)$,
(MON)	$\mathcal{S}(T') \subseteq \mathcal{S}(T)$ and
(COM)	$\mathcal{S}(T) = \bigcup\{\mathcal{S}(T') \mid T' \subseteq_{\mathrm{f}} T\}$.

Table 1: Specification of a monotonic system

we could write down sentences of propositional logic corresponding to the system of Example 2.1 (if substitution instances of the rules are considered). Consequently, the set of conclusions associated with $\mathcal{S}$ would be naturally defined as a *logical closure* of the corresponding sentences. This is why we restrict our attention to systems $\mathcal{S}$ satisfying the standard axioms of *closure operators* [15]. In Table 1 we have listed axioms REF, CLO, MON and COM that specify basic properties for an inference operator $\mathcal{S}$. A closure operator is an operator having the first three properties of these:

Definition 2.2 *An inference operator $\mathcal{S} : \mathbf{2}^{\mathcal{L}(\mathcal{S})} \to \mathbf{2}^{\mathcal{L}(\mathcal{S})}$ is a closure operator if and only if $\mathcal{S}$ satisfies the axioms* REF, CLO *and* MON *in Table 1.*

Reasoning with inference operators satisfying the axioms in Table 1 is governed by the following principles. The *reflexivity* axiom REF ensures that $\mathcal{S}$ includes the sentences of T given as assumptions in the set of theorems $\mathcal{S}(T)$. In the presence of the reflexivity axiom, we say also that the system $\mathcal{S}$ maps a set of *theorems* T to a set of *theorems* $\mathcal{S}(T)$. This is because $T \subseteq \mathcal{S}(T)$ is guaranteed. For the same reason, we can regard any set of sentences $T \subseteq \mathcal{L}(\mathcal{S})$ as a *theory*. By the condition CLO, taking $\mathcal{S}(T)$ as assumptions (instead of T) yields the same set of theorems. In other words, the set $\mathcal{S}(T)$ is *closed* with respect to the operator $\mathcal{S}$, i.e. $\mathcal{S}(T)$ remains unchanged by further applications of the operator $\mathcal{S}$. The requirement MON enforces the *monotonicity* property already discussed with Example 2.1: the set of theorems $\mathcal{S}(T)$ grows monotonically whenever T does. The last requirement COM demands that $\mathcal{S}$ is *compact*, i.e. $\mathcal{S}(T)$ is determined as the union of the sets of theorems $\mathcal{S}(T')$ where T' is a finite subset of T[5]. Note that the other half $\bigcup\{\mathcal{S}(T') \mid T' \subseteq_{\mathrm{f}} T\} \subseteq \mathcal{S}(T)$ of compactness follows already by the requirement MON, since $T' \subseteq_{\mathrm{f}} T$ implies $\mathcal{S}(T') \subseteq \mathcal{S}(T)$. *Monotonic systems* may now be concisely defined as follows:

Definition 2.3 *An inference operator $\mathcal{S} : \mathbf{2}^{\mathcal{L}(\mathcal{S})} \to \mathbf{2}^{\mathcal{L}(\mathcal{S})}$ is a monotonic system if and only if $\mathcal{S}$ is a compact closure operator on $\mathcal{L}(\mathcal{S})$.*

[5]We use $S \subseteq_{\mathrm{f}} T$ to denote that a set S is a finite subset of a set T.

To end our treatment of monotonic systems, we demonstrate in Example 2.4 that systems $\mathcal{S}$ based on classical propositional logic (as introduced in Appendix A) meet the specification. Below we consider a system $\mathcal{S}$ which has a propositional language $\mathcal{L}(\mathcal{S}) = \mathcal{L}$ and which is defined by $\mathcal{S}(T) = \mathrm{Cn}(T)$ for all sets of sentences $T \subseteq \mathcal{L}(\mathcal{S})$. Recall that a sentence $\phi \in \mathcal{L}$ belongs to $\mathrm{Cn}(T)$ if and only if $\mathcal{M} \models \phi$ for all propositional models $\mathcal{M}$ of T.

Example 2.4 It is verified that $\mathcal{S}$ defined by $\mathcal{S}(T) = \mathrm{Cn}(T)$ is indeed a monotonic system. (i) It is clear that sentences of T are true in all models of T so that $T \subseteq \mathcal{S}(T)$. (ii) The second condition is met since $\mathrm{Cn}(T)$ and T have exactly the same models. Thus $\mathcal{S}(\mathcal{S}(T)) = \mathcal{S}(T)$. (iii) Finally, let T' be a superset of T. Now every model of T' is also a model of T. It follows that $\mathcal{S}(T) \subseteq \mathcal{S}(T')$. (iv) Now the set of theorems $\mathcal{S}(T) = \mathrm{Cn}(T)$ equals directly to $\bigcup\{\mathrm{Cn}(T') \,|\, T' \subseteq_{\mathrm{f}} T\} = \bigcup\{\mathcal{S}(T') \,|\, T' \subseteq_{\mathrm{f}} T\}$ by the compactness of propositional logic (Proposition A.3). We note also that a similar analysis can be conducted by starting from a proof-theoretical definition of $\mathrm{Cn}(\Sigma)$. □

By this example, we *identify* classical propositional logic with the family of monotonic systems $\mathcal{S}$ induced by propositional languages $\mathcal{L}$. Such systems $\mathcal{S}$ are called *propositional systems* in the sequel.

2.2 Non-Monotonic Systems

In order to demonstrate systems that do not conform to the specification of Definition 2.3 directly, we take Example 2.1 into reconsideration. We extend our set of rules by incorporating knowledge on correct and incorrect behavior of an and-gate. Our previous rules are augmented with a condition requiring that an and-gate is not broken. It is worth emphasizing that the word "not" appearing in the rules below does not denote classical negation which is often denoted by the symbol "$\neg$". Instead, our notion of negation is defined relative to reasoning with the rules themselves: if we cannot derive `Broken(G)` using the rules, then "not `Broken(G)`" is considered to hold[6]. There is inherent circularity in this definition, and this makes negation as failure a non-trivial concept. Two more rules have been added to describe when an and-gate is malfunctioning. Thus we end up with the five rules listed in Example 2.5. This example illustrates how the use of the negation described above has a profound effect on reasoning with the rules.

Example 2.5 Consider a system based on the following rules.

[6]In the literature [13], this form of negation is often called "negation as failure to derive".

If $\mathtt{In1(x,0)}$ and not $\mathtt{Broken(x)}$, then $\mathtt{Out(x,0)}$.
If $\mathtt{In2(x,0)}$ and not $\mathtt{Broken(x)}$, then $\mathtt{Out(x,0)}$.
If $\mathtt{In1(x,1)}$ and $\mathtt{In2(x,1)}$ and not $\mathtt{Broken(x)}$, then $\mathtt{Out(x,1)}$.
If ($\mathtt{In1(x,0)}$ or $\mathtt{In2(x,0)}$) and $\mathtt{Out(x,1)}$, then $\mathtt{Broken(x)}$.
If $\mathtt{In1(x,1)}$ and $\mathtt{In2(x,1)}$ and $\mathtt{Out(x,0)}$, then $\mathtt{Broken(x)}$.

If we take the atom $\mathtt{In1(G,0)}$ as an assumption, we obtain the atom $\mathtt{Out(G,0)}$ by the first rule, since the atom $\mathtt{Broken(G)}$ is not inferable by the rules. If we take the atom $\mathtt{In2(G,0)}$ as an additional assumption, we can still derive the atom $\mathtt{Out(G,0)}$. However, if extend our assumptions further by the atom $\mathtt{Out(G,1)}$, the conclusion $\mathtt{Out(G,0)}$ is no more obtained, because $\mathtt{Broken(G)}$ can be inferred by the last rule. Thus we have a counterexample showing that $T \subseteq T'$ does not imply $\mathcal{S}(T) \subseteq \mathcal{S}(T')$ in general and therefore the condition MON is not met by the reasoning system $\mathcal{S}$ under consideration. $\square$

The system of Example 2.5 is an example of a *non-monotonic system*, more particularly a rule-based system for *diagnosis* [95] in a slightly simplified setting. There are also other applications such as the previously mentioned systems for logic programming, truth maintenance, inheritance and reasoning about action that break the monotonicity condition MON. In such systems, conclusions may be retracted when new assumptions are introduced, because conclusions are dependent on assumptions that some *sentences are not theorems*. We reserve the term *non-theorem* for such sentences. In Example 2.5, the sentence $\mathtt{Broken(G)}$ is used as a non-theorem when the theorem $\mathtt{Out(G,0)}$ is obtained initially. The term non-theorem has an intensional flavor: a non-theorem ϕ is **not** intended to be a theorem, but we are ready to accept systems in which a non-theorem ϕ is possibly included in the set of theorems. This is a matter of consistency to be addressed when the various semantics of non-monotonic systems are later introduced.

For the moment, we concentrate on extending our specification of monotonic systems to the non-monotonic case. Two observations are necessary. Firstly, we should not abandon the concept of a closure and especially the requirement MON when modeling these kinds of systems. The reason is that once non-theorems have been decided, the resulting set of theorems should still be closed in the sense of Definition 2.2. On the other hand, the effect of non-theorems should be clear: the more sentences can be taken as non-theorems, the more theorems should be inferable. This property is easily verified for systems like the one in Example 2.5. Let us proceed with these intuitions in mind.

To incorporate non-theorems into our specification of monotonic systems, we resort to *binary inference operators* introduced by Thiele [110]. We refine Thiele's approach by allowing the possibility that non-theorems are selected from a sublanguage of the language for theorems. Therefore, for a system

Axiom	For every $T' \subseteq T \subseteq \mathcal{L}^+(\mathcal{S})$ and $N' \subseteq N \subseteq \mathcal{L}^-(\mathcal{S})$:
(REF)	$T \subseteq \mathcal{S}(T, N)$,
(CLO)	$\mathcal{S}(\mathcal{S}(T, N), N) = \mathcal{S}(T, N)$,
(MON)	$\mathcal{S}(T', N') \subseteq \mathcal{S}(T, N)$ and
(COM)	$\mathcal{S}(T, N) = \bigcup\{\mathcal{S}(T', N') \mid T' \subseteq_{\mathrm{f}} T \text{ and } N' \subseteq_{\mathrm{f}} N\}$.

Table 2: Specification of a non-monotonic system

$\mathcal{S}$, the language $\mathcal{L}(\mathcal{S})$ is a pair $\langle \mathcal{L}^+(\mathcal{S}), \mathcal{L}^-(\mathcal{S})\rangle$ of languages where $\mathcal{L}^-(\mathcal{S}) \subseteq \mathcal{L}^+(\mathcal{S})$. The languages $\mathcal{L}^+(\mathcal{S})$ and $\mathcal{L}^-(\mathcal{S})$ are dedicated to theorems and non-theorems, respectively. For non-monotonic systems, the *sublanguage relation* $\sqsubseteq$ is naturally defined by pairwise inclusion. Formally, a system $\mathcal{S}$ is a binary inference operator $\mathcal{S} : \mathbf{2}^{\mathcal{L}^+(\mathcal{S})} \times \mathbf{2}^{\mathcal{L}^-(\mathcal{S})} \to \mathbf{2}^{\mathcal{L}^+(\mathcal{S})}$. In this extended setting, a system $\mathcal{S}$ maps sets of sentences $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $N \subseteq \mathcal{L}^-(\mathcal{S})$ given both as premises or assumptions to a set of conclusions $\mathcal{S}(T, N) \subseteq \mathcal{L}^+(\mathcal{S})$. A specification of non-monotonic systems $\mathcal{S}$ is given by Definition 2.6 and the axioms of Table 2. The names of the axioms coincide with the monotonic case (as given in Table 1) but the axioms are distinguishable by the arity of the inference operator whenever referred to. We extend our terminology concerning theorems to the non-monotonic case: a non-monotonic system $\mathcal{S}$ maps a set of *theorems* T and a set of *non-theorems* N to a set of *theorems* $\mathcal{S}(T, N)$. As discussed earlier, it is possible that the set of non-theorems N is not necessarily disjoint from $\mathcal{S}(T, N)$. Note that $\mathcal{S}(T, N)$ is defined even when $T \cap N \neq \emptyset$. Also, the shorthand $\mathcal{S}(N)$ is introduced for $\mathcal{S}(\emptyset, N)$.[7]

Definition 2.6 *A binary inference operator* $\mathcal{S} : \mathbf{2}^{\mathcal{L}^+(\mathcal{S})} \times \mathbf{2}^{\mathcal{L}^-(\mathcal{S})} \to \mathbf{2}^{\mathcal{L}^+(\mathcal{S})}$ *is a non-monotonic system if and only if* $\mathcal{S}$ *satisfies the axioms* REF, CLO, MON *and* COM *in Table 2.*

Thiele [110] has analyzed properties of reasoning systems based on Reiter's default logic [94] and shown that such reasoning systems satisfy the axioms given in Table 2. Thiele formulated the first three conditions by requiring that $\lambda T.\mathcal{S}(T, N)$[8] is a closure operator for every fixed N and $\lambda N.\mathcal{S}(T, N)$ is a monotonic operator for every fixed T. Our notion of compactness is directly the notion of *double-compactness* proposed by Thiele. Note also that no further criteria are imposed for the set of non-theorems N at this stage.

[7] This notation is useful when we concentrate on a fixed system $\mathcal{S}$ and consider selecting a set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$.

[8] Following the conventions of λ-calculus, the expression $\lambda x.e(x)$ denotes a function that maps an argument x to the value of the expression $e(x)$.

By Definition 2.6, we have given a first class status to these axioms and based our notion of non-monotonic systems on them. In Section 2.3, we extend Thiele's work and show that there are other interesting non-monotonic systems that are based on other formalizations of non-monotonic reasoning than Reiter's default logic. Moreover, we take a considerable effort to use our notion of non-monotonic systems as a meta-theory for logics modeling non-monotonic reasoning. For instance, many properties of the semantics studied in Section 4 are established directly from the specification of Definition 2.6 and independently of a specific formalization of non-monotonic reasoning.

It may be somewhat confusing that our definition of non-monotonic systems is essentially based on a monotonicity condition on sets of theorems and non-theorems. Thus one might argue that it is unjustified to call such a system non-monotonic in the first place. However, at this stage, we are willing to view non-monotonic reasoning to be as close to classical monotonic reasoning as possible. Therefore our notion of non-monotonic systems is just ordinary monotonic systems augmented with an additional parameter, i.e. the set of non-theorems N which supports further derivation of theorems as dictated by the axiom MON. The non-monotonic behavior of these systems comes into play when the semantics of non-monotonic systems is addressed in Section 4 and the principles for selecting non-theorems are analyzed.

Our specification of a non-monotonic system is intentionally as loose as possible. Consequently, there are trivial systems that satisfy our requirements, but are of no practical value. For example, a non-monotonic system $\mathbb{S}$ satisfying $\mathbb{S}(T, N) = T$ for all T and N is certainly uninteresting, but still a system that meets the specification of non-monotonic system. In contrast to this, many useful systems conforming our specification are considered in Section 2.3. These systems are derived from the major non-monotonic logics presented in the literature [68, 72, 94]. Moreover, we are ready to accept further constraints on non-monotonic systems to address particular properties of non-monotonic reasoning within our framework.

2.3 Logic-Based Approaches to Non-Monotonic Reasoning

The foundations of non-monotonic reasoning were laid in the late seventies when a variety of non-monotonic logics were proposed and published in a joint number of the Artificial Intelligence Journal. These included logical frameworks such as McCarthy's *circumscription* [68], Reiter's *default logic* [94] and McDermott and Doyle's *non-monotonic logic* [70] which is a predecessor of Moore's *autoepistemic logic* [72]. This section is dedicated to showing how these logics provide instances of non-monotonic systems satisfying the specification of Definition 2.6. We proceed in the chronological order of the above proposals in Sections 2.3.1 - 2.3.3. Finally, some variants of these logics as well as the limits of the specification are discussed in Section 2.3.4.

For each formalization in turn, we introduce the basic definitions and work out the correspondence to our specification. More precisely, we identify the binary inference operators associated with non-monotonic logics mentioned above. Here we employ *stable semantics* as a guidance in each case. Stable semantics is probably the most obvious semantics for non-monotonic systems and it is based on a principle that a set of non-theorems associated with a non-monotonic system is selected as the complement of the corresponding set of theorems. A detailed definition of stable semantics is postponed until Section 4. Using stable semantics as a starting point of our analysis, we ensure that a uniform definition of stable semantics can be phrased in terms of our framework. This strategy is by no means too restrictive, since later on a variety of semantics turns out to be uniformly definable. Having extracted the associated inference operators for a formalization, it is verified that the conditions of non-monotonic systems is met. In particular, we point out how the requirements MON and COM of the specification of non-monotonic systems is satisfied by the associated inference operators.

It is evident that logic-based approaches to non-monotonic reasoning are emphasized for the moment, whereas practical applications are addressed in Section 10. This classification is superficial to some extent, since implementation techniques for circumscription, default reasoning and autoepistemic reasoning are already well understood nowadays (see, e.g., [80, 81, 82]). This increases considerably the potential of addressing practical applications directly within logical formalizations of non-monotonic reasoning.

2.3.1 Circumscription

McCarthy [68] proposed *circumscription* as a formalization of non-monotonic reasoning where only *minimal models* of a logical theory are taken into account. Here we concentrate on circumscription in the propositional case, but in a more general setting as proposed by Lifschitz [59]. As already discussed in Example 2.4, the set $\mathrm{Cn}(T)$ is model-theoretically defined as the intersection of $\mathrm{True}(\mathcal{M})$ for all models $\mathcal{M}$ of T. It is possible that a set of consequences, which is larger than $\mathrm{Cn}(T)$, is obtained by considering the intersection of $\mathrm{True}(\mathcal{M})$ for all models $\mathcal{M}$ of T that are minimal in the following sense.

Definition 2.7 *Given a set of atoms $P \subseteq \mathcal{A}$, a model $\mathcal{M}$ of $T \subseteq \mathcal{L}$ is P-minimal if and only if there is no model $\mathcal{M}'$ of T such that $\mathcal{M}' \cap P \subset \mathcal{M} \cap P$.*

Generally speaking, the idea behind circumscription is to set false as many propositional atoms from P as possible and minimize $\mathcal{M} \cap P$ in this way. The remaining propositional atoms in $\mathcal{A} - P$ may *vary* freely in the minimization process. A *minimal model theory* can be defined as a pair $\langle T, P \rangle$ where $T \subseteq \mathcal{L}$ is a propositional theory whose P-minimal models are of interest. To fit minimal model reasoning into our specification, we devise an alternative characterization for the P-minimal models in terms of a fixed point condition of Definition 2.8 which is derived from a characterization of circumscription by Etherington [26]. Here we introduce a notation

$$\mathrm{Neg}_P(N) = \neg(P \cap N) = \{\neg\phi \mid \phi \in P \cap N\}$$

for a set of non-theorems N. The idea is to *falsify* the atoms of P that appear in N by including the negative literals of $\mathrm{Neg}_P(N)$ in T. This is how the definition aims at capturing the stable semantics of a minimal model theory $\langle T, P \rangle$: the set of non-theorems N is chosen as $\overline{E} = \mathcal{L} - E$.

Definition 2.8 *For a set of atoms $P \subseteq \mathcal{A}$, a set $E \subseteq \mathcal{L}$ is a P-extension of $T \subseteq \mathcal{L}$ if and only if $E = \mathrm{Cn}(T \cup \mathrm{Neg}_P(\overline{E}))$.*

In Propositions 2.9 and 2.11 we show that P-minimal models of $T \subseteq \mathcal{L}$ coincide with the models of P-extensions of T. This characterization is closely related to the one given by Niemelä [79]. The difference is that we use propositional logic in the characterization whereas Niemelä uses autoepistemic logic [72] and considers a slightly more general setting corresponding to *parallel circumscription* [59] in the propositional case. This variant of circumscription is based on a notion of $\langle P, F \rangle$-minimal models which employs another set of atoms F whose elements are fixed in the minimization process. We shall later describe how this form of minimization is captured with P-minimal models.

Proposition 2.9 *If $\mathcal{M}$ is a P-minimal model of $T \subseteq \mathcal{L}$, then there is a P-extension $E \subseteq \mathcal{L}$ of T such that $\mathcal{M}$ is a model of E.*

Proof. Let $\{\mathcal{M}_i \mid i \in I\}$ be the class of models of T satisfying $\mathcal{M}_i \cap P = \mathcal{M} \cap P$. Thus every model $\mathcal{M}_i$ agrees with $\mathcal{M}$ on the atomic propositions in P and $\mathcal{M} \in \{\mathcal{M}_i \mid i \in I\}$ indicating that this class of models is non-empty. Define E as the intersection $\bigcap\{\mathrm{True}(\mathcal{M}_i) \mid i \in I\}$. It follows by the definition of E that (i) E has a model (i.e. is consistent) and (ii) $E \cap P = \mathcal{M}_i \cap P$ for all $i \in I$. It is then verified that (iii) each $\mathcal{M}_i$ is a model of $\mathrm{Neg}_P(\overline{E})$. So let $\neg\phi \in \mathrm{Neg}_P(\overline{E})$ hold. This implies $\phi \in P$ and $\phi \notin E$. Since $E \cap P = \mathcal{M}_i \cap P$, we obtain $\phi \notin \mathcal{M}_i$, $\mathcal{M}_i \not\models \phi$ and $\mathcal{M}_i \models \neg\phi$. Thus $\mathcal{M}_i \models \mathrm{Neg}_P(\overline{E})$ for all $i \in I$.

It is then proved that (iv) any model $\mathcal{M}'$ of T that is also a model of $\mathrm{Neg}_P(\overline{E})$ belongs to $\{\mathcal{M}_i \mid i \in I\}$. This is established by showing that $\mathcal{M}' \cap P = \mathcal{M} \cap P$.

So consider any $\phi \in P$. ($\subseteq$) Let $\phi \in \mathcal{M}'$ hold so that $\mathcal{M}' \models \phi$. Let us then assume that $\phi \notin E$. It follows that $\neg\phi \in \mathrm{Neg}_P(\overline{E}) = \neg(P \cap \overline{E})$. Since $\mathcal{M}'$ is a model of $\mathrm{Neg}_P(\overline{E})$ we obtain $\mathcal{M}' \models \neg\phi$ and $\mathcal{M}' \not\models \phi$, a contradiction. Thus it must be the case that $\phi \in E$. It follows that $\mathcal{M} \models \phi$ and $\phi \in \mathcal{M}$ by the definition of E. ($\supseteq$) Let $\phi \in \mathcal{M}$ hold. By the definition of E, we obtain that $\mathcal{M} \models \phi$ and $\phi \in E$. This implies $\neg\phi \notin E$ by the consistency of E. Then assume that $\phi \notin \mathcal{M}'$. By ($\subseteq$) we obtain that $\mathcal{M}' \cap P \subset \mathcal{M} \cap P$. Since $\mathcal{M}$ is a model of T, this implies that $\mathcal{M}$ is not a P-minimal model of T, a contradiction. Hence $\phi \in \mathcal{M}'$.

We are now in position to prove that E is the desired P-extension of T, i.e. E satisfies the fixed point condition $E = \mathrm{Cn}(T \cup \mathrm{Neg}_P(\overline{E}))$. ($\subseteq$) Let $\phi \in E$ hold. It follows that $\phi \in \mathrm{True}(\mathcal{M}_i)$, i.e. $\mathcal{M}_i \models \phi$, for all $i \in I$. Let us then assume that $T \cup \mathrm{Neg}_P(\overline{E}) \not\models \phi$. Then there is a model $\mathcal{M}'$ of $T \cup \mathrm{Neg}_P(\overline{E})$ such that $\mathcal{M}' \not\models \phi$. By (iv) $\mathcal{M}'$ belongs to the class $\{\mathcal{M}_i \mid i \in I\}$. This implies $\mathcal{M}' \models \phi$, a contradiction. Hence $T \cup \mathrm{Neg}_P(\overline{E}) \models \phi$. ($\supseteq$) Let $T \cup \mathrm{Neg}_P(\overline{E}) \models \phi$ hold. By (iii) each $\mathcal{M}_i$ is a model of $T \cup \mathrm{Neg}_P(\overline{E})$. It follows that $\mathcal{M}_i \models \phi$ and $\phi \in \mathrm{True}(\mathcal{M}_i)$ for all $i \in I$. Thus $\phi \in E$. $\square$

Lemma 2.10 *If $E \subseteq \mathcal{L}$ is a P-extension of $T \subseteq \mathcal{L}$ and $\mathcal{M}$ is a model of E, then $\mathcal{M} \cap P = E \cap P$.*

Proof. Since E is a P-extension of T, it holds that $E = \mathrm{Cn}(T \cup \mathrm{Neg}_P(\overline{E}))$. Then consider any $\phi \in P$. ($\subseteq$) Let $\phi \in \mathcal{M}$ so that $\mathcal{M} \models \phi$. Let us then assume that $\phi \notin E$. This implies that $\neg\phi \in \mathrm{Neg}_P(\overline{E})$ and $\neg\phi \in E$. Since $\mathcal{M}$ is a model of E we obtain $\mathcal{M} \models \neg\phi$ and $\mathcal{M} \not\models \phi$, a contradiction. Hence $\phi \in E$. ($\supseteq$) Let $\phi \in E$ hold. Since $\mathcal{M}$ is a model of E, we have that $\mathcal{M} \models \phi$. It follows that $\phi \in \mathcal{M}$, since ϕ is atomic. $\square$

Proposition 2.11 *Let $E \subseteq \mathcal{L}$ be a P-extension of $T \subseteq \mathcal{L}$. If $\mathcal{M}$ is a model of E, then $\mathcal{M}$ is a P-minimal model of T.*

Proof. To show the P-minimality of $\mathcal{M}$, we assume there is a model $\mathcal{M}'$ of T such that $\mathcal{M}' \cap P \subset \mathcal{M} \cap P$. Then there is $\phi \in P$ such that $\phi \in \mathcal{M} - \mathcal{M}'$. It follows by Lemma 2.10 that $\phi \in E$ and $T \cup \mathrm{Neg}_P(\overline{E}) \models \phi$, since $E = \mathrm{Cn}(T \cup \mathrm{Neg}_P(\overline{E}))$. Also $\mathrm{Neg}_P(\overline{E}) = \neg(P \cap \overline{E}) = \neg(P - E) = \neg(P - \mathcal{M})$ holds by Lemma 2.10. Thus we have that $T \cup \neg(P - \mathcal{M}) \models \phi$. On the other hand $\mathcal{M}' \cap P \subset \mathcal{M} \cap P$ implies $\neg(P - \mathcal{M}) \subset \neg(P - \mathcal{M}')$ so that $T \cup \neg(P - \mathcal{M}') \models \phi$ follows by the monotonicity of $\models$. But then $T \cup \neg(P - \mathcal{M}')$ is propositionally inconsistent since $\phi \in P$ and $\phi \notin \mathcal{M}'$ imply $\neg\phi \in \neg(P - \mathcal{M}')$. Since $\mathcal{M}'$ is a model of T and $\neg\psi \in \neg(P - \mathcal{M}')$ implies $\psi \in P$, $\psi \notin \mathcal{M}'$ and $\mathcal{M}' \models \neg\psi$, it follows that $\mathcal{M}' \models T \cup \neg(P - \mathcal{M}')$. But this contradicts the propositional inconsistency of $T \cup \neg(P - \mathcal{M}')$. Hence $\mathcal{M}$ is a P-minimal model of T. $\square$

In Theorem 2.12, we relate P-extensions of T with sets of sentences $\mathrm{True}(\mathcal{M})$ induced by P-minimal models $\mathcal{M}$ of T. In a restricted case $P = \mathcal{A}$ (i.e. all atoms are minimized in harmony with McCarthy's approach) the correspondence becomes even tighter: for each P-minimal model $\mathcal{M}$ of T, there is a corresponding P-extension, namely the theory $T = \mathrm{True}(\mathcal{M})$ having the unique model $\mathcal{M}$ (note that $\mathcal{A}$ determines the propositional language $\mathcal{L}$ and $P = \mathcal{A}$). The characterization based on P-extensions is indeed interesting. If we define $\mathrm{Cn}(T)$ proof-theoretically and use the fixed point equation of Definition 2.8, we may redefine circumscription without an explicit reference to the P-minimal models. McCarthy [68] provides such a proof-theoretical characterization, but uses second order logic instead of a fixed point equation.

Theorem 2.12 *Let $T \subseteq \mathcal{L}$ be a set of sentences and $P \subseteq \mathcal{A}$ a set of atoms. The intersection of P-extensions of T coincides with the intersection of the sets of sentences* $\mathrm{True}(\mathcal{M})$ *for all P-minimal models $\mathcal{M}$ of T.*

Proof. ($\subseteq$) Assume that $\phi \in \mathcal{L}$ does not belong to the set of sentences $\mathrm{True}(\mathcal{M})$ for some P-minimal model $\mathcal{M}$ of T. By Proposition 2.9, there is a P-extension of T such that $\mathcal{M}$ is a model of E. Because E is propositionally closed, $\mathcal{M}$ is model of E and $\phi \notin \mathrm{True}(\mathcal{M})$, it follows that $\phi \notin E$. Thus ϕ is not contained in the intersection of P-extensions of T. ($\supseteq$) Assume that $\phi \in \mathcal{L}$ is not in the intersection of P-extensions of T. Then there is a P-extension E of T such that $\phi \notin E$. Since E is propositionally closed, there is a model $\mathcal{M}$ of E such that $\phi \notin \mathrm{True}(\mathcal{M})$. Then $\mathcal{M}$ is a P-minimal model of $\mathcal{M}$ by Proposition 2.11. Thus ϕ does not belong to the intersection of the sets of sentences $\mathrm{True}(\mathcal{M})$ for the P-minimal models of T. □

Starting from the characterization of Theorem 2.12 we may relate circumscription to our framework of non-monotonic systems. Naturally a P-extension $E = \mathrm{Cn}(T \cup \mathrm{Neg}_P(\overline{E}))$ gives us a set of theorems associated with a minimal model theory $\langle T, P \rangle$. The same equation identifies how non-theorems are present in circumscription: the minimization process involves falsifying atoms of P that appear in $\overline{E}$, i.e. are not included in the set of theorems E. These observations lead to a definition a binary inference operator $\mathfrak{C}_P$ given in Definition 2.13. Theorem 2.14 proves $\mathfrak{C}_P$ to be a non-monotonic system in the sense of Definition 2.6. We identify circumscription with the family of non-monotonic systems $\mathfrak{C}_P$ induced by sets of atoms P subject to minimization. Such systems are called *minimal model systems* in the sequel.

Definition 2.13 *For a set of atoms $P \subseteq \mathcal{A}$, define a binary inference operator $\mathfrak{C}_P$ by setting $\mathcal{L}(\mathfrak{C}_P) = \langle \mathcal{L}, \mathcal{L} \rangle$ and for all $T \subseteq \mathcal{L}$ and $N \subseteq \mathcal{L}$,*

$$\mathfrak{C}_P(T, N) = \mathrm{Cn}(T \cup \mathrm{Neg}_P(N)).$$

Theorem 2.14 *For a set of atoms $P \subseteq \mathcal{A}$, the binary inference operator $\mathfrak{C}_P$ is a non-monotonic system.*

Proof. For a fixed $N \subseteq \mathcal{L}$, the operator $\lambda T.\mathfrak{C}_P(T, N)$ is a closure operator by definition. Then let $N_1 \subseteq N_2$. It follows that $\mathrm{Neg}_P(N_1) \subseteq \mathrm{Neg}_P(N_2)$ and $\mathrm{Cn}(T \cup \mathrm{Neg}_P(N_1))$ is a subset of $\mathrm{Cn}(T \cup \mathrm{Neg}_P(N_2))$ by the monotonicity of $\models$. So $\lambda N.\mathfrak{C}_P(T, N)$ is monotonic for a fixed T. Finally, consider the compactness of the operator $\mathfrak{C}_P$ so let T and N be any subsets of $\mathcal{L}$. It should be established that $\mathrm{Cn}(T \cup \mathrm{Neg}_P(N))$ equals to the union of $\mathrm{Cn}(T' \cup \mathrm{Neg}_P(N'))$ for all $T' \subseteq_{\mathrm{f}} T$ and $N' \subseteq_{\mathrm{f}} N$. ($\subseteq$) Let $\phi \in \mathrm{Cn}(T \cup \mathrm{Neg}_P(N))$ hold. Since $\models$ is compact, there is a finite subset S of $T \cup \mathrm{Neg}_P(N)$ such that $\phi \in \mathrm{Cn}(S)$. Let $T' = T \cap S$ and $N' = \{\phi \mid \neg\phi \in \mathrm{Neg}_P(N) \cap S\}$. Thus T' and N' are finite subsets of T and N, respectively. It follows that $\phi \in \mathrm{Cn}(T' \cup \mathrm{Neg}_P(N'))$ for the given $T' \subseteq_{\mathrm{f}} T$ and $N' \subseteq_{\mathrm{f}} N$ so that ϕ belongs to the desired union. ($\supseteq$) The other direction holds directly by the monotonicity of $\models$. □

To end our treatment of circumscription we point out differences to approaches by McCarthy [68] and Lifschitz [59]. These approaches consider circumscription in a wider context of predicate logic whereas we have concentrated on the propositional case. According to McCarthy's approach, all propositional atoms are subject to minimization. The notion of P-minimal models restricts minimization to a fixed set of atomic propositions P. From this point of view, McCarthy considers only $\mathcal{A}$-minimal models, i.e. restricted minimal model theories $\langle T, \mathcal{A}\rangle$ where $T \subseteq \mathcal{L}$ for the propositional language $\mathcal{L}$ based on the set of atoms $\mathcal{A}$. In this restricted case, our $\mathcal{A}$-extensions coincide with the propositional characterization used by Niemelä [83]. However, it seems that it is not very practical from the knowledge representation perspective to have unrestricted minimization and typically $P \subset \mathcal{A}$ is chosen to restrict minimization to specific propositional atoms (see, e.g., Examples 2.35 and 10.20).

This idea is already present in parallel circumscription proposed by Lifschitz [59]. As already mentioned, this form of circumscription involves an additional parameter: a set of atoms $F \subseteq \mathcal{A}$ (disjoint from $P \subseteq \mathcal{A}$) to be *fixed* in the minimization. The corresponding notion of minimality is $\langle P, F\rangle$-minimality: a model $\mathcal{M}$ of T is $\langle P, F\rangle$-minimal if and only if there is no model $\mathcal{M}'$ of T such that $\mathcal{M}' \cap F = \mathcal{M} \cap F$ and $\mathcal{M}' \cap P \subset \mathcal{M} \cap P$. The atoms $\mathcal{A} - (P \cup F)$ may vary freely in parallel circumscription. There are at least two ways of capturing parallel circumscription in our framework. Firstly, it is possible to generalize our notion of P-extensions to a notion of $\langle P, F\rangle$-extensions and generalize Theorem 2.12 for $\langle P, F\rangle$-minimal models. Niemelä has also provided a characterization of parallel circumscription [82]. We resort to another approach proposed by de Kleer and Konolige [16]. The effect of fixing an atomic proposition $\phi \in F$ is obtained by augmenting $T \subseteq \mathcal{L}$ with an equivalence $\overline{\phi} \leftrightarrow \neg\phi$ where $\overline{\phi}$ is a new atom (not appearing in $\mathcal{A}$) and adding both ϕ and

$\overline{\phi}$ in P, to be minimized simultaneously. To make our argument precise, we provide the following proposition without a proof.

Proposition 2.15 *A model $\mathcal{M}$ of a propositional theory $T \subseteq \mathcal{L}$ is $\langle P, F \rangle$-minimal if and only if there is a $P \cup F \cup \{\overline{\phi} \mid \phi \in F\}$-minimal model $\mathcal{M}'$ of $T' = T \cup \{\overline{\phi} \leftrightarrow \neg\phi \mid \phi \in F\}$ such that $\mathcal{M} = \mathcal{M}' \cap \mathcal{A}$.*

2.3.2 Default Logic

Reiter [94] proposed *default logic* as a formalization of default reasoning. We consider default logic in the propositional case. According to Reiter, a *default theory* is a pair $\langle D, T \rangle$ where $T \subseteq \mathcal{L}$ is a propositional theory and D is a set of *default rules* (or simply *defaults* for short) of the form $\frac{\alpha:\beta_1,\ldots,\beta_n}{\gamma}$ where α, $\beta_1, \ldots, \beta_n$, and γ are sentences of $\mathcal{L}$. The idea behind a default is that the *consequent* γ can be inferred if the *prerequisite* α has been inferred and it is consistent to assume the *justifications* $\beta_1, \ldots, \beta_n$. If the prerequisite α is $\top$, then a default is called *prerequisite-free*. A prerequisite-free default theory has only prerequisite-free default rules.

The sets of theorems associated with a default theory are typically called *extensions*. As our first step, we present a definition due to Marek and Truszczyński [67] which is equivalent to that of Reiter's [94]. Given a theory $E \subseteq \mathcal{L}$, Marek and Truszczyński reduce a set of default rules D to a set of *inference rules* $D^E = \{\frac{\alpha}{\gamma} \mid \frac{\alpha:\beta_1,\ldots,\beta_n}{\gamma} \in D$ and $\neg\beta_1 \notin T, \ldots, \neg\beta_n \notin E\}$. Here it is important to note that $\neg\beta_i \notin E$ if and only if $E \cup \{\beta_i\}$ is propositionally consistent – provided that E is propositionally closed. By this connection, the reduction D^E interprets the justifications of default rules correctly for propositionally closed theories E. A set of inference rules R gives rise to a notion of a closure:

Definition 2.16 *Let $T \subseteq \mathcal{L}$ and let R be a set of inference rules in $\mathcal{L}$. Define $\mathrm{Cn}^R(T)$ as the smallest subset of $\mathcal{L}$ that (i) contains T, (ii) is propositionally closed and (iii) is closed under the rules of R, i.e.*

$$\text{if } \tfrac{\alpha}{\gamma} \in R \text{ and } \alpha \in \mathrm{Cn}^R(T), \text{ then also } \gamma \in \mathrm{Cn}^R(T).$$

Using these notions, default extensions can be defined concisely as follows.

Definition 2.17 (Marek and Truszczyński [67]) *For a set of defaults D in $\mathcal{L}$, a set $E \subseteq \mathcal{L}$ is an extension of $T \subseteq \mathcal{L}$ if and only if $E = \mathrm{Cn}^{D^E}(T)$.*

However, this definition is not yet appropriate to identify the role of non-theorems in default reasoning. Therefore we propose a slightly different reduction of D, namely $D_N = \{\frac{\alpha}{\gamma} \mid \frac{\alpha:\beta_1,\ldots,\beta_n}{\gamma} \in D$ and $\{\neg\beta_1, \ldots, \neg\beta_n\} \subseteq N\}$ where

$N \subseteq \mathcal{L}$ is a set of sentences. Using this reduction, we obtain an alternative definition of default extensions in Proposition 2.18.

Proposition 2.18 *For a set of defaults D in $\mathcal{L}$, a set $E \subseteq \mathcal{L}$ is an extension of $T \subseteq \mathcal{L}$ if and only if $E = \mathrm{Cn}^{D_{\overline{E}}}(T)$.*

Proof. Since the extensions E of $T \subseteq \mathcal{L}$ are propositionally closed theories, a justification β_i where $0 < i \leq n$ is consistent with an extension E if and only if $\neg\beta_i \notin E$. This implies that $D_{\overline{E}} = D^E$. Under this equality, the fixed point conditions $E = \mathrm{Cn}^{D_{\overline{E}}}(T)$ and $E = \mathrm{Cn}^{D^E}(T)$ coincide. □

The reformulation of default extensions in Proposition 2.18 gives us an idea how non-theorems are used in default reasoning. In spirit of stable semantics, a set of non-theorems $N = \overline{E}$ is employed to reduce D to a set of inference rules $D_{\overline{E}}$. Following this guideline, we introduce a binary inference operator $\mathfrak{D}_D$ for each set of defaults D in Definition 2.19.

Definition 2.19 *For a set of defaults D in $\mathcal{L}$, define a binary inference operator $\mathfrak{D}_D$ by setting $\mathcal{L}(\mathfrak{D}_D) = \langle \mathcal{L}, \mathcal{L} \rangle$ and for all $T \subseteq \mathcal{L}$ and $N \subseteq \mathcal{L}$,*

$$\mathfrak{D}_D(T, N) = \mathrm{Cn}^{D_N}(T).$$

To prove that the binary inference operator $\mathfrak{D}_D$ is a non-monotonic system, we shall characterize the closure $\mathrm{Cn}^R(T)$ using a notion of *R-proofs* introduced in Definition 2.20. The idea is that each step of an R-proof corresponds to an application of an inference rule in R. Theorem 2.21 shows that the theory $\mathrm{Cn}^R(T)$ is captured by this notion of R-proofs. In the sequel, this equality is often used without an explicit reference to Theorem 2.21. Marek and Truszczyński [67, p. 38] use a similar definition. The difference is that they incorporate inference rules to the definition of a proof system for propositional logic whereas our definition is built upon propositional consequence. Moreover, Marek and Truszczyński take their notion of proofs as a definition of the closure $\mathrm{Cn}^R(T)$ rather than showing that it captures the closure correctly. Nevertheless, the two definitions of the closure $\mathrm{Cn}^R(T)$ coincide.

Definition 2.20 *Let R be a set of inference rules. A sequence $\gamma_1, ..., \gamma_n$ is an R-proof of ϕ from T if and only if $T \cup \{\gamma_1, ..., \gamma_n\} \models \phi$ and for each $0 < i \leq n$ there is a rule $\frac{\alpha_i}{\gamma_i} \in R$ such that $T \cup \{\gamma_1, ..., \gamma_{i-1}\} \models \alpha_i$. Let $T \vdash_R \phi$ denote that ϕ has an R-proof from T.*

Theorem 2.21 *For a set of inference rules R and a theory T, the closure $\mathrm{Cn}^R(T) = \{\phi \,|\, T \vdash_R \phi\}$.*

Proof. ($\subseteq$) It is proved that $T' = \{\phi \,|\, T \vdash_R \phi\}$ satisfies the requirements (i-iii) of Definition 2.16. (i) Let $\phi \in T$ so that $T \models \phi$ holds. Thus ϕ has an empty R-proof from T implying that $T \vdash_R \phi$. It follows that $T \subseteq T'$. (ii) Assume that $T' \models \phi$. By the compactness of $\models$ only finitely many sentences $\phi_1, ..., \phi_n$ from T' are necessary to realize $\{\phi_1, ..., \phi_n\} \models \phi$. Note that each ϕ_i is a member of T' and has an R-proof from T. An R-proof of ϕ is obtained by concatenating the R-proofs of $\phi_1, ..., \phi_n$. Thus $\phi \in T'$ and T' is closed under propositional consequence. (iii) Assume that $\alpha \in T'$ for some $\frac{\alpha}{\gamma} \in R$. It follows that $T \vdash_R \alpha$ and α has an R-proof $\gamma_1, ..., \gamma_n$ from T. An R-proof of γ is obtained as a simple concatenation $\gamma_1, ..., \gamma_n, \gamma$ so that $T \vdash_R \gamma$. Thus $\gamma \in T'$ and T' is closed under the rules of R. By (i – iii) above, the theory T' satisfies the closure properties given in Definition 2.16. Since $\mathrm{Cn}^R(T)$ is the smallest theory with these properties $\mathrm{Cn}^R(T) \subseteq T'$ follows.

($\supseteq$) It is proved that $T \vdash_R \phi$ implies $\phi \in \mathrm{Cn}^R(T)$ by induction on the lengths of R-proofs. In the base case, the R-proof is empty so that $T \models \phi$ holds and $\phi \in \mathrm{Cn}^R(T)$ follows, since $\mathrm{Cn}^R(T)$ contains T and is propositionally closed. Induction step follows. Now ϕ has an R-proof $\gamma_1, ..., \gamma_n$ from T. Then each γ_i is R-provable from T in less than n steps. Thus each γ_i is included in $\mathrm{Cn}^R(T)$ by the inductive hypothesis. Since also $T \subseteq \mathrm{Cn}^R(T)$, $T \cup \{\gamma_1, ..., \gamma_n\} \models \phi$ and $\mathrm{Cn}^R(T)$ is propositionally closed, it follows that $\phi \in \mathrm{Cn}^R(T)$. □

The notion of R-proofs is useful in analyzing the properties of $\mathrm{Cn}^R(T)$. In Proposition 2.22 we provide the basic monotonicity property that can also established from the definition of $\mathrm{Cn}^R(T)$ directly. The subsequent theorem shows a compactness result for reasoning with inference rules. This means that only finitely many inference rules are required to show that a sentence ϕ belongs to $\mathrm{Cn}^R(T)$ even though R and T were infinite.

Proposition 2.22 *If $R \subseteq R'$ and $T \subseteq T'$, then $\mathrm{Cn}^R(T) \subseteq \mathrm{Cn}^{R'}(T')$.*

Proof. Let $\phi \in \mathrm{Cn}^R(T)$, i.e. there is an R-proof $\gamma_1, ..., \gamma_n$ of ϕ from T. By the monotonicity of $\models$ the same sequence is an R-proof of ϕ from T'. This holds also when more inference rules are available. Thus $\gamma_1, ..., \gamma_n$ is an R'-proof of ϕ from T' so $\phi \in \mathrm{Cn}^{R'}(T')$. □

Theorem 2.23 *For a set of inference rules R and a propositional theory T,*

$$\mathrm{Cn}^R(T) = \bigcup \{\mathrm{Cn}^{R'}(T') \,|\, R' \subseteq_{\mathrm{f}} R \textit{ and } T' \subseteq_{\mathrm{f}} T\}.$$

Proof. ($\subseteq$) Let $\phi \in \mathrm{Cn}^R(T)$ hold. By Theorem 2.21, it holds that $T \vdash_R \phi$, i.e. there is an R-proof $\gamma_1, ..., \gamma_n$ of ϕ from T. Let R' be the finite set of inference rules $\frac{\alpha_1}{\gamma_1}, ..., \frac{\alpha_n}{\gamma_n}$ involved in this proof. Obviously the same sequence provides an R'-proof of ϕ from T. Recall that $T \cup \{\gamma_1, ..., \gamma_{i-1}\} \models \alpha_i$ holds for $0 < i \leq$

n. By the compactness of $\models$, for each α_i there is a finite subset T_i of T such that $T_i \cup \{\gamma_1, ..., \gamma_{i-1}\} \models \alpha_i$ still holds. The union $T' = T_1 \cup ... \cup T_n$ is clearly finite and $T' \cup \{\gamma_1, ..., \gamma_{i-1}\} \models \alpha_i$ follows for each α_i by the monotonicity of $\models$. Thus $T' \vdash_{R'} \phi$ and $\phi \in \mathrm{Cn}^{R'}(T')$ by Theorem 2.21. ($\supseteq$) A sentence ϕ belongs to the union only if $\phi \in \mathrm{Cn}^{R'}(T')$ for some $R' \subseteq_{\mathrm{f}} R$ and $T' \subseteq_{\mathrm{f}} T$. It follows that $\phi \in \mathrm{Cn}^{R}(T)$ by Proposition 2.22. $\square$

Thiele [110] has established an equivalent of Theorem 2.24 which states that the binary inference operator $\mathfrak{D}_D$ fulfills our specification. We include a proof of the theorem for the sake of completeness of our presentation. In the proof, we exploit the properties of the closure $\mathrm{Cn}^{R}(T)$ proved so far. From now on, we identify default logic with the family of non-monotonic systems $\mathfrak{D}_D$ induced by sets of defaults D. We call such systems *default systems.*

Theorem 2.24 (Thiele [110]) *For a set of defaults D in $\mathcal{L}$, the binary inference operator $\mathfrak{D}_D$ is a non-monotonic system.*

Proof. For a fixed N, the operator $\lambda T.\mathfrak{D}_D(T, N)$ is a closure operator by definition. Then let $N_1 \subseteq N_2 \subseteq \mathcal{L}$. It follows that $D_{N_1} \subseteq D_{N_2}$ and $\mathrm{Cn}^{D_{N_1}}(T) \subseteq \mathrm{Cn}^{D_{N_2}}(T)$ by Proposition 2.22. Thus $\lambda N.\mathfrak{D}_D(T, N)$ is monotonic for a fixed T. To show compactness of $\mathfrak{D}_D$, let us pick any $T \subseteq \mathcal{L}$ and $N \subseteq \mathcal{L}$. It is shown that $\mathrm{Cn}^{D_N}(T)$ coincides with the union of $\mathrm{Cn}^{D_{N'}}(T')$ for all $T' \subseteq_{\mathrm{f}} T$ and $N' \subseteq_{\mathrm{f}} N$. ($\subseteq$) Let $\phi \in \mathrm{Cn}^{D_N}(T)$. By Theorem 2.23 there are finite subsets R' and T' of D_N and T (respectively) such that $\phi \in \mathrm{Cn}^{R'}(T')$. Let N' be the union of $\{\neg\beta_1, ..., \neg\beta_n\}$ for each $\frac{\alpha}{\gamma} \in R'$ where $\frac{\alpha:\beta_1,...,\beta_n}{\gamma}$ is any of the default rules of D such that $\{\neg\beta_1, ..., \neg\beta_n\} \subseteq N$. These guarantee that $T' \subseteq_{\mathrm{f}} T$, $N' \subseteq_{\mathrm{f}} N$ and $R' \subseteq D_{N'}$. Thus $\phi \in \mathrm{Cn}^{D_{N'}}(T')$ by Proposition 2.22 and ϕ belongs to the desired union. ($\supseteq$) This follows by Proposition 2.22. $\square$

2.3.3 Autoepistemic Logic

It is time to present Moore's *autoepistemic logic* [72, 73] and a strengthening of this logic as suggested by Marek and Truszczyński. After this, we propose a novel variant of autoepistemic logic based on an idea of separating beliefs from disbeliefs. Finally, we explain how Moore's and Marek and Truszczyński's autoepistemic logic are captured in the variant.

Moore's autoepistemic logic is a reconstruction of McDermott and Doyle's non-monotonic logic [69, 70] which models sets of beliefs of an introspective agent. An *autoepistemic theory* Σ is simply a subset of an *autoepistemic language* $\mathcal{L}_{\mathrm{B}}$ which is based on a set of propositional atoms $\mathcal{A}$, propositional connectives and a single modality $\mathbf{B}$ that formalizes a notion of *belief.* Sentences of the form $\mathbf{B}\phi$ are referred to as $\mathbf{B}$-*atoms* and autoepistemic logic treats them as propositional atoms. Therefore, an autoepistemic model $\mathcal{M}$ is simply a subset of $\mathcal{A} \cup \{\mathbf{B}\phi \,|\, \phi \in \mathcal{L}_{\mathrm{B}}\}$, i.e. $\mathcal{M}$ determines which of the propositional atoms and **B**-atoms are true. The logical consequences $\mathrm{Cn}(\Sigma)$ of $\Sigma \subseteq \mathcal{L}_{\mathrm{B}}$ are determined by the autoepistemic models of Σ in the standard way (as described in Appendix A). Moore [72] defines *stable expansions* as the possible sets of beliefs of the agent given an autoepistemic theory as its initial assumptions.

Definition 2.25 (Moore [72]) *A set $\Delta \subseteq \mathcal{L}_{\mathrm{B}}$ is a stable expansion of an autoepistemic theory Σ if and only if $\Delta = \mathrm{Cn}(\Sigma \cup \mathbf{B}\Delta \cup \neg\mathbf{B}\overline{\Delta})$.*

The intuition behind this definition is as follows: an ideal and rational agent believes exactly the logical consequences of its premises Σ, its beliefs $\mathbf{B}\Delta = \{\mathbf{B}\phi \,|\, \phi \in \Delta\}$ obtained by *positive introspection* and its disbeliefs $\neg\mathbf{B}\overline{\Delta} = \{\neg\mathbf{B}\phi \,|\, \phi \in \overline{\Delta}\}$ obtained by *negative introspection*. In the definition, the complement of Δ is taken relative to $\mathcal{L}_{\mathrm{B}}$, i.e. $\overline{\Delta} = \mathcal{L}_{\mathrm{B}} - \Delta$. Moore's original definition is often considered unsatisfactory, since it allows stable expansions where beliefs support themselves in a circular way and consequently beliefs are *weakly grounded* in the premises Σ. Let us illustrate this phenomenon:

Example 2.26 Consider an autoepistemic theory $\Sigma = \{\mathbf{B}\mathrm{p} \rightarrow \mathrm{p}\}$ which has two stable expansions: $\Delta_1 = \{\mathrm{p}, \mathbf{B}\mathrm{p}, \ldots\}$ and $\Delta_2 = \{\neg\mathbf{B}\mathrm{p}, \ldots\}$. In the expansion Δ_1, the belief p is supported by itself: p is a logical consequence of Σ and **B**p which is obtained by positive introspection given that p is believed, i.e. $\mathrm{p} \in \Delta_1$. The expansion Δ_2 is considered more intuitive, since it is unjustifiable to believe p given just the premises Σ of the agent, since $\Sigma \not\models \mathrm{p}$. □

To exclude circularity in positive introspection, Marek and Truszczyński propose *iterative expansions* [66] defined below. It can be shown that iterative expansions of Σ are stable expansions of Σ, but the converse does not hold in general: the stable expansion Δ_1 in Example 2.26 serves as a counter-example.

Definition 2.27 (Marek and Truszczyński [66]) *A set $\Delta \subseteq \mathcal{L}_{\mathrm{B}}$ is an iterative expansion of an autoepistemic theory Σ iff $\Delta = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup \neg\mathbf{B}\overline{\Delta})$.*

Here the positive introspection of a sentence $\phi \in \mathcal{L}_{\mathrm{B}}$ has been replaced with the *necessitation rule* $\phi \vdash \mathbf{B}\phi$ for the modality **B**. By this rule we can infer $\mathbf{B}\phi$ given that we can infer ϕ. The theory $\mathrm{Cn}_{\mathbf{B}}(\Sigma)$ is defined as the least theory which contains Σ and is closed with respect to propositional consequence and the necessitation rule $\phi \vdash \mathbf{B}\phi$. Gottlob defines this closure in terms of **B**-*proofs* [37] whose steps correspond to applications of $\phi \vdash \mathbf{B}\phi$. The definition of **B**-proofs is given below. Note the similarity with respect to the notion of R-proofs for a set of inference rules R introduced in Definition 2.20.

Definition 2.28 *Let Σ be an autoepistemic theory. A sequence $\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n$ of $\mathbf{B}$-atoms is a $\mathbf{B}$-proof of ϕ from Σ if and only if $\Sigma \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_{i-1}\} \models \phi_i$ for each $0 < i \leq n$ and $\Sigma \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\} \models \phi$. Let $\Sigma \vdash_{\mathbf{B}} \phi$ denote that ϕ has a $\mathbf{B}$-proof from Σ.*

It can be shown in analogy to Theorem 2.21 that $\mathrm{Cn}_{\mathbf{B}}(\Sigma)$ coincides with $\{\phi \,|\, \Sigma \vdash_{\mathbf{B}} \phi\}$ and we assume this identification in the sequel. This is also a proper moment for pointing out monotonicity and compactness properties of the closure in Proposition 2.29 and Theorem 2.30. We omit the proofs because of the obvious analogy to the proofs of Proposition 2.22 and 2.23, respectively.

Proposition 2.29 *If $\Sigma \subseteq \Sigma' \subseteq \mathcal{L}_{\mathrm{B}}$, then $\mathrm{Cn}_{\mathbf{B}}(\Sigma) \subseteq \mathrm{Cn}_{\mathbf{B}}(\Sigma')$.*

Theorem 2.30 *For all $\Sigma \subseteq \mathcal{L}_{\mathrm{B}}$, the theory $\mathrm{Cn}_{\mathbf{B}}(\Sigma) = \bigcup\{\mathrm{Cn}_{\mathbf{B}}(\Sigma') \,|\, \Sigma' \subseteq_{\mathrm{f}} \Sigma\}$.*

Separating Beliefs from Disbeliefs

The variant of autoepistemic logic which considers only the iterative expansions of Σ is already very well-suited for non-monotonic reasoning tasks such as default reasoning. To make autoepistemic logic even closer to default logic, we propose an idea of *separating* disbeliefs from beliefs in terms of two modal operators. There are two central aims for this kind of a separation. One is to establish a reasonable notion of stationary semantics [89] for autoepistemic theories, since conventional autoepistemic theories lack such a semantics. The other is to solve problems related with Konolige's translation function [52] which fails to capture default extensions with autoepistemic expansions.

The idea behind separation is to introduce distinguished modalities $\mathbf{B}$ and $\mathbf{D}$ for *belief* and *disbelief*, respectively. In this extended setting, separated autoepistemic theories Σ are subsets of the corresponding propositional bimodal language $\mathcal{L}_{\mathrm{BD}}$ based on a set of propositional atoms $\mathcal{A}$. Sentences of the forms $\mathbf{B}\phi$ and $\mathbf{D}\phi$ are called $\mathbf{B}$-*atoms* and $\mathbf{D}$-*atoms*, respectively. Altogether $\mathbf{B}$-atoms and $\mathbf{D}$-atoms are called $\mathbf{BD}$-atoms. Like before, $\mathbf{BD}$-atoms are treated as propositional atoms. The notion of autoepistemic models is revised to subsets of $\mathcal{A} \cup \{\mathbf{B}\phi \,|\, \phi \in \mathcal{L}_{\mathrm{BD}}\} \cup \{\mathbf{D}\phi \,|\, \phi \in \mathcal{L}_{\mathrm{BD}}\}$. Basically, our idea is that disbelief in a sentence ϕ is now represented in terms of the sentence $\mathbf{D}\phi$ rather than the sentence $\neg\mathbf{B}\phi$. A benefit is that some unwanted propositional interactions of belief literals $\mathbf{B}\phi$ and $\neg\mathbf{B}\phi$ are avoided. For instance, the sentences $\mathbf{B}\phi \vee \neg\mathbf{B}\phi$ and $\neg(\mathbf{B}\phi \wedge \neg\mathbf{B}\phi)$ are propositional tautologies whereas the corresponding bimodal sentences $\mathbf{B}\phi \vee \mathbf{D}\phi$ and $\neg(\mathbf{B}\phi \wedge \mathbf{D}\phi)$ are not. The last two sentences play an important role in the sequel: $\mathbf{B}\phi \vee \mathbf{D}\phi$ expresses that ϕ is *belief complete* (ϕ is either believed or disbelieved) and $\neg(\mathbf{B}\phi \wedge \mathbf{D}\phi)$ expresses ϕ is *belief consistent* (ϕ is not believed and disbelieved simultaneously). Note that $\mathbf{B}\phi \vee \mathbf{D}\phi$ and $\neg(\mathbf{B}\phi \wedge \mathbf{D}\phi)$ are together equivalent to $\mathbf{D}\phi \leftrightarrow \neg\mathbf{B}\phi$. Moreover, the sentence $\mathbf{B}\phi \wedge \mathbf{D}\phi$ is propositionally consistent (the model $\mathcal{M} = \{\mathbf{B}\phi, \mathbf{D}\phi\}$ makes this sentence true) in our approach whereas the corresponding sentence $\mathbf{B}\phi \wedge \neg\mathbf{B}\phi$ is always false if the unimodal approach is adopted. This indicates that the modality $\mathbf{D}$ for disbeliefs permits our variant of autoepistemic to handle consistently situations where a sentence ϕ is believed and disbelieved simultaneously (for some reason or another).

The definitions of the closures $\mathrm{Cn}(\Sigma)$ and $\mathrm{Cn}_{\mathbf{B}}(\Sigma)$ remain intact, but the definitions of stable and iterative expansions have to be adjusted to take the modality $\mathbf{D}$ for disbelief into account. The definition below replaces standard negative introspection $\neg\mathbf{B}\overline{\Delta}$ with $\mathbf{D}\overline{\Delta} = \{\mathbf{D}\phi \,|\, \phi \in \mathcal{L}_{\mathrm{BD}} - \Delta\}$. We have also taken the set of premises $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ as a background theory and incorporated an additional set of assumptions $T \subseteq \mathcal{L}_{\mathrm{BD}}$ in the definition. Expansions are defined relative to T in order to obtain extra flexibility[9] for the binary inference operator to be extracted from this definition. Note that the expansions of Σ are also easily obtained by setting $T = \emptyset$.

Definition 2.31 *Let $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ be an autoepistemic theory. A set $\Delta \subseteq \mathcal{L}_{\mathrm{BD}}$ is a stable expansion (resp. iterative expansion) of $T \subseteq \mathcal{L}_{\mathrm{BD}}$ if and only if $\Delta = \mathrm{Cn}(\Sigma \cup T \cup \mathbf{B}\Delta \cup \mathbf{D}\overline{\Delta})$ holds (resp. $\Delta = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}\overline{\Delta})$ holds).*

Note that stable and iterative expansions Δ conforming to Definition 2.31 are necessarily belief complete, i.e. contain $\mathbf{B}\phi \vee \mathbf{D}\phi$ for every sentence $\phi \in \mathcal{L}_{\mathrm{BD}}$. On the other hand, it is quite possible that for some sentence $\phi \in \mathcal{L}_{\mathrm{BD}}$, the sentence $\mathbf{B}\phi \wedge \mathbf{D}\phi$ belongs to an expansion Δ. If this is the case, we call the expansion Δ *belief inconsistent*. Otherwise (i.e. there is no such ϕ), the expansion Δ is considered to be a belief consistent one. The following example exhibits a belief inconsistent stable expansion.

Example 2.32 Consider an autoepistemic theory $\Sigma = \{\mathbf{D}\mathrm{r} \to \mathrm{q}, \mathbf{B}\mathrm{q} \to \mathrm{p}, \mathbf{D}\mathrm{p} \to \mathrm{s}\}$. The theory $T = \{\mathbf{D}\mathrm{p}\}$ has a unique, belief inconsistent stable expansion $\Delta = \{\mathbf{D}\mathrm{p}, \mathrm{s}, \mathbf{B}\mathrm{s}, \mathbf{D}\mathrm{r}, \mathrm{q}, \mathbf{B}\mathrm{q}, \mathrm{p}, \mathbf{B}\mathrm{p}, \ldots\}$ containing $\mathbf{B}\mathrm{p} \wedge \mathbf{D}\mathrm{p}$. □

The expansion Δ above is strange-looking from the point of view of conventional autoepistemic logic (Δ becomes propositionally inconsistent if $\mathbf{D}$ is interpreted as $\neg\mathbf{B}$). However, we find that this expansion is still acceptable in the sense that it gives a reasonable set of beliefs for an agent performing full introspection under the assumption that the agent does not believe p. Indeed, stable expansions Δ are still closed under Stalnaker's conditions [104] for all sentences $\phi \in \mathcal{L}_{\mathrm{BD}}$: (i) $\Delta \models \phi$ implies $\phi \in \Delta$, (ii) $\phi \in \Delta$ implies $\mathbf{B}\phi \in \Delta$ and (iii) $\phi \notin \Delta$ implies $\mathbf{D}\phi \in \Delta$ with the last condition obviously modified.

[9] This is needed when we address the interconnection of default and autoepistemic logics.

A possibility is to use belief consistency as form of integrity constraint. In Example 2.32, we might interpret the belief inconsistency of Δ as an indication that the integrity constraint $\mathbf{D}\mathrm{p}$ given in T (p is not believed) is violated. Despite the expansion Δ is belief inconsistent, it provides useful information *why* the integrity constraint is violated, i.e. it gives the beliefs that realize p (the atom r is not believed in our example). Such analysis cannot be made directly in the conventional approach because of the resulting propositional inconsistency. Note also that there are interesting subclasses of autoepistemic theories (such as those considered in Lemma 3.30) that have only belief consistent stable and iterative expansions. Moreover, it is also possible to *enforce* belief consistency by introducing sufficient instances of the schema $\neg(\mathbf{B}\phi\wedge\mathbf{D}\phi)$ in the premises. However, the instances of this schema are not included in the underlying monotonic logic directly, because this would ruin the notion of stationary semantics which will be introduced in Section 4. The belief consistency matter will be further discussed therein.

Given an iterative expansion $\Delta \subseteq \mathcal{L}_{\mathrm{BD}}$ of a theory $T \subseteq \mathcal{L}_{\mathrm{BD}}$, the theory Δ is directly a set of theorems associated with T while $N = \overline{\Delta} = \mathcal{L}_{\mathrm{BD}} - \Delta$ gives the corresponding set of non-theorems in the spirit of stable semantics. This clarifies the interconnection of separated autoepistemic theories to our general framework: non-theorems correspond naturally to disbeliefs. In Definition 2.33, we introduce a binary inference operator $\mathcal{A}_\Sigma$ for each autoepistemic theory $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$. This is indeed a non-monotonic system as shown in Theorem 2.34. In the rest of this work, we identify autoepistemic logic with the family of non-monotonic systems $\mathcal{A}_\Sigma$ induced by autoepistemic theories $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$. We call such systems *autoepistemic* in the sequel.

Definition 2.33 *For a theory $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$, define a binary inference operator $\mathcal{A}_\Sigma$ by setting $\mathcal{L}(\mathcal{A}_\Sigma) = \langle \mathcal{L}_{\mathrm{BD}}, \mathcal{L}_{\mathrm{BD}} \rangle$ and for all $T \subseteq \mathcal{L}_{\mathrm{BD}}$ and $N \subseteq \mathcal{L}_{\mathrm{BD}}$,*

$$\mathcal{A}_\Sigma(T, N) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N).$$

Theorem 2.34 *For an autoepistemic theory Σ, the binary inference operator $\mathcal{A}_\Sigma$ is a non-monotonic system.*

Proof. For a fixed N, the operator $\lambda T.\mathcal{A}_\Sigma(T, N)$ is a closure operator by definition. Then let $N_1 \subseteq N_2$. By Proposition 2.29, the theory $\mathrm{Cn}_{\mathbf{B}}(\Sigma\cup T\cup\mathbf{D}N_1)$ is a subset of $\mathrm{Cn}_{\mathbf{B}}(\Sigma\cup T\cup\mathbf{D}N_2)$ for a fixed T. Hence $\lambda N.\mathcal{A}_\Sigma(T, N)$ is monotonic. Finally, let $T \subseteq \mathcal{L}_{\mathrm{BD}}$ and $N \subseteq \mathcal{L}_{\mathrm{BD}}$ for a compactness consideration. It should be established that $\mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$ coincides with the union of $\mathrm{Cn}_{\mathbf{B}}(\Sigma\cup T'\cup\mathbf{D}N')$ for all $T' \subseteq_{\mathrm{f}} T$ and $N' \subseteq_{\mathrm{f}} N$. ($\subseteq$) Let $\phi \in \mathrm{Cn}_{\mathbf{B}}(\Sigma\cup T\cup\mathbf{D}N)$. By Theorem 2.30, there is a finite subset $\Sigma' \subseteq_{\mathrm{f}} \Sigma \cup T \cup \mathbf{D}N$ such that $\phi \in \mathrm{Cn}_{\mathbf{B}}(\Sigma')$. Let $T' = T \cap \Sigma'$ and $N' = \{\phi \mid \mathbf{D}\phi \in \mathbf{D}N \cap \Sigma'\}$ so that $\mathbf{D}N' = \mathbf{D}N \cap \Sigma'$. It follows that $\phi \in \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T' \cup \mathbf{D}N')$ by Proposition 2.29. Thus ϕ belongs to the union. ($\supseteq$) This follows by Proposition 2.29. $\square$

Relationship to Conventional Autoepistemic Logic

Finally, we relate conventional autoepistemic theories with a single modality **B** to our specification of non-monotonic systems. Let us first consider Marek and Truszczyński's variant of autoepistemic logic [66] based on iterative expansions. In analogy to Definition 2.33, the required binary inference operator is $\mathbb{S}(T,N) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \neg\mathbf{B}N)$. The proof of Theorem 2.34 is easily adjusted for the unimodal case to show that this operator meets our specification. It is worth mentioning that by introducing the schema $\mathbf{D}\phi \leftrightarrow \neg\mathbf{B}\phi$ it is possible to capture iterative expansions of conventional autoepistemic theory using bimodal iterative expansions of Definition 2.31. The advantages of our bimodal approach will become clear when the semantics of non-monotonic systems is addressed in Section 4. The two approaches behave very similarly under stable semantics that we have already introduced, but fundamental differences arise when stationary semantics is considered, for instance.

Moore's autoepistemic logic, i.e. conventional autoepistemic theories $\Sigma \subseteq \mathcal{L}_{\mathrm{B}}$ under stable expansions, is also covered by our bimodal approach. This is because propositionally consistent stable expansions of Σ can be captured in terms of Marek and Truszczyński's iterative expansions as shown by the author [46]. This requires augmenting the set of premises Σ with a sentence $\neg\mathbf{B}\neg\mathbf{B}\phi \rightarrow \mathbf{B}\phi$ for each belief atom $\mathbf{B}\phi$ that appears in the premises Σ. Note that the sentences of the form $\neg\mathbf{B}\neg\mathbf{B}\phi \rightarrow \mathbf{B}\phi$ are instances of the standard modal axiom schema **5** and they serve as a replacement of the positive introspection $\mathbf{B}\Delta$ of stable expansions. Marek, Schwarz and Truszczyński have devised a similar characterization of stable expansions, but they use *all* instances of the schema **5**. The schema **5** has a very intuitive reading in the context of beliefs: the belief in ϕ can be established if it is disbelieved that ϕ is disbelieved. This is a kind of a law of "double disbelief" in analogy to the law of double negation $\neg\neg\phi \rightarrow \phi$ in classical logic.

2.3.4 On the Scope of the Framework

Our analysis in Sections 2.3.1 - 2.3.3 has covered three leading formalizations of non-monotonic reasoning. We argue that there are also other interesting formalizations for which non-monotonic systems conforming to Definition 2.6 can be extracted. Many such systems can be obtained as variants of the three systems considered so far. For instance, one possibility is to enhance the underlying monotonic reasoning by adding new syntactic constructs or inference mechanisms. In the following, we discuss some of the possibilities for these kinds of extensions of the formalisms considered. In many cases, the variants seem to remain in the scope of our specification, because the closure axioms REF, CLO and MON are still reasonable requirements for such reasoning. On the other hand, monotonicity does not guarantee compactness and thus COM has to be considered for each system separately.

A way of generalizing circumscription is to allow a set of monotonic inference rules R to be used. The fixed point definition of P-extensions should be rewritten as $E = \mathrm{Cn}^R(T \cup \mathrm{Neg}_P(\overline{E}))$. There is no doubt that the corresponding inference operator $\mathfrak{C}_{P,R}$ defined by $\mathfrak{C}_{P,R}(T, N) = \mathrm{Cn}^R(T \cup \mathrm{Neg}_P(N))$ is a non-monotonic system. Default logic can be enriched by taking inference rules into account disjunctively. To achieve this the closure condition of Definition 2.2 has to be modified as follows: if $\frac{\alpha_1}{\gamma_1}, ..., \frac{\alpha_n}{\gamma_n}$ are inference rules or R and $\alpha_1 \vee ... \vee \alpha_n \in \mathrm{Cn}^R(T)$, then also $\gamma_1 \vee ... \vee \gamma_n \in \mathrm{Cn}^R(T)$. For instance, we could infer c from a disjunction $\mathrm{a} \vee \mathrm{b}$ using the inference rules $\frac{\mathrm{a}}{\mathrm{c}}$ and $\frac{\mathrm{b}}{\mathrm{c}}$.[10] We expect that the inference operator $\mathfrak{D}_D$ is still a non-monotonic system although the monotonic inference rules resulting from the reduction D_N are interpreted in this way. In particular, the compactness is again implied by the compactness of propositional logic. Autoepistemic logic can be enhanced by adding modal axiom schemas. In fact, this has already been done, since such logics have been proposed by McDermott [69] and studied extensively by Marek, Schwarz and Truszczyński [64]. The idea is to use some monotonic modal logic S with the modality $\mathbf{B}$ for monotonic inference instead of the modal logic of necessitation [29]. This idea leads to an inference operator $\boldsymbol{\mathcal{A}}_\Sigma$ defined by $\boldsymbol{\mathcal{A}}_\Sigma(T, N) = \mathrm{Cn}_S(\Sigma \cup T \cup \neg\mathbf{B}N)$. This is certainly a non-monotonic system – as long as the underlying monotonic modal logic S is compact.

Recall also that we made no assumption on the syntax of the sentences and this gives additional degree of freedom. For instance, we may take inference rules or similar objects in the role of sentences. To give an example where this approach might be useful, we take Lehmann and Magidor's *rational closure* [58] and Pearl's *system Z* [87] into consideration. These are closely related, since they both handle assertions of the form $\alpha \mid\!\sim \beta$ where α and β are propositional sentences. Moreover, the notion of 1-entailment in system Z coincides with rational closure [87] and therefore we consider only rational closure in further detail. An assertion $\alpha \mid\!\sim \beta$ states that β is a plausible conclusion from α and $\mid\!\sim$ can be considered as a non-monotonic consequence relation, for instance. Lehmann and Magidor consider the closure of a set of assertions K (i.e. knowledge base) under rules that are inference rules for assertions. Five of the rules are given in Figure 4 on page 79. In addition to these, Lehmann and Magidor introduce a rule called *or* "if $\alpha \mid\!\sim \gamma$ and $\beta \mid\!\sim \gamma$, then $\alpha \vee \beta \mid\!\sim \gamma$" to define the *preferential closure* of K. The corresponding operator is a monotonic system in our sense [58]. Also, Lehmann and Magidor introduce the rule of *rational monotonicity* "if $\alpha \mid\!\sim \gamma$ and $\alpha \not\mid\!\sim \neg\beta$, then $\alpha \wedge \beta \mid\!\sim \gamma$". This rule has a non-monotonic character due to condition $\alpha \not\mid\!\sim \neg\beta$ and therefore we define

[10] Default logic has been criticized for its incapability of this kind of case analysis [17].

a set of rules R_N containing "if $\alpha \mathrel{|\!\sim} \gamma$, then $\alpha \wedge \beta \mathrel{|\!\sim} \gamma$" whenever $\alpha \mathrel{|\!\sim} \neg\beta$ in a set of assertions N that serves as our set of non-theorems. Then we define a non-monotonic system $\mathfrak{S}$ associated with a set of assertions K by taking the closure of $K \cup T$ under the rules of preferential closure and R_N [11] for all sets of assertions T and N. Under stable semantics (i.e. when N contains assertions that are not in $\mathfrak{S}(T, N)$) with the set of assertions $T = \emptyset$, this system captures sets of assertions K' that extend K and satisfy the rules of preferential closure and rational monotonicity. Such K' are called *rational consequence relations* [58]. Moreover, Lehmann and Magidor single out a specific rational extension of K – the *rational closure* of K – using a preference ordering. It remains open whether this ordering can be captured in terms of a non-monotonic system.

Let us then discuss the possibilities of altering the interpretation of non-theorems. We will consider default logic in this respect. For instance Delgrande et al. propose an alternative interpretation for the justifications of default rules.[12] Their principle is the *joint consistency* of the justification β and the consequent γ, i.e. it must be consistent to assume $\beta \wedge \gamma$ in order to apply a default rule $\frac{\alpha:\beta}{\gamma}$. This leads to a modified reduction $D_N = \{\frac{\alpha}{\gamma} \mid \frac{\alpha:\beta}{\gamma} \in D \text{ and } \neg(\beta \wedge \gamma) \in N\}$. It is obvious that this change does not affect the status of the inference operator $\mathfrak{D}_D(T, N) = \mathrm{Cn}^{D_N}(T)$ as a non-monotonic system. However, the approach of Delgrande et al. has more structure, since they introduce an additional set of constraints. The definition of a *constrained extension* of T can be reformulated as a pair $\langle E, C \rangle$ where $E \subseteq C \subseteq \mathcal{L}$, $E = \mathrm{Cn}^{D_{\overline{C}}}(T)$, $C = \mathrm{Cn}^{D'_{\overline{C}}}(T)$ and $D'_N = \{\frac{\alpha}{\beta \wedge \gamma} \mid \frac{\alpha:\beta}{\gamma} \in D \text{ and } \neg(\beta \wedge \gamma) \in N\}$ is another reduction. Since effectively two closures are used, it seems difficult to capture this approach using a single inference operator whereas a pair of inference operators would be certainly enough. In this sense, the approach proposed by Delgrande et al. goes beyond our current framework.

We should also bear in mind the possibility of transforming a system of one kind into a system of other kind (such mappings are considered in Section 3.3, for instance). To illustrate this possibility, let us consider the decision problem of default logic which is $\mathbf{\Pi}_2^{\mathrm{p}}$-complete by Gottlob's results [37] (see Section 7.1 for details of the *polynomial time hierarchy* [2]). Then consider any logic L whose decision problem is in the class $\mathbf{\Pi}_2^{\mathrm{p}}$. Now any instance of the decision problem of L can be transformed to an instance of the decision problem of default logic (in polynomial time). Then it is possible to associate the resulting set of defaults D and the system $\mathfrak{D}_D$ with the logic L such that a non-monotonic system is extracted indirectly. This can be especially useful, if the set of defaults D introduced in the polynomial transformation remains invariant for all instances of the decision problem of the logic L.

[11] It seems that the additional rules in R_N do not affect compactness.
[12] Only defaults of the form $\frac{\alpha:\beta}{\gamma}$ are considered in this approach.

Finally, let us discuss reasons for which some systems might **not** be covered by the current notion of a non-monotonic system. (i) There is no way to extract a reasonable inference operator that would satisfy the axioms REF, CLO and MON. For instance, if we impose *time bounds* on classical monotonic reasoning, then closure axiom CLO is no longer met. (ii) The axiom COM is violated, i.e. the underlying logic is not compact. Compactness is easily lost, e.g., if we introduce infinite sentences such as $\mathrm{p}_1 \wedge \mathrm{p}_2 \wedge \ldots \wedge \mathrm{p}_i \wedge \ldots$ or modal axioms such as the axiom $\Box(\Box\phi \rightarrow \phi) \rightarrow \Box\phi$ in the logic of provability [8, 28]. However, it seems that compactness can be replaced by a more general property to meet the needs of such logics. (iii) Our assumption that the language $\mathcal{L}(\mathcal{S})$ associated with a system $\mathcal{S}$ is countable (i.e. denumerable) is not met. For example, systems with uncountable languages are not covered by our current definition. (iv) Multiple inference operators are needed. As we noted above, two inference operators would be needed to capture the variant of default logic by Delgrande et al. [17]. Also, a multi-agent versions of autoepistemic logic have been proposed (see, e.g., the approach by Lakemeyer [57]) where n operators might very well arise. It seems that these kinds of systems can be approached using a generalization of our framework. For instance, we could join two monotonic inference operators $\mathcal{S}_1$ and $\mathcal{S}_2$ to map a pair $\langle T_1, T_2 \rangle \in \mathbf{2}^{\mathcal{L}(\mathcal{S}_1)} \times \mathbf{2}^{\mathcal{L}(\mathcal{S}_2)}$ to a pair $\langle \mathcal{S}_1(T_1), \mathcal{S}_2(T_2) \rangle$. Then, e.g., the requirement REF would be modified to $\langle T_1, T_2 \rangle \sqsubseteq \mathcal{S}(\langle T_1, T_2 \rangle)$ where $\mathcal{S} = \langle \mathcal{S}_1, \mathcal{S}_2 \rangle$ and $\sqsubseteq$ denotes pairwise inclusion. (v) Several types of parameters (non-theorems) might be needed in the system. It seems that it is possible to devise systems where theorems depend on two independent sources of non-monotonicity. Such systems could be obtained as hybrid systems of non-monotonic systems conforming to our specification. Another issue is whether several types of parameters are really necessary, i.e. is it possible to merge parameters somehow into one type of parameters (consider pairing, for instance).

2.4 Comparison and Examples

So far our analysis has covered three leading formalizations of non-monotonic reasoning: circumscription, default logic and autoepistemic logic. Table 3 provides a collection of the associated non-monotonic systems: minimal model systems $\mathfrak{C}_P$, default systems $\mathfrak{D}_D$ and autoepistemic systems $\mathcal{A}_\Sigma$. Recall also the set of parameters involved: $P \subseteq \mathcal{A}$ is a set of atoms subject to minimization, D is a set of default rules in $\mathcal{L}$ and $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ is an autoepistemic theory. Let us yet emphasize that the role of non-theorems in these systems is uniform at the level of our specification, despite non-theorems have quite different interpretations in the three formalizations of non-monotonic reasoning. Let us now represent Example 2.5 in terms of the non-monotonic logics introduced so far. We shall consider these formalisms under stable semantics by finding

$\mathcal{S}$	$\mathcal{L}(\mathcal{S})$	$\mathcal{S}(T, N)$
$\mathfrak{C}_P$	$\langle \mathcal{L}, \mathcal{L} \rangle$	$\mathrm{Cn}(T \cup \mathrm{Neg}_P(N))$
$\mathfrak{D}_D$	$\langle \mathcal{L}, \mathcal{L} \rangle$	$\mathrm{Cn}^{D_N}(T)$
$\mathcal{A}_\Sigma$	$\langle \mathcal{L}_{\mathrm{BD}}, \mathcal{L}_{\mathrm{BD}} \rangle$	$\mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$

Table 3: Examples of non-monotonic systems

out the minimal models of a minimal model theory, the extensions of a default theory and the iterative expansions of an autoepistemic theory corresponding to our diagnosis example, but limited to rules concerning a single and-gate `G`. We introduce atomic propositions in1(0), in1(1), in2(0), in2(1), out(0), out(1) to describe the inputs and outputs of `G` and an atom b which denotes that `G` is broken. The following three examples demonstrate that each of the three logics is well-suited for formalizing diagnostic reasoning. In each case the default conclusion out(0) is correctly made from the given premise in1(0). On the other hand, this conclusion is withdrawn by the introduction of an additional premise out(1). We can also conclude that the gate `G` is broken.

Example 2.35 For minimal model reasoning we use the propositional theory

$$
\begin{aligned}
T = \{ \quad & \mathrm{in1}(0) \wedge \neg \mathrm{b} \rightarrow \mathrm{out}(0), \\
& \mathrm{in2}(0) \wedge \neg \mathrm{b} \rightarrow \mathrm{out}(0), \\
& \mathrm{in1}(1) \wedge \mathrm{in2}(1) \wedge \neg \mathrm{b} \rightarrow \mathrm{out}(1), \\
& (\mathrm{in1}(0) \vee \mathrm{in2}(0)) \wedge \mathrm{out}(1) \rightarrow \mathrm{b}, \\
& \mathrm{in1}(1) \wedge \mathrm{in2}(1) \wedge \mathrm{out}(0) \rightarrow \mathrm{b} \qquad \}.
\end{aligned}
$$

The atom b is false in all $\{\mathrm{b}\}$-minimal models of $T_1 = T \cup \{\mathrm{in1}(0)\}$. Thus we obtain the desired conclusion out(0). Then consider $T_2 = T \cup \{\mathrm{in1}(0), \mathrm{out}(1)\}$. Now b is a logical consequence of T_2 which implies that the atom b is necessarily true in all $\{\mathrm{b}\}$-minimal models of T_2. This is how we lose the conclusion out(0) and obtain the conclusion out(1) instead. □

Example 2.36 Let T_1 and T_2 be defined as in Example 2.35 and D the set of default rules $\{\frac{:\neg \mathrm{b}}{\neg \mathrm{b}}\}$, i.e. the gate is not broken by default. The default theory $\langle D, T_1 \rangle$ has only one extension $E_1 = \mathrm{Cn}(T_1 \cup \{\neg \mathrm{b}\})$ which includes the desired conclusion out(0). To verify this we note that the set of inference rules $R = \{\frac{\top}{\neg \mathrm{b}}\}$ is obtained as the reduction $D_{\overline{E_1}}$ so that $E_1 = \mathrm{Cn}^R(T_1) = \mathrm{Cn}(T_1 \cup \{\neg \mathrm{b}\})$ holds. The default theory $\langle D, T_2 \rangle$ has only one extension $E_2 = \mathrm{Cn}(T_2)$, since the reduction $D_{\overline{E_2}} = \emptyset$. So the conclusion out(0) is lost while b and out(1) can be concluded from T_2 directly. □

Example 2.37 For autoepistemic reasoning, we introduce the theory

$$\begin{aligned}\Sigma = \{ \quad & \mathrm{in1}(0) \wedge \mathbf{D}\mathrm{b} \rightarrow \mathrm{out}(0), \\ & \mathrm{in2}(0) \wedge \mathbf{D}\mathrm{b} \rightarrow \mathrm{out}(0), \\ & \mathrm{in1}(1) \wedge \mathrm{in2}(1) \wedge \mathbf{D}\mathrm{b} \rightarrow \mathrm{out}(1), \\ & (\mathrm{in1}(0) \vee \mathrm{in2}(0)) \wedge \mathrm{out}(1) \rightarrow \mathrm{b}, \\ & \mathrm{in1}(1) \wedge \mathrm{in2}(1) \wedge \mathrm{out}(0) \rightarrow \mathrm{b} \qquad \}.\end{aligned}$$

The set $\Delta_1 = \{\mathrm{in1}(0), \mathbf{D}\mathrm{b}, \mathrm{out}(0), \mathbf{B}\mathrm{out}(0), \ldots\}$ is the only iterative expansion $T_1 = \{\mathrm{in1}(0)\}$. For $T_2 = \{\mathrm{in1}(0), \mathrm{out}(1)\}$, we obtain the iterative expansion $\Delta_2 = \{\mathrm{in1}(0), \mathrm{out}(1), \mathbf{B}\mathrm{b}, \mathbf{B}\mathrm{out}(1), \ldots\}$. By these two expansions, the conclusion out(0) is retracted as expected and out(1) is concluded. □

The preceding examples showed that circumscription, default logic and autoepistemic logic are capable of very similar reasoning tasks. This is mainly because these logics are uniformly built upon propositional logic. However, we can differentiate circumscription from default and autoepistemic logic in at least one respect, i.e. what comes to reasoning with monotonic inference rules of the form $\frac{\alpha}{\gamma}$. It is clear that such rules are available in default logic as default rules of a restricted form $\frac{\alpha:}{\gamma}$. Such inferences are also easily performed with autoepistemic logic when the necessitation rule is present. More precisely, the following proposition can be established [47]. The proof takes place by induction of the lengths of R-proofs and $\mathbf{B}$-proofs and it can be extracted from the proof of Proposition 3.31 to be presented later.

Proposition 2.38 *Let $T \subseteq \mathcal{L}$ be a propositional theory and R a set of inference rules in $\mathcal{L}$. Let Σ be the translation $\{\mathbf{B}\alpha \rightarrow \gamma \mid \frac{\alpha}{\gamma} \in R\}$ of R into an autoepistemic theory. Then* $\mathrm{Cn}^R(T) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T) \cap \mathcal{L}$.

In contrast to default logic and autoepistemic logic, there is no easy way of performing such inferences when only P-minimal models of a propositional theory are considered. A straightforward idea would be to translate an inference rule $\frac{\alpha}{\gamma}$ into an implication $\alpha \rightarrow \gamma$. However, the latter involves always the contrapositive $\neg\gamma \rightarrow \neg\alpha$ while the former provides no means of inferring $\neg\alpha$ given $\neg\gamma$. Imielinski [42] and Niemelä [79] provide a precise mathematical argument why it is difficult to capture default reasoning and autoepistemic reasoning, respectively, with minimal model theories. Let us present Niemelä's argument for autoepistemic logic, but restricted to iterative expansions and P-minimal models considered in this work. Niemelä considers translating a set of autoepistemic sentences Σ into a set of propositional sentences $\mathrm{Tr}(\Sigma)$ and a set of atoms P to be minimized. Niemelä's results imply that there is no *modular* way to do such a translation, i.e. cannot hold for every $T \subseteq \mathcal{L}$ and $\phi \in \mathcal{L}$ that ϕ belongs to all iterative expansions of $\Sigma \cup T$ if and only if ϕ belongs to all P-minimal models of $\mathrm{Tr}(\Sigma) \cup T$. On the other hand, the complexity results by Eiter and Gottlob [25, 37] indicate that a polynomial translation can be made for particular T and ϕ, but $\mathrm{Tr}(\Sigma)$ is dependent on T and ϕ and has to be recomputed if T and ϕ change. It is obvious that Niemelä's argument applies also to the variant of autoepistemic logic proposed in this work, since Niemelä uses $\Sigma = \{\mathbf{B}\mathrm{a} \rightarrow \mathrm{b}\}$ as a counter-example.[13]

Let us further demonstrate the importance of inference rules with Example 2.37. Suppose we would like to incorporate information on a misbehaving gate, e.g., the output of a faulted and-gate is always 1. It would be natural to express this as an inference rule $\frac{\mathrm{b}}{\mathrm{out}(1)}$, or as an autoepistemic sentence $\mathbf{B}\mathrm{b} \rightarrow \mathrm{out}(1)$ in the spirit of Proposition 2.38. The uniqueness of outputs can be represented with the sentences $\neg(\mathrm{out}(0) \wedge \mathrm{out}(1))$ and $\mathrm{out}(0) \vee \mathrm{out}(1)$. Define Σ' as the set of autoepistemic sentences Σ augmented by these three sentences. Then we obtain the same iterative expansion Δ_2 for $T_2 = \{\mathrm{in1}(0), \mathrm{out}(1)\}$ as before. Note also that $T_3 = \{\mathrm{in1}(1), \mathrm{in2}(1), \mathrm{out}(0)\}$ yields an iterative expansion $\Delta = \mathrm{Cn}_{\mathbf{B}}(\Sigma' \cup T_3) = \mathcal{L}_{\mathrm{BD}}$. For the sake of comparison, we return to Example 2.35. If we represent the above inference rule as an implication $\mathrm{b} \rightarrow \mathrm{out}(1)$, the contrapositive $\neg\mathrm{out}(1) \rightarrow \neg\mathrm{b}$ can be applied to infer $\neg\mathrm{b}$ if we had specified a measured output $\mathrm{out}(0)$ and the uniqueness of outputs. This indicates a great difficulty in representing *fault models* [107] if minimal models are used. By the above arguments and examples, this is due to the incapability of circumscription to capture reasoning with inference rules.

2.5 Fixing Sentences as Theorems and Non-Theorems

To end Section 2, we introduce a technique to derive new systems from a non-monotonic system $\mathcal{S}$ by *fixing* a set of sentences $F^+ \subseteq \mathcal{L}^+(\mathcal{S})$ as theorems and a set of sentences $F^- \subseteq \mathcal{L}^-(\mathcal{S})$ as non-theorems. We let $\mathcal{S}[F^+; F^-]$ denote a non-monotonic system having the sets of sentences F^+ and F^- *positively* and *negatively* fixed, respectively. Note that we may drop F^+ or F^- in case they are empty. In this case, the separating semicolon is also omitted and the type of fixing (positive or negative) is clear from the context. Basically, the non-monotonic system $\mathcal{S}[F^+; F^-]$ operates on the input $\langle T, N\rangle$ in the way $\mathcal{S}$ operates on $\langle T, N\rangle$ with T and N augmented with the fixed theorems and non-theorems. The idea is formalized as follows.

Definition 2.39 *For a non-monotonic system $\mathcal{S}$, a set of theorems $F^+ \subseteq \mathcal{L}^+(\mathcal{S})$ and a set of non-theorems $F^- \subseteq \mathcal{L}^-(\mathcal{S})$, define a binary inference operator $\mathcal{S}[F^+; F^-]$ by setting $\mathcal{L}(\mathcal{S}[F^+; F^-]) = \mathcal{L}(\mathcal{S})$ and for all $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $N \subseteq \mathcal{L}^-(\mathcal{S})$, the set of theorems $\mathcal{S}[F^+; F^-](T, N) = \mathcal{S}(F^+ \cup T, F^- \cup N)$.*

[13] A related discussion goes on in conjunction with Proposition 3.28.

It is verified in Proposition 2.40 that $\mathcal{S}[F^+; F^-]$ satisfies our specification as a self-standing non-monotonic system whose language is $\mathcal{L}(\mathcal{S}[F^+; F^-]) = \mathcal{L}(\mathcal{S})$. In this setting, the non-monotonic systems $\mathcal{S}[\emptyset; \emptyset]$ and $\mathcal{S}$ coincide. By this result, a property established for a non-monotonic system $\mathcal{S}$ extends for an arbitrary $\mathcal{S}[F^+; F^-]$ directly. The fixing construction is especially useful when we (i) define various semantics in Section 4, (ii) address the cumulativity property in Section 5.1 and (iii) propose a new semantics in Section 8.

Proposition 2.40 *For a non-monotonic system $\mathcal{S}$, a set of theorems $F^+ \subseteq \mathcal{L}^+(\mathcal{S})$ and a set of non-theorems $F^- \subseteq \mathcal{L}^-(\mathcal{S})$, the binary inference operator $\mathcal{S}[F^+; F^-]$ is a non-monotonic system.*

Proof. (REF) Let $T \subseteq \mathcal{L}^+(\mathcal{S}[F^+; F^-]) = \mathcal{L}^+(\mathcal{S})$ and $N \subseteq \mathcal{L}^-(\mathcal{S}[F^+; F^-]) = \mathcal{L}^-(\mathcal{S})$. Now $F^+ \cup T$ is a subset of $\mathcal{S}(F^+ \cup T, F^- \cup N)$ by the requirement REF. It follows that $T \subseteq \mathcal{S}(F^+ \cup T, F^- \cup N) = \mathcal{S}[F^+; F^-](T, N)$.

(CLO) Let T' be the set of theorems $\mathcal{S}[F^+; F^-](T, N) = \mathcal{S}(F^+ \cup T, F^- \cup N)$. Now F^+ is a subset of T' by the requirement REF so that $\mathcal{S}[F^+; F^-](T', N) = \mathcal{S}(F^+ \cup T', F^- \cup N) = \mathcal{S}(T', F^- \cup N)$. By the requirement CLO, we obtain $\mathcal{S}(T', F^- \cup N) = \mathcal{S}(F^+ \cup T, F^- \cup N) = T'$. Thus $\mathcal{S}[F^+; F^-](T', N) = T'$.

(MON) Consider sets of theorems $T \subseteq T' \subseteq \mathcal{L}^+(\mathcal{S}[F^+; F^-]) = \mathcal{L}^+(\mathcal{S})$ and sets of non-theorems $N \subseteq N' \subseteq \mathcal{L}^-(\mathcal{S}[F^+; F^-]) = \mathcal{L}^-(\mathcal{S})$. By the requirement MON, the theory $\mathcal{S}(F^+ \cup T, F^- \cup N)$ is a subset of $\mathcal{S}(F^+ \cup T', F^- \cup N')$. Thus $\mathcal{S}[F^+; F^-](T, N) \subseteq \mathcal{S}[F^+; F^-](T', N')$.

(COM) Let T and N be any subsets of the languages $\mathcal{L}^+(\mathcal{S}[F^+; F^-]) = \mathcal{L}^+(\mathcal{S})$ and $\mathcal{L}^-(\mathcal{S}[F^+; F^-]) = \mathcal{L}^-(\mathcal{S})$, respectively. Now the set of theorems $\mathcal{S}[F^+; F^-](T, N) = \mathcal{S}(F^+ \cup T, F^- \cup N)$ equals by the requirement COM to $S = \bigcup\{\mathcal{S}(T', N') \mid T' \subseteq_{\mathrm{f}} F^+ \cup T \text{ and } N' \subseteq_{\mathrm{f}} F^- \cup N\}$. This theory is a subset of $S' = \bigcup\{\mathcal{S}[F^+; F^-](T', N') \mid T' \subseteq_{\mathrm{f}} T \text{ and } N' \subseteq_{\mathrm{f}} N\}$. To see this, note that $\phi \in S$ implies $\phi \in \mathcal{S}(T', N')$ for some $T' \subseteq_{\mathrm{f}} F^+ \cup T$ and $N' \subseteq_{\mathrm{f}} F^- \cup N$. It follows by MON that $\phi \in \mathcal{S}(F^+ \cup T'', F^- \cup N'')$ for $T'' = T' \cap T$ and $N'' = N' \cap T$. Thus $\phi \in \mathcal{S}[F^+; F^-](T'', N'')$ for $T'' \subseteq_{\mathrm{f}} T$ and $N'' \subseteq_{\mathrm{f}} N$. This implies $\phi \in S'$. Hence, the theory $\mathcal{S}[F^+; F^-](T, N) \subseteq S'$. Inclusion in the other direction follows directly by the requirement MON. $\square$

It is also meaningful to perform fixing repeatedly. Assume that we want to fix sets of theorems F^+_α and sets of non-theorems F^-_α indexed by an ordinal α in a sequence. Then, a non-monotonic system $\mathcal{S}[F^+_0; F^-_0][F^+_1; F^-_1]...[F^+_\alpha; F^-_\alpha]$ is naturally defined as the non-monotonic system $\mathcal{S}[F^+; F^-]$ where $F^+ = \bigcup_{0 \leq \beta \leq \alpha} F^+_\beta$ and $F^- = \bigcup_{0 \leq \beta \leq \alpha} F^-_\beta$, i.e. we take simply the unions of the sets of theorems and non-theorems. Recall also that we introduced the shorthand $\mathcal{S}(N)$ for $\mathcal{S}(\emptyset, N)$. This is used, e.g., when a set of sentences $T \subseteq \mathcal{L}^+(\mathcal{S})$ is positively fixed. Then $\mathcal{S}[T](N) = \mathcal{S}(T, N)$ by our definitions.

3 Interconnections of Non-Monotonic Systems

So far we have studied how the leading formalizations of non-monotonic reasoning fit our specification of non-monotonic systems. The interconnections of these formalisms have also been extensively studied in terms of translations that are targeted to embed a non-monotonic theory in another such that sets of conclusions are preserved. For instance, Etherington [26] translates minimal model theories into default ones. Moreover, the study of the interconnection of default and autoepistemic logic has emerged a large number of translations of a default theory into an autoepistemic theory [39, 52, 53, 65, 112] and back [46, 67]. One of our goals is to describe translations and their properties within the framework of non-monotonic systems. For this purpose, we need to establish basic relations among non-monotonic systems as follows. First of all, we define when a non-monotonic system is a *subsystem* of another in Section 3.1. A notion of a *functional equivalence* for non-monotonic systems is established in 3.2. In Section 3.3, the notions of subsystems and functional equivalence are combined to describe what an *embedding* is at the level of non-monotonic systems. Finally, we consider a number of translations in Section 3.4 and show how translation functions between non-monotonic theories lead to natural embeddings between the associated non-monotonic systems.

3.1 Subsystems

The *subsystem* relation for non-monotonic systems is established in two phases: monotonic systems are considered at first and then a generalization to the non-monotonic case is made. After that subsystems of the systems $\mathfrak{C}_P$, $\mathfrak{D}_D$ and $\mathcal{A}_\Sigma$ are established in Sections 3.1.1, 3.1.2 and 3.1.3, respectively. Subsystems for minimal model systems and default systems are easy to devise, while much more effort is needed when autoepistemic systems are considered.

As suggested by the term subsystem already, we would like to identify systems $\mathcal{S}'$ related to a given system $\mathcal{S}$ that have only a partial functionality compared to that of $\mathcal{S}$. An obvious way of deriving such systems is to restrict $\mathcal{S}$ by a sublanguage of $\mathcal{S}$. To motivate our forthcoming definition of subsystems, we consider a monotonic system in Example 3.1.

Example 3.1 Consider a monotonic system $\mathcal{S}$ defined by $\mathcal{S}(T) = \mathrm{Cn}_{\mathbf{B}}(T)$ where T is a subset of the language $\mathcal{L}(\mathcal{S}) = \mathcal{L}_{\mathrm{B}}$. For instance, if $T = \{\mathrm{p}, \mathrm{p} \rightarrow \mathrm{q}\}$, then $\mathcal{S}(T) = \{\mathrm{p}, \mathrm{p} \rightarrow \mathrm{q}, \mathrm{q}, \mathbf{B}\mathrm{p}, \mathbf{B}\mathrm{q}, \ldots\}$ and the intersection $\mathcal{S}(T) \cap \mathcal{L}$ contains the desired propositional conclusion q among others. $\square$

The system $\mathcal{S}$ is a simplified autoepistemic system lacking negative introspection, and thus a monotonic system. This system is capable of propositional inference if we restrict T to the propositional sublanguage $\mathcal{L}$ of $\mathcal{L}_{\mathrm{BD}}$ and take $\mathrm{Cn}_{\mathbf{B}}(T) \cap \mathcal{L}$ as the resulting set of theorems. This observation on Example 3.1 leads to a notion of subsystems introduced in Definition 3.2.

Definition 3.2 *Let $\mathcal{S}$ be a monotonic system. A monotonic system $\mathcal{S}'$ is a subsystem of $\mathcal{S}$, denoted by $\mathcal{S}' \sqsubseteq \mathcal{S}$, if and only if $\mathcal{L}(\mathcal{S}')$ is a sublanguage of $\mathcal{L}(\mathcal{S})$ and it holds for all $T' \subseteq \mathcal{L}(\mathcal{S}')$ that*

$$\mathcal{S}'(T') = \mathcal{S}(T') \cap \mathcal{L}(\mathcal{S}').$$

Our definition gives emphasis on the functionality of $\mathcal{S}$ and $\mathcal{S}'$ as inference operators over a suitable language rather than exact structure of $\mathcal{S}_1$ and $\mathcal{S}_2$. In Example 3.3, we establish a subsystem for the system of Example 3.1. Note that we can also take Definition 3.2 as a definition of a subsystem $\mathcal{S}'$ for an arbitrary sublanguage of $\mathcal{S}$. In most cases, this sublanguage has to be carefully chosen in order to make the resulting subsystem $\mathcal{S}'$ interesting.

Example 3.3 Consider the monotonic system $\mathcal{S}$ of Example 3.1 which was defined by $\mathcal{S}(T) = \mathrm{Cn}_{\mathbf{B}}(T)$ for all $T \subseteq \mathcal{L}_{\mathrm{BD}}$. Let $\mathcal{L}$ be the propositional sublanguage of $\mathcal{L}_{\mathrm{BD}}$ and $\mathcal{S}'$ the propositional system defined by $\mathcal{S}'(T) = \mathrm{Cn}(T)$ for all $T \subseteq \mathcal{L}$. The systems $\mathcal{S}'$ is shown to be a subsystem of $\mathcal{S}$ as follows.

Let $T \subseteq \mathcal{L}$ and $\phi \in \mathcal{L}$. It is shown that $\phi \in \mathrm{Cn}(T)$ if and only if $\phi \in \mathrm{Cn}_{\mathbf{B}}(T)$. ($\Rightarrow$) Let $\phi \in \mathrm{Cn}(T)$, i.e. $T \models \phi$. Thus ϕ is $\mathbf{B}$-provable from T and $\phi \in \mathrm{Cn}_{\mathbf{B}}(T)$. ($\Leftarrow$) Let $\phi \in \mathrm{Cn}_{\mathbf{B}}(T)$, i.e. there is a $\mathbf{B}$-proof $\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n$ of ϕ from T such that $T \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\} \models \phi$. Since T and ϕ are purely propositional, it follows by Lemma 3.14 (to be presented later) that $T \models \phi$ so that $\phi \in \mathrm{Cn}(T)$. □

Definition 3.4 generalizes our notion of subsystems to the non-monotonic case. Now sets of non-theorems are taken into account, but a further constraint is introduced. Rather than simply requiring that $\mathcal{S}'(T', N')$ always equals to $\mathcal{S}(T', N') \cap \mathcal{L}^+(\mathcal{S}')$ in analogy to Definition 3.2, we insist that the same set of theorems is obtained as $\mathcal{S}(T', N) \cap \mathcal{L}^+(\mathcal{S}')$ for any set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ such that $N' = N \cap \mathcal{L}^-(\mathcal{S}')$. This is a further "projection property" that we require from non-monotonic subsystems[14].

Definition 3.4 *Let $\mathcal{S}$ be a non-monotonic system. A non-monotonic system $\mathcal{S}'$ is a subsystem of $\mathcal{S}$, denoted by $\mathcal{S}' \sqsubseteq \mathcal{S}$, if and only if $\mathcal{L}(\mathcal{S}') \sqsubseteq \mathcal{L}(\mathcal{S})$ and it holds for all $T' \subseteq \mathcal{L}^+(\mathcal{S}')$ and $N \subseteq \mathcal{L}^-(\mathcal{S})$ that*

$$\mathcal{S}'(T', N') = \mathcal{S}(T', N) \cap \mathcal{L}^+(\mathcal{S}')$$

where $N' = N \cap \mathcal{L}^-(\mathcal{S}')$.

[14] This property is essential, since our finitary characterizations of default extensions [81] and iterative expansions [37] will be based on subsystems.

In the sequel, it is shown that interesting subsystems conforming to Definition 3.4 exist for the non-monotonic systems introduced in Section 2.3. This is mainly because it is often the case with non-monotonic systems that only few non-theorems from $N \subseteq \mathcal{L}^-(\mathcal{S})$ contribute to a set of theorems $\mathcal{S}(T, N)$ after all. This is demonstrated by Example 3.5 where we introduce a subsystem for a default system. We let $\mathrm{J}(D)$ denote the set of sentences that appear as justifications $\beta_1, ..., \beta_n$ in the rules $\frac{\alpha:\beta_1,...,\beta_n}{\gamma}$ of a set of defaults D.

Example 3.5 Let $\mathfrak{D}_D$ be a default system induced by a set of defaults D in a propositional language $\mathcal{L}$. Recall that $\mathfrak{D}_D$ has the language $\langle \mathcal{L}, \mathcal{L} \rangle$. Define a system $\mathcal{S}'$ with a language $\langle \mathcal{L}, \neg\mathrm{J}(D) \rangle$ by setting $\mathcal{S}'(T', N') = \mathrm{Cn}^{D_{N'}}(T')$. Then $\mathcal{S}'(T', N') = \mathfrak{D}_D(T', N)$ for any $N \subseteq \mathcal{L}$ of N' such that $N' = N \cap (\neg\mathrm{J}(D))$. This is because $D_{N'} = D_N$ when $N' = N \cap (\neg\mathrm{J}(D))$. $\square$

Before presenting further subsystems for non-monotonic systems introduced in Section 2.3, we establish basic properties of the subsystem relation $\sqsubseteq$ for non-monotonic systems in Proposition 3.6. Counter examples which show that the relation $\sqsubseteq$ is not symmetric in general are obtained from subsystems $\mathcal{S}'$ having a *proper* sublanguage of a non-monotonic system $\mathcal{S}$, i.e. $\mathcal{L}(\mathcal{S}') \sqsubseteq \mathcal{L}(\mathcal{S})$ and $\mathcal{L}(\mathcal{S}') \neq \mathcal{L}(\mathcal{S})$ so that $\mathcal{L}(\mathcal{S}) \sqsubseteq \mathcal{L}(\mathcal{S}')$ cannot hold.

Proposition 3.6 *The subsystem relation $\sqsubseteq$ is reflexive and transitive.*

Proof. (i) Trivially, $\mathcal{L}(\mathcal{S}) \sqsubseteq \mathcal{L}(\mathcal{S})$ and it holds for each $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $N \subseteq \mathcal{L}^-(\mathcal{S})$ that $\mathcal{S}(T, N) = \mathcal{S}(T, N) \cap \mathcal{L}^+(\mathcal{S})$ where $N = N \cap \mathcal{L}^-(\mathcal{S})$. Thus $\mathcal{S} \sqsubseteq \mathcal{S}$ for all non-monotonic systems $\mathcal{S}$. (ii) To verify the transitivity of the subsystem relation, let $\mathcal{S}_1 \sqsubseteq \mathcal{S}_2$ and $\mathcal{S}_2 \sqsubseteq \mathcal{S}_3$ hold. Since $\mathcal{L}(\mathcal{S}_1) \sqsubseteq \mathcal{L}(\mathcal{S}_2)$ and $\mathcal{L}(\mathcal{S}_2) \sqsubseteq \mathcal{L}(\mathcal{S}_3)$, it obviously holds that $\mathcal{L}(\mathcal{S}_1) \sqsubseteq \mathcal{L}(\mathcal{S}_3)$. Then let $T \subseteq \mathcal{L}^+(\mathcal{S}_1)$ and $N_3 \subseteq \mathcal{L}^-(\mathcal{S}_3)$. Define $N_2 = N_3 \cap \mathcal{L}^-(\mathcal{S}_2)$ and $N_1 = N_2 \cap \mathcal{L}^-(\mathcal{S}_1)$. Because $\mathcal{L}^-(\mathcal{S}_1) \subseteq \mathcal{L}^-(\mathcal{S}_2) \subseteq \mathcal{L}^-(\mathcal{S}_3)$ it holds that $N_1 = N_3 \cap \mathcal{L}^-(\mathcal{S}_1)$ and $N_1 \subseteq N_2 \subseteq N_3$. Since $\mathcal{S}_1 \sqsubseteq \mathcal{S}_2$, $\mathcal{S}_2 \sqsubseteq \mathcal{S}_3$ and $\mathcal{L}^+(\mathcal{S}_1) \subseteq \mathcal{L}^+(\mathcal{S}_2)$, we obtain $\mathcal{S}_1(T, N_1) = \mathcal{S}_2(T, N_2) \cap \mathcal{L}^+(\mathcal{S}_1) = [\mathcal{S}_3(T, N_3) \cap \mathcal{L}^+(\mathcal{S}_2)] \cap \mathcal{L}^+(\mathcal{S}_1) = \mathcal{S}_3(T, N_3) \cap \mathcal{L}^+(\mathcal{S}_1)$ where $N_1 = N_3 \cap \mathcal{L}^-(\mathcal{S}_1)$. Because this holds for arbitrary $T \subseteq \mathcal{L}^+(\mathcal{S}_1)$ and $N_3 \subseteq \mathcal{L}^-(\mathcal{S}_3)$, it follows that $\mathcal{S}_1 \sqsubseteq \mathcal{S}_3$ as desired. $\square$

It is shown in Lemma 3.7 that fixing sentences as theorems or non-theorems does not affect the subsystem relation.

Lemma 3.7 *If $\mathcal{S}' \sqsubseteq \mathcal{S}$, then $\mathcal{S}'[F^+; F_1^-] \sqsubseteq \mathcal{S}[F^+; F_2^-]$ for all sets of theorems $F^+ \subseteq \mathcal{L}^+(\mathcal{S}')$ and for all sets of non-theorems $F_1^- \subseteq \mathcal{L}^-(\mathcal{S}')$ and $F_2^- \subseteq \mathcal{L}^-(\mathcal{S})$ such that $F_1^- = F_2^- \cap \mathcal{L}^-(\mathcal{S}')$.*

Proof. Let $T' \subseteq \mathcal{L}^+(\mathcal{S}'[F^+; F_1^-]) = \mathcal{L}^+(\mathcal{S}')$ and $N \subseteq \mathcal{L}^-(\mathcal{S}[F^+; F_2^-]) = \mathcal{L}^-(\mathcal{S})$ and $N' = N \cap \mathcal{L}^-(\mathcal{S}'[F^+; F_1^-]) = N \cap \mathcal{L}^-(\mathcal{S}')$. Now

$$\begin{aligned}\mathcal{S}'[F^+;F_1^-](T',N') &= \mathcal{S}'(F^+\cup T',F_1^-\cup N')\\ &= \mathcal{S}(F^+\cup T',F_2^-\cup N)\cap\mathcal{L}^+(\mathcal{S}')\\ &= \mathcal{S}[F^+;F_2^-](T',N)\cap\mathcal{L}^+(\mathcal{S}'[F^+;F^-]),\end{aligned}$$

since $\mathcal{S}'$ is a subsystem of $\mathcal{S}$, $F^+\cup T'\subseteq\mathcal{L}^+(\mathcal{S}')$, $F_2^-\cup N\subseteq\mathcal{L}^-(\mathcal{S})$ and $F_1^-\cup N'$ is the projection $(F_2^-\cup N)\cap\mathcal{L}^-(\mathcal{S}')$. □

In the following, we propose concrete examples of subsystems for minimal model systems $\mathfrak{C}_P$, default systems $\mathfrak{D}_D$ and autoepistemic systems $\mathcal{A}_\Sigma$. In fact, we introduce a range of subsystems parameterized by a set of sentences Γ for each of these systems. The reason for this is that we want to constrain the language of the subsystem in a way that is suitable for our later purposes. For minimal model systems and default systems Γ gives directly the language $\mathcal{L}^-(\mathcal{S})$ for a subsystem $\mathcal{S}$. For these systems, establishing subsystems is straightforward but necessary for our forthcoming treatment of these systems. Autoepistemic systems are more complicated. The parameter Γ affects both $\mathcal{L}^+(\mathcal{S})$ and $\mathcal{L}^-(\mathcal{S})$ in a way to be explained later (auxiliary definitions are needed to extract a suitable sublanguage). The presented subsystems will be used in two contexts mainly. First of all, they are required to establish embeddings among non-monotonic systems in Section 3.4. Secondly, subsystems obtained with a suitable choice of Γ will play an important role when algorithms for non-monotonic inference are considered in Sections 7.3 and 9.1.

3.1.1 Minimal Model Subsystems

Consider a set of atoms $P\subseteq\mathcal{A}$ to be minimized and the associated non-monotonic system $\mathfrak{C}_P$. The language $\mathcal{L}(\mathfrak{C}_P)$ was defined as $\langle\mathcal{L},\mathcal{L}\rangle$ where $\mathcal{L}$ is a propositional language based on the set of atomic propositions $\mathcal{A}$. We identify subsystems for $\mathfrak{C}_P$ in Definition 3.8. The definitions of $\mathfrak{C}_P^\Gamma$ and $\mathfrak{C}_P$ are basically the same, but N ranges in $\mathbf{2}^\Gamma$ for the former operator. Therefore, these operator produce the same sets of theorems for every $T\subseteq\mathcal{L}$ and $N\subseteq\Gamma$.

Definition 3.8 *For a set of atoms $P\subseteq\mathcal{A}$ and $\Gamma\subseteq\mathcal{L}$, define a binary inference operator $\mathfrak{C}_P^\Gamma$ by setting $\mathcal{L}(\mathfrak{C}_P^\Gamma)=\langle\mathcal{L},\Gamma\rangle$ and for all $T\subseteq\mathcal{L}$ and $N\subseteq\Gamma$,*

$$\mathfrak{C}_P^\Gamma(T,N)=\mathrm{Cn}(T\cup\mathrm{Neg}_P(N)).$$

However, this definition does not yield a subsystem of $\mathfrak{C}_P$ for every $\Gamma\subseteq\mathcal{L}$. Proposition 3.9 gives a sufficient condition for this to be the case. This implies that the simplest subsystem $\mathfrak{C}_P^P$ uses the language $\langle\mathcal{L},P\rangle$. This is because the atoms of P have the key role in order to determine the P-extensions (and thus P-minimal models) of a set of sentences $T\subseteq\mathcal{L}$.

Proposition 3.9 *For a set of atoms $P \subseteq \mathcal{A}$ and $\Gamma \subseteq \mathcal{L}$ such that $P \subseteq \Gamma$,*

$$\mathfrak{C}_P^\Gamma \sqsubseteq \mathfrak{C}_P.$$

Proof. Let $P \subseteq \mathcal{A}$ and $\Gamma \subseteq \mathcal{L}$ such that $P \subseteq \Gamma$. Then $\mathcal{L}(\mathfrak{C}_P^\Gamma) = \langle \mathcal{L}, \Gamma \rangle$ is a sublanguage $\mathcal{L}(\mathfrak{C}_P) = \langle \mathcal{L}, \mathcal{L} \rangle$ by definition. Then let $T' \subseteq \mathcal{L}$ and $N \subseteq \mathcal{L}$ in accordance with Definition 3.4. Because $P \subseteq \Gamma$, it follows that $\mathrm{Neg}_P(N) = \neg(P \cap N) = \mathrm{Neg}_P(N')$ where $N' = N \cap \Gamma$. Thus $\mathfrak{C}_P^\Gamma(T', N') = \mathrm{Cn}(T' \cup \mathrm{Neg}_P(N')) = \mathrm{Cn}(T' \cup \mathrm{Neg}_P(N)) \cap \mathcal{L} = \mathfrak{C}_P(T', N) \cap \mathcal{L}^+(\mathfrak{C}_P^\Gamma)$. Since T' and N were chosen arbitrarily, it follows that $\mathfrak{C}_P^\Gamma \sqsubseteq \mathfrak{C}_P$. □

3.1.2 Default Subsystems

Let us then take sets of defaults D and the associated non-monotonic systems $\mathfrak{D}_D$ into consideration. Recall that the language $\mathcal{L}^+(\mathfrak{D}_D)$ is $\langle \mathcal{L}, \mathcal{L} \rangle$ where $\mathcal{L}$ is the propositional language associated with the set of default rules D. Subsystems are identified in Definition 3.10. A straightforward relationship of $\mathfrak{D}_D^\Gamma$ and $\mathfrak{D}_D$ results, since $\mathfrak{D}_D^\Gamma(T, N) = \mathfrak{D}_D(T, N)$ for all $T \subseteq \mathcal{L}$ and $N \subseteq \Gamma$.

Definition 3.10 *For a set of defaults D in $\mathcal{L}$ and $\Gamma \subseteq \mathcal{L}$, define a binary inference operator $\mathfrak{D}_D^\Gamma$ by setting $\mathcal{L}(\mathfrak{D}_D^\Gamma) = \langle \mathcal{L}, \Gamma \rangle$ and for all $T \subseteq \mathcal{L}$ and $N \subseteq \Gamma$,*

$$\mathfrak{D}_D^\Gamma(T, N) = \mathrm{Cn}^{D_N}(T).$$

We indicate in Proposition 3.11 on what condition $\mathfrak{D}_D^\Gamma$ is a subsystem of $\mathfrak{D}_D$. The simplest possible subsystem which is obtained with $\Gamma = \neg\mathrm{J}(D)$ was already addressed in Example 3.5. Thus it is clear that the justifications of $\mathrm{J}(D)$ play a central role in default reasoning.

Proposition 3.11 *For a set D of defaults and $\Gamma \subseteq \mathcal{L}$ such that $\neg\mathrm{J}(D) \subseteq \Gamma$,*

$$\mathfrak{D}_D^\Gamma \sqsubseteq \mathfrak{D}_D.$$

Proof. Let D be a set of defaults in $\mathcal{L}$ and $\Gamma \subseteq \mathcal{L}$ such that $\neg\mathrm{J}(D) \subseteq \Gamma$. It holds clearly that $\mathcal{L}(\mathfrak{D}_D^\Gamma) = \langle \mathcal{L}, \Gamma \rangle$ is a sublanguage $\mathcal{L}(\mathfrak{D}_D) = \langle \mathcal{L}, \mathcal{L} \rangle$. Then let us choose $T' \subseteq \mathcal{L}$ and $N \subseteq \mathcal{L}$ and define $N' = N \cap \Gamma$. The reductions D_N and $D_{N'}$ coincide by the choice of Γ and N', since $\neg\mathrm{J}(D) \subseteq \Gamma$. Consequently, it follows that $\mathfrak{D}_D^\Gamma(T', N') = \mathrm{Cn}^{D_{N'}}(T') = \mathrm{Cn}^{D_N}(T') \cap \mathcal{L} = \mathfrak{D}_D(T, N) \cap \mathcal{L}^+(\mathfrak{D}_D^\Gamma)$. Thus $\mathfrak{D}_D^\Gamma \sqsubseteq \mathfrak{D}_D$ holds by the free choice of T' and N. □

3.1.3 Autoepistemic Subsystems

Finally, we propose subsystems for the non-monotonic system $\boldsymbol{\mathcal{A}}_\Sigma$ associated with an autoepistemic theory $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ where $\mathcal{L}_{\mathrm{BD}}$ is based on a set of atoms $\mathcal{A}$. In particular, the presented subsystems provide *finitary characterizations* of autoepistemic expansions (see e.g. the full set approach [37, 47, 75]) in our framework. Recall that the language $\boldsymbol{\mathcal{A}}_\Sigma$ was defined as $\langle \mathcal{L}_{\mathrm{BD}}, \mathcal{L}_{\mathrm{BD}} \rangle$. To define subsystems for $\boldsymbol{\mathcal{A}}_\Sigma$ we need some auxiliary definitions to specify a sublanguage of $\langle \mathcal{L}_{\mathrm{BD}}, \mathcal{L}_{\mathrm{BD}} \rangle$ which is suitable for our later purposes. For a sentence $\phi \in \mathcal{L}_{\mathrm{BD}}$ we let $\mathrm{At}^*_{\mathbf{BD}}(\phi)$ denote the set of **BD**-atoms that appear in ϕ *recursively.* A precise inductive definition of $\mathrm{At}^*_{\mathbf{BD}}(\phi)$ is given in Definition 3.12. We need also the non-recursive variant of this definition. The set of **BD**-atoms $\mathrm{At}_{\mathbf{BD}}(\phi)$ is defined by replacing the second and the third items of Definition 3.12 with $\mathrm{At}_{\mathbf{BD}}(\mathbf{B}\chi) = \{\mathbf{B}\chi\}$ and $\mathrm{At}_{\mathbf{BD}}(\mathbf{D}\chi) = \{\mathbf{D}\chi\}$. Note that we drop the symbol $*$ from the notation in the non-recursive case.

Definition 3.12 *For all $\chi \in \mathcal{L}_{\mathrm{BD}}$ and $\psi \in \mathcal{L}_{\mathrm{BD}}$ define*

1. $\mathrm{At}^*_{\mathbf{BD}}(\chi) = \emptyset$ *if* $\chi \in \mathcal{A}$,
2. $\mathrm{At}^*_{\mathbf{BD}}(\mathbf{B}\chi) = \{\mathbf{B}\chi\} \cup \mathrm{At}^*_{\mathbf{BD}}(\chi)$,
3. $\mathrm{At}^*_{\mathbf{BD}}(\mathbf{D}\chi) = \{\mathbf{D}\chi\} \cup \mathrm{At}^*_{\mathbf{BD}}(\chi)$,
4. $\mathrm{At}^*_{\mathbf{BD}}(\neg\chi) = \mathrm{At}^*_{\mathbf{BD}}(\chi)$ *and*
5. $\mathrm{At}^*_{\mathbf{BD}}(\chi \circ \psi) = \mathrm{At}^*_{\mathbf{BD}}(\chi) \cup \mathrm{At}^*_{\mathbf{BD}}(\psi)$

where $\circ$ is any of the binary connectives $\wedge$, $\vee$, $\rightarrow$ and $\leftrightarrow$.

Sometimes we want to pick up all **B**-atoms or all **D**-atoms from a set of **BD**-atoms. For this purpose we define $[\Lambda]^{\mathbf{B}} = \{\mathbf{B}\phi \,|\, \mathbf{B}\phi \in \Lambda\}$ and $[\Lambda]^{\mathbf{D}} = \{\mathbf{D}\phi \,|\, \mathbf{B}\phi \in \Lambda\}$ for a set of **BD**-atoms Λ. The **B**-atoms appearing in ϕ are captured by the definitions $\mathrm{At}_{\mathbf{B}}(\phi) = [\mathrm{At}_{\mathbf{BD}}(\phi)]^{\mathbf{B}}$ and $\mathrm{At}^*_{\mathbf{B}}(\phi) = [\mathrm{At}^*_{\mathbf{BD}}(\phi)]^{\mathbf{B}}$. Similar definitions are introduced for **D**-atoms. We extend the definitions of $\mathrm{At}_{\mathbf{B}}(\phi)$, $\mathrm{At}_{\mathbf{D}}(\phi)$, $\mathrm{At}_{\mathbf{BD}}(\phi)$, $\mathrm{At}^*_{\mathbf{B}}(\phi)$, $\mathrm{At}^*_{\mathbf{D}}(\phi)$ and $\mathrm{At}^*_{\mathbf{BD}}(\phi)$ for sets of sentences $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ in the obvious way. In some cases, we are also interested in the *subsentences* χ on which a set of **BD**-atoms is based. Therefore, we define e.g. $\mathrm{SS}^*_{\mathbf{BD}}(\phi)$ to be $\{\chi \,|\, \mathbf{B}\chi \in \mathrm{At}^*_{\mathbf{BD}}(\phi)$ or $\mathbf{D}\chi \in \mathrm{At}^*_{\mathbf{BD}}(\phi)\}$. A similar definition is introduced for the other operators that extract **BD**-atoms from a sentence $\phi \in \mathcal{L}_{\mathrm{BD}}$ or a set of sentences $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$. To give some differentiating examples, we note that $\mathrm{At}^*_{\mathbf{D}}(\mathbf{B}(\mathbf{B}\mathrm{p} \wedge \mathbf{D}\mathrm{p})) = \{\mathbf{D}\mathrm{p}\}$, $\mathrm{At}_{\mathbf{B}}(\mathbf{B}(\mathbf{B}\mathrm{p} \wedge \mathbf{D}\mathrm{p})) = \{\mathbf{B}(\mathbf{B}\mathrm{p} \wedge \mathbf{D}\mathrm{p})\}$ but $\mathrm{At}^*_{\mathbf{BD}}(\mathbf{B}(\mathbf{B}\mathrm{p} \wedge \mathbf{D}\mathrm{p})) = \{\mathbf{B}(\mathbf{B}\mathrm{p} \wedge \mathbf{D}\mathrm{p}), \mathbf{B}\mathrm{p}, \mathbf{D}\mathrm{p}\}$. For the autoepistemic theory Σ of Example 2.37, the set $\mathrm{SS}^*_{\mathbf{D}}(\Sigma) = \mathrm{SS}_{\mathbf{D}}(\Sigma) = \{\mathrm{b}\}$.

In Definition 3.13, we identify a range of subsystems for an autoepistemic system $\boldsymbol{\mathcal{A}}_\Sigma$ based on an autoepistemic theory $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$. The role of Γ becomes clear by now. The language for non-theorems is obtained from **D**-atoms that appear in Γ recursively. The language for theorems respects also the **B**-atoms in Γ. Note that e.g. $\Gamma = \{\mathbf{B}\mathrm{p}, \mathbf{D}\mathrm{q}\}$ gives rise to a sublanguage $\langle\{\mathrm{p},\mathrm{q}\},\{\mathrm{q}\}\rangle$ of $\langle\mathcal{L}_{\mathrm{BD}},\mathcal{L}_{\mathrm{BD}}\rangle$ according to our definition below.

Definition 3.13 *For an autoepistemic theory Σ and $\Gamma \subseteq \mathcal{L}_{\mathrm{BD}}$, define a binary inference operator $\boldsymbol{\mathcal{A}}^{\Gamma}_{\Sigma}$ by setting $\mathcal{L}(\boldsymbol{\mathcal{A}}^{\Gamma}_{\Sigma}) = \langle \mathrm{SS}^{*}_{\mathbf{BD}}(\Gamma), \mathrm{SS}^{*}_{\mathbf{D}}(\Gamma)\rangle$ and for all $T \subseteq \mathrm{SS}^{*}_{\mathbf{BD}}(\Gamma)$ and $N \subseteq \mathrm{SS}^{*}_{\mathbf{D}}(\Gamma)$,*

$$\boldsymbol{\mathcal{A}}^{\Gamma}_{\Sigma}(T,N) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N) \cap \mathrm{SS}^{*}_{\mathbf{BD}}(\Gamma).$$

To show on what conditions the non-monotonic systems introduced in Definition 3.13 are indeed subsystems of $\boldsymbol{\mathcal{A}}_{\Sigma}$, we have to derive some subsidiary results as follows. We note that only the **BD**-atoms in $\mathrm{At}_{\mathbf{BD}}(\phi)$ are significant when the truth value of ϕ is determined with respect to an autoepistemic model. This idea extends for logical consequences in Lemma 3.14. Niemelä [75] has established a similar result for conventional autoepistemic logic.

Lemma 3.14 *Let $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ be a set of sentences, Λ a set of* **BD***-atoms, $\phi \in \mathcal{L}_{\mathrm{BD}}$ a sentence and $\Lambda' = \Lambda \cap \mathrm{At}_{\mathbf{BD}}(\Sigma \cup \{\phi\})$. Then*

$$\Sigma \cup \Lambda \models \phi \textit{ implies } \Sigma \cup \Lambda' \models \phi.$$

Proof. We show the contrapositive. So let $\Sigma \cup \Lambda' \not\models \phi$ hold. Then there is a model $\mathcal{M}$ of $\Sigma \cup \Lambda'$ such that $\mathcal{M} \not\models \phi$. Consider a model $\mathcal{M}' = \mathcal{M} \cup (\Lambda - \Lambda')$ which is a model of $\Lambda - \Lambda'$ by definition.

Let us then show that for every $\mathbf{B}\chi \in \mathrm{At}_{\mathbf{BD}}(\Sigma \cup \{\phi\})$ it holds that $\mathbf{B}\chi \in \mathcal{M}$ if and only if $\mathbf{B}\chi \in \mathcal{M}'$. ($\Rightarrow$) This direction holds trivially since $\mathcal{M} \subseteq \mathcal{M}'$. ($\Leftarrow$) Let $\mathbf{B}\chi \in \mathcal{M}'$ hold and assume that $\mathbf{B}\chi \notin \mathcal{M}$. It follows that $\mathbf{B}\chi \in \Lambda - \Lambda'$ by the definition of $\mathcal{M}'$. Since $\mathbf{B}\chi \in \mathrm{At}_{\mathbf{BD}}(\Sigma \cup \{\phi\})$ and $\mathbf{B}\chi \in \Lambda$, it follows that $\mathbf{B}\chi \in \Lambda'$. But this is a contradiction. Thus $\mathbf{B}\chi \in \mathcal{M}$.

It can be similarly proved for $\mathbf{D}\chi \in \mathrm{At}_{\mathbf{BD}}(\Sigma \cup \{\phi\})$ that $\mathbf{D}\chi \in \mathcal{M}$ if and only if $\mathbf{D}\chi \in \mathcal{M}'$. Thus the models $\mathcal{M}'$ and $\mathcal{M}$ assign the same truth values for **BD**-atoms in $\mathrm{At}_{\mathbf{BD}}(\Sigma \cup \{\phi\})$. It follows that also $\mathcal{M}' \models \Sigma \cup \Lambda'$ but $\mathcal{M}' \not\models \phi$. Since $\mathcal{M}' \models \Lambda - \Lambda'$, we have that $\mathcal{M}' \models \Sigma \cup \Lambda$. Hence $\Sigma \cup \Lambda \not\models \phi$. □

An analogous result is deduced for **B**-provability in Lemmas 3.15 and 3.16. The difference is that we have to take **B**-atoms into account recursively. An interesting feature of Lemma 3.16 is how **B**-atoms are separated from **D**-atoms, since only **B**-atoms are involved in **B**-proofs.

Lemma 3.15 *Let $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ be a set of sentences, Λ a set of* **BD***-atoms, $\phi \in \mathcal{L}_{\mathrm{BD}}$ a sentence and $\Lambda' = \Lambda \cap \mathrm{At}^{*}_{\mathbf{BD}}(\Sigma \cup \{\phi\})$. Then*

$$\Sigma \cup \Lambda \models \phi \textit{ implies } \Sigma \cup \Lambda' \models \phi.$$

Proof sketch. This follows by the monotonicity of $\models$ and Lemma 3.14. □

Lemma 3.16 *For a set of sentences $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$, a set of **D**-atoms Λ, a sentence $\phi \in \mathcal{L}_{\mathrm{BD}}$ and $\Lambda' = \Lambda \cap \mathrm{At}^*_{\mathbf{D}}(\Sigma \cup \{\phi\})$,*

$$\Sigma \cup \Lambda \vdash_{\mathbf{B}} \phi \textit{ implies } \Sigma \cup \Lambda' \vdash_{\mathbf{B}} \phi.$$

*Moreover, there is a **B**-proof of ϕ from $\Sigma \cup \Lambda'$ which contains only belief atoms from $\mathrm{At}^*_{\mathbf{B}}(\Sigma \cup \{\phi\})$.*

Proof. Let $\Sigma \cup \Lambda \vdash_{\mathbf{B}} \phi$ hold, i.e. there is a **B**-proof $\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n$ of ϕ from $\Sigma \cup \Lambda$. Let $\mathbf{B}\phi'_1, ..., \mathbf{B}\phi'_m$ be the reduction of $\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n$ with respect to $\mathrm{At}^*_{\mathbf{B}}(\Sigma \cup \{\phi\})$ such that the order of the belief atoms is preserved. Define a function $f : \{1, ..., m\} \to \{1, ..., n\}$ to give the correspondence between the two sequences of belief atoms such that $\phi'_i = \phi_{f(i)}$. It should be clear that $f(i)$ gives the position of $\mathbf{B}\phi'_i$ in the original **B**-proof $\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n$. It is then proved for each $0 < i \leq m$ that

$$\Sigma \cup \Lambda \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_{f(i)-1}\} \models \phi_{f(i)} \text{ implies } \Sigma \cup \Lambda' \cup \{\mathbf{B}\phi'_1, ..., \mathbf{B}\phi'_{i-1}\} \models \phi'_i.$$

So let $\Sigma \cup \Lambda \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_{f(i)-1}\} \models \phi_{f(i)}$ hold. Since $\Lambda \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_{f(i)-1}\}$ is a set of **BD**-atoms, we obtain by Lemma 3.15 that

$$\Sigma \cup (\Lambda \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_{f(i)-1}\}) \cap \mathrm{At}^*_{\mathbf{BD}}(\Sigma \cup \{\phi_{f(i)}\}) \models \phi_{f(i)}.$$

Since $\phi_{f(i)} = \phi'_i$, we have that $\mathbf{B}\phi_{f(i)} \in \mathrm{At}^*_{\mathbf{BD}}(\Sigma \cup \{\phi\})$. It follows that $\mathrm{At}^*_{\mathbf{BD}}(\Sigma \cup \{\phi_{f(i)}\}) \subseteq \mathrm{At}^*_{\mathbf{BD}}(\Sigma \cup \{\phi\})$. Therefore

$$\begin{aligned}
& (\Lambda \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_{f(i)-1}\}) \cap \mathrm{At}^*_{\mathbf{BD}}(\Sigma \cup \{\phi_{f(i)}\}) \\
\subseteq\ & (\Lambda \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_{f(i)-1}\}) \cap \mathrm{At}^*_{\mathbf{BD}}(\Sigma \cup \{\phi\}) \\
=\ & [\Lambda \cap \mathrm{At}^*_{\mathbf{D}}(\Sigma \cup \{\phi\})] \cup [\{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_{f(i)-1}\} \cap \mathrm{At}^*_{\mathbf{B}}(\Sigma \cup \{\phi\})] \\
=\ & \Lambda' \cup \{\mathbf{B}\phi'_1, ..., \mathbf{B}\phi'_{i-1}\}
\end{aligned}$$

and $\Sigma \cup \Lambda' \cup \{\mathbf{B}\phi'_1, ..., \mathbf{B}\phi'_{i-1}\} \models \phi'_i$ follows by the monotonicity of $\models$.

It follows that $\mathbf{B}\phi'_1, ..., \mathbf{B}\phi'_m$ satisfies the requirements of a **B**-proof from $\Sigma \cup \Lambda'$, since $\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n$ is a **B**-proof from $\Sigma \cup \Lambda$. What remains to be shown is that $\mathbf{B}\phi'_1, ..., \mathbf{B}\phi'_m$ is a **B**-proof of ϕ from $\Sigma \cup \Lambda'$. Since $\Sigma \cup \Lambda \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\} \models \phi$ and $(\Lambda \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\}) \cap \mathrm{At}^*_{\mathbf{BD}}(\Sigma \cup \{\phi\}) = \Lambda' \cup \{\mathbf{B}\phi'_1, ..., \mathbf{B}\phi'_m\}$, it follows that $\Sigma \cup \Lambda' \cup \{\mathbf{B}\phi'_1, ..., \mathbf{B}\phi'_m\} \models \phi$ by Lemma 3.16. Finally, recall that the **B**-proof $\mathbf{B}\phi'_1, ..., \mathbf{B}\phi'_n$ contains only **B**-atoms from $\mathrm{At}^*_{\mathbf{B}}(\Sigma \cup \{\phi\})$. □

We may now establish autoepistemic subsystems.

Proposition 3.17 *For an autoepistemic theory* $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ *and* $\Gamma \subseteq \mathcal{L}_{\mathrm{BD}}$ *such that* $\mathrm{At}^*_{\mathbf{BD}}(\Sigma) \subseteq \mathrm{At}^*_{\mathbf{BD}}(\Gamma)$,

$$\mathcal{A}^{\Gamma}_{\Sigma} \sqsubseteq \mathcal{A}_{\Sigma}.$$

Proof. Now $\mathrm{SS}^*_{\mathbf{D}}(\Gamma) \subseteq \mathrm{SS}^*_{\mathbf{BD}}(\Gamma)$ holds by definition so that the language $\mathcal{L}(\mathcal{A}^{\Gamma}_{\Sigma}) = \langle \mathrm{SS}^*_{\mathbf{BD}}(\Gamma), \mathrm{SS}^*_{\mathbf{D}}(\Gamma)\rangle$ is a sublanguage of $\mathcal{L}(\mathcal{A}_{\Sigma}) = \langle \mathcal{L}_{\mathrm{BD}}, \mathcal{L}_{\mathrm{BD}}\rangle$. Let us pick any set of theorems $T' \subseteq \mathrm{SS}^*_{\mathbf{BD}}(\Gamma)$ and any set of non-theorems $N \subseteq \mathcal{L}_{\mathrm{BD}}$ and define $N' = N \cap \mathrm{SS}^*_{\mathbf{D}}(\Gamma)$. It is shown below for all $\phi \in \mathrm{SS}^*_{\mathbf{BD}}(\Gamma)$ that $\Sigma \cup T' \cup \mathbf{D}N' \vdash_{\mathbf{B}} \phi$ if and only if $\Sigma \cup T' \cup \mathbf{D}N \vdash_{\mathbf{B}} \phi$. This implies that $\mathcal{A}^{\Gamma}_{\Sigma}(T', N') = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T' \cup \mathbf{D}N') \cap \mathrm{SS}^*_{\mathbf{BD}}(\Gamma) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T' \cup \mathbf{D}N) \cap \mathrm{SS}^*_{\mathbf{BD}}(\Gamma) = \mathcal{A}_{\Sigma}(T', N) \cap \mathcal{L}^+(\mathcal{A}^{\Gamma}_{\Sigma})$. Thus $\mathcal{A}^{\Gamma}_{\Sigma} \sqsubseteq \mathcal{A}_{\Sigma}$ holds by the free choice of T' and N.

($\Rightarrow$) This follows by the monotonicity of **B**-provability (Proposition 2.29), since $\Sigma \cup T' \cup \mathbf{D}N'$ is a subset of $\Sigma \cup T' \cup \mathbf{D}N$. ($\Leftarrow$) Let $\Sigma \cup T' \cup \mathbf{D}N \vdash_{\mathbf{B}} \phi$ hold. Since $\mathrm{At}^*_{\mathbf{BD}}(\Sigma) \subseteq \mathrm{At}^*_{\mathbf{BD}}(\Gamma)$ by our requirement and $T' \cup \{\phi\} \subseteq \mathrm{SS}^*_{\mathbf{BD}}(\Gamma)$, it follows that $\mathrm{At}^*_{\mathbf{BD}}(\Sigma \cup T' \cup \{\phi\}) \subseteq \mathrm{At}^*_{\mathbf{BD}}(\Gamma)$. Thus $\mathbf{D}N \cap \mathrm{At}^*_{\mathbf{D}}(\Sigma \cup T' \cup \{\phi\})$ is a subset of $\mathbf{D}N'$ and $\Sigma \cup T' \cup \mathbf{D}N \vdash_{\mathbf{B}} \phi$ implies $\Sigma \cup T' \cup \mathbf{D}N' \vdash_{\mathbf{B}} \phi$ by Lemma 3.16 and the monotonicity of **B**-provability. □

3.2 Functional Equivalence

To formalize a notion of equivalence for non-monotonic systems $\mathcal{S}_1$ and $\mathcal{S}_2$ (with possibly heterogenous languages) we address mappings between the languages associated with $\mathcal{S}_1$ and $\mathcal{S}_2$ first. The goal is to establish a correspondence between $\mathcal{L}(\mathcal{S}_1)$ and $\mathcal{L}(\mathcal{S}_2)$ by identifying *counter-sentences* in these languages. For this purpose, we introduce a pair $\langle f^+, f^-\rangle$ of functions where $f^+ : \mathcal{L}^+(\mathcal{S}_1) \to \mathcal{L}^+(\mathcal{S}_2)$ and $f^- : \mathcal{L}^-(\mathcal{S}_1) \to \mathcal{L}^-(\mathcal{S}_2)$ map sentences of $\mathcal{L}(\mathcal{S}_1)$ to sentences of $\mathcal{L}(\mathcal{S}_2)$. The function f^+ is extended for subsets $S \subseteq \mathcal{L}^+(\mathcal{S}_1)$ by the definition $f^+(S) = \bigcup\{f^+(\phi) \mid \phi \in S\}$ which is obviously the image of a set of sentences $S \subseteq \mathcal{L}^+(\mathcal{S})$ under f^+. A similar definition extends f^- for sets of sentences $S \subseteq \mathcal{L}^-(\mathcal{S}_1)$. Definition 3.18 provides our obvious requirement for a *language morphism* (an *l-morphism* for short) from $\mathcal{S}_1$ to $\mathcal{S}_2$:[15] the image of $\mathcal{L}(\mathcal{S}_1) = \langle \mathcal{L}^+(\mathcal{S}_1), \mathcal{L}^-(\mathcal{S}_1)\rangle$ under $\langle f^+, f^-\rangle$ must be a sublanguage of $\mathcal{L}(\mathcal{S}_2) = \langle \mathcal{L}^+(\mathcal{S}_2), \mathcal{L}^-(\mathcal{S}_2)\rangle$. An l-morphism $f = \langle f^+, f^-\rangle$ is *injective* if $f^+ : \mathcal{L}^+(\mathcal{S}_1) \to \mathcal{L}^+(\mathcal{S}_2)$ and $f^- : \mathcal{L}^-(\mathcal{S}_1) \to \mathcal{L}^-(\mathcal{S}_2)$ are injective functions. Analogous definitions are introduced for *surjectivity* and *bijectivity* of l-morphisms in a standard way.

Definition 3.18 *Let* $\mathcal{S}_1$ *and* $\mathcal{S}_2$ *be non-monotonic systems. A pair of functions* $f = \langle f^+, f^-\rangle$ *where* $f^+ : \mathcal{L}^+(\mathcal{S}_1) \to \mathcal{L}^+(\mathcal{S}_2)$ *and* $f^- : \mathcal{L}^-(\mathcal{S}_1) \to \mathcal{L}^-(\mathcal{S}_2)$ *is an l-morphism from* $\mathcal{S}_1$ *to* $\mathcal{S}_2$ *if and only if* $f^-(\mathcal{L}^-(\mathcal{S}_1)) \subseteq f^+(\mathcal{L}^+(\mathcal{S}_1))$.

[15] Indeed, we say that an l-morphism is a mapping between the systems $\mathcal{S}_1$ and $\mathcal{S}_2$ (and not between $\mathcal{L}(\mathcal{S}_1)$ and $\mathcal{L}(\mathcal{S}_2)$), since it should be clear that an l-morphism maps sentences.

In Definition 3.19, we give a condition on which two sentences ϕ and ϕ' are considered to be *equivalent* in a non-monotonic system $\mathcal{S}$. The notion of $\mathcal{S}$-equivalence may differ from what is understood by the logical equivalence of sentences in classical logic. To provide a simple example in the monotonic case, consider a system $\mathcal{S}$ based on a propositional language $\mathcal{L}$ and defined by $\mathcal{S}(T) = \mathrm{Cn}(\{\mathrm{a}\} \cup T)$ where a is an atomic sentence. Within this system, sentences b and $\mathrm{a} \rightarrow \mathrm{b}$ are $\mathcal{S}$-equivalent even though they are not logically equivalent sentences in propositional logic. The relationship is straightforward in the other direction as long as we are dealing with systems based on closures under propositional logic or its derivatives. In our previous example, propositionally equivalent sentences (e.g. a and $\neg\neg\mathrm{a}$) are necessarily $\mathcal{S}$-equivalent sentences. Similar observations can be made on non-monotonic systems.

Definition 3.19 *Two sentences ϕ and ϕ' of $\mathcal{L}^+(\mathcal{S})$ are $\mathcal{S}$-equivalent given that it holds for all $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $N \subseteq \mathcal{L}^-(\mathcal{S})$ that $\phi \in \mathcal{S}(T, N)$ iff $\phi' \in \mathcal{S}(T, N)$.*

In Definition 3.20, we single out l-morphisms f that establish the *functional equivalence* of two non-monotonic systems $\mathcal{S}_1$ and $\mathcal{S}_2$. We write $\mathcal{S}_1 \sim \mathcal{S}_2$ to denote that $\mathcal{S}_1$ is functionally equivalent to $\mathcal{S}_2$. To emphasize the role of the l-morphism f intermediating the functional equivalence of $\mathcal{S}_1$ and $\mathcal{S}_2$, we say that $\mathcal{S}_1 \sim \mathcal{S}_2$ *via* f. A short justification follows the definition.

Definition 3.20 $\mathcal{S}_1$ *and* $\mathcal{S}_2$ *are functionally equivalent if and only if there is an l-morphism $f = \langle f^+, f^- \rangle$ from $\mathcal{S}_1$ to $\mathcal{S}_2$ such that (i) f is a bijection (ii) it holds for each $\phi \in \mathcal{L}^-(\mathcal{S}_1)$ that the sentences $f^-(\phi)$ and $f^+(\phi)$ are $\mathcal{S}_2$-equivalent and (iii) it holds for all $T \subseteq \mathcal{L}^+(\mathcal{S}_1)$ and $N \subseteq \mathcal{L}^-(\mathcal{S}_1)$ that*

$$f^+(\mathcal{S}_1(T, N)) = \mathcal{S}_2(f^+(T), f^-(N)).$$

(i) The first condition guarantees that f^+ and f^- map sentences of $\mathcal{L}(\mathcal{S}_1)$ to unique counter sentences of $\mathcal{L}(\mathcal{S}_2)$. (ii) It is permitted that f^+ and f^- map a sentence $\phi \in \mathcal{L}^-(\mathcal{S}_1)$ to different sentences, i.e. $f^+(\phi) \neq f^-(\phi)$, as long as these sentences are $\mathcal{S}_2$-equivalent. We need this flexibility in conjunction of l-morphisms to be introduced in Section 3.4. (iii) The l-morphism f preserves the functionality of $\mathcal{S}_1$. This is how $\mathcal{S}_2$ has the same functional capability as $\mathcal{S}_1$. In Proposition 3.21, we address the basic properties of $\sim$. By the symmetry of $\sim$, the inverse l-morphism $(f)^{-1} = \langle (f^+)^{-1}, (f^-)^{-1} \rangle$ preserves the functionality of $\mathcal{S}_2$, and consequently the systems $\mathcal{S}_1$ and $\mathcal{S}_2$ have the same functionality as non-monotonic systems.

Proposition 3.21 *The relation $\sim$ is an equivalence relation.*

Proof. For the reflexivity, consider the identity l-morphism $i = \langle i^+, i^- \rangle$ for the language $\mathcal{L}(\mathcal{S})$. (i) This is clearly a bijection. (ii) A sentence $\phi \in \mathcal{L}^-(\mathcal{S})$ is mapped to sentences $i^+(\phi) = \phi$ and $i^-(\phi) = \phi$ which are the same and thus trivially $\mathcal{S}$-equivalent. (iii) It holds for all $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $N \subseteq \mathcal{L}^-(\mathcal{S})$ that $i^+(\mathcal{S}(T,N)) = \mathcal{S}(T,N) = \mathcal{S}(i^+(T), i^-(N))$. Thus $\mathcal{S} \sim \mathcal{S}$ via i.

To see the symmetry of the functional equivalence relation, let $\mathcal{S}_1 \sim \mathcal{S}_2$ hold. Then there is an l-morphism $f = \langle f^+, f^- \rangle$ from $\mathcal{S}_1$ to $\mathcal{S}_2$ such that $\mathcal{S}_1 \sim \mathcal{S}_2$ via f. This implies that f is a bijection and consequently $g = (f)^{-1}$ is a well-defined l-morphism. (i) It is clear that $g = \langle g^+, g^- \rangle = \langle (f^+)^{-1}, (f^-)^{-1} \rangle$ is a bijection. (ii) Let ϕ be any sentence of $\mathcal{L}^-(\mathcal{S}_2)$ and T and N any subsets of $\mathcal{L}^+(\mathcal{S}_1)$ and $\mathcal{L}^-(\mathcal{S}_1)$, respectively.

$$
\begin{array}{rl|l}
\text{Now} & g^-(\phi) \in \mathcal{S}_1(T,N) & \\
\text{iff} & f^+(g^-(\phi)) \in \mathcal{S}_2(f^+(T), f^-(N)) & \mathcal{S}_1 \sim \mathcal{S}_2 \text{ via } f \\
\text{iff} & f^-(g^-(\phi)) \in \mathcal{S}_2(f^+(T), f^-(N)) & \mathcal{S}_2\text{-equivalence} \\
\text{iff} & \phi \in \mathcal{S}_2(f^+(T), f^-(N)) & g^- \circ f^- = i^- \\
\text{iff} & g^+(\phi) \in \mathcal{S}_1(T,N). & \mathcal{S}_1 \sim \mathcal{S}_2 \text{ via } f \text{ and } g^+ = (f^+)^{-1}
\end{array}
$$

This indicates that g^+ and g^- map a sentence $\phi \in \mathcal{L}^-(\mathcal{S}_2)$ into $\mathcal{S}_1$-equivalent sentences. (iii) Let us pick up any $T \subseteq \mathcal{L}^+(\mathcal{S}_2)$ and $N \subseteq \mathcal{L}^-(\mathcal{S}_2)$.

$$
\begin{array}{rl|l}
\text{Then} & g^+(\mathcal{S}_2(T,N)) & \\
= & g^+[\mathcal{S}_2(f^+[g^+(T)], f^-[g^-(N)])] & g \circ f = i \\
= & g^+[f^+(\mathcal{S}_1(g^+(T), g^-(N))] & \mathcal{S}_1 \sim \mathcal{S}_2 \text{ via } f \\
= & \mathcal{S}_1(g^+(T), g^-(N)) & f^+ \circ g^+ = i^+
\end{array}
$$

by the properties of f. Thus the inverse l-morphism g establishes the functional equivalence of $\mathcal{S}_2$ to $\mathcal{S}_1$. By (i-iii), it holds that $\mathcal{S}_2 \sim \mathcal{S}_1$ via g.

For the transitivity, assume that $\mathcal{S}_1 \sim \mathcal{S}_2$ and $\mathcal{S}_2 \sim \mathcal{S}_3$ hold. Then there are l-morphisms $f_1 = \langle f_1^+, f_1^- \rangle$ from $\mathcal{S}_1$ to $\mathcal{S}_2$ and $f_2 = \langle f_2^+, f_2^- \rangle$ from $\mathcal{S}_2$ to $\mathcal{S}_3$ such that $\mathcal{S}_1 \sim \mathcal{S}_2$ via f_1 and $\mathcal{S}_2 \sim \mathcal{S}_3$ via f_2. Let f be the obvious composition $f_1 \circ f_2$ of f_1 and f_2 defined by $\langle f_1^+ \circ f_2^+, f_1^- \circ f_2^- \rangle$. (i) It is clear that f is a bijection, since it is a composition of the bijections f_1 and f_2. (ii) Let $\phi \in \mathcal{L}^-(\mathcal{S}_1)$ and T and N any subsets of $\mathcal{L}^+(\mathcal{S}_3)$ and $\mathcal{L}^-(\mathcal{S}_3)$, respectively.

$$
\begin{array}{rl|l}
\text{Now} & f_2^+(f_1^+(\phi)) \in \mathcal{S}_3(T,N) & \\
\text{iff} & f_1^+(\phi) \in \mathcal{S}_2((f_2^+)^{-1}(T), (f_2^-)^{-1}(N)) & \mathcal{S}_3 \sim \mathcal{S}_2 \text{ via } (f_2)^{-1} \\
\text{iff} & f_1^-(\phi) \in \mathcal{S}_2((f_2^+)^{-1}(T), (f_2^-)^{-1}(N)) & \mathcal{S}_2\text{-equivalence} \\
\text{iff} & f_2^+(f_1^-(\phi)) \in \mathcal{S}_3(T,N) & \mathcal{S}_2 \sim \mathcal{S}_3 \text{ via } f_2 \\
\text{iff} & f_2^-(f_1^-(\phi)) \in \mathcal{S}_3(T,N). & \mathcal{S}_3\text{-equivalence}
\end{array}
$$

Thus $f^+(\phi) = f_2^+(f_1^+(\phi))$ and $f^-(\phi) = f_2^-(f_1^-(\phi))$ are $\mathcal{S}_3$-equivalent sentences. (iii) Let T and N be any subsets of the languages $\mathcal{L}^+(\mathcal{S}_1)$ and $\mathcal{L}^-(\mathcal{S}_1)$, respectively. Since $\mathcal{S}_1 \sim \mathcal{S}_2$ and $\mathcal{S}_2 \sim \mathcal{S}_3$, it follows that $f_2^+(f_1^+(\mathcal{S}_1(T,N))) = f_2^+(\mathcal{S}_2(f_1^+(T), f_1^-(N))) = \mathcal{S}_3(f_2^+(f_1^+(T)), f_2^-(f_1^-(N)))$. Thus $f^+(\mathcal{S}_1(T,N) =$

$\mathcal{S}_3(f^+(T), f^-(N))$ by the definition of f. We know by the cases (i-iii) that $\mathcal{S}_1 \sim \mathcal{S}_3$ via the composition $f = f_1 \circ f_2$. □

Functional equivalence preserves subsystems as shown in Proposition 3.22, given that the language of the subsystem $\mathcal{L}(\mathcal{S}'_1)$ is mapped to an acceptable sublanguage $\mathcal{L}(\mathcal{S}'_2)$ of $\mathcal{L}(\mathcal{S}_2)$, i.e. $f^-(\mathcal{L}^-(\mathcal{S}'_1)) \subseteq f^+(\mathcal{L}^+(\mathcal{S}'_1))$ holds.

Proposition 3.22 *If $\mathcal{S}_1 \sim \mathcal{S}_2$ via $f = \langle f^+, f^- \rangle$, $\mathcal{S}'_1 \sqsubseteq \mathcal{S}_1$ and $f^-(\mathcal{L}^-(\mathcal{S}'_1)) \subseteq f^+(\mathcal{L}^+(\mathcal{S}'_1))$, then there is a subsystem $\mathcal{S}'_2 \sqsubseteq \mathcal{S}_2$ such that $\mathcal{S}'_1 \sim \mathcal{S}'_2$.*

Proof. Let $\mathcal{S}_1 \sim \mathcal{S}_2$ via $f = \langle f^+, f^- \rangle$. Define $g = \langle g^+, g^- \rangle$ as f restricted to the sublanguage $\mathcal{L}(\mathcal{S}'_1)$ of $\mathcal{L}(\mathcal{S}_1)$. Define a non-monotonic system $\mathcal{S}'_2$ by setting $\mathcal{L}(\mathcal{S}'_2) = \langle g^+(\mathcal{L}^+(\mathcal{S}'_1)), g^-(\mathcal{L}^-(\mathcal{S}'_1)) \rangle$ and

$$\mathcal{S}'_2(T', N') = g^+(\mathcal{S}'_1((g^+)^{-1}(T'), (g^-)^{-1}(N')))$$

for all $T' \subseteq \mathcal{L}^-(\mathcal{S}'_2)$ and $N' \subseteq \mathcal{L}^-(\mathcal{S}'_2)$. Our requirement $f^-(\mathcal{L}^-(\mathcal{S}'_1)) \subseteq f^+(\mathcal{L}^+(\mathcal{S}'_1))$ implies that $g^-(\mathcal{L}^-(\mathcal{S}'_1)) \subseteq g^+(\mathcal{L}^+(\mathcal{S}'_1))$ so that the function g is an l-morphism. Thus $\mathcal{L}(\mathcal{S}'_2)$ is indeed a sublanguage of $\mathcal{L}(\mathcal{S}_2)$.

To establish $\mathcal{S}'_2 \sqsubseteq \mathcal{S}_2$ we pick an arbitrary set of theorems $T' \subseteq \mathcal{L}^+(\mathcal{S}'_2)$ and an arbitrary set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S}_2)$ and define $N' = N \cap \mathcal{L}^-(\mathcal{S}'_2)$ in the spirit of the definition of subsystems. Note that $(g^-)^{-1}(N') = (g^-)^{-1}(N \cap \mathcal{L}^-(\mathcal{S}'_2)) = (f^-)^{-1}(N) \cap \mathcal{L}^-(\mathcal{S}'_1)$, since g is f restricted to $\mathcal{L}(\mathcal{S}'_1)$.

$$
\begin{array}{rll|l}
\text{Now} & \mathcal{S}'_2(T', N') & & \\
= & g^+(\mathcal{S}'_1((g^+)^{-1}(T'), (g^-)^{-1}(N'))) & & \text{Definition of } \mathcal{S}'_2 \\
= & g^+(\mathcal{S}'_1((f^+)^{-1}(T'), (f^-)^{-1}(N) \cap \mathcal{L}^-(\mathcal{S}'_1))) & & \text{See above} \\
= & g^+(\mathcal{S}_1((f^+)^{-1}(T'), (f^-)^{-1}(N)) \cap \mathcal{L}^+(\mathcal{S}'_1)) & & \mathcal{S}'_1 \sqsubseteq \mathcal{S}_1 \\
= & f^+(\mathcal{S}_1((f^+)^{-1}(T'), (f^-)^{-1}(N))) \cap \mathcal{L}^+(\mathcal{S}'_2) & & f^+ \text{ extends } g^+ \\
= & \mathcal{S}_2(f^+((f^+)^{-1}(T')), f^-((f^-)^{-1}(N))) \cap \mathcal{L}^+(\mathcal{S}'_2) & & \mathcal{S}_1 \sim \mathcal{S}_2 \text{ via } f \\
= & \mathcal{S}_2(T', N) \cap \mathcal{L}^+(\mathcal{S}'_2). & & (f)^{-1} \circ f = i
\end{array}
$$

Let us then verify the requirements of Definition 3.20 to establish $\mathcal{S}'_1 \sim \mathcal{S}'_2$. (i) First of all, the function g is a bijection, since f is. (ii) Then consider any $\phi \in \mathcal{L}^-(\mathcal{S}'_1)$. Let T' and N' be any subsets of $\mathcal{L}^+(\mathcal{S}'_2)$ and $\mathcal{L}^-(\mathcal{S}'_2)$, respectively.

$$
\begin{array}{rl|l}
\text{Now} & g^+(\phi) \in \mathcal{S}'_2(T', N') & \\
\text{iff} & g^+(\phi) \in \mathcal{S}_2(T', N') \cap \mathcal{L}^+(\mathcal{S}'_2) & \mathcal{S}'_2 \sqsubseteq \mathcal{S}_2 \\
\text{iff} & f^+(\phi) \in \mathcal{S}_2(T', N') \cap \mathcal{L}^+(\mathcal{S}'_2) & f^+(\phi) = g^+(\phi) \\
\text{iff} & f^-(\phi) \in \mathcal{S}_2(T', N') \cap \mathcal{L}^+(\mathcal{S}'_2) & \mathcal{S}_2\text{-equivalence} \\
\text{iff} & g^-(\phi) \in \mathcal{S}'_2(T', N'). & f^-(\phi) = g^-(\phi)
\end{array}
$$

(iii) Finally, it holds for all $T \subseteq \mathcal{L}^+(\mathcal{S}'_1)$ and $N \subseteq \mathcal{L}^-(\mathcal{S}'_1)$ that $g^+(\mathcal{S}'_1(T, N)) = g^+(\mathcal{S}'_1((g^+)^{-1}(g^+(T)), (g^-)^{-1}(g^-(N)))) = \mathcal{S}'_2(g^+(T), g^-(N))$ by the definition of $\mathcal{S}'_2$. Thus $\mathcal{S}'_1$ is functionally equivalent to $\mathcal{S}'_2$ via g. □

Our last subsidiary result concerning functional equivalence is Lemma 3.23 which addresses the fixing operation.

Lemma 3.23 *If $\mathcal{S}' \sim \mathcal{S}$ via $f = \langle f^+, f^- \rangle$, then $\mathcal{S}'[F^+; F^-]$ is functionally equivalent to $\mathcal{S}[f^+(F^+); f^-(F^-)]$ via f for all sets of theorems $F^+ \subseteq \mathcal{L}^+(\mathcal{S}')$ and non-theorems $F^- \subseteq \mathcal{L}^-(\mathcal{S}')$ fixed for $\mathcal{S}'$.*

Proof. Let $T \subseteq \mathcal{L}^+(\mathcal{S}'[F^+; F^-]) = \mathcal{L}^+(\mathcal{S}')$ and $N \subseteq \mathcal{L}^-(\mathcal{S}'[F^+; F^-]) = \mathcal{L}^-(\mathcal{S}')$. It follows by the definitions of functional equivalence and fixing that

$$\begin{aligned}
f^+(\mathcal{S}'[F^+; F^-](T, N)) &= f^+(\mathcal{S}'(F^+ \cup T, F^- \cup N)) \\
&= \mathcal{S}(f^+(F^+ \cup T), f^-(F^- \cup N)) \\
&= \mathcal{S}(f^+(F^+) \cup f^+(T), f^-(F^-) \cup f^-(N)) \\
&= \mathcal{S}[f^+(F^+); f^-(F^-)](f^+(T), f^-(N)).
\end{aligned}$$

□

3.3 Embeddings

We are now ready to introduce a notion of an *embedding* by combining the notions of a subsystem and functional equivalence. Embeddings will be repeatedly used in Section 3.4 where we propose syntactic translations that give rise to embeddings between non-monotonic systems in a systematic way. Our intuition with embeddings is as follows. An l-morphism f embeds $\mathcal{S}_1$ in $\mathcal{S}_2$ if $\mathcal{S}_1$ is functionally equivalent with a subsystem $\mathcal{S}$ of $\mathcal{S}_2$ via f. We introduce the notation $\mathcal{S}_1 \mapsto_f \mathcal{S}_2$ to denote this. We write $\mathcal{S}_1 \mapsto \mathcal{S}_2$ if there is an l-morphism f such that $\mathcal{S}_1 \mapsto_f \mathcal{S}_2$ and we want to ignore f.

Definition 3.24 *An l-morphism $f = \langle f^+, f^- \rangle$ from $\mathcal{S}_1$ to $\mathcal{S}_2$ is an embedding of $\mathcal{S}_1$ in $\mathcal{S}_2$, denoted by $\mathcal{S}_1 \mapsto_f \mathcal{S}_2$, if and only if there is a subsystem $\mathcal{S} \sqsubseteq \mathcal{S}_2$ such that $\mathcal{S}_1 \sim \mathcal{S}$ via the l-morphism f.*

To provide a basic understanding about the embedding relation $\mapsto$ we address its properties in Proposition 3.25. Notice also that $\mapsto$ is not symmetric in general and thus not an equivalence relation.

Proposition 3.25 *For any non-monotonic systems $\mathcal{S}$, $\mathcal{S}_1$ and $\mathcal{S}_2$ as well as the identity l-morphism i and l-morphisms f and g,*

(i) $\mathcal{S} \mapsto_i \mathcal{S}$,
(ii) $\mathcal{S}_1 \sqsubseteq \mathcal{S}_2$ *implies* $\mathcal{S}_1 \mapsto_i \mathcal{S}_2$
(iii) $\mathcal{S}_1 \sim \mathcal{S}_2$ *via* f *implies* $\mathcal{S}_1 \mapsto_f \mathcal{S}_2$ *and*

if $g^-(\mathcal{L}^-(\mathcal{S}_2')) \subseteq g^+(\mathcal{L}^+(\mathcal{S}_2'))$ for the subsystem $\mathcal{S}_2'$ involved in $\mathcal{S}_1 \mapsto_f \mathcal{S}_2$, *(iv)* $\mathcal{S}_1 \mapsto_f \mathcal{S}_2$ *and* $\mathcal{S}_2 \mapsto_g \mathcal{S}_3$ *imply* $\mathcal{S}_1 \mapsto_{f\circ g} \mathcal{S}_3$.

Proof. (i) By Proposition 3.21, it holds that $\mathcal{S} \sim \mathcal{S}$ via the identity l-morphism i. Since also $\mathcal{S} \sqsubseteq \mathcal{S}$ by Proposition of 3.6, we have that $\mathcal{S} \mapsto_i \mathcal{S}$. The claim (ii) follows directly from Definition 3.24, since $\mathcal{S}_1 \sim \mathcal{S}_1$ via i by Proposition 3.21. (iii) This holds also by Definition 3.24, because $\mathcal{S}_2 \sqsubseteq \mathcal{S}_2$ by Proposition 3.6.

(iv) Let f be an embedding of $\mathcal{S}_1$ in $\mathcal{S}_2$ and g be an embedding of $\mathcal{S}_2$ in $\mathcal{S}_3$. Let $\mathcal{S}_2'$ and $\mathcal{S}_3'$ be the involved subsystems of $\mathcal{S}_2$ and $\mathcal{S}_3$, respectively. Since $\mathcal{S}_2$ is functionally equivalent to $\mathcal{S}_3'$, $\mathcal{S}_2' \sqsubseteq \mathcal{S}_2$ and $g^-(\mathcal{L}^-(\mathcal{S}_2')) \subseteq g^+(\mathcal{L}^+(\mathcal{S}_2'))$, there is a subsystem $\mathcal{S} \sqsubseteq \mathcal{S}_3'$ such that $\mathcal{S}_2' \sim \mathcal{S}$ by Proposition 3.22. Thus the system $\mathcal{S}$ is also a subsystem of $\mathcal{S}_3$ by Proposition 3.6, as $\mathcal{S}_3'$ is. Moreover, the system $\mathcal{S}_1$ functionally equivalent to $\mathcal{S}_2'$. It follows that $\mathcal{S}_1$ is functionally equivalent to $\mathcal{S}$ by Proposition 3.21. Hence $\mathcal{S}_1 \sim \mathcal{S}$ via the composite l-morphism $f \circ g$ and $\mathcal{S} \sqsubseteq \mathcal{S}_3$ so that $\mathcal{S}_1 \mapsto_{f\circ g} \mathcal{S}_3$ holds. □

3.4 Translation Functions

We are now in the position to apply the methodology developed in Sections 3.1, 3.2 and 3.3 to study the relationships of non-monotonic systems $\mathfrak{C}_P$, $\mathfrak{D}_D$ and $\mathcal{A}_\Sigma$ introduced in Section 2.3. Recall the sets of parameters in these systems: P is a set of atoms to be minimized, D is a set of defaults and Σ is an autoepistemic theory. Basically, we are looking for syntactic *translation functions* Tr that map a set of parameters X to $\mathrm{Tr}(X)$ such that an embedding f between the associated non-monotonic systems $\mathcal{N}_X$ and $\mathcal{N}_{\mathrm{Tr}(X)}$ is established in the sense of Definition 3.24, i.e. $\mathcal{N}_X \mapsto_f \mathcal{N}_{\mathrm{Tr}(X)}$. We emphasize that the translation $\mathrm{Tr}(X)$ and the l-morphism f are constant in this relationship. In this way, the syntactic translation function Tr preserves the non-monotonic system $\mathcal{N}_X$ in a *modular* way [42]: the set of theorems $f^+(\mathcal{N}_X(T, N))$ is obtained as $\mathcal{N}_{\mathrm{Tr}(X)}(f^+(T), f^-(N)) \cap f^+(\mathcal{L}^+(\mathcal{N}_\Sigma))$ for all $T \subseteq \mathcal{L}^+(\mathcal{N}_X)$ and $N \subseteq \mathcal{L}^-(\mathcal{N}_X)$. The conditions on which a translation function Tr preserves a particular semantics of non-monotonic systems are analyzed in Section 6.

What follows is the study of the interconnections of non-monotonic logics introduced in Section 2.3. Our framework of embeddings is applied in the following way. We introduce translation functions that are mappings between syntactic constructs of these logics. By identifying suitable subsystems and functional equivalences, we establish embeddings between non-monotonic systems associated with a non-monotonic theory and its translation. Minimal model systems are embedded in default systems in Section 3.4.1. Embeddings between default systems and autoepistemic systems are considered in Sections 3.4.2 and 3.4.3. Also compositions of translation functions are addressed.

3.4.1 Embedding Circumscription in Default Logic

We begin with circumscription and consider translating a set of atoms $P \subseteq \mathcal{A}$ to be minimized into a set of defaults. As discussed in Section 2.3, the idea behind minimal model reasoning is to set false atomic propositions in $P \subseteq \mathcal{A}$ as long as the theory remains consistent. As proposed by Etherington [26], this principle is naturally expressed by a default rule $\frac{:\neg\phi}{\neg\phi}$ for an atomic proposition $\phi \in P$. This leads to the following translation function Tr_{E}.

Definition 3.26 *Let $P \subseteq \mathcal{A}$ be a set of atoms to be minimized. Define the translation $\mathrm{Tr}_{\mathrm{E}}(P)$ of P as the set of defaults $\{\frac{:\neg\phi}{\neg\phi} \mid \phi \in P\}$.*

It is time to apply our notion of embeddings. In Theorem 3.27 we establish a very detailed correspondence for the associated non-monotonic systems.

Theorem 3.27 *Let $P \subseteq \mathcal{A}$ be a set of atoms to be minimized. Define functions $i^+ : \mathcal{L} \to \mathcal{L}$ and $f^- : \mathcal{L} \to \neg\neg\mathcal{L}$ by setting $i^+(\phi) = \phi$ and $f^-(\phi) = \neg\neg\phi$ for all $\phi \in \mathcal{L}$. Then it holds for $f = \langle i^+, f^- \rangle$ that $\mathfrak{C}_P \mapsto_f \mathfrak{D}_{\mathrm{Tr}_{\mathrm{E}}(P)}$.*

Proof. The set of default rules $D = \mathrm{Tr}_{\mathrm{E}}(P)$ is based on the propositional language $\mathcal{L}$ induced by $\mathcal{A}$. We employ the subsystem $\mathfrak{D}_D^{\neg\neg\mathcal{L}}$ in the proof. Recall that the languages of $\mathfrak{C}_P$ and $\mathfrak{D}_D^{\neg\neg\mathcal{L}}$ are $\langle \mathcal{L}, \mathcal{L} \rangle$ and $\langle \mathcal{L}, \neg\neg\mathcal{L} \rangle$, respectively. By Proposition 3.11, the system $\mathfrak{D}_D^{\neg\neg\mathcal{L}}$ is indeed a subsystem of $\mathfrak{D}_D$, since $\neg\mathrm{J}(D) = \neg\neg P$ is contained in $\neg\neg\mathcal{L}$. It is clear that $\langle i^+, f^- \rangle$ is an l-morphism from $\mathfrak{C}_P$ to $\mathfrak{D}_D^{\neg\neg\mathcal{L}}$ since $f^-(\mathcal{L}^-(\mathfrak{C}_P)) = f^-(\mathcal{L}) = \neg\neg\mathcal{L}$ is contained in $i^+(\mathcal{L}^+(\mathfrak{C}_P)) = i^+(\mathcal{L}) = \mathcal{L}$. Then let $T \subseteq \mathcal{L}^+(\mathfrak{C}_P) = \mathcal{L}$ and $N \subseteq \mathcal{L}^-(\mathfrak{C}_P) = \mathcal{L}$. It follows that $f^-(N) = \neg\neg N$ so that the reduction

$$
\begin{aligned}
D_{f^-(N)} &= \{\tfrac{\top}{\neg\phi} \mid \tfrac{:\neg\phi}{\neg\phi} \in D \text{ and } \neg\neg\phi \in \neg\neg N\} \\
&= \{\tfrac{\top}{\neg\phi} \mid \phi \in P \text{ and } \phi \in N\} \;=\; \{\tfrac{\top}{\neg\phi} \mid \neg\phi \in \mathrm{Neg}_P(N)\}.
\end{aligned}
$$

Thus $\mathrm{Cn}(T \cup \mathrm{Neg}_P(N)) = \mathrm{Cn}^{D_{f^-(N)}}(T)$ for any $T \subseteq \mathcal{L}^+(\mathfrak{C}_P) = \mathcal{L}$ so that

$$
\begin{array}{rcll}
i^+(\mathfrak{C}_P(T,N)) &=& i^+(\mathrm{Cn}(T \cup \mathrm{Neg}_P(N))) & \text{Definition of } \mathfrak{C}_P \\
&=& i^+(\mathrm{Cn}^{D_{f^-(N)}}(T)) & \text{As shown above} \\
&=& \mathrm{Cn}^{D_{f^-(N)}}(i^+(T)) & i^+ \text{ is the identity} \\
&=& \mathfrak{D}_D^{\neg\neg\mathcal{L}}(i^+(T), f^-(N)). & \text{Definition of } \mathfrak{D}_D^{\neg\neg\mathcal{L}}
\end{array}
$$

In addition, the l-morphism $\langle i^+, f^- \rangle$ is a bijection between the languages $\mathcal{L}(\mathfrak{C}_P)$ and $\mathcal{L}(\mathfrak{D}_D^{\neg\neg\mathcal{L}})$ and the functions i^+ and f^- map a sentence $\phi \in \mathcal{L}^-(\mathfrak{C}_P) = \mathcal{L}$ to $\mathfrak{D}_D^{\neg\neg\mathcal{L}}$-equivalent sentences ϕ and $\neg\neg\phi$. So $\mathfrak{C}_P$ and $\mathfrak{D}_D^{\neg\neg\mathcal{L}}$ are functionally equivalent via $\langle i^+, f^- \rangle$. Since $\mathfrak{D}_D^{\neg\neg\mathcal{L}} \sqsubseteq \mathfrak{D}_D$ holds, the l-morphism $f = \langle i^+, f^- \rangle$ embeds $\mathfrak{C}_P$ in $\mathfrak{D}_D$, i.e. $\mathfrak{C}_P \mapsto_f \mathfrak{D}_D$ holds. $\square$

Finding a translation that would lead to an embedding in the converse direction is impossible. As discussed in Section 2.4 already, this is due to the difficulty of encoding inference rules as propositional sentences, as implied by Imielinski's results [42]. Proposition 3.28 demonstrates this phenomenon in terms of monotonic systems. The same arguments apply to translations of a set of defaults D into a background theory Σ (to be fixed as theorems) and a set of atoms P to be minimized: $\mathfrak{D}_D \mapsto_i \mathfrak{C}_P[\Sigma]$ cannot be established.

Proposition 3.28 *There is no way of translating a set of inference rules R into a fixed propositional theory Σ such that for all $T \subseteq \mathcal{L}$,*

$$\mathrm{Cn}^R(T) = \mathrm{Cn}(\Sigma \cup T).$$

Proof. Assume there is a translation Σ of $R = \{\frac{\mathrm{a}}{\mathrm{b}}\}$ such that the above equality holds for all $T \subseteq \mathcal{L}$. Then consider $T = \emptyset$ for which $\mathrm{Cn}^R(T) = \mathrm{Cn}(\emptyset)$. Since $\mathrm{a} \to \mathrm{b} \notin \mathrm{Cn}(\emptyset)$ and $T = \emptyset$, it follows that $\mathrm{a} \to \mathrm{b} \notin \mathrm{Cn}(\Sigma)$ by the above equality. So there is a model $\mathcal{M}$ of Σ such that $\mathcal{M} \models \mathrm{a}$ and $\mathcal{M} \not\models \mathrm{b}$. Then let $T' = \{\mathrm{a}\}$. It is clear that $\mathcal{M}$ is a model of $\Sigma \cup T'$ such that $\mathcal{M} \not\models \mathrm{b}$. This implies that $\mathrm{b} \notin \mathrm{Cn}(\Sigma \cup T')$. By the above equality, we obtain $\mathrm{b} \notin \mathrm{Cn}^R(T')$. But this is a contradiction, since clearly $\mathrm{b} \in \mathrm{Cn}^R(T')$ holds. □

3.4.2 Embedding Default Logic in Autoepistemic Logic

For the next, we consider a way of translating default theories into autoepistemic theories. The key question is how a default rule $\frac{\alpha:\beta_1,\ldots,\beta_n}{\gamma}$ should be expressed on the autoepistemic side. This is where the notions of belief introduced in Section 2.3 become useful. Given a default rule of the form above, the derivability of prerequisite α is naturally expressed as a belief in α whereas the consistency of a justification β is expressed as a disbelief in $\neg\beta$. It should be mentioned that this is exactly the idea behind Konolige's translation [52, 53] for conventional autoepistemic theories. The difference is that we represent disbeliefs in terms of $\mathbf{D}$ rather than $\neg\mathbf{B}$. A later comparison reveals how this is a crucial step to embed default logic in autoepistemic logic where positive introspection is defined in terms of the necessitation rule. Definition 3.29 gives the resulting translation function for default theories.

Definition 3.29 *Let D be a set of defaults. Define the translation* $\mathrm{Tr}_{\mathrm{K}}(D)$ *of D as the set of autoepistemic sentences*

$$\{\mathbf{B}\alpha \wedge \mathbf{D}\neg\beta_1 \wedge \ldots \wedge \mathbf{D}\neg\beta_n \to \gamma \mid \frac{\alpha : \beta_1, \ldots, \beta_n}{\gamma} \in D\}.$$

In order to relate the corresponding non-monotonic systems, some subsidiary results are needed. Niemelä [79] has proved Lemma 3.30 for autoepistemic theories with a single modality **B** for beliefs. We provide here the obvious generalization for theories with an additional modality **D** for disbeliefs.

Lemma 3.30 *Let Σ be a set of sentences of the form*

$$\mathbf{B}\phi_1 \wedge ... \wedge \mathbf{B}\phi_n \wedge \mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \rightarrow \chi$$

where $\chi \in \mathcal{L}$ and let Λ be a set of **BD***-atoms. Then for every $\phi \in \mathcal{L}$,*

$$\Sigma \cup \Lambda \models \phi \textit{ if and only if } \mathrm{Hds}(\Sigma, \Lambda) \models \phi$$

where $$\mathrm{Hds}(\Sigma, \Lambda) = \{\chi \,|\, (\mathbf{B}\phi_1 \wedge ... \wedge \mathbf{B}\phi_n \wedge \mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \rightarrow \chi) \in \Sigma \textit{ and } \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\} \cup \{\mathbf{D}\psi_1, ..., \mathbf{D}\psi_m\} \subseteq \Lambda \}.$$

Proof. Let $\phi \in \mathcal{L}$. ($\Rightarrow$) We prove that $\mathrm{Hds}(\Sigma, \Lambda) \not\models \phi$ implies $\Sigma \cup \Lambda \not\models \phi$. So let $\mathrm{Hds}(\Sigma, \Lambda) \not\models \phi$ hold. Then there is a purely propositional model $\mathcal{M} \subseteq \mathcal{A}$ such that $\mathcal{M} \models \mathrm{Hds}(\Sigma, \Lambda)$ and $\mathcal{M} \not\models \phi$. Let $\mathcal{M}'$ be the autoepistemic model $\mathcal{M} \cup \Lambda$. Clearly, $\mathcal{M}' \models \Lambda$. Let us then assume that $\mathcal{M}' \not\models \Sigma$, i.e. there is a sentence $(\mathbf{B}\phi_1 \wedge ... \wedge \mathbf{B}\phi_n \wedge \mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \rightarrow \chi) \in \Sigma$ which is false in $\mathcal{M}'$. It follows that $\mathcal{M}' \models \mathbf{B}\phi_1 \wedge ... \wedge \phi_n \wedge \mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m$ but $\mathcal{M}' \not\models \chi$. Consequently $\{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\} \cup \{\mathbf{D}\psi_1, ..., \mathbf{D}\psi_m\} \subseteq \Lambda$ and $\chi \in \mathrm{Hds}(\Sigma, \Lambda)$. Thus $\mathcal{M} \models \mathrm{Hds}(\Sigma, \Lambda)$ implies $\mathcal{M} \models \chi$. Since $\chi \in \mathcal{L}$ and $\mathcal{M}$ and $\mathcal{M}'$ coincide on propositional atoms, we obtain $\mathcal{M}' \models \chi$. But this is a contradiction. Hence, $\mathcal{M}' \models \Sigma$. Because $\phi \in \mathcal{L}$, the models $\mathcal{M}$ and $\mathcal{M}'$ coincide on propositional atoms and $\mathcal{M} \not\models \phi$, we have that $\mathcal{M}' \not\models \phi$. It follows that $\Sigma \cup \Lambda \not\models \phi$.

($\Leftarrow$) Let $\Sigma \cup \Lambda \not\models \phi$ hold. Then there is an autoepistemic model $\mathcal{M}$ such that $\mathcal{M} \models \Sigma \cup \Lambda$ but $\mathcal{M} \not\models \phi$. Let $\mathcal{M}'$ be the purely propositional model $\mathcal{M} \cap \mathcal{A}$. It is clear that $\mathcal{M}' \not\models \phi$ holds as $\phi \in \mathcal{L}$. Let us then assume that $\mathcal{M}' \not\models \mathrm{Hds}(\Sigma, \Lambda)$, i.e. there is a sentence $\chi \in \mathrm{Hds}(\Sigma, \Lambda)$ such that $\mathcal{M}' \not\models \chi$. This is possible only if there is a sentence $(\mathbf{B}\phi_1 \wedge ... \wedge \mathbf{B}\phi_n \wedge \mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \rightarrow \chi) \in \Sigma$ such that $\{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\} \cup \{\mathbf{D}\psi_1, ..., \mathbf{D}\psi_m\} \subseteq \Lambda$. Because $\mathcal{M} \models \Lambda$ it follows that $\mathbf{B}\phi_1 \wedge ... \wedge \mathbf{B}\phi_n \wedge \mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m$ is true in $\mathcal{M}$. On the other hand, it holds that $\mathcal{M} \not\models \chi$, because $\mathcal{M}' \not\models \chi$ and $\chi \in \mathcal{L}$. Thus $\mathbf{B}\phi_1 \wedge ... \wedge \phi_n \wedge \mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \rightarrow \chi$ is false in $\mathcal{M}$, i.e. $\mathcal{M} \not\models \Sigma$. But this is a contradiction. Therefore $\mathcal{M}' \models \mathrm{Hds}(\Sigma, \Lambda)$ and $\mathrm{Hds}(\Sigma, \Lambda) \not\models \phi$ follows. □

Recall that we characterize the closure $\mathrm{Cn}^R(T)$ for a set of inference rules R and a propositional theory T using a notion of R-proofs in Theorem 2.21. This characterization is applied in the proof of Proposition 3.31 as follows.

Proposition 3.31 *Let D be a set of defaults in $\mathcal{L}$, $T \subseteq \mathcal{L}$, $N \subseteq \mathcal{L}$ and Σ the translation $\mathrm{Tr_K}(D)$. Then $\mathrm{Cn}^{D_N}(T) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N) \cap \mathcal{L}$.*

Proof. ($\subseteq$) We prove for all $\phi \in \mathcal{L}$ that $T \vdash_{D_N} \phi$ implies $\phi \in \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$ by induction on the lengths of D_N-proofs. In the base case, the sentence ϕ is D_N-provable in 0 steps so that $T \models \phi$ holds. Since $T \subseteq \Sigma \cup T$ by definition, it follows that $\Sigma \cup T \cup \mathbf{D}N \models \phi$ by the monotonicity of $\models$ and ϕ is $\mathbf{B}$-provable from $\Sigma \cup T \cup \mathbf{D}N$ in 0 steps. Thus $\phi \in \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$.

Inductions step follows. Now there is an D_N-proof $\gamma_1, ..., \gamma_n$ of ϕ from T such that $T \cup \{\gamma_1, ..., \gamma_n\} \models \phi$ and for each γ_i there is an inference rule $\frac{\alpha_i}{\gamma_i} \in D_N$ such that $T \cup \{\gamma_1, ..., \gamma_{i-1}\} \models \alpha_i$. Now each α_i is D_N-provable from T in less than n steps so that $\{\alpha_1, ..., \alpha_n\} \subseteq \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$ follows by the inductive hypothesis. Since the theory $\mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$ is closed with respect to the necessitation rule, also $\{\mathbf{B}\alpha_1, ..., \mathbf{B}\alpha_n\} \subseteq \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$. Then consider any γ_i. Because $\frac{\alpha_i}{\gamma_i} \in D_N$, there is a default rule $\frac{\alpha_i : \beta_1, ..., \beta_m}{\gamma_i} \in D$ such that $\{\neg\beta_1, ..., \neg\beta_m\} \subseteq N$. So there is an implication $\mathbf{B}\alpha_i \wedge \mathbf{D}\neg\beta_1 \wedge ... \wedge \mathbf{D}\neg\beta_m \rightarrow \gamma_i$ in the translation Σ such that $\{\mathbf{D}\neg\beta_1, ..., \mathbf{D}\neg\beta_m\} \subseteq \mathbf{D}N$. Since $\mathbf{B}\alpha_i \in \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$ and $\mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$ is propositionally closed, it follows that $\gamma_i \in \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$. Thus $\{\gamma_1, ..., \gamma_n\} \subseteq \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$. Since also $T \subseteq \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$ and $T \cup \{\gamma_1, ..., \gamma_n\} \models \phi$ holds by the properties of the D_N-proof $\gamma_1, ..., \gamma_n$, it follows that $\phi \in \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$.

($\supseteq$) Let us then prove by induction on the lengths of $\mathbf{B}$-proofs that $\phi \in \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N) \cap \mathcal{L}$ implies $\phi \in \mathrm{Cn}^{D_N}(T)$. Consider the base case. Now $\phi \in \mathcal{L}$ is $\mathbf{B}$-provable from $\Sigma \cup \mathbf{D}N$ in 0 steps, i.e. $\Sigma \cup T \cup \mathbf{D}N \models \phi$ holds. Note that $\mathrm{Hds}(\Sigma \cup T, \mathbf{D}N) = T$[16] so that $T \models \phi$ follows by Lemma 3.30. Since $\mathrm{Cn}^{D_N}(T)$ is propositionally closed and $\phi \in \mathcal{L}$, it follows that $\phi \in \mathrm{Cn}^{D_N}(T)$. Induction step follows. Let $\phi \in \mathcal{L}$ have a $\mathbf{B}$-proof $\mathbf{B}\alpha_1, ..., \mathbf{B}\alpha_n$ of length n from $\Sigma \cup T \cup \mathbf{D}N$. By Lemma 3.16 we may assume that $\{\mathbf{B}\alpha_1, ..., \mathbf{B}\alpha_n\}$ is a subset of $\mathrm{At}^*_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N \cup \{\phi\}) = \{\mathbf{B}\alpha \mid \frac{\alpha : \beta_1, ..., \beta_n}{\gamma} \in D\}$. Now the reduction $\mathrm{Hds}(\Sigma \cup T, \{\mathbf{B}\alpha_1, ..., \mathbf{B}\alpha_n\} \cup \mathbf{D}N)$ equals to $T \cup \Gamma$ where

$$\Gamma = \{\gamma \mid \frac{\alpha}{\gamma} \in D_N \text{ and } \alpha \in \{\alpha_1, ..., \alpha_n\}\}.$$

By Lemma 3.30 we know that $\Sigma \cup T \cup \mathbf{D}N \cup \{\mathbf{B}\alpha_1, ..., \mathbf{B}\alpha_n\} \models \phi$ implies $T \cup \Gamma \models \phi$. Now each $\alpha_i \in \mathcal{L}$ is $\mathbf{B}$-provable from $\Sigma \cup T \cup \mathbf{D}N$ in less than n steps. Thus $\{\alpha_1, ..., \alpha_n\} \subseteq \mathrm{Cn}^{D_N}(T)$ by the inductive hypothesis. Consequently, it follows that $\Gamma \subseteq \mathrm{Cn}^{D_N}(T)$ and $\phi \in \mathrm{Cn}^{D_N}(T)$, since the theory $\mathrm{Cn}^{D_N}(T)$ is propositionally closed. $\square$

We may now prove that the non-monotonic system $\mathfrak{D}_D$ associated with a set of defaults D is realized on the autoepistemic side via the translation $\mathrm{Tr}_{\mathrm{K}}(D)$.

Theorem 3.32 *Let D be a set of defaults in $\mathcal{L}$ and $i = \langle i^+, i^- \rangle$ the identity l-morphism for the language $\langle \mathcal{L}, \mathcal{L} \rangle$. Then $\mathfrak{D}_D \mapsto_i \boldsymbol{\mathcal{A}}_{\mathrm{Tr}_{\mathrm{K}}(D)}$.*

[16] Recall that even "prerequisite-free" defaults have $\top$ as prerequisite.

Proof. For a set of defaults D based on a propositional language $\mathcal{L}$, the language $\mathcal{L}(\mathfrak{D}_D)$ is $\langle\mathcal{L},\mathcal{L}\rangle$. Let Σ be the translation $\mathrm{Tr}_{\mathrm{K}}(D)$ and $\mathcal{L}_{\mathrm{BD}}$ the corresponding autoepistemic language satisfying $\mathcal{L}\subseteq\mathcal{L}_{\mathrm{BD}}$. We will use the system $\boldsymbol{\mathcal{A}}_{\Sigma}^{\mathbf{B}\mathcal{L}\cup\mathbf{D}\mathcal{L}}$ in the proof. By Proposition 3.17 this is indeed a subsystem of $\boldsymbol{\mathcal{A}}_{\Sigma}$, since $\mathrm{At}^{*}_{\mathbf{BD}}(\Sigma\cup\mathbf{B}\mathcal{L}\cup\mathbf{D}\mathcal{L})=\mathbf{B}\mathcal{L}\cup\mathbf{D}\mathcal{L}$ and $\mathrm{At}^{*}_{\mathbf{D}}(\Sigma\cup\mathbf{B}\mathcal{L}\cup\mathbf{D}\mathcal{L})=\mathbf{D}\mathcal{L}$ implying that that $\mathcal{L}(\boldsymbol{\mathcal{A}}_{\Sigma}^{\mathbf{B}\mathcal{L}\cup\mathbf{D}\mathcal{L}})$ is $\langle\mathcal{L},\mathcal{L}\rangle$. Then let $T\subseteq\mathcal{L}$ and $N\subseteq\mathcal{L}$.

Now	$i^{+}(\mathfrak{D}_D(T,N))$	
$=$	$\mathfrak{D}_D(T,N)$	i^{+} is the identity
$=$	$\mathrm{Cn}^{D_N}(T)$	Definition of $\mathfrak{D}_D$
$=$	$\mathrm{Cn}_{\mathbf{B}}(\Sigma\cup T\cup\mathbf{D}N)\cap\mathcal{L}$	Proposition 3.31
$=$	$\mathrm{Cn}_{\mathbf{B}}(\Sigma\cup i^{+}(T)\cup\mathbf{D}(i^{-}(N)))\cap\mathcal{L}$	i is the identity
$=$	$\boldsymbol{\mathcal{A}}_{\Sigma}^{\mathbf{B}\mathcal{L}\cup\mathbf{D}\mathcal{L}}(i^{+}(T),i^{-}(N))$.	Definition of $\boldsymbol{\mathcal{A}}_{\Sigma}^{\mathbf{B}\mathcal{L}\cup\mathbf{D}\mathcal{L}}$

It is clear that i is a bijection and an l-morphism that maps a sentence $\phi\in\mathcal{L}$ into $\boldsymbol{\mathcal{A}}_{\Sigma}^{\mathbf{B}\mathcal{L}\cup\mathbf{D}\mathcal{L}}$-equivalent sentences $i^{+}(\phi)=\phi$ and $i^{-}(\phi)=\phi$. Thus $\mathfrak{D}_D\sim\boldsymbol{\mathcal{A}}_{\Sigma}^{\mathbf{B}\mathcal{L}\cup\mathbf{D}\mathcal{L}}$ via i by Definition 3.20. Since $\boldsymbol{\mathcal{A}}_{\Sigma}^{\mathbf{B}\mathcal{L}\cup\mathbf{D}\mathcal{L}}\sqsubseteq\boldsymbol{\mathcal{A}}_{\Sigma}$, the l-morphism $i=\langle i^{+},i^{-}\rangle$ embeds $\mathfrak{D}_D$ in $\boldsymbol{\mathcal{A}}_{\Sigma}$, i.e. $\mathfrak{D}_D\mapsto_i\boldsymbol{\mathcal{A}}_{\Sigma}$ holds. □

It is illustrated in Example 3.33 that Theorem 3.32 ceases to hold if Konolige's original translation and unimodal autoepistemic logic are considered. By the results of Marek and Truszczyński [66, 112] more complex translations are required to capture defaults in unimodal autoepistemic logic. We take the simplification obtained by the modality **D** for disbeliefs as one of the advantages of the bimodal autoepistemic logic proposed in this work. In this way, all the knowledge representation capabilities of default logic become available on the autoepistemic side through the straightforward translation Tr_{K}.

Example 3.33 Recall the inference operator $\boldsymbol{\mathcal{A}}_{\Sigma}$ associated with a conventional autoepistemic theory $\Sigma\subseteq\mathcal{L}_{\mathrm{B}}$ (under Marek and Truszczyński's iterative expansions). In Section 2.3, this was defined by $\boldsymbol{\mathcal{A}}_{\Sigma}(T,N)=\mathrm{Cn}_{\mathbf{B}}(\Sigma\cup T\cup\neg\mathbf{B}N)$ for all $T\subseteq\mathcal{L}_{\mathrm{B}}$ and $N\subseteq\mathcal{L}_{\mathrm{B}}$. Let us then consider a set of defaults $D=\{\frac{\neg\neg\mathrm{p}:}{\mathrm{p}},\frac{:\neg\mathrm{p}}{\mathrm{p}}\}$ due to Marek and Truszczyński and the associated inference operator $\mathfrak{D}_D$ defined by $\mathfrak{D}_D(T,N)=\mathrm{Cn}^{D_N}(T)$ for all $T\subseteq\mathcal{L}$ and $N\subseteq\mathcal{L}$. Then let $T=\emptyset$ and $N=\emptyset$. Now $\mathfrak{D}_D(T,N)=\mathrm{Cn}^{\{\frac{\neg\neg\mathrm{p}}{\mathrm{p}}\}}(\emptyset)=\mathrm{Cn}(\emptyset)$. However, the translation of D under Konolige's original translation [52] is $\Sigma=\{\mathbf{B}\neg\neg\mathrm{p}\rightarrow\mathrm{p},\mathbf{B}\top\wedge\neg\mathbf{B}\neg\neg\mathrm{p}\rightarrow\mathrm{p}\}$. In analogy to Theorem 3.32, we use the identity l-morphism $i=\langle i^{+},i^{-}\rangle$ to map T and N to the autoepistemic side. Now $\boldsymbol{\mathcal{A}}_{\Sigma}^{\mathbf{B}\mathcal{L}}(i^{+}(T),i^{-}(N))=\boldsymbol{\mathcal{A}}_{\Sigma}(T,N)\cap\mathcal{L}=\mathrm{Cn}_{\mathbf{B}}(\Sigma)\cap\mathcal{L}=\mathrm{Cn}(\{\mathrm{p}\})$ because p is directly **B**-provable from Σ. Thus i does not embed $\mathfrak{D}_D$ in $\boldsymbol{\mathcal{A}}_{\Sigma}$.

In our approach, the translation of D is $\Sigma'=\{\mathbf{B}\neg\neg\mathrm{p}\rightarrow\mathrm{p},\mathbf{B}\top\wedge\mathbf{D}\neg\neg\mathrm{p}\rightarrow\mathrm{p}\}$. Note that p is not **B**-provable from Σ' mainly because $\mathbf{B}\neg\neg\mathrm{p}\vee\mathbf{D}\neg\neg\mathrm{p}$ is not. In contrast to this, the corresponding sentence $\mathbf{B}\neg\neg\mathrm{p}\vee\neg\mathbf{B}\neg\neg\mathrm{p}$ in $\mathcal{L}_{\mathrm{B}}$ is a propositional tautology and thus **B**-provable from the translation Σ. □

The above problem with Konolige's translation has lead to the introduction of other translations. One of the successful ones is due to Truszczyński [113]. He translates a default rule $\frac{\alpha:\beta_1,\ldots,\beta_n}{\gamma}$ into an autoepistemic sentence $\mathbf{B}\alpha \wedge \mathbf{B}\neg\mathbf{B}\neg\beta_1 \wedge \ldots \wedge \mathbf{B}\neg\mathbf{B}\neg\beta_n \rightarrow \alpha$. Unfortunately, even this translation is unsatisfactory from the point of view of our framework, i.e. the non-monotonic system $\mathfrak{D}_D$ is not always realized on the autoepistemic side. This is illustrated in Example 3.34 where a set of defaults proposed by Niemelä [78] is used.

Example 3.34 Consider a set of defaults $D = \{\frac{\neg\mathrm{b}:}{\neg\mathrm{a}}, \frac{:\mathrm{b}}{\neg\mathrm{b}}\}$. For $T = \{\mathrm{a}\}$ and $N = \emptyset$, we obtain $\mathfrak{D}_D(T, N) = \mathrm{Cn}^{\{\frac{\neg\mathrm{b}}{\neg\mathrm{a}}\}}(\{\mathrm{a}\}) = \mathrm{Cn}(\{\mathrm{a}\})$. Truszczyński's translation function yields an autoepistemic theory $\Sigma = \{\mathbf{B}\neg\mathrm{b} \rightarrow \neg\mathrm{a}, \mathbf{B}\top \wedge \mathbf{B}\neg\mathbf{B}\neg\mathrm{b} \rightarrow \neg\mathrm{b}\}$. It follows that $\mathcal{A}_{\Sigma}^{\mathbf{B}\mathcal{L}}(i^+(T), i^-(N)) = \mathcal{A}_{\Sigma}(T, N) \cap \mathcal{L} = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T) \cap \mathcal{L} = \mathcal{L}$, because $\perp$ is directly $\mathbf{B}$-provable from $\Sigma \cup T$. Therefore, the identity l-morphism i does not embed $\mathfrak{D}_D$ in $\mathcal{A}_{\Sigma}$. $\square$

Gottlob [39] presents a surprising translation function for conventional autoepistemic theories to capture iterative expansions of $\Sigma \subseteq \mathcal{L}_{\mathrm{B}}$ with stable expansions of the translation $\mathrm{Tr}_{\mathrm{G}}(\Sigma)$. This effect is obtained by augmenting Σ with a sentence $\mathbf{B}G(\Sigma)$ where the subsentence $G(\Sigma)$ captures the groundedness condition of iterative expansions. The existence of $G(\Sigma)$ is guaranteed by a complexity result that the problem whether $\phi \in \mathcal{L}_{\mathrm{B}}$ is $\mathbf{B}$-provable from $\Sigma \subseteq \mathcal{L}_{\mathrm{B}}$ is **co-NP**-complete and thus polynomially reducible to the validity problem of propositional logic. An interesting feature of the translation function Tr_{G} is that it is not a *modular* one. Roughly speaking, modularity means that local changes in Σ cause only local changes in the translation $\mathrm{Tr}_{\mathrm{G}}(\Sigma)$ such that the one-to-one correspondence of expansions is retained. In fact, Gottlob shows that a modular translation is not possible and non-modularity implies that $G(\Sigma)$ may change unexpectedly whenever Σ changes. By composing Tr_{G} with the translation function proposed by Truszczyński [113], Gottlob obtains a non-modular translation of a default theory $\langle D, T\rangle$ into an autoepistemic theory such that extensions of $\langle D, T\rangle$ are captured with Moore's stable expansions. Schwarz [101] proposes a direct (but necessarily non-modular) translation function for the same purpose. Gottlob [39] considers also translations in the converse direction, i.e. to capture Moore's stable expansions with default extensions. This is to be addressed in the end of Section 3.4.3.

To demonstrate compositions of translations functions at the level of inference operators, we form a composition of the translation function Tr_{E} mapping a set of atoms $P \subseteq \mathcal{A}$ under minimization to a set of defaults and the translation function Tr_{K} mapping a set of defaults to a set of autoepistemic sentences. In Definition 3.35, we have formed the composite translation function $\mathrm{Tr}_{\mathrm{EK}}$ that maps to a set of atoms $P \subseteq \mathcal{A}$ under minimization to a set of autoepistemic sentences $\mathrm{Tr}_{\mathrm{EK}}(P)$. Corollary 3.36 gives the resulting embedding for the non-monotonic systems involved in the composition.

Definition 3.35 *Let $P \subseteq \mathcal{A}$ be a set of atoms to be minimized. Define the translation* $\mathrm{Tr}_{\mathrm{EK}}(P)$ *of* P *as the set of autoepistemic sentences*

$$\{\mathbf{D}\neg\neg\phi \rightarrow \neg\phi \,|\, \phi \in P\}^{17}.$$

Corollary 3.36 *Let $P \subseteq \mathcal{A}$ be a set of atoms to be minimized and f the l-morphism introduced in Theorem 3.27. Then* $\mathfrak{C}_P \mapsto_f \boldsymbol{\mathcal{A}}_{\mathrm{Tr}_{\mathrm{EK}}(P)}$.

Proof. Let D be the set of defaults $\mathrm{Tr}_{\mathrm{E}}(P)$, so that $\Sigma = \mathrm{Tr}_{\mathrm{EK}}(P) = \mathrm{Tr}_{\mathrm{K}}(D)$. Recall the subsystem $\mathfrak{D}_D^{\neg\neg\mathcal{L}}$ used in Theorem 3.27. The language $\mathcal{L}(\mathfrak{D}_D^{\neg\neg\mathcal{L}}) = \langle \mathcal{L}, \neg\neg\mathcal{L}\rangle$. This is mapped by the identity l-morphism $i = \langle i^+, i^-\rangle$ used in Theorem 3.32 to a reasonable sublanguage of $\boldsymbol{\mathcal{A}}_\Sigma$, i.e $i^-(\neg\neg\mathcal{L}) = \neg\neg\mathcal{L}$ is contained in $i^+(\mathcal{L}) = \mathcal{L}$. It follows by Proposition 3.25 that $\mathfrak{C}_P \mapsto_{f\circ i} \boldsymbol{\mathcal{A}}_\Sigma$. $\square$

3.4.3 Embedding Autoepistemic Logic in Default Logic

To end this section, we consider ways of translating autoepistemic theories into default theories. Since the converse translation maps default rules into autoepistemic sentences, one possibility would be a mapping from autoepistemic sentences back to default rules in spirit of Konolige's translation. A drawback of this approach is that it is by no means general: a normal form is required from autoepistemic sentences. A specific procedure [67] has to be applied to transform arbitrary autoepistemic sentences to this form. The procedure is based on **KD45**-characterization of stable expansions and it seems that this is not applicable if iterative expansions are considered. We propose a completely different approach which arises from an idea of representing autoepistemic introspection in terms default rules. Positive and negative introspection of a sentence $\phi \in \mathcal{L}_{\mathrm{BD}}$ is captured by default rules $\frac{\phi:}{\mathbf{B}\phi}$ and $\frac{:\neg\phi}{\mathbf{D}\phi}$, respectively. In particular, testing the consistency of $\neg\phi$ amounts to testing if ϕ is not derivable. If this is the case, then $\mathbf{D}\phi$ is a justified conclusion. Such a representation of autoepistemic introspection principles leads to the translation function Tr_{J} introduced Definition 3.37.

Definition 3.37 *Let $\Sigma \subseteq \mathcal{L}_{\mathrm{B}}$ be a set of autoepistemic sentences. Define the translation* $\mathrm{Tr}_{\mathrm{J}}(\Sigma)$ *of* Σ *as the set of defaults*

$$\{\frac{\phi:}{\mathbf{B}\phi} \,|\, \phi \in \mathcal{L}_{\mathrm{BD}}\} \cup \{\frac{:\neg\phi}{\mathbf{D}\phi} \,|\, \phi \in \mathcal{L}_{\mathrm{BD}}\} \cup \{\frac{\top:}{\phi} \,|\, \phi \in \Sigma\}.^{18}$$

[17] Implications of the form $\mathbf{D}\neg\neg\phi \rightarrow \neg\phi$ can be simplified to the form $\mathbf{D}\phi \rightarrow \neg\phi$ if desired.

[18] The translation $\mathrm{Tr}_{\mathrm{J}}(\Sigma)$ is an infinite set of defaults. However, if $\mathrm{At}^*_{\mathbf{BD}}(\Sigma)$ is finite, it is possible to reduce this translation to finite one by a technique presented by the author elsewhere [46]. The idea is to include the defaults $\frac{\phi:}{\mathbf{B}\phi}$ and $\frac{:\neg\phi}{\mathbf{D}\phi}$ in the translation only if $\mathbf{B}\phi \in \mathrm{At}^*_{\mathbf{BD}}(\Sigma)$ and $\mathbf{D}\phi \in \mathrm{At}^*_{\mathbf{BD}}(\Sigma)$, respectively.

The language of the translation requires a further comment. If $\mathcal{L}_{\mathrm{BD}}$ is based on a set of atomic sentences $\mathcal{A}$, the language $\mathcal{L}$ of $\mathrm{Tr}_{\mathrm{J}}(\Sigma)$ is based on the set of atomic sentences $\mathcal{A} \cup \mathbf{B}\mathcal{L}_{\mathrm{BD}} \cup \mathbf{D}\mathcal{L}_{\mathrm{BD}}$. This is how the autoepistemic principle of "treating belief atoms as atomic sentences" is taken into account in the translation. We take the languages $\mathcal{L}_{\mathrm{BD}}$ and $\mathcal{L}$ to be exactly the same languages (they contain the same sentences) so that we can do without a further translation function between these languages. The following proposition shows how the positive introspection is realized on the default side.

Proposition 3.38 *For an autoepistemic theory $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ and the set of inference rules $R = \{\frac{\psi}{\mathbf{B}\psi} \mid \psi \in \mathcal{L}_{\mathrm{BD}}\}$, the theories $\mathrm{Cn}_{\mathbf{B}}(\Sigma)$ and $\mathrm{Cn}^R(\Sigma)$ coincide.*

Proof. This becomes obvious once we have established that ϕ has a **B**-proof from Σ if and only if ϕ has an R-proof from Σ. Then the claim follows by Theorem 2.21 and its analog for **B**-proofs. ($\Rightarrow$) So let $\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n$ be a **B**-proof of ϕ from Σ. Then $\Sigma \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\} \models \phi$ and for each ϕ_i also $\Sigma \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_{i-1}\} \models \phi_i$. Let us then choose $\alpha_i = \phi_i$ and $\gamma_i = \mathbf{B}\phi_i$. The sequence $\gamma_1, ..., \gamma_n$ satisfies $\Sigma \cup \{\gamma_1, ..., \gamma_n\} \models \phi$ and there are inference rules $\frac{\alpha_1}{\gamma_1}, ..., \frac{\alpha_n}{\gamma_n}$ in R such that $\Sigma \cup \{\gamma_1, ..., \gamma_{i-1}\} \models \alpha_i$ for each α_i where $0 < i \leq n$. Thus $\gamma_1, ..., \gamma_n$ is an R-proof of ϕ from Σ. ($\Leftarrow$) Assume $\gamma_1, ..., \gamma_n$ is an R-proof of ϕ from Σ. Then $\Sigma \cup \{\gamma_1, ..., \gamma_n\} \models \phi$ holds and there are inference rules $\frac{\alpha_1}{\gamma_1}, ..., \frac{\alpha_n}{\gamma_n}$ in R such that $\Sigma \cup \{\gamma_1, ..., \gamma_{i-1}\} \models \alpha_i$ for each α_i. Recall that the rules of R are of the form $\frac{\phi}{\mathbf{B}\phi}$ where $\phi \in \mathcal{L}_{\mathrm{BD}}$. Thus each $\frac{\alpha_i}{\gamma_i}$ equals to $\frac{\phi_i}{\mathbf{B}\phi_i}$ for some $\phi_i \in \mathcal{L}_{\mathrm{BD}}$. It follows that $\Sigma \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\} \models \phi$ and for each ϕ_i also $\Sigma \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_{i-1}\} \models \phi_i$. Thus $\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n$ is a **B**-proof of ϕ from Σ. □

In Theorem 3.39, we embed a non-monotonic system associated with an autoepistemic theory $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ in a default system.

Theorem 3.39 *Let $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$. Define functions $i^+ : \mathcal{L}_{\mathrm{BD}} \to \mathcal{L}_{\mathrm{BD}}$ and $f^- : \mathcal{L}_{\mathrm{BD}} \to \neg\neg\mathcal{L}_{\mathrm{BD}}$ by setting $i^+(\phi) = \phi$ and $f^-(\phi) = \neg\neg\phi$ for all $\phi \in \mathcal{L}_{\mathrm{BD}}$. Then it holds for $f = \langle i^+, f^- \rangle$ that $\mathcal{A}_\Sigma \mapsto_f \mathfrak{D}_{\mathrm{Tr}_{\mathrm{J}}(\Sigma)}$.*

Proof. Recall that we are assuming that the languages $\mathcal{L}_{\mathrm{BD}}$ and $\mathcal{L}$ associated with the theory Σ and the set of defaults $D = \mathrm{Tr}_{\mathrm{J}}(\Sigma)$ (respectively) coincide. The system $\mathfrak{D}_D^{\neg\neg\mathcal{L}}$ is used in the proof. This is a subsystem of $\mathfrak{D}_D$ by Proposition 3.11, since $\neg\mathrm{J}(D) = \neg\neg\mathcal{L}_{\mathrm{BD}} = \neg\neg\mathcal{L}$. Let R be the set of inference rules $\{\frac{\phi}{\mathbf{B}\phi} \mid \phi \in \mathcal{L}_{\mathrm{BD}}\}$. Now $f^-(N) = \neg\neg N$ and $D_{f^-(N)}$ is

$$R \cup \{\frac{\top}{\phi} \mid \phi \in \Sigma\} \cup \{\frac{\top}{\mathbf{D}\phi} \mid \phi \in \mathcal{L}_{\mathrm{BD}} \text{ and } \neg\neg\phi \in \neg\neg N\}.$$

Thus $D_{f^-(N)} = R \cup \{\frac{\top}{\phi} \mid \phi \in \Sigma\} \cup \{\frac{\top}{\mathbf{D}\phi} \mid \phi \in N\}$. Consequently, it holds for all $T \subseteq \mathcal{L}_{\mathrm{BD}}$ and $N \subseteq \mathcal{L}_{\mathrm{BD}}$ that

$$
\begin{array}{rcll}
i^+(\mathcal{A}_\Sigma(T,N)) & = & i^+(\mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)) & \text{Definition of } \mathcal{A}_\Sigma \\
& = & \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N) & i^+ \text{ is the identity} \\
& = & \mathrm{Cn}^R(\Sigma \cup T \cup \mathbf{D}N) & \text{Proposition 3.38} \\
& = & \mathrm{Cn}^{D_{f^-(N)}}(T) & \text{See } D_{f^-(N)} \text{ above} \\
& = & \mathrm{Cn}^{D_{f^-(N)}}(i^+(T)) & i^+ \text{ is the identity} \\
& = & \mathfrak{D}_D^{\neg\neg\mathcal{L}}(i^+(T), f^-(N)). & \text{Definition of } \mathfrak{D}_D^{\neg\neg\mathcal{L}}
\end{array}
$$

Moreover, the l-morphism $\langle i^+, f^- \rangle$ is a bijection between the languages $\mathcal{L}(\mathcal{A}_\Sigma)$ and $\mathcal{L}(\mathfrak{D}_D^{\neg\neg\mathcal{L}})$ and the functions i^+ and f^- map a sentence $\phi \in \mathcal{L}^+(\mathfrak{S})$ to $\mathfrak{D}_D^{\neg\neg\mathcal{L}}$-equivalent sentences $i^+(\phi) = \phi$ and $f^-(\phi) = \neg\neg\phi$. It follows that $\mathcal{A}_\Sigma$ is functionally equivalent with $\mathfrak{D}_D^{\neg\neg\mathcal{L}}$ via $f = \langle i^+, f^- \rangle$. Since $\mathfrak{D}_D^{\neg\neg\mathcal{L}} \sqsubseteq \mathfrak{D}_D$ holds, the l-morphism f embeds $\mathcal{A}_\Sigma$ in $\mathfrak{D}_D$, i.e. $\mathcal{A}_\Sigma \mapsto_f \mathfrak{D}_D$ holds. □

The technique behind the translation function Tr_{J} sheds new light on the interconnection of default logic and autoepistemic logic and their expressibility. Gottlob [39] considers the possibility of capturing Moore's autoepistemic expansions in terms of default extensions and comes to the conclusion that such a translation is not possible. However, the translation function Tr_{J} leads to a new point of view – in contrast with Gottlob's conclusion. In the following, we restrict ourselves to the unimodal case and explain the results obtained by the author [46] in this case. A unimodal variant of the translation function Tr_{J} is able to capture iterative expansions of $\Sigma \subseteq \mathcal{L}_{\mathrm{B}}$ with default extensions of $\mathrm{Tr}_{\mathrm{J}}(\Sigma)$. Moreover, another translation function, which adds suitable instances of the schema **5** to $\Sigma \subseteq \mathcal{L}_{\mathrm{B}}$, can be used in order to capture stable expansions of Σ with the iterative expansions of the translation. Consequently, the composition of the two translations functions allows us to capture Moore's stable expansions with default extensions of the translation.

In order to explain the difference between these results, we review Gottlob's argument. Gottlob [39] takes only *faithful* translations into account, i.e. the propositional parts of stable expansions of $\Sigma \subseteq \mathcal{L}_{\mathrm{B}}$ are assumed to coincide with the extensions of the translation, and shows that there is no faithful translation, using $\Sigma = \{\mathbf{B}\mathrm{p} \to \mathrm{p}\}$ as a counter-example. We agree with Gottlob – as long as the language of the translation is assumed to be the propositional sublanguage $\mathcal{L}$ of $\mathcal{L}_{\mathrm{B}}$. If new atoms are allowed to extend $\mathcal{L}$ to $\mathcal{L}'$, the situation changes drastically. To demonstrate this subtle detail, we construct a default theory whose extensions correspond to the expansions of Σ (up to the above mentioned sublanguage $\mathcal{L}$ of $\mathcal{L}_{\mathrm{B}}$) in the following example.

Example 3.40 The stable expansions of a conventional autoepistemic theory $\Sigma = \{\mathbf{B}\mathrm{p} \to \mathrm{p}\}$ are $\Delta_1 = \{\neg\mathbf{B}\mathrm{p}, \mathbf{B}\neg\mathbf{B}\mathrm{p}, \ldots\}$ and $\Delta_2 = \{\mathbf{B}\mathrm{p}, \mathrm{p}, \neg\mathbf{B}\neg\mathbf{B}\mathrm{p}, \ldots\}$. Gottlob notes that the propositional parts of Δ_1 and Δ_2 are $\mathrm{Cn}(\emptyset)$ and $\mathrm{Cn}(\{\mathrm{p}\})$, respectively. Since $\mathrm{Cn}(\emptyset) \subseteq \mathrm{Cn}(\{\mathrm{p}\})$, there is no default theory with extensions $\mathrm{Cn}(\emptyset)$ and $\mathrm{Cn}(\{\mathrm{p}\})$[19]. Let us then apply the translation proposed by the author [46] and extend Σ by an instance $\neg\mathbf{B}\neg\mathbf{B}\phi \rightarrow \mathbf{B}\phi$ of the schema **5** for each belief atom $\mathbf{B}\phi \in \mathrm{At}^*_{\mathbf{B}}(\Sigma)$ to obtain $\Sigma' = \{\mathbf{B}\mathrm{p} \rightarrow \mathrm{p}, \neg\mathbf{B}\neg\mathbf{B}\mathrm{p} \rightarrow \mathbf{B}\mathrm{p}\}$. It is easy to see that Δ_1 and Δ_2 are the iterative expansions of Σ'. In particular, note that $\neg\mathbf{B}\mathrm{p} \notin \Delta_2$ implies that $\neg\mathbf{B}\neg\mathbf{B}\mathrm{p} \in \neg\mathbf{B}\overline{\Delta_2}$ so that $\mathbf{B}\mathrm{p}$ and p are $\mathbf{B}$-provable from $\Sigma' \cup \neg\mathbf{B}\overline{\Delta_2}$. Thus the iterative expansions of the translation Σ' capture the stable expansions of Σ. The next step is to apply the technique behind Tr_{J} (see Definition 3.37). We use an extended propositional language $\mathcal{L}'$ based on a set of atoms $\mathcal{A}' = \mathcal{A} \cup \{\mathbf{B}\mathrm{p}, \mathbf{B}\neg\mathbf{B}\mathrm{p}\}$ where $\mathcal{A} = \{\mathrm{p}\}$ is the set of atoms on which $\mathcal{L}$ is based. We may now form the default rules that simulate the introspection of the sentences p and $\neg\mathbf{B}\mathrm{p}$. So let

$$D = \{\frac{\mathrm{p}:}{\mathbf{B}\mathrm{p}}, \frac{:\neg\mathrm{p}}{\neg\mathbf{B}\mathrm{p}}, \frac{\neg\mathbf{B}\mathrm{p}:}{\mathbf{B}\neg\mathbf{B}\mathrm{p}}, \frac{:\neg\neg\mathbf{B}\mathrm{p}}{\neg\mathbf{B}\neg\mathbf{B}\mathrm{p}}\}.$$

Consequently, the extensions of the resulting default theory $\langle D, \Sigma'\rangle$ are $E_1 = \mathrm{Cn}(\Sigma' \cup \{\neg\mathbf{B}\mathrm{p}, \mathbf{B}\neg\mathbf{B}\mathrm{p}\})$ and $E_2 = \mathrm{Cn}(\Sigma' \cup \{\mathbf{B}\mathrm{p}, \mathrm{p}, \neg\mathbf{B}\neg\mathbf{B}\mathrm{p}\})$. To check that this is really the case, note that the reductions of D are

$$D_{E_1} = \{\frac{\mathrm{p}}{\mathbf{B}\mathrm{p}}, \frac{\neg\mathbf{B}\mathrm{p}}{\mathbf{B}\neg\mathbf{B}\mathrm{p}}, \frac{\top}{\neg\mathbf{B}\mathrm{p}}\} \text{ and } D_{E_2} = \{\frac{\mathrm{p}}{\mathbf{B}\mathrm{p}}, \frac{\neg\mathbf{B}\mathrm{p}}{\mathbf{B}\neg\mathbf{B}\mathrm{p}}, \frac{\top}{\neg\mathbf{B}\neg\mathbf{B}\mathrm{p}}\}.$$

It follows that the intersections $E_1 \cap \mathcal{L}$ and $E_2 \cap \mathcal{L}$ are $\mathrm{Cn}(\emptyset)$ and $\mathrm{Cn}(\{\mathrm{p}\})$, respectively. Thus stable expansions of Σ and extensions of $\langle D, \Sigma'\rangle$ coincide up to the sublanguage $\mathcal{L}$ of $\mathcal{L}_{\mathrm{B}}$ and $\mathcal{L}'$. Moreover, we want to emphasize that $E_1 \not\subseteq E_2$ (as required by the properties of extensions), although $E_1 \cap \mathcal{L}$ is contained in $E_2 \cap \mathcal{L}$. This kind of relationship between extensions became possible when we added the atoms $\mathbf{B}\mathrm{p}$ and $\mathbf{B}\neg\mathbf{B}\mathrm{p}$ to the language $\mathcal{L}$. □

Our example illustrates that the translational technique behind Tr_{J} provides the means to capture stable and iterative expansions of conventional autoepistemic theories. Note also that the set of defaults introduced in our example is finite in contrast to the set of defaults produced by Tr_{J}. In fact, the techniques needed in Example 3.40 give rise to a polynomial translation of finite autoepistemic theories. Since the translation handles effectively belief atoms that appear in Σ and Σ', it is obvious that the translation function in question is *modular* in the following way: if we extend Σ with any propositional theory $T \subseteq \mathcal{L}$, the theory $\Sigma \cup T$ gets translated into $\langle D, \Sigma' \cup T\rangle$. Existence of a polynomial and modular translation speaks for the view that the expressive power of Moore's autoepistemic logic does not exceed that of default logic.

[19] It is easy to show that extensions of a default theory cannot be subsets of each other.

4 Semantics for Non-Monotonic Systems

As already discussed in Section 2, we consider non-monotonic reasoning as a form of parametrized monotonic reasoning. This parameterization is present in our definition of non-monotonic systems as a set of non-theorems N. In Definition 2.6, we made an overall stipulation how non-theorems affect the underlying monotonic reasoning: the larger set of non-theorems N is provided as assumptions the larger set of theorems results. However, the criteria for selecting non-theorems were intentionally left open. The choice of non-theorems is the main issue of this section. We equip non-monotonic systems with a variety of *semantics* that provide principles to decide non-theorems in a justifiable manner. Once the conditions for non-theorems are clear, it is straightforward to define the corresponding set of theorems for a non-monotonic system.

Our analysis is initiated in Section 4.1 where the general notion of a semantics is introduced. At this point, we identify some basic properties for sets of non-theorems associated with forthcoming semantics such as *consistency* and *completeness*. These properties are the cornerstones of *stable semantics* employed in various contexts (see, e.g., [32, 72, 94]). Stable semantics is analyzed in Section 4.2. However, stable semantics is not satisfactory in all respects. One of the early ramifications is *stationary semantics* [89] which was first introduced in the context of logic programming as the *well-founded semantics* of normal logic programs [115]. In Section 4.3, we address stationary semantics and its relationship to stable semantics. A property to be established is that stationary semantics is considerably *weaker* semantics than stable semantics, i.e. the former provides less theorems than the latter. In this sense, stationary semantics can be considered an *approximation* of stable semantics.

Many of the results presented in this section have appeared in a restricted form in previous works related with non-monotonic reasoning and logic programming. Our contribution is a generalization of these results to our framework of non-monotonic systems. It should yet be emphasized that the analysis presented in this section is also relevant to the bimodal variant of autoepistemic logic introduced in Section 2.3. Moreover, we consider alternatives to stable and stationary semantics in Section 8 where we identify a source for the weakness of stationary semantics and its derivatives [3, 118] and propose a systematic refinement technique for stationary semantics. This technique leads to a notion of *cautious semantics* which is an attractive semantics in the *gap* between stationary and stable semantics, i.e. a stronger semantics than stationary semantics but a weaker semantics than stable semantics.

4.1 Setting up Semantic Operators

It is time to address the question what we mean with a *semantics* in the context of monotonic and non-monotonic systems introduced in Section 2. These systems have a logical language associated with them. From the logical point of view, defining the semantics of such a language amounts to defining which sentences are plausible conclusions and which are not. Historically, there are two leading paradigms for such identification: model-theoretical and proof-theoretical. The former relies on a notion of *models* to define *valid* sentences while the latter employs a notion of *proofs* to single out sentences that are provable or *theorems*. These approaches are typically compatible, because one typically persists in proof methods that are *sound* and *complete* so that a sentence is a theorem if and only if it is valid.

In Section 2, we chose the term *theorem* to denote conclusions provided by the system $\mathcal{S}$, though the term has a proof-theoretical flavor. This does not mean that we insist on a specific paradigm (say model-theoretic or proof-theoretic) to define a semantics. What we need is a notion of semantics that matches with our level of abstraction. In the monotonic case, we have already presented the solution. It is the inference operator $\mathcal{S}$ itself that provides the semantics for the language $\mathcal{L}(\mathcal{S})$ on this level of abstraction. The operator $\mathcal{S}$ defines the set of theorems $\mathcal{S}(T)$ given a set of sentences T as assumptions. Thus we adopt the view that a semantics is simply a function from $\mathbf{2}^{\mathcal{L}(\mathcal{S})}$ to $\mathbf{2}^{\mathcal{L}(\mathcal{S})}$. In light of Example 2.1, this interpretation of a semantics is well-justified for classical monotonic logics. By Definition 4.1 given below, we resort to a similar approach in the non-monotonic case.

Definition 4.1 *A semantics of a non-monotonic system $\mathcal{S}$ is a function*

$$\mathrm{SEM}_{\mathcal{S}} : \mathbf{2}^{\mathcal{L}^+(\mathcal{S})} \rightarrow \mathbf{2}^{\mathcal{L}^+(\mathcal{S})}.$$

In the sequel, our task is to compose *semantic operators* $\mathrm{SEM}_{\mathcal{S}}$ conforming to Definition 4.1 from the binary inference operators $\mathcal{S} : \mathbf{2}^{\mathcal{L}^+(\mathcal{S})} \times \mathbf{2}^{\mathcal{L}^-(\mathcal{S})} \rightarrow \mathbf{2}^{\mathcal{L}^+(\mathcal{S})}$ corresponding to non-monotonic systems. Basically, we define a semantics for a non-monotonic system $\mathcal{S}$ in three steps explained below.

1. By Definition 4.1, a semantic operator $\mathrm{SEM}_{\mathcal{S}}$ takes a set of sentences $T \subseteq \mathcal{L}^+(\mathcal{S})$ as argument. The argument T has a very implicit role in many forthcoming definitions and results and it is simply dropped out. This is possible by the result established in Proposition 2.40: the system $\mathcal{S}[T]$ is also a non-monotonic system. Thus various definitions and results formulated for a non-monotonic system $\mathcal{S}$ cover also the system $\mathcal{S}[T]$.

2. The next step is to set up the criteria for the set of non-theorems N which is the second argument for the binary inference operator $\mathcal{S}$. Here further complications typically arise: (i) certain criteria do not lead to a unique set of non-theorems N and (ii) a set of non-theorems N meeting the criteria might not exist at all. Two fundamental criteria – namely the notions of $\mathcal{S}$-consistency and $\mathcal{S}$-completeness for sets of non-theorems – are introduced in Section 4.1.1.

3. The last step is to define the resulting set of theorems $\mathrm{SEM}_{\mathcal{S}}(T)$. This is fairly simple, since each pair $\langle T, N\rangle$ is mapped to a set of theorems $\mathcal{S}(T, N)$ by $\mathcal{S}$. If there are several sets of non-theorems N meeting the given criteria, we adopt a *skeptical* approach and take the intersection of the corresponding sets of theorems $\mathcal{S}(T, N)$ as the final set of theorems $\mathrm{SEM}_{\mathcal{S}}(T)$. If there are no sets of non-theorems meeting the given criteria, then $\mathrm{SEM}_{\mathcal{S}}(T)$ is naturally the empty intersection of subsets of $\mathcal{L}^{+}(\mathcal{S})$, i.e. the whole language $\mathcal{L}^{+}(\mathcal{S})$. In this case, the set of sentences T can be understood to be inconsistent with respect to the semantics $\mathrm{SEM}_{\mathcal{S}}$.

The three steps above describe how a set of theorems T is to be mapped to a set of theorems $\mathrm{SEM}_{\mathcal{S}}(T)$ as insisted by Definition 4.1. In addition to the skeptical approach, also a *credulous* approach is available within our framework. In the credulous approach, the sets of theorems $\mathcal{S}(T, N)$ induced by sets of non-theorems N (satisfying the criteria under consideration) are of interest instead of their intersection. For instance, in diagnostic reasoning (to be addressed in Section 10.2) the credulous approach tries to find single diagnoses while skeptical approach is concerned with what all diagnoses have in common.

4.1.1 Consistency and Completeness of Sets of Non-Theorems

We have already sketched on a general level, how one is to establish a semantics for non-monotonic systems. In the following, we present two fundamental criteria for selecting non-theorems: notions of $\mathcal{S}$-*consistency* and $\mathcal{S}$-*completeness* for a set of non-theorems N. Non-theorems were earlier introduced as sentences that are not theorems – or more precisely – are not supposed to be theorems. Indeed, our specification does not require that a set of non-theorems N and the resulting set of theorems $\mathcal{S}(N)$ are disjoint. To distinguish sets of non-theorems N with this property, we resort to the notion of $\mathcal{S}$-consistency below. Proposition 2.40 allows us to incorporate a set of theorems T to this definition by fixing T a set of theorems (and talking about $\mathcal{S}[T]$-consistency of a set of non-theorems N instead of $\mathcal{S}$-consistency of the pair $\langle T, N\rangle$).

Definition 4.2 *For a non-monotonic system $\mathcal{S}$, a set of non-theorems $N \subseteq \mathcal{L}^{-}(\mathcal{S})$ is $\mathcal{S}$-consistent if and only if $\mathcal{S}(N) \cap N = \emptyset$.*

Generally speaking, the notion of $\mathcal{S}$-consistency is different from that of classical consistency, but these notions are strongly related in restricted forms of minimal model reasoning, for instance. This is addressed by the following.

Example 4.3 Let $P = \mathcal{A}$ be the set of atoms to be minimized. Consider the subsystem $\mathcal{S} = \mathcal{C}_P^{\mathcal{A}}$ of $\mathcal{C}_P$ having the language $\langle \mathcal{L}, \mathcal{A} \rangle$ and a propositionally consistent set of sentences $T \subseteq \mathcal{L}$. Now a set of non-theorems $N \subseteq \mathcal{A}$ is $\mathcal{S}[T]$-consistent if and only if $T \cup \mathrm{Neg}_P(N)$ is propositionally consistent. □

It is easy to see from Definition 4.2 that the empty set of non-theorems $\emptyset$ is $\mathcal{S}$-consistent. If we take larger and larger sets of non-theorems N, it is to be expected that larger and larger sets of theorems $\mathcal{S}(N)$ result by the requirement MON and N may become $\mathcal{S}$-*inconsistent* finally. Lemma 4.4 provides a straightforward result concerning $\mathcal{S}$-consistency and the requirement MON.

Lemma 4.4 *If N is $\mathcal{S}$-consistent, then every $N' \subseteq N$ is also $\mathcal{S}$-consistent.*

Proof. Assume that N' is not $\mathcal{S}$-consistent. Then there is a sentence $\phi \in N'$ such that $\phi \in \mathcal{S}(N')$. Thus $\phi \in N$ and $\phi \in \mathcal{S}(N)$ by the requirement MON. This contradicts the $\mathcal{S}$-consistency of N. Hence, N' is $\mathcal{S}$-consistent. □

We argue that the set of theorems $\mathcal{S}(N)$ obtained with $N = \emptyset$ is too weak for practical purposes, since the system $\mathcal{S}$ performs only monotonic reasoning in this case. This argument can be understood on the basis of Example 2.5 where non-theorems are required in order to reason about the correct behavior of an and-gate G. Therefore, we need a condition that forces us to select a set of non-theorems N which is larger than $\emptyset$. Such a condition is obtained by identifying the sentences of $\mathcal{L}^-(\mathcal{S})$ that are not *determined* by N. More precisely, a sentence $\phi \in \mathcal{L}^-(\mathcal{S})$ is *undetermined* by N, if it is not included in $\mathcal{S}(N)$ nor in N. In Definition 4.5, we define $\mathcal{S}$-complete sets of non-theorems N by denying undetermined sentences. Lemma 4.6 relates the notion of $\mathcal{S}$-completeness with our monotonicity requirement in the obvious way.

Definition 4.5 *For a non-monotonic system $\mathcal{S}$, a set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}$-complete if and only if $\mathcal{L}^-(\mathcal{S}) - (\mathcal{S}(N) \cup N) = \emptyset$.*

Lemma 4.6 *If N in $\mathcal{S}$-complete, then every $N' \supseteq N$ is also $\mathcal{S}$-complete.*

Proof. Assume that N in $\mathcal{S}$-complete and $N \subseteq N'$. Now $\mathcal{S}(N) \cup N$ is a subset of $\mathcal{S}(N') \cup N'$ by the requirement MON. It follows that $\mathcal{L}^-(\mathcal{S}) - (\mathcal{S}(N') \cup N')$ is a subset of $\mathcal{L}^-(\mathcal{S}) - (\mathcal{S}(N) \cup N) = \emptyset$. Hence, $\mathcal{L}^-(\mathcal{S}) - (\mathcal{S}(N') \cup N') = \emptyset$. □

An example on our notions of $\mathcal{S}$-consistency and $\mathcal{S}$-completeness follows.

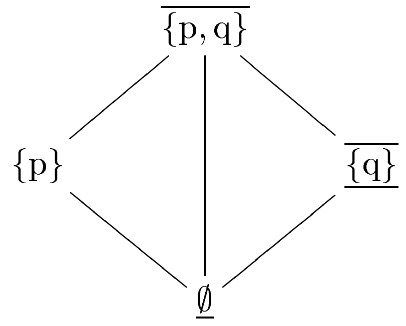


Figure 2: Sets of non-theorems in Example 4.7

Example 4.7 Consider an autoepistemic theory $\Sigma = \{\mathbf{D}\mathrm{p} \to \mathrm{p}, \mathbf{D}\mathrm{q} \to \mathrm{p}\}$ and the subsystem $\mathcal{S} = \boldsymbol{\mathcal{A}}^{\Sigma}_{\Sigma}$ which has the language $\langle\{\mathrm{p}, \mathrm{q}\}, \{\mathrm{p}, \mathrm{q}\}\rangle$.

For this system, the set of non-theorems $N_1 = \emptyset$ is trivially $\mathcal{S}$-consistent but not $\mathcal{S}$-complete, since both p and q are undetermined by N_1. The set of non-theorems $N_2 = \{\mathrm{p}\}$ is not $\mathcal{S}$-consistent, because $\mathrm{p} \in \mathcal{S}(N_2) = \{\mathrm{p}\}$. The set of non-theorems N_2 is neither $\mathcal{S}$-complete, since q remains undetermined by N_2. If we extend N_2 to $N_3 = \{\mathrm{p}, \mathrm{q}\}$, we obtain a set of non-theorems that is $\mathcal{S}$-complete, but not $\mathcal{S}$-consistent by Lemma 4.4. The set of non-theorems $N_4 = \{\mathrm{q}\}$ is both $\mathcal{S}$-consistent and $\mathcal{S}$-complete, since $\mathcal{S}(N_4) = \{\mathrm{p}\}$. $\square$

We introduce a graphical representation of sets of non-theorems for a given non-monotonic system $\mathcal{S}$. For Example 4.7 there were four possible sets of non-theorems, namely $\emptyset$, $\{\mathrm{p}\}$, $\{\mathrm{q}\}$ and $\{\mathrm{p}, \mathrm{q}\}$. These are naturally ordered by the subset relation $\subseteq$. We have depicted the resulting complete lattice in Figure 2. The proper inclusions have been represented by solid lines. In the sequel, we shall often omit these lines, since inclusions are quite obvious by the organization of the figure. The $\mathcal{S}$-consistent and $\mathcal{S}$-complete sets of non-theorems have been underlined and overlined, respectively.

4.2 Stable Semantics

Probably the most well-known semantics of non-monotonic reasoning is obtained in terms of the criteria introduced in Section 4.1.1. *Stable semantics* distinguishes sets of non-theorems which are both $\mathcal{S}$-consistent and $\mathcal{S}$-complete. This is achieved by the fixed point condition of Definition 4.8.

Definition 4.8 *For a non-monotonic system $\mathcal{S}$, a set of non-theorems $N \subseteq \mathcal{L}^{-}(\mathcal{S})$ is $\mathcal{S}$-stable if and only if $N = \overline{\mathcal{S}}(N) = \mathcal{L}^{-}(\mathcal{S}) - \mathcal{S}(N)$.*

The operator $\overline{\mathcal{S}} : \mathbf{2}^{\mathcal{L}^{-}(\mathcal{S})} \to \mathbf{2}^{\mathcal{L}^{-}(\mathcal{S})}$ is an analog of the well-known Gelfond-Lifschitz operator for normal logic programs [32]. The major difference is that

$\overline{\mathcal{S}}$ is formulated for sets of non-theorems rather than sets of theorems and in this sense it is a dual operator. This operator will play a substantial role in the rest of this work, since it allows capturing $\mathcal{S}$-consistency and $\mathcal{S}$-completeness conditions by simple subset relations as shown in Propositions 4.9 and 4.10.

Proposition 4.9 *A set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}$-consistent if and only if $N \subseteq \overline{\mathcal{S}}(N)$.*

Proof. ($\Rightarrow$) Let N be $\mathcal{S}$-consistent, i.e. $\mathcal{S}(N) \cap N = \emptyset$. Let $\phi \in N$ hold. Then assume that $\phi \notin \overline{\mathcal{S}}(N)$, i.e. $\phi \notin \mathcal{L}^-(\mathcal{S})$ or $\phi \in \mathcal{S}(N)$. Since $\phi \in N$, $\phi \in \mathcal{L}^-(\mathcal{S})$ and $\phi \in \mathcal{S}(N)$ must be the case. It follows that $\phi \in \mathcal{S}(N) \cap N$, a contradiction. Hence $\phi \in \overline{\mathcal{S}}(N)$. We have thus established that $N \subseteq \overline{\mathcal{S}}(N)$. ($\Leftarrow$) Let $N \subseteq \overline{\mathcal{S}}(N)$ hold, i.e. $N \subseteq \mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N)$. Assume that N is not $\mathcal{S}$-consistent. Then there is a sentence $\phi \in \mathcal{S}(N) \cap N$ so that $\phi \in \mathcal{S}(N)$ and $\phi \in N$. The latter implies that $\phi \in \mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N)$ so that $\phi \in \mathcal{L}^-(\mathcal{S})$ and $\phi \notin \mathcal{S}(N)$, a contradiction. Thus N is $\mathcal{S}$-consistent. □

Proposition 4.10 *A set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}$-complete if and only if $\overline{\mathcal{S}}(N) \subseteq N$.*

Proof. ($\Rightarrow$) Let N be $\mathcal{S}$-complete, i.e. $\mathcal{L}^-(\mathcal{S}) - (\mathcal{S}(N) \cup N) = \emptyset$. Let $\phi \in \overline{\mathcal{S}}(N)$ hold, i.e. $\phi \in \mathcal{L}^-(\mathcal{S})$ but $\phi \notin \mathcal{S}(N)$. Assuming that $\phi \notin N$ implies that $\phi \notin \mathcal{S}(N) \cup N$ and $\phi \in \mathcal{L}^-(\mathcal{S}) - (\mathcal{S}(N) \cup N)$, a contradiction. Hence $\phi \in N$ and we have that $\overline{\mathcal{S}}(N) \subseteq N$. ($\Leftarrow$) Let $\overline{\mathcal{S}}(N) \subseteq N$ hold, i.e. $\mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N) \subseteq N$. Then assume that N is not $\mathcal{S}$-complete, i.e. there is $\phi \in \mathcal{L}^-(\mathcal{S}) - (\mathcal{S}(N) \cup N)$. It follows that $\phi \in \mathcal{L}^-(\mathcal{S})$ but $\phi \notin \mathcal{S}(N) \cup N$. This implies that $\phi \notin \mathcal{S}(N)$ and $\phi \notin N$. Consequently $\phi \in \mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N)$ but $\phi \notin N$, a contradiction. Therefore N is $\mathcal{S}$-complete. □

The preceding two propositions give rise to Corollary 4.11: the fixed point equation $N = \overline{\mathcal{S}}(N)$ really captures sets of non-theorems that are both $\mathcal{S}$-consistent and $\mathcal{S}$-complete.

Corollary 4.11 *A set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}$-stable if and only if N is $\mathcal{S}$-consistent and $\mathcal{S}$-complete.*

Example 4.12 We have already encountered a set of non-theorems that is $\mathcal{S}$-stable, namely the set of non-theorems $N_3 = \{\mathrm{q}\}$ in Example 4.7. To re-confirm this we may compute $\overline{\mathcal{S}}(\{\mathrm{q}\}) = \mathcal{L}^-(\mathcal{S}) - \mathcal{S}(\{\mathrm{q}\}) = \{\mathrm{p}, \mathrm{q}\} - \{\mathrm{p}\} = \{\mathrm{q}\}$. This is also easily read from Figure 2 since $N_3 = \{\mathrm{q}\}$ is the only set of non-theorems that has been both underlined and overlined. □

We have treated sets of non-theorems as first class objects in Definition 4.8. This is especially advantageous if $\mathcal{L}^-(\mathcal{S})$ is considerably smaller than $\mathcal{L}^+(\mathcal{S})$.

More traditional approaches put sets of theorems in a similar role. To bring the differences into daylight, we have phrased a stability condition for theorems in Definition 4.13. In fact, this definition includes as special cases the definitions of P-extensions proposed in this work, default extensions [94] and iterative expansions [47, 66]. Moreover, the operator $\mathrm{ST}_{\mathcal{S}}$ coincides with the previously mentioned Gelfond-Lifschitz operator [32] if formulated for normal logic programs and the fixed point condition $T = \mathrm{ST}_{\mathcal{S}}(T)$ captures *stable models* [32] of normal logic programs.

Definition 4.13 *A set of theorems $T \subseteq \mathcal{L}^+(\mathcal{S})$ is $\mathcal{S}$-stable if and only if $T = \mathrm{ST}_{\mathcal{S}}(T) = \mathcal{S}(\mathcal{L}^-(\mathcal{S}) - T)$.*

In Proposition 4.14, we build a bridge between $\mathcal{S}$-stable sets of non-theorems and theorems. By this connection, the fixed point condition of Definition 4.8 captures the above mentioned notions of extensions and expansions.

Proposition 4.14 *(i) If $N \subseteq \mathcal{L}^-(\mathcal{S})$ is an $\mathcal{S}$-stable set of non-theorems, then $T = \mathcal{S}(N)$ is an $\mathcal{S}$-stable set of theorems. (ii) If $T \subseteq \mathcal{L}^+(\mathcal{S})$ is an $\mathcal{S}$-stable set of theorems, then $N = \mathcal{L}^-(\mathcal{S}) - T$ is an $\mathcal{S}$-stable set of non-theorems.*

Proof. (i) Let $N \subseteq \mathcal{L}^-(\mathcal{S})$ be $\mathcal{S}$-stable, i.e. $N = \overline{\mathcal{S}}(N) = \mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N) = \mathcal{L}^-(\mathcal{S}) - T$. It follows that $T = \mathcal{S}(N) = \mathcal{S}(\mathcal{L}^-(\mathcal{S}) - T)$ and T is $\mathcal{S}$-stable. (ii) Let $T \subseteq \mathcal{L}^+(\mathcal{S})$ be $\mathcal{S}$-stable, i.e. $T = \mathcal{S}(\mathcal{L}^-(\mathcal{S}) - T) = \mathcal{S}(N)$. It follows that $N = \mathcal{L}^-(\mathcal{S}) - T = \mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N)$. Thus $N = \overline{\mathcal{S}}(N)$ is $\mathcal{S}$-stable. $\square$

Basically, the subsets of $\mathcal{L}^-(\mathcal{S})$ form the candidates for $\mathcal{S}$-stable sets of non-theorems N. In Example 4.7 we found a unique $\mathcal{S}$-stable set of non-theorems. Example 4.15 demonstrates that the number of $\mathcal{S}$-stable sets of non-theorems may vary. We have presented the candidates of the example in Figure 3. The lines describing subset inclusion have been dropped, arrows have been added to mark applications of the operator $\overline{\mathcal{S}}$. For each set of non-theorems N, an arrow leads from N to $\overline{\mathcal{S}}(N)$, i.e. $N \longrightarrow \overline{\mathcal{S}}(N)$.

Example 4.15 Let $D = \{\frac{:\neg\mathrm{p}}{\mathrm{p}}\}$ be a set of defaults and $\mathcal{S} = \mathfrak{D}_D^{\{\neg\neg\mathrm{p}\}}$ the subsystem having the language $\langle\mathcal{L}, \{\neg\neg\mathrm{p}\}\rangle$. There are two candidates for $\mathcal{S}$-stable sets, namely $N_1 = \emptyset$ and $N_2 = \{\neg\neg\mathrm{p}\}$. None of these two is $\mathcal{S}$-stable, since N_1 is only $\mathcal{S}$-consistent and N_2 is only $\mathcal{S}$-complete. Let us clarify this with some details. (i) The reduction $D_{N_1} = \emptyset$ so that $\mathcal{S}(N_1) = \mathrm{Cn}^{D_{N_1}}(\emptyset) = \mathrm{Cn}(\emptyset)$ follows. Since $\neg\neg\mathrm{p} \notin \mathrm{Cn}(\emptyset)$, we obtain that $\overline{\mathcal{S}}(N_1) = \{\neg\neg\mathrm{p}\} = N_2$. (ii) The reduction $D_{N_2} = \{\frac{\top}{\mathrm{p}}\}$ implying that $\mathcal{S}(N_2) = \mathrm{Cn}^{D_{N_2}}(\emptyset) = \mathrm{Cn}(\{\mathrm{p}\})$. Since $\neg\neg\mathrm{p} \in \mathrm{Cn}(\{\mathrm{p}\})$, we have that $\overline{\mathcal{S}}(N_2) = \emptyset = N_1$.

Consider a set of atoms $P = \{\mathrm{p}, \mathrm{q}\} = \mathcal{A}$ to be minimized and the subsystem $\mathcal{S}' = \mathfrak{C}_P^{\mathcal{A}}$ having the language $\langle\mathcal{L}, \{\mathrm{p}, \mathrm{q}\}\rangle$. Let T be the theory $\{\mathrm{p} \vee \mathrm{q}\}$. There are four candidates for $\mathcal{S}'[T]$-stable sets, namely $N'_1 = \emptyset$, $N'_2 = \{\mathrm{p}\}$, $N'_3 = \{\mathrm{q}\}$ and $N'_4 = \{\mathrm{p}, \mathrm{q}\}$. The first three are $\mathcal{S}'[T]$-consistent since T, $T \cup \{\neg\mathrm{p}\}$ and $T \cup \{\neg\mathrm{q}\}$ are propositionally consistent theories. On the other hand, the last three are $\mathcal{S}'[T]$-complete, because $T \cup \{\neg\mathrm{p}\}$, $T \cup \{\neg\mathrm{q}\}$ and $T \cup \{\neg\mathrm{p}, \neg\mathrm{q}\}$ are propositionally complete[20] theories (recall that $\mathcal{L}$ is based on a set of atoms $\mathcal{A}$ which contains only p and q). Hence N_2 and N_3 are $\mathcal{S}'[T]$-stable. $\square$

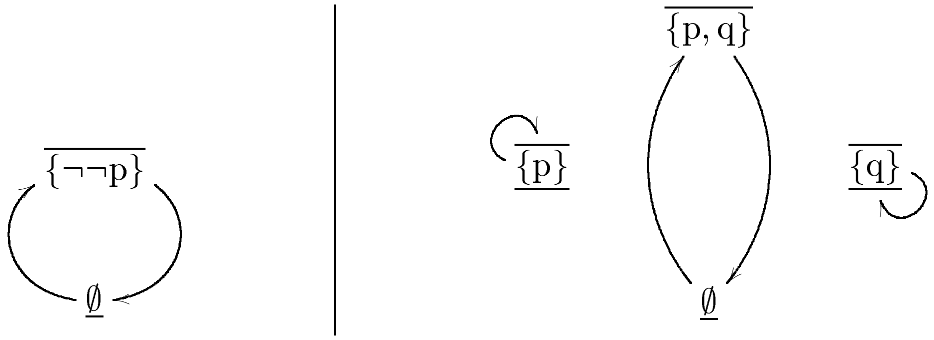


Figure 3: Candidates for stable sets of non-theorems in Example 4.15.

We shall later encounter non-monotonic systems $\mathcal{S}$ where the number of $\mathcal{S}$-stable sets of non-theorems is even exponential in the size of $\mathcal{L}^-(\mathcal{S})$. On the other hand, it is not possible that every $N \subseteq \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}$-stable simultaneously (given that $\mathcal{L}^-(\mathcal{S})$ is non-empty). This is because an $\mathcal{S}$-stable set of non-theorems excludes candidates substantially, namely proper subsets and proper supersets of N are not $\mathcal{S}$-stable as shown in Proposition 4.16. This indicates that the class of $\mathcal{S}$-stable sets of non-theorems forms an *antichain* [15].

Proposition 4.16 *$\mathcal{S}$-stable sets of non-theorems are incomparable by $\subset$, i.e. $N_1 \subset N_2$ never holds for any $\mathcal{S}$-stable sets of non-theorems N_1 and N_2.*

Proof. Assume there are $\mathcal{S}$-stable N_1 and N_2 such that $N_1 \subset N_2$, i.e. $N_1 \subseteq N_2$ and $N_1 \neq N_2$. It follows that $\mathcal{S}(N_1) \subseteq \mathcal{S}(N_2)$ by the requirement MON. This implies that $N_2 = \mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N_2) = \overline{\mathcal{S}}(N_2)$ is a subset of $N_1 = \mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N_1) = \overline{\mathcal{S}}(N_1)$. Hence $N_1 = N_2$ holds and we have a contradiction. $\square$

We may now define precisely the stable semantics of a non-monotonic system $\mathcal{S}$. We follow the lines sketched in Section 4.1. First of all, we identify the set of theorems T which is the argument to the semantic operator $\mathrm{STAB}_{\mathcal{S}}$ to be defined. This takes place by considering $\mathcal{S}[T]$-stable sets of non-theorems. For an $\mathcal{S}[T]$-stable set of non-theorems N, we obtain by Proposition 4.14 the corresponding set of theorems, i.e. $\mathcal{S}[T](N) = \mathcal{S}(T, N)$. As discussed in conjunction with Example 4.15, a non-monotonic system $\mathcal{S}$ may have several $\mathcal{S}[T]$-stable sets of non-theorems. Therefore the intersection of the corresponding sets of theorems $\mathcal{S}[T](N) = \mathcal{S}(T, N)$ is defined as the set of theorems $\mathrm{STAB}_{\mathcal{S}}(T)$.

[20] A theory $T \subseteq \mathcal{L}$ is propositionally complete if it has at most one model.

Definition 4.17 *The operator* $\mathrm{STAB}_{\mathcal{S}}$ *for the stable semantics of a non-monotonic system* $\mathcal{S}$ *is defined for all* $T \subseteq \mathcal{L}^+(\mathcal{S})$ *by*

$$\mathrm{STAB}_{\mathcal{S}}(T) = \bigcap\{\mathcal{S}(T, N) \mid N \subseteq \mathcal{L}^-(\mathcal{S}) \textit{ is } \mathcal{S}[T]\textit{-stable}\}.$$

By this definition, a set of theorems T is mapped to the set of theorems $\mathrm{STAB}_{\mathcal{S}}(T)$ in the spirit of Definition 4.1. Having made the semantic operator $\mathrm{STAB}_{\mathcal{S}}$ explicit, many properties of the semantics are then easily expressed. In Example 4.18, it is demonstrated that stable semantics is indeed non-monotonic. More precisely, the semantic operator $\mathrm{STAB}_{\mathcal{S}}$ is *not* a monotonic inference operator in the sense of Definition 2.2. This is as one would expect, since we are dealing with *non-monotonic* systems.

Example 4.18 Consider an autoepistemic theory

$$\Sigma = \{\mathbf{D}\mathrm{a} \to \mathrm{b}, \mathbf{D}\mathrm{b} \to \mathrm{a}, \mathbf{D}\mathrm{c} \to \mathrm{b}\}$$

and the subsystem $\mathcal{S} = \mathcal{A}^{\Sigma}_{\Sigma}$ having the language $\langle\{\mathrm{a}, \mathrm{b}, \mathrm{c}\}, \{\mathrm{a}, \mathrm{b}, \mathrm{c}\}\rangle$.

There is only one $\mathcal{S}$-stable set of non-theorems, namely $N = \{\mathrm{a}, \mathrm{c}\}$ implying that $\mathrm{STAB}_{\mathcal{S}}(\emptyset) = \{\mathrm{b}\}$. For the set of sentences $T = \{\mathrm{c}\}$, the sets of non-theorems $N_1 = \{\mathrm{a}\}$ and $N_2 = \{\mathrm{b}\}$ are $\mathcal{S}[T]$-stable. These give rise to sets of theorems $\mathcal{S}(T, N_1) = \{\mathrm{b}, \mathrm{c}\}$ and $\mathcal{S}(T, N_2) = \{\mathrm{a}, \mathrm{c}\}$. Consequently, we obtain $\mathrm{STAB}_{\mathcal{S}}(T) = \{\mathrm{c}\}$. Thus $\mathrm{STAB}_{\mathcal{S}}(\emptyset) \not\subseteq \mathrm{STAB}_{\mathcal{S}}(T)$, although $\emptyset \subseteq T$. So we have obtained a counter-example for the monotonicity of $\mathrm{STAB}_{\mathcal{S}}$. □

4.3 Stationary Semantics

If a non-monotonic system $\mathcal{S}$ does not have $\mathcal{S}[T]$-stable sets of non-theorems at all, it follows that $\mathrm{STAB}_{\mathcal{S}}(T) = \mathcal{L}^+(\mathcal{S})$. In this case, the set of theorems T can be considered to be inconsistent under stable semantics. This situation is realized in Example 4.15 for the system $\mathcal{S}$ with the set of theorems $T = \emptyset$. Strictly speaking, this phenomenon is one of the pitfalls of stable semantics, since it affects properties of the corresponding semantic operator $\mathrm{STAB}_{\mathcal{S}}$.[21] Therefore we would like to loosen the condition of stable semantics so that at least one set of non-theorems is accepted in any case.

A weaker condition is obtained by the following line of reasoning. Suppose we are given a non-monotonic system $\mathcal{S}$ and an $\mathcal{S}$-consistent set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$. This set gives rise to a set of theorems $\mathcal{S}(N)$ which is disjoint from N by our assumption on N. Then $N \subseteq \overline{\mathcal{S}}(N)$ follows by Proposition 4.9 and we may consider the set $\overline{\mathcal{S}}(N)$ as the set of *potential non-theorems*

[21] It is later shown in Section 5.1 that $\mathrm{STAB}_{\mathcal{S}}$ is not cumulative.

given N, since none of the sentences in $\overline{\mathcal{S}}(N)$ belongs to the set of theorems $\mathcal{S}(N)$ by definition. The set of potential non-theorems $\overline{\mathcal{S}}(N)$ may really be larger than the set of non-theorems N we started with, giving rise to the set of *potential theorems* $\mathcal{S}(\overline{\mathcal{S}}(N))$. By the requirement MON it holds that $\mathcal{S}(N) \subseteq \mathcal{S}(\overline{\mathcal{S}}(N))$. Then we consider sentences that are not potential theorems to be non-theorems for sure. This leads to the set of non-theorems $\overline{\mathcal{S}}(\overline{\mathcal{S}}(N))$. An interesting situation is when this particular set of non-theorems coincides with the set of non-theorems N we started with. This idea is behind the notion of an $\mathcal{S}$-stationary set of non-theorems given in Definition 4.19.

Definition 4.19 *A set of non-theorems $N \subseteq \mathcal{L}^{-}(\mathcal{S})$ is $\mathcal{S}$-stationary if and only if $N = \overline{\mathcal{S}}^2(N) = \overline{\mathcal{S}}(\overline{\mathcal{S}}(N))$ and $N \subseteq \overline{\mathcal{S}}(N)$.*

This is also a fixed point condition, but formulated in terms of the operator $\overline{\mathcal{S}}^2$ which corresponds to two subsequent applications of the operator $\overline{\mathcal{S}}$. The requirement $N \subseteq \overline{\mathcal{S}}(N)$ ensures that $\mathcal{S}$-stationary sets of non-theorems are $\mathcal{S}$-consistent by Lemma 4.9. Given an $\mathcal{S}$-stationary set of non-theorems N, the corresponding set of theorems is naturally $\mathcal{S}(N)$. In this way, an $\mathcal{S}$-stationary set of non-theorems N leaves the sentences in $\mathcal{L}^{-}(\mathcal{S}) - (\mathcal{S}(N) \cup N)$ undetermined – indicating that N is not necessarily $\mathcal{S}$-complete.

Example 4.20 Let us look back to the system $\mathcal{S}$ of Example 4.15 with the candidates $N_1 = \emptyset$ and $N_2 = \{\neg\neg\mathrm{p}\}$. The set of non-theorems N_1 is $\mathcal{S}$-stationary, since $\overline{\mathcal{S}}^2(N_1) = \overline{\mathcal{S}}(N_2) = \emptyset = N_1$ and N_1 is $\mathcal{S}$-consistent. In this case, the set of potential non-theorems is $N_2 = \{\neg\neg\mathrm{p}\}$ and the set of potential theorems is $\mathcal{S}(N_2) = \mathrm{Cn}(\{\mathrm{p}\})$. The $\mathcal{S}$-stationarity of N_1 is easily read from Figure 3, since we have a loop of two arrows starting from N_1 which is also $\mathcal{S}$-consistent as indicated by the underlining. On the other hand, the set of non-theorems N_2 is not $\mathcal{S}$-consistent nor $\mathcal{S}$-stationary. $\square$

The operator $\overline{\mathcal{S}}$ provides an interesting mapping *between* the classes of $\mathcal{S}$-consistent and $\mathcal{S}$-complete sets of non-theorems as indicated by Lemma 4.21. Therefore, for an $\mathcal{S}$-stationary set of non-theorems N, the set of potential non-theorems $\overline{\mathcal{S}}(N)$ is $\mathcal{S}$-complete, since N is $\mathcal{S}$-consistent. This information is also easily readable from Figure 3: arcs lead from $\mathcal{S}$-consistent sets of non-theorems to $\mathcal{S}$-complete sets of non-theorems and back.

Lemma 4.21 *(i) If $N \subseteq \mathcal{L}^{-}(\mathcal{S})$ is $\mathcal{S}$-consistent, then $\overline{\mathcal{S}}(N)$ is $\mathcal{S}$-complete. (ii) If $N \subseteq \mathcal{L}^{-}(\mathcal{S})$ is $\mathcal{S}$-complete, then $\overline{\mathcal{S}}(N)$ is $\mathcal{S}$-consistent.*

Proof. The proofs of claims (i) and (ii) are postponed until Lemma 4.25. $\square$

In analogy to Proposition 4.14 for the $\mathcal{S}$-stable sets of non-theorems, the $\mathcal{S}$-stationary sets of non-theorems are in close relationship with the *stationary* *extensions* of default theories [89], the *stationary expansions* of autoepistemic theories proposed by the author [47] and the *well-founded models* of normal logic programs [115]. This is how Definition 4.19 captures the corresponding notions in various contexts, but in terms of sets of non-theorems rather than sets of theorems. The definition of stationary semantics is analogous to that of stable semantics by the principles introduced in Section 4.1.

Definition 4.22 *The operator* $\mathrm{STAT}_{\mathcal{S}}$ *for the stationary semantics of a non-monotonic system* $\mathcal{S}$ *is defined for all* $T \subseteq \mathcal{L}^+(\mathcal{S})$ *by*

$$\mathrm{STAT}_{\mathcal{S}}(T) = \bigcap\{\mathcal{S}(T,N) \mid N \subseteq \mathcal{L}^-(\mathcal{S}) \textit{ is } \mathcal{S}[T]\textit{-stationary}\}.$$

4.3.1 Relationship to Stable Semantics

Let us now address the relationship of stationary semantics to stable semantics. For this purpose, we have to introduce an ordering on semantic operators. Given two semantic operators $\mathrm{SEM}^1_{\mathcal{S}} : \mathbf{2}^{\mathcal{L}^+(\mathcal{S})} \to \mathbf{2}^{\mathcal{L}^+(\mathcal{S})}$ and $\mathrm{SEM}^2_{\mathcal{S}} : \mathbf{2}^{\mathcal{L}^+(\mathcal{S})} \to \mathbf{2}^{\mathcal{L}^+(\mathcal{S})}$ for a non-monotonic system $\mathcal{S}$, we write $\mathrm{SEM}^1_{\mathcal{S}} \sqsubseteq \mathrm{SEM}^2_{\mathcal{S}}$ to indicate that $\mathrm{SEM}^1_{\mathcal{S}}$ is *weaker* than $\mathrm{SEM}^2_{\mathcal{S}}$ (or alternatively, $\mathrm{SEM}^2_{\mathcal{S}}$ is *stronger* than $\mathrm{SEM}^1_{\mathcal{S}}$). This relation has the obvious definition given as follows. In Theorem 4.24, we relate the two semantics introduced so far.

Definition 4.23 *For the semantics* $\mathrm{SEM}^1_{\mathcal{S}}$ *and* $\mathrm{SEM}^2_{\mathcal{S}}$ *of* $\mathcal{S}$*, define* $\mathrm{SEM}^1_{\mathcal{S}} \sqsubseteq \mathrm{SEM}^2_{\mathcal{S}}$ *if and only if* $\mathrm{SEM}^1_{\mathcal{S}}(T) \subseteq \mathrm{SEM}^2_{\mathcal{S}}(T)$ *for all* $T \subseteq \mathcal{L}^+(\mathcal{S})$.

Theorem 4.24 $\mathrm{STAT}_{\mathcal{S}} \sqsubseteq \mathrm{STAB}_{\mathcal{S}}$ *for every non-monotonic system* $\mathcal{S}$.

Proof. Every $\mathcal{S}$-stable set of non-theorems N is also also $\mathcal{S}$-stationary. This is because $N = \overline{\mathcal{S}}(N)$ implies $\overline{\mathcal{S}}(N) = \overline{\mathcal{S}}(\overline{\mathcal{S}}(N)) = \overline{\mathcal{S}}^2(N)$ so that $N = \overline{\mathcal{S}}^2(N)$ and $N \subseteq \overline{\mathcal{S}}(N)$ hold. Then let $T \subseteq \mathcal{L}^+(\mathcal{S})$ and assume that $\phi \notin \mathrm{STAB}_{\mathcal{S}}(T)$. So there is an $\mathcal{S}[T]$-stable set of non-theorems N such that $\phi \notin \mathcal{S}[T](N)$. Since N is also $\mathcal{S}[T]$-stationary $\phi \notin \mathrm{STAT}_{\mathcal{S}}(T)$. Thus $\mathrm{STAT}_{\mathcal{S}}(T) \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$. In particular, this holds if there are no $\mathcal{S}[T]$-stable sets of non-theorems so that $\mathrm{STAB}_{\mathcal{S}}(T) = \mathcal{L}^+(\mathcal{S})$. □

The operators $\overline{\mathcal{S}}$ and $\overline{\mathcal{S}}^2$ associated with stable and stationary semantics have the following well-known (anti)monotonicity properties. It is also the right moment to prove Lemma 4.21 presented earlier.

Proposition 4.25 *Let* $\mathcal{S}$ *be a non-monotonic system. For every* $N_1 \subseteq N_2 \subseteq \mathcal{L}^-(\mathcal{S})$*, (i)* $\overline{\mathcal{S}}(N_2) \subseteq \overline{\mathcal{S}}(N_1)$ *and (ii)* $\overline{\mathcal{S}}^2(N_1) \subseteq \overline{\mathcal{S}}^2(N_2)$.

Proof. (i) Because $N_1 \subseteq N_2$ it follows that $\mathcal{S}(N_1) \subseteq \mathcal{S}(N_2)$ by the requirement MON. Thus $\mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N_2)$ is a subset of $\mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N_1)$ so that $\overline{\mathcal{S}}(N_2) \subseteq \overline{\mathcal{S}}(N_1)$. (ii) Then let $N_1' = \overline{\mathcal{S}}(N_2)$ and $N_2' = \overline{\mathcal{S}}(N_1)$. Since $N_1' \subseteq N_2' \subseteq \mathcal{L}^-(\mathcal{S})$, we get $\overline{\mathcal{S}}(N_2') \subseteq \overline{\mathcal{S}}(N_1')$ by (i). Thus $\overline{\mathcal{S}}^2(N_1) \subseteq \overline{\mathcal{S}}^2(N_2)$. □

Proof of Lemma 4.21. (i) Assume that N is $\mathcal{S}$-consistent, i.e. $N \subseteq \overline{\mathcal{S}}(N)$ by Proposition 4.9. It follows that $\overline{\mathcal{S}}(\overline{\mathcal{S}}(N)) \subseteq \overline{\mathcal{S}}(N)$ by Proposition 4.25. Thus $N' = \overline{\mathcal{S}}(N)$ is $\mathcal{S}$-complete by Proposition 4.10. (ii) Assume that N is $\mathcal{S}$-complete, i.e. $\overline{\mathcal{S}}(N) \subseteq N$ by Proposition 4.10. Thus $\overline{\mathcal{S}}(N) \subseteq \overline{\mathcal{S}}(\overline{\mathcal{S}}(N))$ by Proposition 4.25 and $N' = \overline{\mathcal{S}}(N)$ is $\mathcal{S}$-consistent by Proposition 4.9. □

The monotonicity of $\overline{\mathcal{S}}^2$ has an important consequence in Proposition 4.27 when $N = \emptyset$: the least fixed point of the operator $\overline{\mathcal{S}}^2$ can be iteratively constructed. These results were first established for normal logic programs by Van Gelder et al. [115] and for default logic by Przymusinska and Przymusinski [89]. This result extends for the bimodal autoepistemic logic proposed in this work by Theorem 2.34. We have formulated a stronger result in Proposition 4.27 for sets of non-theorems N that satisfy $N \subseteq \overline{\mathcal{S}}(N)$ and $N \subseteq \overline{\mathcal{S}}^2(N)$. For such N, we introduce the notation $\mathrm{lfp}(\overline{\mathcal{S}}^2, N)$ to denote the least fixed point of $\overline{\mathcal{S}}^2$ that extends N. We need to establish Lemma 4.26 to prove Proposition 4.27. By this lemma, taking a limit of a non-decreasing sequence of $\mathcal{S}$-consistent sets of non-theorems preserves $\mathcal{S}$-consistency. The requirement COM has a key role in the proof.

Lemma 4.26 *Let α be a limit ordinal and N_β a non-decreasing sequence of sets of non-theorems indexed by $\beta < \alpha$ such that N_α is the limit $\bigcup_{\beta<\alpha} N_\beta$. If $N_\beta \subseteq \overline{\mathcal{S}}(N_\beta)$ for all $\beta < \alpha$, then $N_\alpha \subseteq \overline{\mathcal{S}}(N_\alpha)$.*

Proof. Let us assume that $N_\alpha \not\subseteq \overline{\mathcal{S}}(N_\alpha)$, i.e. there is a sentence $\phi \in N_\alpha$ such that $\phi \in \mathcal{S}(N_\alpha)$. Since our sequence of sets of non-theorems is non-decreasing, we define β' as the least ordinal $\beta' < \alpha$ such that $\phi \in N_{\beta'}$. By the requirement COM, we obtain $\mathcal{S}(N_\alpha) = \bigcup\{\mathcal{S}(N') \mid N' \subseteq_{\mathrm{f}} N_\alpha\}$. So there is $N' \subseteq_{\mathrm{f}} N_\alpha$ such that $\phi \in \mathcal{S}(N')$ and least $\beta < \alpha$ such that $N' \subseteq N_\beta$ and $\phi \in \mathcal{S}(N_\beta)$ by the requirement MON. Two cases arise. (i) If $\beta' \leq \beta$, we obtain $\phi \in N_\beta$, since our sequence is non-decreasing. Moreover, $\phi \in \mathcal{S}(N_\beta)$ implies $\phi \notin \overline{\mathcal{S}}(N_\beta)$ so that $N_\beta \not\subseteq \overline{\mathcal{S}}(N_\beta)$. (ii) If $\beta' > \beta$, we obtain $\phi \in \mathcal{S}(N_{\beta'})$ by the requirement MON. Thus $\phi \notin \overline{\mathcal{S}}(N_{\beta'})$ and $N_{\beta'} \not\subseteq \overline{\mathcal{S}}(N_{\beta'})$. By the cases (i) and (ii) we have a contradiction with our assumptions. Hence $N_\alpha \subseteq \overline{\mathcal{S}}(N_\alpha)$. □

Proposition 4.27 *Let N be a set of non-theorems such that $N \subseteq \overline{\mathcal{S}}(N)$ and $N \subseteq \overline{\mathcal{S}}^2(N)$. Then $N' = \mathrm{lfp}(\overline{\mathcal{S}}^2, N)$ is the least $\mathcal{S}$-stationary set of non-theorems such that $N \subseteq N'$.*

Proof. By Proposition 4.25, the operator $\overline{\mathbb{S}}^2$ is monotonic in the complete lattice consisting of all subsets of $\mathcal{L}^-(\mathbb{S})$ ordered by $\subseteq$. Since $N \subseteq \overline{\mathbb{S}}^2(N)$ the iteration sequence of $\overline{\mathbb{S}}^2$ defined by $N_\alpha = \overline{\mathbb{S}}^2 \uparrow^\alpha (N)$ for all ordinals α is non-decreasing. By the famous Knaster-Tarski lemma, this sequence of sets of non-theorems reaches a fixed point N' of $\overline{\mathbb{S}}^2$ which equals to N_α for a sufficiently large (and possibly transfinite) ordinal number α and $N \subseteq N'$.

To show that N' is $\mathbb{S}$-stationary, we have to ensure that N' is yet $\mathbb{S}$-consistent, i.e. $N' \subseteq \overline{\mathbb{S}}(N')$. So let us show by transfinite induction on α that $N_\alpha \subseteq \overline{\mathbb{S}}(N_\alpha)$ holds for every $N_\alpha = \overline{\mathbb{S}}^2 \uparrow^\alpha (N)$. The base case $\alpha = 0$ holds, since $N_0 = N$ satisfies $N_0 \subseteq \overline{\mathbb{S}}(N_0)$ by our requirements on N. Induction step follows. Let α be any successor ordinal. By the inductive hypothesis, it holds that $N_{\alpha-1} \subseteq \overline{\mathbb{S}}(N_{\alpha-1})$. It follows that $\overline{\mathbb{S}}^2(N_{\alpha-1}) \subseteq \overline{\mathbb{S}}^2(\overline{\mathbb{S}}(N_{\alpha-1})) = \overline{\mathbb{S}}(\overline{\mathbb{S}}^2(N_{\alpha-1}))$ by Proposition 4.25. Since $N_\alpha = \overline{\mathbb{S}}^2(N_{\alpha-1})$, we obtain $N_\alpha \subseteq \overline{\mathbb{S}}(N_\alpha)$ as desired. Finally, consider any limit ordinal α. Now $N_\beta \subseteq \overline{\mathbb{S}}(N_\beta)$ holds for all $\beta < \alpha$ by the inductive hypothesis. By Lemma 4.26 we know that $N_\alpha \subseteq \overline{\mathbb{S}}(N_\alpha)$ holds for the limit $N_\alpha = \bigcup_{\beta<\alpha} N_\beta$. Thus $N_\alpha \subseteq \overline{\mathbb{S}}(N_\alpha)$ holds for all ordinals α. It follows that also $N' \subseteq \overline{\mathbb{S}}(N')$. Thus N' is indeed $\mathbb{S}$-consistent and $\mathbb{S}$-stationary.

Finally, we show by transfinite induction that $N_\alpha \subseteq M$ for all fixed points M of $\overline{\mathbb{S}}^2$ such that $N \subseteq M$. The base case $\alpha = 0$ holds, since $N_0 = N$ is a subset of M. Then consider any successor ordinal α. By the inductive hypothesis $N_{\alpha-1} \subseteq M$. By Proposition 4.25, it holds that $\overline{\mathbb{S}}^2(N_{\alpha-1}) = N_\alpha$ is a subset of $\overline{\mathbb{S}}^2(M) = M$. For the last, we analyze a limit α. By the inductive hypothesis, each N_β is a subset of M for all $\beta < \alpha$. Hence $N_\alpha = \bigcup_{\beta<\alpha} N_\beta$ is a subset of M. Thus $N_\alpha \subseteq M$ holds for all ordinals α and $\mathrm{lfp}(\overline{\mathbb{S}}^2, N)$ is contained in every fixed point M of $\overline{\mathbb{S}}^2$ satisfying $N \subseteq M$.

Thus we are entitled to call N' the least fixed point of $\overline{\mathbb{S}}^2$ that extends N. Moreover, this fixed point is also the least $\mathbb{S}$-stationary set of non-theorems that extends N, since $\mathbb{S}$-stationary sets of non-theorems extending N are also fixed points of $\overline{\mathbb{S}}^2$ and so the fixed point N' is contained in all of them. □

A characterization of the stationary semantics results.

Theorem 4.28 *The stationary semantics* $\mathrm{STAT}_{\mathbb{S}}(T)$ *is determined by the least fixed point* $N = \mathrm{lfp}(\overline{\mathbb{S}[T]}^2, \emptyset)$, *i.e.* $\mathrm{STAT}_{\mathbb{S}}(T) = \mathbb{S}(T, N)$.

Proof. The set of non-theorems $\emptyset$ satisfies trivially the requirements of Proposition 4.27 with respect to the system $\mathbb{S}[T]$. Therefore, there is the least $\mathbb{S}[T]$-stationary set of non-theorems N which is the least fixed point $\mathrm{lfp}(\overline{\mathbb{S}[T]}^2, \emptyset)$. This particular set of non-theorems is contained in all $\mathbb{S}[T]$-stationary sets of non-theorems. Therefore, it is straightforward to see on the basis of Definition 4.22 and the requirement MON that a sentence $\phi \in \mathcal{L}^{+}(\mathcal{S})$ belongs to $\mathrm{STAT}_{\mathcal{S}}(T)$ if and only if $\phi \in \mathcal{S}(T, N)$. □

4.3.2 On the Applicability of Stationary Semantics

We showed in Theorem 4.24 that stationary semantics is a weaker semantics than stable semantics. Generally speaking, the weakest possible $\mathcal{S}[T]$-stationary set of non-theorems is naturally $\emptyset$ and the resulting set of theorems is $\mathcal{S}[T](\emptyset)$. In particular, the empty set of non-theorems is obtained as the least $\mathcal{S}[T]$-stationary set of non-theorems if the set of potential theorems $\mathcal{S}[T](\overline{\mathcal{S}[T]}(\emptyset))$ covers the whole language $\mathcal{L}^{-}(\mathcal{S})$. In such cases, the stationary semantics $\mathrm{STAT}_{\mathcal{S}}(T)$ *collapses* to the set of theorems $\mathcal{S}[T](\emptyset)$ which is based on monotonic inference only and non-theorems are simply of no use. In Example 4.29, we demonstrate this phenomenon with a minimal model system. Stationary semantics collapses in this case, because the set of potential theorems $\mathcal{S}[T](\overline{\mathcal{S}[T]}(\emptyset))$ is propositionally inconsistent.

Example 4.29 Recall Example 4.15 where the set of atoms $P = \{\mathrm{p}, \mathrm{q}\} = \mathcal{A}$ was minimized. The language of the subsystems $\mathcal{S}' = \mathcal{C}_P^{\mathcal{A}}$ of the minimal model system $\mathcal{C}_P$ was $\langle \mathcal{L}, \mathcal{A} \rangle$. Let us then consider the stationary semantics of the set of sentences $T = \{\mathrm{p} \vee \mathrm{q}\}$. Now $\overline{\mathcal{S}'[T]}(\emptyset) = \{\mathrm{p}, \mathrm{q}\}$ and the set of potential theorems $\mathcal{S}'[T](\{\mathrm{p}, \mathrm{q}\}) = \mathrm{Cn}(T \cup \{\neg\mathrm{p}, \neg\mathrm{q}\}) = \mathcal{L}$ is propositionally inconsistent. Consequently, the stationary semantics $\mathrm{STAT}_{\mathcal{S}'}(T)$ is just the set of logical consequences $\mathrm{Cn}(T)$. □

Let us further analyze what happens if stationary semantics is applied to the subsystem $\mathcal{S} = \mathcal{C}_P^{\mathcal{A}}$ for a set of atoms $P \subseteq \mathcal{A}$ to be minimized. For the initial set of non-theorems $N = \emptyset$ in the iterative construction of the least $\mathcal{S}[T]$-stationary set of non-theorems, the set of potential non-theorems $N' = \{\phi \in P \,|\, \phi \notin \mathrm{Cn}(T)\}$ is constructed and the set of potential theorems $\mathrm{Cn}(T \cup \neg N')$ is considered. If this is consistent, we obtain N' as the least fixed point of $\overline{\mathcal{S}[T]}^2$ and the resulting set of theorems is $\mathrm{Cn}(T \cup \neg N')$. Otherwise, we obtain $N = \emptyset$ as the least fixed point of $\overline{\mathcal{S}[T]}^2$ and $\mathrm{Cn}(T)$ as the corresponding set of theorems. Surprisingly, the set of theorems $\mathrm{Cn}(T \cup \neg N')$ above coincides with the simplest form of Reiter's *closed world assumption* [93] on a propositional theory T if $P = \mathcal{A}$. As we demonstrated in Example 4.29, making such closed world assumption is not always successful. A successful case is considered in Example 4.30. We note also that in this example, the stationary semantics of T' coincides with the stable semantics of T'. This indicates that stationary semantics provides a reasonable approximation of stable semantics in circumscription, too.

Example 4.30 Consider the stationary semantics of $T' = \{\mathrm{p} \vee \mathrm{q}, \mathrm{p} \rightarrow \mathrm{q}\}$ for the system $\mathcal{S}'$ in Example 4.29. Now q is a logical consequence of T' directly. Thus $N' = \overline{\mathcal{S}'[T'](\emptyset)} = \{\mathrm{p}\}$ and $\mathcal{S}[T'](N') = \mathrm{Cn}(T' \cup \{\neg\mathrm{p}\}) = \mathrm{Cn}(\{\neg\mathrm{p}, \mathrm{q}\})$. Thus we obtain N' as the least $\mathcal{S}[T']$-stationary set of non-theorems. Thus $\mathrm{STAT}_{\mathcal{S}}(T') = \mathrm{Cn}(\{\neg\mathrm{p}, \mathrm{q}\})$ is different from $\mathrm{Cn}(T')$. □

Since the minimization principle of circumscription is easily translated into a set of defaults, it is no wonder that the above example is easily converted to an example demonstrating similar behavior of a default system. For a set of defaults D, stationary semantics is in danger to collapse, e.g. if for some sentence $\phi \in \mathcal{L}$, both ϕ and $\neg\phi$ are consequents in D. Then it is possible that the set of conclusions $\mathrm{Cn}^R(T)$ is propositionally inconsistent for the set of inference rules $R = \{\frac{\alpha}{\gamma} \mid \frac{\alpha:\beta_1,\ldots,\beta_n}{\gamma} \in D\}$. This particular set of inference rules R is obtained as the reduction D_N in the first round of the stationary construction, if $N = \mathcal{L} - \mathrm{Cn}^{D_\emptyset}(T)$ contains $\neg\mathrm{J}(D)$. This is the case if each justification of $\mathrm{J}(D)$ is consistent with the closure $\mathrm{Cn}^{D_\emptyset}(T)$ based on the monotone rules (justification-free) $D_\emptyset = \{\frac{\alpha}{\gamma} \mid \frac{\alpha:}{\gamma} \in D\}$ of D only. However, there are interesting sets of defaults that avoid this consistency problem and a satisfactory notion of stationary semantics results. For instance, a translation of normal logic programs proposed by Bidoit and Froidevaux [5] produces sets of defaults with a non-collapsing stationary semantics as shown by Przymusinska and Przymusinski [89]. The resulting semantics coincides with the well-founded semantics of normal logic programs [115].

For the rest of our analysis of stationary semantics, we want to distinguish non-monotonic systems for which stationary semantics collapses systematically, i.e. for all set of sentences $T \subseteq \mathcal{L}^+(\mathcal{S})$. We have formulated this case as a property STCOLL below. Unlike our earlier requirements for non-monotonic systems, we suggest that a reasonable system $\mathcal{S}$ does *not* have this property.

Property	For all $T \subseteq \mathcal{L}^+(\mathcal{S})$:
(STCOLL)	$\mathrm{STAT}_{\mathcal{S}}(T) = \mathcal{S}[T](\emptyset)$.

We show in Example 4.31 that the requirement STAT fails for practically all non-monotonic systems associated with standard autoepistemic theories. These include translations of normal logic programs into autoepistemic theories suggested by Gelfond and Lifschitz [32]. Consequently, the well-founded semantics of normal logic programs [115] is not properly captured. Example 4.31 can also be extended for *non-monotonic modal logics* studied by McDermott [69] as well as Marek, Schwartz and Truszczyński [64]. These logics lead to non-monotonic systems analogous to autoepistemic systems, but with the exception that a monotonic modal logic S is used for positive introspection.

Example 4.31 Consider an autoepistemic theory $\Sigma \subseteq \mathcal{L}_{\mathrm{B}}$ with a single modality $\mathbf{B}$. The definition of Marek and Truszczyński's iterative expansions [66] suggests the corresponding non-monotonic system $\boldsymbol{\mathcal{A}}_\Sigma$ defined by $\boldsymbol{\mathcal{A}}_\Sigma(T,N) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \neg\mathbf{B}N)$ for all $T \subseteq \mathcal{L}_{\mathrm{B}}$ and $N \subseteq \mathcal{L}_{\mathrm{B}}$. However, the system $\boldsymbol{\mathcal{A}}_\Sigma$ does not have a satisfactory notion of stationary semantics, because the set of potential theorems is $\mathcal{L}^+(\boldsymbol{\mathcal{A}}_\Sigma) = \mathcal{L}_{\mathrm{B}}$ in cases of interest.

If $\mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T) = \mathcal{L}_{\mathrm{B}}$, it is immediately clear that the stationary semantics $\mathrm{STAT}_{\boldsymbol{\mathcal{A}}_\Sigma}(T)$ collapses to $\boldsymbol{\mathcal{A}}_\Sigma[T](\emptyset) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T)$. Let us then assume that $\mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T) \neq \mathcal{L}_{\mathrm{B}}$. Consequently, the set of potential non-theorems $N = \overline{\boldsymbol{\mathcal{A}}_\Sigma[T](\emptyset)} = \mathcal{L}_{\mathrm{B}} - \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T)$ is non-empty. Then we would like to pick a sentence $\phi \in N$ such that also $\neg\mathbf{B}\phi \in N$. (i) If there is ϕ with the desired property, both $\neg\mathbf{B}\neg\mathbf{B}\phi$ and $\mathbf{B}\neg\mathbf{B}\phi$ belong to the set of potential theorems $\boldsymbol{\mathcal{A}}_\Sigma[T](\overline{\boldsymbol{\mathcal{A}}_\Sigma[T](\emptyset)}) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \neg\mathbf{B}N)$ which is therefore propositionally inconsistent and equals to $\mathcal{L}^+(\boldsymbol{\mathcal{A}}_\Sigma) = \mathcal{L}_{\mathrm{B}}$. (ii) If there is no ϕ with the desired property, then $\neg\mathbf{B}\phi \in \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T)$ holds for every $\phi \in N$. It follows that $\mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \neg\mathbf{B}N)$ and $N = \overline{\boldsymbol{\mathcal{A}}_\Sigma[T](N)}$. Consequently, N is $\boldsymbol{\mathcal{A}}_\Sigma[T]$-stable as well as $\boldsymbol{\mathcal{A}}_\Sigma[T]$-stationary[22]. By the cases (i) and (ii), the set of theorems $\mathrm{STAT}_{\boldsymbol{\mathcal{A}}_\Sigma}(\emptyset) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T)$ regardless of the theories Σ and T. We have therefore established that $\mathrm{STAT}_{\boldsymbol{\mathcal{A}}_\Sigma}(T) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T)$ for all $T \subseteq \mathcal{L}_{\mathrm{B}}$. This indicates that $\boldsymbol{\mathcal{A}}_\Sigma$ has the unwanted property STCOLL. □

It is demonstrated in Example 4.32 that our bimodal variant of autoepistemic avoids the consistency problem addressed above. In Section 6, it will be shown that the translation function of Section 3.4 that maps default theories to autoepistemic ones preserves stationary semantics. Thus the well-founded semantics of normal logic programs is captured by the results of Przymusinska and Przymusinski [89]. The translation required is obtained as a composition of the translation by Bidoit and Froidevaux and the translation Tr_{K} of Definition 3.29. This is one of the major advantages of our reconstruction of autoepistemic logic with separate modalities for belief and disbelief.

Example 4.32 Consider $\Sigma = \{\mathbf{D}\mathrm{p} \rightarrow \mathrm{p}, \mathbf{D}\mathrm{q} \rightarrow \mathrm{r}\}$ having no separated stable expansions. The least $\boldsymbol{\mathcal{A}}_\Sigma$-stationary set of non-theorems gives rise to the set of theorems $\Delta_1 = \{\mathbf{D}\mathrm{q}, \mathrm{r}, \mathbf{B}\mathrm{r}, \ldots\}$ and the set of potential theorems $\Delta_2 = \Delta_1 \cup \{\mathbf{D}\mathrm{p}, \mathrm{p}, \mathbf{B}\mathrm{p}, \ldots\}$ which is propositionally consistent, but belief inconsistent.[23] Note also that r can be concluded from this expansion. By the analysis in Example 4.31, we know that such a conclusion cannot be drawn if conventional autoepistemic logic is used in the construction (stationary semantics of $\Sigma' = \{\neg\mathbf{B}\mathrm{p} \rightarrow \mathrm{p}, \neg\mathbf{B}\mathrm{q} \rightarrow \mathrm{r}\}$ collapses to the closure $\mathrm{Cn}_{\mathbf{B}}(\Sigma')$). □

[22] This case is not of practical value, since $\neg\mathbf{B}\phi$ should be $\mathbf{B}$-provable from the autoepistemic theory $\Sigma \cup T$ for every ϕ in a stable set of non-theorems N.

[23] The pair $\langle \Delta_1, \Delta_2 \rangle$ where $\Delta_1 \subseteq \Delta_2$ is a stationary expansion of Σ in the sense of [47].

Let us compare two further approaches with ours. Lin and Shoham [60] propose the *logic of knowledge and justified assumptions* GK. As suggested by the name of the logic, modalities **K** for knowledge and **A** for assumption are used. The semantics of GK is based on Kripke-style models for the two modalities. In particular, Lin and Shoham impose a preference order on models and the semantics of a theory in GK is determined by its *preferred models.* The system GK captures propositionally consistent extensions of default theories by a translation $\mathbf{K}\alpha \wedge \neg\mathbf{A}\neg\beta_1 \wedge ... \wedge \neg\mathbf{A}\neg\beta_n \rightarrow \mathbf{K}\gamma$. Let us now consider the default theory $\langle\{\frac{:\neg \mathrm{p}}{p}\}, \emptyset\rangle$. Lin and Shoham translate the above default into $\mathbf{K}\top \wedge \neg\mathbf{A}\neg\neg\mathrm{p} \rightarrow \mathbf{K}\mathrm{p}$. In GK, this is equivalent to $\neg\mathbf{A}\mathrm{p} \rightarrow \mathbf{K}\mathrm{p}$. This sentence does not have *preferred* models [60, Example 3.4]. This indicates that the system GK is not capable of capturing stationary extensions, unless the underlying Kripke-style models are modified.

Yuan [120] proposes a variant of autoepistemic logic. Our comparison will restrict to the propositional case. In Yuan's approach, the scope of the belief operator **B** is restricted to propositional atoms whereas separated autoepistemic logic is based on a full modal language. To embed default logic in his system, Yuan translates a default rule to a sentence $\mathbf{B}\mathrm{p}_\alpha \wedge \neg\mathbf{B}\mathrm{p}_{\bar{\beta}_1} \wedge ... \wedge \neg\mathbf{B}\mathrm{p}_{\bar{\beta}_\mathrm{n}} \rightarrow \gamma$ where p_α and $p_{\bar{\beta}_1}, ..., p_{\bar{\beta}_n}$ are new atoms. Moreover, he needs additional equivalences $p_\alpha \leftrightarrow \alpha$ and $p_{\bar{\beta}_i} \leftrightarrow \neg\beta_i$ for $0 < i \leq n$. Compared to this, our approach is much simpler: we can do without extra atoms because of the full modal language and the separate modality **D** for disbeliefs. In addition, Yuan's results cover only Reiter's extensions and there is no indication whether Przymusinska and Przymusinski's stationary extensions are properly captured. We see the following obstacle for this to be the case. Consider the set of defaults $D = \{\frac{:\neg \mathrm{p}}{\mathrm{p}}, \frac{:\mathrm{q}}{\mathrm{q}}\}$. Yuan translates this set of defaults into a set of autoepistemic sentences $T = \{\neg\mathbf{B}\mathrm{p}' \rightarrow \mathrm{p}, \mathrm{p}' \leftrightarrow \neg\neg\mathrm{p}, \neg\mathbf{B}\mathrm{q}' \rightarrow \mathrm{q}, \mathrm{q}' \leftrightarrow \neg\mathrm{q}\}$ where p' and q' are new atoms introduced in the translation. Yuan's notion of well-founded autoepistemic expansions E of a set of premises T is based on an equation $E = \{\phi \,|\, T \cup \{\neg\mathbf{B}\mathrm{a} \,|\, T \cup \{\neg\mathbf{B}\mathrm{a} \,|\, a \notin E\} \not\vdash_\mathbf{B} \mathrm{a}\} \vdash_\mathbf{B} \phi\}$.[24] We argue that $E = \{\phi \,|\, T \vdash_\mathbf{B} \phi\}$ is the least well-founded expansion for T above. To see this, note that p' and q' do not belong to E so that $\neg\mathbf{B}\mathrm{p}'$ and $\neg\mathbf{B}\mathrm{q}'$ are added to T in the above equation. Then p' becomes a logical consequence of the theory $T \cup \{\neg\mathbf{B}\mathrm{p}', \neg\mathbf{B}\mathrm{q}'\}$ which is therefore inconsistent under the necessitation rule ($\mathbf{B}\mathrm{p}'$ can also be derived). Thus $\{\neg\mathbf{B}\mathrm{a} \,|\, T \cup \{\neg\mathbf{B}\mathrm{a} \,|\, a \notin E\} \not\vdash_\mathbf{B} \mathrm{a}\} = \emptyset$ and $E = \{\phi \,|\, T \vdash_\mathbf{B} \phi\}$. Then $\mathrm{q} \notin E$ holds in contrast to the least stationary extension $E' = \mathrm{Cn}(\{\mathrm{q}\})$ of the default theory $\langle D, \emptyset\rangle$. So it seems that the consistency problem addressed Example 4.31 arises in Yuan's approach, too.

[24] The necessitation rule is restricted to the belief atoms $\mathbf{B}\mathrm{a}$ in the language of T.

5 Structural Properties of Semantics

Despite our specification of non-monotonic systems includes a monotonicity condition, the semantics introduced in Section 4 are inherently non-monotonic. More precisely, for each semantics considered so far, the operator $\mathrm{SEM}_{\mathbf{S}}$ is not a monotonic one. This raises the question whether there are other properties that these semantics satisfy or ought to satisfy. Such analysis of the *structural properties* of non-monotonic reasoning has been carried out by e.g. Gabbay [31], Makinson [63], Kraus, Lehmann and Magidor [56], Thiele [110] and Dix [19, 20]. The properties considered in these works provide a basis for classifying various semantics proposed for non-monotonic logics. Moreover, these works suggest that *cumulativity* is one of the basic properties to be met by any semantics. Unfortunately, this is not the case in general. For instance, default logic is not cumulative under stable semantics. In spite of this, the cumulativity property has been considered so important that a number of modifications to default logic have been proposed to enforce cumulativity (see e.g. approaches by Brewka [11], Risch [96] and Schaub [99]). The rough idea in these approaches is to change the interpretation of the justifications from their interpretation under Reiter's approach to gain the cumulativity property.

We address the cumulativity property as well as some related properties as follows. In Section 5.1, we present the rules of cumulative inference proposed by Kraus, Lehmann and Magidor [56] and reformulate the properties constituting cumulativity (cumulative closure and cumulative monotonicity) for our abstract framework. The reformulation aims at making the definitions of the properties independent on syntax and at extending the definitions to handle sets of sentences rather than single sentences. A similar process goes on when some weaker properties related to cumulative inference are addressed in Section 5.2. It is shown that stationary and stable semantics have these straightforward properties. Sections 5.3 and 5.4 concentrate on the cumulative closure and cumulative monotonicity properties, respectively. These properties are addressed in conjunction with stable semantics and stationary semantics. It turns out that both semantics are cumulatively closed, but only stationary semantics is cumulatively monotonic. A property of non-monotonic systems named as *local consistency property* is identified in Section 5.5. It is shown that stable semantics is cumulative for non-monotonic systems sharing this property. Also systems having the local consistency property are presented. Possible weakenings of the cumulativity property are considered in Section 5.6. The methodology developed in this section is later applied in the analysis of cautious semantics in Section 8.

$$\frac{\alpha \mid\!\sim \beta \quad \alpha \wedge \beta \mid\!\sim \gamma}{\alpha \mid\!\sim \gamma} \qquad \frac{\alpha \mid\!\sim \beta \quad \alpha \mid\!\sim \gamma}{\alpha \wedge \beta \mid\!\sim \gamma}$$

(cut) (cautious monotonicity)

$$\alpha \mid\!\sim \alpha \qquad \frac{\models \alpha \leftrightarrow \beta \quad \alpha \mid\!\sim \gamma}{\beta \mid\!\sim \gamma} \qquad \frac{\models \alpha \rightarrow \beta \quad \gamma \mid\!\sim \alpha}{\gamma \mid\!\sim \beta}$$

(reflexivity) (left logical equivalence) (right weakening)

Figure 4: Rules for cumulative inference

5.1 Definition of Cumulativity

Kraus, Lehmann and Magidor [56, Definition 3.1] formulate rules for cumulative inference involving a non-monotonic consequence relation $\mid\!\sim$. The consequence relation $\mid\!\sim$ is assumed to relate *single* sentences ($\alpha \mid\!\sim \beta$) rather than a sentence with a set of sentences ($\Sigma \mid\!\sim \beta$). The rules *cut*, *cautious monotonicity*, *reflexivity*, *left logical equivalence*, and *right weakening* are given in Figure 4. The last three of these are taken into consideration in Section 5.2. Meanwhile, we concentrate on formulating the cut rule and the cautious monotonicity rule for our framework. Let us emphasize the role played by the sentence β in these inference rules. This sentence is a non-monotonic consequence which is used as a *lemma* when the sentence γ is inferred from the sentence α. The cut rule can be interpreted as a requirement that non-monotonic conclusions obtained with lemmas are also obtainable without. In other words, the use of lemmas is *sound*. The cautious monotonicity rule states the corresponding completeness property: non-monotonic conclusions are not lost if lemmas are introduced. Altogether, the two rules dictate that lemmas do not interfere the underlying non-monotonic inference mechanism and lemmas can be freely used in non-monotonic inference governed by the consequence relation $\mid\!\sim$.

We adopt the approach of Makinson [63] and Thiele [110] to formulate the properties constituting cumulativity for inference operators. In the following two definitions, we single out *cumulatively closed* and *cumulatively monotonic* inference operators. An inference operator $\mathbb{S}$ is *cumulative*, if it is reflexive in the sense given in Table 1 and both cumulatively closed and monotonic[25]. If this is the case, then $T \subseteq T' \subseteq \mathbb{S}(T)$ implies $\mathbb{S}(T') = \mathbb{S}(T)$ for all $T \subseteq T' \subseteq \mathcal{L}(\mathbb{S})$. A semantics for a non-monotonic system $\mathbb{S}$ is cumulative if the corresponding semantic operator $\mathrm{SEM}_{\mathbb{S}} : \mathbf{2}^{\mathcal{L}^+(\mathbb{S})} \to \mathbf{2}^{\mathcal{L}^+(\mathbb{S})}$ is cumulative.

[25] Thiele [110] uses equivalently closure instead of cumulative closure.

Definition 5.1 *An inference operator* $\mathcal{S} : \mathbf{2}^{\mathcal{L}(\mathcal{S})} \rightarrow \mathbf{2}^{\mathcal{L}(\mathcal{S})}$ *is cumulatively closed iff* $T \subseteq T' \subseteq \mathcal{S}(T)$ *implies* $\mathcal{S}(T') \subseteq \mathcal{S}(T)$ *for all* $T \subseteq T' \subseteq \mathcal{L}(\mathcal{S})$.

Definition 5.2 *An inference operator* $\mathcal{S} : \mathbf{2}^{\mathcal{L}(\mathcal{S})} \rightarrow \mathbf{2}^{\mathcal{L}(\mathcal{S})}$ *is cumulatively monotonic iff* $T \subseteq T' \subseteq \mathcal{S}(T)$ *implies* $\mathcal{S}(T) \subseteq \mathcal{S}(T')$ *for all* $T \subseteq T' \subseteq \mathcal{L}(\mathcal{S})$.

The cut rule and the cautious monotonicity rule in Figure 4 correspond to the properties given in Definitions 5.1 and 5.2, respectively. The same principles are embodied in these definitions, but for sets of sentences rather than single sentences: the sentences in $T' - T$ can be viewed as lemmas. Moreover, the definitions are independent on the structure of sentences, i.e. syntax. For instance, the connective $\wedge$ is not assigned a distinguished role. As stated by Makinson [63] the approach based on single sentences is inherently finitary while the other is not. For finite theories, the two approaches are compatible.

The requirements for a closure operator are stronger than those for a cumulative operator: it is shown in Proposition 5.3 that closure operators are cumulative operators while the converse does not hold in general. Anyway, the definitions of closure operators and cumulative operators are closely related, and by the relationship given in Proposition 5.3 it is justifiable to view cumulative operators as weak closure operators.

Proposition 5.3 *Every closure operator* $\mathcal{S}$ *is also a cumulative operator.*

Proof. First of all, every closure operator is reflexive by definition. Then let $T \subseteq T' \subseteq \mathcal{S}(T)$ hold. It follows that $\mathcal{S}(T) \subseteq \mathcal{S}(T')$ and $\mathcal{S}(T') \subseteq \mathcal{S}(\mathcal{S}(T))$ by the requirement MON. Moreover, the set of theorems $\mathcal{S}(\mathcal{S}(T))$ equals to $\mathcal{S}(T)$ by the requirement CLO. This implies that $\mathcal{S}(T') \subseteq \mathcal{S}(T)$ as well. □

Proposition 5.3 is also applicable for non-monotonic systems $\mathcal{S}$ (see Corollary 5.4), since the operator $\lambda T.\mathcal{S}(T, N)$ is a closure operator for fixed N. We have listed the properties CCLO and CMON for non-monotonic systems in Table 4. The cumulativity property CUM is the conjunction of CCLO and CMON.

Corollary 5.4 *A non-monotonic system* $\mathcal{S}$ *satisfies the requirements* CCLO, CMON *and* CUM *in Table 4.*

5.2 Weak Properties Related to Cumulativity

Three additional rules for cumulative inference were presented in Figure 4: reflexivity, left logical equivalence and right weakening. In the following, we consider mapping these rules to our framework. The resulting properties for

Property	For every $T \subseteq T' \subseteq \mathcal{L}^+(\mathcal{S})$ and $N \subseteq \mathcal{L}^-(\mathcal{S})$:
(CCLO)	$T \subseteq T' \subseteq \mathcal{S}(T,N)$ implies $\mathcal{S}(T',N) \subseteq \mathcal{S}(T,N)$
(CMON)	$T \subseteq T' \subseteq \mathcal{S}(T,N)$ implies $\mathcal{S}(T,N) \subseteq \mathcal{S}(T',N)$ and
(CUM)	$T \subseteq T' \subseteq \mathcal{S}(T,N)$ implies $\mathcal{S}(T',N) = \mathcal{S}(T,N)$.

Table 4: Derived properties of non-monotonic systems

non-monotonic systems are relatively weak. As an indication of this, both stationary and stable semantics are shown to share these properties in a straightforward way. First of all, the reflexivity rule coincides with the notion of reflexivity that we introduced for monotonic systems as the requirement REF in Table 1. The semantic operators $\mathrm{STAT}_{\mathcal{S}}$ and $\mathrm{STAB}_{\mathcal{S}}$ have this property:

Theorem 5.5 *Stationary semantics* $\mathrm{STAT}_{\mathcal{S}}$ *and stable semantics* $\mathrm{STAB}_{\mathcal{S}}$ *are reflexive.*

Proof. Let $T \subseteq \mathcal{L}^+(\mathcal{S})$ be any set of sentences. Then $T \subseteq \mathcal{S}(T,N)$ holds for all $\mathcal{S}[T]$-stationary and $\mathcal{S}[T]$-stable sets of non-theorems N by the requirement REF. Thus $T \subseteq \mathrm{STAT}_{\mathcal{S}}(T)$ and $T \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$ for all $T \subseteq \mathcal{L}^+(\mathcal{S})$. □

The rule of left logical equivalence raises the connective $\leftrightarrow$ to a distinguished role. Thus the rule makes only sense for systems based on propositional logic having this particular connective. Rather than imposing such a restriction on the systems under consideration, we describe what logical equivalence means from the point of view of a system $\mathcal{S}$. Moreover, we consider the equivalence of sets of sentences rather than single sentences in analogy to our notion of cumulativity. Recall that we considered single sentences already in Definition 3.19. These observations suggest the notion of $\mathcal{S}$-equivalence of Definition 5.6 which is then characterized by Proposition 5.7. Notice also that T_1 and T_2 need not to be in the subset relation in order to be $\mathcal{S}$-equivalent.

Definition 5.6 *The sets of sentences* $T_1 \subseteq \mathcal{L}^+(\mathcal{S})$ *and* $T_2 \subseteq \mathcal{L}^+(\mathcal{S})$ *and* $\mathcal{S}$-*equivalent if and only if it holds for all* $T \subseteq \mathcal{L}^+(\mathcal{S})$ *and* $N \subseteq \mathcal{L}^+(\mathcal{S})$ *that*

$$T_1 \subseteq \mathcal{S}(T,N) \text{ if and only if } T_2 \subseteq \mathcal{S}(T,N).$$

Proposition 5.7 *The sets of sentences* $T_1 \subseteq \mathcal{L}^+(\mathcal{S})$ *and* $T_2 \subseteq \mathcal{L}^+(\mathcal{S})$ *and* $\mathcal{S}$-*equivalent if and only if* $\mathcal{S}(T_1,N) = \mathcal{S}(T_2,N)$ *for all* $N \subseteq \mathcal{L}^-(\mathcal{S})$.

Proof. ($\Rightarrow$) Assume that T_1 and T_2 are $\mathcal{S}$-equivalent and let N be any subset of $\mathcal{L}^-(\mathcal{S})$. Let us show that $\mathcal{S}(T_1,N) = \mathcal{S}(T_2,N)$. Now $T_1 \subseteq \mathcal{S}(T_1,N)$

holds by REF so that $T_2 \subseteq \mathcal{S}(T_1, N)$ by the $\mathcal{S}$-equivalence of T_1 and T_2. Then $\mathcal{S}(T_2, N) \subseteq \mathcal{S}(\mathcal{S}(T_1, N), N)$ holds by the requirement MON. It follows that $\mathcal{S}(T_2, N) \subseteq \mathcal{S}(T_1, N)$ by CLO. The converse inclusion $\mathcal{S}(T_1, N) \subseteq \mathcal{S}(T_2, N)$ follows by the symmetry of $\mathcal{S}$-equivalence. ($\Leftarrow$) Let $\mathcal{S}(T_1, N) = \mathcal{S}(T_2, N)$ hold for all N. Then let T and N be any subsets of $\mathcal{L}^+(\mathcal{S})$ and $\mathcal{L}^-(\mathcal{S})$, respectively. Then assume that $T_1 \subseteq \mathcal{S}(T, N)$ holds. It follows that $\mathcal{S}(T_1, N) \subseteq \mathcal{S}(\mathcal{S}(T, N), N)$ by the requirement MON. Thus CLO implies $\mathcal{S}(T_1, N) \subseteq \mathcal{S}(T, N)$. Since $\mathcal{S}(T_1, N) = \mathcal{S}(T_2, N)$ holds, we obtain $\mathcal{S}(T_2, N) \subseteq \mathcal{S}(T, N)$. Then $T_2 \subseteq \mathcal{S}(T, N)$ follows, as $T_2 \subseteq \mathcal{S}(T_2, N)$ by the requirement REF. To conclude, we have established that $T_1 \subseteq \mathcal{S}(T, N)$ implies $T_2 \subseteq \mathcal{S}(T, N)$. The converse follows by symmetry. □

To give an example of $\mathcal{S}$-equivalent sets of sentences, consider an autoepistemic system $\mathcal{A}_\Sigma$ induced by an autoepistemic theory $\Sigma = \{\mathrm{a}\}$. Now $T = \{\mathrm{b}\}$ and $T' = \{\mathrm{a} \rightarrow \mathrm{b}, \mathbf{B}\mathrm{b}\}$ are $\mathcal{A}_\Sigma$-equivalent sets of sentences, but obviously not propositionally equivalent. On the other hand, it is clear that $\models \alpha \leftrightarrow \beta$ implies that $T = \{\alpha\}$ and $T' = \{\beta\}$ are $\mathcal{S}$-equivalent for any system $\mathcal{S}$ based on a propositional closure. These examples demonstrate that our interpretation of the non-monotonic inference rule is looser than that by Kraus et al. [56]. This makes Theorem 5.8 stronger in the sense that it is not restricted to propositionally equivalent sentences only.

Theorem 5.8 *If $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $T' \subseteq \mathcal{L}^+(\mathcal{S})$ are $\mathcal{S}$-equivalent sets of sentences, then* $\mathrm{STAT}_{\mathcal{S}}(T) = \mathrm{STAT}_{\mathcal{S}}(T')$ *and* $\mathrm{STAB}_{\mathcal{S}}(T) = \mathrm{STAB}_{\mathcal{S}}(T')$.

Proof. Assume that a set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}[T]$-stable, i.e. $N = \overline{\mathcal{S}[T]}(N)$. Since T and T' are $\mathcal{S}$-equivalent, we obtain $\mathcal{S}[T](N) = \mathcal{S}(T, N) = \mathcal{S}(T', N) = \mathcal{S}[T'](N)$ by Proposition 5.7. It follows that $\overline{\mathcal{S}[T']}(N) = \overline{\mathcal{S}[T]}(N)$ so that $N = \overline{\mathcal{S}[T']}(N)$ is $\mathcal{S}$-stable. By the symmetry of $\mathcal{S}$-equivalence, it holds that N is $\mathcal{S}[T]$-stable if and only if N is $\mathcal{S}[T']$-stable. It can be similarly shown that N is $\mathcal{S}[T]$-stationary if and only if N is $\mathcal{S}[T']$-stationary.

Then assume that $\phi \in \mathrm{STAB}_{\mathcal{S}}(T)$, i.e. $\phi \in \mathcal{S}[T](N)$ for an $\mathcal{S}[T]$-stable set of non-theorems N. By the above N is also $\mathcal{S}[T']$-stable. Moreover, it follows that $\phi \in \mathcal{S}[T'](N)$ by the $\mathcal{S}$-equivalence of T and T' and Proposition 5.7. Thus $\phi \in \mathrm{STAB}_{\mathcal{S}}(T')$. We have therefore established that $\mathrm{STAB}_{\mathcal{S}}(T) \subseteq \mathrm{STAB}_{\mathcal{S}}(T')$. The converse inclusion follows by symmetry so that $\mathrm{STAB}_{\mathcal{S}}(T) = \mathrm{STAB}_{\mathcal{S}}(T')$ holds. The equality $\mathrm{STAT}_{\mathcal{S}}(T) = \mathrm{STAT}_{\mathcal{S}}(T')$ is proved similarly. □

The rule of right weakening is handled very similarly. In Definition 5.9, we have formulated a condition for sets of sentences that captures also the relationship $\models \phi \rightarrow \psi$ when we have a system based on propositional logic under consideration (i.e. $\{\phi\}$ $\mathcal{S}$-entails $\{\psi\}$). Our notion of $\mathcal{S}$-entailment is compatible with our notion of $\mathcal{S}$-equivalence introduced in Definition 5.6: if T_1 $\mathcal{S}$-entails T_2 and T_2 $\mathcal{S}$-entails T_1, then T_1 and T_2 are $\mathcal{S}$-equivalent.

Definition 5.9 *A set of sentences* $T_1 \subseteq \mathcal{L}^+(\mathcal{S})$ $\mathcal{S}$*-entails a set of sentences* $T_2 \subseteq \mathcal{L}^+(\mathcal{S})$ *if and only if it holds for all* $T \subseteq \mathcal{L}^+(\mathcal{S})$ *and* $N \subseteq \mathcal{L}^+(\mathcal{S})$ *that*

$$T_1 \subseteq \mathcal{S}(T, N) \textit{ implies } T_2 \subseteq \mathcal{S}(T, N).$$

Theorem 5.10 *If a set of sentences* $T_1 \subseteq \mathcal{L}^+(\mathcal{S})$ $\mathcal{S}$*-entails a set of sentences* $T_2 \subseteq \mathcal{L}^+(\mathcal{S})$*, then* $T_1 \subseteq \mathrm{STAT}_{\mathcal{S}}(T)$ *implies* $T_2 \subseteq \mathrm{STAT}_{\mathcal{S}}(T)$ *and (ii)* $T_1 \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$ *implies* $T_2 \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$.

Proof. (i) Assume that $T_1 \subseteq \mathrm{STAT}_{\mathcal{S}}(T)$. It follows that $T_1 \subseteq \mathcal{S}[T](N)$ and $T_1 \subseteq \mathcal{S}(T, N)$, for every $\mathcal{S}[T]$-stationary set of non-theorems N. Since T_2 is $\mathcal{S}$-entailed by T_1, we obtain $T_2 \subseteq \mathcal{S}(T, N)$ and $T_2 \subseteq \mathcal{S}[T](N)$ for every $\mathcal{S}[T]$-stationary set of non-theorems N. Hence $T_2 \subseteq \mathrm{STAT}_{\mathcal{S}}(T)$ holds. The claim (ii) is proved similarly, but N has to be $\mathcal{S}[T]$-stable. □

5.3 Semantics Satisfying Cumulative Closure

Let us now apply the machinery introduced in Section 5.1 to verify that stable and stationary semantics are cumulatively closed. In particular, we remind that $\mathcal{S}[T]$ denotes a non-monotonic system $\mathcal{S}$ having the sentences of T fixed as theorems. This notion is central with cumulative closure and monotonicity properties, since we intend to fix the sentences in the difference $T' - T$ (i.e. the lemmas discussed in the beginning of Section 5.1) as theorems in the system $\mathcal{S}$. Lemma 5.11 is needed in the sequel to prove cumulative closure for stable and stationary semantics.

Lemma 5.11 *For a set of non-theorems* N *and a set of theorems* $T \subseteq \mathcal{S}(N)$*, (i)* $\mathcal{S}(N) = \mathcal{S}[T](N)$ *and (ii)* $\overline{\mathcal{S}}(N) = \overline{\mathcal{S}[T]}(N)$.

Proof. Because $T \subseteq \mathcal{S}(N)$, the set of theorems $\mathcal{S}(N)$ equals to $\mathcal{S}(T, N) = \mathcal{S}[T](N)$ by the property CUM. Thus $\mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N) = \mathcal{L}^-(\mathcal{S}) - \mathcal{S}[T](N) = \mathcal{L}^-(\mathcal{S}[T]) - \mathcal{S}[T](N)$ and $\overline{\mathcal{S}}(N) = \overline{\mathcal{S}[T]}(N)$ follows. □

It is shown in Proposition 5.12 that we do not lose a $\mathcal{S}$-stable set of non-theorems N by fixing a set of sentences $T \subseteq \mathcal{S}(N)$ as theorems, i.e. N is also $\mathcal{S}[T]$-stable. This is a key property of $\mathcal{S}$-stable sets of non-theorems in order to establish the cumulative closure property in Theorem 5.13.

Proposition 5.12 *Let* N *be an* $\mathcal{S}$*-stable sets of non-theorems. Then* N *is* $\mathcal{S}[T]$*-stable for all sets of theorems* $T \subseteq \mathcal{S}(N)$.

Proof. The $\mathcal{S}$-stability of N implies $N = \overline{\mathcal{S}}(N)$. Since $T \subseteq \mathcal{S}(N)$, we obtain $\overline{\mathcal{S}}(N) = \overline{\mathcal{S}[T]}(N)$ by Lemma 5.11. It follows that $N = \overline{\mathcal{S}[T]}(N)$ holds and therefore N is $\mathcal{S}[T]$-stable by Definition 4.8. □

Theorem 5.13 *Stable semantics* $\mathrm{STAB}_{\mathcal{S}}$ *is cumulatively closed.*

Proof. Let $T \subseteq T' \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$ hold. It it then proved that $\mathrm{STAB}_{\mathcal{S}}(T') \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$. So let $\phi \notin \mathrm{STAB}_{\mathcal{S}}(T)$ hold. Then there is a $\mathcal{S}[T]$-stable set of non-theorems N such that $\phi \notin \mathcal{S}[T](N)$. Since $T' \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$, it follows that $T' \subseteq \mathcal{S}[T](N)$ by the definition of $\mathrm{STAB}_{\mathcal{S}}(T)$ and the $\mathcal{S}[T]$-stability of N. Thus by Proposition 5.12, the set of non-theorems N is $\mathcal{S}[T][T']$-stable and $\mathcal{S}[T](N) = \mathcal{S}[T][T'](N)$ by Lemma 5.11. Because $T \subseteq T'$, the non-monotonic systems $\mathcal{S}[T][T']$ and $\mathcal{S}[T']$ coincide so that N is $\mathcal{S}[T']$-stable and $\mathcal{S}[T](N) = \mathcal{S}[T'](N)$. Thus $\phi \notin \mathcal{S}[T'](N)$ and $\phi \notin \mathrm{STAB}_{\mathcal{S}}(T')$. □

Let us then take stationary semantics into consideration. Proposition 5.14 is an analog of Proposition 5.12 for stationary semantics. We omit the proof of Theorem 5.15 which is obtained in a straightforward way from that of Theorem 5.13 by replacing stable sets of non-theorems and Proposition 5.12 by stationary sets of non-theorems and Proposition 5.14, respectively.

Proposition 5.14 *Let N be an $\mathcal{S}$-stationary set of non-theorems. Then N is $\mathcal{S}[T]$-stationary for all sets of theorems $T \subseteq \mathcal{S}(N)$.*

Proof. Now $N = \overline{\mathcal{S}}^2(N)$ and $N \subseteq \overline{\mathcal{S}}(N)$ hold, since N is $\mathcal{S}$-stationary. Because $T \subseteq \mathcal{S}(N)$, we obtain $\overline{\mathcal{S}}(N) = \overline{\mathcal{S}[T]}(N)$ by Lemma 5.11. It follows that $N \subseteq \overline{\mathcal{S}[T]}(N)$. Since $N \subseteq \overline{\mathcal{S}}(N)$, the set of theorems T is also contained in $\mathcal{S}(\overline{\mathcal{S}}(N))$ by the requirement MON so that $\overline{\mathcal{S}}(\overline{\mathcal{S}}(N)) = \overline{\mathcal{S}[T]}(\overline{\mathcal{S}}(N))$ by Lemma 5.11. This implies $N = \overline{\mathcal{S}}^2(N) = \overline{\mathcal{S}[T]}^2(N)$ by the equality $\overline{\mathcal{S}}(N) = \overline{\mathcal{S}[T]}(N)$. Hence N is $\mathcal{S}[T]$-stationary. □

Theorem 5.15 *Stationary semantics* $\mathrm{STAT}_{\mathcal{S}}$ *is cumulatively closed.*

5.4 Semantics Satisfying Cumulative Monotonicity

Previously, we proved that both semantics proposed in Section 4 are cumulatively closed. As indicated by Dix [20], this is typically the case if a semantics is defined in terms of a fixed point condition. This applies to the semantics addressed so far, implying that the results of Section 5.3 are not surprising in this respect. However, the question whether a semantics satisfies the other half of the cumulativity, namely the cumulative monotonicity property, is quite a different matter. The following example is a counter-example for stable semantics. Originally, this example was formulated for normal logic programs by

Van Gelder, Ross and Schlipf [115], but we present an autoepistemic version of the example instead.

Example 5.16 Consider an autoepistemic theory $\Sigma = \{\mathbf{D}\mathrm{b} \rightarrow \mathrm{a}, \mathbf{D}\mathrm{a} \rightarrow \mathrm{b}, \mathbf{D}\mathrm{p} \rightarrow \mathrm{p}, \mathbf{B}\mathrm{a} \rightarrow \mathrm{p}\}$ and the subsystem $\mathcal{S} = \mathcal{A}^{\Sigma}_{\Sigma}$ having the language

$$\langle\{\mathrm{a}, \mathrm{b}, \mathrm{p}\}, \{\mathrm{a}, \mathrm{b}, \mathrm{p}\}\rangle.$$

There is a unique $\mathcal{S}$-stable set of non-theorems $N_1 = \{\mathrm{b}\}$. This gives rise to the set of theorems $\mathcal{S}(N_1) = \{\mathrm{a}, \mathrm{p}\}$. However, if we fix the set of theorems $T = \{\mathrm{p}\}$, there are two $\mathcal{S}[T]$-stable sets of non-theorems, namely N_1 and $N_2 = \{\mathrm{a}\}$ that give rise to sets of theorems $\mathcal{S}[T](N_1) = \{\mathrm{a}, \mathrm{p}\}$ and $\mathcal{S}[T](N_2) = \{\mathrm{b}, \mathrm{p}\}$. Consequently, we obtain $\mathrm{STAB}_{\mathcal{S}}(\emptyset) = \{\mathrm{a}, \mathrm{p}\}$ and $\mathrm{STAB}_{\mathcal{S}}(T) = \{\mathrm{p}\}$. Since $\emptyset \subseteq T \subseteq \mathrm{STAB}_{\mathcal{S}}(\emptyset)$, but not $\mathrm{STAB}_{\mathcal{S}}(\emptyset) \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$, the operator $\mathrm{STAB}_{\mathcal{S}}$ is not cumulatively monotonic nor cumulative. $\square$

The lack of cumulative monotonicity (and therefore also cumulativity) has been one of the major objections to stable semantics, since cumulativity seems a plausible property of a semantic operator and it is obvious that such an operator cannot be monotonic. Let us briefly explain why $\mathrm{STAB}_{\mathcal{S}}$ cannot be cumulatively monotonic. A fundamental obstacle is that the existence of $\mathcal{S}$-stable non-theorems is not guaranteed as it was shown in Example 4.15. The above example can even be simplified to $\Sigma = \{\mathbf{D}\mathrm{p} \rightarrow \mathrm{p}\}$ if we consider the system $\mathcal{S} = \mathcal{A}_{\Sigma}$. Then $\mathrm{STAB}_{\mathcal{S}}(\emptyset) = \mathcal{L}_{\mathrm{BD}}$, but for $T = \{\mathrm{p}\} \subseteq \mathcal{L}_{\mathrm{BD}}$ there is a $\mathcal{S}[T]$-stable set of non-theorems N that does not contain p. The corresponding set of theorems $\mathcal{S}[T](N)$ is propositionally consistent and therefore does not contain e.g. $\mathrm{p} \wedge \neg\mathrm{p}$. Thus $\mathrm{STAB}_{\mathcal{S}}(\emptyset)$ is not a subset of $\mathrm{STAB}_{\mathcal{S}}(T)$. However, the existence of $\mathcal{S}$-stable sets of non-theorems can be guaranteed in restricted cases and also the cumulativity of stable semantics can be established. Such systems will be addressed in Section 5.5.

Let us now address stationary semantics and show that it has the cumulative monotonicity property. Our proof will be based on the fixed point characterization of the least $\mathcal{S}$-stationary set of non-theorems provided by Theorem 4.28. In Lemma 5.17, it is shown that the set of theorems T (to be fixed) is included in the set of potential theorems $\mathcal{S}(\overline{\mathcal{S}}(N_\alpha))$ throughout the construction of N by the iteration $N_\alpha = \overline{\mathcal{S}}^2 \uparrow^\alpha (\emptyset)$. It is then shown in Proposition 5.18 that fixing a set of theorems $T \subseteq \mathcal{S}(N)$ based on the least $\mathcal{S}$-stationary set of non-theorems N does not obstruct the reconstruction of N, i.e. N is contained in the least $\mathcal{S}[T]$-stationary set of non-theorems.

Lemma 5.17 *Let $N = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$, $T \subseteq \mathcal{S}(N)$ and $N_\alpha = \overline{\mathcal{S}}^2 \uparrow^\alpha (\emptyset)$ for all ordinals α. Then $T \subseteq \mathcal{S}(\overline{\mathcal{S}}(N_\alpha))$ for all ordinals α.*

Proof. Let α be any ordinal. By Proposition 4.27, the set of non-theorems N_α is contained in N. Thus $\overline{\mathcal{S}}(N) \subseteq \overline{\mathcal{S}}(N_\alpha)$ follows by Proposition 4.25. Since $N \subseteq \overline{\mathcal{S}}(N)$ by Theorem 4.28, it follows that $N \subseteq \overline{\mathcal{S}}(N_\alpha)$. Then $T \subseteq \mathcal{S}(N)$ implies $T \subseteq \mathcal{S}(\overline{\mathcal{S}}(N_\alpha))$ by the requirement MON. □

Proposition 5.18 *Let* $N = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$ *be the least* $\mathcal{S}$*-stationary set of non-theorems and* $T \subseteq \mathcal{S}(N)$ *a set of theorems. Then* $N \subseteq \mathrm{lfp}(\overline{\mathcal{S}[T]}^2, \emptyset)$.

Proof. Define $M = \mathrm{lfp}(\overline{\mathcal{S}[T]}^2, \emptyset)$ and $N_\alpha = \overline{\mathcal{S}}^2 \uparrow^\alpha (\emptyset)$ for all ordinals α. Let us then prove by transfinite induction on α that $N_\alpha \subseteq M$. The base case $\alpha = 0$ holds, since N_0 is empty by definition. Induction step is as follows.

Let α be any successor ordinal. By the inductive hypothesis $N_{\alpha-1} \subseteq M$. It follows by Proposition 4.25 that $\overline{\mathcal{S}[T]}^2(N_{\alpha-1})$ is a subset of $\overline{\mathcal{S}[T]}^2(M) = M$. By the requirement MON, we have that $\mathcal{S}(N_{\alpha-1}) \subseteq \mathcal{S}(T, N_{\alpha-1}) = \mathcal{S}[T](N_{\alpha-1})$ which implies $\overline{\mathcal{S}[T]}(N_{\alpha-1}) \subseteq \overline{\mathcal{S}}(N_{\alpha-1})$. It follows that $\overline{\mathcal{S}[T]}(\overline{\mathcal{S}}(N_{\alpha-1}))$ is a subset of $\overline{\mathcal{S}[T]}^2(N_{\alpha-1})$ by Proposition 4.25. Thus $\overline{\mathcal{S}[T]}(\overline{\mathcal{S}}(N_{\alpha-1})) \subseteq M$.

On the other hand, the set of theorems $T \subseteq \mathcal{S}(\overline{\mathcal{S}}(N_{\alpha-1}))$ by Lemma 5.17. Consequently, the set of theorems $\mathcal{S}[T](\overline{\mathcal{S}}(N_{\alpha-1})) = \mathcal{S}(\overline{\mathcal{S}}(N_{\alpha-1}))$ by the derived property CUM and furthermore $\overline{\mathcal{S}[T]}(\overline{\mathcal{S}}(N_{\alpha-1})) = \overline{\mathcal{S}}^2(N_{\alpha-1})$. It follows that $N_\alpha = \overline{\mathcal{S}}^2(N_{\alpha-1})$ is a subset of M for all ordinals α.

Finally, consider an infinite limit ordinal α. It is clear that $N_\alpha = \bigcup_{\beta<\alpha} N_\beta$ is a subset of M, since $N_\beta \subseteq M$ for all $\beta < \alpha$ by the inductive hypothesis. Hence $N_\alpha \subseteq M$ for all ordinals α and consequently N is a subset of M. □

The cumulativity of the semantic operator $\mathrm{STAT}_\mathcal{S}$ is shown in Theorem 5.19. Dix [20] has earlier proved Theorem 5.19 for the well-founded semantics of normal logic programs and Przymusinska and Przymusinski [89] have proved a result corresponding to Proposition 5.18 for default logic. Our results extend the same properties for the bimodal autoepistemic logic proposed in this work as well as any unforeseen non-monotonic systems conforming to our specification. Corollary 5.20 is a direct consequence of Theorems 5.15, 5.19 and 5.5 (the reflexivity of $\mathrm{STAT}_\mathcal{S}$ as shown in Section 5.2).

Theorem 5.19 *Stationary semantics* $\mathrm{STAT}_\mathcal{S}$ *is cumulatively monotonic.*

Proof. Let $T \subseteq T' \subseteq \mathrm{STAT}_\mathcal{S}(T)$. It is then proved that $\mathrm{STAT}_\mathcal{S}(T) \subseteq \mathrm{STAT}_\mathcal{S}(T')$. So let $\phi \in \mathrm{STAT}_\mathcal{S}(T)$ hold. Then $\phi \in \mathcal{S}[T](N)$ follows by Theorem 4.28 for the least $\mathcal{S}[T]$-stationary set of non-theorems N. Since $T' \subseteq \mathrm{STAT}_\mathcal{S}(T)$, it follows that also $T' \subseteq \mathcal{S}[T](N)$. By Proposition 5.18, the set $N \subseteq N'$ where N' is the least $\mathcal{S}[T][T']$-stationary set of non-theorems. Because $T \subseteq T'$, it follows by the definition of fixing that N' is the least $\mathcal{S}[T']$

stationary set of non-theorems. Also $\phi \in \mathcal{S}[T'](N')$ by the requirement MON. This implies that $\phi \in \mathrm{STAT}_{\mathcal{S}}(T')$ by Theorem 4.28. $\square$

Corollary 5.20 *Stationary semantics* $\mathrm{STAT}_{\mathcal{S}}$ *is cumulative.*

Corollary 5.20 leaves an interesting setting for further research, since it holds by Theorem 4.24 that $\mathrm{STAT}_{\mathcal{S}} \sqsubseteq \mathrm{STAB}_{\mathcal{S}}$ where $\mathrm{STAT}_{\mathcal{S}}$ is cumulative by Corollary 5.20, but $\mathrm{STAB}_{\mathcal{S}}$ is not cumulative by Example 5.16. The question is whether there is a *reasonable* cumulative semantics $\mathrm{SEM}_{\mathcal{S}}$ such that $\mathrm{STAT}_{\mathcal{S}} \sqsubseteq \mathrm{SEM}_{\mathcal{S}} \sqsubseteq \mathrm{STAB}_{\mathcal{S}}$. This matter is further discussed when the design decisions of cautious semantics are made in Section 8.

5.5 Local Consistency Property

In the preceding section, we provided a counter-example which shows that stable semantics $\mathrm{STAB}_{\mathcal{S}}$ is not cumulatively monotonic in general. In Definition 5.21, we introduce a property of non-monotonic systems $\mathcal{S}$ that allows us to prove the existence of $\mathcal{S}$-stable sets of non-theorems and to establish cumulative monotonicity for $\mathrm{STAB}_{\mathcal{S}}$. The intuition behind the *local consistency property* is that given $T \subseteq \mathcal{L}^{+}(\mathcal{S})$ the condition $\phi \notin \mathcal{S}[T](N)$ is a sufficient and necessary to extend an $\mathcal{S}[T]$-consistent set of non-theorems N to $N \cup \{\phi\}$ such that $\mathcal{S}[T]$-consistency is not lost.

Definition 5.21 *A non-monotonic system* $\mathcal{S}$ *has the local consistency property if and only if it holds for all* $T \subseteq \mathcal{L}^{+}(\mathcal{S})$, $N \subseteq \mathcal{L}^{-}(\mathcal{S})$ *and* $\phi \in \mathcal{L}^{-}(\mathcal{S})$ *that if* N *is* $\mathcal{S}[T]$*-consistent and* $\phi \notin \mathcal{S}[T](N)$*, then* $N \cup \{\phi\}$ *is* $\mathcal{S}[T]$*-consistent.*

The local consistency property allows us to extend any $\mathcal{S}[T]$-consistent set of non-theorems into an $\mathcal{S}[T]$-stable one. In the following, we introduce a systematic construction to carry out such a completion. For this purpose we need (possibly transfinite) *enumerations* of a set of sentences S. An enumeration is effectively a well-ordering of S and therefore isomorphic to an ordinal, the *index of* ε, denoted by $\mathrm{Ind}(\varepsilon)$. The idea is that $|\mathrm{Ind}(\varepsilon)| = |S|$ and for each successor ordinal $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$ there is a unique sentence ε_{α} of S *enumerated* by ε. Given an enumeration ε, and ordinals $0 \leq \alpha_1 \leq \mathrm{Ind}(\varepsilon)$ and $0 \leq \alpha_2 \leq \mathrm{Ind}(\varepsilon)$, we write $\varepsilon_{\alpha_1 \ldots \alpha_2}$ to denote the *sub-enumeration* of ε that enumerates all ε_{β} such that β is a successor ordinal in the range $\alpha_1 \leq \beta \leq \alpha_2$ in the same order as ε. Sometimes we use also the notation $\{\varepsilon\}$ to denote the set $\{\varepsilon_{\alpha} \mid 0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$ and α is a successor ordinal$\}$ enumerated by ε. The enumerations of a finite set S are naturally finite. To demonstrate the transfinite case in terms of an example, consider $S = \{\mathrm{p}_1, \mathrm{q}_1, \mathrm{p}_2, \mathrm{q}_2, \ldots\}$ and its enumeration $\varepsilon = \mathrm{p}_1, \mathrm{p}_2, \ldots; \mathrm{q}_1, \mathrm{q}_2, \ldots$[26] which consists of a concatenation of two infinite enumerations. The index $\mathrm{Ind}(\varepsilon)$ of the enumeration ε is $\omega + \omega$, but the cardinality $|\omega + \omega| = \omega = |S|$. The set $\{\varepsilon_{0 \ldots \omega}\}$ is $\{\mathrm{p}_1, \mathrm{p}_2, \ldots\}$.

A set of non-theorems N is systematically extended to $\mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N)$ as follows. The definition is formulated for a non-monotonic system $\mathcal{S}$, but note that the definition covers also any system $\mathcal{S}[T]$ where $T \subseteq \mathcal{L}^{+}(\mathcal{S})$ by Proposition 2.40.

Definition 5.22 *Let $N \subseteq \mathcal{L}^{-}(\mathcal{S})$ be a set of non-theorems and ε an enumeration of a subset of $\mathcal{L}^{-}(\mathcal{S}) - N$. Define $\mathrm{Nt}^{\varepsilon,0}_{\mathcal{S}}(N) = N$ and for all successor ordinals α in the range $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$, the set of non-theorems*

$$\mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}}(N) = \begin{cases} \mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}}(N) \cup \{\varepsilon_\alpha\}, & \text{if } \varepsilon_\alpha \notin \mathcal{S}(\mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}}(N)) \text{ and} \\ \mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}}(N), & \text{otherwise.} \end{cases}$$

For all limit ordinals α in the range $0 < \alpha \leq \mathrm{Ind}(\varepsilon)$, define the set of non-theorems

$$\mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}}(N) = \bigcup_{\beta < \alpha} \mathrm{Nt}^{\varepsilon,\beta}_{\mathcal{S}}(N).$$

and finally the set of non-theorems $\mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N) = \mathrm{Nt}^{\varepsilon,\mathrm{Ind}(\varepsilon)}_{\mathcal{S}}(N)$.

The construction of Definition 5.22 is an analog of Lindenbaum's construction for classical (modal) logic to build maximal consistent theories (see, e.g., [41, p. 20]). A similar construction has been used by Niemelä and the author in conjunction with autoepistemic theories [48, 78] and normal logic programs [44]. We may now apply the enumeration-based construction to establish the cumulativity of stable semantics for non-monotonic systems with the local consistency property. In Lemma 5.23 we extend an $\mathcal{S}[T]$-consistent set of non-theorems such that $\mathcal{S}[T]$-consistency is preserved. The resulting set of non-theorems $N' = \mathrm{Nt}^{\varepsilon}_{\mathcal{S}[T]}(N)$ is shown to be $\mathcal{S}[T]$-complete in Lemma 5.24.

Lemma 5.23 *Let $\mathcal{S}$ have the local consistency property and let $T \subseteq \mathcal{L}^{+}(\mathcal{S})$. If N is an $\mathcal{S}[T]$-consistent set of non-theorems, then the set of non-theorems $\mathrm{Nt}^{\varepsilon}_{\mathcal{S}[T]}(N)$ is $\mathcal{S}[T]$-consistent for all enumerations ε of $\mathcal{L}^{-}(\mathcal{S}[T]) - N$.*

Proof. Let ε be any enumeration of $\mathcal{L}^{-}(\mathcal{S}[T]) - N$. It is then proved by transfinite induction for all $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$ that $\mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}[T]}(N)$ is $\mathcal{S}[T]$-consistent. The base case $\alpha = 0$ holds, since $\mathrm{Nt}^{\varepsilon,0}_{\mathcal{S}[T]}(N) = N$ is $\mathcal{S}[T]$-consistent by assumption.

Induction step follows. Let α be any successor ordinal. By the inductive hypothesis $\mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}[T]}(N)$ is $\mathcal{S}[T]$-consistent. There are two cases to consider. (i) Firstly, let $\varepsilon_\alpha \in \mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}[T]}(N)$ hold. By Definition 5.22 this happens only

[26] Limits are denoted by semicolons.

if $\varepsilon_\alpha \notin \mathcal{S}[T](\mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}[T]}(N))$. Since $\mathcal{S}$ has the local consistency property and $\mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}[T]}(N)$ is $\mathcal{S}[T]$-consistent, it follows that $\mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}[T]}(N)$ is $\mathcal{S}[T]$-consistent. (ii) Otherwise, the set $\mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}[T]}(N) = \mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}[T]}(N)$ by Definition 5.22. Thus $\mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}[T]}(N)$ is directly $\mathcal{S}[T]$-consistent. By the cases (i) and (ii) $\mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}[T]}(N)$ is $\mathcal{S}[T]$-consistent. Then consider an infinite limit ordinal α. By Definition 5.22, the sequence $\mathrm{Nt}^{\varepsilon,\beta}_{\mathcal{S}[T]}(N)$ indexed by $\beta < \alpha$ is non-decreasing. Moreover, the set $\mathrm{Nt}^{\varepsilon,\beta}_{\mathcal{S}[T]}(N)$ is $\mathcal{S}[T]$-consistent for all $\beta < \alpha$ by the inductive hypothesis. Thus

$$\mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}[T]}(N) = \bigcup_{\beta<\alpha} \mathrm{Nt}^{\varepsilon,\beta}_{\mathcal{S}[T]}(N)$$

is $\mathcal{S}[T]$-consistent by Lemma 4.26 and Proposition 4.9. By this inductive proof, the set of non-theorems $\mathrm{Nt}^{\varepsilon}_{\mathcal{S}[T]}(N)$ is $\mathcal{S}[T]$-consistent for all ordinals $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$ and thus $\mathrm{Nt}^{\varepsilon}_{\mathcal{S}[T]}(N) = \mathrm{Nt}^{\varepsilon,\mathrm{Ind}(\varepsilon)}_{\mathcal{S}[T]}(N)$ is $\mathcal{S}[T]$-consistent. □

Lemma 5.24 *Let $N \subseteq \mathcal{L}^-(\mathcal{S})$ be a set of non-theorems. Then the set of non-theorems $\mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N)$ is $\mathcal{S}$-complete for all enumerations ε of $\mathcal{L}^-(\mathcal{S}) - N$.*

Proof. Assume that $\mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N)$ is not $\mathcal{S}$-complete. Then there is $\phi \in \mathcal{L}^-(\mathcal{S})$ such that $\phi \notin \mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N)$ and $\phi \notin \mathcal{S}(\mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N))$. The former implies that $\phi \notin N$. Then ϕ is necessarily included in ε, say as the element ε_α for a successor ordinal α. Now $\varepsilon_\alpha \notin \mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N)$ implies $\varepsilon_\alpha \notin \mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}}(N)$ by Definition 5.22. But then $\phi \in \mathcal{S}(\mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}}(N))$ must be the case. Because $\mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}}(N) \subseteq \mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N)$ by definition, we obtain $\phi \in \mathcal{S}(\mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N))$ by the requirement MON, a contradiction. □

By Lemmas 5.23 and 5.24 any $\mathcal{S}[T]$-consistent set of non-theorems can be extended to an $\mathcal{S}[T]$-stable one using the enumeration-based technique. Of course this holds only if the non-monotonic system $\mathcal{S}$ has the local consistency property. For such non-monotonic systems the existence of at least one $\mathcal{S}[T]$-stable set of non-theorems is guaranteed, since $\emptyset$ is always an $\mathcal{S}[T]$-consistent set of non-theorems. The cumulative monotonicity property is established for the semantic operator $\mathrm{STAB}_{\mathcal{S}}$ in Proposition 5.25 and Theorem 5.26.

Proposition 5.25 *Let $\mathcal{S}$ have the local consistency property of Definition 5.21. Let $T \subseteq T' \subseteq \mathcal{L}^+(\mathcal{S})$ be sets of theorems such that $T' \subseteq \mathcal{S}[T](N)$ for all $\mathcal{S}[T]$-stable sets of non-theorems N. Then for all $N \subseteq \mathcal{L}^-(\mathcal{S})$,*

if N is $\mathcal{S}[T']$-stable, N is also $\mathcal{S}[T]$-stable.

Proof. Let N be $\mathcal{S}[T']$-stable so that $N = \overline{\mathcal{S}[T']}(N)$ holds. By the requirement MON, the set of theorems $\mathcal{S}[T](N) = \mathcal{S}(T, N)$ is a subset of $\mathcal{S}[T'](N) = \mathcal{S}(T', N)$ implying that $\overline{\mathcal{S}[T']}(N) \subseteq \overline{\mathcal{S}[T]}(N)$. Thus $N \subseteq \overline{\mathcal{S}[T]}(N)$, i.e. N is

$\mathcal{S}[T]$-consistent. Since $\mathcal{S}$ has the local consistency property and N is $\mathcal{S}[T]$-consistent, the set of non-theorems N can be extended to N' which is $\mathcal{S}[T]$-stable by Lemmas 5.23 and 5.24 and Corollary 4.11. It remains to be shown that also $N' \subseteq N$ so that N and N' coincide. Since $T' \subseteq \mathcal{S}[T](N')$ by assumption and N' is $\mathcal{S}[T]$-stable, it follows that $N' = \overline{\mathcal{S}[T']}(N')$ holds by Proposition 5.12. Moreover, it holds by Proposition 4.25 that $\overline{\mathcal{S}[T']}(N') \subseteq \overline{\mathcal{S}[T']}(N)$, because $N \subseteq N'$. Thus $N' = \overline{\mathcal{S}[T']}(N') \subseteq N = \overline{\mathcal{S}[T']}(N)$. $\square$

Theorem 5.26 *Stable semantics* $\mathrm{STAB}_{\mathcal{S}}$ *is cumulatively monotonic if* $\mathcal{S}$ *has the local consistency property of Definition 5.21.*

Proof. Let $T \subseteq T' \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$ hold. It it then proved that $\mathrm{STAB}_{\mathcal{S}}(T) \subseteq \mathrm{STAB}_{\mathcal{S}}(T')$. So let $\phi \notin \mathrm{STAB}_{\mathcal{S}}(T')$ hold. Then there is a $\mathcal{S}[T']$-stable set of non-theorems N such that $\phi \notin \mathcal{S}[T'](N)$. Since $T' \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$, it follows by the definition of $\mathrm{STAB}_{\mathcal{S}}(T)$ that $T' \subseteq \mathcal{S}[T](N')$ for all $\mathcal{S}[T]$-stable sets of non-theorems N'. Thus N is $\mathcal{S}[T]$-stable by Proposition 5.25. Now $\phi \notin \mathcal{S}[T'](N)$ implies $\phi \notin \mathcal{S}[T](N)$ by the requirement MON so that $\phi \notin \mathrm{STAB}_{\mathcal{S}}(T)$. $\square$

Theorems 5.13, 5.26 and 5.5 (which gives the reflexivity of stable semantics) have the following corollary.

Corollary 5.27 *Stable semantics* $\mathrm{STAB}_{\mathcal{S}}$ *is cumulative if* $\mathcal{S}$ *has the local consistency property of Definition 5.21.*

Having proved the cumulativity of the semantic operator $\mathrm{STAB}_{\mathcal{S}}$ for non-monotonic systems $\mathcal{S}$ having the local consistency property, our next objective is to present non-monotonic systems with this property. So let us begin with minimal model systems $\mathcal{C}_P$ induced by sets of atoms $P \subseteq \mathcal{A}$. Our proposition considers only a subsystem of the minimal model system $\mathcal{C}_P$. It is to be expected that the techniques to be introduced in Section 6 extend our restricted cumulativity result for the unrestricted system $\mathcal{C}_P$.

Proposition 5.28 *Let* $P \subseteq \mathcal{A}$ *be a set of atoms to be minimized. The system* $\mathcal{C}_P^P$ *has the local consistency property.*

Proof sketch. The proof is obtained from that of Proposition 5.30 (to be presented later) using the translation $D = \mathrm{Tr}_{\mathrm{E}}(P) = \{\frac{:\neg\phi}{\neg\phi} \mid \phi \in P\}$ and

$$\mathcal{S} = \mathcal{C}_P^P \text{ and } \mathcal{S}' = \mathfrak{D}_D^{\neg \mathrm{J}(D)}.$$

It holds for any $T \subseteq \mathcal{L}$ that (i) a set of non-theorems $N \subseteq P$ is $\mathcal{S}[T]$-consistent if and only if $\neg\neg N \subseteq \neg \mathrm{J}(D) = \neg\neg P$ is $\mathcal{S}'[T]$-consistent and (ii) $\phi \notin \mathcal{S}[T](N)$ if and only if $\neg\neg\phi \notin \mathcal{S}'[T](\neg\neg N)$ for all $\phi \in P$. By the properties (i) and (ii), the system $\mathcal{S}$ has the local consistency property, since $\mathcal{S}'$ has. $\square$

Our counter-example for the cumulativity of stable semantics presented in Section 5.4 applies to default systems as well. Therefore we would like to impose syntactic restrictions to reach cumulativity in limited cases. This is the aim of the normality condition in Definition 5.29.[27] However, Makinson [63] has shown that stable semantics is not cumulative even if only normal sets of defaults are considered. Prerequisites play a crucial role in his counter-example. This suggests a further restriction to be imposed on defaults: the rules should be prerequisite-free as well.

Definition 5.29 *A set of defaults D is normal if its rules are of the form $\frac{\alpha:\beta}{\beta}$.*

In Proposition 5.30, it turns out that these syntactic restrictions are sufficient to establish the local consistency and the cumulativity of stable semantics in conjunction with restricted default systems.

Proposition 5.30 *Let D be set of defaults which is normal and prerequisite-free. The system $\mathfrak{D}_D^{\neg\mathrm{J}(D)}$ has the local consistency property.*

Proof. The system $\mathcal{S} = \mathfrak{D}_D^{\neg\mathrm{J}(D)}$ has the language $\mathcal{L}(\mathcal{S}) = \langle\mathcal{L}, \neg\mathrm{J}(D)\rangle$.

Let $T \subseteq \mathcal{L}$ be any set of sentences. Let $N \subseteq \neg\mathrm{J}(D)$ be an $\mathcal{S}[T]$-consistent set of non-theorems and $\neg\beta \in \neg\mathrm{J}(D)$ such that $\neg\beta \notin \mathcal{S}[T](N)$. Because the rules of D are normal and prerequisite free, the reduction $D_N = \{\frac{\top}{\beta'} \mid \neg\beta' \in N\}$. It follows that $\mathrm{Cn}^{D_N}(T) = \mathrm{Cn}(T \cup N')$ where $N' = \{\beta' \mid \neg\beta' \in N\}$ and $\neg\beta \notin \mathrm{Cn}(T \cup N')$. This implies that $T \cup N' \cup \{\beta\}$ is propositionally consistent. Let us then assume that $N \cup \{\neg\beta\}$ is not $\mathcal{S}[T]$-consistent. So there is a sentence $\neg\beta' \in N \cup \{\neg\beta\}$ such that $\neg\beta' \in \mathcal{S}[T](N \cup \{\beta\}) = \mathrm{Cn}^{D_{N\cup\{\neg\beta\}}}(T)$. Since $\neg\beta$ belongs to $\neg\mathrm{J}(D)$ and the default rules of D are normal and prerequisite-free, it follows that the reduct $D_{N\cup\{\neg\beta\}} = D_N \cup \{\frac{\top}{\beta}\}$. Thus $\mathrm{Cn}^{D_{N\cup\{\neg\beta\}}}(T)$ equals to $\mathrm{Cn}(T \cup N' \cup \{\beta\})$. It follows that $\neg\beta' \in \mathrm{Cn}(T \cup N' \cup \{\beta\})$, i.e. β' is not consistent with $T \cup N' \cup \{\beta\}$. Since $\beta' \in N' \cup \{\beta\}$, this is a contradiction. Hence the set of non-theorems $N \cup \{\neg\beta\}$ is $\mathcal{S}[T]$-consistent. $\square$

5.6 Weaker Forms of Cumulativity

In the previous section, we considered the local consistency property which allowed us to establish cumulativity in restricted cases. In this section, we follow a different approach and consider possible weakenings of the cumulativity property. A very straightforward solution is obtained by considering sentences

[27]This can be weakened to a requirement that default are of the form $\frac{\alpha:\beta}{\beta'}$ where β and β' are propositionally equivalent sentences.

ϕ that appear in every set of theorems $\mathcal{S}(T, N)$ induced by sets $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $N \subseteq \mathcal{L}^-(\mathcal{S})$. Such sentences are named as $\mathcal{S}$-tautologies in Definition 5.31. It is easy to establish by the requirement MON that $\phi \in \mathcal{L}^+(\mathcal{S})$ is an $\mathcal{S}$-tautology if and only if $\phi \in \mathcal{S}(\emptyset, \emptyset)$.

Definition 5.31 *A sentence $\phi \in \mathcal{L}^+(\mathcal{S})$ is an $\mathcal{S}$-tautology if $\phi \in \mathcal{S}(T, N)$ for all $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $N \subseteq \mathcal{L}^-(\mathcal{S})$.*

To give an example of an $\mathcal{S}$-tautology we note that $\mathbf{B}(\mathbf{B}\mathrm{a} \rightarrow \mathrm{b} \vee \neg\mathrm{b})$ is $\mathcal{A}_\Sigma$-tautology for the autoepistemic theory $\Sigma = \{\mathrm{a}\}$. These kinds of sentences do not interfere with our notion of $\mathcal{S}$-stability as shown in Proposition 5.32 and a weaker notion of cumulativity can be established for stable semantics.

Proposition 5.32 *If $T \subseteq \mathcal{L}^+(\mathcal{S})$ is a set of $\mathcal{S}$-tautologies and $N \subseteq \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}[T]$-stable, then N is also $\mathcal{S}$-stable.*

Proof. Let N be $\mathcal{S}[T]$-stable. Since T is a set of $\mathcal{S}$-tautologies, it follows that $T \subseteq \mathcal{S}(N)$. Thus $\mathcal{S}[T](N) = \mathcal{S}(T, N) = \mathcal{S}(N)$ by the property CUM. It follows that $\overline{\mathcal{S}[T]}(N) = \overline{\mathcal{S}}(N)$ and $N = \overline{\mathcal{S}}(N)$. Thus N is $\mathcal{S}$-stable. □

Theorem 5.33 *If $T \subseteq T' \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$ such that $T' - T$ is a set of $\mathcal{S}[T]$-tautologies, then $\mathrm{STAB}_{\mathcal{S}}(T) = \mathrm{STAB}_{\mathcal{S}}(T')$.*

Proof. ($\subseteq$) Let $\phi \notin \mathrm{STAB}_S(T')$, i.e. there is an $\mathcal{S}[T']$-stable set of non-theorems such that $\phi \notin \mathcal{S}[T'](N)$. Since $T' - T$ is a set of $\mathcal{S}[T]$-tautologies, we know that N is $\mathcal{S}[T]$-stable by Proposition 5.32 and $\phi \notin \mathcal{S}[T](N)$ by the requirement MON. Thus $\phi \notin \mathrm{STAB}_{\mathcal{S}}(T')$. ($\supseteq$) This holds by Theorem 5.13. □

The resulting notion of lemmas is relatively weak. However, quite similar principles have been introduced by Schaub [100, p. 307] in conjunction with constrained default logic. In his main example, Schaub adds a default rule $\frac{:\mathrm{a}}{\mathrm{a} \vee \mathrm{b}}$ (which is considered to be a lemma) to a set of default rules $D = \{\frac{:\mathrm{a}}{\mathrm{a}}, \frac{\mathrm{a} \vee \mathrm{b} : \neg\mathrm{a}}{\neg\mathrm{a}}\}$. As indicated by Schaub, the default extensions obtained with D and the extended set of defaults D' coincide. In order to understand the interconnection of Schaub's observation to Theorem 5.33, we resort to an autoepistemic interpretation of D. This is because the language $\mathcal{L}^+(\mathfrak{D}_D)$ consists of propositional sentences according to our definition of a default system $\mathfrak{D}_D$ and thus considering a default $\frac{:\mathrm{a}}{\mathrm{a} \vee \mathrm{b}}$ as a $\mathfrak{D}_D$-tautology makes no sense. Now the translation $\mathrm{Tr}_{\mathrm{K}}(D)$ is $\{\mathbf{B}\top \wedge \mathbf{D}\neg\mathrm{a} \rightarrow \mathrm{a}, \mathbf{B}(\mathrm{a} \vee \mathrm{b}) \wedge \mathbf{D}\neg\neg\mathrm{a} \rightarrow \neg\mathrm{a}\}$ and it is straightforward to see that the translation $\mathbf{B}\top \wedge \mathbf{D}\neg\mathrm{a} \rightarrow \mathrm{a} \vee \mathrm{b}$ of the default rule $\frac{:\mathrm{a}}{\mathrm{a} \vee \mathrm{b}}$ is $\mathbf{B}$-provable from $\mathrm{Tr}_{\mathrm{K}}(D)$. Therefore, the sentence $\mathbf{B}\top \wedge \mathbf{D}\neg\mathrm{a} \rightarrow \mathrm{a} \vee \mathrm{b}$ is an $\mathcal{A}_{\mathrm{Tr}_{\mathrm{K}}(D)}$-tautology and by Theorem 5.33 the stable semantics of $\mathcal{A}_{\mathrm{Tr}_{\mathrm{K}}(D)}$ is not affected when such an addition is made. Then we know indirectly by

Corollary 6.23 (to be established in Section 6) that the stable semantics of $\mathfrak{D}_D$ is not affected if the default rule $\frac{:\mathrm{a}}{\mathrm{a}\vee\mathrm{b}}$ is added in the set of defaults D.

Yet another possibility is to restrict the set of sentences T' in the definition of the cumulativity by a weaker semantics than stable semantics. This is what Schlipf does (as reported by Dix [20]) to establish the following theorem in the context of normal logic programs.

Theorem 5.34 *If* $T \subseteq T' \subseteq \mathrm{STAT}_{\mathcal{S}}(T)$, *then* $\mathrm{STAB}_{\mathcal{S}}(T) = \mathrm{STAB}_{\mathcal{S}}(T')$.

Proof sketch. ($\subseteq$) Let $\phi \notin \mathrm{STAB}_{\mathcal{S}}(T')$, i.e. there is an $\mathcal{S}[T']$-stable set of non-theorems N' such that $\phi \notin \mathcal{S}[T'](N')$. Define $N_\alpha = \overline{\mathcal{S}[T]}^2 \uparrow^\alpha (\emptyset)$ for all ordinals α. Using Lemma 5.17 it can be shown that $N_\alpha \subseteq N'$ for all ordinals α, because $T' - T \subseteq \mathcal{S}[T](N)$ for the least $\mathcal{S}[T]$-stationary set of non-theorems $N = \mathrm{lfp}(\overline{\mathcal{S}[T]}^2, \emptyset)$. Thus N is also contained in N'. It follows that $T' - T \subseteq \mathcal{S}[T](N')$ by the requirement MON. Thus $\mathcal{S}[T'](N') = \mathcal{S}[T](N')$ by the property CUM and $N' = \overline{\mathcal{S}[T]}(N')$ holds. Since also $\phi \notin \mathcal{S}[T](N')$ by the requirement MON, we obtain $\phi \notin \mathrm{STAB}_{\mathcal{S}}(T)$ as desired. ($\supseteq$) It holds by theorem 4.24 that $\mathrm{STAT}_{\mathcal{S}}(T) \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$. Thus $T \subseteq T' \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$ so that $\mathrm{STAB}_{\mathcal{S}}(T') \subseteq \mathrm{STAB}_{\mathcal{S}}(T)$ follows by Theorem 5.13. $\square$

6 Semantics Preserving Translations

We introduced three basic relations among non-monotonic systems in Section 3: subsystems, functional equivalences and embeddings. This section is devoted to analysing these relations in conjunction with the notions of $\mathcal{S}$-consistency and $\mathcal{S}$-completeness presented in Section 4.1.1 as well as semantics based on these notions. At the highest level we obtain results indicating that the syntactic translations of Section 3.4 preserve the semantics of involved non-monotonic systems. Our analysis proceeds as follows.

We study in Section 6.1 how the notions of $\mathcal{S}$-consistency, $\mathcal{S}$-completeness, $\mathcal{S}$-stability and $\mathcal{S}$-stationarity relate with subsystems $\mathcal{S}'$ of $\mathcal{S}$. For example, it is shown how a set of non-theorems N having one of these four properties with respect to the system $\mathcal{S}$ is converted into a set of non-theorems N' having the same property with respect to the subsystem $\mathcal{S}'$ and vice versa. A quite similar analysis goes on with functionally equivalent systems $\mathcal{S}$ and $\mathcal{S}'$ in Section 6.2. Since our notion of an embedding is modularly defined in terms of subsystems and functional equivalences our results are straightforwardly extended for embeddings in Section 6.3. The same applies to stable semantics and stationary semantics which are addressed in Section 6.4.

6.1 Subsystems

So let us begin with subsystems and the notions of $\mathcal{S}$-consistency and $\mathcal{S}$-completeness. We shall use the operator $\overline{\mathcal{S}}$ in this analysis, since we established the interconnection of $\overline{\mathcal{S}}$ and the notions of $\mathcal{S}$-consistency and $\mathcal{S}$-completeness in Propositions 4.9 and 4.10. Recall that a basic property of a subsystem $\mathcal{S}'$ of $\mathcal{S}$ is that $\mathcal{S}'(N')$ is obtained as $\mathcal{S}(N) \cap \mathcal{L}^+(\mathcal{S}')$ where $N \subseteq \mathcal{L}^-(\mathcal{S})$ is any set of non-theorems such that $N' = N \cap \mathcal{L}^-(\mathcal{S}')$. This is the "projection property" we imposed on sets of non-theorems in connection with subsystems. In Lemma 6.1, we establish the same interconnection for the operators $\overline{\mathcal{S}'}$ and $\overline{\mathcal{S}}$ induced by $\mathcal{S}'$ and $\mathcal{S}$, respectively. The corresponding result is also deduced for the operators $\overline{\mathcal{S}'}^2$ and $\overline{\mathcal{S}}^2$ in Lemma 6.2.

Lemma 6.1 *Let $\mathcal{S}$ be any non-monotonic system. If $\mathcal{S}' \sqsubseteq \mathcal{S}$, then it holds for all $N \subseteq \mathcal{L}^-(\mathcal{S})$ that $\overline{\mathcal{S}'}(N') = \overline{\mathcal{S}}(N) \cap \mathcal{L}^-(\mathcal{S}')$ where $N' = N \cap \mathcal{L}^-(\mathcal{S}')$.*

Proof. Let $\mathcal{S}' \sqsubseteq \mathcal{S}$ and N any subset of $\mathcal{L}^-(\mathcal{S})$. Define $N' = N \cap \mathcal{L}^-(\mathcal{S}')$. Now

$$
\begin{array}{rcll}
\overline{\mathcal{S}'}(N') & = & \mathcal{L}^-(\mathcal{S}') - \mathcal{S}'(N') & \text{Definition of } \overline{\mathcal{S}'} \\
& = & \mathcal{L}^-(\mathcal{S}') - (\mathcal{S}(N) \cap \mathcal{L}^+(\mathcal{S}')) & \text{Definition of } \mathcal{S}' \sqsubseteq \mathcal{S} \\
& = & (\mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N)) \cap \mathcal{L}^-(\mathcal{S}') & \mathcal{L}^-(\mathcal{S}') \subseteq \mathcal{L}^+(\mathcal{S}') \\
& = & \overline{\mathcal{S}}(N) \cap \mathcal{L}^-(\mathcal{S}'). & \text{Definition of } \overline{\mathcal{S}}
\end{array}
$$

Lemma 6.2 *Let $\mathcal{S}$ be any non-monotonic system. If $\mathcal{S}' \sqsubseteq \mathcal{S}$, then it holds for all $N \subseteq \mathcal{L}^-(\mathcal{S})$ that $\overline{\mathcal{S}'}^2(N') = \overline{\mathcal{S}}^2(N) \cap \mathcal{L}^-(\mathcal{S}')$ where $N' = N \cap \mathcal{L}^-(\mathcal{S}')$.*

Proof. Let $\mathcal{S}' \sqsubseteq \mathcal{S}$ and N any subset of $\mathcal{L}^-(\mathcal{S})$. Define $N' = N \cap \mathcal{L}^-(\mathcal{S}')$ and $M' = \overline{\mathcal{S}'}(N')$. It follows by Lemma 6.1 that $\overline{\mathcal{S}'}(N') = \overline{\mathcal{S}}(N) \cap \mathcal{L}^-(\mathcal{S}')$. Thus we let $M = \overline{\mathcal{S}}(N)$ so that $M' = M \cap \mathcal{L}^-(\mathcal{S}')$. Then $\overline{\mathcal{S}'}^2(N') = \overline{\mathcal{S}'}(M') = \overline{\mathcal{S}}(M) \cap \mathcal{L}^-(\mathcal{S}') = \overline{\mathcal{S}}^2(N) \cap \mathcal{L}^-(\mathcal{S}')$ holds by Lemma 6.1. □

Using Lemmas 6.1 and 6.2 we are able to show how various properties of a set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ are projected as properties of $N' = N \cap \mathcal{L}^-(\mathcal{S}')$.

Proposition 6.3 *Let $\mathcal{S}$ be any non-monotonic system and $\mathcal{S}' \sqsubseteq \mathcal{S}$ its subsystem. If $N \subseteq \mathcal{L}^-(\mathcal{S})$ is (i) $\mathcal{S}$-consistent, (ii) $\mathcal{S}$-complete, (iii) $\mathcal{S}$-stable or alternatively (iv) $\mathcal{S}$-stationary, then $N' = N \cap \mathcal{L}^-(\mathcal{S}')$ has the corresponding property with respect to the subsystem $\mathcal{S}'$.*

Proof. (i) Since N is $\mathcal{S}$-consistent, we obtain $N \subseteq \overline{\mathcal{S}}(N)$ by Proposition 4.9. It follows by Lemma 6.1 that $N' = N \cap \mathcal{L}(\mathcal{S}') \subseteq \overline{\mathcal{S}}(N) \cap \mathcal{L}^-(\mathcal{S}') = \overline{\mathcal{S}'}(N')$. This implies that N' is $\mathcal{S}'$-consistent by Proposition 4.9. (ii) This is proved as (i), but $\mathcal{S}$-completeness and $\mathcal{S}'$-completeness are captured by the inclusions $\overline{\mathcal{S}}(N) \subseteq N$ and $\overline{\mathcal{S}'}(N') \subseteq N'$, respectively (Proposition 4.10). (iii) This follows by Definition 4.8 and (i) and (ii) above. (iv) Since N is $\mathcal{S}$-stationary, it holds by Definition 4.19 that $N = \overline{\mathcal{S}}^2(N)$ and $N \subseteq \overline{\mathcal{S}}(N)$. It follows by Lemma 6.1 that $N' = N \cap \mathcal{L}^-(\mathcal{S}') \subseteq \overline{\mathcal{S}}(N) \cap \mathcal{L}^-(\mathcal{S}') = \overline{\mathcal{S}}(N')$ and by Lemma 6.2 that $N' = N \cap \mathcal{L}^-(\mathcal{S}') = \overline{\mathcal{S}}^2(N) \cap \mathcal{L}^-(\mathcal{S}') = \overline{\mathcal{S}'}^2(N')$. Consequently, the projection N' is $\mathcal{S}'$-stationary by Definition 4.19. □

The opposite direction is more complicated, especially what comes to the notion of $\mathcal{S}$-stability. Firstly, we are forced to apply the construction introduced in Definition 5.22 to extend a set of non-theorems $N' \subseteq \mathcal{L}^-(\mathcal{S}')$ to a set of non-theorems $N = \mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N') \subseteq \mathcal{L}^-(\mathcal{S})$ using an enumeration ε of the difference $\mathcal{L}^-(\mathcal{S}) - \mathcal{L}^-(\mathcal{S}')$. It is clear by Definition 5.22 that the resulting set of non-theorems N satisfies $N' = N \cap \mathcal{L}^-(\mathcal{S}')$. We begin our analysis with our notion of $\mathcal{S}$-completeness, since our enumeration-based construction is effectively a "completion procedure" for the system $\mathcal{S}$.

Lemma 6.4 *Let $\mathcal{S}$ be any non-monotonic system. If $\mathcal{S}' \sqsubseteq \mathcal{S}$ and $N' \subseteq \mathcal{L}^-(\mathcal{S}')$ is $\mathcal{S}'$-complete, then $N = \mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N') \subseteq \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}$-complete for any enumeration ε of the difference $\mathcal{L}^-(\mathcal{S}) - \mathcal{L}^-(\mathcal{S}')$.*

Proof. Let us assume that N is not $\mathcal{S}$-complete. Then there is $\phi \in \mathcal{L}^-(\mathcal{S})$ such that $\phi \notin N$ and $\phi \notin \mathcal{S}(N)$. Recall that that $\mathcal{L}^-(\mathcal{S}') \subseteq \mathcal{L}^-(\mathcal{S})$ giving rise to two cases depending whether $\phi \in \mathcal{L}^-(\mathcal{S}')$ or not. (i) If $\phi \in \mathcal{L}^-(\mathcal{S}')$, then also $\phi \in \mathcal{L}^+(\mathcal{S}')$. Since $\mathcal{S}' \sqsubseteq \mathcal{S}$ and $N' = N \cap \mathcal{L}^-(\mathcal{S}')$, it holds that $\mathcal{S}'(N') = \mathcal{S}(N) \cap \mathcal{L}^+(\mathcal{S}')$. It follows that $\phi \notin \mathcal{S}'(N')$. Since $\phi \notin N$ implies $\phi \notin N'$, we have a contradiction with $\mathcal{S}'$-completeness of N'. (ii) If $\phi \notin \mathcal{L}^-(\mathcal{S}')$, then ϕ appears in ε, say as the element ε_α for a successor ordinal α. Then $\phi \notin N$ implies $\phi \notin \mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N')$. This is the case only if $\phi \in \mathcal{S}(\mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}}(N'))$ by Definition 5.22. It follows that $\phi \in \mathcal{S}(N)$ by the requirement MON, because $\mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}}(N') \subseteq N$. But this is a contradiction. By the cases (i) and (ii) we have a contradiction. Hence, the set of non-theorems N is $\mathcal{S}$-complete. □

Our second concern is on what conditions our completion procedure generates an $\mathcal{S}$-consistent set of non-theorems $N = \mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N')$ from a $\mathcal{S}'$-consistent set of non-theorems N'. Unfortunately, we cannot use an arbitrary enumeration of ε of $\mathcal{L}^-(\mathcal{S}) - \mathcal{L}^-(\mathcal{S}')$ in general. We have singled out in Definition 6.5 enumerations with a sufficient property. This notion is a slight variant of the local consistency property introduced in Section 5.5. The difference is that a specific ordering of sentences is involved in the consistency condition.

Definition 6.5 *Let $\mathcal{S}$ be any system and $\mathcal{S}' \sqsubseteq \mathcal{S}$ its subsystem. An enumeration ε of $\mathcal{L}^-(\mathcal{S}) - \mathcal{L}^-(\mathcal{S}')$ preserves $\mathcal{S}$-consistency if it holds for all successor ordinals α that if $\mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}}(N')$ is $\mathcal{S}$-consistent and $\varepsilon_\alpha \in \mathrm{Nt}^{\varepsilon,\alpha-1}_{\mathcal{S}}(N')$, then $\mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}}(N')$ is $\mathcal{S}$-consistent.*

Lemma 6.6 *Let $\mathcal{S}$ be any system. If $\mathcal{S}' \sqsubseteq \mathcal{S}$, $N' \subseteq \mathcal{L}^-(\mathcal{S}')$ is $\mathcal{S}'$-consistent and ε is an $\mathcal{S}$-consistency preserving enumeration of $\mathcal{L}^-(\mathcal{S}) - \mathcal{L}^-(\mathcal{S}')$, then $N = \mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N') \subseteq \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}$-consistent.*

Proof sketch. It can be proved by transfinite induction for all $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$ that $\mathrm{Nt}^{\varepsilon,\alpha}_{\mathcal{S}}(N')$ is $\mathcal{S}$-consistent in analogy to the proof of Lemma 5.23. Consequently, the final set of non-theorems $\mathrm{Nt}^{\varepsilon}_{\mathcal{S}}(N')$ is also $\mathcal{S}$-consistent. □

Let us then present consistency preserving enumerations for the non-monotonic systems and their subsystems introduced in Section 3.1. The enumerations for minimal model systems, default systems and autoepistemic systems are addressed in Propositions 6.7, 6.8 and 6.11, respectively. It turns out that that any enumeration goes in conjunction with minimal model systems and default systems while an additional **D**-hierarchicness condition on enumerations is needed if autoepistemic systems are considered.

Proposition 6.7 *Let $P \subseteq \mathcal{A}$ be a set of atoms to be minimized and Γ a set of sentences such that $P \subseteq \Gamma \subseteq \mathcal{L}$. Let $\mathcal{S}$ and $\mathcal{S}'$ be the non-monotonic systems $\mathcal{C}_P$ and $\mathcal{C}^{\Gamma}_P$, respectively. Then every enumeration ε of $\mathcal{L}^-(\mathcal{S}[T]) - \mathcal{L}^-(\mathcal{S}'[T])$ where $T \subseteq \mathcal{L}^+(\mathcal{S}')$ preserves $\mathcal{S}[T]$-consistency.*

Proof. Recall that the language $\mathcal{L}(\mathbf{S})$ is $\langle\mathcal{L},\mathcal{L}\rangle$ and the language $\mathcal{L}(\mathbf{S}')$ is $\langle\mathcal{L},\Gamma\rangle$ so that $T\subseteq\mathcal{L}$. Thus $\mathbf{S}'[T]\sqsubseteq\mathbf{S}[T]$ by Proposition 3.9 and Lemma 3.7. Now the enumeration ε contains only sentences from $\mathcal{L}-\Gamma$. For simplicity of notation, let $N_\alpha=\mathrm{Nt}^{\varepsilon,\alpha}_{\mathbf{S}[T]}(N')$ for all ordinals α. Let us then consider any ε_α for which $\varepsilon_\alpha\in N_\alpha$ and $N_{\alpha-1}$ is $\mathbf{S}[T]$-consistent. Then assume that N_α is not $\mathbf{S}[T]$-consistent. So there is $\phi\in N_\alpha$ such that $\phi\in\mathbf{S}[T](N_\alpha)$. Now $N_\alpha=N_{\alpha-1}\cup\{\varepsilon_\alpha\}$. Since ε_α is not included in Γ nor in $P\subseteq\Gamma$, it follows that $\mathrm{Neg}_P(N_\alpha)=\mathrm{Neg}_P(N_{\alpha-1})$ so that $\mathbf{S}[T](N_\alpha)=\mathrm{Cn}(T\cup\mathrm{Neg}_P(N_\alpha))=\mathrm{Cn}(T\cup\mathrm{Neg}_P(N_{\alpha-1}))=\mathbf{S}[T](N_{\alpha-1})$. Thus also $\phi\in\mathbf{S}[T](N_{\alpha-1})$. Two cases arise. (i) If $\phi\neq\varepsilon_\alpha$, it follows that $\phi\in N_{\alpha-1}$. But this contradicts $\mathbf{S}[T]$-consistency of $N_{\alpha-1}$. (ii) If $\phi=\varepsilon_\alpha$, it follows that $\varepsilon_\alpha\in\mathbf{S}[T](N_{\alpha-1})$, i.e. $\varepsilon_\alpha\notin N_\alpha$, a contradiction. Hence, $N_\alpha=\mathrm{Nt}^{\varepsilon,\alpha}_{\mathbf{S}[T]}(N')$ is $\mathbf{S}[T]$-consistent. □

Proposition 6.8 *Let D be a set of defaults D and Γ a set of sentences such that $\neg\mathrm{J}(D)\subseteq\Gamma\subseteq\mathcal{L}$. Let $\mathbf{S}$ and $\mathbf{S}'$ be non-monotonic systems $\mathfrak{D}_D$ and $\mathfrak{D}^\Gamma_D$, respectively. Then every enumeration ε of $\mathcal{L}^-(\mathbf{S}[T])-\mathcal{L}^-(\mathbf{S}'[T])$ where $T\subseteq\mathcal{L}^+(\mathbf{S}')$ preserves $\mathbf{S}[T]$-consistency.*

Proof sketch. The proof is similar to that of Proposition 6.7, e.g. the definitions of the languages $\mathcal{L}(\mathbf{S})$ and $\mathcal{L}(\mathbf{S}')$ are the same so that $T\subseteq\mathcal{L}$. Let us just explain the major differences here. Now $\mathbf{S}'[T]\sqsubseteq\mathbf{S}[T]$ by Proposition 3.11 and Lemma 3.7. An obvious modification in the proof is that the reductions of D with respect to N_α and $N_{\alpha-1}$ are the same, since ε_α is not included in $\neg\mathrm{J}(D)$. It follows that $\mathbf{S}(N_\alpha)=\mathrm{Cn}^{D_{N_\alpha}}(T)=\mathrm{Cn}^{D_{N_{\alpha-1}}}(T)=\mathbf{S}(N_{\alpha-1})$. □

In contrast to minimal model theories and default theories, arbitrary enumerations do not preserve $\mathbf{S}$-consistency for non-monotonic systems associated with autoepistemic theories. However, there is a simple syntactic restriction on enumerations that does the trick. A similar condition was introduced by Niemelä [76] for conventional autoepistemic theories. In Definition 6.9 we give the corresponding notion for autoepistemic logic with a separate modality for disbeliefs. The idea behind a **D**-*hierarchic* enumeration is that all the disbelief atoms $\mathrm{At}^*_{\mathbf{D}}(\varepsilon_\alpha)$ that appear in a sentence ε_α are processed before the sentence ε_α is processed by the completion procedure. To realize this we have to place the sentences of $\mathrm{SS}^*_{\mathbf{D}}(\varepsilon_\alpha)$ before ε_α in the enumeration ε.

Definition 6.9 *An enumeration ε of a subset N of $\mathcal{L}_{\mathrm{BD}}$ is* **D**-*hierarchic if for every ε_α the sentences of $\mathrm{SS}^*_{\mathbf{D}}(\varepsilon_\alpha)\cap N$ precede ε_α in ε.*

Proposition 6.10 *There is a* **D**-*hierarchic enumeration ε of any $N\subseteq\mathcal{L}_{\mathrm{BD}}$.*

Proof. The language $\mathcal{L}_{\mathrm{BD}}$ is denumerable (given that $\mathcal{A}$ is), i.e. there is an enumeration ε of $\mathcal{L}_{\mathrm{BD}}$ such that $\mathrm{Ind}(\varepsilon)=\omega$. Thus we can define enumerations ε^i as enumeration ε restricted to sentences that appear in N and satisfy

$|\mathrm{SS}_{\mathbf{D}}(\phi)| = i$. This condition guarantees that the enumeration ε' formed as the concatenation $\varepsilon^0 \cdot \varepsilon^1 \cdot \ldots \cdot \varepsilon^i \cdot \ldots$ is **D**-hierarchic. This is because $\psi \in \mathrm{SS}_{\mathbf{D}}(\phi)$ and $\psi \notin \mathrm{SS}_{\mathbf{D}}(\psi)$ imply $|\mathrm{SS}_{\mathbf{D}}(\psi)| < |\mathrm{SS}_{\mathbf{D}}(\phi)|$ so that ψ is before ϕ in ε'. □

Proposition 6.11 *Let $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ and $\Gamma \subseteq \mathcal{L}_{\mathrm{BD}}$ be sets of autoepistemic sentences such that $\mathrm{At}^*_{\mathbf{BD}}(\Sigma) \subseteq \mathrm{At}^*_{\mathbf{BD}}(\Gamma)$. Let $\mathcal{S}$ and $\mathcal{S}'$ be the non-monotonic systems $\mathcal{A}_\Sigma$ and $\mathcal{A}^\Gamma_\Sigma$, respectively. Then every **D**-hierarchic enumeration ε of $\mathcal{L}^-(\mathcal{S}[T]) - \mathcal{L}^-(\mathcal{S}'[T])$ where $T \subseteq \mathcal{L}^+(\mathcal{S}')$ preserves $\mathcal{S}[T]$-consistency.*

Proof. Recall that languages $\mathcal{L}(\mathcal{S})$ and $\mathcal{L}(\mathcal{S}')$ of the systems $\mathcal{S}$ and $\mathcal{S}'$ are $\langle \mathcal{L}_{\mathrm{BD}}, \mathcal{L}_{\mathrm{BD}} \rangle$ and $\langle \mathrm{SS}^*_{\mathbf{BD}}(\Gamma), \mathrm{SS}^*_{\mathbf{D}}(\Gamma) \rangle$, respectively, so that $T \subseteq \mathrm{SS}^*_{\mathbf{BD}}(\Gamma)$. Then $\mathcal{S}'[T] \sqsubseteq \mathcal{S}[T]$ follows by Proposition 3.17 and Lemma 3.7. Now ε is a **D**-hierarchic enumeration of $\mathcal{L}_{\mathrm{BD}} - \mathrm{SS}^*_{\mathbf{D}}(\Gamma)$. Like in the proof of Proposition 6.7 we are assuming that $\varepsilon_\alpha \in N_\alpha$ and $N_{\alpha-1}$ is $\mathcal{S}[T]$-consistent. A further assumption that N_α is not $\mathcal{S}[T]$-consistent implies that there is a sentence $\phi \in N_\alpha$ such that $\phi \in \mathcal{S}[T](N_\alpha)$, i.e. $\Sigma \cup T \cup \mathbf{D}N_\alpha \vdash_{\mathbf{B}} \phi$.

We need to establish that $\mathbf{D}N_\alpha \cap \mathrm{At}^*_{\mathbf{D}}(\Sigma \cup T \cup \{\phi\}) = \mathbf{D}N_{\alpha-1} \cap \mathrm{At}^*_{\mathbf{D}}(\Sigma \cup T \cup \{\phi\})$. This is the case if $\mathbf{D}\varepsilon_\alpha \notin \mathrm{At}^*_{\mathbf{D}}(\Sigma \cup T \cup \{\phi\})$. So let us assume for the sake of contradiction that $\mathbf{D}\varepsilon_\alpha \in \mathrm{At}^*_{\mathbf{D}}(\Sigma \cup T \cup \{\phi\})$, i.e. $\varepsilon_\alpha \in \mathrm{SS}^*_{\mathbf{D}}(\Sigma \cup T \cup \{\phi\})$. Now it is impossible that $\varepsilon_\alpha \in \mathrm{SS}^*_{\mathbf{D}}(\Sigma \cup T)$, since ε is an enumeration of $\mathcal{L}_{\mathrm{BD}} - \mathrm{SS}^*_{\mathbf{D}}(\Gamma)$, and $\mathrm{At}^*_{\mathbf{BD}}(\Sigma) \subseteq \mathrm{At}^*_{\mathbf{BD}}(\Gamma)$ and $T \subseteq \mathrm{SS}^*_{\mathbf{BD}}(\Gamma)$ imply that $\mathrm{SS}^*_{\mathbf{D}}(\Sigma \cup T) \subseteq \mathrm{SS}^*_{\mathbf{D}}(\Gamma)$. Thus $\varepsilon_\alpha \in \mathrm{SS}^*_{\mathbf{D}}(\phi)$ must be the case and then ϕ cannot appear in $\{\varepsilon_{0\ldots\alpha}\}$, because ε is **D**-hierarchic. Thus $\phi \in N_\alpha$ implies $\phi \in N'$. It follows that $\phi \in \mathrm{SS}^*_{\mathbf{D}}(\Gamma)$, because $N' \subseteq \mathrm{SS}^*_{\mathbf{D}}(\Gamma)$. Thus also $\varepsilon_\alpha \in \mathrm{SS}^*_{\mathbf{D}}(\Gamma)$, a contradiction.

Now $\Sigma \cup T \cup \mathbf{D}N_\alpha \vdash_{\mathbf{B}} \phi$ implies $\Sigma \cup T \cup (\mathbf{D}N_\alpha \cap \mathrm{At}^*_{\mathbf{D}}(\Sigma \cup T \cup \{\phi\})) \vdash_{\mathbf{B}} \phi$ by Lemma 3.16. Then $\Sigma \cup T \cup (\mathbf{D}N_{\alpha-1} \cap \mathrm{At}^*_{\mathbf{D}}(\Sigma \cup T \cup \{\phi\})) \vdash_{\mathbf{B}} \phi$ follows, since $\mathbf{D}N_\alpha \cap \mathrm{At}^*_{\mathbf{D}}(\Sigma \cup T \cup \{\phi\}) = \mathbf{D}N_{\alpha-1} \cap \mathrm{At}^*_{\mathbf{D}}(\Sigma \cup T \cup \{\phi\})$ holds by the above. This implies that $\Sigma \cup T \cup \mathbf{D}N_{\alpha-1} \vdash_{\mathbf{B}} \phi$ by the monotonicity of **B**-provability (Proposition 2.29). Thus $\phi \in \mathcal{S}[T](N_{\alpha-1})$. From this we obtain a contradiction in the same way as we did in the proof of Proposition 6.7. Hence N_α is $\mathcal{S}[T]$-consistent and the **D**-hierarchic enumeration ε preserves $\mathcal{S}[T]$-consistency. □

Proposition 6.12 provides a converse result compared to that of Proposition 6.3: it shows how sets of non-theorems $N' \subseteq \mathcal{L}^-(\mathcal{S}')$ are extended to sets of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ such that the properties of our interest are preserved. The stationary case is relatively easy once we have proved Lemma 6.2 and we can apply $\overline{\mathcal{S}}^2$ iteratively to (re)construct an $\mathcal{S}$-stationary set of non-theorems N such that $N' = N \cap \mathcal{L}^-(\mathcal{S})$.

Proposition 6.12 *Let $\mathcal{S}' \sqsubseteq \mathcal{S}$ and $N' \subseteq \mathcal{L}^-(\mathcal{S}')$. If there is an $\mathcal{S}$-consistency preserving enumeration of $\mathcal{L}^-(\mathcal{S}) - \mathcal{L}^-(\mathcal{S}')$ and N' is (i) $\mathcal{S}'$-consistent, (ii) $\mathcal{S}'$-complete, or alternatively (iii) $\mathcal{S}'$-stable, then there is $N \subseteq \mathcal{L}^-(\mathcal{S})$ having the corresponding property with respect to the system $\mathcal{S}$ such that $N' = N \cap \mathcal{L}^-(\mathcal{S}')$.*

Moreover, if N' is (iv) $\mathcal{S}'$-stationary, then there is an $\mathcal{S}$-stationary set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ such that $N' = N \cap \mathcal{L}^-(\mathcal{S}')$.

Proof. (i-iii) These follow by Lemmas 6.4 and 6.6, given there is an $\mathcal{S}$-consistency preserving enumeration of $\mathcal{L}^-(\mathcal{S}) - \mathcal{L}^-(\mathcal{S}')$. (iv) Let N' be $\mathcal{S}$-stationary, i.e. $N' = \overline{\mathcal{S}'}^2(N')$ and $N' \subseteq \overline{\mathcal{S}'}(N')$ hold by Definition 4.19. It follows by Lemmas 6.1 and 6.2 (with $N = N'$) that $N' \subseteq \overline{\mathcal{S}}^2(N')$ and $N' \subseteq \overline{\mathcal{S}}(N')$. Note that $\mathcal{L}^-(\mathcal{S}') \subseteq \mathcal{L}^-(\mathcal{S})$. Then define $N_\alpha = \overline{\mathcal{S}}^2 \uparrow^\alpha (N')$ for all ordinals. By Proposition 4.27, the sequence $N_0, N_1, \ldots, N_\alpha, \ldots$ of sets of non-theorems is non-decreasing and N_α is $\mathcal{S}$-consistent for all ordinals α.

Let us then prove by induction on α that $N_\alpha \cap \mathcal{L}^-(\mathcal{S}') = N'$. The base case $\alpha = 0$ holds, since $N_0 = N'$. Induction step is as follows. Consider a successor ordinal α. It holds that $N_{\alpha-1} \cap \mathcal{L}^-(\mathcal{S}') = N'$ by the inductive hypothesis. It follows by Lemma 6.2 that $N_\alpha \cap \mathcal{L}^-(\mathcal{S}') = \overline{\mathcal{S}}^2(N_{\alpha-1}) \cap \mathcal{L}^-(\mathcal{S}') = \overline{\mathcal{S}'}^2(N') = N'$. Finally, consider a limit ordinal α. By the inductive hypothesis, it holds that $N_\beta \cap \mathcal{L}^-(\mathcal{S}') = N'$ for all ordinals $\beta < \alpha$. Thus $N_\alpha \cap \mathcal{L}^-(\mathcal{S}') = (\bigcup_{\beta<\alpha} N_\beta) \cap \mathcal{L}^-(\mathcal{S}') = \bigcup_{\beta<\alpha}(N_\beta \cap \mathcal{L}^-(\mathcal{S}')) = \bigcup_{\beta<\alpha} N' = N'$. We have therefore established that the fixed point $N = \mathrm{lfp}(\overline{\mathcal{S}}^2, N')$ is an $\mathcal{S}$-stationary set of non-theorems satisfying $N \cap \mathcal{L}^-(\mathcal{S}) = N'$. □

6.2 Functional Equivalences

The notion of functional equivalence is considered next. In analogy to Lemmas 6.1 and 6.2 in the preceding section, we show in Lemma 6.13 that the functional equivalence relation $\sim$ preserves the operators $\overline{\mathcal{S}}$ and $\overline{\mathcal{S}}^2$ for the system $\mathcal{S}$.

Lemma 6.13 *If $\mathcal{S} \sim \mathcal{S}'$ via $f = \langle f^+, f^- \rangle$, then it holds for all $N \subseteq \mathcal{L}^-(\mathcal{S})$ that (i) $f^-(\overline{\mathcal{S}}(N)) = \overline{\mathcal{S}'}(f^-(N))$ and (ii) $f^-(\overline{\mathcal{S}}^2(N)) = \overline{\mathcal{S}'}^2(f^-(N))$.*

Proof. (i) At first we consider $\phi \in \mathcal{L}^-(\mathcal{S}')$ to establish an iff-relationship. Define $\phi' = (f^-)^{-1}(\phi)$ so that $\phi = f^-(\phi')$. Define also $\phi'' = f^+(\phi')$. Since f^- and f^+ must map ϕ' into $\mathcal{S}'$-equivalent sentences (ϕ and ϕ''), it follows that $\phi \in \mathcal{S}'(f^-(N))$ if and only if $\phi'' \in \mathcal{S}'(f^-(N))$. This holds if and only if $\phi' \in (f^+)^{-1}(\mathcal{S}'(f^-(N)))$, or equivalently $\phi \in f^-((f^+)^{-1}(\mathcal{S}'(f^-(N))))$. Thus

$$
\begin{array}{rcl|l}
f^-(\overline{\mathcal{S}}(N)) & = & f^-(\mathcal{L}^-(\mathcal{S}) - \mathcal{S}(N)) & \text{Definition of } \overline{\mathcal{S}} \\
 & = & f^-(\mathcal{L}^-(\mathcal{S}) - (f^+)^{-1}(\mathcal{S}'(f^-(N)))) & \mathcal{S} \sim \mathcal{S}' \text{ via } f \\
 & = & \mathcal{L}^-(\mathcal{S}') - f^-((f^+)^{-1}(\mathcal{S}'(f^-(N)))) & \text{Apply } f^- \\
 & = & \mathcal{L}^-(\mathcal{S}') - \mathcal{S}'(f^-(N)) & \text{See above} \\
 & = & \overline{\mathcal{S}'}(f^-(N)). & \text{Definition of } \overline{\mathcal{S}'}
\end{array}
$$

(ii) It holds for every set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ that $f^-(\overline{\mathcal{S}}^2(N)) = f^-(\overline{\mathcal{S}}(\overline{\mathcal{S}}(N))) = \overline{\mathcal{S}'}(f^-(\overline{\mathcal{S}}(N))) = \overline{\mathcal{S}'}(\overline{\mathcal{S}'}(f^-(N))) = \overline{\mathcal{S}'}^2(f^-(N))$ by (i). □

Proposition 6.14 *Let $\mathcal{S} \sim \mathcal{S}'$ via $f = \langle f^+, f^- \rangle$ and $N \subseteq \mathcal{L}^-(\mathcal{S})$. If N is (i) $\mathcal{S}$-consistent, (ii) $\mathcal{S}$-complete, (iii) $\mathcal{S}$-stable or alternatively (iv) $\mathcal{S}$-stationary, then $N' = f^-(N)$ has the corresponding property with respect to $\mathcal{S}'$.*

Proof. By Lemma 6.13 we know that (i) $N \subseteq \overline{\mathcal{S}}(N)$ implies $f^-(N) \subseteq \overline{\mathcal{S}'}(f^-(N))$, (ii) $\overline{\mathcal{S}}(N) \subseteq N$ implies $\overline{\mathcal{S}'}(f^-(N)) \subseteq f^-(N)$ so that (iii) $N = \overline{\mathcal{S}}(N)$ implies $f^-(N) = \overline{\mathcal{S}'}(f^-(N))$. Also (iv) $N = \overline{\mathcal{S}}^2(N)$ and $N \subseteq \overline{\mathcal{S}}(N)$ imply $f^-(N) = \overline{\mathcal{S}'}^2(f^-(N))$ and $f^-(N) \subseteq \overline{\mathcal{S}'}(f^-(N))$. The rest follows by Propositions 4.9 and 4.10 and Definitions 4.8 and 4.19. □

Let us yet note that Lemma 6.13 and Proposition 6.14 above are naturally symmetric with respect to the involved non-monotonic systems $\mathcal{S}$ and $\mathcal{S}'$, since the relation $\sim$ is an equivalence relation. Thus we have established that functional equivalences preserve our notions of $\mathcal{S}$-consistency and $\mathcal{S}$-completeness for sets of non-theorems as well as $\mathcal{S}$-stable and $\mathcal{S}$-stationary sets of non-theorems.

6.3 Embeddings

The notion of a embedding introduced in Section 3.3 is modularly based on the notions of subsystem and functional equivalence: $\mathcal{S}' \mapsto_f \mathcal{S}$ if and only if $\mathcal{S}'$ is functionally equivalent to a subsystem of $\mathcal{S}$ via the l-morphism f. We may now utilize the modularity of the definition of an embedding and establish that embeddings preserve various properties of sets of non-theorems because subsystems and functional equivalences do. This is how Corollaries 6.15 and 6.16 arise. In particular, the restrictions of Proposition 6.12 (existence of an $\mathcal{S}$-consistency preserving enumeration) carry over to Corollary 6.16.

Corollary 6.15 *Let $\mathcal{S}' \mapsto_f \mathcal{S}$ for an l-morphism $f = \langle f^+, f^- \rangle$ and let $N \subseteq \mathcal{L}^-(\mathcal{S})$. If N is (i) $\mathcal{S}$-consistent (ii) $\mathcal{S}$-complete (iii) $\mathcal{S}$-stable or alternatively (iv) $\mathcal{S}$-stationary, then the set of non-theorems $N' = (f^-)^{-1}(N \cap f^-(\mathcal{L}^-(\mathcal{S}')))$ has the corresponding property with respect to the system $\mathcal{S}'$.*

Proof. This is a corollary of Propositions 6.3 and 6.14. □

Corollary 6.16 *Let $\mathcal{S}' \mapsto_f \mathcal{S}$ for $f = \langle f^+, f^- \rangle$ and let $N' \subseteq \mathcal{L}^-(\mathcal{S}')$.*

If there is an $\mathcal{S}$-consistency preserving enumeration of $\mathcal{L}^-(\mathcal{S}) - f^-(\mathcal{L}^-(\mathcal{S}'))$ and N' is (i) $\mathcal{S}'$-consistent (ii) $\mathcal{S}'$-complete, or alternatively (iii) $\mathcal{S}'$-stable, then there is $N \subseteq \mathcal{L}^-(\mathcal{S})$ having the corresponding property with respect to the system $\mathcal{S}$ such that $N' = (f^-)^{-1}(N \cap f^-(\mathcal{L}^-(\mathcal{S}')))$.

Moreover, if N' is (iv) $\mathcal{S}'$-stationary, then there is an $\mathcal{S}$-stationary set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ such that $N' = (f^-)^{-1}(N \cap f^-(\mathcal{L}^-(\mathcal{S}')))$.

Proof. This is a corollary of Propositions 6.12 and 6.14. $\square$

6.4 Stable and Stationary Semantics

We have already established that the relations $\sqsubseteq$, $\sim$ and $\mapsto$ preserve $\mathcal{S}$-stable and $\mathcal{S}$-stationary sets of non-theorems. In this section, we handle semantic operators $\mathrm{STAB}_{\mathcal{S}}$ and $\mathrm{STAT}_{\mathcal{S}}$ and show that these operators are also preserved by these relations. We begin with subsystems and use Propositions 6.3 and 6.12 to establish a correspondence for the stable semantics of two non-monotonic systems $\mathcal{S}$ and $\mathcal{S}'$ such that $\mathcal{S} \sqsubseteq \mathcal{S}'$. Recall that we showed in Propositions 6.7, 6.8 and 6.11 that there are consistency preserving enumerations for minimal model systems $\mathfrak{C}_P$, default systems $\mathfrak{D}_D$ and autoepistemic systems $\mathcal{A}_\Sigma$ introduced in Section 2.3 and their subsystems distinguished in Section 3.1. This is how Theorem 6.17 covers these systems.

Theorem 6.17 *Let $\mathcal{S} \sqsubseteq \mathcal{S}'$ such that there is an $\mathcal{S}'[T]$-consistency preserving enumeration of $\mathcal{L}^-(\mathcal{S}'[T]) - \mathcal{L}^-(\mathcal{S}[T])$ for all $T \subseteq \mathcal{L}^+(\mathcal{S})$. Then for all $T \subseteq \mathcal{L}^+(\mathcal{S})$, the stable semantics $\mathrm{STAB}_{\mathcal{S}}(T) = \mathrm{STAB}_{\mathcal{S}'}(T) \cap \mathcal{L}^+(\mathcal{S})$.*

Proof. Consider a sentence $\phi \in \mathcal{L}^+(\mathcal{S})$. It is then shown that $\phi \in \mathrm{STAB}_{\mathcal{S}}(T)$ if and only if $\phi \in \mathrm{STAB}_{\mathcal{S}'}(T)$. ($\Rightarrow$) So let $\phi \notin \mathrm{STAB}_{\mathcal{S}'}(T)$ hold, i.e. there is an $\mathcal{S}'[T]$-stable set of non-theorems N' such that $\phi \notin \mathcal{S}[T](N') = \mathcal{S}(T, N')$. Since $\mathcal{S}[T] \sqsubseteq \mathcal{S}'[T]$ by Lemma 3.7, it follows by Proposition 6.3 that $N = N' \cap \mathcal{L}^-(\mathcal{S}[T]) = N' \cap \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}[T]$-stable. Since $\mathcal{S} \sqsubseteq \mathcal{S}'$, the set of theorems $\mathcal{S}(T, N) = \mathcal{S}'(T, N') \cap \mathcal{L}^+(\mathcal{S})$. It follows that $\phi \notin \mathcal{S}(T, N) = \mathcal{S}[T](N)$, because $\phi \in \mathcal{L}^+(\mathcal{S})$. Since N is $\mathcal{S}[T]$-stable, $\phi \notin \mathrm{STAB}_{\mathcal{S}}(T)$. ($\Leftarrow$) Let $\phi \notin \mathrm{STAB}_{\mathcal{S}}(T)$ hold. Then there is a $\mathcal{S}[T]$-stable set of non-theorems N such that $\phi \notin \mathcal{S}[T](N) = \mathcal{S}(T, N)$. Since $\mathcal{S}[T] \sqsubseteq \mathcal{S}'[T]$ by Lemma 3.7, there is $\mathcal{S}'[T]$-stable set of non-theorems N' such that $N = N' \cap \mathcal{L}^-(\mathcal{S}[T]) = N' \cap \mathcal{L}^-(\mathcal{S})$ by Proposition 6.12. Since $\mathcal{S} \sqsubseteq \mathcal{S}'$, $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $N' \subseteq \mathcal{L}^-(\mathcal{S}')$ such that $N = N' \cap \mathcal{L}^-(\mathcal{S})$, it follows that $\mathcal{S}(T, N) = \mathcal{S}'(T, N') \cap \mathcal{L}^+(\mathcal{S})$. Thus $\phi \notin \mathcal{S}'(T, N') = \mathcal{S}'[T](N')$ and $\phi \notin \mathrm{STAB}_{\mathcal{S}'}(T)$. $\square$

The same result extends for stationary semantics, but the existence of consistency preserving enumerations need not to be assumed.

Theorem 6.18 *For $\mathcal{S} \sqsubseteq \mathcal{S}'$ and for all $T \subseteq \mathcal{L}^+(\mathcal{S})$, the stationary semantics $\mathrm{STAT}_{\mathcal{S}}(T) = \mathrm{STAT}_{\mathcal{S}'}(T) \cap \mathcal{L}^+(\mathcal{S})$.*

Proof sketch. The proof is obtained from the proof of Theorem 6.17 by replacing stable sets of non-theorems with stationary sets of non-theorems. $\square$

Theorems 6.17 and 6.18 will be important in our forthcoming complexity analysis in Section 7. Our complexity results will cover non-monotonic systems $\mathcal{S}$ with an infinite language $\mathcal{L}^-(\mathcal{S})$ given that there are suitable subsystems $\mathcal{S}'$ of $\mathcal{S}$ such that $\mathcal{L}^-(\mathcal{S}')$ is finite. This is the case, for instance, with the non-monotonic system $\boldsymbol{\mathcal{A}}_\Sigma$ associated with a finite autoepistemic theory $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$. Our next objective is generalizing the results of Theorems 6.17 and 6.18 for the embedding relation $\mapsto$. As a preparation for this, functional equivalences are taken into consideration by the following two theorems.

Theorem 6.19 *For $\mathcal{S} \sim \mathcal{S}'$ via $f = \langle f^+, f^- \rangle$ and for all $T \subseteq \mathcal{L}^+(\mathcal{S})$, the stable semantics $f^+(\mathrm{STAB}_{\mathcal{S}}(T)) = \mathrm{STAB}_{\mathcal{S}'}(f^+(T))$.*

Proof. ($\subseteq$) Let $\phi' \notin \mathrm{STAB}_{\mathcal{S}'}(f^+(T))$. Then there is an $\mathcal{S}'[f^+(T)]$-stable set of non-theorems N' such that $\phi' \notin \mathcal{S}'[f^+(T)](N') = \mathcal{S}'(f^+(T), N')$. Define $N = (f^-)^{-1}(N')$ so that $N' = f^-(N)$. Since $\mathcal{S}[T] \sim \mathcal{S}'[f^+(T)]$ via f holds by Lemma 3.23, it follows that N is $\mathcal{S}[T]$-stable by Proposition 6.14 applied to the inverse $(f)^{-1} = \langle (f^+)^{-1}, (f^-)^{-1} \rangle$. Moreover, it holds that $f^+(\mathcal{S}[T](N)) = f^+(\mathcal{S}(T, N)) = \mathcal{S}'(f^+(T), f^-(N)) = \mathcal{S}'[f^+(T)](N')$. This implies that $\phi' \notin f^+(\mathcal{S}[T](N))$ and $\phi' \notin f^+(\mathrm{STAB}_{\mathcal{S}}(T))$. ($\supseteq$) Since $\sim$ is symmetric, it holds that $\mathcal{S}' \sim \mathcal{S}$ via $(f)^{-1}$ and by ($\subseteq$) that $(f^+)^{-1}(\mathrm{STAB}_{\mathcal{S}'}(T')) \subseteq \mathrm{STAB}_{\mathcal{S}}((f^+)^{-1}(T'))$ where $T' = f^+(T)$. Therefore, the set of theorems $f^+((f^+)^{-1}(\mathrm{STAB}_{\mathcal{S}'}(T'))) = \mathrm{STAB}_{\mathcal{S}'}(f^+(T))$ is a subset of $f^+(\mathrm{STAB}_{\mathcal{S}}((f^+)^{-1}(T'))) = f^+(\mathrm{STAB}_{\mathcal{S}}(T))$. $\square$

Theorem 6.20 *For $\mathcal{S} \sim \mathcal{S}'$ via $f = \langle f^+, f^- \rangle$ and for all $T \subseteq \mathcal{L}^+(\mathcal{S})$, the stationary semantics $f^+(\mathrm{STAT}_{\mathcal{S}}(T)) = \mathrm{STAT}_{\mathcal{S}'}(f^+(T))$.*

Proof sketch. The proof is obtained from the proof of Theorem 6.19 by replacing stable sets of non-theorems with stationary sets of non-theorems. $\square$

Thus we are ready to extend Theorems 6.17 and 6.18 for embeddings.

Corollary 6.21 *Let $\mathcal{S} \mapsto_f \mathcal{S}'$ for an l-morphism $f = \langle f^+, f^- \rangle$.*

(i) If there is an $\mathcal{S}'[T]$-consistency preserving enumeration of $\mathcal{L}^-(\mathcal{S}'[T]) - f^-(\mathcal{L}^-(\mathcal{S}[T]))$ for every $T \subseteq \mathcal{L}^+(\mathcal{S})$, then the stable semantics

$$f^+(\mathrm{STAB}_{\mathcal{S}}(T)) = \mathrm{STAB}_{\mathcal{S}'}(f^+(T)) \cap f^+(\mathcal{L}^+(\mathcal{S}))$$

for all $T \subseteq \mathcal{L}^+(\mathcal{S})$. (ii) For all $T \subseteq \mathcal{L}^+(\mathcal{S})$, the stationary semantics

$$f^+(\mathrm{STAT}_{\mathcal{S}}(T)) = \mathrm{STAT}_{\mathcal{S}'}(f^+(T)) \cap f^+(\mathcal{L}^+(\mathcal{S})).$$

Proof. This is a corollary of Theorems 6.17, 6.18, 6.19 and 6.20. $\square$

The first equation provided by Corollary 6.21 can be rewritten as $\mathrm{STAB}_{\mathcal{S}}(T) = (f^{+})^{-1}(\mathrm{STAB}_{\mathcal{S}'}(f^{+}(T)) \cap f^{+}(\mathcal{L}^{+}(\mathcal{S})))$. From this equation, it is obvious that $\mathrm{STAB}_{\mathcal{S}}(T)$ can be determined using the system $\mathcal{S}'$ in which f embeds $\mathcal{S}$. The same can be concluded from a similar equation for stationary semantics. This indicates that $\mathcal{S}'$ has at least the same functionality as $\mathcal{S}$ under stable semantics as well as stationary semantics. Let us then apply Corollary 6.21 to the natural embeddings between non-monotonic systems obtained by the translation functions presented in Section 3.4. Theorems 3.27, 3.32 and 3.39 give rise to Corollaries 6.22, 6.23 and 6.24, respectively. Recall that these theorems were systematically based on l-morphisms $f = \langle f^{+}, f^{-} \rangle$ where f^{+} is the identity i^{+}. Also the languages of the involved non-monotonic systems coincide in Theorems 3.27 and 3.39. Thus we have a good opportunity to simplify the equations of Corollary 6.21 for Corollaries 6.22 - 6.24 below.

Corollary 6.22 *Let $P \subseteq \mathcal{A}$ be a set of atoms to be minimized. For all $T \subseteq \mathcal{L}$,*

$$\mathrm{STAB}_{\mathcal{C}_P}(T) = \mathrm{STAB}_{\mathcal{D}_{\mathrm{Tr_E}(P)}}(T) \textit{ and } \mathrm{STAT}_{\mathcal{C}_P}(T) = \mathrm{STAT}_{\mathcal{D}_{\mathrm{Tr_E}(P)}}(T).$$

Corollary 6.23 *Let D be a set of defaults. For all $T \subseteq \mathcal{L}$,*

$$\mathrm{STAB}_{\mathcal{D}_D}(T) = \mathrm{STAB}_{\mathcal{A}_{\mathrm{Tr_K}(D)}}(T) \cap \mathcal{L} \textit{ and}$$
$$\mathrm{STAT}_{\mathcal{D}_D}(T) = \mathrm{STAT}_{\mathcal{A}_{\mathrm{Tr_K}(D)}}(T) \cap \mathcal{L}.$$

Corollary 6.24 *Let Σ be an autoepistemic theory. For all $T \subseteq \mathcal{L}_{\mathrm{BD}}$,*

$$\mathrm{STAB}_{\mathcal{A}_\Sigma}(T) = \mathrm{STAB}_{\mathcal{D}_{\mathrm{Tr_J}(\Sigma)}}(T) \textit{ and } \mathrm{STAT}_{\mathcal{A}_\Sigma}(T) = \mathrm{STAT}_{\mathcal{D}_{\mathrm{Tr_J}(\Sigma)}}(T).$$

Upon these corollaries, it is justifiable to conclude that the translation functions of Section 3.4 *preserve* the stable semantics and the stationary semantics of the non-monotonic systems under consideration. The same result can be expected from various semantics that are built upon the operators $\overline{\mathcal{S}}$ and $\overline{\mathcal{S}}^2$. Examples of such semantics will be considered later in Section 8. Moreover, Corollary 6.24 provides new insight to the relationship of default logic and (bimodal) autoepistemic logic: the translation $\mathrm{Tr_J}(\Sigma)$ is *modular* (i.e. $\mathrm{Tr_J}(\Sigma)$ does not depend on $T \subseteq \mathcal{L}_{\mathrm{BD}}$) and *faithful* (i.e. expansions of T and extensions of $\langle \mathrm{Tr_J}(\Sigma), T \rangle$ coincide for all $T \subseteq \mathcal{L}_{\mathrm{BD}}$). Analogous modularity and faithfulness arguments apply to $\mathrm{Tr_K}$ on the basis of Corollary 6.23. The modularity of the translation functions $\mathrm{Tr_J}(\Sigma)$ and $\mathrm{Tr_K}(D)$ can be interpreted a strong indication that default theories and bimodal autoepistemic theories are of equal expressive power (under stable and stationary semantics).

7 Computational Complexity

In this section, we turn our attention to the computational aspects of non-monotonic reasoning. In particular, we are interested in how complex non-monotonic reasoning is compared to classical monotonic reasoning. From the early 90s, many researchers have classified the *decision problems* of the major formalizations of non-monotonic reasoning using the measures introduced in computational complexity theory (see, e.g., [2, 50, 86]). Generally speaking, the work on the complexity issues has turned out to be very fruitful. For instance, Gottlob and Eiter [25] have analyzed the complexity of closed world assumption and circumscription. Stillman [106] and Gottlob [37, 38] have classified the decision problems of default logic. Gottlob [37] and Niemelä [77] have considered the decision problems of autoepistemic logic. To summarize, the decision problems of circumscription, default logic and conventional autoepistemic logic under stable semantics were shown to reside on the second level of *polynomial time hierarchy* [2]. In contrast to this, the main decision problems of propositional logic – on which all the above mentioned non-monotonic logics are based – are on the first level in the polynomial time hierarchy. This indicates that the expressive power of non-monotonic logics is greater than that of propositional logic, and that non-monotonic reasoning tends to be more complex than similar monotonic reasoning in general. On the other hand, the complexity analysis revealed the major sources of computational complexity in non-monotonic reasoning: the underlying monotonic reasoning and the determination of extensions/expansions. This has turned out to be valuable information when efficient implementation methods of non-monotonic reasoning have been gradually developed (see [80, 84, 85], for instance).

In the sequel, we take a more general view to the computational aspects of non-monotonic reasoning in terms of our framework of non-monotonic systems. Our analysis proceeds as follows. In Section 7.1, we review the basic notions of complexity theory that are relevant for our forthcoming discussion. These include decision problems, various kinds of Turing machines as well as the polynomial time hierarchy in which decision problems are classified. Reducibilities are needed in the definition of the polynomial time hierarchy as well as to define the hardest problems in the complexity classes of the hierarchy. In Section 7.2, decision problems associated with minimal model systems $\mathfrak{C}_P$, default systems $\mathfrak{D}_D$ and autoepistemic systems $\boldsymbol{\mathcal{A}}_\Sigma$ are analyzed and classified in the polynomial time hierarchy. The analysis of the decision problems related to stable semantics and stationary semantics takes place in Section 7.3. The algorithms and reducibility results presented lead to a classification of decision problems in the polynomial time hierarchy. In particular, we are interested to know how the time complexity of autoepistemic reasoning is affected by the separate modality $\mathbf{D}$ for disbeliefs.

7.1 Classifying Decision Problems

From an abstract point of view, a *decision problem* involves deciding the membership of an element e in some set A of interest. The answer to a decision problem concerning an element e and a set A is either "yes" or "no" depending whether e is a member of A or not. From now on we identify a decision problem with the set of its yes-instances, i.e. the set A itself. Thus a class of decision problems satisfying certain criteria is effectively a class of sets. For instance, in the context of autoepistemic reasoning where beliefs play a major role, we might like to find out whether a given sentence $\phi \in \mathcal{L}_{\mathrm{BD}}$ belongs to the intersection of the iterative expansions of a given a finite autoepistemic theory $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$. By the above discussion, it would be natural to define the corresponding decision problem as a set of pairs $\langle \Sigma, \phi \rangle$ where $\phi \in \mathcal{L}_{\mathrm{BD}}$ belongs to every iterative expansion of an autoepistemic theory $\Sigma \subseteq_{\mathrm{f}} \mathcal{L}_{\mathrm{BD}}$.

7.1.1 Turing Machines

We use *Turing machines* [2] as a model of computation to *solve* decision problems and to *accept* specific sets of our interest. In particular, a decision problem A is said to be *decidable* if there is a *deterministic* Turing machine that provides a correct answer to the problem in finite *time*, i.e. in a finite number of state transitions performed by the machine. A deterministic Turing machine performs computations such that in each state there is only one successor state determined by the *transition relation* of the machine. In contrast to this, multiple successor states are allowed in a *nondeterministic* Turing machine. Of these, one successor state is nondeterministically chosen by the machine. Consequently, a nondeterministic machine may have many different *computations* – depending on the nondeterministic choices made. A *halting state* is a state where the computation of a Turing machine stops, i.e. there are no successor states given by the transition relation. In particular, special halting states – *accepting* and *rejecting* states – are distinguished when Turing machines are used to accept specific sets. A nondeterministic Turing machine *accepts* a set A if and only if there is at least one accepting computation for each element $a \in A$ given as input. Moreover, the computations of a nondeterministic Turing machine can be simulated using a deterministic Turing machine. The difference between these types of Turing machines is that a deterministic Turing machine may require an exponential time when accepting a particular set while a nondeterministic Turing machine can do the same task in polynomial time. This illustrates the hidden power of nondeterministic choice.

Let us then introduce *oracle machines* [2]. An oracle machine is either deterministic or nondeterministic Turing machine that is capable of carrying out a membership test in an *oracle set* in one time unit (i.e. in one step of computation). Since we are representing problems as sets of yes-instances, this capability means that an oracle machine can solve a specific problem in one time unit. We will use $\mathbf{P}(\mathbf{C})$ to denote the class of sets accepted by deterministic oracle machines using oracles from the class $\mathbf{C}$ in polynomial time. Similarly, the notation $\mathbf{NP}(\mathbf{C})$ is used in the nondeterministic case. Oracle sets typically correspond to some decision problems of interest. In this way, oracle machines provide us a powerful tool to study the remaining time complexity of a problem assuming some subproblem can be solved in one time unit. By this technique it becomes possible to identify several sources of time complexity in a specific decision problem.

Turing machines will be represented in a Pascal-like programming notation in this work. Slight extensions to a standard notation are required to allow concise description of the special features of deterministic and nondeterministic Turing machines. The idea is, of course, that we rely on our intuition how to construct actual Turing machines described by this notation. The nondeterministic choice of an element e from a finite set $\mathbf{S}$ is presented by a statement **Choose** e **From** S. When this statement is executed, the variable e is assigned to an arbitrary element of the set S. The computation of a machine halts in an accepting or rejecting state when the statements **Accept** and **Reject**, respectively, are executed. Notice also that **For**-loops are assumed to execute a statement for all elements in a finite set (the order is not relevant) and for all elements in a finite enumeration (which determines the order). If a function is computed using a Turing machine, then we use the **Return**-statement to declare the value of the function computed.

7.1.2 Polynomial Time Hierarchy

One of our goals is to classify the various decision problems related with non-monotonic reasoning with respect to the *polynomial time hierarchy* (see e.g. [2] for further details). Each level $k \geq 0$ of polynomial time hierarchy consists of problem classes $\mathbf{\Sigma}^{\mathrm{p}}_{k}$, $\mathbf{\Pi}^{\mathrm{p}}_{k}$ and $\mathbf{\Delta}^{\mathrm{p}}_{k}$. The intuition is that the higher a problem lies in the polynomial time hierarchy, i.e. the greater value k has, the greater time complexity the problem has. The lowest level of the polynomial time hierarchy ($k = 0$) consists of the problems that can be solved by a deterministic Turing machine in time polynomial with respect to the *length of the input* (i.e. problem instance). This class of problems is denoted by $\mathbf{P}$ and the classes $\mathbf{\Sigma}^{\mathrm{p}}_{0}$, $\mathbf{\Pi}^{\mathrm{p}}_{0}$ and $\mathbf{\Delta}^{\mathrm{p}}_{0}$ equal to $\mathbf{P}$ by definition. The problems in the class $\mathbf{NP}$ (nondeterministic polynomial) are solvable by a nondeterministic Turing machine in polynomial time, i.e. the yes-instances of the problem can be accepted by a nondeterministic Turing machine in polynomial time.

The rest of the hierarchy is defined as follows. The class $\mathbf{\Sigma}_1^{\mathrm{p}} = \mathbf{NP}(\mathbf{\Sigma}_0^{\mathrm{p}}) = \mathbf{NP}(\mathbf{P}) = \mathbf{NP}$. For a class $\mathbf{C}$ the class **co-C** consists of the complements of the sets in $\mathbf{C}$. In terms of problems, this means that no-instances take the role of yes-instances and vice versa. The class $\mathbf{\Pi}_1^{\mathrm{p}}$ is the class **co-NP**. The class $\mathbf{\Delta}_1^{\mathrm{p}}$ contains problems that are polynomial time deterministic Turing-reducible to classes $\mathbf{\Sigma}_0^{\mathrm{p}}$ (or $\mathbf{\Pi}_0^{\mathrm{p}}$). It follows that $\mathbf{\Delta}_1^{\mathrm{p}} = \mathbf{P}$. The higher levels are defined analogously. Given any $k \geq 0$, define $\mathbf{\Sigma}_{k+1}^{\mathrm{p}} = \mathbf{NP}(\mathbf{\Sigma}_k^{\mathrm{p}})$, $\mathbf{\Pi}_{k+1}^{\mathrm{p}} = \mathbf{co\text{-}NP}(\mathbf{\Sigma}_{\mathbf{k}}^{\mathrm{p}})$ and $\mathbf{\Delta}_{k+1}^{\mathrm{p}} = \mathbf{P}(\mathbf{\Sigma}_k^{\mathrm{p}}) = \mathbf{P}(\mathbf{\Pi}_k^{\mathrm{p}})$. The classes of PH form a hierarchy of subclasses as follows: $\mathbf{\Sigma}_k^{\mathrm{p}} \cup \mathbf{\Pi}_k^{\mathrm{p}} \subseteq \mathbf{\Delta}_{k+1}^{\mathrm{p}} \subseteq \mathbf{\Sigma}_{k+1}^{\mathrm{p}} \cap \mathbf{\Pi}_{k+1}^{\mathrm{p}}$ for all $k \geq 0$.

7.1.3 Reducibilities and Hard Problems

In complexity analysis, the *reducibility* of a problem to another is a very central concept. A frequently used reduction is *polynomial time (many-to-one) reducibility* or simply *m-reducibility* for short. A set A_1 is m-reducible to a set A_2, denoted by $A_1 \leq_{\mathrm{m}}^{\mathrm{p}} A_2$, if and only if there exists a function $f : A_1 \to A_2$ computable by a deterministic Turing machine in time polynomial to the length of a such that $a \in A_1$ if and only if $f(a) \in A_2$. One can understand such a function f as a polynomial-time transformation of problem instances. We write $A_1 \equiv_{\mathrm{m}}^{\mathrm{p}} A_2$ to denote that $A_1 \leq_{\mathrm{m}}^{\mathrm{p}} A_2$ and $A_2 \leq_{\mathrm{m}}^{\mathrm{p}} A_1$. Every class of PH is closed with respect to m-reducibility. On the other hand, oracle machines give rise to another kind of reducibility between sets as follows. A set A_1 is polynomial time deterministic (nondeterministic) Turing-reducible to a set A_2 if A_1 accepted in polynomial time by a deterministic (nondeterministic) Turing machine with oracle set A_2. Each of the classes Δ_k for $k \geq 0$ is closed with respect to polynomial time deterministic Turing-reducibility.

Reducibilities offer us the means to define the hardest problems in a class of problems $\mathbf{C}$ in the following way. For a class $\mathbf{C}$ of sets, we say that a set A is $\mathbf{C}$-*hard* (with respect to m-reducibility) if every set A' in $\mathbf{C}$ can be m-reduced to A in polynomial time. We use only m-reducibility to define hardness with respect to a class of sets. A set A is $\mathbf{C}$-*complete* if A belongs to $\mathbf{C}$ and A is $\mathbf{C}$-hard. For example, an $\mathbf{NP}$-complete set A belongs to the class $\mathbf{NP}$ and all problems of the class $\mathbf{NP}$ are m-reducible to A. This is how the $\mathbf{NP}$-complete problems are the hardest problems in $\mathbf{NP}$. Note also that every problem in $\mathbf{P}$ is also in $\mathbf{NP}$, since deterministic Turing machines form a subclass of nondeterministic Turing machines. A widely accepted hypothesis is that the inclusion of $\mathbf{P}$ and $\mathbf{NP}$ is proper and a deterministic Turing machine cannot solve an $\mathbf{NP}$-complete problem in polynomial time in general. This is because there is so much practical evidence on the matter: despite of many attempts **no** algorithm solving $\mathbf{NP}$-complete problems in polynomial time has been found. On the other hand, **if** such an algorithm is found, then every $\mathbf{NP}$-complete problem becomes solvable in polynomial time.

7.2 Complexity of Non-monotonic Systems

This section is devoted to analyze the computational complexity of the non-monotonic systems $\mathfrak{C}_P$, $\mathfrak{D}_D$ and $\mathcal{A}_\Sigma$ associated with circumscription, default logic and autoepistemic logic, respectively. Each of these systems carries a parameter set: P is set set of atoms to be minimized, D is a set of defaults and Σ is a set of autoepistemic sentences. Let us now abstract on these systems and parameter sets and write $\mathfrak{N}_X$ to denote any of these systems parametrized by a set X. In the sequel, decision problems and algorithms will be presented relative to the system $\mathfrak{N}_X$. Consequently, the parameter set X of a system $\mathfrak{N}_X$ is visible in algorithms, i.e. considered as one of the inputs. This is an important detail to be taken into account, since the time bounds of Turing machines are given relative to the length of problem instances, and we have to take the length of the set of parameters X into account. There is no doubt, e.g., that the length of a set of defaults D affects the time requirements of an algorithm implementing the stable semantics of a default system $\mathfrak{D}_D$.

In Definition 7.1, we introduce the basic decision problem $\mathbf{TH}_{\mathrm{L}}$ which is the *theorem problem* of the non-monotonic logic L under consideration and to which our complexity results will be related. This decision problem corresponds to deciding whether a sentence $\phi \in \mathcal{L}^+(\mathfrak{N}_X)$ belongs to the set of *theorems* $\mathfrak{N}_X(T, N)$ given *finite* set of sentences $T \subseteq \mathcal{L}^+(\mathfrak{N}_X)$ and $N \subseteq \mathcal{L}^-(\mathfrak{N}_X)$ where $\mathfrak{N}_X$ is the system induced by a *finite* set of parameters X. The parameter set X is assumed to range in the domain specific to the logic L under consideration. Let us yet emphasize how $\mathbf{TH}_{\mathrm{L}}$ consists exactly of the yes-instances of the decision problem in question. In this section, we analyze the time complexity of the problems $\mathbf{TH}_{\mathrm{CIRC}}$, $\mathbf{TH}_{\mathrm{DL}}$ and $\mathbf{TH}_{\mathrm{AEL}}$.

Definition 7.1

$$\mathbf{TH}_{\mathrm{L}} \textit{ is the set of quadruples } \langle X, T, N, \phi \rangle \textit{ such that } \phi \in \mathfrak{N}_X(T, N)$$

where X is a finite set of parameters in logic L, sets $T \subseteq \mathcal{L}^+(\mathfrak{N}_X)$ and $N \subseteq \mathcal{L}^-(\mathfrak{N}_X)$ are finite sets of sentences and $\phi \in \mathcal{L}^+(\mathfrak{N}_X)$ is a sentence.

The finiteness of the sets X, T and N is essential, because we have to be able to represent these sets on our model of computation in finite space. For the same reason, we have to assume that the sentences of the language of $\mathfrak{N}_X$ as well as the parameters of X can be represented in finite space. This holds of course for the languages and the sets of parameters associated with the non-monotonic systems presented in Section 2.3. Given a set S, we adopt the notation $||S||$ used by Gottlob [38] to denote the *length* of S, i.e. space required to represent the set S symbolically. Note that this differs from the cardinality of S which is denoted by $|S|$ and clearly $|S| < ||S||$ for all S. Quite similarly,

$||e||$ gives the length of an element $e \in S$. Our assumptions on the sets X, T, N as well as the languages and parameters associated with the non-monotonic systems imply that the lengths $||X||$, $||T||$ and $||N||$ are finite.

Our next goal is to analyze the computational complexity of the problem $\mathbf{TH}_{\mathrm{L}}$ for the three logics L under consideration. For this purpose, we explicate in Definitions 7.2 and 7.3 related decision problems **CN**, **CNI** and **CNN** for minimal model systems, default systems and autoepistemic systems, respectively. The problem **SAT** is a variant[28] of the famous *satisfiability problem* of propositional logic [14]. The problem **SAT** is **NP**-complete (i.e. $\mathbf{\Sigma}_1^{\mathrm{p}}$-complete) and therefore decidable. Decision procedures for **SAT** can be obtained e.g. from Smullyan's tableaux systems [103]. There are also more efficient procedures that solve this problem such as the Davis-Putnam procedure [27].

Definition 7.2

> **SAT** *is the set of sets* Σ *such that* Σ *is satisfiable,*
>
> **CN** *is the set of pairs* $\langle \Sigma, \phi \rangle$ *such that* $\phi \in \mathrm{Cn}(\Sigma)$ *and*
>
> **CNI** *is the set of triples* $\langle \Sigma, R, \phi \rangle$ *such that* $\phi \in \mathrm{Cn}^R(\Sigma)$

where $\Sigma \subseteq \mathcal{L}$ *is a finite set of sentences,* R *a finite set of inference rules and* $\phi \in \mathcal{L}$ *is a sentence.*

Definition 7.3

> **CNN** *is the set of pairs* $\langle \Sigma, \phi \rangle$ *such that* $\phi \in \mathrm{Cn}_{\mathbf{B}}(\Sigma)$

where $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ *is finite set of sentences and* $\phi \in \mathcal{L}_{\mathrm{BD}}$ *a sentence.*

Let us now classify the problems **CN**, **CNI** and **CNN** in the polynomial time hierarchy. It is established in Propositions 7.4 and 7.5 that **CN** $\leq_{\mathrm{m}}^{\mathrm{p}}$ **CNN** $\leq_{\mathrm{m}}^{\mathrm{p}}$ **CNI** where the problems **CN** and **CNI** are $\mathbf{\Pi}_1^{\mathrm{p}}$-complete (**co-NP**-complete).

Proposition 7.4 **CN** $\leq_{\mathrm{m}}^{\mathrm{p}}$ **CNN**.

Proof. There is no need for a transformation, since $\langle \Sigma, \phi \rangle \in$ **CN** if and only if $\langle \Sigma, \phi \rangle \in$ **CNN** by the definition of **B**-provability and Lemma 3.16. Note that Σ and ϕ above are propositional and thus they do not contain **BD**-atoms. □

Proposition 7.5 **CNN** $\leq_{\mathrm{m}}^{\mathrm{p}}$ **CNI**.

[28] Typically, this problem is defined as the set of satisfiable sentences, rather than satisfiable finite sets of sentences. However, the two approaches are equivalent under $\leq_{\mathrm{m}}^{\mathrm{p}}$.

Proof. An instance $\langle \Sigma, \phi \rangle$ of the problem **CNN** is transformed to an instance $\langle \Sigma, R, \phi \rangle$ of the problem **CNI** where R is the set of inference rules $\{\frac{\phi}{\mathbf{B}\phi} \mid \mathbf{B}\phi \in \mathrm{At}^*_{\mathbf{B}}(\Sigma \cup \{\phi\})\}$. It is obvious that a deterministic Turing machine can compute R by one linear pass of Σ and ϕ. By Lemma 3.16 we may assume that any **B**-proof of ϕ from Σ contains only **B**-atoms from $\mathrm{At}^*_{\mathbf{B}}(\Sigma \cup \{\phi\})$. Then it is easy to see that ϕ has a **B**-proof from Σ if and only if ϕ has an R-proof from Σ. This implies that $\langle \Sigma, \phi \rangle \in \mathbf{CNN}$ if and only if $\langle \Sigma, R, \phi \rangle \in \mathbf{CNI}$. □

The problems **CN**, **CNI** and **CNN** are classified in Theorem 7.6. The novelty of the theorem is that **CNN** is $\mathbf{\Pi}^{\mathrm{p}}_1$-complete (i.e. **co-NP**-complete) even if **CNN** is based on a bimodal language $\mathcal{L}_{\mathrm{BD}}$. Gottlob [37, 39] and also Schwarz and Truszczyński [102] have established this result in the unimodal case.

Theorem 7.6 *The problems* **CN**, **CNI** *and* **CNN** *given in Definitions 7.2 and 7.3 are all* $\mathbf{\Pi}^{\mathrm{p}}_1$*-complete (***co-NP***-complete).*

Proof. The problem **CN** is $\mathbf{\Pi}^{\mathrm{p}}_1$-complete. This is obvious by the following m-reducibilities. First of all, it holds that $\mathbf{CN} \leq^{\mathrm{p}}_{\mathrm{m}} \overline{\mathbf{SAT}}$, since $\langle \Sigma, \phi \rangle \in \mathbf{CN}$ if and only if $\Sigma \cup \{\neg\phi\} \in \overline{\mathbf{SAT}}$. Moreover, we know that $\overline{\mathbf{SAT}} \leq^{\mathrm{p}}_{\mathrm{m}} \mathbf{CN}$, because $\Sigma \in \overline{\mathbf{SAT}}$ if and only if $\langle \Sigma, \bot \rangle \in \mathbf{CN}$. Since **SAT** is $\mathbf{\Sigma}^{\mathrm{p}}_1$-complete, the complementary problem $\overline{\mathbf{SAT}}$ is $\mathbf{\Pi}^{\mathrm{p}}_1$-complete. Because $\mathbf{CN} \equiv^{\mathrm{p}}_{\mathrm{m}} \overline{\mathbf{SAT}}$ by the above, it follows that **CN** is also $\mathbf{\Pi}^{\mathrm{p}}_1$-complete. Gottlob [38] has shown that **CNI** is $\mathbf{\Pi}^{\mathrm{p}}_1$-complete. Since **CN** and **CNI** are $\mathbf{\Pi}^{\mathrm{p}}_1$-complete, it holds that $\mathbf{CNI} \leq^{\mathrm{p}}_{\mathrm{m}} \mathbf{CN}$. Thus we obtain by Propositions 7.5 and the transitivity of $\leq^{\mathrm{p}}_{\mathrm{m}}$ that $\mathbf{CNN} \leq^{\mathrm{p}}_{\mathrm{m}} \mathbf{CN}$. It follows that $\mathbf{CN} \equiv^{\mathrm{p}}_{\mathrm{m}} \mathbf{CNN}$ by Proposition 7.4. Thus the $\mathbf{\Pi}^{\mathrm{p}}_1$-completeness of **CN** implies the $\mathbf{\Pi}^{\mathrm{p}}_1$-completeness of **CNN**. □

Having classified the problems **CN**, **CNI** and **CNN** in the polynomial time hierarchy, we obtain the decidability of these problems for free. This is because all problems in PH are decidable. However, let us concretize this result a little bit further. Since these problems are $\mathbf{\Pi}^{\mathrm{p}}_1$-complete and therefore m-reducible to $\overline{\mathbf{SAT}}$, any instance of these problems can be solved by transforming the instance to an instance of $\overline{\mathbf{SAT}}$ and applying a decision procedure for $\overline{\mathbf{SAT}}$. This is obtained from a decision procedure for **SAT** by changing the return value of the procedure (True is replaced with False and vice versa). On the other hand, Marek and Truszczyński [67, p. 44] and Gottlob [37] provide dedicated decision procedures for the problems **CNI** and **CNN**, respectively.

Corollary 7.7 *The problems* **CN**, **CNI** *and* **CNN** *are decidable.*

Theorem 7.6 provides us the means to classify the problem $\mathbf{TH}_{\mathrm{L}}$ in the polynomial time hierarchy. Theorem 7.8 indicates that the non-monotonic systems introduced in Section 2.3 are of the same computational complexity. The obvious decidability result implied by Theorem 7.8 is given as Corollary 7.9.

Theorem 7.8 *The problems* $\mathbf{TH}_{\mathrm{CIRC}}$, $\mathbf{TH}_{\mathrm{DL}}$ *and* $\mathbf{TH}_{\mathrm{AEL}}$ *are* $\mathbf{\Pi}_1^{\mathrm{p}}$*-complete.*

Proof. Consider the problem $\mathbf{TH}_{\mathrm{CIRC}}$ based on minimal model systems $\mathfrak{C}_P$ defined by $\mathfrak{C}_P(T,N) = \mathrm{Cn}(T \cup \mathrm{Neg}_P(N))$ for all $T \subseteq \mathcal{L}$ and $N \subseteq \mathcal{L}$. Let T and N be finite subsets of $\mathcal{L}$ and $\phi \in \mathcal{L}$. A deterministic Turing machine can compute the theory $T \cup \mathrm{Neg}_P(N)$ in time polynomial to $||T|| + ||N|| + ||P||$. Since $\langle P, T, N, \phi \rangle \in \mathbf{TH}_{\mathrm{CIRC}}$ if and only if $\langle T \cup \mathrm{Neg}_P(N), \phi \rangle \in \mathbf{CN}$, it follows that $\mathbf{TH}_{\mathrm{CIRC}} \leq_{\mathrm{m}}^{\mathrm{p}} \mathbf{CN}$. Moreover, $\langle \Sigma, \phi \rangle \in \mathbf{CN}$ if and only if $\langle \emptyset, \Sigma, \emptyset, \phi \rangle \in \mathbf{TH}_{\mathrm{CIRC}}$. This implies that $\mathbf{CN} \leq_{\mathrm{m}}^{\mathrm{p}} \mathbf{TH}_{\mathrm{CIRC}}$ so that $\mathbf{TH}_{\mathrm{CIRC}} \equiv_{\mathrm{m}}^{\mathrm{p}} \mathbf{CN}$. Since $\mathbf{CN}$ is $\mathbf{\Pi}_1^{\mathrm{p}}$-complete by Theorem 7.6, so is the problem $\mathbf{TH}_{\mathrm{CIRC}}$.

Consider the problem $\mathbf{TH}_{\mathrm{DL}}$ based on default systems $\mathfrak{D}_D$ which are defined by $\mathfrak{D}_D(T,N) = \mathrm{Cn}^{D_N}(T)$ for all $T \subseteq \mathcal{L}$ and $N \subseteq \mathcal{L}$. Then let T and N be finite subsets of $\mathcal{L}$ and $\phi \in \mathcal{L}$. The reduct D_N can be computed by a deterministic Turing machine in time polynomial to $||D|| + ||N||$. Because $\langle D, T, N, \phi \rangle \in \mathbf{TH}_{\mathrm{DL}}$ if and only if $\langle T, D_N, \phi \rangle \in \mathbf{CNI}$, it holds that $\mathbf{TH}_{\mathrm{DL}} \leq_{\mathrm{m}}^{\mathrm{p}} \mathbf{CNI}$. On the other hand, $\langle \Sigma, \phi \rangle \in \mathbf{CN}$ if and only if $\langle \emptyset, \Sigma, \emptyset, \phi \rangle \in \mathbf{TH}_{\mathrm{DL}}$. Thus $\mathbf{CN} \leq_{\mathrm{m}}^{\mathrm{p}} \mathbf{TH}_{\mathrm{DL}} \leq_{\mathrm{m}}^{\mathrm{p}} \mathbf{CNI}$. The problem $\mathbf{TH}_{\mathrm{DL}}$ is $\mathbf{\Pi}_1^{\mathrm{p}}$-complete, since $\mathbf{CN}$ and $\mathbf{CNI}$ are $\mathbf{\Pi}_1^{\mathrm{p}}$-complete problems by Theorem 7.6.

Finally, consider the problem $\mathbf{TH}_{\mathrm{AEL}}$ based on autoepistemic systems $\mathfrak{A}_\Sigma$ defined by $\mathfrak{A}_\Sigma(T,N) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup T \cup \mathbf{D}N)$ for all $T \subseteq \mathcal{L}_{\mathrm{BD}}$ and $N \subseteq \mathcal{L}_{\mathrm{BD}}$. Then let T and N be finite subsets of $\mathcal{L}_{\mathrm{BD}}$ and $\phi \in \mathcal{L}_{\mathrm{BD}}$. A deterministic Turing machine can compute the theory $\Sigma \cup T \cup \mathbf{D}N$ in time polynomial to $||\Sigma|| + ||T|| + ||N||$. Since $\langle \Sigma, T, N, \phi \rangle \in \mathbf{TH}_{\mathrm{AEL}}$ if and only if $\langle \Sigma \cup T \cup \mathbf{D}N, \phi \rangle \in \mathbf{CNN}$, it follows that $\mathbf{TH}_{\mathrm{AEL}} \leq_{\mathrm{m}}^{\mathrm{p}} \mathbf{CNN}$. In addition, it holds that $\langle \Sigma, \phi \rangle \in \mathbf{CNN}$ if and only if $\langle \emptyset, \Sigma, \emptyset, \phi \rangle \in \mathbf{TH}_{\mathrm{AEL}}$ so that $\mathbf{CNN} \leq_{\mathrm{m}}^{\mathrm{p}} \mathbf{TH}_{\mathrm{AEL}}$ and $\mathbf{TH}_{\mathrm{AEL}} \equiv_{\mathrm{m}}^{\mathrm{p}} \mathbf{CNN}$. Consequently, the problem $\mathbf{TH}_{\mathrm{AEL}}$ is $\mathbf{\Pi}_1^{\mathrm{p}}$-complete. □

Corollary 7.9 *The problems* $\mathbf{TH}_{\mathrm{CIRC}}$, $\mathbf{TH}_{\mathrm{DL}}$ *and* $\mathbf{TH}_{\mathrm{AEL}}$ *are decidable.*

7.3 Complexity of Stable and Stationary Semantics

The time complexity of stable and stationary semantics is analyzed next. So far we have formulated the decision problem $\mathbf{TH}_{\mathrm{L}}$ for non-monotonic logics under consideration. Now we turn our attention to the semantics introduced in Section 4 and identify the decision problems of our interest. The sets $\mathbf{STAB}_{\mathrm{L}}$ and $\mathbf{STAT}_{\mathrm{L}}$ given in Definition 7.10 are the decision problems for the stable semantics and the stationary semantics of a non-monotonic logic L. By this definition, we restrict our analysis mainly to the skeptical reasoning strategy as set up in Section 4.1. In the literature, the complexity of the credulous reasoning strategy has also been studied. In this strategy, the problem is to find out whether ϕ belongs to the set of theorems $\mathfrak{N}_X(T,N)$ for *some* set of non-theorems N satisfying the criteria of a semantics (such as $\mathcal{N}_X[T]$-stability or $\mathcal{N}_X[T]$-stationarity). The complexity of the decision problems related to the credulous approach can also be analyzed within our framework, but this is not done in detail in this work. Nevertheless, we point out the differences to the skeptical approach when Theorems 7.18 and 7.20 concerning the complexity of stable and stationary semantics are presented.

Definition 7.10

> $\mathbf{STAB}_{\mathrm{L}}$ *is the set of triples* $\langle X, T, \phi \rangle$ *such that* $\phi \in \mathrm{STAB}_{\mathcal{N}_X}(T)$ *and*
>
> $\mathbf{STAT}_{\mathrm{L}}$ *is the set of triples* $\langle X, T, \phi \rangle$ *such that* $\phi \in \mathrm{STAT}_{\mathcal{N}_X}(T)$

where X *is a finite set of parameters in logic L,* $T \subseteq \mathcal{L}^+(\mathcal{N}_X)$ *is a finite set of sentences and* $\phi \in \mathcal{L}^+(\mathcal{N}_X)$ *is a sentence.*

Our next objective is to develop decision procedures operating on Turing machines for these decision problems. Since we are willing to reduce the problems $\mathbf{STAB}_{\mathrm{L}}$ and $\mathbf{STAT}_{\mathrm{L}}$ to the problem $\mathbf{TH}_{\mathrm{L}}$, we assume there is a function TH_X[29] which returns for all finite sets of parameters X and for all inputs $T \subseteq_{\mathrm{f}} \mathcal{L}^+(\mathcal{N}_X)$, $N \subseteq_{\mathrm{f}} \mathcal{L}^-(\mathcal{N}_X)$ and $\phi \in \mathcal{L}^+(\mathcal{N}_X)$ a *Boolean value* True or False to indicate whether $\phi \in \mathcal{N}_X(T, N)$ or not. It is possible to regard the function TH_X as a decision procedure for the problem $\mathbf{TH}_{\mathrm{L}}$ or as an oracle for the set $\mathbf{TH}_{\mathrm{L}}$ depending on the context, i.e. whether we have decidability or reducibility results in mind. Recall that such a decision procedure exists by Corollary 7.9 for the non-monotonic logics L under consideration.

Our first algorithm in Figure 5 computes the set of non-theorems $\overline{\mathcal{S}[T]}(N)$ for a subsystem $\mathcal{S}$ of $\mathcal{N}_X$ (the name NT_X is derived from the word non-theorem). It is necessary to assume $\mathcal{L}^-(\mathcal{S})$ to be finite, as $\mathcal{L}^-(\mathcal{S})$ is passed as the input L^- to the algorithm and if L^- were infinite, the **For**-loop would not terminate.

The correctness of the algorithm in Figure 5 is addressed in Proposition 7.12. In Lemma 7.11, we note that the decision procedure TH_X can be used to solve queries concerning a subsystem $\mathcal{S} \sqsubseteq \mathcal{N}_X$.

Lemma 7.11 *If there is a subsystem* $\mathcal{S} \sqsubseteq \mathcal{N}_X$ *such that* $T \cup \{\phi\} \subseteq \mathcal{L}^+(\mathcal{S})$ *and* $N \subseteq \mathcal{L}^-(\mathcal{S})$*, the call* $\mathsf{TH}_X(T, N, \phi)$ *returns* True *if and only if* $\phi \in \mathcal{S}(T, N)$.

Proof. Since $\mathcal{S} \sqsubseteq \mathcal{N}_X$, $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $N \subseteq \mathcal{L}^-(\mathcal{S})$, it follows that $\mathcal{S}(T, N) = \mathcal{N}_X(T, N) \cap \mathcal{L}^+(\mathcal{S})$ by the definition of subsystems. Since $\phi \in \mathcal{L}^+(\mathcal{S})$, it holds that $\phi \in \mathcal{N}_X(T, N)$ if and only if $\phi \in \mathcal{S}(T, N)$ and the call $\mathsf{TH}_X(T, N, \phi)$ returns True if and only if $\phi \in \mathcal{S}(T, N)$ by our assumption on TH_X. □

[29] Recall that the set of parameters X is global and considered to be one of the inputs.

Function $\mathsf{NT}_X(T, N, L^-)$: a finite subset of $\mathcal{L}^-(\mathbf{N}_X)$;
Var
  T: a finite subset of $\mathcal{L}^+(\mathbf{N}_X)$;
  N, M, L^-: finite subsets of $\mathcal{L}^-(\mathbf{N}_X)$;
  ϕ: a sentence of $\mathcal{L}^+(\mathbf{N}_X)$
Begin
  $M := \emptyset$;
  For $\phi \in L^-$ **Do**
    If Not $\mathsf{TH}_X(T, N, \phi)$ **Then** $M := M \cup \{\phi\}$;
  Return M
End

Figure 5: A function computing $\overline{\mathcal{S}[T]}(N)$ for $\mathcal{S} \sqsubseteq \mathbf{N}_X$

Proposition 7.12 *If there is a subsystem $\mathcal{S} \sqsubseteq \mathbf{N}_X$ such that $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $N \subseteq \mathcal{L}^-(\mathcal{S}) = L^-$, the call $\mathsf{NT}_X(T, N, L^-)$ returns the set of non-theorems $\overline{\mathcal{S}[T]}(N)$ by making a total of $|L^-|$ calls to the function TH_X.*

Proof. A sentence $\phi \in L^-$ is included in M if and only if $\mathsf{TH}_X(T, N, \phi)$ returns False. By Lemma 7.11 this is the case if and only if $\phi \notin \mathcal{S}(T, N)$. By these arguments, the variable M has the value $\{\phi \in L^- \mid \phi \notin \mathcal{S}[T](N)\} = \overline{\mathcal{S}[T]}(N)$ at the end of the **For**-loop, since $L^- = \mathcal{L}^-(\mathcal{S}) = \mathcal{L}^-(\mathcal{S}[T])$. Moreover, exactly $|L^-|$ calls to the function TH_X are required, because the **For**-loop is executed once for each sentence $\phi \in L^-$. □

To prepare the classification of the problem $\mathbf{STAB}_{\mathrm{L}}$ in the polynomial time hierarchy, we provide a reduction of $\overline{\mathbf{STAB}_{\mathrm{L}}}$ to the problem $\mathbf{TH}_{\mathrm{L}}$. A non-deterministic Turing machine for this purpose is presented in Figure 6. The overall idea in the algorithm is to choose nondeterministically an $\mathcal{S}[T]$-stable set of non-theorems N that serves as a counter-example for the query ϕ with respect to stable semantics, i.e. $\phi \notin \mathcal{S}[T](N)$.

Here we assume that calls to the function TH_X are answered by an oracle for $\mathbf{TH}_{\mathrm{L}}$ in one time unit. The purpose of the function LangN_X is to compute the language $\mathcal{L}^-(\mathcal{S})$ for a subsystem $\mathcal{S}$ of $\mathbf{N}_X$ such that $T \cup \{\phi\} \subseteq \mathcal{L}^+(\mathcal{S})$ and $\mathcal{L}^-(\mathcal{S})$ is finite. In Lemma 7.13, we point out the languages of subsystems that have to be computed for algorithms involving minimal model systems, default systems and autoepistemic systems (see Section 3.1 for subsystems).

Lemma 7.13 *The call $\mathsf{LangN}_X(T, \phi)$ computes in time polynomial to $||X|| + ||T|| + ||\phi||$ the language $\mathcal{L}^-(\mathcal{S})$ for a subsystem $\mathcal{S} \sqsubseteq \mathbf{N}_X$ such that the length $||\mathcal{L}^-(\mathcal{S})||$ is polynomial in $||X|| + ||T|| + ||\phi||$ and $T \cup \{\phi\} \subseteq \mathcal{L}^+(\mathcal{S})$.*

Machine $\mathsf{CoStab}_X(T,\phi)$;
Var
  T: a finite subset of $\mathcal{L}^+(\mathfrak{N}_X)$;
  N, L^-: finite subsets of $\mathcal{L}^-(\mathfrak{N}_X)$;
  ϕ: a sentence of $\mathcal{L}^+(\mathfrak{N}_X)$
Begin
  $L^- := \mathsf{LangN}_X(T,\phi)$;
  Choose N **From** $\mathbf{2}^{L^-}$;
  If $N = \mathsf{NT}_X(T,N,L^-)$ **And Not** $\mathsf{TH}_X(T,N,\phi)$ **Then**
    Accept
  Else
    Reject
End

Figure 6: An oracle machine accepting $\overline{\mathbf{STAB}_{\mathrm{L}}}$

Proof. For a minimal model system $\mathfrak{C}_P$, the subsystem $\mathfrak{S} = \mathfrak{C}_P^P$ is used. The language $\mathcal{L}^-(\mathfrak{S}) = P$ can be computed in constant time, the length $||\mathcal{L}^-(\mathfrak{S})||$ is linear in $||P||$ and clearly $T \cup \{\phi\} \subseteq \mathcal{L}^+(\mathfrak{S}) = \mathcal{L}$. For a default system $\mathfrak{D}_D$, the subsystem $\mathfrak{S} = \mathfrak{D}_D^{\neg \mathrm{J}(D)}$ is used. The language $\mathcal{L}^-(\mathfrak{S}) = \neg\mathrm{J}(D)$ can be computed in time quadratic to $||D||$, the length $||\mathcal{L}^-(\mathfrak{S})||$ is linear in $||D||$ and $T \cup \{\phi\} \subseteq \mathcal{L}^+(\mathfrak{S}) = \mathcal{L}$. For an autoepistemic system $\mathcal{A}_\Sigma$, the subsystem $\mathfrak{S} = \mathcal{A}_\Sigma^\Gamma$ where $\Gamma = \mathbf{B}\Sigma \cup \mathbf{B}T \cup \{\mathbf{B}\phi\}$ is used. The language $\mathcal{L}^-(\mathfrak{S}) = \mathrm{SS}^*_{\mathbf{D}}(\Gamma)$ can be computed in time quadratic to $||\Sigma|| + ||T|| + ||\phi||$, the length $||\mathcal{L}^-(\mathfrak{S})||$ is quadratic to $||\Sigma|| + ||T|| + ||\phi||$ and $T \cup \{\phi\} \subseteq \mathcal{L}^+(\mathfrak{S}) = \mathrm{SS}^*_{\mathbf{BD}}(\Gamma)$. $\square$

A reducibility result for $\overline{\mathbf{STAB}_{\mathrm{L}}}$ is worked out in Proposition 7.14. The machine CoStab_X involves a nondeterministic choice indicating that we are using nondeterministic Turing-reducibility to reduce $\overline{\mathbf{STAB}_{\mathrm{L}}}$ to $\mathbf{TH}_{\mathrm{L}}$.

Proposition 7.14 $\overline{\mathbf{STAB}_{\mathrm{L}}}$ *is nondeterministic Turing-reducible to* $\mathbf{TH}_{\mathrm{L}}$ *in polynomial time for circumscription, default logic and autoepistemic logic.*

Proof. It is shown that the machine CoStab_X accepts the set $\overline{\mathbf{STAB}_{\mathrm{L}}}$, i.e. CoStab_X has an accepting computation on the inputs X, T and ϕ if and only if $\langle X, T, \phi\rangle \in \overline{\mathbf{STAB}_{\mathrm{L}}}$. By Lemma 7.13, the language L^- computed by the call $\mathsf{LangN}_X(T,\phi)$ is the language $\mathcal{L}^-(\mathfrak{S})$ for a subsystem $\mathfrak{S} \sqsubseteq \mathfrak{N}_X$ such that $T \cup \{\phi\} \subseteq \mathcal{L}^+(\mathfrak{S})$. The machine CoStab_X has an accepting computation on the inputs X, T and ϕ if and only if the set of non-theorems N is chosen such that the call $\mathsf{NT}_X(T,N,L^-)$ returns N and the call $\mathsf{TH}_X(T,N,\phi)$ returns False. By Proposition 7.12 and Lemma 7.11, this is the case if and only if $N = \overline{\mathfrak{S}[T]}(N)$ and $\phi \notin \mathfrak{S}[T](N)$, i.e. $\phi \notin \mathrm{STAB}_{\mathfrak{S}}(T)$. Equivalently, we have that $\phi \notin \mathrm{STAB}_{\mathfrak{N}_X}(T)$, i.e. $\langle X, T, \phi\rangle \in \overline{\mathbf{STAB}_{\mathrm{L}}}$, since $T \cup \{\phi\} \subseteq \mathcal{L}^+(\mathfrak{S})$ by

Lemma 7.13 and $\mathrm{STAB}_{\mathbb{S}}(T) = \mathrm{STAB}_{\mathbf{N}_X}(T) \cap \mathcal{L}^+(\mathbb{S})$ by Theorem 6.17.

The time requirement of the machine CoStab_X is verified as follows. The set L^- is computed in time polynomial to the length of input (i.e. $||X|| + ||T|| + ||\phi||$) by Lemma 7.13. Moreover, the length $||L^-||$ is polynomial to the length of input. Thus, the selection of a subset N of $\mathcal{L}^-(\mathbb{S})$ requires a total of $|L^-|$ nondeterministic choices which can be completed in polynomial time. By Proposition 7.12 the call $\mathsf{NT}_X(T, N, L^-)$ causes $|L^-|$ calls to TH_X each of which is completed in one time unit by our oracle. Since $|L^-| < ||L^-||$ is polynomial to the length of input, the call $\mathsf{NT}_X(T, N, L^-)$ is finished in polynomial time. Finally, the oracle call $\mathsf{TH}_X(T, N, \phi)$ takes only one time unit. By these observations, the machine CoStab_X accepts the set $\overline{\mathbf{STAB}_{\mathrm{L}}}$ in time polynomial to the length of its input. □

Our next effort is to devise a decision procedure for the problem $\mathbf{STAT}_{\mathrm{L}}$. In Figure 7, we present a function $\mathsf{Iterate}_X$ for the computation of the least fixed point of an operator $\overline{\mathbb{S}[T;B]}^2$ where $B \subseteq \mathcal{L}^-(\mathbb{S})$ is an additional set of sentences to be fixed as non-theorems. The extra argument B is reserved for later use in Section 8. Meanwhile, the argument B is always $\emptyset$ in conjunction with stationary semantics. Again, the computations are restricted to a subsystem $\mathbb{S}$ of $\mathbf{N}_X$. The correctness and time requirements of the function $\mathsf{Iterate}_X$ are analyzed in Proposition 7.15.

```
Function Iterate_X(T, B, L⁻): a finite subset of ℒ⁻(N_X);
Var
  T: a finite subset of ℒ⁺(N_X);
  B: a finite subset of ℒ⁻(N_X);
  L⁻, N, N′, M: finite subsets of ℒ⁻(N_X)
Begin
  N := ∅;
  Repeat
    M := N;
    N′ := NT_X(T, B ∪ N, L⁻);
    N := NT_X(T, B ∪ N′, L⁻)
  Until N = M;
  Return N
End
```

Figure 7: A function computing $\mathrm{lfp}(\overline{\mathbb{S}[T;B]}^2, \emptyset)$ for $\mathbb{S} \sqsubseteq \mathbf{N}_X$

Proposition 7.15 *If there is $\mathbb{S} \sqsubseteq \mathbf{N}_X$ such that $T \subseteq \mathcal{L}^+(\mathbb{S})$ and $B \subseteq \mathcal{L}^-(\mathbb{S}) = L^-$, the call* $\mathsf{Iterate}_X(T, B, L^-)$ *returns* $\mathrm{lfp}(\overline{\mathbb{S}[T;B]}^2, \emptyset)$ *by making at most a total of* $2 \times |L^-|^2 + 2 \times |L^-|$ *calls to* TH_X.

Proof. Consider a single execution of the **Repeat**-loop. At first, the variable M is assigned the old value of N. Then the variables N' and N are assigned the expressions $\mathsf{NT}_X(T, B \cup N, L^-)$ and $\mathsf{NT}_X(T, B \cup N', L^-)$ (respectively) computed by two subsequent calls to the function NT_X. By Proposition 7.12, the call $\mathsf{NT}_X(T, B \cup N, L^-)$ returns the value $\overline{\mathcal{S}[T]}(B \cup N) = \overline{\mathcal{S}[T;B]}(N)$ for N' so that the call $\mathsf{NT}_X(T, B \cup N', L^-)$ returns $\overline{\mathcal{S}[T]}(B \cup \overline{\mathcal{S}[T;B]}(N)) = \overline{\mathcal{S}[T;B]}(\overline{\mathcal{S}[T;B]}(N)) = \overline{\mathcal{S}[T;B]}^2(N)$. Thus N is assigned to $\overline{\mathcal{S}[T;B]}^2(N)$ whose computation requires a total of $2 \times |L^-|$ calls to TH_X.

Let N_i denote the value of the variable N at the i^{th} iteration of the **Repeat**-loop. It is not difficult to see by our analysis of a single execution of the loop that the invariant $N_i = \overline{\mathcal{S}[T;B]}^2 \uparrow^i (\emptyset)$ is satisfied. Recall that by Proposition 4.25 the sequence N_i where $i = 0, 1, 2, \ldots$ is non-decreasing. Moreover, the value N_i reaches the fixed point $\mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$ at the latest when i reaches the value $|L^-|$. To see this, assume the converse. Then $N_i \subset N_{i+1}$ for all i in the range $0 \leq i \leq |L^-|$. This implies that $|N_{|L^-|+1}|$ is greater than $|L^-|$. But this is a contradiction, since the values of N are always subsets of L^-.

After N has reached the fixed point $\mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$, one more execution of the loop is needed to detect this. This updates the value of M to that of N so that the execution of the loop terminates. After this the function $\mathsf{Iterate}_X$ returns the current value of N, i.e. the fixed point $\mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$. In the worst case, the required computations make a total of $2 \times |L^-| \times (|L^-| + 1) = 2 \times |L^-|^2 + 2 \times |L^-|$ calls to the function TH_X. □

Our next objective is to derive a reducibility result for stationary semantics. Therefore, we introduce a deterministic Turing machine Stat_X in Figure 8. The heart of the algorithm is the call to the function $\mathsf{Iterate}_X$ to find out the least fixed point of $\overline{\mathcal{S}[T]}^2$. Recall our earlier assumption on the calls to the function TH_X which are answered by an oracle for $\mathbf{TH}_{\mathrm{L}}$. Proposition 7.16 reduces the problem $\mathbf{STAT}_{\mathrm{L}}$ to the problem $\mathbf{TH}_{\mathrm{L}}$.

Machine $\mathsf{Stat}_X(T, \phi)$;
Var
  T: a finite subset of $\mathcal{L}^+(\mathbf{N}_X)$;
  N, L^-: finite subsets of $\mathcal{L}^-(\mathbf{N}_X)$;
  ϕ: a sentence of $\mathcal{L}^+(\mathbf{N}_X)$
Begin
  $L^- := \mathsf{LangN}_X(T, \phi)$;
  $N := \mathsf{Iterate}_X(T, \emptyset, L^-)$;
  If $\mathsf{TH}_X(T, N, \phi)$ **Then Accept Else Reject**
End

Figure 8: An oracle machine accepting $\mathbf{STAT}_{\mathrm{L}}$

Proposition 7.16 $\mathbf{STAT}_{\mathrm{L}}$ *is deterministic Turing-reducible to* $\mathbf{TH}_{\mathrm{L}}$ *in polynomial time for circumscription, default logic and autoepistemic logic.*

Proof. By Lemma 7.13 the call $\mathsf{LangN}_X(T, \phi)$ returns $\mathcal{L}^-(\mathcal{S})$ for a subsystem $\mathcal{S}$ of $\mathcal{N}_X$ such that $T \cup \{\phi\} \subseteq \mathcal{L}^+(\mathcal{S})$. It is then shown that the machine Stat_X accepts the set $\mathbf{STAT}_{\mathrm{L}}$. It is clear by the description of the machine Stat_X that Stat_X has an accepting computation on the inputs T and ϕ if and only if the call $\mathsf{TH}_X(T, N, \phi)$ returns True where N is the return value of the call $\mathsf{Iterate}_X(T, \emptyset, L^-)$. By Proposition 7.15 and Lemma 7.11, this holds if and only if $\phi \in \mathcal{S}[T](N)$ for $N = \mathrm{lfp}(\overline{\mathcal{S}[T]}^2, \emptyset)$, i.e. $\phi \in \mathrm{STAT}_{\mathcal{S}}(T)$ by Theorem 4.28. Equivalently, $\phi \in \mathrm{STAT}_{\mathcal{N}_X}(T)$, i.e. $\langle X, T, \phi \rangle \in \mathbf{STAT}_{\mathrm{L}}$, since $T \cup \{\phi\}$ is contained in $\mathcal{L}^+(\mathcal{S})$ by Lemma 7.13 and $\mathcal{S} \sqsubseteq \mathcal{N}_X$ implies $\mathrm{STAT}_{\mathcal{S}}(T) = \mathrm{STAT}_{\mathcal{N}_X}(T) \cap \mathcal{L}^+(\mathcal{S})$ by Theorem 6.18.

The required computation time of the machine Stat_X is calculated as follows. By Proposition 7.15 the call $\mathsf{Iterate}_X(T, B, L^-)$ makes $2 \times |L^-|^2 + 2 \times |L^-|$ calls to TH_X. Each one of these calls is completed in one time unit by our oracle for $\mathbf{TH}_{\mathrm{L}}$. The same applies to the final call $\mathsf{TH}_X(T, N, \phi)$. Since $|L^-| < ||L^-||$ is polynomial to $||X|| + ||T|| + ||\phi||$, we conclude that the computation of the machine Stat_X takes a time polynomial to the length of its input. □

To prepare the exact classification of the problem $\mathbf{STAB}_{\mathrm{L}}$ in the polynomial time hierarchy, we show that it is $\mathbf{\Pi}^{\mathrm{p}}_2$-hard in Proposition 7.17.

Proposition 7.17 $\mathbf{STAB}_{\mathrm{L}}$ *is* $\mathbf{\Pi}^{\mathrm{p}}_2$*-hard for circumscription, default logic and the variant of autoepistemic logic proposed in this work.*

Proof. Eiter and Gottlob [25] have shown that $\mathbf{STAB}_{\mathrm{CIRC}}$ is $\mathbf{\Pi}^{\mathrm{p}}_2$-hard and Gottlob [37] has shown that $\mathbf{STAB}_{\mathrm{DL}}$ is $\mathbf{\Pi}^{\mathrm{p}}_2$-hard. By Corollary 6.23 it holds that $\langle D, T, \phi \rangle \in \mathbf{STAB}_{\mathrm{DL}}$ if and only if $\langle \mathrm{Tr}_{\mathrm{K}}(D), T, \phi \rangle \in \mathbf{STAB}_{\mathrm{AEL}}$. The transformation between the two problems is linear. This implies that the problem $\mathbf{STAB}_{\mathrm{DL}}$ is m-reducible the problem $\mathbf{STAB}_{\mathrm{AEL}}$, i.e. $\mathbf{STAB}_{\mathrm{DL}} \leq^{\mathrm{p}}_{\mathrm{m}} \mathbf{STAB}_{\mathrm{AEL}}$. Since $\mathbf{STAB}_{\mathrm{DL}}$ is $\mathbf{\Pi}^{\mathrm{p}}_2$-hard, so is $\mathbf{STAB}_{\mathrm{AEL}}$. □

Theorem 7.18 classifies the problem $\mathbf{STAB}_{\mathrm{L}}$ in the polynomial hierarchy. The $\mathbf{\Pi}^{\mathrm{p}}_2$-completeness of circumscription and default logic has been established by Eiter and Gottlob [25, 37]. Our contribution is that this theorem covers the bimodal variant of autoepistemic logic proposed in this work.

Theorem 7.18 $\mathbf{STAB}_{\mathrm{L}}$ *is* $\mathbf{\Pi}^{\mathrm{p}}_2$*-complete for circumscription, default logic and the variant of autoepistemic logic proposed in this work.*

Proof. By Theorem 7.8 the problem $\mathbf{TH}_{\mathrm{L}}$ is $\mathbf{\Pi}^{\mathrm{p}}_1$-complete. Since $\overline{\mathbf{STAB}_{\mathrm{L}}}$ is nondeterministic Turing-reducible to $\mathbf{TH}_{\mathrm{L}}$ by Proposition 7.14, it follows that

$\mathbf{STAB}_\mathrm{L} \in \mathbf{\Pi}^\mathrm{p}_2$. The $\mathbf{\Pi}^\mathrm{p}_2$-hardness of $\mathbf{STAB}_\mathrm{L}$ is established in Proposition 7.17 implying that $\mathbf{STAB}_\mathrm{L}$ is $\mathbf{\Pi}^\mathrm{p}_2$-complete. □

Let us now discuss how the result of Theorem 7.18 changes if the credulous approach is adopted. Let $\mathbf{STAB'}_\mathrm{L}$ be the corresponding decision problem, i.e. $\langle X, T, \phi\rangle \in \mathbf{STAB'}_\mathrm{L}$ if and only if $\phi \in \mathfrak{N}_X(T, N)$ for *some* $\mathfrak{N}_X[T]$-stable set of non-theorems N. This problem is nondeterministic Turing-reducible to the problem $\mathbf{TH}_\mathrm{L}$. To establish this, the oracle machine CoStab_X in Figure 6 has to be modified to accepts its input $\langle X, T, \phi\rangle$ if and only if $\mathsf{NT}_X(T, N, L)$ returns the chosen set of non-theorems N and $\mathsf{TH}_X(T, N, \phi)$ returns True. This reducibility result implies that $\mathbf{STAB'}_\mathrm{L}$ belongs to $\mathbf{\Sigma}^\mathrm{p}_2$ for the non-monotonic logics under consideration (that satisfy the assumption that $\mathbf{TH}_\mathrm{L}$ is $\mathbf{\Pi}^\mathrm{p}_1$-complete). Gottlob [37] has established that $\mathbf{STAB'}_\mathrm{DL}$ is $\mathbf{\Sigma}^\mathrm{p}_2$-complete. The $\mathbf{\Sigma}^\mathrm{p}_2$-hardness of $\mathbf{STAB'}_\mathrm{AEL}$ follows by Theorem 3.32 and Corollaries 6.15 and 6.16: a set of non-theorems N is $\mathfrak{D}_D[T]$-stable if and only if there is an $\mathfrak{A}_{\mathrm{Tr_K}(D)}[T]$-stable set of non-theorems N' such that $N = N' \cap \mathcal{L}$. Thus $\langle D, T, \phi\rangle \in \mathbf{STAB'}_\mathrm{DL}$ if and only if $\langle \mathrm{Tr_K}(D), T, \phi\rangle \in \mathbf{STAB'}_\mathrm{AEL}$, i.e. $\mathbf{STAB'}_\mathrm{DL} \leq^\mathrm{p}_\mathrm{m} \mathbf{STAB'}_\mathrm{AEL}$. To conclude, the decision problem of stable semantics tends to remain a complete problem on the second level of the polynomial time hierarchy even if the credulous approach is followed.

The classification of $\mathbf{STAT}_\mathrm{L}$ in the polynomial time hierarchy follows. Gottlob [38] has classified this problem for default logic. Our contribution limits to the bimodal autoepistemic logic proposed in this work. Note that this result cannot be established for conventional (unimodal) autoepistemic logic because of the consistency problem discussed in Section 4.3.2.

Proposition 7.19 $\mathbf{STAT}_\mathrm{L}$ *is* $\mathbf{\Delta}^\mathrm{p}_2$*-hard for default logic and the variant of autoepistemic logic proposed in this work.*

Proof. Gottlob [38] has shown that $\mathbf{STAT}_\mathrm{DL}$ is $\mathbf{\Delta}^\mathrm{p}_2$-hard. By Corollary 6.23 it holds that $\langle D, T, \phi\rangle \in \mathbf{STAT}_\mathrm{DL}$ if and only if $\langle \mathrm{Tr_K}(D), T, \phi\rangle \in \mathbf{STAT}_\mathrm{AEL}$. The transformation of D is linear so that $\mathbf{STAT}_\mathrm{DL} \leq^\mathrm{p}_\mathrm{m} \mathbf{STAT}_\mathrm{AEL}$ follows. Thus $\mathbf{STAT}_\mathrm{AEL}$ is $\mathbf{\Delta}^\mathrm{p}_2$-hard. □

Theorem 7.20 *(i)* $\mathbf{STAT}_\mathrm{CIRC}$ *belongs to* $\mathbf{\Delta}^\mathrm{p}_2$ *and (ii)* $\mathbf{STAT}_\mathrm{L}$ *is* $\mathbf{\Delta}^\mathrm{p}_2$*-complete for default logic and the variant of autoepistemic logic of this work.*

Proof. (i) Eiter and Gottlob [25] have shown that the decision problem associated Reiter's closed world assumption (arbitrary propositional queries) is in $\mathbf{\Delta}^\mathrm{p}_2$, i.e. $\mathbf{STAT}_\mathrm{CIRC} \in \mathbf{\Delta}^\mathrm{p}_2$. However, they argue that $\mathbf{STAT}_\mathrm{CIRC}$ is not $\mathbf{\Delta}^\mathrm{p}_2$-complete. Further evidence is provided by Gottlob [38]. He combines closed world assumption and *priorities* using a total ordering of the set atoms P to be minimized. Consequently, the modified decision problem becomes

$\mathbf{\Delta}_2^{\mathrm{p}}$-complete. (ii) By Theorem 7.8 the problem $\mathbf{TH}_{\mathrm{L}}$ is $\mathbf{\Pi}_1^{\mathrm{p}}$-complete. It follows that $\mathbf{STAT}_{\mathrm{L}} \in \mathbf{\Delta}_2^{\mathrm{p}}$, since $\mathbf{STAT}_{\mathrm{L}}$ is deterministically Turing-reducible to $\mathbf{TH}_{\mathrm{L}}$ in polynomial time. By Proposition 7.19, the problem $\mathbf{STAT}_{\mathrm{L}}$ is $\mathbf{\Delta}_2^{\mathrm{p}}$-hard implying that $\mathbf{STAT}_{\mathrm{L}}$ is $\mathbf{\Delta}_2^{\mathrm{p}}$-complete. □

Let us then briefly discuss the complexity of stationary semantics under the credulous approach. Let $\mathbf{STAT'}_{\mathrm{L}}$ be the corresponding decision problem, i.e. $\langle X, T, \phi\rangle \in \mathbf{STAT'}_{\mathrm{L}}$ if and only if $\phi \in \mathbf{N}_X(T, N)$ for some $\mathbf{N}_X[T]$-stationary set of non-theorems N. To reduce $\mathbf{STAT'}_{\mathrm{L}}$ in $\mathbf{TH}_{\mathrm{L}}$, the machine CoStab_X in Figure 6 has to be modified to accept its input $\langle X, T, \phi\rangle$ if and only if (i) the chosen N is contained in N' returned by $\mathsf{NT}_X(T, N, L)$, (ii) $\mathsf{NT}_X(T, N', L)$ returns N and $\mathsf{TH}_X(T, N, \phi)$ returns True. Using such a machine, the membership of $\mathbf{STAT'}_{\mathrm{L}}$ in $\mathbf{\Sigma}_2^{\mathrm{p}}$ can be established. Gottlob [38] has established $\mathbf{\Sigma}_2^{\mathrm{p}}$-hardness of $\mathbf{STAT'}_{\mathrm{DL}}$. Then $\mathbf{STAT'}_{\mathrm{AEL}}$ is also $\mathbf{\Sigma}_2^{\mathrm{p}}$-hard for the bimodal autoepistemic logic proposed in this work, since $\mathbf{STAT'}_{\mathrm{DL}} \leq_{\mathrm{m}}^{\mathrm{p}} \mathbf{STAT'}_{\mathrm{AEL}}$ by Theorem 3.32 as well as Corollaries 6.15 and 6.16.

The reader is reminded of the obvious consequence of Theorems 7.18 and 7.20:

Corollary 7.21 *The problems* $\mathbf{STAB}_{\mathrm{L}}$ *and* $\mathbf{STAT}_{\mathrm{L}}$ *are decidable for circumscription, default logic and autoepistemic logic.*

By Theorems 7.18 and 7.20 we have classified the time complexity of the major decision procedures of bimodal autoepistemic logic proposed in this work. Gottlob [37] has also classified the problem $\mathbf{STAB}_{\mathrm{AEL}}$ for conventional (unimodal) autoepistemic logic (under Marek and Truszczyński's iterative expansions). By his results the problem $\mathbf{STAB}_{\mathrm{AEL}}$ is $\mathbf{\Pi}_2^{\mathrm{p}}$-complete. This indicates that the complexity of autoepistemic reasoning is not affected under stable semantics even though a separate modality for disbeliefs is introduced. Thus default logic and bimodal autoepistemic logic (under iterative expansions) are of equal expressive power if measured by computational complexity. This view is strengthened by Corollaries 6.23 and 6.24 which establish embeddings between the two logics in a *modular* way. In the bimodal setting, stationary semantics becomes non-trivial and the time complexity of the problem $\mathbf{STAT}_{\mathrm{AEL}}$ is strictly lower than that of $\mathbf{STAB}_{\mathrm{AEL}}$. This suggests that further forms of tractable autoepistemic reasoning become available (in addition to stratified autoepistemic reasoning [84]) if the time complexity of the underlying monotonic reasoning is reduced. This can be achieved by restricting autoepistemic theories syntactically to Horn clauses. This technique is used by Niemelä and Rintanen [84] for unimodal autoepistemic logic.

8 Cautious Semantics

In this section, we investigate the *gap* between stationary semantics and stable semantics which was established by Theorem 4.24 giving the relationship $\mathrm{STAT}_{\mathbb{S}} \sqsubseteq \mathrm{STAB}_{\mathbb{S}}$ in general, and by our examples of non-monotonic systems $\mathbb{S}$ for which the proper inclusion $\mathrm{STAT}_{\mathbb{S}}(T) \subset \mathrm{STAB}_{\mathbb{S}}(T)$ resulted. This suggests that one could find a reasonable semantics $\mathrm{SEM}_{\mathbb{S}}$ in this gap, i.e. a semantics which *refines* stationary semantics but is still *sound* with respect to stable semantics. These requirements can be expressed more formally as $\mathrm{STAT}_{\mathbb{S}} \sqsubseteq \mathrm{SEM}_{\mathbb{S}} \sqsubseteq \mathrm{STAB}_{\mathbb{S}}$.[30] Of course, we are assuming that the semantics $\mathrm{SEM}_{\mathbb{S}}$ differs from stationary semantics and stable semantics significantly.

The above requirements for the semantics $\mathrm{SEM}_{\mathbb{S}}$ are still relatively weak, so we have to come up with other criteria to judge what kind of a semantics is actually searched for. This is where the abstract properties of semantics introduced in Sections 5 and 7 become useful. Let us discuss the role of some properties. First of all, the computational time complexity of a semantics is tightly connected to its *expressive power*. The results of Section 7 indicate that the decision problems of stationary semantics and stable semantics reside on different levels of the polynomial time hierarchy. This means that stationary semantics and stable semantics differ considerably in expressive power. Therefore, if a semantics provides a real refinement with respect to stationary semantics, the decision problem of the semantics $\mathrm{SEM}_{\mathbb{S}}$ should be of higher complexity than that of stationary semantics, but it does not seem reasonable to exceed stable semantics in this respect. Secondly, the cumulativity property can also make a difference. It was shown in Section 5.1 that stationary semantics is cumulative in general while stable semantics is not. More precisely, the two semantics are both cumulatively closed, but stable semantics is not cumulatively monotonic. Therefore, the requirement CMON differentiates stationary semantics from stable semantics. For the third, there is a substantial number of examples from various application domains that can be used as *benchmark problems*. Obvious questions are then whether specific problems of the domain can be solved by the semantics $\mathrm{SEM}_{\mathbb{S}}$ under consideration or not, and what conclusions can be drawn from a given example. This is necessary to judge the applicability of a semantics in a particular application domain.

In this section, we apply the methodology set up in the preceding sections to develop and analyze a novel semantics – keeping the above requirements and observations in mind. The semantics is established and analyzed as follows. In Section 8.1, we present a refinement technique for stationary semantics

[30] Note that there are non-trivial systems $\mathbb{S}$ for which $\mathrm{STAT}_{\mathbb{S}}(T) = \mathrm{STAB}_{\mathbb{S}}(T)$ even for all $T \subseteq \mathcal{L}^+(\mathbb{S})$. Therefore, we cannot simply require that $\mathrm{STAT}_{\mathbb{S}}(T) \subset \mathrm{SEM}_{\mathbb{S}}(T)$ and $\mathrm{SEM}_{\mathbb{S}}(T) \subset \mathrm{STAB}_{\mathbb{S}}(T)$ for all $T \subseteq \mathcal{L}^+(\mathbb{S})$.

called *biasing*. Roughly speaking, the idea is to distort the equilibrium of the least stationary set of non-theorems by fixing an undetermined sentence as a non-theorem. It is shown that the least fixed point may consequently grow. In Section 8.2, larger and larger biases are constructed using this biasing technique systematically. This leads to a groundedness condition for biases which gives rise to the notion of *cautious semantics*. The properties of cautious semantics are considered in Section 8.3. It is shown that cautious semantics is indeed a semantics in the gap between the stationary and stable semantics and that it is preserved under the translations presented in Section 3.4. The cumulativity property and complexity issues are also addressed. Finally, in Section 8.4, we compare cautious semantics with other related approaches presented in the literature [3, 12, 19, 32, 44, 88, 115, 118].

8.1 Refining Stationary Semantics

We showed already in Theorem 4.24 that stable semantics is stronger than stationary semantics. For the non-monotonic system of Example 4.7, the two semantics coincide. It is demonstrated in Example 8.1 how this inclusion can be strict. The atomic propositions have been chosen so that the lattice in Figure 3 depicts the sets of non-theorems for this example as well. This example is an autoepistemic version of the floating conclusions example presented in the context of logic programs, for instance.

Example 8.1 Consider a set of autoepistemic sentences

$$\Sigma = \{\mathbf{D}\mathrm{q} \rightarrow \mathrm{p}, \mathbf{D}\mathrm{p} \rightarrow \mathrm{q}, \mathbf{B}\mathrm{p} \rightarrow \mathrm{r}, \mathbf{B}\mathrm{q} \rightarrow \mathrm{r}\}$$

and the subsystem $\mathbb{S} = \boldsymbol{\mathcal{A}}_{\Sigma}^{\Sigma\cup\{\mathbf{B}\mathrm{r}\}}$ which has the language $\langle\{\mathrm{p},\mathrm{q},\mathrm{r}\},\{\mathrm{p},\mathrm{q}\}\rangle$.

There are two $\mathbb{S}$-stable sets of non-theorems, namely $N_1 = \{\mathrm{p}\}$ and $N_2 = \{\mathrm{q}\}$. It follows that $\mathrm{r} \in \mathbb{S}(N_i)$ for $i \in \{1,2\}$ so that $\mathrm{r} \in \mathrm{STAB}_{\mathbb{S}}(\emptyset)$. On the other hand, the least $\mathbb{S}$-stationary set of non-theorems is computed as follows. Let $N_0 = \emptyset$. Now $\mathrm{p} \notin \mathbb{S}(N_0)$ and $\mathrm{q} \notin \mathbb{S}(N_0)$ implying that the set of potential non-theorems $\overline{\mathbb{S}}(N_0)$ is $\{\mathrm{p},\mathrm{q}\}$. However, both p and q are included in $\mathbb{S}(\overline{\mathbb{S}}(N_0)) = \mathrm{Cn}_{\mathbf{B}}(\Sigma \cup \{\mathbf{D}\mathrm{p}, \mathbf{D}\mathrm{q}\}) \cap \{\mathrm{p},\mathrm{q},\mathrm{r}\} = \{\mathrm{p},\mathrm{q},\mathrm{r}\}$ so that $\overline{\mathbb{S}}^2(N_0) = \emptyset = N_0$. Hence N_0 determines $\mathrm{STAT}_{\mathbb{S}}(\emptyset)$ by Theorem 4.28 implying that $\mathrm{STAT}_{\mathbb{S}}(\emptyset) = \mathbb{S}(N_0) = \emptyset$ and $\mathrm{STAT}_{\mathbb{S}}(\emptyset) \subset \mathrm{STAB}_{\mathbb{S}}(\emptyset)$. □

A careful analysis of this example reveals the reason why the stationary semantics $\mathrm{STAT}_{\mathbb{S}}(\emptyset)$ is so weak. The set of potential non-theorems $\overline{\mathbb{S}}(N_0) = \{\mathrm{p},\mathrm{q}\}$ is simply unjustified: if either one of p and q is taken as a non-theorem, the other is necessarily a theorem by the implications $\mathbf{D}\mathrm{p} \rightarrow \mathrm{q}$ and $\mathbf{D}\mathrm{q} \rightarrow \mathrm{p}$ in Σ. For this reason, we consider the set of potential non-theorems $\overline{\mathbb{S}}(N_0)$ to be *weakly grounded* in the premises Σ. This observation suggests a technique of avoiding such weakly grounded set of potential non-theorems. What we do is that we pick one of our potential non-theorems, e.g. $\mathrm{p} \in \overline{\mathcal{S}}(N_0)$, and fix it as a *non-theorem*. Then we reconstruct the least stationary set of non-theorems using the non-monotonic system $\mathcal{S}[F^-]$ with the set of non-theorems $F^- = \{\mathrm{p}\}$ fixed. A desirable effect results with Example 8.1 which is taken into reconsideration in Example 8.2 below. The set of potential non-theorems $\{\mathrm{p}, \mathrm{q}\}$ is no longer produced by the iteration of $\overline{\mathcal{S}[F^-]}^2$. Instead, the sets of potential non-theorems $\{\mathrm{p}\}$ and $\{\mathrm{q}\}$ are obtained. In light of our preceding discussion, we take these to be strongly grounded in the premises Σ.

Example 8.2 For the non-monotonic system $\mathcal{S}$ of Example 8.1, consider the computation of $\mathrm{lfp}(\overline{\mathcal{S}[F_1^-]}^2, \emptyset)$ for the set of non-theorems $F_1^- = \{\mathrm{p}\}$. So let $N_0 = \emptyset$. Now $\mathcal{S}[F_1^-](N_0) = \{\mathrm{q}, \mathrm{r}\}$ so that $\overline{\mathcal{S}[F_1^-]}(N_0) = \{\mathrm{p}\}$. In a similar way, it is obtained that $N_1 = \overline{\mathcal{S}[F_1^-]}(\{\mathrm{p}\}) = \{\mathrm{p}\}$. This indicates that N_1 is already the least fixed point of the operator $\overline{\mathcal{S}[F_1^-]}^2$. It holds also that $\mathrm{lfp}(\overline{\mathcal{S}[F_2^-]}^2, \emptyset) = \{\mathrm{q}\}$ for the set of non-theorems $F_2^- = \{\mathrm{q}\}$ by the obvious symmetry present in our example. □

By fixing a sentence ϕ as a non-theorem, we distort the equilibrium obtained by the operator $\overline{\mathcal{S}}^2$ and a larger set of non-theorems results as the least fixed point of $\overline{\mathcal{S}[F^-]}^2$ where $F^- = \{\phi\}$. In fact, we obtained all possible $\mathcal{S}$-stable sets of non-theorems in Example 8.2. Upon these, we can conclude that the sentence r is a theorem, so that the gap demonstrated in Example 8.1 is no longer realized. In Definition 8.3, we single out sentences of $\mathcal{L}^-(\mathcal{S})$ that seem to be plausible *biases* for $\mathcal{S}$. These are exactly sentences among the potential non-theorems $\overline{\mathcal{S}}(N)$ for the least $\mathcal{S}$-stationary set of non-theorems N.

Definition 8.3 *A sentence $\phi \in \mathcal{L}^-(\mathcal{S})$ is a bias for a non-monotonic system $\mathcal{S}$ if and only if $\phi \in \overline{\mathcal{S}}(N)$ where $N = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$.*

A bias $\phi \in \mathcal{L}^-(\mathcal{S})$ for $\mathcal{S}$ is acceptable for two reasons. First of all, it is shown in Proposition 8.5 that the least $\mathcal{S}[\{\phi\}]$-stationary set of non-theorems extends the least $\mathcal{S}$-stationary set of non-theorems. We are especially looking for biases ϕ for which this inclusion becomes proper, and a proper refinement of the least $\mathcal{S}$-stationary set of non-theorems results. On the other hand, the resulting set of non-theorems is $\mathcal{S}$-consistent as established in Proposition 8.6. To prove these propositions, we need the straightforward result of Lemma 8.4.

Lemma 8.4 *If $F_1^+ \subseteq F_2^+ \subseteq \mathcal{L}^+(\mathcal{S})$ and $F_1^- \subseteq F_2^- \subseteq \mathcal{L}^-(\mathcal{S})$, then it holds for all $N \subseteq \mathcal{L}^-(\mathcal{S})$ that $\overline{\mathcal{S}[F_2^+; F_2^-]}(N) \subseteq \overline{\mathcal{S}[F_1^+; F_1^-]}(N)$.*

Proof. Now $\mathcal{S}(F_1^+, F_1^- \cup N) \subseteq \mathcal{S}(F_2^+, F_2^- \cup N)$ by MON. The rest follows by the definitions of fixing and the operators $\overline{\mathcal{S}[F_1^+; F_1^-]}$ and $\overline{\mathcal{S}[F_2^+; F_2^-]}$. $\square$

Proposition 8.5 *If $N = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$ and $F^- \subseteq \overline{\mathcal{S}}(N)$, then the set of non-theorems N is a subset of $N' = \mathrm{lfp}(\overline{\mathcal{S}[F^-]}^2, \emptyset)$.*

Proof. Define $N_\alpha = \overline{\mathcal{S}}^2 \uparrow^\alpha (\emptyset)$ for all ordinals α. By Theorem 4.28 and Proposition 4.25, we know that $N_\alpha \subseteq N$ and $\overline{\mathcal{S}}(N) \subseteq \overline{\mathcal{S}}(N_\alpha)$ for all ordinals α. Since $F^- \subseteq \overline{\mathcal{S}}(N)$ by the requirements given, it follows that $F^- \subseteq \overline{\mathcal{S}}(N_\alpha)$ for all ordinals α. Let us then prove by transfinite induction on α that N_α is a subset of N'. The base case $\alpha = 0$ holds, since $N_0 = \emptyset$. Induction step follows. Let α be any successor ordinal. By the inductive hypothesis, the set $N_{\alpha-1}$ is a subset of N'. It follows by Lemma 8.4 and Proposition 4.25 that $\overline{\mathcal{S}[F^-]}(N') \subseteq \overline{\mathcal{S}}(N') \subseteq \overline{\mathcal{S}}(N_{\alpha-1})$. Thus $\overline{\mathcal{S}[F^-]}(\overline{\mathcal{S}}(N_{\alpha-1})) \subseteq \overline{\mathcal{S}[F^-]}^2(N') = N'$ follows by Proposition 4.25. As we noted earlier, the set F^- is contained in $\overline{\mathcal{S}}(N_{\alpha-1})$. It follows that $\overline{\mathcal{S}[F^-]}(\overline{\mathcal{S}}(N_{\alpha-1})) = \overline{\mathcal{S}}(\overline{\mathcal{S}}(N_{\alpha-1})) = N_\alpha$. Thus we arrive at the desired conclusion $N_\alpha \subseteq N'$. Finally, consider any limit ordinal α. Since $N_\beta \subseteq N'$ holds for all $\beta < \alpha$ by the inductive hypothesis, it is clear that the limit $N_\alpha = \bigcup_{\beta<\alpha} N_\beta$ is contained in N'. $\square$

Proposition 8.6 *If $\phi \in \mathcal{L}^-(\mathcal{S})$ is a bias for $\mathcal{S}$, then the set of non-theorems $N = \mathrm{lfp}(\overline{\mathcal{S}[\{\phi\}]}^2, \emptyset)$ is $\mathcal{S}$-consistent.*

Proof. It holds for $F^- = \{\phi\}$ that $N \subseteq \overline{\mathcal{S}[F^-]}(N)$, since N is $\mathcal{S}[F^-]$-stationary. It follows that $\overline{\mathcal{S}[F^-]}(N) \subseteq \overline{\mathcal{S}}(N)$ by Lemma 8.4. Thus also $N \subseteq \overline{\mathcal{S}}(N)$ implying that N is $\mathcal{S}$-consistent by Proposition 4.9. $\square$

8.2 Constructing Biases Systematically

So far we have considered using only a single non-theorem $\phi \in \mathcal{L}^-(\mathcal{S})$ for biasing purposes. It is justifiable to ask whether we may use sets of non-theorems $B \subseteq \mathcal{L}^-(\mathcal{S})$ for biasing.[31] The answer to this question is positive, but we cannot accept arbitrary sets of non-theorems B as biases. Let us present a further bias candidate $B_{12} = \{\mathrm{p}, \mathrm{q}\}$ for Example 8.2 to demonstrate that this is the case. The resulting fixed point $\mathrm{lfp}(\overline{\mathcal{S}[B_{12}]}^2, \emptyset) = \emptyset$ coincides with $\mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$. Thus accepting B_{12} as one of the biases would cause the resulting skeptical semantics to collapse to stationary semantics, since the set of theorems $\mathcal{S}(N)$ induced by $N = \mathrm{lfp}(\overline{\mathcal{S}[B_{12}]}^2, \emptyset)$ would then be included in the intersection. An obvious idea is to use single non-theorems for biasing

[31] From now on, we use B, B', B_1, B_2, etc. to denote sets of non-theorems used as biases.

repeatedly to obtain larger biases and in this way refine the least $\mathcal{S}$-stationary set of non-theorems even further. In this approach, biasing takes place in some particular order of sentences. This leads to a condition given in Definition 8.7 where a biasing enumeration is identified.

Definition 8.7 *Let $\mathcal{S}$ be a non-monotonic system. A set $B \subseteq \mathcal{L}^{-}(\mathcal{S})$ is an $\mathcal{S}$-bias if and only if there is an enumeration ε of B such that ε_α is a bias for $\mathcal{S}[\{\varepsilon_{0\ldots(\alpha-1)}\}]$ for all successor ordinals α in the range $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$.*

Recall that in Definition 8.7 $\{\varepsilon_{0\ldots(\alpha-1)}\}$ denotes the set of non-theorems $\{\varepsilon_\beta \mid 0 \leq \beta \leq \alpha - 1$ and β is a successor ordinal $\}$. The definition of $\mathcal{S}$-biases is constructive, since the enumeration giving the biasing ordering is given explicitly. If an enumeration ε of B satisfies the condition of Definition 8.7 (indicating that B is a bias) we say that ε satisfies the *bias condition* for $\mathcal{S}$. This condition can be also interpreted as a *groundedness condition* on sets of potential non-theorems B. Let us look back to Example 8.2 to demonstrate the effects of Definition 8.7. The $\mathcal{S}$-biases $B_1 = \{\mathrm{p}\}$ and $B_2 = \{\mathrm{q}\}$ led to proper refinements of the least $\mathcal{S}$-stationary set of non-theorems. As we noted above, this is not the case for the union $B_{12} = \{\mathrm{p}, \mathrm{q}\}$. Fortunately, this is not an $\mathcal{S}$-bias at all, since the condition of Definition 8.7 is not satisfied. This becomes clear having noticed that q is not a bias for $\mathcal{S}[B_1]$ and p is not a bias for $\mathcal{S}[B_2]$. Therefore, it is unjustifiable to take p and q as non-theorems *simultaneously* given the system $\mathcal{S}$ and in this sense the candidate B_{12} is *weakly grounded*. This is how $\mathcal{S}$-biases introduced in Definition 8.7 avoid the unwanted fixed point $\mathrm{lfp}(\overline{\mathcal{S}[B_{12}]}^2, \emptyset) = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset) = \emptyset$ and thus a refinement with respect to stationary semantics is not preempted. Note also that the set B_{12} is also used as a set of potential non-theorems when the least $\mathcal{S}$-stationary set of non-theorems is constructed in Examples 8.1 and 8.2.

Before presenting further examples, let us now address two aspects of $\mathcal{S}$-biases. In Lemma 8.8, we provide a condition on which larger $\mathcal{S}$-biases can be constructed simply by taking disjoint unions. In Proposition 8.9, we give an inductive formulation of $\mathcal{S}$-biases where the enumeration ε satisfying the bias condition is intentionally left implicit. Notice also that only the items (i) and (ii) are required if $\mathcal{L}^{-}(\mathcal{S})$ is finite, but we consider also limits of increasing sequences of biases for the sake of completeness of our treatment of biases.

Lemma 8.8 *If $B \subseteq \mathcal{L}^{-}(\mathcal{S})$ is an $\mathcal{S}$-bias and $B' \subseteq \mathcal{L}^{-}(\mathcal{S}) - B$ is an $\mathcal{S}[B]$-bias, then $B \cup B'$ is an $\mathcal{S}$-bias.*

Proof. Since B is an $\mathcal{S}$-bias, there is an enumeration of ε of B that fulfills the bias condition for $\mathcal{S}$. Similarly, there is an enumeration ε' of B' satisfying the bias condition for $\mathcal{S}[B']$. Since B and B' are disjoint, we can form an enumeration of $B \cup B'$ as a simple concatenation of ε and ε', denoted by $\varepsilon \cdot \varepsilon'$. It is easy to see that also the enumeration $\varepsilon \cdot \varepsilon'$ satisfies the bias condition for $\mathbb{S}$ due to the locality of the condition. Thus the union $B \cup B'$ is an $\mathbb{S}$-bias. $\square$

Proposition 8.9 *A set of non-theorems $B \subseteq \mathcal{L}^-(\mathbb{S})$ is an $\mathbb{S}$-bias if and only if (i) $B = \emptyset$ or (ii) $B = B' \cup \{\phi\}$ where B' is an $\mathbb{S}$-bias and ϕ is a bias for $\mathbb{S}[B']$ or (iii) B is the limit $\bigcup_{\beta<\alpha} B_\beta$ for an increasing sequence $B_0, B_1, \ldots, B_\beta, \ldots$ of $\mathbb{S}$-biases obtained by step (ii).*

Proof. ($\Leftarrow$) Let us show that (i-iii) imply that B is an $\mathbb{S}$-bias. (i) It is clear that $B = \emptyset$ is an $\mathbb{S}$-bias, since $\varepsilon = \lambda$ satisfies trivially the bias condition. (ii) Then assume B' is an $\mathbb{S}$-bias and ϕ is a bias for $\mathbb{S}[B']$. So there is an enumeration ε' of B' satisfying the bias condition. Let us extend ε' to ε by adding ϕ at the end of ε'. Thus ε is an enumeration of $B = B' \cup \{\phi\}$ and ε satisfies also the bias condition, especially for the successor ordinal $\mathrm{Ind}(\varepsilon)$ which gives the position of ϕ in ε. Thus B is an $\mathbb{S}$-bias. (iii) Then assume that B is the limit $\bigcup_{\beta<\alpha} B_\beta$ for an increasing sequence $B_0, B_1, \ldots, B_\beta, \ldots$ of $\mathbb{S}$-biases. Since B_0 is an $\mathbb{S}$-bias there is an enumeration ε^0 of B_0 that satisfies the bias condition. Since each B_β with $\beta > 0$ is obtained by adding a bias ϕ for $\mathbb{S}[B_{\beta-1}]$ in $B_{\beta-1}$, an enumeration ε^β satisfying the bias condition is obtained by adding ϕ at the end of $\varepsilon^{\beta-1}$. Let us define ε as the limit of ε^β satisfying $\beta < \alpha$. It is clear that the limit enumeration ε satisfies the bias condition for $\mathbb{S}$, because each enumeration ε^β does. It follows that the limit $B = \bigcup_{\beta<\alpha} B_\beta$ is an $\mathbb{S}$-bias, since ε is an enumeration of B.

($\Rightarrow$) Let us then use transfinite induction on $\mathrm{Ind}(\varepsilon)$ to show that at least one of (i-iii) holds for the set B enumerated by ε. In the base case $\mathrm{Ind}(\varepsilon) = 0$ implying that $\varepsilon = \lambda$ and $B = \emptyset$. Thus (i) holds. Induction step follows. Let $\mathrm{Ind}(\varepsilon)$ be a successor ordinal, $\phi = \varepsilon_{\mathrm{Ind}(\varepsilon)}$ be the last sentence of ε and ε' the enumeration ε without ϕ. By Definition 8.7, the set $B' = B - \{\phi\}$ is an $\mathbb{S}$-bias, since ε' is an enumeration of B' with the required property. Since $\mathrm{Ind}(\varepsilon') < \mathrm{Ind}(\varepsilon)$, we know that at least one of (i-iii) holds for B'. In each case, B' is an $\mathbb{S}$-bias by ($\Leftarrow$). Since ϕ is the last sentence of ε, it follows that ϕ is a bias for $\mathbb{S}[\{\varepsilon'\}] = \mathbb{S}[B']$ by Definition 8.7. Thus (ii) holds. Finally, assume that $\mathrm{Ind}(\varepsilon)$ is a limit ordinal so that $B = \bigcup_{\alpha \leq \beta < \mathrm{Ind}(\varepsilon)} B_\beta$ where $B_\beta = \{\varepsilon_{0\ldots\beta}\}$ and α is the greatest limit ordinal such that $\alpha < \mathrm{Ind}(\varepsilon)$. It is clear that every β in the range $\alpha < \beta < \mathrm{Ind}(\varepsilon)$ is a successor ordinal. Moreover, each B_β is an $\mathbb{S}$-bias obtained from $B_{\beta-1}$ by adding a bias ε_β for $\mathbb{S}[B_{\beta-1}]$ in $B_{\beta-1}$. Thus (iii) holds for the sequence $B_\alpha, B_{\alpha+1}, \ldots, B_\beta, \ldots$. $\square$

Having now refined the conditions on which a set of non-theorems $B \subseteq \mathcal{L}^-(\mathbb{S})$ is an $\mathbb{S}$-bias, let us present a couple of examples where subsequent biasing is demonstrated. In Example 8.10, it is illustrated that biasing with a single sentence is insufficient. The transfinite case is considered in Example 8.11.

Example 8.10 Consider an autoepistemic theory

$$\Sigma = \{\mathbf{D}\mathrm{q}_1 \wedge \mathbf{D}\mathrm{q}_2 \rightarrow \mathrm{p}, \mathbf{D}\mathrm{p} \rightarrow \mathrm{q}_1, \mathbf{D}\mathrm{p} \rightarrow \mathrm{q}_2\}$$

and the subsystem $\mathcal{S} = \boldsymbol{\mathcal{A}}^{\Sigma}_{\Sigma}$ having the finite language $\langle\{\mathrm{p}, \mathrm{q}_1, \mathrm{q}_2\}, \{\mathrm{p}, \mathrm{q}_1, \mathrm{q}_2\}\rangle$. It is obvious that the least $\mathcal{S}$-stationary set of non-theorems is $\emptyset$. Thus $B_0 = \{\mathrm{p}\}$, $B_1 = \{\mathrm{q}_1\}$ and $B_2 = \{\mathrm{q}_2\}$ are the possible singleton biases. The $\mathcal{S}$-bias B_0 is successful in the sense that $\mathrm{lfp}(\overline{\mathcal{S}[B_0]}^2, \emptyset) = \{\mathrm{p}\}$ and a proper refinement results. However, for both B_1 and B_2 the least $\mathcal{S}[B_i]$-stationary set of non-theorems is still $\emptyset$. Further biasing is necessary. Note that $B_{12} = \{\mathrm{q}_1, \mathrm{q}_2\}$ is an $\mathcal{S}$-bias, since q_2 is a bias for $\mathcal{S}[B_1]$, or alternatively q_1 is a bias for $\mathcal{S}[B_2]$. This is sufficient for a proper refinement: $\mathrm{lfp}(\overline{\mathcal{S}[B_{12}]}^2, \emptyset) = \{\mathrm{q}_1, \mathrm{q}_2\}$. □

Example 8.11 Consider an infinite autoepistemic theory

$$\Sigma = \{\mathbf{D}\mathrm{q}_0 \rightarrow \mathrm{p}_0, \mathbf{D}\mathrm{p}_0 \rightarrow \mathrm{q}_0, \mathbf{D}\mathrm{q}_1 \rightarrow \mathrm{p}_1, \mathbf{D}\mathrm{q}_1 \rightarrow \mathrm{p}_1, \ldots\}$$

and the subsystem $\mathcal{S} = \boldsymbol{\mathcal{A}}^{\Sigma}_{\Sigma}$ having the language $\langle A, A\rangle$ where A is the set of atoms $\{\mathrm{p}_0, \mathrm{q}_0, \mathrm{p}_1, \mathrm{q}_1, \ldots\}$. By our experience on Example 8.2 it is easy to see that $\mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset) = \emptyset$. Let us then show by induction on i that $B_i = \{\mathrm{p}_0, \ldots, \mathrm{p}_i\}$ is an $\mathcal{S}$-bias and $\mathrm{lfp}(\overline{\mathcal{S}[B_i]}^2, \emptyset) = B_i$. The base case holds as $B_0 = \emptyset$ and $\mathrm{lfp}(\overline{\mathcal{S}[B_0]}^2, \emptyset) = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset) = \emptyset$. Induction step follows. By the inductive hypothesis, the set B_{i-1} is an $\mathcal{S}$-bias and the fixed point $\mathrm{lfp}(\overline{\mathcal{S}[B_{i-1}]}^2, \emptyset) = B_{i-1}$. It follows that p_i is a bias for $\mathcal{S}[B_{i-1}]$ so that B_i is an $\mathcal{S}$-bias. It is also clear that $\mathrm{lfp}(\overline{\mathcal{S}[B_i]}^2, \emptyset)$ is B_i. By item (iii) in Proposition 8.9, the limit $B_\omega = \{\mathrm{p}_0, \mathrm{p}_1, \ldots\}$ is an $\mathcal{S}$-bias. Moreover, the fixed point $\mathrm{lfp}(\mathcal{S}[B_\omega], \emptyset) = B_\omega$ implying that $\mathrm{STAT}_{\mathcal{S}[B_\omega]}(\emptyset) = \{\mathrm{q}_0, \mathrm{q}_1, \ldots\}$. □

Our explanation for the weakness of stationary semantics is that it is often determined by a set of potential non-theorems which is not an $\mathcal{S}$-bias in the sense of Definition 8.7. On the other hand, restricting to the biases satisfying only the condition of Definition 8.7 is not sufficient. This is because $B = \emptyset$ is always an $\mathcal{S}$-bias, for instance. As to be expected, this bias reproduces the least $\mathcal{S}$-stationary set of non-theorems so that we have to force biases to be larger somehow. This effect is achieved by introducing the following notion of finishedness for $\mathcal{S}$-biases.

Definition 8.12 *An $\mathcal{S}$-bias $B \subseteq \mathcal{L}^-(\mathcal{S})$ is finished if and only if the set of non-theorems $B \cup N$ where $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$ is $\mathcal{S}$-complete.*

In an optimal setting, the non-theorems of an $\mathcal{S}$-bias B are *realized* as non-theorems in the fixed point $\mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$. This was the case, e.g., in Example 8.11. However, to cope with non-monotonic systems $\mathcal{S}$ without $\mathcal{S}$-stable sets of non-theorems, we cannot take this as a general requirement. In Example 8.13, we encounter an $\mathcal{S}$-bias B whose sentences are not realized as non-theorems.

Example 8.13 Consider a set of defaults

$$D = \{\frac{:\neg \mathrm{p}}{\mathrm{p}}\}$$

and the subsystem $\mathcal{S} = \mathfrak{D}_D^{\neg \mathrm{J}(D)}$ having the language $\langle \mathcal{L}, \{\neg\neg \mathrm{p}\}\rangle$. The least $\mathcal{S}$-stationary set of non-theorems is empty. So is the least $\mathcal{S}[B]$-stationary set of non-theorems obtained for the only possible non-empty $\mathcal{S}$-bias $B = \{\neg\neg \mathrm{p}\}$. Note that B is finished, since $B \cup \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset) = B$ is $\mathcal{S}$-complete. □

The notion of a finished $\mathcal{S}$-bias has an intuitive computational interpretation if we take a bottom-up view of Definition 8.7 as well as Proposition 8.9. We are effectively requiring that an $\mathcal{S}$-bias B can be constructed by a step-by-step process where our current set of non-theorems $\mathrm{lfp}(\overline{\mathcal{S}[B']}^2, \emptyset)$ is extended by a potential non-theorem ϕ. In a way, this is done by force, since we use fixing. As indicated by Example 8.13 there is no guarantee whether ϕ is eventually realized as a non-theorem. The notion of finishedness provides a termination condition for this biasing process: when B is finished there is no sentence ϕ in the set of potential non-theorems $\overline{\mathcal{S}[B]}(N)$ for the fixed point $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$ that would be useful in extending B, since $\overline{\mathcal{S}[B]}(N) \subseteq B \cup N$. These intuitions provide the basis for proving the result of Theorem 8.14 that covers all non-monotonic systems $\mathcal{S}$. The proof of the theorem embodies an important principle that leads to an systematic algorithm (see the function $\mathsf{FindBias}_X$ in Figure 14) to compute an $\mathcal{S}$-bias for subsystems $\mathcal{S}$ associated with circumscription, default logic and autoepistemic logic.

Theorem 8.14 *There is at least one finished $\mathcal{S}$-bias for every $\mathcal{S}$.*

Proof. Let ε be any enumeration of $\mathcal{L}^-(\mathcal{S})$. In the following, we define a sequence B_α of $\mathcal{S}$-biases and resulting fixed points $N_\alpha = \mathrm{lfp}(\overline{\mathcal{S}[B_\alpha]}^2, \emptyset)$ for all α in the range $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$. We use transfinite induction. In the base case $\alpha = 0$, we define $B_0 = \emptyset$. Then consider a successor ordinals α. If $\varepsilon_\alpha \in \overline{\mathcal{S}[B_{\alpha-1}]}(N_{\alpha-1})$, then $B_\alpha = B_{\alpha-1} \cup \{\varepsilon_\alpha\}$ and $B_\alpha = B_{\alpha-1}$ otherwise. For limit ordinals α, the set B_α is defined as the limit $\bigcup_{\beta<\alpha} B_\beta$. Note that this construction is an analog of the one given in Definition 5.22, but the stationary construction is used to decide the status of ε_β instead of the system $\mathcal{S}$.

The set B obtained by this construction is an $\mathcal{S}$-bias, since a sentence ε_β is inserted into B_α only if ε_β is a potential non-theorem given the bias $B_{\alpha-1}$. This conforms to the requirement of Definition 8.7 and the order in which the sentences are inserted to B gives us the required enumeration. Moreover, the bias B is definitely finished, because every sentence of $\mathcal{L}^-(\mathcal{S})$ is considered by the construction. The initial enumeration ε determines which one of the biases we get in the case there are several finished $\mathcal{S}$-biases. $\square$

Before presenting the definition of a semantics for $\mathcal{S}$ based on finished $\mathcal{S}$-biases, we present Proposition 8.15 which provides an interesting connection to $\mathcal{S}[B]$-stable sets of non-theorems given that B is finished.

Proposition 8.15 *If an $\mathcal{S}$-bias $B \subseteq \mathcal{L}^-(\mathcal{S})$ is finished, then the set of non-theorems $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$ is $\mathcal{S}[B]$-stable.*

Proof. The set of non-theorems N is $\mathcal{S}[B]$-stationary that implies that N is $\mathcal{S}[B]$-consistent by definition. On the other hand, the finishedness of B implies that $B \cup N$ is $\mathcal{S}$-complete so that N is $\mathcal{S}[B]$-complete. $\square$

We are finally in position to present *cautious semantics*[32] $\mathrm{CAUT}_{\mathcal{S}}$ based on finished biases. It is important to note that the sets of theorems appearing in the definition is $\mathcal{S}[T](N)$ but not $\mathcal{S}[T;B](N)$. This is because the bias B may contain non-theorems that are not realized as non-theorems in the fixed point N, i.e. $B - N \neq \emptyset$. Let us look back to Example 8.13 to understand this detail. Recall that T was empty and the only finished $\mathcal{S}$-bias was $B = \{\neg\neg \mathrm{p}\}$ which gave rise to the least $\mathcal{S}[B]$-stationary set of non-theorems $N = \emptyset$ so that $B - N$ is non-empty. Let us then note that $\mathcal{S}(N) = \mathrm{Cn}(\emptyset)$ but $\mathcal{S}[B](N) = \mathrm{Cn}(\{\mathrm{p}\})$ contains also $\neg\neg\mathrm{p}$. Since we cannot decide the status of $\neg\neg\mathrm{p}$ as a non-theorem, we are not willing to accept p as a theorem either. This explains the choice made in our definition of the cautious semantics $\mathrm{CAUT}_{\mathcal{S}}$.

Definition 8.16 *The operator* $\mathrm{CAUT}_{\mathcal{S}}$ *for the cautious semantics of a non-monotonic system* $\mathcal{S}$ *is defined for all* $T \subseteq \mathcal{L}^+(\mathcal{S})$ *by* $\mathrm{CAUT}_{\mathcal{S}}(T) =$

$$\bigcap \{\mathcal{S}(T,N) \mid N = \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset) \text{ and } B \text{ is a finished } \mathcal{S}[T]\text{-bias.}\}$$

8.3 Characteristics of Cautious Semantics

Let us now concentrate on establishing some properties of cautious semantics $\mathrm{CAUT}_{\mathcal{S}}$ using the methods introduced in previous sections. In particular, we will study to which degree cautious semantics fulfills the requirements set up at the beginning of Section 8. The rest of Section 8.3 is organized as follows. Two properties of biases that are needed in the sequel are established in Section 8.3.1. It is then shown in Section 8.3.2 that cautious semantics $\mathrm{CAUT}_{\mathcal{S}}$ is indeed a semantics in the gap between stationary semantics and

[32]The name coincides with the semantics presented by the author elsewhere [44]. These semantics are related, but different to some extent.

stable semantics. The cumulativity property is addressed in Section 8.3.3. It is also established in Section 8.3.4 that the various translations introduced in Section 3.4 preserve cautious semantics.

8.3.1 Subsidiary Results on Biases

Proposition 8.5 implies that a single non-theorem used as a bias $\phi \in \mathcal{L}^-(\mathbb{S})$ may give rise to a refinement of the least $\mathbb{S}$-stationary set of non-theorems. By the following proposition, this result is generalized for arbitrary $\mathbb{S}$-biases $B \subseteq \mathcal{L}^-(\mathbb{S})$ satisfying the groundedness condition of Definition 8.7.

Proposition 8.17 *If $N = \mathrm{lfp}(\overline{\mathbb{S}}^2, \emptyset)$ and B is an $\mathbb{S}$-bias, then the set of non-theorems N is contained in $\mathrm{lfp}(\overline{\mathbb{S}[B]}^2, \emptyset)$ and $B \subseteq \overline{\mathbb{S}}(N)$.*

Proof. Let $B \subseteq \mathcal{L}^-(\mathbb{S})$ be an $\mathbb{S}$-bias. By Definition 8.7 there is an enumeration ε of B such that for each successor ordinal α the sentence ε_α is a bias for $\mathbb{S}[B_{\alpha-1}]$ where $B_{\alpha-1} = \{\varepsilon_{0\ldots(\alpha-1)}\}$. It us then shown by transfinite induction on α that $B_\alpha = \{\varepsilon_{0\ldots\alpha}\}$ is a subset of $\overline{\mathbb{S}}(N)$ and $N \subseteq N_\alpha = \mathrm{lfp}(\overline{\mathbb{S}[B_\alpha]}^2, \emptyset)$ for all α in the range $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$. This implies that N is contained in the final fixed point $N_{\mathrm{Ind}(\varepsilon)} = \mathrm{lfp}(\overline{\mathbb{S}[B]}^2, \emptyset)$ where $B = B_{\mathrm{Ind}(\varepsilon)}$ and $B \subseteq \overline{\mathbb{S}}(N)$.

The base case $\alpha = 0$ holds trivially, since $B_0 = \emptyset$ and $N = \mathrm{lfp}(\overline{\mathbb{S}[B_0]}^2, \emptyset)$. Induction step follows. Let α be a successor ordinal. It holds by the inductive hypothesis that $B_{\alpha-1} \subseteq \overline{\mathbb{S}}(N)$ and N is a subset of $N_{\alpha-1} = \mathrm{lfp}(\overline{\mathbb{S}[B_{\alpha-1}]}^2, \emptyset)$. Since ε_α is a bias for $\mathbb{S}[B_{\alpha-1}]$, it follows that $\varepsilon_\alpha \in \overline{\mathbb{S}[B_{\alpha-1}]}(N_{\alpha-1})$. Consequently, the sentence $\varepsilon_\alpha \in \overline{\mathbb{S}}(N)$, because $\overline{\mathbb{S}[B_{\alpha-1}]}(N_{\alpha-1}) \subseteq \overline{\mathbb{S}}(N_{\alpha-1}) \subseteq \overline{\mathbb{S}}(N)$ by Lemma 8.4 and Proposition 4.25. Thus $B_\alpha = B_{\alpha-1} \cup \{\varepsilon_\alpha\}$ is contained in $\overline{\mathbb{S}}(N)$ and $N \subseteq N_\alpha$ follows by by Proposition 8.5. Finally, consider a limit ordinal α. By the inductive hypothesis, it holds that $B_\beta \subseteq \overline{\mathbb{S}}(N)$ for all ordinals $\beta < \alpha$. Thus the limit $B_\alpha = \bigcup_{\beta<\alpha} B_\beta$ is also contained in $\overline{\mathbb{S}}(N)$. It follows by Proposition 8.5 that $N \subseteq N_\alpha$. □

Our second observation is that our definition of an $\mathbb{S}$-bias B allows us to insert non-theorems from the fixed point $N = \mathrm{lfp}(\overline{\mathbb{S}[B]}^2, \emptyset)$ to the bias B. This does not affect the fixed point as to be expected.

Proposition 8.18 *If B is an $\mathbb{S}$-bias and $N = \mathrm{lfp}(\overline{\mathbb{S}[B]}^2, \emptyset)$, then $B' = B \cup N$ is also an $\mathbb{S}$-bias such that $N = \mathrm{lfp}(\overline{\mathbb{S}[B']}^2, \emptyset)$.*

Proof. Let ε be any enumeration of $N - B$ and define $B_\alpha = B \cup \{\varepsilon_{0\ldots\alpha}\}$ and $N_\alpha = \mathrm{lfp}(\overline{\mathbb{S}[B_\alpha]}^2, \emptyset)$ for all ordinals $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$. Let us then show that $N_\alpha = N$ holds for all ordinals α in this range. ($\subseteq$) Since $B_\alpha - B \subseteq N \subseteq \overline{\mathcal{S}[B]}(N)$ by Proposition 4.27, we know that $\overline{\mathcal{S}[B_\alpha]}(N) = \overline{\mathcal{S}[B]}(N)$ and $\overline{\mathcal{S}[B_\alpha]}^2(N) = \overline{\mathcal{S}[B_\alpha]}(\overline{\mathcal{S}[B]}(N)) = \overline{\mathcal{S}[B]}^2(N) = N$. Thus N is also $\mathcal{S}[B_\alpha]$-stationary. This implies that $N_\alpha \subseteq N$ by Proposition 4.27. ($\supseteq$) Define $M_\beta = \overline{\mathcal{S}[B]}^2 \uparrow^\beta (\emptyset)$ to show that $M_\beta \subseteq N_\alpha$ for all ordinals β. The base case $\beta = 0$ holds, since $M_0 = \emptyset$. For the induction step, consider a successor ordinal β. Now $M_{\beta-1} \subseteq N_\alpha$ by the inductive hypothesis. Thus $\overline{\mathcal{S}[B_\alpha]}(N_\alpha) \subseteq \overline{\mathcal{S}[B]}(M_{\beta-1})$ by Proposition 4.25 and Lemma 8.4, since $B \subseteq B_\alpha$. It follows by Proposition 4.25 that $\overline{\mathcal{S}[B]}^2(M_{\beta-1}) = M_\beta$ is a subset of $\overline{\mathcal{S}[B]}(\overline{\mathcal{S}[B_\alpha]}(N_\alpha))$. Because $N_\alpha \subseteq N$ and $B \subseteq B_\alpha \subseteq B \cup N$, it holds that $\overline{\mathcal{S}[B]}(N) = \overline{\mathcal{S}[B_\alpha]}(N) \subseteq \overline{\mathcal{S}[B_\alpha]}(N_\alpha)$ by Proposition 4.25. Thus $N \subseteq \overline{\mathcal{S}[B]}(N)$ implies that $N \subseteq \overline{\mathcal{S}[B_\alpha]}(N_\alpha)$ so that the difference $B_\alpha - B \subseteq \overline{\mathcal{S}[B_\alpha]}(N_\alpha)$. Then $\overline{\mathcal{S}[B]}(\overline{\mathcal{S}[B_\alpha]}(N_\alpha)) = \overline{\mathcal{S}[B_\alpha]}^2(N_\alpha) = N_\alpha$ and $M_\beta \subseteq N_\alpha$ follows. For a limit ordinal β, we know by the inductive hypothesis that $M_{\beta'} \subseteq N_\alpha$ for all $\beta' < \beta$. Thus $M_\beta = \bigcup_{\beta'<\beta} M_{\beta'} \subseteq N_\alpha$. So we have established that $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset) \subseteq N_\alpha$.

It remains to be shown that ε satisfies the bias condition for $\mathcal{S}[B]$. So consider any successor ordinal α and the corresponding sentence ε_α in the enumeration ε of $N - B$. Now $\varepsilon_\alpha \in N$. Since $N \subseteq \overline{\mathcal{S}[B]}(N)$ and $B \subseteq B_{\alpha-1} \subseteq B \cup N$, we obtain $\overline{\mathcal{S}[B_{\alpha-1}]}(N) = \overline{\mathcal{S}[B]}(N)$ and $N \subseteq \overline{\mathcal{S}[B_{\alpha-1}]}(N)$. Thus $\varepsilon_\alpha \in \overline{\mathcal{S}[B_{\alpha-1}]}(N)$. Since N coincides with $N_{\alpha-1} = \mathrm{lfp}(\overline{\mathcal{S}[B_{\alpha-1}]}^2, \emptyset)$, we know that ε_α is a bias for $\mathcal{S}[B_{\alpha-1}]$. Consequently, the set of sentences $N - B$ enumerated by ε is an $\mathcal{S}[B]$-bias and $B' = B \cup (N - B) = B \cup N$ is an $\mathcal{S}$-bias by Lemma 8.8. $\square$

8.3.2 Relationship to Stationary and Stable Semantics

In Theorem 8.19, we address the relationship between stationary semantics and cautious semantics in terms of the ordering $\sqsubseteq$ that we imposed on semantics. We have already considered non-monotonic systems $\mathcal{S}$ (such as the one given in Example 8.2) and sets of sentences $T \subseteq \mathcal{L}^+(\mathcal{S})$ such that $\mathrm{STAT}_{\mathcal{S}}(T) \subset \mathrm{CAUT}_{\mathcal{S}}(T)$ results, i.e. cautious semantics provides theorems that remain undetermined under stationary semantics.

Theorem 8.19 $\mathrm{STAT}_{\mathcal{S}} \sqsubseteq \mathrm{CAUT}_{\mathcal{S}}$ *for every non-monotonic system* $\mathcal{S}$*.*

Proof. Let $T \subseteq \mathcal{L}^+(\mathcal{S})$ and assume that $\phi \notin \mathrm{CAUT}_{\mathcal{S}}(T)$. Then there is a finished $\mathcal{S}[T]$-bias B such that $\phi \notin \mathcal{S}[T](N)$ for the least fixed point $N = \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$. By Proposition 8.17, the least fixed point $N' = \mathrm{lfp}(\overline{\mathcal{S}[T]}^2, \emptyset)$ is contained in N. It follows by MON that $\phi \notin \mathcal{S}[T](N')$ either. Thus $\phi \notin \mathrm{STAT}_{\mathcal{S}}(T)$. Hence $\mathrm{STAT}_{\mathcal{S}}(T) \subseteq \mathrm{CAUT}_{\mathcal{S}}(T)$ for all $T \subseteq \mathcal{L}^+(\mathcal{S})$. $\square$

Let us then take stable semantics into consideration. It is shown in Proposition 8.20 that an $\mathbb{S}$-stable set of non-theorems N can be constructed by subsequent biasing using sentences from the difference $N - \mathrm{lfp}(\overline{\mathbb{S}}^2, \emptyset)$.

Proposition 8.20 *If N is an $\mathbb{S}$-stable set of non-theorems, then there is a finished $\mathbb{S}$-bias B such that $N = \mathrm{lfp}(\overline{\mathbb{S}[B]}^2, \emptyset)$.*

Proof. Let $N' = \mathrm{lfp}(\overline{\mathbb{S}}^2, \emptyset)$ and B the difference $N - N'$. Let ε be any enumeration of B and define $B_\alpha = \{\varepsilon_{0\ldots\alpha}\}$ for all α with $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$. It is shown for all α that $N_\alpha = \mathrm{lfp}(\overline{\mathbb{S}[B_\alpha]}^2, \emptyset) \subseteq N$ and B_α is an $\mathbb{S}$-bias.

To show that $N_\alpha \subseteq N$, we define the iteration sequence $M_\beta = \overline{\mathbb{S}[B_\alpha]}^2 \uparrow^\beta (\emptyset)$ for all ordinals β. Let us then show by transfinite induction on β that $M_\beta \subseteq N$. This implies that $N_\alpha = \mathrm{lfp}(\overline{\mathbb{S}[B_\alpha]}^2, \emptyset)$ is contained in N. The base case $\beta = 0$ holds, since $M_0 = \emptyset \subseteq N$. Induction step follows. Consider a successor ordinal β. By the inductive hypothesis, the set $M_{\beta-1}$ is a subset of N. Since $B_\alpha \subseteq N$ and N is $\mathbb{S}$-stable, we obtain $N = \overline{\mathbb{S}[B_\alpha]}(N)$ so that $N \subseteq \overline{\mathbb{S}[B_\alpha]}(M_{\beta-1})$ follows by Proposition 4.25. Thus $\overline{\mathbb{S}[B_\alpha]}^2(M_{\beta-1}) = M_\beta$ is subset of $\overline{\mathbb{S}[B_\alpha]}(N) = N$ by Proposition 4.25. Then consider a limit ordinal β. Now $M_{\beta'} \subseteq N$ for all $\beta' < \beta$ by the inductive hypothesis, so that the limit $M_\beta = \bigcup_{\beta'<\beta} M_{\beta'} \subseteq N$.

To show that B_α is an $\mathbb{S}$-bias, let us consider any successor ordinal α in the range $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$. Let us then assume that $\varepsilon_\alpha \notin \overline{\mathbb{S}[B_{\alpha-1}]}(N_{\alpha-1})$. This implies that $\varepsilon_\alpha \in \mathbb{S}[B_{\alpha-1}](N_{\alpha-1}) = \mathbb{S}(B_{\alpha-1} \cup N_{\alpha-1})$. Since $B_{\alpha-1} \subseteq N$ and $N_{\alpha-1} \subseteq N$, we obtain $\varepsilon_\alpha \in \mathbb{S}(N)$ by the requirement MON. This implies $\varepsilon_\alpha \notin N$ as well as $\varepsilon_\alpha \notin B$, which is a contradiction as ε is an enumeration of B. Hence, the sentence $\varepsilon_\alpha \in \overline{\mathbb{S}[B_\alpha]}(N_{\alpha-1})$ for all successor ordinals α indicating that the set of non-theorems B is indeed an $\mathbb{S}$-bias. This implies that $N' \subseteq N_{\mathrm{Ind}(\varepsilon)} \subseteq N$ by Proposition 8.17. Now $B = N - N'$ and $N' \subseteq N_{\mathrm{Ind}(\varepsilon)}$ imply that $N \subseteq B \cup N_{\mathrm{Ind}(\varepsilon)}$. It follows that $\overline{\mathbb{S}[B]}(N_{\mathrm{Ind}(\varepsilon)}) \subseteq \overline{\mathbb{S}}(N) = N$ by Proposition 4.25. For the same reason, $\overline{\mathbb{S}[B]}(N) = N$ is a subset of $\overline{\mathbb{S}[B]}^2(N_{\mathrm{Ind}(\varepsilon)}) = N_{\mathrm{Ind}(\varepsilon)}$. So $N_{\mathrm{Ind}(\varepsilon)} = N$ has been established and it is not difficult to see that B is finished, since the $\mathbb{S}$-completeness of N implies that of $B \cup N_{\mathrm{Ind}(\varepsilon)}$. Thus B fulfills the requirements of the proposition. $\square$

In Theorem 8.21, we give the ordering between cautious semantics and stable semantics using $\sqsubseteq$. For the non-monotonic system $\mathbb{S}$ in Example 8.13, stable semantics provides more theorems than cautious semantics, since there are no $\mathbb{S}$-stable sets of non-theorems implying that $\mathrm{STAB}_{\mathbb{S}}(\emptyset) = \mathcal{L}^+(\mathbb{S})$. On the other hand, cautious semantics gives us a set of theorems $\mathrm{CAUT}_{\mathbb{S}}(\emptyset) = \mathrm{Cn}(\emptyset)$. Another example where stable semantics provides strictly more theorems is given later as Example 8.28. In this case, there are $\mathbb{S}$-stable sets of non-theorems so that also $\mathrm{STAB}_{\mathbb{S}}(\emptyset) \neq \mathcal{L}^+(\mathbb{S})$.

Theorem 8.21 $\mathrm{CAUT}_{\mathbb{S}} \sqsubseteq \mathrm{STAB}_{\mathbb{S}}$ *for every non-monotonic system* $\mathbb{S}$.

Proof. Let $T \subseteq \mathcal{L}^{+}(\mathbb{S})$ and assume that $\phi \notin \mathrm{STAB}_{\mathbb{S}}(T)$. Then there is an $\mathbb{S}[T]$-stable set of non-theorems N such that $\phi \notin \mathbb{S}[T](N)$. By Proposition 8.20, there is a finished $\mathbb{S}[T]$-bias B such that $N = \mathrm{lfp}(\overline{\mathbb{S}[B]}^{2}, \emptyset)$. This implies that $\phi \notin \mathrm{CAUT}_{\mathbb{S}}(T)$. Hence $\mathrm{CAUT}_{\mathbb{S}}(T) \subseteq \mathrm{STAB}_{\mathbb{S}}(T)$ for all $T \subseteq \mathcal{L}^{+}(\mathbb{S})$. □

Minimal model systems provide an interesting case where cautious semantics and stable semantics coincide. The result is established in Lemma 8.22 and Theorem 8.23 that follow. This interconnection is especially useful when the computational complexity of cautious semantics is analyzed later.

Lemma 8.22 *Let $P \subseteq \mathcal{A}$ be a set of atoms to be minimized. If B is a $\mathfrak{C}_P[T]$-bias and T is propositionally consistent, then the set of sentences $T \cup \mathrm{Neg}_P(B)$ is propositionally consistent.*

Proof. Since B is a $\mathfrak{C}_P[T]$-bias, there is an enumeration ε of B satisfying the bias condition for $\mathfrak{C}_P[T]$. Let us then define $B_\alpha = \{\varepsilon_{0\ldots\alpha}\}$ and $N_\alpha = \mathrm{lfp}(\overline{\mathfrak{C}_P[T; B_\alpha]}^{2}, \emptyset)$ and prove by transfinite induction on α that $T \cup \mathrm{Neg}_P(B_\alpha)$ is propositionally consistent.

The base case $\alpha = \emptyset$ holds, since $T \cup \mathrm{Neg}_P(B_0) = T$ is propositionally consistent by assumption. Induction step follows. Let us consider a successor ordinal α and assume that $T \cup \mathrm{Neg}_P(B_\alpha)$ is not propositionally consistent. Now $B_\alpha = B_{\alpha-1} \cup \{\varepsilon_\alpha\}$ and two cases arise depending on whether $\varepsilon_\alpha \in P$ or not. (i) If $\varepsilon_\alpha \notin P$, it follows that $T \cup \mathrm{Neg}_P(B_\alpha) = T \cup \mathrm{Neg}_P(B_{\alpha-1})$. The latter is propositionally consistent by the inductive hypothesis. Thus we have a contradiction. (ii) If $\varepsilon_\alpha \in P$, we obtain $\mathrm{Neg}_P(B_\alpha) = \mathrm{Neg}_P(B_{\alpha-1}) \cup \{\neg\varepsilon_\alpha\}$ and thus $T \cup \mathrm{Neg}_P(B_{\alpha-1}) \cup \{\neg\varepsilon_\alpha\}$ is propositionally inconsistent, which makes ε_α a logical consequence of $T \cup \mathrm{Neg}_P(B_{\alpha-1})$. It follows that $\varepsilon_\alpha \in \mathrm{Cn}(T \cup \mathrm{Neg}_P(B_{\alpha-1} \cup N_{\alpha-1}))$ by the monotonicity of $\models$. Thus $\varepsilon_\alpha \in \mathfrak{C}_P[T; B_{\alpha-1}](N_{\alpha-1})$ and $\varepsilon_\alpha \notin \overline{\mathfrak{C}_P[T; B_{\alpha-1}]}(N_{\alpha-1})$ which indicates that ε_α is not a bias for $\mathfrak{C}_P[T; B_{\alpha-1}]$, a contradiction. By the cases (i) and (ii), we have a contradiction. Hence $T \cup \mathrm{Neg}_P(B_\alpha)$ is propositionally consistent. Finally, consider a limit ordinal α. Since $B_\alpha = \bigcup_{\beta<\alpha} B_\beta$, it is clear that $T \cup \mathrm{Neg}_P(B_\alpha)$ coincides with $T \cup \neg(P \cap B_\alpha) = \bigcup_{\beta<\alpha}(T \cup \neg(P \cap B_\beta)) = \bigcup_{\beta<\alpha}(T \cup \mathrm{Neg}_P(B_\beta))$. Because each $T \cup \mathrm{Neg}_P(B_\beta)$ is propositionally consistent for all ordinals $\beta < \alpha$ by the inductive hypothesis, it follows that $T \cup \mathrm{Neg}_P(B_\alpha)$ is propositionally consistent by the compactness of propositional logic (as defined in Appendix A). □

Theorem 8.23 *Let $P \subseteq \mathcal{A}$ be a set of atoms to be minimized. Then it holds for all $T \subseteq \mathcal{L}^{+}(\mathfrak{C}_P)$ that* $\mathrm{CAUT}_{\mathfrak{C}_P}(T) = \mathrm{STAB}_{\mathfrak{C}_P}(T)$.

Proof. ($\subseteq$) Let $\phi \notin \mathrm{CAUT}_{\mathfrak{C}_P}(T)$ hold. Then there is a finished $\mathfrak{C}_P[T]$-bias B such that $\phi \notin \mathfrak{C}_P(T, N)$ for the set of non-theorems $N = \mathrm{lfp}(\overline{\mathfrak{C}_P[T;B]}^2, \emptyset)$. Then $\phi \notin \mathfrak{C}_P(T, N)$ implies $\phi \notin \mathrm{Cn}(T \cup \mathrm{Neg}_P(N))$ so that T must be propositionally consistent. By Proposition 8.15, the set of non-theorems N satisfies $N = \overline{\mathfrak{C}_P[T;B]}(N) = \mathcal{L} - \mathrm{Cn}(T \cup \mathrm{Neg}_P(B \cup N))$. Let us then show that $P \cap B \subseteq N$. So let $\psi \in P \cap B$ hold, i.e. $\psi \in P$ and $\psi \in B$. Then assume that $\psi \notin N$. It follows that $\psi \in \mathfrak{C}_P[T;B](N)$, i.e. $\psi \in \mathrm{Cn}(T \cup \mathrm{Neg}_P(B \cup N))$. Since $\neg\psi \in \mathrm{Neg}_P(B \cup N) = \neg(P \cap (B \cup N))$, the theory $T \cup \mathrm{Neg}_P(B \cup N)$ is propositionally inconsistent. This implies that $N = \emptyset$ and thus also $T \cup \mathrm{Neg}_P(B)$ is propositionally inconsistent, a contradiction with Lemma 8.22. Hence $P \cap B \subseteq N$ implying that $\mathrm{Neg}_P(B \cup N) = \neg(P \cap (B \cup N)) = \neg(P \cap N) = \mathrm{Neg}_P(N)$. Therefore $N = \overline{\mathfrak{C}_P[T]}(N)$, i.e. the set of non-theorems N is also $\mathfrak{C}_P[T]$-stable. It follows that $\phi \notin \mathrm{STAB}_{\mathfrak{C}_P}(T)$. ($\supseteq$) Let $\phi \notin \mathrm{STAB}_{\mathfrak{C}_P}(T)$. Then there is a $\mathfrak{C}_P[T]$-stable set of non-theorems N such that $\phi \notin \mathfrak{C}_P(T, N)$. By Proposition 8.20, there is a finished $\mathfrak{C}_P[T]$-bias B such that $N = \mathrm{lfp}(\overline{\mathfrak{C}_P[T;B]}^2, \emptyset)$. Since $\phi \notin \mathfrak{C}_P(T, N)$, it follows that $\phi \notin \mathrm{CAUT}_{\mathfrak{C}_P}(T)$. $\square$

8.3.3 Cumulativity Analysis

In Section 5.1, we introduced cumulativity as a reasonable substitute for monotonicity, which ceases to hold if non-monotonic systems are considered. Our results indicate that stationary semantics $\mathrm{STAT}_{\mathfrak{S}}$ has this property in general (regardless of the non-monotonic system $\mathfrak{S}$ under consideration) while stable semantics $\mathrm{STAB}_{\mathfrak{S}}$ satisfies only one half of the property, namely the requirement CCLO, but not the requirement CMON. In the following, we investigate the cumulativity property in conjunction with cautious semantics. We begin with the requirement CMON and show that – in contrast to stable semantics – cautious semantics satisfies this requirement. Then the requirement CCLO is addressed: it turns out that cautious semantics does not have this property in general. Finally, the possibilities of strengthening cautious semantics to meet the requirement CCLO are analyzed.

To establish CMON for cautious semantics (Theorem 8.27), we have to deduce some subsidiary results in Lemma 8.24 and Propositions 8.25 and 8.26 that follow. Basically, it is shown in three phases that for a finished $\mathfrak{S}[T]$-bias B – which is obtained using a set of sentences $T \subseteq \mathcal{L}^+(\mathfrak{S})$ as lemmas under cautious semantics – there is a finished $\mathfrak{S}$-bias B' such that the resulting least stationary fixed points $\mathrm{lfp}(\overline{\mathfrak{S}[T;B]}^2, \emptyset)$ and $\mathrm{lfp}(\overline{\mathfrak{S}[B']}^2, \emptyset)$ coincide.

Lemma 8.24 *Let $T \subseteq \mathcal{L}^+(\mathfrak{S})$ such that $T \subseteq \mathfrak{S}(N)$ holds for every $N = \mathrm{lfp}(\overline{\mathfrak{S}[B]}^2, \emptyset)$ induced by a finished $\mathfrak{S}$-bias B. Let B be an $\mathfrak{S}[T]$-bias and $N = \mathrm{lfp}(\overline{\mathfrak{S}[T;B]}^2, \emptyset)$. If B is also an $\mathfrak{S}$-bias, then $\mathrm{lfp}(\overline{\mathfrak{S}[B]}^2, \emptyset) \subseteq N$.*

Proof. Define $N_\alpha = \overline{\mathcal{S}[B]}^2 \uparrow^\alpha (\emptyset)$ for all ordinals α. Let us then show by transfinite induction on α that $N_\alpha \subseteq N$ for all ordinals α. For the base case $\alpha = 0$, we note that $N_0 = \emptyset$ is a subset of N. Induction step follows.

Consider a successor ordinal α. It holds that $N_{\alpha-1} \subseteq N$ by the inductive hypothesis. Thus $\overline{\mathcal{S}[T;B]}(N)$ is a subset of $\overline{\mathcal{S}[T;B]}(N_{\alpha-1})$ by Proposition 4.25. The latter is a subset of $\overline{\mathcal{S}[B]}(N_{\alpha-1})$ by Lemma 8.4. These imply together that $\overline{\mathcal{S}[T;B]}(N) \subseteq \overline{\mathcal{S}[B]}(N_{\alpha-1})$. It follows that $\overline{\mathcal{S}[T;B]}(\overline{\mathcal{S}[B]}(N_{\alpha-1}))$ is a subset $\overline{\mathcal{S}[T;B]}^2(N) = N$ by Proposition 4.25. It is shown below that the set of theorems T is contained in $\mathcal{S}[B](\overline{\mathcal{S}[B]}(N_{\alpha-1}))$. This implies that $\overline{\mathcal{S}[T;B]}(\overline{\mathcal{S}[B]}(N_{\alpha-1})) = \overline{\mathcal{S}[B]}^2(N_{\alpha-1}) = N_\alpha$ is a subset of N as desired.

Since B is an $\mathcal{S}$-bias by assumption, it can be extended by Theorem 8.14 to a finished $\mathcal{S}$-bias B' which gives rise to $N' = \mathrm{lfp}(\overline{\mathcal{S}[B']}^2, \emptyset)$. Then we know that $T \subseteq \mathcal{S}(N')$ by our assumption on T. Also $N_{\alpha-1} \subseteq N'$ by Proposition 8.17, since $B' - B$ is an $\mathcal{S}[B]$-bias. Thus $\overline{\mathcal{S}[B']}(N') \subseteq \overline{\mathcal{S}[B']}(N_{\alpha-1}) \subseteq \overline{\mathcal{S}[B]}(N_{\alpha-1})$ by Proposition 4.25 and Lemma 8.4. Since N' is $\mathcal{S}[B']$-stationary, it holds that $N' \subseteq \overline{\mathcal{S}[B']}(N')$ so that also $N' \subseteq \overline{\mathcal{S}[B]}(N_{\alpha-1})$. Since $T \subseteq \mathcal{S}(N')$, it follows that also $T \subseteq \mathcal{S}[B](\overline{\mathcal{S}[B]}(N_{\alpha-1}))$ by the requirement MON.

Finally, consider a limit ordinal α. By the inductive hypothesis, $N_\beta \subseteq N$ for all $\beta < \alpha$. Thus the limit $N_\alpha = \bigcup_{\beta<\alpha} N_\beta$ is also a subset of N. $\square$

Proposition 8.25 *Let $T \subseteq \mathcal{L}^+(\mathcal{S})$ such that $T \subseteq \mathcal{S}(N)$ holds for every $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$ induced by a finished $\mathcal{S}$-bias B. If B is an $\mathcal{S}[T]$-bias, then B is also an $\mathcal{S}$-bias and $\mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset) \subseteq \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$.*

Proof. Since B is an $\mathcal{S}[T]$-bias, there is an enumeration ε of B that satisfies the bias condition of Definition 8.7. Let us then define $B_\alpha = \{\varepsilon_{0\ldots\alpha}\}$ for all ordinals $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$. In the following, it is shown by transfinite induction on α for all $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$ that (i) B_α is an $\mathcal{S}$-bias and (ii) the fixed point $N'_\alpha = \mathrm{lfp}(\overline{\mathcal{S}[B_\alpha]}^2, \emptyset)$ is a subset of $N_\alpha = \mathrm{lfp}(\overline{\mathcal{S}[T;B_\alpha]}^2, \emptyset)$.

For the base case $\alpha = 0$ we note firstly that (i) $B_0 = \emptyset$ is trivially an $\mathcal{S}$-bias and thus (ii) $N'_0 = \mathrm{lfp}(\overline{\mathcal{S}[B_0]}^2, \emptyset)$ is a subset of N_0 by Lemma 8.24. Induction step follows. Let α be a successor ordinal. By the inductive hypothesis, the set of non-theorems $B_{\alpha-1}$ is an $\mathcal{S}$-bias and $N'_{\alpha-1} \subseteq N_{\alpha-1}$. (i) Now the sentence ε_α is a bias for $\mathcal{S}[T; B_{\alpha-1}]$, i.e. $\varepsilon_\alpha \in \overline{\mathcal{S}[T; B_{\alpha-1}]}(N_{\alpha-1})$. If follows by Proposition 4.25 and Lemma 8.4 that $\overline{\mathcal{S}[T; B_{\alpha-1}]}(N_{\alpha-1}) \subseteq \overline{\mathcal{S}[T; B_{\alpha-1}]}(N'_{\alpha-1}) \subseteq \overline{\mathcal{S}[B_{\alpha-1}]}(N'_{\alpha-1})$. This implies that $\varepsilon_\alpha \in \overline{\mathcal{S}[B_{\alpha-1}]}(N'_{\alpha-1})$ so that ε_α is a bias for $\mathcal{S}[B_{\alpha-1}]$. Hence $B_\alpha = B_{\alpha-1} \cup \{\varepsilon_\alpha\}$ is an $\mathcal{S}$-bias. (ii) It follows that $N'_\alpha \subseteq N_\alpha$ by Lemma 8.24. Finally, consider a limit ordinal α. By the inductive hypothesis, each B_β where $\beta < \alpha$ is an $\mathcal{S}$-bias. Thus the limit $B_\alpha = \bigcup_{\beta<\alpha} B_\beta$ is also an $\mathcal{S}$-bias by Proposition 8.9. This implies that $N'_\alpha \subseteq N_\alpha$ by Lemma 8.24. $\square$

Proposition 8.26 *Let $T \subseteq \mathcal{L}^+(\mathcal{S})$ such that $T \subseteq \mathcal{S}(N)$ holds for every $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$ induced by a finished $\mathcal{S}$-bias B. If B is a finished $\mathcal{S}[T]$-bias and $N = \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$, then there is a finished $\mathcal{S}$-bias B' such that $N = \mathrm{lfp}(\overline{\mathcal{S}[B']}^2, \emptyset)$.*

Proof. Let B be a finished $\mathcal{S}[T]$-bias and $N = \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$. Then it follows by Proposition 8.18 that $B' = B \cup N$ is an $\mathcal{S}[T]$-bias such that $N = \mathrm{lfp}(\overline{\mathcal{S}[T;B']}^2, \emptyset)$. It is also obvious that B' is a finished $\mathcal{S}[T]$-bias, since $B \cup N$ and $B' \cup N = B \cup N$ are $\mathcal{S}[T]$-complete sets of non-theorems. Thus $N = \overline{\mathcal{S}[T;B']}(N)$ by Proposition 8.15. By Proposition 8.25, we know that B' is also an $\mathcal{S}$-bias such that $N' = \mathrm{lfp}(\overline{\mathcal{S}[B']}^2, \emptyset) \subseteq N$, but there is no guarantee that B' is a finished $\mathcal{S}$-bias anymore. Therefore, we extend B' to a finished $\mathcal{S}$-bias B'' such that $B' \subseteq B''$ and $N' \subseteq N'' = \mathrm{lfp}(\overline{\mathcal{S}[B'']}^2, \emptyset)$ by Proposition 8.17. This is always possible by Theorem 8.14. Now our requirement on T implies $T \subseteq \mathcal{S}(N'')$. Because B'' is a finished $\mathcal{S}$-bias it holds that $N'' = \overline{\mathcal{S}[B'']}(N'') = \overline{\mathcal{S}[T;B'']}(N'')$ by Propositions 8.15 and 5.12.

Since $N \subseteq B' \subseteq B''$, it holds that $N'' = \overline{\mathcal{S}[T;B'']}(N'') = \overline{\mathcal{S}[T;B'']}(N'' \cup N)$. Then $N'' \subseteq \overline{\mathcal{S}[T;B'']}(N) \subseteq \overline{\mathcal{S}[T;B']}(N) = N$ by Proposition 4.25 and Lemma 8.4. Thus $T \subseteq \mathcal{S}(N)$ by the requirement MON. Assume then that $\phi \in B'' - B'$, i.e. ϕ was introduced when B' was extended to B''. It follows by Proposition 8.17 that $\phi \in \overline{\mathcal{S}[B']}(N')$, since B'' is an $\mathcal{S}[B']$-bias. Since $N' \subseteq N \subseteq B'$, we obtain $B' \cup N = B' \cup N'$ and $\phi \in \overline{\mathcal{S}[B']}(N)$. Moreover, it holds by Lemma 5.11 that $\overline{\mathcal{S}[B']}(N) = \overline{\mathcal{S}[T;B']}(N)$, since $T \subseteq \mathcal{S}(N)$. Thus $\phi \in \overline{\mathcal{S}[T;B']}(N)$ and $\phi \in N$. But then $N \subseteq B'$ implies $\phi \in B'$, a contradiction. Hence $B'' - B'$ is empty so that $B'' = B'$, $N' = N''$ and $N'' \subseteq N$ hold. Thus $N = \overline{\mathcal{S}[T;B']}(N) = \overline{\mathcal{S}[T;B'']}(N)$ is a subset of $\overline{\mathcal{S}[B'']}(N'') = N''$ by Proposition 4.25. So we have established that $N'' = N$ and in fact N, N' and N'' are all equal. Thus B' is a finished $\mathcal{S}$-bias such that $N = \mathrm{lfp}(\overline{\mathcal{S}[B']}^2, \emptyset)$. □

Proposition 8.26 is now strong enough so that we can prove in Theorem 8.27 that cautious semantics is cumulatively monotonic. This indicates that – in contrast to stable semantics – non-monotonic conclusions are not lost if lemmas are introduced under cautious semantics.

Theorem 8.27 *Cautious semantics* $\mathrm{CAUT}_{\mathcal{S}}$ *is cumulatively monotonic.*

Proof. Let $T \subseteq T' \subseteq \mathrm{CAUT}_{\mathcal{S}}(T)$. It is then shown that $\mathrm{CAUT}_{\mathcal{S}}(T) \subseteq \mathrm{CAUT}_{\mathcal{S}}(T')$. So let us assume that $\phi \notin \mathrm{CAUT}_{\mathcal{S}}(T')$. Then there is a finished $\mathcal{S}[T']$-bias B' such that $\phi \notin \mathcal{S}[T'](N')$ where N' is the fixed point $\mathrm{lfp}(\overline{\mathcal{S}[T';B']}^2, \emptyset)$. The difference $T' - T$ satisfies the requirement of Proposition 8.26 for the system $\mathcal{S}[T]$. It follows that there is a finished $\mathcal{S}[T]$-bias B

such that $N' = \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$. Since $T \subseteq T'$ holds, we obtain $\phi \notin \mathcal{S}[T](N')$ by the requirement MON. Therefore $\phi \notin \mathrm{CAUT}_{\mathcal{S}}(T)$. $\square$

Let us then take the requirement CCLO into consideration. Example 8.28 shows that cautious semantics is not cumulatively closed in general.

Example 8.28 Let $D = \{\frac{:\neg\mathrm{p}}{\mathrm{q}}, \frac{:\neg\mathrm{q}}{\mathrm{p}}, \frac{\mathrm{r}:\neg\mathrm{s}}{\mathrm{t}}, \frac{\mathrm{r}:\neg\mathrm{t}}{\mathrm{s}}\}$ and $T = \{\mathrm{p} \vee \mathrm{q} \rightarrow \mathrm{r}, \mathrm{s} \vee \mathrm{t} \rightarrow \mathrm{u}\}$.

Let $\mathcal{S}$ be the subsystem $\mathfrak{D}_D^{\neg\mathrm{J}(D)}$ with the language $\langle \mathcal{L}, \{\neg\neg\mathrm{p}, \neg\neg\mathrm{q}, \neg\neg\mathrm{s}, \neg\neg\mathrm{t}\}\rangle$. This example is a variant of the one presented in Example 8.1, though rewritten with defaults. In a way, this example is a representation of two case-analysis steps in a "cascade". It is easy to see, that the least $\mathcal{S}[T]$-stationary set of non-theorems is $\emptyset$. Moreover, there are four $\mathcal{S}[T]$-stable sets of non-theorems N each of which contains exactly one non-theorem from $\{\neg\neg\mathrm{p}, \neg\neg\mathrm{q}\}$ and exactly one non-theorem from $\{\neg\neg\mathrm{s}, \neg\neg\mathrm{t}\}$. On the basis of these and the sentences of T, we may argue that the sentences r as well as u are desired theorems for the non-monotonic system $\mathcal{S}[T]$.

In our biasing approach, we might first use the set of non-theorems $B_1 = \{\neg\neg\mathrm{p}\}$ as an $\mathcal{S}$-bias to obtain $N_1 = \mathrm{lfp}(\overline{\mathcal{S}[T;B_1]}^2, \emptyset) = \{\neg\neg\mathrm{p}\}$. Now $D_{N_1} = \{\frac{\top}{\mathrm{q}}\}$. Thus we obtain the theorems q and r already. However, the $\mathcal{S}[T]$-bias bias B_1 is not finished, since $B_1 \cup N_1$ is not $\mathcal{S}[T]$-complete. Therefore we have take one of the potential non-theorems $\overline{\mathcal{S}[T;B_1]}(N_1) = \{\neg\neg\mathrm{p}, \neg\neg\mathrm{s}, \neg\neg\mathrm{t}\}$ to extend B_1. So let us pick $\neg\neg\mathrm{s}$ and form our second bias $B_2 = \{\neg\neg\mathrm{p}, \neg\neg\mathrm{s}\}$. Consequently $N_2 = \mathrm{lfp}(\overline{\mathcal{S}[T;B_2]}^2, \emptyset) = \{\neg\neg\mathrm{p}, \neg\neg\mathrm{s}\}$ results. Now D_{N_2} is $\{\frac{\top}{\mathrm{q}}, \frac{\mathrm{r}}{\mathrm{t}}\}$. Thus we obtain also the theorems t and u by the resulting set of non-theorems N_2. Moreover, the $\mathcal{S}[T]$-bias B_2 is already finished and further extension is not necessary nor possible, since $\overline{\mathcal{S}[T;B_2]}(N_2) = N_2$.

Unfortunately, there are also biasing orderings that lead to complications from the point of view of cumulativity. Suppose we consider the sentences in the order $\neg\neg\mathrm{s}, \neg\neg\mathrm{t}, \neg\neg\mathrm{p}$. Let us verify that this is an acceptable enumeration in the sense of Definition 8.7. For $B_1 = \{\neg\neg\mathrm{s}\}$ and $B_2 = \{\neg\neg\mathrm{s}, \neg\neg\mathrm{t}\}$ we obtain $\mathrm{lfp}(\overline{\mathcal{S}[T;B_1]}^2, \emptyset) = \mathrm{lfp}(\overline{\mathcal{S}[T;B_2]}^2, \emptyset) = \emptyset$ and $\overline{\mathcal{S}[T;B_1]}(\emptyset) = \overline{\mathcal{S}[T;B_2]}(\emptyset) = \mathcal{L}^-(\mathcal{S})$. Thus we form $B_3 = \{\neg\neg\mathrm{s}, \neg\neg\mathrm{t}, \neg\neg\mathrm{p}\}$. This is a crucial step, since $N_3 = \mathrm{lfp}(\overline{\mathcal{S}[T;B_3]}^2, \emptyset) = \{\neg\neg\mathrm{p}\}$ results. Unfortunately, the $\mathcal{S}$-bias B_3 is finished ($B_3 \cup N_3$ is $\mathcal{S}[T]$-complete) and thus we are ready with biasing. But then we lose the theorem u, since $D_{N_3} = \{\frac{\top}{\mathrm{q}}\}$ and $\mathrm{u} \notin \mathrm{Cn}^{D_{N_3}}(W)$.

This phenomenon has far-reaching consequences on the properties of cautious semantics. It turns out that with the system of $\mathcal{S}$ of this example, the sentence r is always included as a theorem in $\mathcal{S}[T](N)$ for all fixed points $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$ induced by $\mathcal{S}[T]$-biases B. Thus $\mathrm{r} \in \mathrm{CAUT}_{\mathcal{S}}(T)$. This entitles us to fix the sentence r as a theorem in order to verify whether our biasing framework satisfies cumulative closure. Once r is fixed as a theorem, the biases change in a profound way, namely biases like B_3 are not realized anymore and for finished $\mathbb{S}[T \cup \{\mathrm{r}\}]$-biases, the sentence u is always a theorem. Thus $\mathbb{S}[T \cup \{\mathrm{r}\}]$ provides more theorems under cautious semantics. Consequently, the operator $\mathrm{CAUT}_{\mathbb{S}}$ does not satisfy the requirement CCLO in Definition 5.1, because $\mathrm{CAUT}_{\mathbb{S}}(T \cup \{\mathrm{r}\}) \not\subseteq \mathrm{CAUT}_{\mathbb{S}}(T)$ follows by the analysis above. □

A way of understanding this example is that our biasing principle is local therefore unable to see the global effect. By introducing some non-theorems too early (like $\neg\neg\mathrm{t}$ in B_3 above), we may block future possibilities to extend the resulting least stationary set of non-theorems. In our example, we could postpone biasing with $\neg\neg\mathrm{t}$, and obtain the bias $\{\neg\neg\mathrm{p}, \neg\neg\mathrm{s}\}$ directly and thus a larger set of non-theorems would result. This observation suggests that we should impose further conditions on biases. The problem is that such conditions tend to endanger the other half of cumulativity, namely the cumulative monotonicity requirement CMON. To illustrate the delicate balance between the properties CMON and CCLO, we present a challenging example which was proposed by You (as reported by Dix [20]). Originally, this example was presented as a logic program, but we handle a representation with defaults [5].

Example 8.29 Consider a set of defaults $D = \{\frac{:\neg\mathrm{b}}{\mathrm{a}}, \frac{:\neg\mathrm{b}}{\mathrm{c}}, \frac{\mathrm{c}:\neg\mathrm{a}}{\mathrm{b}}\}$ and the subsystem $\mathbb{S} = \mathfrak{D}_D^{\neg\mathrm{J}(D)}$ having the language $\langle \mathcal{L}, \{\neg\neg\mathrm{a}, \neg\neg\mathrm{b}\}\rangle$.

The least $\mathbb{S}$-stationary set of non-theorems N is $\emptyset$. There are two finished $\mathbb{S}$-biases, namely $B_1 = \{\neg\neg\mathrm{a}, \neg\neg\mathrm{b}\}$ and $B_2 = \{\neg\neg\mathrm{b}\}$ that induce least fixed points $N_1 = \mathrm{lfp}(\overline{\mathbb{S}[B_1]}^2, \emptyset) = \emptyset$ and $N_2 = \mathrm{lfp}(\overline{\mathbb{S}[B_2]}^2, \emptyset) = \{\neg\neg\mathrm{b}\}$, respectively. There is only one $\mathbb{S}$-stable set of non-theorems $N_2 = \{\neg\neg\mathrm{b}\}$.

On the basis of these observations, let us show how the requirement CMON fails for stable semantics, but not for cautious semantics. Since there is only one $\mathbb{S}$-stable set of non-theorems, we obtain $\mathrm{STAB}_{\mathbb{S}}(\emptyset) = \mathbb{S}(N_2) = \mathrm{Cn}(\{\mathrm{a}, \mathrm{c}\})$. Therefore, we consider the semantics of $T = \{\mathrm{c}\}$. Now two $\mathbb{S}[T]$-stable sets of non-theorems result, namely N_2 and $N_3 = \{\neg\neg\mathrm{a}\}$. These give rise to the sets of theorems $\mathrm{Cn}(\{\mathrm{a}, \mathrm{c}\})$ and $\mathrm{Cn}(\{\mathrm{b}, \mathrm{c}\})$, respectively. Thus $\mathrm{STAB}_{\mathbb{S}}(T) = \mathrm{Cn}(\{\mathrm{a}, \mathrm{c}\}) \cap \mathrm{Cn}(\{\mathrm{b}, \mathrm{c}\})$ and we lose our earlier conclusion a. This indicates that $\emptyset \subseteq T \subseteq \mathrm{STAB}_{\mathbb{S}}(\emptyset)$ *does not* imply $\mathrm{STAB}_{\mathbb{S}}(\emptyset) \subseteq \mathrm{STAB}_{\mathbb{S}}(T)$. This trap is avoided by cautious semantics, since there appears an additional set of non-theorems $N_1 = \emptyset$ (induced by B_1). Thus $\mathrm{CAUT}_{\mathbb{S}}(\emptyset) = \mathrm{Cn}(\emptyset)$ and therefore $T = \{\mathrm{c}\}$ does not satisfy $\emptyset \subseteq T \subseteq \mathrm{CAUT}_{\mathbb{S}}(\emptyset)$. □

Let us now address the balance between the properties CMON and CCLO in terms of the example above and Example 8.28 presented earlier. The latter demonstrated that cautious semantics is not cumulatively closed. To enforce this property for cautious semantics we should by one way or another discard biases (such as $B = \{\neg\neg s, \neg\neg t, \neg\neg p\}$ in Example 8.28) that counteract the closure property CCLO. Formulating conditions for such discrimination among biases is non-trivial, because the cumulative monotonicity property is easily lost when such a condition is introduced. To demonstrate this, we suggest a condition on biases that excludes a $\mathfrak{S}$-bias B if there is another $\mathfrak{S}$-bias B' that induces a greater fixed point, i.e. $\mathrm{lfp}(\overline{\mathfrak{S}[B]}^2, \emptyset) \subset \mathrm{lfp}(\overline{\mathfrak{S}[B']}^2, \emptyset)$. This maximisation principle is a remedy to the lack of cumulative closure in Example 8.28, since only biases inducing $\mathfrak{S}$-stable sets of non-theorems are accepted and as it was shown in Proposition 5.12 these remain $\mathfrak{S}[T]$-stable sets of non-theorems if a set of lemmas T is introduced.

Unfortunately, we lose the cumulative monotonicity property by this maximisation principle. To see this, note that the bias B_1 in Example 8.29 is discarded and then the arguments presented for stable semantics apply. Let us yet note that the maximisation principle under consideration produces a variant of cautious semantics which is still weaker than stable semantics. This is easily inspected from Example 5.16 where a finished $\mathfrak{S}$-bias $B = \{a, p\}$ produces a fixed point $N = \mathrm{lfp}(\overline{\mathfrak{S}[B]}^2, \emptyset) = \{a\}$ which is not an $\mathfrak{S}$-stable set of non-theorems. Consequently, this example is not a counter-example to the cumulative monotonicity of the varied cautious semantics (by the maximisation principle under consideration), even though it shows that stable semantics is not cumulatively monotonic in general.

We have considered several other conditions to discard biases counteracting the cumulative closure property. The examples considered so far rule out the following approaches within our biasing framework: (i) selecting biases B that produce maximal least fixed points $\mathrm{lfp}(\overline{\mathfrak{S}[B]}^2, \emptyset)$ (this approach was discussed above), (ii) selecting $\mathfrak{S}$-biases B that are finished but of minimal size, (iii) selecting biases for which the inclusion given by Proposition 8.5 is always proper and (iv) selecting biases B that are realized as non-theorems, i.e. $B \subseteq \mathrm{lfp}(\overline{\mathfrak{S}[B]}^2, \emptyset)$. Thus we have to come up with another principle, even though – to gain the property CCLO – our overall goal is finding biases B that produce larger and larger fixed points $\mathrm{lfp}(\overline{\mathfrak{S}[B]}^2, \emptyset)$. We leave finding such conditions as an open research problem. Nevertheless, there are interesting cases where cautious semantics (as given in Definition 8.16) becomes cumulatively closed and thus cumulative. By Theorem 8.23 cautious semantics and stable semantics coincide for minimal model systems $\mathfrak{C}_P$. Then it holds by Theorems 5.13 and 8.27 that both semantics are cumulative in this particular case. Therefore the three theorems provide an indirect proof of the cumulativity result obtained by Corollary 5.27 and Proposition 5.28.

8.3.4 Preservation by Translations

Our notion of biases for a non-monotonic system $\mathcal{S}$ was built upon the notion of the least stationary set of non-theorems. In this section, we ensure that the bias construction is preserved by the relations introduced in Section 3: subsystems, functional equivalences and embeddings. These relations were already considered in conjunction with stationary semantics in Section 6. The results obtained therein give a natural starting point for our analysis. To begin with, we consider how biases are mapped for subsystems and back.

Lemma 8.30 *Let $\mathcal{S}$ be any non-monotonic system and $\mathcal{S}' \sqsubseteq \mathcal{S}$ its subsystem. If $N = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$, then $N' = N \cap \mathcal{L}^-(\mathcal{S}') = \mathrm{lfp}(\overline{\mathcal{S}'}^2, \emptyset)$.*

Proof. Let $N = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$. Since N is $\mathcal{S}$-stationary, so is $N' = N \cap \mathcal{L}^-(\mathcal{S}')$ by Proposition 6.3. Then assume that N' is not the least $\mathcal{S}'$-stationary set of non-theorems, i.e. there is $\mathcal{S}'$-stationary set of non-theorems M' such that $M' \subset N'$. Then there is an $\mathcal{S}$-stationary set of non-theorems M such that $M' = M \cap \mathcal{L}^-(\mathcal{S}')$ by Proposition 6.12. Moreover, the set of non-theorems M is obtained as $\mathrm{lfp}(\overline{\mathcal{S}}^2, M')$. Then M is the least $\mathcal{S}$-stationary set of non-theorems satisfying $M' \subseteq M$ by Proposition 4.27. Since also $M' \subseteq N$, it follows that $M \subseteq N$. Recall that N is the least $\mathcal{S}$-stationary set of non-theorems, so that $N \subseteq M$ and $N = M$ follow. Thus also $N' = M'$, a contradiction. Hence we have established that $N' = \mathrm{lfp}(\overline{\mathcal{S}'}^2, \emptyset)$. □

Proposition 8.31 *Let $\mathcal{S}$ be any non-monotonic system and $\mathcal{S}' \sqsubseteq \mathcal{S}$ its subsystem. If $B \subseteq \mathcal{L}^-(\mathcal{S})$ is a finished $\mathcal{S}$-bias and $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$, then $B' = B \cap \mathcal{L}^-(\mathcal{S})$ is a finished $\mathcal{S}'$-bias such that $N' = N \cap \mathcal{L}^-(\mathcal{S})$ holds for $N' = \mathrm{lfp}(\overline{\mathcal{S}'[B']}^2, \emptyset)$.*

Proof. Let $B \subseteq \mathcal{L}^-(\mathcal{S})$ be an $\mathcal{S}$-bias and $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$. By Definition 8.7, there is an enumeration ε of B which satisfies the bias condition for $\mathcal{S}$. Then define ε' as a projection of ε with respect to the language $\mathcal{L}^-(\mathcal{S}')$. Let f be a function from ordinals $0 \le \alpha \le \mathrm{Ind}(\varepsilon')$ to ordinals $0 \le \alpha \le \mathrm{Ind}(\varepsilon)$ defined as follows. First of all, we let $f(0) = 0$. If α is a successor ordinal, then $f(\alpha)$ gives the position of ε'_α in the enumeration ε, i.e. $\varepsilon'_\alpha = \varepsilon_{f(\alpha)}$. For limit ordinals α, the value $f(\alpha)$ is the limit of $f(\beta)$ where $\beta < \alpha$. Moreover, define $B_\alpha = \{\varepsilon_{0\ldots\alpha}\}$ and $N_\alpha = \mathrm{lfp}(\overline{\mathcal{S}[B_\alpha]}^2, \emptyset)$ for all ordinals $0 \le \alpha \le \mathrm{Ind}(\varepsilon)$ and $B'_\alpha = \{\varepsilon'_{0\ldots\alpha}\}$ and $N'_\alpha = \mathrm{lfp}(\overline{\mathcal{S}'[B'_\alpha]}^2, \emptyset)$ for all ordinals $0 \le \alpha \le \mathrm{Ind}(\varepsilon')$. Then it is not difficult to see that $B'_\alpha = B_{f(\alpha)} \cap \mathcal{L}^-(\mathcal{S}')$ for all ordinals $0 \le \alpha \le \mathrm{Ind}(\varepsilon')$.

Let α be any ordinal in the range $0 \le \alpha \le \mathrm{Ind}(\varepsilon')$. It follows that $\mathcal{S}'[B'_\alpha]$ is a subsystem of $\mathcal{S}[B_{f(\alpha)}]$ by Lemma 3.7. Then N'_α is obtained as the intersection

$N_\alpha \cap \mathcal{L}^-(\mathcal{S}')$ by Lemma 8.30. By the same arguments, we know that $\mathcal{S}'[B']$ is a subsystem of $\mathcal{S}[B]$ and N' is obtained as the intersection $N \cap \mathcal{L}^-(\mathcal{S}')$.

It is then verified that ε' satisfies the bias condition for $\mathcal{S}'$. So consider any successor ordinal α in the range $0 \leq \alpha \leq \overline{\mathrm{Ind}(\varepsilon')}$. Let us assume that ε'_α is not a bias for $\mathcal{S}'[B'_{\alpha-1}]$. It follows that $\varepsilon'_\alpha \not\in \overline{\mathcal{S}'[B'_{\alpha-1}]}(N'_{\alpha-1})$, i.e. $\varepsilon'_\alpha \in \mathcal{S}'[B'_{\alpha-1}](N'_{\alpha-1})$. Note that $B'_{\alpha-1} \cup N'_{\alpha-1} = (B_{f(\alpha-1)} \cup N_{f(\alpha-1)}) \cap \mathcal{L}^-(\mathcal{S}')$ by the above. Since $\mathcal{S}' \sqsubseteq \mathcal{S}$ and the sentence ε'_α belongs also to $\mathcal{L}^+(\mathcal{S}')$, we obtain $\varepsilon'_\alpha \in \mathcal{S}[B_{f(\alpha-1)}](N_{f(\alpha-1)})$. Because $f(\alpha-1) \leq f(\alpha)-1$ and B is an $\mathcal{S}$-bias, it is obvious that $B_{f(\alpha-1)} \subseteq B_{f(\alpha)-1}$ and $N_{f(\alpha-1)} \subseteq N_{f(\alpha)-1}$ by Proposition 8.17. Moreover, the sentence ε'_α is the sentence $\varepsilon_{f(\alpha)}$ by the definition of f. These imply that $\varepsilon_{f(\alpha)} \in \mathcal{S}[B_{f(\alpha)-1}](N_{f(\alpha)-1})$ by the requirement MON. Thus $\varepsilon_{f(\alpha)}$ is not a bias for $\mathcal{S}[B_{f(\alpha)-1}]$ and ε does not satisfy the bias condition for $\mathcal{S}$, a contradiction. Hence ε'_α is a bias for $\mathcal{S}'[B'_{\alpha-1}]$ and ε' satisfies the bias condition for $\mathcal{S}'$. The set of non-theorems $B \cup N$ is $\mathcal{S}$-complete, because B is a finished $\mathcal{S}$-bias. It follows by Proposition 6.3 that $(B \cup N) \cap \mathcal{L}^-(\mathcal{S}') = B' \cup N'$ is $\mathcal{S}'$-complete, since $\mathcal{S}' \sqsubseteq \mathcal{S}$. Thus B' is a finished $\mathcal{S}'$-bias. □

Lemma 8.32 *Let $\mathcal{S}$ be any non-monotonic system and $\mathcal{S}' \sqsubseteq \mathcal{S}$ its subsystem. If $N' = \mathrm{lfp}(\overline{\mathcal{S}'}^2, \emptyset)$, then $N' = N \cap \mathcal{L}^-(\mathcal{S}')$ holds for $N = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$.*

Proof. Let $N' = \mathrm{lfp}(\overline{\mathcal{S}'}^2, \emptyset)$. Since N' is $\mathcal{S}'$-stationary, we know by Proposition 6.12 that there is an $\mathcal{S}$-stationary set of non-theorems N such that $N' = N \cap \mathcal{L}^-(\mathcal{S}')$. Then assume that N is not the least $\mathcal{S}$-stationary set of non-theorems. Then $M = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$ is a proper subset of N. It follows that $M' = M \cap \mathcal{L}^-(\mathcal{S}')$ is the least $\mathcal{S}'$-stationary set of non-theorems by Lemma 8.30. Since the least $\mathcal{S}'$-stationary set of non-theorems is unique, it follows that $M' = N'$. It was shown in the proof of Proposition 6.12 that N is obtained as $\mathrm{lfp}(\overline{\mathcal{S}}^2, N')$. By Proposition 4.27, this implies that N is the least fixed point of $\overline{\mathcal{S}}^2$ satisfying $N' \subseteq N$. Since $N' = M'$ is contained in M, also M has this property. Then we obtain a contradiction $N = M$ by the uniqueness of least fixed points. Thus $N = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$ and $N' = N \cap \mathcal{L}(\mathcal{S}')$. □

Proposition 8.33 *Let $\mathcal{S}$ be any non-monotonic system and $\mathcal{S}' \sqsubseteq \mathcal{S}$ its subsystem. If $B' \subseteq \mathcal{L}^-(\mathcal{S}')$ is a finished $\mathcal{S}'$-bias such that $N' \subseteq B'$ for $N' = \mathrm{lfp}(\overline{\mathcal{S}'[B']}^2, \emptyset)$, then there is a finished $\mathcal{S}$-bias $B \subseteq \mathcal{L}^-(\mathcal{S})$ such that $B' = B \cap \mathcal{L}^-(\mathcal{S}')$ and $N' = N \cap \mathcal{L}^-(\mathcal{S}')$ holds for $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$.*

Proof. Let B' be a finished $\mathcal{S}'$-bias such that $N' \subseteq B'$ for $N' = \mathrm{lfp}(\overline{\mathcal{S}'[B']}^2, \emptyset)$. Then there is an enumeration ε' of B' that satisfies the bias condition for $\mathcal{S}'$. Let us define a bias $B'_\alpha = \{\varepsilon'_{0\ldots\alpha}\}$ and sets of non-theorems $N'_\alpha = \mathrm{lfp}(\overline{\mathcal{S}'[B'_\alpha]}^2, \emptyset)$ and $N_\alpha = \mathrm{lfp}(\overline{\mathcal{S}[B'_\alpha]}^2, \emptyset)$ for all ordinals α in the range $0 \leq$

$\alpha \leq \mathrm{Ind}(\varepsilon')$. It follows that $\mathcal{S}'[B'_\alpha] \sqsubseteq \mathcal{S}[B'_\alpha]$ and $N'_\alpha = N_\alpha \cap \mathcal{L}^-(\mathcal{S}')$ for any ordinal α in this range by Lemmas 3.7 and 8.32, respectively.

Let us then show that ε' satisfies the bias condition for $\mathcal{S}$ also. So let α be any successor ordinal such that $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon')$. Assume that ε'_α is not a bias for $\mathcal{S}[B'_{\alpha-1}]$. It follows that $\varepsilon'_\alpha \notin \overline{\mathcal{S}[B'_{\alpha-1}]}(N_{\alpha-1})$ for $N_{\alpha-1}$ defined above. Thus $\varepsilon'_\alpha \in \mathcal{S}[B'_{\alpha-1}](N_{\alpha-1})$. Since $\mathcal{S}'[B'_{\alpha-1}]$ is a subsystem of $\mathcal{S}[B'_{\alpha-1}]$, $N'_{\alpha-1} = N_{\alpha-1} \cap \mathcal{L}^-(\mathcal{S}')$ and also $\varepsilon'_\alpha \in \mathcal{L}^+(\mathcal{S}')$, we obtain $\varepsilon'_\alpha \in \mathcal{S}'[B'_{\alpha-1}](N'_{\alpha-1})$ by the definition of subsystems. Thus $\varepsilon'_\alpha \notin \overline{\mathcal{S}'[B'_{\alpha-1}]}(N'_{\alpha-1})$ indicating that ε'_α is not a bias for $\mathcal{S}'[B'_{\alpha-1}]$ and ε' does not satisfy the bias condition for $\mathcal{S}'$, a contradiction. Hence, the sentence ε'_α is a bias for $\mathcal{S}[B'_{\alpha-1}]$ and ε' satisfies the bias condition for $\mathcal{S}$.

We have therefore established that B' is also an $\mathcal{S}$-bias and $N' = N \cap \mathcal{L}^-(\mathcal{S}')$ holds for the resulting fixed point $N = \mathrm{lfp}(\overline{\mathcal{S}[B']}^2, \emptyset)$. There is no guarantee that the bias B' is finished, so we extend B' to a finished $\mathcal{S}$-bias B by Theorem 8.14. Let M be the resulting fixed point $\mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$. Let us then assume that $B' \neq B \cap \mathcal{L}^-(\mathcal{S}')$. Since $B' \subseteq B$, there must be $\phi \in \mathcal{L}^-(\mathcal{S}')$ such that $\phi \notin B'$ but $\phi \in B$, i.e. $\phi \in B - B'$. Since $B - B'$ is an $\mathcal{S}[B']$-bias, we know by Proposition 8.17 that $\phi \in \overline{\mathcal{S}[B']}(N)$ for the set of non-theorems N defined above. Because $\mathcal{S}'[B'] \sqsubseteq \mathcal{S}[B']$ by Lemma 3.7, $N' = N \cap \mathcal{L}^-(\mathcal{S}')$ and also $\phi \in \mathcal{L}^-(\mathcal{S}')$, it follows that $\phi \in \overline{\mathcal{S}'[B']}(N')$ by Lemma 6.1. Since $N' = \overline{\mathcal{S}'[B']}(N')$ by Proposition 8.15, we obtain $\phi \in N'$. Then our requirement $N' \subseteq B'$ implies $\phi \in B'$, a contradiction. Hence $B' = B \cap \mathcal{L}^-(\mathcal{S}')$. Now $\mathcal{S}'[B'] \sqsubseteq \mathcal{S}[B]$ by Lemma 3.7 so that $N' = M \cap \mathcal{L}^-(\mathcal{S}')$ by Lemma 8.32. □

Propositions 8.31 and 8.33 allow us to prove the following theorem:

Theorem 8.34 *For $\mathcal{S} \sqsubseteq \mathcal{S}'$ and for all $T \subseteq \mathcal{L}^+(\mathcal{S})$, the cautious semantics* $\mathrm{CAUT}_{\mathcal{S}}(T) = \mathrm{CAUT}_{\mathcal{S}'}(T) \cap \mathcal{L}^+(\mathcal{S})$.

Proof. Let $\mathcal{S} \sqsubseteq \mathcal{S}'$ and $T \subseteq \mathcal{L}^+(\mathcal{S})$ so that $\mathcal{S}[T] \sqsubseteq \mathcal{S}'[T]$ holds by Lemma 3.7. Then consider any $\phi \in \mathcal{L}^+(\mathcal{S})$. ($\subseteq$) Let $\phi \notin \mathrm{CAUT}_{\mathcal{S}'}(T)$ hold, i.e. there is a finished $\mathcal{S}'[T]$-bias B' such that $\phi \notin \mathcal{S}'[T](N')$ for $N' = \mathrm{lfp}(\overline{\mathcal{S}'[T;B']}^2, \emptyset)$. Since $\mathcal{S}[T] \sqsubseteq \mathcal{S}'[T]$ holds, Proposition 8.31 implies that $B = B' \cap \mathcal{L}^-(\mathcal{S})$ is a finished $\mathcal{S}[T]$-bias B such that $N = N' \cap \mathcal{L}^-(\mathcal{S})$ holds for $N = \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$. It follows that $\phi \notin \mathcal{S}[T](N)$ by the definition of subsystems. Thus $\phi \notin \mathrm{CAUT}_{\mathcal{S}}(T)$. ($\supseteq$) Let $\phi \notin \mathrm{CAUT}_{\mathcal{S}}(T)$ hold, i.e. there is a finished $\mathcal{S}[T]$-bias B such that $\phi \notin \mathcal{S}[T](N)$ for $N = \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$. By Proposition 8.18, we may assume without loss of generality that N is contained in the bias B. By Proposition 8.33 there is a finished $\mathcal{S}'[T]$-bias B' such that $B = B' \cap \mathcal{L}^-(\mathcal{S})$ and $N = N' \cap \mathcal{L}^-(\mathcal{S})$ holds for $N' = \mathrm{lfp}(\overline{\mathcal{S}[T;B']}^2, \emptyset)$. It follows also that $\phi \notin \mathcal{S}'[T](N')$ by the definition of subsystems. Thus $\phi \notin \mathrm{CAUT}_{\mathcal{S}'}(T)$ holds by the definition of cautious semantics. □

The analysis of functional equivalences follow. The same strategy which was applied to subsystems is used: it is first shown in Lemma 8.35 that functional equivalences preserve the least fixed points of $\overline{\mathcal{S}}^2$ and then biases are taken into consideration in Proposition 8.36.

Lemma 8.35 *Let $\mathcal{S} \sim \mathcal{S}'$ via $f = \langle f^+, f^- \rangle$. If $N = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$, then $N' = f^-(N) = \mathrm{lfp}(\overline{\mathcal{S}'}^2, \emptyset)$.*

Proof. Let $N = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$. Since N is $\mathcal{S}$-stationary, it follows that $N' = f^-(N)$ is $\mathcal{S}'$-stationary by Proposition 6.14. Thus $M' = \mathrm{lfp}(\overline{\mathcal{S}'}^2, \emptyset) \subseteq N'$ by Proposition 4.27. Then assume that $M' \subset N'$. It follows that $M = (f^-)^{-1}(M')$ is $\mathcal{S}$-stationary by Proposition 6.14 and $M \subset N$, a contradiction, since N is the least $\mathcal{S}$-stationary set of non-theorems. Hence $M' = N'$. □

Proposition 8.36 *Let $\mathcal{S} \sim \mathcal{S}'$ via $f = \langle f^+, f^- \rangle$. If $B \subseteq \mathcal{L}^-(\mathcal{S})$ is a finished $\mathcal{S}$-bias and $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$, then $B' = f^-(B)$ is a finished $\mathcal{S}'$-bias such that $N' = f^-(N) = \mathrm{lfp}(\overline{\mathcal{S}'[B']}^2, \emptyset)$.*

Proof. Let B be a finished $\mathcal{S}$-bias and $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$. Then there is an enumeration ε of B which satisfies the bias condition for $\mathcal{S}$. So let us define the biases $B_\alpha = \{\varepsilon_{0\ldots\alpha}\}$ and $B'_\alpha = f^-(B_\alpha)$ and the sets of non-theorems $N_\alpha = \mathrm{lfp}(\overline{\mathcal{S}[B_\alpha]}^2, \emptyset)$ and $N'_\alpha = f^-(N_\alpha)$ for all ordinals α in the range $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon)$. It follows that $\mathcal{S}[B_\alpha] \sim \mathcal{S}'[B'_\alpha]$ via f and $N'_\alpha = \mathrm{lfp}(\overline{\mathcal{S}'[B'_\alpha]}^2, \emptyset)$ by Lemmas 3.23 and 8.35, respectively.

It is then verified that the obvious enumeration $\varepsilon' = f^-(\varepsilon)$ of $B' = f^-(B)$ satisfies the bias condition for $\mathcal{S}'$. So consider any successor ordinal α in the range $0 \leq \alpha \leq \mathrm{Ind}(\varepsilon') = \mathrm{Ind}(\varepsilon)$. Now ε_α is a bias for $\mathcal{S}[B_\alpha]$, i.e. $\varepsilon_\alpha \in \overline{\mathcal{S}[B_\alpha]}(N_\alpha)$. It follows by Proposition 6.14 that $\varepsilon'_\alpha \in \overline{\mathcal{S}'[B'_\alpha]}(N'_\alpha)$. Thus ε'_α is a bias for $\mathcal{S}'[B'_\alpha]$. Therefore ε' satisfies the bias condition for $\mathcal{S}'$ and B' is an $\mathcal{S}'$-bias such that $N' = f^-(N) = \mathrm{lfp}(\overline{\mathcal{S}'[B']}^2, \emptyset)$. Moreover, the $\mathcal{S}$-bias B is finished, so that the set of non-theorems $B \cup N$ is $\mathcal{S}$-complete. It follows by Proposition 6.14 that the set of non-theorems $f^-(B \cup N) = B' \cup N'$ is $\mathcal{S}'$-complete. Hence, the $\mathcal{S}'$-bias B' is also finished. □

Theorem 8.37 *For $\mathcal{S} \sim \mathcal{S}'$ via $f = \langle f^+, f^- \rangle$ and for all $T \subseteq \mathcal{L}^+(\mathcal{S})$, the cautious semantics $f^+(\mathrm{CAUT}_{\mathcal{S}}(T)) = \mathrm{CAUT}_{\mathcal{S}'}(f^+(T))$.*

Proof sketch. The proof is obtained from that of Theorem 6.19 by (i) replacing stable sets of non-theorems with the least stationary sets of non-theorems induced by finished biases and (ii) by using Proposition 8.36 instead of Proposition 6.14. Note that the symmetry of $\sim$ can be utilized like before. □

The results deduced for subsystems and functional equivalence can now be modularly combined for embeddings. Finished biases are considered in Corollaries 8.38 and 8.39 while Corollary 8.40 addresses cautious semantics.

Corollary 8.38 *Let $\mathcal{S}' \mapsto_f \mathcal{S}$ hold for $f = \langle f^+, f^- \rangle$. If $B \subseteq \mathcal{L}^-(\mathcal{S})$ is a finished $\mathcal{S}$-bias and $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$, then $B' = (f^-)^{-1}(B \cap f^-(\mathcal{L}^-(\mathcal{S}')))$ is a finished $\mathcal{S}'$-bias such that*

$$N' = \mathrm{lfp}(\overline{\mathcal{S}'[B']}^2, \emptyset) = (f^-)^{-1}(N \cap f^-(\mathcal{L}^-(\mathcal{S}'))).$$

Proof. This is a corollary of Propositions 8.31 and 8.36. □

Corollary 8.39 *Let $\mathcal{S}' \mapsto_f \mathcal{S}$ hold for $f = \langle f^+, f^- \rangle$. If $B' \subseteq \mathcal{L}^-(\mathcal{S}')$ is a finished $\mathcal{S}'$-bias such that $N' \subseteq B'$ for $N' = \mathrm{lfp}(\overline{\mathcal{S}[B']}^2, \emptyset)$, then there is a finished $\mathcal{S}$-bias $B \subseteq \mathcal{L}^-(\mathcal{S})$ such that $B' = (f^-)^{-1}(B \cap f^-(\mathcal{L}^-(\mathcal{S}')))$ and $N' = (f^-)^{-1}(N \cap f^-(\mathcal{L}^-(\mathcal{S}')))$ holds for $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$.*

Proof. This is a corollary of Propositions 8.33 and 8.36. □

Corollary 8.40 *If $\mathcal{S} \mapsto_f \mathcal{S}'$ for $f = \langle f^+, f^- \rangle$, then the cautious semantics $f^+(\mathrm{CAUT}_{\mathcal{S}}(T)) = \mathrm{CAUT}_{\mathcal{S}'}(f^+(T)) \cap f^+(\mathcal{L}^+(\mathcal{S}))$ for all $T \subseteq \mathcal{L}^+(\mathcal{S})$.*

In Section 6, we specialized Corollary 6.21 for stable semantics and stationary semantics as well as the various translation functions that were introduced in Section 3.3. The following three corollaries of Corollary 8.40 do a similar specialization for cautious semantics.

Corollary 8.41 *Let $P \subseteq \mathcal{A}$ be a set of atoms to be minimized. For all $T \subseteq \mathcal{L}$,*

$$\mathrm{CAUT}_{\mathcal{C}_P}(T) = \mathrm{CAUT}_{\mathcal{D}_{\mathrm{Tr}_{\mathrm{E}}(P)}}(T).$$

Corollary 8.42 *Let D be a set of defaults. For all $T \subseteq \mathcal{L}$,*

$$\mathrm{CAUT}_{\mathcal{D}_D}(T) = \mathrm{CAUT}_{\mathcal{A}_{\mathrm{Tr}_{\mathrm{K}}(D)}}(T) \cap \mathcal{L}.$$

Corollary 8.43 *Let Σ be an autoepistemic theory. For all $T \subseteq \mathcal{L}_{\mathrm{BD}}$,*

$$\mathrm{CAUT}_{\mathcal{A}_\Sigma}(T) = \mathrm{CAUT}_{\mathcal{D}_{\mathrm{Tr}_{\mathrm{J}}(\Sigma)}}(T).$$

8.4 Comparison with Other Approaches

Numerous semantics in the gap between stable semantics and stationary semantics have been proposed [3, 12, 19, 32, 44, 88, 115, 118]. In particular, the field of logic programming has been very productive in this respect. This is mainly due to different classes of logic programs that have been systematically investigated. In this section, we take a look at the other semantics (than stable and stationary semantics) to compare them with cautious semantics, but restrict our comparison to semantics that have mainly arisen in the context of normal logic programs (other classes of logic programs will be considered later in Section 10). This is because many of these semantics are easily generalized for non-monotonic systems, allowing a comparison with cautious semantics on this level of abstraction. First of all, we consider semantics that are cumulatively closed but not cumulatively monotonic. *Regular semantics* by You and Yuan [119] is analyzed in Section 8.4.1. *Stable class semantics* by Baral and Subrahmanian [3] and its variant *stationary class semantics* are addressed in Section 8.4.2. Then we concentrate on semantics that are cumulative, but relatively weak. We study *codiagonal semantics* by Thiele [110] in Section 8.4.3. The semantics WFS^+ and WFS' proposed by Dix [19] are addressed in Section 8.4.4. Finally, we provide a brief comparison with O-semantics by Pereira, Alferes and Aparício [88] and a framework by Brewka and Gottlob [12] that aims at refining the stationary semantics of default logic.

8.4.1 Regular Semantics

You and Yuan [119] have proposed *regular semantics* that is based on maximal fixed points of $\overline{\mathcal{S}}^2$. As shown by You and Yuan [118], the same idea has been independently embodied in Saccà and Zaniolo's *partial stable model semantics* [97] and Dung's *preferential semantics* [23]. We lift the same idea to the level of non-monotonic systems in Definition 8.44 by introducing the notion of an $\mathcal{S}$-regular set of non-theorems. Note that for each non-monotonicity system $\mathcal{S}$ there is at least one $\mathcal{S}$-regular set of non-theorems, since there is at least one $\mathcal{S}$-stationary set of non-theorems. The definition of the semantic operator $\mathrm{REG}_{\mathcal{S}}$ is analogous to that of $\mathrm{STAT}_{\mathcal{S}}$ (cf. Definition 4.19) but $\mathcal{S}$-regular sets of non-theorems are used instead of $\mathcal{S}$-stationary sets of non-theorems.

Definition 8.44 *A set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}$-regular if and only if N is $\mathcal{S}$-stationary and there is no $\mathcal{S}$-stationary $N' \subseteq \mathcal{L}^-(\mathcal{S})$ such that $N \subset N'$.*

It is relatively easy to see that regular semantics satisfies $\mathrm{STAT}_{\mathcal{S}} \sqsubseteq \mathrm{REG}_{\mathcal{S}} \sqsubseteq \mathrm{STAB}_{\mathcal{S}}$, since $\mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$ is contained in every $\mathcal{S}$-regular set of non-theorems and every $\mathcal{S}$-stable set of non-theorems is also $\mathcal{S}$-regular. However, regular semantics is $\sqsubseteq$-incomparable with the cautious semantics. This is easily inspected from Examples 8.29 and 8.45. In the former example, there are two $\mathcal{S}$-stationary sets of non-theorems $N_1 = \emptyset$ and $N_2 = \{\neg\neg\mathrm{b}\}$ so that N_2 is $\mathcal{S}$-regular. This implies $\mathrm{CAUT}_{\mathcal{S}}(T) \subset \mathrm{REG}_{\mathcal{S}}(T) = \mathrm{Cn}(\{\mathrm{a}, \mathrm{c}\})$ holds for $T = \emptyset$. On the other hand, only $N = \emptyset$ is $\mathcal{S}$-stationary in the latter example (see below). Thus $\mathrm{REG}_{\mathcal{S}}(T) = \emptyset$ for $T = \emptyset$. In contrast to this, cautious semantics provides the conclusion s on the basis of three finished $\mathcal{S}[T]$-biases pointed out below. This implies that $\mathrm{REG}_{\mathcal{S}}(T) \subset \mathrm{CAUT}_{\mathcal{S}}(T)$ for $T = \emptyset$.

Example 8.45 Consider an autoepistemic theory

$$\Sigma = \{\mathbf{D}\mathrm{p} \rightarrow \mathrm{q}, \mathbf{D}\mathrm{q} \rightarrow \mathrm{r}, \mathbf{D}\mathrm{r} \rightarrow \mathrm{p}, \mathbf{B}\mathrm{p} \rightarrow \mathrm{s}, \mathbf{B}\mathrm{q} \rightarrow \mathrm{s}, \mathbf{B}\mathrm{r} \rightarrow \mathrm{s}\}$$

and the subsystem $\mathcal{S} = \boldsymbol{\mathcal{A}}_{\Sigma}^{\Sigma\cup\{\mathbf{B}\mathrm{s}\}}$ having the language $\langle\{\mathrm{p}, \mathrm{q}, \mathrm{r}, \mathrm{s}\}, \{\mathrm{p}, \mathrm{q}, \mathrm{r}\}\rangle$. The $\mathcal{S}$-biases $B_1 = \{\mathrm{p}\}$, $B_2 = \{\mathrm{q}\}$ and $B_3 = \{\mathrm{r}\}$ for $\mathcal{S}$ are all finished and give rise to the sets of non-theorems $N_1 = \{\mathrm{r}\}$, $N_2 = \{\mathrm{p}\}$ and $N_3 = \{\mathrm{q}\}$, respectively. Note that s belongs to $\mathcal{S}(N_i)$ for each $i \in \{1, 2, 3\}$. Note also that in this particular example, a sentence in a bias B_i ends up in the set of theorems $\mathcal{S}(N_i)$ for the fixed point N_i induced by B_i. For instance, the sentence p is included in B_1 as well as $\mathcal{S}(N_1)$. □

In contrast to cautious semantics, the regular semantics $\mathrm{REG}_{\mathcal{S}}$ does not satisfy the cumulative monotonicity requirement CMON. This is evident by our discussion on Example 8.29. Nevertheless, regular semantics satisfies the cumulative closure requirement CCLO as established by Dix [19] in the context of normal logic programs.

8.4.2 Stable and Stationary Class Semantics

Stable class semantics by Baral and Subrahmanian [3] involves a stability condition for *classes* of sets of theorems in contrast to stable semantics [32] where a single set of theorems is subject to a stability condition. The notion of stable classes generalises for arbitrary non-monotonic systems $\mathcal{S}$ and sets of non-theorems as follows.

Definition 8.46 *A class $\mathbf{C}$ of sets of non-theorems $N \subseteq \mathcal{L}^{-}(\mathcal{S})$ is $\mathcal{S}$-stable if and only if every $N \in \mathbf{C}$ satisfies $\overline{\mathcal{S}}(N) \in \mathbf{C}$.*

An $\mathcal{S}$-stable class $\mathbf{C}$ is *strict* if there is no non-empty $\mathcal{S}$-stable class $\mathbf{C}'$ such that $\mathbf{C}' \subset \mathbf{C}$. Baral and Subrahmanian impose a point-wise ordering on non-empty and strict $\mathcal{S}$-stable classes as follows: $\mathbf{C} \preceq \mathbf{C}'$ if and only if for each $N \in \mathbf{C}$ there is $M \in \mathbf{C}'$ such that $N \subseteq M$. Stable class semantics considers only non-empty and strict $\mathcal{S}$-stable classes $\mathbf{C}$ which are minimal with respect to this ordering. The set of theorems $\mathrm{STABCL}_{\mathbb{S}}(T)$ is defined as the intersection of all $\mathbb{S}(T,N)$ for all N belonging to the $\preceq$-minimal non-empty and strict $\mathbb{S}[T]$-stable classes $\mathbf{C}$. It is easy to see that an $\mathbb{S}[T]$-stable set of non-theorems N gives rise to a non-empty strict $\mathbb{S}[T]$-stable class $\mathbf{C} = \{N\}$. The same applies to the least fixed point $M = \mathrm{lfp}(\overline{\mathbb{S}[T]}^2, \emptyset)$ which induces a non-empty strict $\mathbb{S}[T]$-stable class $\mathbf{C} = \{M, \overline{\mathbb{S}[T]}(M)\}$ which is the greatest class relative to the ordering $\preceq$. Therefore, it is straightforward to show that the operator $\mathrm{STABCL}_{\mathbb{S}}$ satisfies $\mathrm{STAT}_{\mathbb{S}} \sqsubseteq \mathrm{STABCL}_{\mathbb{S}} \sqsubseteq \mathrm{STAB}_{\mathbb{S}}$.

Let us now reconsider our examples to investigate the relationship of stable class semantics to other semantics. We have depicted the relevant sets of non-theorems as well as the applications of the operator $\overline{\mathbb{S}}$ for Example 8.29 in Figure 9 (the diagram on the left). The two non-empty and strict $\mathbb{S}$-classes are easily read from the figure: $\mathbf{C}_1 = \{\emptyset, \{\neg\neg\mathrm{a}, \neg\neg\mathrm{b}\}\}$ and $\mathbf{C}_2 = \{\{\neg\neg\mathrm{b}\}\}$. The latter is $\preceq$-minimal and determines the stable class semantics of $\mathbb{S}$. Thus it holds that $\mathrm{CAUT}_{\mathbb{S}}(T) \subset \mathrm{STABCL}_{\mathbb{S}}(T)$ results for the set of sentences $T = \emptyset$. Moreover, this example serves as a counter example for the cumulativity of stable class semantics by arguments analogous to those presented in Example 8.29. The other diagram in Figure 9 is for Example 8.45. Again, there are two non-empty and strict $\mathbb{S}$-stable classes $\mathbf{C}_1 = \{\emptyset, \{\mathrm{p}, \mathrm{q}, \mathrm{r}\}\}$ and $\mathbf{C}_2 = \{\{\mathrm{p}\}, \{\mathrm{q}\}, \{\mathrm{r}\}, \{\mathrm{p}, \mathrm{q}\}, \{\mathrm{p}, \mathrm{r}\}, \{\mathrm{q}, \mathrm{r}\}\}$. Now $\mathbf{C}_2$ is $\preceq$-minimal. This implies that $\mathrm{s} \in \mathrm{STABCL}_{\mathbb{S}}(T)$ when $T = \emptyset$. Thus $\mathrm{REG}_{\mathbb{S}}(T) \subset \mathrm{STABCL}_{\mathbb{S}}(T)$ in this particular example. This indicates that higher degrees of the operator $\overline{\mathbb{S}}$ can be used to refine a semantics: note that the members of $\mathbf{C}_2$ are stable under six subsequent applications of the operator $\overline{\mathbb{S}}$, i.e. the operator $\overline{\mathbb{S}}^6$.

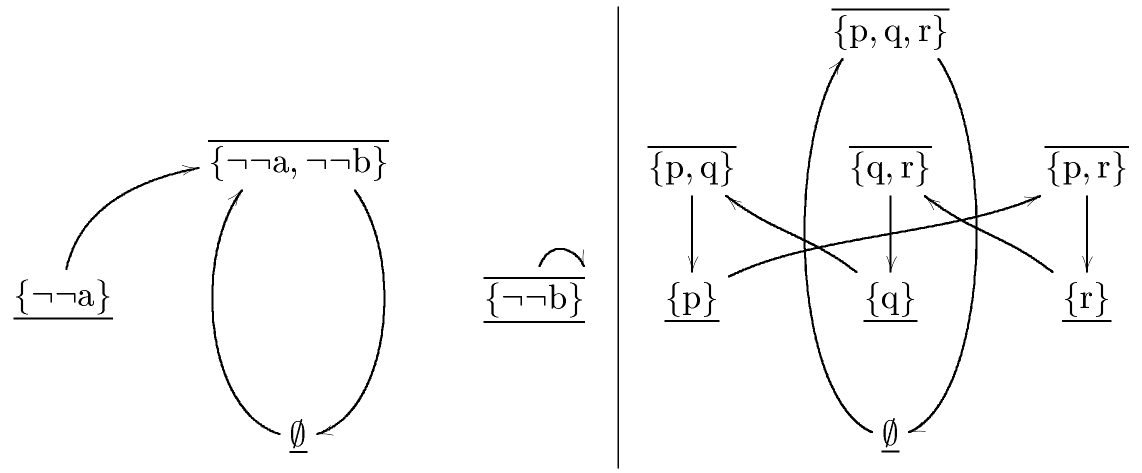


Figure 9: Sets of non-theorems in Examples 8.29 and 8.45

In Example 8.47, we present yet another non-monotonic system $\mathbb{S}$ for which a diagram is given in Figure 10. There are no $\mathbb{S}$-stable sets of non-theorems for this system and $N = \emptyset$ is the only $\mathbb{S}$-stationary and $\mathbb{S}$-regular set of non-theorems so that $\mathbf{C} = \{\emptyset, \{\neg\neg\mathrm{a}, \neg\neg\mathrm{b}, \neg\neg c\}\}$ is the only non-empty and strict

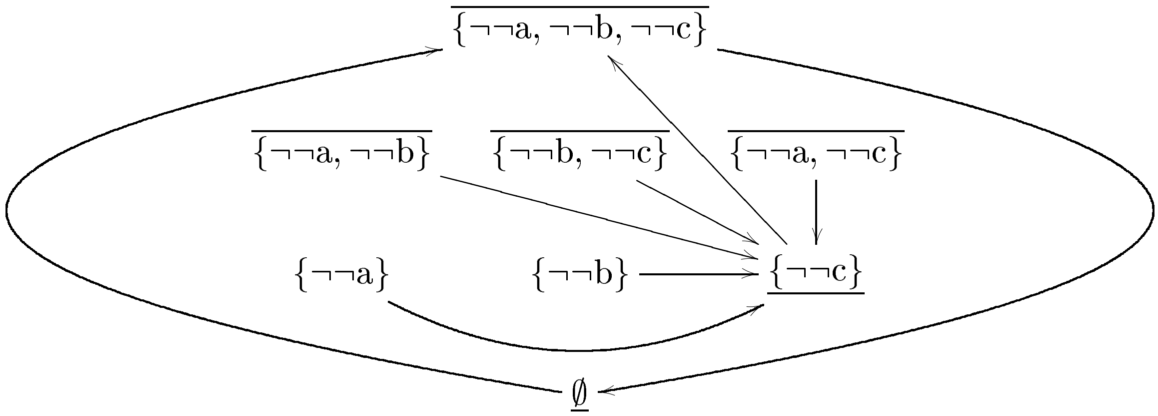


Figure 10: Sets of non-theorems in Example 8.47

$\mathcal{S}$-stable class. Therefore, for $T = \emptyset$, the semantics $\mathrm{STAT}_{\mathcal{S}}(T) = \mathrm{REG}_{\mathcal{S}}(T) = \mathrm{STABCL}_{\mathcal{S}}(T) = \mathrm{Cn}(\emptyset)$ coincide and $\mathrm{STAB}_{\mathcal{S}}(T) = \mathcal{L}^+(\mathcal{S}) = \mathcal{L}$. In contrast to all these semantics, cautious semantics gives us $\mathrm{CAUT}_{\mathcal{S}}(T) = \mathrm{Cn}(\{\mathrm{d}\})$. This indicates that cautious semantics performs case analysis which is not achieved by semantics based on fixed points (or fixed classes of points) of $\overline{\mathcal{S}}$. In addition, it holds that $\mathrm{STABCL}_{\mathcal{S}}(T) \subset \mathrm{CAUT}_{\mathcal{S}}(T)$ for $T = \emptyset$. Thus we conclude upon the examples considered so far that stable class semantics is not $\sqsubseteq$-comparable with cautious semantics.

Example 8.47 Consider a set of defaults $D = \{\frac{:\neg \mathrm{a}}{\mathrm{a} \wedge \mathrm{b}}, \frac{:\neg \mathrm{b}}{\mathrm{b} \wedge \mathrm{a}}, \frac{:\neg \mathrm{a}, \neg \mathrm{b}}{\mathrm{c}}, \frac{:\neg \mathrm{c}}{d}\}$ and the subsystem $\mathcal{S} = \mathfrak{D}_D^{\neg \mathrm{J}(D)}$ which has the language $\langle \mathcal{L}, \{\neg\neg \mathrm{a}, \neg\neg \mathrm{b}, \neg\neg \mathrm{c}\}\rangle$.

There are essentially two finished $\mathcal{S}$-biases, $B_1 = \{\neg\neg \mathrm{a}\}$ and $B_2 = \{\neg\neg \mathrm{b}\}$ that both give rise to the least fixed point $N = \{\neg\neg \mathrm{c}\}$. Thus d can be concluded under cautious semantics. On the other hand, there is only one non-empty and strict $\mathcal{S}$-stable class, namely $\{\emptyset, \{\neg\neg \mathrm{a}, \neg\neg \mathrm{b}, \neg\neg \mathrm{c}\}\}$. This is determined by $\mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset) = \emptyset$ so that stable class semantics collapses to stationary semantics. Thus $\mathrm{STABCL}_{\mathcal{S}}(T) \subset \mathrm{CAUT}_{\mathcal{S}}(T)$ for $T = \emptyset$. □

We noted already earlier that Example 8.29 serves as a counter-example showing that stable class semantics does not satisfy CMON. Nevertheless, Proposition 5.12 can be generalized for stable classes to show that stable class semantics satisfies CCLO. Finally, we consider the effects of improving stable class semantics to *stationary class semantics*. As motivation we present an example due to Yuan [120] which illustrates a weakness of stable class semantics. The diagram for this example is given in Figure 11.

Example 8.48 Consider an autoepistemic theory

$$\Sigma = \{\mathbf{D}\mathrm{a} \to \mathrm{a}, \mathbf{D}\mathrm{b} \to \mathrm{b}, \mathbf{D}\mathrm{a} \to c, \mathbf{D}\mathrm{b} \to c\}$$

and the subsystem $\mathcal{S} = \mathcal{A}_{\Sigma}^{\Sigma \cup \{\mathbf{B}\mathrm{c}\}}$ which has the language $\langle \{\mathrm{a}, \mathrm{b}, \mathrm{c}\}, \{\mathrm{a}, \mathrm{b}\}\rangle$.

There are two non-empty and strict $\mathcal{S}$-stable classes, namely $\mathbf{C}_1 = \{\emptyset, \{\mathrm{a}, \mathrm{b}\}\}$ and $\mathbf{C}_2 = \{\{\mathrm{a}\}, \{\mathrm{b}\}\}$. The latter of these is $\preceq$-minimal so that c is included in $\mathrm{STABCL}_{\mathcal{S}}(T)$ when $T = \emptyset$. But the conclusion c is drawn from the sets of non-theorems $N_1 = \{\mathrm{a}\}$ and $N_2 = \{\mathrm{b}\}$ none of which is $\mathcal{S}$-consistent! □

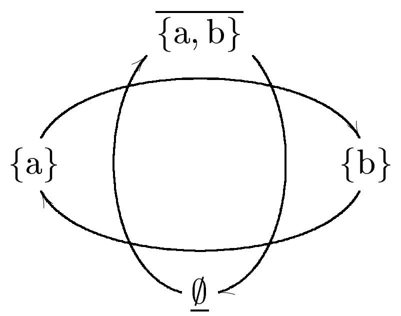


Figure 11: Sets of non-theorems in Example 8.48

The above example suggests that only $\mathcal{S}$-consistent sets of non-theorems should be taken into account. By inspecting Figure 9 once more we note that $\mathcal{S}$-stable classes may include both $\mathcal{S}$-consistent and $\mathcal{S}$-complete sets of non-theorems. On the other hand, we know by Lemma 4.21 that the operator $\overline{\mathcal{S}}$ maps an $\mathcal{S}$-consistent set of non-theorems into an $\mathcal{S}$-complete one and vice versa. Therefore, to exclude sets of non-theorems that are not $\mathcal{S}$-consistent, we have to formulate the stability condition in terms of $\overline{\mathcal{S}}^2$ rather than $\overline{\mathcal{S}}$. These observations lead to the following notion of a *stationary class*.

Definition 8.49 *A class* $\mathbf{C}$ *of sets of non-theorems* $N \subseteq \mathcal{L}^-(\mathcal{S})$ *is* $\mathcal{S}$*-stationary if and only if every* $N \in \mathbf{C}$ *satisfies (i)* $\overline{\mathcal{S}}^2(N) \in \mathbf{C}$ *and (ii)* $N \subseteq \overline{\mathcal{S}}(N)$.

The definition of strictness need not to be modified. The ordering among non-empty and strict $\mathcal{S}$-stationary classes becomes more natural, since only $\mathcal{S}$-consistent sets of non-theorems are contained in the classes. We call a semantics which takes only the $\preceq$-maximal non-empty and strict $\mathcal{S}$-stationary classes into account *stationary class semantics* $\mathrm{STATCL}_{\mathcal{S}}$. Looking back to Example 8.48 in Figure 11, there is only one non-empty and strict $\mathcal{S}$-stationary class, namely $\mathbf{C} = \{\emptyset\}$ so that the problem pointed out by Yuan is avoided.

It is not difficult to show that each set of non-theorems N in an $\mathcal{S}$-stationary class contains $\mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$, and for each $\mathcal{S}$-stable set of non-theorems N, the class $\mathbf{C} = \{N\}$ is non-empty, strict, $\mathcal{S}$-stationary and maximal with respect to $\preceq$. Consequently, stationary class semantics satisfies $\mathrm{STAT}_{\mathcal{S}} \sqsubseteq \mathrm{STATCL}_{\mathcal{S}} \sqsubseteq \mathrm{STAB}_{\mathcal{S}}$. The modified semantics is $\sqsubseteq$-incomparable with cautious semantics by the same examples used for stable class semantics. Stationary class semantics is not cumulative: the requirement CCLO is met, but Example 8.29 applies to the modified semantics so that CMON is not satisfied.

8.4.3 Codiagonal Semantics

Thiele [110] proposes *codiagonal semantics* by loosening the stability condition of stable semantics. The idea is that an arbitrary set of theorems $T \subseteq \mathcal{L}^+(\mathcal{S})$ may be introduced to stabilize a set of non-theorems N. This idea is made precise in Definition 8.50. To emphasize the role of $T \subseteq \mathcal{L}^+(\mathcal{S})$ in the definition, we say that N is $\mathcal{S}$-codiagonal by T. Codiagonal semantics $\text{CODIAG}_{\mathcal{S}}(T)$ is then defined as the intersection of the sets of theorems $\mathcal{S}[T \cup T'](N)$ where the set of non-theorems N is $\mathcal{S}[T]$-codiagonal by T'.

Definition 8.50 *A set of non-theorems $N \subseteq \mathcal{L}^-(\mathcal{S})$ is $\mathcal{S}$-codiagonal if and only if N is $\mathcal{S}[T]$-stable for some $T \subseteq \mathcal{L}^+(\mathcal{S})$.*

Thiele's original definition [110] requires that N is the relative complement $\mathcal{L}^-(\mathcal{S}) - T$ in the fixed point equation $T = \mathcal{S}[T](N)$. Thiele assumes also that $\mathcal{L}^-(\mathcal{S}) = \mathcal{L}^+(\mathcal{S})$, but we may drop this restriction. It is not difficult to see that a pair $\langle T, N \rangle$ satisfying these criteria consists of a set of non-theorems N which is $\mathcal{S}$-codiagonal by T and vice versa. A problem of Thiele's approach is that the resulting semantics is not even a refinement of stationary semantics and thus very weak. This phenomenon is demonstrated in Example 8.51.

Example 8.51 Consider an autoepistemic theory $\Sigma = \{\mathbf{D}\text{p} \to \text{q}, \mathbf{D}\text{q} \to \text{r}\}$ and the subsystem $\mathcal{S} = \mathcal{A}_{\Sigma}^{\Sigma \cup \{\mathbf{B}\text{r}\}}$ with the language $\langle \{\text{p}, \text{q}, \text{r}\}, \{\text{p}, \text{q}\} \rangle$.

The least $\mathcal{S}$-stationary set of non-theorems is $N_1 = \{\text{p}\}$ which gives rise to the set of theorems $\mathcal{S}(N) = \{\text{q}\}$. Note that N_1 is also $\mathcal{S}$-stable and thus $\mathcal{S}$-codiagonal by $T_1 = \emptyset$. However, also $N_2 = \{\text{q}\}$ and $N_3 = \emptyset$ are $\mathcal{S}$-codiagonal by $T_2 = \{\text{p}\}$ and $T_3 = \{\text{p}, \text{q}\}$, respectively. Thus $\mathcal{S}$-codiagonal sets of non-theorems give rise to sets of theorems $\mathcal{S}[T_1](N_1) = \{\text{q}\}$, $\mathcal{S}[T_2](N_2) = \{\text{p}, \text{r}\}$ and $\mathcal{S}[T_3](N_3) = \{\text{p}, \text{q}\}$. The intersection of these is empty which implies that the codiagonal semantics $\text{CODIAG}_{\mathcal{S}}(T) = \emptyset$ for $T = \emptyset$. Note that the cautious semantics $\text{CAUT}_{\mathcal{S}}(T) = \text{STAT}_{\mathcal{S}}(T) = \text{STAB}_{\mathcal{S}}(T) = \{\text{q}\}$ and also $\text{REG}_{\mathcal{S}}$, $\text{STABCL}_{\mathcal{S}}$ and $\text{STATCL}_{\mathcal{S}}$ produce the same set of theorems. □

In contrast to proposals considered so far, codiagonal semantics is cumulative. The cumulative closure property CCLO follows directly from Proposition 5.12. The cumulative monotonicity holds, since a set of non-theorems N that is $\mathcal{S}[T]$-codiagonal by T' is directly $\mathcal{S}$-codiagonal by $T \cup T'$ (in this argument, the set of theorems T is contained in every set of theorems $\mathcal{S}[T'](N)$ induced by a set of non-theorems $\mathcal{S}$-codiagonal by T'). Finally, we refer back to Theorem 8.52 to relate the fixed points $N = \text{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$ induced by finished $\mathcal{S}$-biases B for $\mathcal{S}$ with the codiagonal approach. Such a set of non-theorems is $\mathcal{S}$-codiagonal by the set of theorems $T = \mathcal{L}^-(\mathcal{S}) - (N \cup \mathcal{S}(N))$.

Theorem 8.52 *If B is a finished $\mathcal{S}$-bias and $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$, then N is $\mathcal{S}[T]$-stable where $T = \mathcal{L}^-(\mathcal{S}) - (N \cup \mathcal{S}(N))$.*

Proof. Let us consider $\phi \in T$ at first. It follows that $\phi \notin N$ by the definition of T. Recall that $N = \overline{\mathcal{S}[B]}(N)$ by Proposition 8.15. Thus $\phi \notin \overline{\mathcal{S}[B]}(N)$, i.e. $\phi \in \mathcal{S}[B](N)$. Hence $T \subseteq \mathcal{S}[B](N)$. This implies $\mathcal{S}[B](N) = \mathcal{S}[T; B](N)$ by the property CUM so that also $N = \overline{\mathcal{S}[T; B]}(N)$. Let us then check the $\mathcal{S}[T]$-stability of N. (i) Since $N \subseteq \overline{\mathcal{S}[T; B]}(N)$, we obtain $N \subseteq \overline{\mathcal{S}[T]}(N)$ by Lemma 8.4. But then N is $\mathcal{S}[T]$-consistent by Proposition 4.9. (ii) Assume that N is not $\mathcal{S}[T]$-complete. So there is $\phi \in \mathcal{L}^-(\mathcal{S}[T]) = \mathcal{L}^-(\mathcal{S})$ such that $\phi \notin N$ and $\phi \notin \mathcal{S}[T](N)$. The latter implies that $\phi \notin \mathcal{S}(N)$ and $\phi \notin T$ by the requirements MON and REF. Because $\phi \in \mathcal{L}^-(\mathcal{S})$, $\phi \notin N$ and $\phi \notin \mathcal{S}(N)$, it follows that $\phi \in T$, a contradiction. Hence N is $\mathcal{S}[T]$-complete. □

8.4.4 A Cumulative Extension of Stationary Semantics

Dix [19] presents a refinement of stationary semantics that is cumulative. This semantics has been formulated for normal logic programs solely and as cumulative extensions of well-founded semantics [115]. To describe these semantics at the level of non-monotonic systems, we have to provide a link between normal logic programs and the non-monotonic systems considered so far. In Section 10, we translate the rules of a (normal) logic program into autoepistemic sentences. Under this translation, $\mathcal{S}$-codiagonal sets of non-theorems N determine the total (two-valued) models of the program, and thus the classical consequences of the program. This is why the semantics proposed by Dix is so closely related to codiagonal semantics.

Dix [19] names the variant as WFS^+. This semantics constructs (repeatedly) the union of the well-founded model of a logic program and the intersection of all total models of the program. Schlipf has independently proposed the same idea. This idea is easily formulated at the level of non-monotonic systems by defining a semantic operator $\mathrm{STDIX}_{\mathcal{S}}$ for all $T \subseteq \mathcal{L}^+(\mathcal{S})$ by setting $\mathrm{STDIX}_{\mathcal{S}}(T) = \mathrm{STAT}_{\mathcal{S}}(T) \cup \mathrm{CODIAG}_{\mathcal{S}}(T)$. The semantics $\mathrm{STDIX}_{\mathcal{S}}$ satisfies a requirement called *rationality* RAT[33] which is a stronger requirement than CMON. Dix defines the final semantics $\mathrm{STDIX^+}_{\mathcal{S}}(T)$ as the limit of the iteration sequence $\mathrm{STDIX}_{\mathcal{S}} \uparrow^{\alpha} (T)$. The resulting semantics satisfies the requirement RAT and the requirement CCLO due to iteration. Hence $\mathrm{STDIX^+}_{\mathcal{S}}$ is also cumulative. Example 8.53 demonstrates that $\mathrm{STDIX}_{\mathcal{S}}$ and $\mathrm{CAUT}_{\mathcal{S}}$ provide quite different sets of theorems for systems $\mathcal{S}$ subject to instability.

[33]Stationary semantics is also rational.

Example 8.53 Consider a set of defaults $D = \{\frac{:\neg \mathrm{p}}{\mathrm{p}}\}$ and the subsystem $\mathcal{S} = \mathfrak{D}_D^{\neg \mathrm{J}(D)}$ having the language $\langle \mathcal{L}, \{\neg\neg \mathrm{p}\}\rangle$.

The least $\mathcal{S}$-stationary set of non-theorems is $N = \emptyset$ so that $\mathrm{STAT}_{\mathcal{S}}(\emptyset) = \mathrm{Cn}(\emptyset)$. It is obvious that there are no $\mathcal{S}$-stable sets of non-theorems. The only way to enforce stability of $N = \emptyset$ is to introduce a set of theorems $T \subseteq \mathcal{L}$ such that $\neg\neg\mathrm{p} \in \mathcal{S}[T](\emptyset)$. In other words, the set of non-theorems $N = \emptyset$ has to be $\mathcal{S}$-codiagonal by T. Thus $\neg\neg\mathrm{p}$ and p are included in $\mathcal{S}[T](N)$ for all such T as well as their intersection $\mathrm{CODIAG}_{\mathcal{S}}(\emptyset) = \mathrm{Cn}(\{\mathrm{p}\})$. Thus also $\mathrm{STDIX}_{\mathcal{S}}(\emptyset) = \mathrm{Cn}(\{\mathrm{p}\})$. It follows by similar analysis that $\mathrm{STDIX}_{\mathcal{S}}(\{\mathrm{p}\}) = \mathrm{Cn}(\{\mathrm{p}\})$ and $\mathrm{STDIX^+}_{\mathcal{S}}(\emptyset) = \mathrm{Cn}(\{\mathrm{p}\})$. In contrast to this, only $B = \{\neg\neg\mathrm{p}\}$ is a finished $\mathcal{S}$-bias – giving rise to the set of non-theorems $N = \emptyset$. Thus the cautious semantics $\mathrm{CAUT}_{\mathcal{S}}(\emptyset)$ is just the set of tautologies $\mathrm{Cn}(\emptyset)$. □

This example provided a system $\mathcal{S}$ such that $\mathrm{CAUT}_{\mathcal{S}}(T) \subset \mathrm{STDIX}_{\mathcal{S}}(T)$ for a set of sentences $T = \emptyset$. To give an example of a proper inclusion in the other direction, we take an autoepistemic variant of Example 8.47 into consideration as follows. Note that by these relationships, the semantic operators $\mathrm{CAUT}_{\mathcal{S}}$ and $\mathrm{STDIX}_{\mathcal{S}}$ are mutually incomparable with respect to the ordering $\sqsubseteq$.

Example 8.54 Consider an autoepistemic theory $\Sigma = \{\mathbf{D}\mathrm{a} \rightarrow \mathrm{b}, \mathbf{D}\mathrm{b} \rightarrow \mathrm{a}, \mathbf{D}\mathrm{a} \wedge \mathbf{D}\mathrm{b} \rightarrow \mathrm{c}, \mathbf{D}\mathrm{c} \rightarrow \mathrm{d}\}$ and the subsystem $\mathcal{S} = \mathcal{A}_{\Sigma}^{\Sigma\cup\{\mathbf{B}\mathrm{d}\}}$ which has the language $\langle\{\mathrm{a},\mathrm{b},\mathrm{c},\mathrm{d}\}, \{\mathrm{a},\mathrm{b},\mathrm{c}\}\rangle$.

It is clear that $N = \emptyset$ is the least $\mathcal{S}$-stationary set of non-theorems implying that $\mathrm{STAT}_{\mathcal{S}}(\emptyset) = \emptyset$. Moreover, only $N_1 = \{\mathrm{a},\mathrm{c}\}$ and $N_2 = \{\mathrm{b},\mathrm{c}\}$ are $\mathcal{S}$-stable, so that $\mathrm{STAB}_{\mathcal{S}}(\emptyset) = \{\mathrm{d}\}$. Also $\mathrm{CAUT}_{\mathcal{S}}(\emptyset) = \{\mathrm{d}\}$, since the finished $\mathcal{S}$-biases are essentially $B_1 = \{\mathrm{a}\}$ and $B_2 = \{\mathrm{b}\}$ which give rise to the least fixed points N_1 and N_2, respectively. To the contrary, there are more $\mathcal{S}$-codiagonal sets of non-theorems such as $N_3 = \{\mathrm{b},\mathrm{d}\}$ induced by $T_3 = \{\mathrm{c}\}$. This contributes the set of theorems $\mathcal{S}[T_3](N_3) = \{\mathrm{a},\mathrm{c}\}$ to the intersection, so that $\mathrm{CODIAG}_{\mathcal{S}}(\emptyset) = \emptyset$, $\mathrm{STDIX}_{\mathcal{S}}(\emptyset) = \emptyset$ and consequently also $\mathrm{STDIX^+}_{\mathcal{S}}(\emptyset) = \emptyset$. □

Let us yet analyze the semantics $\mathrm{STDIX^+}_{\mathcal{S}}$ in terms of other examples already presented in this work. This semantics handles correctly the "floating conclusion" r in Example 8.1. The $\mathcal{S}$-codiagonal sets of non-theorems are essentially $N_1 = \{\mathrm{p}\}$ and $N_2 = \{\mathrm{q}\}$ obtained by $T = \emptyset$ and $N_3 = \emptyset$ obtained by $T = \{\mathrm{p},\mathrm{q}\}$. Thus r is included in $\mathrm{CODIAG}_{\mathcal{S}}(\emptyset)$, $\mathrm{STDIX}_{\mathcal{S}}(\emptyset)$ as well as $\mathrm{STDIX^+}_{\mathcal{S}}(\emptyset)$. The fact that $\mathrm{STDIX^+}_{\mathcal{S}}$ is cumulatively closed becomes apparent with the system presented in Example 8.28. In this case, both r and u are included in the set of theorems $\mathrm{STDIX^+}_{\mathcal{S}}(\emptyset)$.

Dix [19] proposes a further cumulative extension of the well-founded semantics of a normal logic program P by refining the well-founded model of P with the intersection of classical models and regular models of P. Therefore, the resulting extension WFS′ of well-founded semantics is weaker than WFS$^+$. This approach can be characterized in the framework of non-monotonic systems in terms of the operator $\mathrm{STDIX}'_{\mathfrak{S}}(T) = \mathrm{STAT}_{\mathfrak{S}}(T) \cup (\mathrm{CODIAG}_{\mathfrak{S}}(T) \cap \mathrm{REG}_{\mathfrak{S}}(T))$. This is a rational semantics and a cumulative semantics can be obtained through iteration in analogy to $\mathrm{STDIX}^+{}_{\mathfrak{S}}$. Example 8.54 differentiates this semantics from cautious semantics: since $\mathrm{STAT}_{\mathfrak{S}}(T) = \emptyset$ and $\mathrm{CODIAG}_{\mathfrak{S}}(T) = \emptyset$ hold, it follows that $\mathrm{STDIX}'_{\mathfrak{S}}(T) = \emptyset$ whereas $\mathrm{CAUT}_{\mathfrak{S}}(T) = \{\mathrm{d}\}$.

8.4.5 Other Semantics

Let us yet comment on two further approaches. Pereira, Alferes and Aparício [88] add closed world assumptions to well-founded semantics to obtain *O-semantics.* This approach resembles cautious semantics in the sense that a set of atoms $N \subseteq \mathrm{Hb}(P)$ is fixed as non-theorems (negative assumptions in their terminology). However, Pereira and al. use a different inference operator for normal logic programs which does not satisfy our specification of non-monotonic systems.[34] Therefore, O-semantics is quite different from cautious semantics. The author [44] has pointed out that O-semantics suffers a consistency problem, i.e. O-semantics may provide a three-valued model for a normal logic program which is not extendible to a two-valued model. Such a completion is always possible under cautious semantics (see Theorem 8.52 and the discussion in the beginning of Section 8.4.4).

Brewka and Gottlob [12] establish a range of well-founded semantics for default logic by composing monotone operators from two anti-monotonic operators in analogy to stationary semantics [89]. They introduce various notions of default proofs[35] to refine stationary semantics gradually to full stable semantics. By Theorem 8.21 it is to be expected that our biasing technique captures at least their notion of *extendable* default proofs (proofs that are valid in a default extension). As a motivating example, Brewka and Gottlob present a set of defaults $D = \{\frac{:\mathrm{b}}{\mathrm{b}}, \frac{:\neg\mathrm{a}}{\neg\mathrm{a}}, \frac{:\mathrm{a}}{\mathrm{a}}\}$ to demonstrate the weakness of stationary semantics, since $\mathrm{STAT}_{\mathfrak{D}_D}(\emptyset) = \mathrm{Cn}(\emptyset)$. In this example, the aims of their approach and our approach are quite the same, since $\mathrm{CAUT}_{\mathfrak{D}_D}(\emptyset) = \mathrm{STAB}_{\mathfrak{D}_D}(\emptyset)$ contains the desired conclusion b. However, Brewka's and Gottlob's approach covers only *coherent* default theories, i.e. default theories that have at least one extension. For this reason, this framework cannot be applied for the set of defaults D given in Example 8.47, for instance. Note that a reasonable case analysis is performed for the given set of defaults D under cautious semantics.

[34] This is because they remove rules that have an atom $\mathrm{a} \in N$ as a positive subgoal and thus the requirement MON fails.

[35] We reduce a set of defaults D to a set of monotonic inference rules R and use the R-proofs of Definition 2.20 instead of default proofs.

9 Computational Aspects of Cautious Semantics

Let us now turn our attention to the computational aspects of cautious semantics. The analysis is based on the methods and algorithms introduced in Section 7. In particular, the function $\mathsf{Iterate}_X$ is needed constantly to compute the least stationary fixed point induced by a bias (candidate) B. In Section 9.1, the time complexity of cautious semantics is studied. It turns out that the main decision problem of cautious semantics is $\mathbf{\Pi}_2^{\mathrm{p}}$-complete for the logics under consideration and thus of the same computational complexity as the corresponding decision problem of stable semantics. Also a brief comparison is made with the semantics presented in Section 8.4. In Section 9.2, algorithms for (i) finding a single $\mathcal{S}[T]$-bias $B \subseteq \mathcal{L}^-(\mathcal{S}[T])$ and (ii) testing whether a given $B \subseteq \mathcal{L}^-(\mathcal{S}[T])$ is an $\mathcal{S}[T]$-bias are presented. Also various optimization techniques are considered to enhance a decision procedure for cautious semantics. Finally, an approximation method for cautious semantics is considered in Section 9.3. A size restriction α is imposed on biases B and it is shown that a range of gradually refining approximations of cautious semantics, which is parametrized by α, is obtained. The weakest approximation coincides with stationary semantics while the strongest coincides with the full cautious semantics. The required changes to decision procedures are also discussed.

9.1 Time Complexity

By the following definition, we set up the obvious decision problem associated with cautious semantics. In analogy to Section 7, we consider the problem $\mathbf{CAUT}_{\mathrm{L}}$ for three logics L of our interest: circumscription, default logic and the variant of autoepistemic logic proposed in this work. Like before, we write $\mathcal{N}_X$ to denote any of the systems $\mathfrak{C}_P$, $\mathfrak{D}_D$ and $\mathcal{A}_\Sigma$ associated with these logics.

Definition 9.1

$$\mathbf{CAUT}_{\mathrm{L}} \textit{ is the set of triples } \langle X, T, \phi\rangle \textit{ such that } \phi \in \mathrm{CAUT}_{\mathcal{N}_X}(T)$$

where X is a finite subset of parameters in logic L, $T \subseteq \mathcal{L}^+(\mathcal{N}_X)$ is a finite set of sentences and $\phi \in \mathcal{L}^+(\mathcal{N}_X)$ is a sentence.

Our first algorithm related with cautious semantics is given in Figure 12. The function EBias_X tests whether a given enumeration ε satisfies the bias condition of Definition 8.7. Note that we are using the functions NT_X and $\mathsf{Iterate}_X$ introduced in Section 7.3 as subroutines in the algorithm. The properties of the algorithm are analyzed in Proposition 9.2.

Function $\mathsf{EBias}_X(T, \varepsilon, L^-)$: Boolean;
Var
  T: a finite subset of $\mathcal{L}^+(\mathbf{N}_X)$;
  ε: a finite enumeration of a subset of $\mathcal{L}^-(\mathbf{N}_X)$;
  B, N, N', L^-: finite subsets of $\mathcal{L}^-(\mathbf{N}_X)$;
  ϕ: a sentence of $\mathcal{L}^-(\mathbf{N}_X)$;
  A: Boolean
Begin
  $A :=$ True;
  $B := \emptyset$;
  For $\phi \in \varepsilon$ **Do**
    Begin
      $N := \mathsf{Iterate}_X(T, B, L^-)$;
      $N' := \mathsf{NT}_X(T, B \cup N, L^-)$;
      If $\phi \notin N'$ **Then** $A :=$ False;
      $B := B \cup \{\phi\}$
    End;
  Return A
End

Figure 12: A function that accepts biasing enumerations ε for $\mathbb{S}[T]$

Proposition 9.2 *If there is a subsystem* $\mathbb{S} \sqsubseteq \mathbf{N}_X$ *such that* $T \subseteq \mathcal{L}^+(\mathbb{S})$ *and* ε *enumerates a subset of* $\mathcal{L}^-(\mathbb{S}) = L^-$*, then the call* $\mathsf{EBias}_X(T, \varepsilon, L^-)$ *(i) returns* True *if and only if* ε *satisfies the bias condition for* $\mathbb{S}[T]$ *and (ii) causes at most* $|\varepsilon| \times (2 \times |L^-|^2 + 3 \times |L^-|)$ *subsequent calls to the function* TH_X*.*

Proof. Let us first show that each execution of the **For**-loop tests correctly whether ϕ is a bias for $\mathbb{S}[T; B]$ or not. So let ϕ be any sentence of $\mathcal{L}^-(\mathbb{S})$ and B any subset of $\mathcal{L}^-(\mathbb{S})$. By Proposition 7.15 the call $\mathsf{Iterate}_X(T, B, L^-)$ returns the least fixed point $\mathrm{lfp}(\overline{\mathbb{S}[T; B]}^2, \emptyset)$ which is the new value of N. Consequently, the call $\mathsf{NT}_X(T, B \cup N, L^-)$ returns the set of potential non-theorems $\overline{\mathbb{S}[T; B]}(N)$ by Proposition 7.12. This is set of non-theorems is assigned as the value of N'. Thus the test $\phi \notin N'$ succeeds if and only if ϕ is not a bias for $\mathbb{S}[T; B]$. Let us then establish the correctness of the whole algorithm.

It is easy to see that whenever ϕ is assigned to an element ε_i of the enumeration, the value of B is $B_{i-1} = \{\varepsilon_{0\ldots(i-1)}\}$ (except after B is assigned to its new value B_i at the end of the **For**-loop). (i) The call $\mathsf{EBias}_X(T, \varepsilon, L^-)$ returns True if and only if the test $\phi \notin N'$ fails for each $\phi = \varepsilon_i$ where $0 < i \leq |\varepsilon|$. By the analysis above, this is the case if and only if each $\phi = \varepsilon_i$ is a bias for $\mathbb{S}[T; B_{i-1}]$ where $B_{i-1} = \{\varepsilon_{0\ldots(i-1)}\}$. Equivalently, the enumeration ε satisfies the bias condition for $\mathbb{S}[T]$. (ii) The **For**-loop of the algorithm is executed

$|\varepsilon|$ times. Each loop execution involves subsequent calls to functions $\mathsf{Iterate}_X$ and NT_X. By Propositions 7.15 and 7.12, these calls cause at most a total of $2 \times |L^-|^2 + 3 \times |L^-|$ calls to the function TH_X. Thus the number of calls to TH_X is at most $|\varepsilon| \times (2 \times |L^-|^2 + 3 \times |L^-|)$. □

In Figure 13, we have designed a nondeterministic Turing machine to accept the set $\overline{\mathbf{CAUT}_\mathrm{L}}$. Machine CoCaut_X chooses nondeterministically a bias candidate B as well as its enumeration ε and test whether this leads to a counter example for the query ϕ with respect to cautious semantics, i.e. the set B turns out to be a finished $\mathbb{S}[T]$-bias such that $\phi \notin \mathbb{S}[T](N)$ for the least fixed point $N = \mathrm{lfp}(\overline{\mathbb{S}[T;B]}^2, \emptyset)$. In the algorithm, we use $\mathcal{E}(S)$ to denote the set of all enumerations of a set S. Note that the sets $\mathcal{E}(B)$ and $\mathbf{2}^{L^-}$ need not be constructed explicitly so that a subset B of L^- and an enumeration ε of B can be chosen nondeterministically in polynomial time and space.

Machine $\mathsf{CoCaut}_X(T, \phi)$;
Var
  T: a finite subset of $\mathcal{L}^+(\mathbb{N}_X)$;
  B, N, L^-: finite subsets of $\mathcal{L}^-(\mathbb{N}_X)$;
  ϕ: a sentence of $\mathcal{L}^+(\mathbb{N}_X)$
Begin
  $L^- := \mathsf{LangN}_X(T, \phi)$;
  Choose B **From** $\mathbf{2}^{L^-}$;
  Choose ε **From** $\mathcal{E}(B)$;
  $N := \mathsf{Iterate}_X(T, B, L^-)$;
  If $\mathsf{EBias}_X(T, \varepsilon, L^-)$ **And** $\mathsf{NT}_X(T, B \cup N, L^-) \subseteq B \cup N$
    And Not $\mathsf{TH}_X(T, N, \phi)$ **Then Accept**
  Else
    Reject
End

Figure 13: An oracle machine accepting $\overline{\mathbf{CAUT}_\mathrm{L}}$

Proposition 9.3 $\overline{\mathbf{CAUT}_\mathrm{L}}$ *is nondeterministic Turing-reducible to the problem* $\mathbf{TH}_\mathrm{L}$ *in polynomial time for circumscription, default logic and autoepistemic logic.*

Proof. It is shown that the machine CoCaut_X accepts the set $\overline{\mathbf{CAUT}_\mathrm{L}}$. By Lemma 7.13, the language L^- computed by the call $\mathsf{LangN}_X(T, \phi)$ is the language $\mathcal{L}^-(\mathbb{S})$ for a subsystem $\mathbb{S} \sqsubseteq \mathbb{N}_X$ such that $T \cup \{\phi\} \subseteq \mathcal{L}^+(\mathbb{S})$. The machine CoCaut_X has an accepting computation on the inputs T and ϕ if and only if a set of non-theorems B and an enumeration ε of B is chosen such that (i) $\mathsf{EBias}_X(T, B, L^-)$ returns True, (ii) $\mathsf{NT}_X(T, B \cup N, L^-)$ returns a subset of $B \cup N$ where N has been computed by the call $\mathsf{Iterate}_X(T, B, L^-)$ and (iii) $\mathsf{TH}_X(T, N, \phi)$ returns False. By Propositions 9.2, 7.12 and 7.15, Lemma 7.11 as well as Definition 8.12, this is the case if and only if (i) B is an $\mathcal{S}[T]$-bias (ii) B is finished and (iii) $\phi \notin \mathcal{S}[T](N)$ for the least fixed point $N = \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$. These hold if and only if $\phi \notin \mathrm{CAUT}_{\mathcal{S}}(T)$. Equivalently, we have that $\phi \notin \mathrm{CAUT}_{\mathcal{N}_X}(T)$ by Theorem 8.34. By Definition 9.1, this holds if and only if $\langle X, T, \phi \rangle \in \overline{\mathbf{CAUT}_{\mathrm{L}}}$.

Let us then ensure that the computation of the machine CoCaut_X takes place in polynomial time. It is obvious that the nondeterministic choice of B and an enumeration ε of B can be accomplished in time polynomial to $|L^-|$. Since $|\varepsilon| \leq |L^-|$, the call $\mathsf{EBias}_X(T, \varepsilon, L^-)$ makes at most $2 \times |L^-|^3 + 3 \times |L^-|^2$ calls to the oracle TH_X by Proposition 9.2. The call $\mathsf{Iterate}_X(T, B, L^-)$ makes at most $2 \times |L^-|^2 + 2 \times |L^-|$ calls to the oracle TH_X by Proposition 7.15. The call $\mathsf{NT}_X(T, B \cup N, L^-)$ causes $|L^-|$ calls to the oracle TH_X. In addition to these, the machine CoCaut_X makes one more call to TH_X. Therefore the total number of calls to the oracle TH_X is bounded by $2 \times |L^-|^3 + 5 \times |L^-|^2 + 3 \times |L^-|^2 + 1$ where $|L^-| < ||L^-||$ is polynomial to the length of input $||X|| + ||T|| + ||\phi||$. Thus we conclude that the computation of the machine CoCaut_X takes a polynomial time. $\square$

It was shown in Theorem 8.23 that cautious semantics and stable semantics coincide for minimal model systems. This interconnection is used in the following to establish lower limits for the complexity of the problem $\mathbf{CAUT}_{\mathrm{L}}$ for the three non-monotonic logics under consideration.

Proposition 9.4 $\mathbf{CAUT}_{\mathrm{L}}$ *is* $\mathbf{\Pi}^{\mathrm{p}}_2$*-hard for circumscription, default logic and the variant of autoepistemic logic proposed in this work.*

Proof. Eiter and Gottlob [25] have shown that $\mathbf{STAB}_{\mathrm{CIRC}}$ is $\mathbf{\Pi}^{\mathrm{p}}_2$-hard. Theorem 8.23 implies that the triple $\langle P, T, \phi \rangle \in \mathbf{STAB}_{\mathrm{CIRC}}$ if and only if $\langle P, T, \phi \rangle \in \mathbf{CAUT}_{\mathrm{CIRC}}$, i.e. only a trivial transformation is needed to m-reduce the problem $\mathbf{STAB}_{\mathrm{CIRC}}$ to $\mathbf{CAUT}_{\mathrm{CIRC}}$. Thus $\mathbf{STAB}_{\mathrm{CIRC}} \leq^{\mathrm{p}}_{\mathrm{m}} \mathbf{CAUT}_{\mathrm{CIRC}}$ so that the problem $\mathbf{CAUT}_{\mathrm{CIRC}}$ is also $\mathbf{\Pi}^{\mathrm{p}}_2$-hard. On the other hand, it holds by Corollary 8.41 that $\langle P, T, \phi \rangle \in \mathbf{CAUT}_{\mathrm{CIRC}}$ if and only if $\langle \mathrm{Tr}_{\mathrm{E}}(P), T, \phi \rangle \in \mathbf{CAUT}_{\mathrm{DL}}$. The required transformation is linear so that $\mathbf{CAUT}_{\mathrm{CIRC}} \leq^{\mathrm{p}}_{\mathrm{m}} \mathbf{CAUT}_{\mathrm{DL}}$. Thus $\mathbf{CAUT}_{\mathrm{DL}}$ is $\mathbf{\Pi}^{\mathrm{p}}_2$-hard. Finally, we know by Corollary 8.42 that $\langle D, T, \phi \rangle \in \mathbf{CAUT}_{\mathrm{DL}}$ if and only if $\langle \mathrm{Tr}_{\mathrm{K}}(D), T, \phi \rangle \in \mathbf{CAUT}_{\mathrm{AEL}}$. Again, the required transformation is linear. It follows that $\mathbf{CAUT}_{\mathrm{DL}} \leq^{\mathrm{p}}_{\mathrm{m}} \mathbf{CAUT}_{\mathrm{AEL}}$. Consequently, the problem $\mathbf{CAUT}_{\mathrm{AEL}}$ is $\mathbf{\Pi}^{\mathrm{p}}_2$-hard. $\square$

Theorem 9.5 $\mathbf{CAUT}_{\mathrm{L}}$ *is* $\mathbf{\Pi}^{\mathrm{p}}_2$*-complete for circumscription, default logic and the variant of autoepistemic logic proposed in this work.*

Proof. By Theorem 7.8 the problem $\mathbf{TH}_{\mathrm{L}}$ is $\mathbf{\Pi}^{\mathrm{p}}_1$-complete. This implies that $\mathbf{CAUT}_{\mathrm{L}} \in \mathbf{\Pi}^{\mathrm{p}}_2$, since $\overline{\mathbf{CAUT}_{\mathrm{L}}}$ is nondeterministic Turing-reducible to $\mathbf{TH}_{\mathrm{L}}$ by Proposition 9.3. The $\mathbf{\Pi}^{\mathrm{p}}_2$-hardness of $\mathbf{CAUT}_{\mathrm{L}}$ was established in Proposition 9.4. Thus $\mathbf{CAUT}_{\mathrm{L}}$ is $\mathbf{\Pi}^{\mathrm{p}}_2$-complete. □

Having analyzed the complexity of cautious semantics, let us present a couple of remarks on the approaches addressed in Section 8.4. It seems that the techniques of Section 9.1 can be applied to prove that the decision problem of regular semantics [119] is $\mathbf{\Pi}^{\mathrm{p}}_2$-complete for the logics under consideration, i.e. of the same complexity as the decision problem of cautious semantics. Baral and Subrahmanian [3] do not analyze the computational complexity of stable class semantics. An open question in this respect is whether a non-empty and strict $\mathcal{N}_X$-stable class of sets of non-theorems can always be represented in polynomial space, or are such classes exponential sometimes. The computational complexity of Thiele's codiagonal semantics [110] has not been analyzed. However, by the similarity of codiagonal semantics and stable semantics, it is to be expected that the complexity of stable semantics is not exceeded. Dix does not analyze the complexity of his cumulative extension of well-founded semantics. By the interconnection to codiagonal semantics, this semantics is probably at least as complex as codiagonal semantics.

9.2 Algorithms for Cautious Semantics

In the following, we develop algorithms for computing biases and verifying queries under cautious semantics. The goal is to present relatively optimized algorithms that (i) find one finished $\mathcal{S}[T]$-bias, (ii) test whether a given $B \subseteq \mathcal{L}^-(\mathcal{S}[T])$ is an $\mathcal{S}[T]$-bias and (iii) decide whether $\phi \in \mathrm{CAUT}_{\mathcal{S}}(T)$. Here $\mathcal{S}$ refers to a suitably chosen subsystem of $\mathcal{N}_X$ in analogy to algorithms and Turing machines presented so far. So let us begin with an algorithm which computes one finished $\mathcal{S}[T]$-bias in a bottom-up fashion. The algorithm is given as function $\mathsf{FindBias}_X$ in Figure 14.

Proposition 9.6 *If there is $\mathcal{S} \sqsubseteq \mathcal{N}_X$ such that $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $\mathcal{L}^-(\mathcal{S}) = L^-$, then the call $\mathsf{FindBias}_X(T, L^-)$ returns a finished $\mathcal{S}[T]$-bias B by making at most a total of $2 \times |L^-|^3 + 3 \times |L^-|^2$ calls to TH_X.*

Proof. Let us show that the **For**-loop of the algorithm satisfies an invariant: the value of B is always an $\mathcal{S}[T]$-bias. Clearly, this holds when the loop is entered and $B = \emptyset$. Then consider any later execution of the loop. First of all, the call to function $\mathsf{Iterate}_X$ computes the least fixed point $N = \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset)$ by Proposition 7.15. Thus the call $\mathsf{NT}_X(T, B \cup N, L^-)$ returns the set of non-theorems $\overline{\mathcal{S}[T;B]}(N)$ by Proposition 7.12. This is then assigned as the value of N'. The sentence ϕ is added to the bias if it belongs to $N' - N$, i.e.

Function $\mathsf{FindBias}_X(T, L^-)$: a finite subset of $\mathcal{L}^-(\mathcal{N}_X)$;
Var
  T: a finite subset of $\mathcal{L}^+(\mathcal{N}_X)$;
  B, N, N', L^-: finite subsets of $\mathcal{L}^-(\mathcal{N}_X)$;
  ϕ: a sentence of $\mathcal{L}^-(\mathcal{N}_X)$
Begin
  $B := \emptyset$;
  For $\phi \in L^-$ **Do**
    Begin
      $N := \mathsf{Iterate}(T, B, L^-)$;
      $N' := \mathsf{NT}_X(T, B \cup N, L^-)$;
      If $\phi \in N' - N$ **Then** $B := B \cup \{\phi\}$;
      $B := B \cup N$
    End
  Return B
End

Figure 14: An algorithm to compute a finished $\mathcal{S}[T]$-bias B

$\phi \in \overline{\mathcal{S}[T]}(N) - N$. This implies that ϕ is a bias for $\mathcal{S}[T; B]$ so that $B \cup \{\phi\}$ is an $\mathcal{S}[T]$-bias. Thus the new value of B is also an $\mathcal{S}[T]$-bias. As the final operation in the loop it is ensured that the sentences of N are included in the bias. This is a legible operation by Proposition 8.18 where it is shown that the sentences in $N - B$ can be added to the $\mathcal{S}[T]$-bias B such that $B \cup N$ is an $\mathcal{S}[T]$-bias. Thus our loop invariant still holds at the end of the for loop. This trick increases the efficiency of our algorithm, since the least fixed point N need not to be reconstructed by the function $\mathsf{Iterate}_X$ over and over again.

By the above, it is clear that $\mathsf{FindBias}_X(T, L^-)$ returns an $\mathcal{S}[T]$-bias B such that $N = \mathrm{lfp}(\overline{\mathcal{S}[T]}^2, \emptyset) \subseteq B$. Let us then assume that the final bias B is not finished, i.e. $B \cup N = B$ is not $\mathcal{S}[T]$-complete. Then there is $\phi \in \mathcal{L}^-(\mathcal{S}) = L^-$ such that $\phi \notin B$ and $\phi \notin \mathcal{S}[T](B)$. So let us analyze the round when ϕ was processed by the **For**-loop. Let B_ϕ be the value of B when that loop execution started and N_ϕ the corresponding least fixed point. Note that $B_\phi \subseteq B$ and $N_\phi \subseteq N$ by Proposition 8.17. Since ϕ was not included in B, we conclude that $\phi \notin N'$ must have been the case. This implies that $\phi \notin \overline{\mathcal{S}[T; B_\phi]}(N_\phi)$ and $\phi \in \mathcal{S}[T; B_\phi](N_\phi)$. But then we obtain a contradiction $\phi \in \mathcal{S}[T](B)$ by the requirement MON, since both B_ϕ and $N_\phi \subseteq N$ are included in B. Hence B is necessarily a finished $\mathcal{S}[T]$-bias.

Finally, let us analyze the worst case time complexity of the algorithm. The **For**-loop of the algorithm is executed $|L^-|$ times. Each execution involves a call to the function $\mathsf{Iterate}_X$ and a call to the function NT_X. By Propositions

7.15 and 7.12, these cause at most $2 \times |L^-|^2 + 2 \times |L^-|$ and $|L^-|$ calls to the function TH_X, respectively. Thus the number of calls to the function TH_X is bounded by $|L^-| \times (2 \times |L^-|^2 + 2 \times |L^-| + |L^-|) = 2 \times |L^-|^3 + 3 \times |L^-|^2$. □

The next algorithm is given in Figure 15. The purpose of the function Bias_X is to test whether a given set $B \subseteq \mathcal{L}^-(\mathcal{S}[T])$ is an $\mathcal{S}[T]$-bias. The idea behind the algorithm is that a enumeration satisfying the bias condition is searched for in a reverse order, i.e. the last sentence in the enumeration is determined first and then removed from the bias candidate B. Our algorithm applies this operation repeatedly in order to reduce B to $\emptyset$ eventually. If this succeeds, the set B given as the input of the algorithm is an $\mathcal{S}[T]$-bias. By this strategy, our algorithm avoids constructing all enumerations of B. A substantial saving in execution time results by this strategy, since the number of enumerations of a bias B of size n is $n!$ which grows much faster than 2^n when n grows. In fact, the function Bias_X requires only a polynomial number of calls to TH_X as to be shown later in Proposition 9.9.

Function $\mathsf{Bias}_X(T, B, L^-)$: Boolean;
Var
  T: a finite subset of $\mathcal{L}^+(\mathcal{N}_X)$;
  B, B', N, N', L^-: finite subsets of $\mathcal{L}^-(\mathcal{N}_X)$;
  ϕ: a sentence of $\mathcal{L}^-(\mathcal{N}_X)$
Begin
  Repeat
    $B' := B$;
    For $\phi \in B$ **Do**
      Begin
        $N := \mathsf{Iterate}_X(T, B - \{\phi\}, L^-)$;
        $N' := \mathsf{NT}_X(T, (B - \{\phi\}) \cup N, L^-)$;
        If $\phi \in N'$ **Then** $B := B - \{\phi\}$
      End
  Until $B = B'$;
  If $B = \emptyset$ **Then Return** True **Else Return** False
End

Figure 15: A function that accepts biases B for $\mathcal{S}[T]$

To prove the correctness of the function Bias_X in Figure 15 we have to establish two technical results at first. In Lemma 9.7 we provide a sufficient condition to judge that a given $B \subseteq \mathcal{L}^-(\mathcal{S})$ is not an $\mathcal{S}$-bias. Proposition 9.8 shows that the reduction step of the algorithm preserves the (possible) bias property of $B \subseteq \mathcal{L}^-(\mathcal{S})$. Since our algorithms handle only subsystems $\mathcal{S}$ with a finite language $\mathcal{L}^-(\mathcal{S})$, we restrict Lemma 9.7 and Proposition 9.8 to the same scope. The correctness of the function Bias_X is addressed in Proposition 9.9.

Lemma 9.7 *Let $\mathcal{S}$ be a non-monotonic system such that $\mathcal{L}^-(\mathcal{S})$ is finite. If $B \subseteq \mathcal{L}^-(\mathcal{S})$ is a set of non-theorems such that* **no** *$\phi \in B$ is a bias for the system $\mathcal{S}[B - \{\phi\}]$, then B is not an $\mathcal{S}$-bias.*

Proof. Assume that $B \subseteq \mathcal{L}^-(\mathcal{S})$ satisfies the above requirement, and that B is an $\mathcal{S}$-bias. By Definition 8.7 there is an enumeration ε of B such that ε satisfies the bias condition for $\mathcal{S}$. Obviously, the enumeration ε is finite, since the set $B \subseteq \mathcal{L}^-(\mathcal{S})$ is. Let ϕ be the last sentence in ε and $B' = B - \{\phi\}$. Then ϕ must be a bias for $\mathcal{S}[B']$, a contradiction. Hence B is not an $\mathcal{S}$-bias. □

Proposition 9.8 *Let $\mathcal{S}$ be a non-monotonic system such that $\mathcal{L}^-(\mathcal{S})$ is finite. If a set of non-theorem $B \subseteq \mathcal{L}^-(\mathcal{S})$ is an $\mathcal{S}$-bias and $\phi \in B$ is a bias for $\mathcal{S}[B - \{\phi\}]$, then $B' = B - \{\phi\}$ is an $\mathcal{S}$-bias such that $\mathrm{lfp}(\overline{\mathcal{S}[B']}^2, \emptyset) \subseteq \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$.*

Proof. Let $B \subseteq \mathcal{L}^-(\mathcal{S})$ be an $\mathcal{S}$-bias and ϕ a bias for $\mathcal{S}[B']$ where $B' = B - \{\phi\}$. Let $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$ and $N' = \mathrm{lfp}(\overline{\mathcal{S}[B']}^2, \emptyset)$, respectively. It follows by Proposition 8.5 already that $N' \subseteq N$. Since B is an $\mathcal{S}$-bias and B is finite, there is a finite enumeration ε of B such that ε satisfies the bias condition for $\mathcal{S}$. Define $B_i = \{\varepsilon_{0\ldots i}\}$ and $N_i = \mathrm{lfp}(\overline{\mathcal{S}[B_i]}^2, \emptyset)$ for all natural numbers i in the range $0 \leq i \leq |\varepsilon|$. It is clear that each B_i is an $\mathcal{S}$-bias and $N_0, N_1, \ldots, N_i, \ldots$ is a non-decreasing sequence of sets of non-theorems. Let us then define $B'_i = B_i - \{\phi\}$ and $N'_i = \mathrm{lfp}(\overline{\mathcal{S}[B'_i]}^2, \emptyset)$ for all natural numbers i in the range $0 \leq i \leq |\varepsilon|$. It is proved by induction for all i in this range that

(i) B'_i is an $\mathcal{S}$-bias, (ii) $N'_i \subseteq N_i$ and (iii) $N'_i \subseteq N'$.

Let us check the base case $i = 0$. (i) Now $B'_0 = \emptyset$ is trivially an $\mathcal{S}$-bias by Proposition 8.9. (ii) Since both B'_0 and B_0 are empty, it holds that $N'_0 = N_0 = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$. (iii) Since B is an $\mathcal{S}$-bias, it follows by Proposition 8.17 that $B \subseteq \overline{\mathcal{S}}(N_0)$ for $N_0 = \mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$. Thus also $B' = B - \{\phi\} \subseteq \overline{\mathcal{S}}(N_0)$ and $N'_0 = N_0$ is a subset of $N' = \mathrm{lfp}(\overline{\mathcal{S}[B']}^2, \emptyset)$ by Proposition 8.5.

Induction step follows. Two cases arise depending whether $\varepsilon_i = \phi$ or not.

(A) If $\varepsilon_i = \phi$, then $B'_i = B'_{i-1}$ and B'_i and N'_i have the required properties (i), (ii) and (iii) directly by the inductive hypothesis.

(B) Assume that $\varepsilon_i \neq \phi$ so that $B'_i = B'_{i-1} \cup \{\varepsilon_i\}$. (i) Now ε_i is a bias for $\mathcal{S}[B_{i-1}]$, i.e. $\varepsilon_i \in \overline{\mathcal{S}[B_{i-1}]}(N_{i-1})$. Since $B'_{i-1} \subseteq B_{i-1}$ by definition and $N'_{i-1} \subseteq N_{i-1}$ by the inductive hypothesis, we obtain $\overline{\mathcal{S}[B_{i-1}]}(N_{i-1}) \subseteq \overline{\mathcal{S}[B_{i-1}]}(N'_{i-1}) \subseteq \overline{\mathcal{S}[B'_{i-1}]}(N'_{i-1})$ by Proposition 4.25 and Lemma 8.4. It follows that also $\varepsilon_i \in \overline{\mathcal{S}[B'_{i-1}]}(N'_{i-1})$ so that ε_i is a bias for $\mathcal{S}[B'_{i-1}]$. Because B'_{i-1} is an $\mathcal{S}$-bias by the inductive hypothesis, so is $B'_i = B'_{i-1} \cup \{\varepsilon_i\}$ by Proposition 8.9.

(ii – iii) Define $M_j = \overline{\mathcal{S}[B_i']}^2 \uparrow^j (\emptyset)$ for all natural numbers j. Thus $M_j \subseteq N_i'$ for all j. Let us then prove by induction on j that

$$\text{(iv) } M_j \subseteq N_i \text{ and (v) } M_j \subseteq N'.$$

The base case holds, since $M_0 = \emptyset$ is contained in both N_i and N'. (Sub)induction step follows. Let us first establish the inclusions

$$\text{(vi) } \overline{\mathcal{S}[B_i]}(N_i) \subseteq \overline{\mathcal{S}[B_i']}(M_{j-1}) \text{ and (vii) } \overline{\mathcal{S}[B']}(N') \subseteq \overline{\mathcal{S}[B_i']}(M_{j-1}).$$

Since $B_i' \subseteq B_i$ and $B_i' \subseteq B'$ by definition and $M_{j-1} \subseteq N_i$ and $M_{j-1} \subseteq N'$ by the (sub)inductive hypothesis, we know that the inclusions (vi) and (vii) hold by Proposition 4.25 and Lemma 8.4. Let us proceed to the induction step.

(iv) It follows from (vi) that $\overline{\mathcal{S}[B_i]}(\overline{\mathcal{S}[B_i']}(M_{j-1})) \subseteq \overline{\mathcal{S}[B_i]}^2(N_i) = N_i$ by Proposition 4.25. Since ϕ is a bias for $B' = B - \{\phi\}$ we know that $\phi \in \overline{\mathcal{S}[B']}(N')$. Then we obtain $\phi \in \overline{\mathcal{S}[B_i']}(M_{j-1})$ by (vii). Since $B_i' = B_i - \{\phi\}$, it follows that $M_j = \overline{\mathcal{S}[B_i']}^2(M_{j-1}) = \overline{\mathcal{S}[B_i]}(\overline{\mathcal{S}[B_i']}(M_{j-1})) \subseteq N_i$.

(v) Now (vii) implies $\overline{\mathcal{S}[B']}(\overline{\mathcal{S}[B_i']}(M_{j-1})) \subseteq \overline{\mathcal{S}[B']}^2(N') = N'$ by Proposition 4.25. Then consider any sentence ψ in the difference $B' - B_i'$. It follows that $\psi \neq \phi$ and $\psi \in B - B_i$. Since $B - B_i$ is an $\mathcal{S}[B_i]$-bias, it follows that $\psi \in \overline{\mathcal{S}[B_i]}(N_i)$ by Proposition 8.17. Then $\psi \in \overline{\mathcal{S}[B_i']}(M_{j-1})$ follows by (vi). Thus we have established that $B' - B_i' \subseteq \overline{\mathcal{S}[B_i']}(M_{j-1})$. It follows that $M_j = \overline{\mathcal{S}[B_i']}^2(M_{j-1}) = \overline{\mathcal{S}[B']}(\overline{\mathcal{S}[B_i']}(M_{j-1}))$ is a subset of N'.

To conclude, we have shown that (iv) and (v) hold for all j. This implies that the set of non-theorems $N_i' = \mathrm{lfp}(\overline{\mathcal{S}[B_i']}^2, \emptyset)$ satisfies (ii) $N_i' \subseteq N_i$ and (iii) $N_i' \subseteq N$. This completes the case (B) and our proof on the claims (i – iii). By setting $i = |\varepsilon|$ in (i) and (ii), we know that $B' = B - \{\phi\}$ is an $\mathcal{S}$-bias such that $N' = \mathrm{lfp}(\overline{\mathcal{S}[B']}^2, \emptyset)$ is contained in $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^2, \emptyset)$. $\square$

Proposition 9.9 *If there is $\mathcal{S} \sqsubseteq \mathbf{N}_X$ such that $T \subseteq \mathcal{L}^+(\mathcal{S})$ and $B \subseteq \mathcal{L}^-(\mathcal{S}) = L^-$, then the call $\mathsf{Bias}_X(T, B, L^-)$ returns* True *if and only if B is an $\mathcal{S}[T]$-bias and at most $|B|^2 \times (2 \times |L^-|^2 + 3 \times |L^-|)$ calls to the function TH_X is made.*

Proof. (i) First of all we consider the statements of the **For**-loop for a fixed value of B. The sets of non-theorems N and N' are computed by subsequent calls $\mathsf{Iterate}_X(T, B - \{\phi\}, L^-)$ and $\mathsf{NT}_X(T, (B - \{\phi\}) \cup N, L^-)$, respectively. By the correctness of the function $\mathsf{Iterate}_X$ (see Proposition 7.15) the variable N has the value $\mathrm{lfp}(\overline{\mathcal{S}[T; B - \{\phi\}]}^2, \emptyset)$. Then the variable N' has the value $\overline{\mathcal{S}[T; B - \{\phi\}]}(N)$ the correctness of NT_X (Proposition 7.12). Thus the test $\phi \in N'$ succeeds if and only if $\phi \in \overline{\mathcal{S}[T; B - \{\phi\}]}(N)$ for $N = \mathrm{lfp}(\overline{\mathcal{S}[T; B - \{\phi\}]}^2, \emptyset)$, i.e. the sentence ϕ is a bias for $\mathcal{S}[T; B - \{\phi\}]$.

Then let n be the size of B when the algorithm is entered and let B_n denote the initial value of the variable B. The value of B is only changed when the test $\phi \in N'$ succeeds and ϕ is removed from B. Thus we define B_{n-i} as the value of B when a total of i removals has been accomplished. We let ϕ_i denote the i^{th} sentence removed by the algorithm. By the analysis above, it is easy to see that each sentence ϕ_i removed is a bias for $\mathbb{S}[T; B_{n-i}]$.

(A) Let us assume that the call $\mathsf{Bias}_X(T, B, L^-)$ returns True. This implies that the final value of B is $\emptyset$ indicating that all sentences of B have been subsequently removed. Then define an enumeration $\varepsilon = \phi_n, \phi_{n-1}, \ldots, \phi_2, \phi_1$ (i.e. the sentences removed in the reverse order). Since ϕ_j is a bias for $\mathbb{S}[T; B_{n-j}]$ for all $0 < j \leq n$, it follows that each $\varepsilon_i = \phi_{(n+1)-i}$ where $0 < i \leq n$ is a bias for $\{\varepsilon_{0\ldots(i-1)}\} = \{\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_{i-1}\} = B_{i-1}$.[36] Thus ε satisfies the bias condition for $\mathbb{S}[T]$. Since ε enumerates the whole B_n, we know that B_n is an $\mathbb{S}[T]$-bias by Definition 8.7 and the return value True of the function Bias_X is correct. (B) Then consider the case that $\mathsf{Bias}_X(T, B, L^-)$ returns False. Then the algorithm must have stopped after $i < n$ removals. Also the last execution of the **For**-loop has not removed any sentences from B. This indicates that the test $\phi \in N'$ must have failed for each $\phi \in B_{n-i}$. Thus no $\phi \in B_{n-i}$ is a bias for $\mathbb{S}[T; B_{n-i} - \{\phi\}]$ by our above analysis. By Lemma 9.7, this means that B_{n-i} is not an $\mathbb{S}[T]$-bias. Let us then assume that B_n is an $\mathbb{S}[T]$-bias. We take this as a base case for our inductive argument that B_{n-j} where $0 \leq j \leq i$ is $\mathbb{S}[T]$-bias. Induction step follows. We know by the inductive hypothesis that $B_{n-(j-1)}$ is an $\mathbb{S}[T]$-bias. Since ϕ_j is a bias for $B_{n-(j-1)}$ and $B_{n-j} = B_{n-(j-1)} - \{\phi_j\}$, it follows by Proposition 9.8 that B_{n-j} is a also an $\mathbb{S}[T]$-bias. By this inductive proof, also B_i is an $\mathbb{S}[T]$-bias, a contradiction. Hence B_n is not an $\mathbb{S}$-bias and the return value False of the function Bias_X is correct. By the cases (A) and (B) analyzed above, the call $\mathsf{Bias}_X(T, B, L^-)$ returns True if and only if the set of non-theorems B is an $\mathbb{S}[T]$-bias.

(ii) The **Repeat – Until**-loop and the **For**-loop of the algorithm are executed at most n times. Therefore, the statements inside the **For**-loop are executed at most n^2 times. The execution of these statements involves further calls to $\mathsf{Iterate}_X$ and NT_X. By Propositions 7.15 and 7.12, these calls cause at most $2 \times |L^-|^2 + 2 \times |L^-|$ and $|L^-|$ calls to TH_X, respectively. Thus the upper bound for the number of calls to TH_X is $n^2 \times (2 \times |L^-|^2 + 3 \times |L^-|)$. □

Our next objective is to devise a decision procedure for cautious semantics. We start with a naive algorithm $\mathsf{NaiveCaut}_X$ given in Figure 16. It is based on four straightforward optimizations. (i) The set of non-theorems $\overline{\mathbb{S}[T]}(N)$ is computed for the fixed point $N = \mathrm{lfp}(\overline{\mathbb{S}[T]}^2, \emptyset)$. This is because $\mathbb{S}[T]$-biases B are always subsets of $\overline{\mathbb{S}[T]}(N)$ by Proposition 8.17. (ii) It is assumed that an $\mathbb{S}[T]$-bias B is finished and contains $M = \mathrm{lfp}(\overline{\mathbb{S}[T; B]}^2, \emptyset)$ so that

[36] Note that if $j = (n + 1) - i$, then $n - j = i - 1$.

$M = \overline{\mathcal{S}[T]}(B) \subseteq B$. This assumption is justifiable by Proposition 8.15 and Proposition 8.18. (iii) Rather than generating all enumerations $\overline{\mathcal{S}[T]}(N) - N$ and testing enumerations with the function EBias_X, the algorithm generates all subsets of $\overline{\mathcal{S}[T]}(N)$ that extend N and performs bias-tests with the function Bias_X. This is important, since the factorial function $n!$ (number of all enumerations of a set S of size n) grows much faster that the exponential function 2^n (number of all subsets of S). (iv) The query ϕ is employed to cut down the search space. The relatively expensive (but polynomial in number of calls to TH_X) bias test is run only if $\phi \notin \mathcal{S}[T](M)$, i.e. only if it is possible that B gives a counter-example for $\phi \in \mathrm{CAUT}_{\mathcal{S}}(T)$ to hold.

Let us then address the correctness of the algorithm. Note that the function $\mathsf{NaiveCaut}_X$ returns False if and only if a set of non-theorems $B \subseteq \overline{\mathcal{S}[T]}(N)$ is chosen such that (i) the return value M of $\mathsf{NT}_X(T, B, L^-)$ is contained in B, (ii) $\mathsf{TH}_X(T, M, \phi)$ returns False and $\mathsf{Bias}_X(T, B, L^-)$ returns True. By the correctness of NT_X, Bias_X and TH_X and the discussion above on the optimizations, this is the case if and only if (i) B is $\mathcal{S}[T]$-complete and $M \subseteq B$, (ii) $\phi \notin \mathcal{S}[T](M)$ and (iii) B is a finished $\mathcal{S}[T]$-bias so that $M = \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^2, \emptyset) = \overline{\mathcal{S}[T]}(B)$. To conclude, we have shown that the call $\mathsf{NaiveCaut}_X(T, \phi, L^-)$ returns True if and only if $\phi \in \mathrm{CAUT}_{\mathcal{S}}(T)$.

Function $\mathsf{NaiveCaut}_X(T, \phi, L^-)$: Boolean;
Var
  T: a finite subset of $\mathcal{L}^+(\mathbf{N}_X)$;
  B, B', N, N', M, L^-: finite subsets of $\mathcal{L}^-(\mathbf{N}_X)$;
  ϕ: a sentence of $\mathcal{L}^+(\mathbf{N}_X)$;
  A: Boolean
Begin
  $A :=$ True;
  $N := \mathsf{Iterate}_X(T, \emptyset, L^-)$;
  $N' := \mathsf{NT}_X(T, N, L^-)$;
  For $B' \subseteq N' - N$ **Do**
    Begin
      $B := N \cup B'$;
      $M := \mathsf{NT}_X(T, B, L^-)$;
      If $M \subseteq B$ **And Not** $\mathsf{TH}_X(T, M, \phi)$ **And** $\mathsf{Bias}_X(T, B, L^-)$
        Then $A :=$ False
    End;
  Return A
End

Figure 16: A naive decision procedure for $\mathbf{CAUT}_{\mathrm{L}}$

The function $\mathsf{NaiveCaut}_X$ constructs biases in "generate and test" fashion (guesses a candidate B and runs the bias test) rather than bottom-up (constructs sets B that are biases). From the point of view of approximation techniques to be presented in Section 9.3, it would be preferable to use a bottom-up algorithm. To build such an algorithm, we have the following techniques in mind in addition to those already discussed.

- To avoid explicit construction of all enumerations of $\mathcal{L}^-(\mathbb{S})$, we can utilize a binary search tree for the bias candidates B. The two branches leading from a node n are labeled with $+\phi$ and $-\phi$ for a sentence $\phi \in \mathcal{L}^-(\mathbb{S}[T])$ which does not appear in any labels on the path from the root of the tree to the node n. The label $+\phi$ means that $\phi \in B$ while the label $-\phi$ means that $\phi \notin B$. We identify a path p with the set of labels on it. A path p is complete if it contains either $+\phi$ or $-\phi$ for all $\phi \in \mathcal{L}^-(\mathbb{S}[T])$. For a path p, we define $p^+ = \{\phi \mid +\phi \in p\}$ and $p^- = \{\phi \mid -\phi \in p\}$. Thus $p^+ \cup p^- = \mathcal{L}^-(\mathbb{S}[T])$ holds for a complete path p in the search tree.

- Stationary semantics should be used as an approximation as far as it is possible [85]. More precisely, a node n at the end of a path p should be labeled with a least fixed point such as $N = \mathrm{lfp}(\overline{\mathbb{S}[T;B]}^2, \emptyset)$ where $B = p^+$.

- Pruning techniques try to cut down the search tree at a node n by the information provided by a path p leading to n, the least fixed point computed for the node n and bias tests using Bias_X.

- Algorithm computes only one branch of the search tree at a time to save space considerably in cases where there is an exponential number solutions.

It seems non-trivial to apply these ideas simultaneously. Let us sketch the difficulties briefly here. (i) In Example 8.29 when $T = \emptyset$ we might have a path $p = \{+(\neg\neg\mathrm{b})\}$ in the binary search tree. This implies that $\neg\neg\mathrm{b} \in B$. However, if we compute $N = \mathrm{lfp}(\overline{\mathbb{S}[B]}^2, \emptyset)$ for $B = \{\neg\neg\mathrm{b}\}$ we obtain $N = \{\neg\neg\mathrm{b}\}$ and $\overline{\mathbb{S}[B]}(N) = \{\neg\neg\mathrm{b}\}$ also. This *suggests* that there are no more biases extending B and thus no reason to extend p (e.g. with $+(\neg\neg\mathrm{a})$). But this not the case: also $B = \{\neg\neg\mathrm{a}, \neg\neg\mathrm{b}\}$ is an $\mathbb{S}$-bias. Hence the approximation $N = \mathrm{lfp}(\overline{\mathbb{S}[T;B]}^2, \emptyset)$ is too strong to be computed for the nodes.

(ii) The other difficulty arises if we try to utilize p^- in the approximation, i.e. sentences ϕ such that $\phi \notin B$. A possibility is to use an operator $\overline{\mathbb{S}^C}$ defined by $\overline{\mathbb{S}^C}(N) = \overline{\mathbb{S}}(N - C)$ This is an anti-monotonic operator so that the operator $\overline{\mathbb{S}^C}^2$ is monotonic. Thus $N = \mathrm{lfp}(\overline{\mathbb{S}[T]^C}^2, \emptyset)$ could be used as an approximation by setting $C = p^-$ to utilize information obtained from the path p. In Example 8.29 with $T = \emptyset$ and $p = \{-(\neg\neg\mathrm{b})\}$ we get $C = \{\neg\neg\mathrm{b}\}$

and $N = \{\neg\neg\mathrm{a}, \neg\neg\mathrm{b}\}$. Since $N \subseteq B$ is assumed, this could be interpreted as a contradiction ($\neg\neg\mathrm{b} \in B$) indicating that there is no finished $\mathcal{S}$-bias B such that $\neg\neg\mathrm{b} \notin B$. In fact, this is the case. Unfortunately, we are in trouble with Example 8.45. With $T = \emptyset$ and $p = \{-\mathrm{p}\}$ we get $C = \{\mathrm{p}\}$ and $N = \mathrm{lfp}(\overline{\mathcal{S}^C}^2, \emptyset) = \{\mathrm{p}, \mathrm{q}\}$. However, if we now conclude that there are no finished $\mathcal{S}$-biases B such that $\mathrm{p} \notin B$, we are making a wrong conclusion, since there is a finished $\mathcal{S}$-bias $B = \{\mathrm{r}, \mathrm{q}\}$ for $\mathcal{S}$.

(iii) The third problem is that the fact that B is an $\mathcal{S}$-bias does not necessarily imply that $B' \subseteq B$ is an $\mathcal{S}$-bias. To see this, consider an autoepistemic theory $\Sigma = \{\mathbf{D}\mathrm{b} \to \mathrm{a}, \mathbf{D}\mathrm{b} \to \mathrm{c}, \mathbf{D}\mathrm{a} \wedge \mathrm{c} \to \mathrm{b}, \mathbf{D}\mathrm{a} \wedge \mathbf{D}\mathrm{b} \to \mathrm{d}, \mathbf{D}\mathrm{b} \wedge \mathbf{D}\mathrm{d} \to \mathrm{e}, \mathbf{D}\mathrm{e} \to \mathrm{b}\}$. Then $B = \{\mathrm{a}, \mathrm{b}, \mathrm{e}\}$ is an $\mathcal{S}$-bias for the system $\mathcal{S} = \mathcal{A}^{\Sigma}_{\Sigma}$, but $B' = \{\mathrm{b}, \mathrm{e}\}$ is not. Therefore, if we detect that $B = p^{+}$ is *not* an $\mathcal{S}$-bias we cannot simply cut totally the search tree from that particular node. Anyway, it seems that the result of Proposition 9.8 can provide some pruning information in this case.

```
Function Caut_X(T, φ, B, C, R, L⁻): Boolean;
Var
  T: a finite subset of ℒ⁺(𝒩_X);
  B, C, R, N, L⁻: finite subsets of ℒ⁻(𝒩_X);
  φ: a sentence of ℒ⁺(𝒩_X);
  ψ: a sentence of ℒ⁻(𝒩_X);
  A: Boolean
Begin
  A := True;
  If R = ∅ Then Begin
    N = NT_X(T, B, L⁻);
    If N ⊆ B And Not TH_X(T, N, φ) And Bias(T, B, L⁻)
      Then A := False
  End Else Begin
    Choose ψ From R;
    A := Caut_X(T, B ∪ {ψ}, C, R − {ψ}, φ, L⁻)
         And Caut_X(T, B, C ∪ {ψ}, R − {ψ}, φ, L⁻)
  End;
  Return A
End
```

Figure 17: A skeletal decision procedure for $\mathbf{CAUT}_{\mathrm{L}}$

Finally, we present a skeletal bottom-up algorithm in Figure 17. It is assumed that this algorithm is invoked with $\mathsf{Caut}_X(T, \phi, \emptyset, \emptyset, R, L^{-})$ where R is initialized as $\overline{\mathcal{S}[T]}(N)$ for $N = \mathrm{lfp}(\overline{\mathcal{S}[T]}^2, \emptyset)$ and the language $\mathcal{L}^{-}(\mathcal{S})$ is passed as the argument L^{-}. The set B is the bias under construction while the set C is its complement relative R. Note that the sets B and C correspond the sets

p^+ and p^- (respectively) for a path p in the search tree. Due to difficulties (i-iii) discussed above we have not incorporated a stationary approximation in the algorithm to prune search space. We leave finding an approximation that utilizes the information provided by B and C as an open research problem.

9.3 Gradually Refining Approximations

In this section, we propose an approximation method for cautious semantics. Since we founded cautious semantics on the notion of biases, it is natural that we try to approximate them. Our idea is to impose a restriction on sizes of biases to be taken into account when biases are computed. In this way, we are expecting to gain computational advantage since the number of candidates for biases is smaller when size restrictions are introduced for biases. It is to be expected that all queries cannot be answered on the basis of approximations. Therefore, it would be useful if a range of gradually refining approximations could be identified. Then the more computation time is allocated for refining the current approximation, the more queries can be answered. On the other hand, approximating cautious semantics may not be easy as indicated by recent results by Gottlob and Truszczyński [40] on approximating the stable semantics of normal logic programs: it is **NP**-hard to decide whether the size of a polynomially computed approximation of the intersection of stable models is within a constant factor from the size of the intersection.

We proceed as follows. In Section 9.3.1, we introduce a size restriction for biases and define the resulting semantics for non-monotonic systems. It is then shown in Section 9.3.2 that in fact we have obtained a range of gradually refining semantics for a non-monotonic system $\mathbb{S}$. Algorithms for computing approximations are briefly considered in Section 9.3.3.

9.3.1 Establishing a Range of Approximations

By Definition 9.10 we limit the size of biases $B \subseteq \mathcal{L}^-(\mathbb{S})$ to be at most α. This parameter α is defined to be a cardinal number in the range $0 \leq \alpha \leq |\mathcal{L}^-(\mathbb{S})|$. Therefore, the parameter α will always correspond to a natural number for non-monotonic systems $\mathbb{S}$ with a finite language $\mathcal{L}^-(\mathbb{S})$ for non-theorems. Note that an $\mathbb{S}$-bias B with size less than α is α-finished if and only if B is finished. On the other hand, $\mathbb{S}$-biases with $|B| = \alpha$ are α-finished were they finished or not. Nevertheless, we know by Theorem 8.14 that such $\mathbb{S}$-biases can be extended to finished ones.

Definition 9.10 *An $\mathbb{S}$-bias B is α-finished if and only if (i) B is finished and $|B| < \alpha$ or (ii) $|B| = \alpha$.*

With the definition of α-finished biases available we may proceed to the resulting semantic operator $\mathrm{CAUT}^{\alpha}_{\mathcal{S}}$ where the parameter α ranges within $0 \leq \alpha \leq |\mathcal{L}^{-}(\mathcal{S})|$ in correspondence to the definition of α-finished biases.

Definition 9.11 *The operator* $\mathrm{CAUT}^{\alpha}_{\mathcal{S}}$ *for the* α*-approximation of the cautious semantics of* $\mathcal{S}$ *is defined for all* $T \subseteq \mathcal{L}^{+}(\mathcal{S})$ *by*

$$\mathrm{CAUT}^{\alpha}_{\mathcal{S}}(T) = \bigcap\{\mathcal{S}(T,N) \mid B \text{ is an } \alpha\text{-finished } \mathcal{S}[T]\text{-bias and } N = \mathrm{lfp}(\overline{\mathcal{S}[T;B]}^{2}, \emptyset)\}$$

Having set up the approximations parametrized by α for cautious semantics, we provide a couple of examples. Example 9.12 illustrates how biases of different sizes appear in practise. However, note that the conclusions provided by the approximations for $T = \emptyset$ are $\mathrm{CAUT}^{0}_{\mathcal{S}}(T) = \mathrm{CAUT}^{1}_{\mathcal{S}}(T) = \emptyset$ and $\mathrm{CAUT}^{2}_{\mathcal{S}}(T) = \mathrm{CAUT}^{3}_{\mathcal{S}}(T) = \{\mathrm{r}\}$. So r can be concluded once the parameter α becomes greater than the threshold value 1. In Example 9.13, we provide a system where the set of conclusions refines gradually as α grows. It seems that this is because all sentence included in the biases are realized as non-theorems (i.e. biases B are contained in the fixed points $N = \mathrm{lfp}(\overline{\mathcal{S}[B]}^{2}, \emptyset)$ induced) in this example. This suggests that our notion of approximations could be useful for at least minimal model systems $\mathfrak{C}_P$ for which this is always the case.

Example 9.12 Consider a set of autoepistemic sentences $\Sigma = \{\mathbf{D}\mathrm{p} \rightarrow \mathrm{q}_1 \wedge \mathrm{q}_2, \mathbf{D}\mathrm{q}_1 \wedge \mathbf{D}\mathrm{q}_2 \rightarrow \mathrm{p}, \mathbf{B}\mathrm{q}_1 \rightarrow \mathrm{r}, \mathbf{B}\mathrm{q}_2 \rightarrow \mathrm{r}, \mathbf{B}\mathrm{p} \rightarrow \mathrm{r}\}$ and the non-monotonic system $\mathcal{S} = \mathcal{A}^{\Sigma \cup \{\mathbf{B}\mathrm{r}\}}_{\Sigma}$ having the language $\langle\{\mathrm{p}, \mathrm{q}_1, \mathrm{q}_2, \mathrm{r}\}, \{\mathrm{p}, \mathrm{q}_1, \mathrm{q}_2\}\rangle$. Thus $\mathcal{S}$-biases are subsets of $\{\mathrm{p}, \mathrm{q}_1, \mathrm{q}_2\}$. It is clear that $B_0 = \emptyset$ is the only 0-finished $\mathcal{S}$-bias. There are three 1-finished $\mathcal{S}$-biases, namely $B_1 = \{\mathrm{p}\}$, $B_2 = \{\mathrm{q}_1\}$ and $B_3 = \{\mathrm{q}_2\}$. The first one of these is finished while the last two are not. There are two 2-finished $\mathcal{S}$-biases, the sets B_1 and $B_{23} = \{\mathrm{q}_1, \mathrm{q}_2\}$. The 3-finished $\mathcal{S}$-biases are the same as 2-finished $\mathcal{S}$-biases. □

Example 9.13 Consider sets of autoepistemic sentences

$$\Sigma = \{\mathbf{D}\mathrm{p}_1 \rightarrow \mathrm{q}_1, \mathbf{D}\mathrm{q}_1 \rightarrow \mathrm{p}_1, \mathbf{D}\mathrm{p}_2 \rightarrow \mathrm{q}_2, \mathbf{D}\mathrm{q}_2 \rightarrow \mathrm{p}_2\}$$

and $T = \{\mathrm{p}_1 \vee \mathrm{q}_1 \rightarrow \mathrm{r}_1, \mathrm{p}_2 \vee \mathrm{q}_2 \rightarrow \mathrm{r}_2\}$ as well as the system $\mathcal{S} = \mathcal{A}^{\Sigma \cup \{\mathbf{B}\mathcal{L}\}}_{\Sigma}$ with the language $\mathcal{L}^{-}(\mathcal{S}) = \{\mathrm{p}_1, \mathrm{q}_1, \mathrm{p}_2, \mathrm{q}_2\}$. The only 0-finished $\mathcal{S}[T]$-bias $B = \emptyset$ provides us the set of conclusions $\mathrm{Cn}(T) \cap \mathcal{L}$, i.e. $\mathrm{r}_1 \vee \mathrm{r}_2$ and $\mathrm{r}_1 \wedge \mathrm{r}_2$ cannot be concluded. The 1-finished $\mathcal{S}[T]$-biases give rise to the least stationary fixed points $N_1 = \{\mathrm{p}_1\}$, $N_2 = \{\mathrm{p}_2\}$, $N_3 = \{\mathrm{q}_1\}$ and $N_4 = \{\mathrm{q}_2\}$. Upon these $\mathrm{r}_1 \vee \mathrm{r}_2$ can be concluded, but not $\mathrm{r}_1 \wedge \mathrm{r}_2$. The 2-finished $\mathcal{S}[T]$-biases induce fixed points N which contain one non-theorem from $\{\mathrm{p}_1, \mathrm{q}_1\}$ and the other from $\{\mathrm{p}_2, \mathrm{q}_2\}$. On the basis of these, both $\mathrm{r}_1 \vee \mathrm{r}_2$ and $\mathrm{r}_1 \wedge \mathrm{r}_2$ can be concluded. □

9.3.2 Relationship to Stationary and Cautious Semantics

Our next objective is to relate the α-approximations of cautious semantics to other semantics. It is not surprising that the weakest approximation $\mathrm{CAUT}^0_{\mathbb{S}}$ gives us exactly the stationary semantics $\mathrm{STAT}_{\mathbb{S}}$. In Theorem 9.15 it is shown that the approximations of $\mathrm{CAUT}_{\mathbb{S}}(T)$ refine while α grows.

Theorem 9.14 $\mathrm{CAUT}^0_{\mathbb{S}} = \mathrm{STAT}_{\mathbb{S}}$ *for all non-monotonic systems* $\mathbb{S}$.[37]

Proof. For all $\mathbb{S}$, the only 0-finished $\mathbb{S}[T]$-bias is $B = \emptyset$. This induces the set of non-theorems $N = \mathrm{lfp}(\overline{\mathbb{S}[T;B]}^2, \emptyset) = \mathrm{lfp}(\overline{\mathbb{S}[T]}^2, \emptyset)$. Thus $\mathrm{CAUT}^0_{\mathbb{S}}(T) = \mathbb{S}[T](N) = \mathrm{STAT}_{\mathbb{S}}(T)$ by Definition 9.10 and Theorem 4.28. □

Theorem 9.15 *For all non-monotonic systems* $\mathbb{S}$ *and for all cardinal numbers* α_1 *and* α_2 *in the range* $0 \leq \alpha_1 \leq \alpha_2 \leq |\mathcal{L}^-(\mathbb{S})|$, $\mathrm{CAUT}^{\alpha_1}_{\mathbb{S}} \sqsubseteq \mathrm{CAUT}^{\alpha_2}_{\mathbb{S}}$.

Proof. Let $T \subseteq \mathcal{L}^+(\mathbb{S})$ be any set of theorems and $\phi \in \mathcal{L}^+(\mathbb{S})$ a sentence. Then assume that $\phi \notin \mathrm{CAUT}^{\alpha_2}_{\mathbb{S}}(T)$, i.e. there is an α_2-finished $\mathbb{S}$-bias B such that $\phi \notin \mathbb{S}[T](N)$ where $N = \mathrm{lfp}(\overline{\mathbb{S}[T;B]}^2, \emptyset)$. Two cases arise. (i) If B is also α_1-finished, then $\phi \notin \mathrm{CAUT}^{\alpha_1}_{\mathbb{S}}(T)$ follows directly. (ii) If B is not α_1-finished, it must be the case that $\alpha_1 < |B|$. By Definition 8.7, there is an enumeration ε of B that satisfies the bias condition for $\mathbb{S}$. Define B' as $\{\varepsilon_{0\ldots\beta}\}$ where $0 \leq \beta \leq \mathrm{Ind}(\varepsilon)$ is the least ordinal such that $|\beta| = \alpha_1$. This implies that $|B'| = \alpha_1$ and B' is an $\mathbb{S}$-bias such that $B' \subseteq B$ and $N' = \mathrm{lfp}(\overline{\mathbb{S}[T;B']}^2, \emptyset)$ is a subset of N. It follows that $\phi \notin \mathbb{S}[T](N')$ by the requirement MON. Since B' is α_1-finished, it holds that $\phi \notin \mathrm{CAUT}^{\alpha_1}_{\mathbb{S}}(T)$. By the cases (i) and (ii), it follows that $\phi \notin \mathrm{CAUT}^{\alpha_1}_{\mathbb{S}}(T)$. Hence, $\mathrm{CAUT}^{\alpha_1}_{\mathbb{S}}(T) \subseteq \mathrm{CAUT}^{\alpha_2}_{\mathbb{S}}(T)$. □

The relationship to cautious semantics is examined next. Theorem 9.16 shows that our approximations are complete in the sense that for a sufficiently large α (i.e. the cardinality of $\mathcal{L}^-(\mathbb{S})$), the resulting approximation coincides with cautious semantics. The whole idea of approximating cautious semantics relies on the intuition that $\mathrm{CAUT}^{\alpha}_{\mathbb{S}}$ could be useful already for considerably small values of α as demonstrated in Example 9.13.

Theorem 9.16 *For all non-monotonic systems* $\mathbb{S}$, $\mathrm{CAUT}^{\alpha}_{\mathbb{S}} = \mathrm{CAUT}_{\mathbb{S}}$ *when* α *reaches the value* $|\mathcal{L}^-(\mathbb{S})|$.

Proof. Let $\alpha = |\mathcal{L}^-(\mathbb{S})|$. Every finished $\mathbb{S}$-bias $B \subseteq \mathcal{L}^-(\mathbb{S})$ is also a α-finished $\mathbb{S}$-bias, since $B \subseteq \mathcal{L}^-(\mathbb{S})$ implies that $|B| \leq \alpha$. On the other hand, every

[37]The equivalence of two semantic operators $\mathrm{SEM}^1_{\mathbb{S}}$ and $\mathrm{SEM}^2_{\mathbb{S}}$ means that $\mathrm{SEM}^1_{\mathbb{S}} \sqsubseteq \mathrm{SEM}^2_{\mathbb{S}}$ $\mathrm{SEM}^2_{\mathbb{S}} \sqsubseteq \mathrm{SEM}^1_{\mathbb{S}}$, i.e. $\mathrm{SEM}^1_{\mathbb{S}}(T) = \mathrm{SEM}^2_{\mathbb{S}}(T)$ for all $T \subseteq \mathcal{L}^+(\mathbb{S})$.

α-finished $\mathbb{S}$-bias B is necessarily a finished $\mathbb{S}$-bias by Definition 9.10. The definitions of $\mathrm{CAUT}^{\alpha}_{\mathbb{S}}$ and $\mathrm{CAUT}_{\mathbb{S}}$ coincide under this equivalence of finished $\mathbb{S}$-biases and α-finished $\mathbb{S}$-biases. $\square$

Soundness of the approximations is established as follows.

Corollary 9.17 *For all non-monotonic systems $\mathbb{S}$ and for all cardinal numbers α in the range $0 \leq \alpha \leq |\mathcal{L}^-(\mathbb{S})|$, $\mathrm{CAUT}^{\alpha}_{\mathbb{S}} \sqsubseteq \mathrm{CAUT}_{\mathbb{S}}$.*

Proof. This follows from Theorems 9.15 and 9.16 with $\alpha_2 = |\mathcal{L}^-(\mathbb{S})|$. $\square$

Finally, we relate our approximations to stationary and stable semantics:

Corollary 9.18 *For all non-monotonic systems $\mathbb{S}$ and for all cardinal numbers α in the range $0 \leq \alpha \leq |\mathcal{L}^-(\mathbb{S})|$, $\mathrm{STAT}_{\mathbb{S}} \sqsubseteq \mathrm{CAUT}^{\alpha}_{\mathbb{S}} \sqsubseteq \mathrm{STAB}_{\mathbb{S}}$.*

Proof. This follows from Theorems 9.14 and 9.15 with $\alpha_1 = 0$ as well as Corollary 9.17 and Theorem 8.21 where $\mathrm{CAUT}_{\mathbb{S}} \sqsubseteq \mathrm{STAB}_{\mathbb{S}}$ was established. $\square$

Finally, let us depict the resulting chain of gradually refining approximations:

$$\mathrm{STAT}_{\mathbb{S}} = \mathrm{CAUT}^{0}_{\mathbb{S}} \sqsubseteq \ldots \sqsubseteq \mathrm{CAUT}^{\alpha}_{\mathbb{S}} \sqsubseteq \ldots \sqsubseteq \mathrm{CAUT}^{|\mathcal{L}(\mathbb{S})|}_{\mathbb{S}} = \mathrm{CAUT}_{\mathbb{S}}.$$

9.3.3 Algorithms for Computing Approximations

The size limit k of biases can be easily taken into account when biases are computed in a bottom-up fashion. This is the case for the function Caut_X in Figure 17. Let us then discuss the modifications to the algorithm. (i) The fixed point $N = \mathrm{lfp}(\overline{\mathbb{S}[T;B]}^2, \emptyset)$ should be computed instead of $N = \overline{\mathbb{S}[T]}(B)$, since the biases B of our interest are not necessarily finished any more. (ii) The test $R = \emptyset$ should be replaced by a completeness test conforming to Definition 9.11. Since $\overline{\mathbb{S}[B]}(N)$ is no longer available (see (i)), the test becomes ($|B| < k$ **And** $\mathsf{NT}_X(T, B \cup N, L^-) \subseteq B \cup N$) **Or** $|B| = k$ for k-finished biases. (iii) The search tree should not be extended when $|B|$ has reached k. This affects the branching at the end of algorithm: if $|B| < k$, the search tree is extended as before, but otherwise Caut_X can return True directly.

10 Applications

This section considers possible applications of the semantics presented in Sections 4 and 8. In particular, we are interested in what role cautious semantics has in various application domains. The domains addressed in Sections 10.1 and 10.2 are *logic programming* [61] and *consistency-based diagnosis* [94], respectively. Our analysis is basically restricted to the propositional case. However, our results extend easily for applications where variables are used, but a sentence containing variables is seen as shorthand for all *instances* of the sentence over a finite domain of objects. For example, the sentences of Example 2.35 were obtained by instantiation from the rules introduced for the reasoning system of Example 2.5. The instantiation procedure yields effectively finite sets of propositional sentences and consequently the non-monotonic systems with propositional languages are applicable.

10.1 Logic Programming

The field of logic programming has emerged from the idea of using logic for programming purposes [114]. Gradually this idea has developed into full-fledged programming paradigms such as PROLOG [105]. To enhance the knowledge representation capabilities of logic programs, Clark [13] proposed a new form of negation for logic programs: *negation as failure to prove*. This kind of negation was easy to implement into resolution theorem provers used as interpreters of logic programs (see [61] for a detailed treatment of SLDNF-resolution). However, it is non-trivial to provide a formal semantics for such a form of negation in order to justify the answers provided by an interpreter. Basically, there have been two approaches to this problem. The first approach tries to give a semantics directly in logic programming terms. For instance, Clark [13] proposes a systematic procedure to convert the logical implications corresponding to the rules of a logic program into logical equivalences (so called *Clark's completion*). The second possibility is to define the semantics of the negation as failure using the primitives of non-monotonic logics.

In the following, we analyze how the variant of autoepistemic logic proposed in this work could give a semantics for negation as failure. We proceed as follows. In Section 10.1.1, we introduce a very general class of logic programs called *hyper logic programs* and consider two ways of interpreting the rules of such programs as autoepistemic sentences. Then the semantics of hyper logic programs can be defined using the resulting translations of hyper logic programs and autoepistemic systems. In Section 10.1.2, we concentrate on a subclass of hyper logic programs called *normal logic programs* and discuss the role of stationary semantics, stable semantics and cautious semantics in this context. In Section 10.1.3, we present *program transformations* that have been used to justify that particular semantics are reasonable. The above mentioned semantics of normal logic programs are shown to be invariant under these transformations. Other autoepistemic characterizations of the stationary semantics of normal logic programs are briefly discussed in Section 10.1.4. Then some extensions of normal logic programs are considered. The possibilities of incorporating classical negation in the program rules are addressed in Section 10.1.5. As a final step, we study in Section 10.1.6 what kind of semantics is obtained for *disjunctive* logic programs under our autoepistemic interpretation of the program rules. Limitations of conventional approaches – due to unrestricted minimisation – are pointed out. A short comparison with disjunctive well-founded semantics [9] and static semantics [10] is made.

10.1.1 Hyper Logic Programs

In the following, we introduce a very general class of logic programs to be called *hyper logic programs.* Then three interesting subclasses of hyper logic programs are presented: *normal*, *extended* and *disjunctive* logic programs. Subsequently, the framework of non-monotonic systems is applied to give a semantics for these classes of logic programs. This requires us to identify the non-monotonic systems associated with hyper logic programs. Two ways of such an identification are considered. Our main approach gives an autoepistemic interpretation for the rules of a hyper logic program and uses autoepistemic systems. The other possibility is to define the non-monotonic system associated with a hyper logic program directly. Such a formulation is also presented to allow comparison with the autoepistemic interpretation. Let us now proceed and present the class of hyper logic programs.

Definition 10.1 *A* hyper logic program *P is a set of rules of the form*

$$\phi \leftarrow \phi_1, ..., \phi_n, {\sim}\psi_1, ..., {\sim}\psi_m$$ [38]

where $n \geq 0$ and $m \geq 0$ and where ϕ, $\phi_1, ..., \phi_n$ and $\psi_1, ..., \psi_m$ are propositional sentences and the symbol $\sim$ presents negation as failure to derive*.*

The intuition behind a rule of a hyper logic program P is that if the *positive subgoals* $\phi_1, ..., \phi_n$ of the rule are derivable by the rules of P but not the *negative subgoals* $\psi_1, ..., \psi_m$ of the rule, then the *head* ϕ can be derived.[39] The *Herbrand base* $\mathrm{Hb}(P)$ of a logic program P is the set of propositional atoms that appear in the rules of P. In the sequel, we will consider three subclasses

[38] The order of the subgoals is assumed to be irrelevant.

[39] There is inherent circularity in this kind of interpretation of negative subgoals.

of hyper logic programs which are obtained by imposing syntactic restrictions on the propositional sentences that appear as heads and positive and negative subgoals in the rules. The classes of our interest are the following:

1. *Normal logic programs* which contain only propositional atoms $\mathrm{a}, \mathrm{b}, \ldots$.
2. *Extended logic programs* [33] which contain only *positive or negative literals* that are propositional atoms $\mathrm{a}, \mathrm{b}, \ldots$ or their negations $\neg\mathrm{a}, \neg\mathrm{b}, \ldots$.
3. *Disjunctive logic programs* [62] which contain only disjunctions $\mathrm{a}_1 \vee ... \vee \mathrm{a}_n$ of propositional atoms $\mathrm{a}_1, ..., \mathrm{a}_n$.

Definition 10.1 covers all these three classes of logic programs. Even further classes of logic programs can be extracted from this definition by introducing other syntactic restrictions. In fact, our notion of hyper logic programs is even slightly more general than that of *super logic programs* proposed by Brass, Dix and Przymusinski [10], since they restrict negative subgoals to be *positive sentences*, i.e. arbitrary disjunctions of conjunctions of atoms. These observations justify the name "hyper logic programs" given in Definition 10.1.

Having set up the syntax of hyper logic programs, our next concern is their semantics. We would like to apply our framework of non-monotonic systems to give semantics for hyper logic programs. Therefore, we should identify the non-monotonic system $\mathcal{H}_P$ associated with a hyper logic program P. Once such an identification has been carried out, various semantics defined in Sections 4 and 8 cover hyper logic programs immediately. Moreover, the properties of semantics established from the specification of non-monotonic systems carry over. Basically, we have two ways of obtaining non-monotonic systems for hyper logic programs. (i) A possibility is to define the binary inference operator $\mathcal{H}_P$ associated with a hyper logic program P directly on the basis of Definition 10.1. (ii) An alternative approach is to take advantage of the non-monotonic systems such as default systems $\mathcal{D}_D$ or autoepistemic systems $\mathcal{A}_\Sigma$ introduced in Section 2.3. This approach involves representing a hyper logic program P as a set of defaults D or as an autoepistemic theory Σ. In fact, both approaches are considered in the sequel. To demonstrate the applicability of the bimodal autoepistemic logic introduced in this work, we present an autoepistemic interpretation of a hyper logic program P and use autoepistemic systems to establish $\mathcal{H}_P$ indirectly. On the basis of this definition, we extract a direct definition at the end of this section.[40]

To give an autoepistemic interpretation for hyper logic programs, we have to decide how the rules of a hyper logic program P are translated into autoepistemic sentences. Generally speaking, two main approaches arise from the

[40]The latter definition may be helpful for a reader who is familiar with logic programs and propositional logic, but unfamiliar with autoepistemic logic.

translations that have been presented in the literature. Gelfond and Lifschitz [32] translate rules $\phi \leftarrow \phi_1, ..., \phi_n, \sim\psi_1, ..., \sim\psi_m$ of a normal logic program into autoepistemic sentences $\neg\mathbf{B}\psi_1 \wedge ... \wedge \neg\mathbf{B}\psi_m \wedge \phi_1 \wedge ... \wedge \phi_n \rightarrow \phi$ to define *stable models* for normal logic programs via the stable expansions of the translation. This translational idea gives rise to the translation function given in Definition 10.2. Some remarks are worth making. First of all, we generalize this definition to hyper logic programs, i.e. the restriction to normal logic programs is simply forgotten. Secondly, the modality $\mathbf{D}$ is used for referring to disbeliefs rather than $\neg\mathbf{B}$. Finally, it is possible to consider the iterative expansions of the translation instead of stable expansions, since the translation is free of $\mathbf{B}$-atoms. This is important, since autoepistemic systems were derived from iterative expansions rather than stable expansions in Section 2.3.

Definition 10.2 *The autoepistemic interpretation* $\mathrm{Tr}_{\mathrm{GL}}(P)$ *of a hyper logic program is obtained by translating each rule* $\phi \leftarrow \phi_1, ..., \phi_n, \sim\psi_1, ..., \sim\psi_m$ *of* P *into an autoepistemic sentence* $\mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \phi_1 \wedge ... \wedge \phi_n \rightarrow \phi$.

Another translation function is obtained by composing a translation function proposed by Bidoit and Froidevaux [5] with the translation function Tr_{K} given in Definition 3.29. Bidoit and Froidevaux translate a rule $\phi \leftarrow \phi_1, ..., \phi_n, \sim\psi_1, ..., \sim\psi_m$ of a normal logic program into a default

$$\frac{\phi_1 \wedge ... \wedge \phi_n : \neg\psi_1, ..., \neg\psi_m}{\phi}.$$

When this translation function is composed with Tr_{K}, the total effect is that a rule given above is translated into an autoepistemic sentence $\mathbf{D}\neg\neg\psi_1 \wedge ... \wedge \mathbf{D}\neg\neg\psi_m \wedge \mathbf{B}(\phi_1 \wedge ... \wedge \phi_n) \rightarrow \phi$. This is practically equivalent to $\mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \mathbf{B}\phi_1 \wedge ... \wedge \mathbf{B}\phi_n \rightarrow \phi$ if the iterative expansions of the translation are considered. In analogy to Definition 10.2, we generalize this translation function for hyper logic programs as follows.

Definition 10.3 *The autoepistemic interpretation* $\mathrm{Tr}_{\mathrm{BFK}}(P)$ *of a hyper logic program is obtained by translating each rule* $\phi \leftarrow \phi_1, ..., \phi_n, \sim\psi_1, ..., \sim\psi_m$ *of* P *into an autoepistemic sentence* $\mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \mathbf{B}\phi_1 \wedge ... \wedge \mathbf{B}\phi_n \rightarrow \phi$.

The major difference between the translations $\mathrm{Tr}_{\mathrm{GL}}(P)$ and $\mathrm{Tr}_{\mathrm{BFK}}(P)$ is that the latter translation enforces the inference rule character of program rules while the former does not. To define the corresponding non-monotonic systems $\mathfrak{H}_P$ and $\mathfrak{I}_P$ for a hyper logic program P directly, we have to introduce two reductions for P to interpret the negative subgoals.[41] Given $N \subseteq \mathcal{L}$, we let P^N denote the set of implications $\phi_1 \wedge ... \wedge \phi_n \rightarrow \phi$ for which there is a rule

[41]Note the similarity with the reduction D_N introduced for sets of defaults D.

$\phi \leftarrow \phi_1, ..., \phi_n, \sim\psi_1, ..., \sim\psi_m$ of P such that $\{\psi_1, ..., \psi_m\} \subseteq N$. Similarly, we let P_N denote the corresponding set of inference rules $\frac{\phi_1, \ldots, \phi_n}{\phi}$.

Definition 10.4 *For a hyper logic program P, define binary inference operators $\mathfrak{H}_P$ and $\mathfrak{I}_P$ by setting $\mathcal{L}(\mathfrak{H}_P) = \mathcal{L}(\mathfrak{I}_P) = \langle \mathcal{L}, \mathcal{L} \rangle$ and for all $T \subseteq \mathcal{L}$ and $N \subseteq \mathcal{L}$, $\mathfrak{H}_P(T, N) = \mathrm{Cn}(T \cup P^N)$ and $\mathfrak{I}_P(T, N) = \mathrm{Cn}^{P_N}(T)$.*

A few observations about this definition are necessary. Firstly, if syntactic restrictions are imposed, it seems reasonable to restrict the language $\langle \mathcal{L}, \mathcal{L} \rangle$ accordingly. For instance, if a normal logic program P is considered, it would be natural to use the language $\langle \mathrm{Hb}(P), \mathrm{Hb}(P) \rangle$ based on the atoms appearing in P. Secondly, only the negative subgoals of program rules are significant when the reductions P^N and P_N are determined. This implies that the non-monotonic systems $\mathfrak{H}_P$ and $\mathfrak{I}_P$ above have subsystems $\mathfrak{S}$ where the language for non-theorems $\mathcal{L}^-(\mathfrak{S})$ contains only sentences that appear in the negative subgoals of P. Such systems provide, e.g., a finitary characterization of $\mathfrak{H}_P$-stable sets of non-theorems and thus a basis of implementing the stable semantics of hyper logic programs. Finally, we have presented the non-monotonic systems $\mathfrak{H}_P$ and $\mathfrak{I}_P$ here for the sake of comparison and to point out that such systems can be directly extracted for hyper logic programs. Nevertheless, the translations presented and autoepistemic systems of hyper logic programs will be used in our subsequent analysis rather than the systems $\mathfrak{H}_P$ and $\mathfrak{I}_P$.

10.1.2 Normal Logic Programs

Let us now restrict our attention to *normal logic programs*, i.e. programs where only propositional atoms appear in the rules. Thus the language associated with a normal logic program P is $\mathrm{Hb}(P)$. It is established in Proposition 10.5 that the autoepistemic systems associated with the translations $\mathrm{Tr}_{\mathrm{GL}}(P)$ and $\mathrm{Tr}_{\mathrm{BFK}}(P)$ of a normal logic program P yield exactly the same inference operators if the sublanguage $\mathcal{A} = \mathrm{Hb}(P)$ of $\mathcal{L}_{\mathrm{BD}}$ is considered. This restriction is indeed essential. To see this, consider $P = \{\mathrm{a} \leftarrow \mathrm{b}\}$. Now $\mathrm{b} \rightarrow \mathrm{a}$ is $\mathbf{B}$-provable from the translation $\mathrm{Tr}_{\mathrm{GL}}(P) = \{\mathrm{b} \rightarrow \mathrm{a}\}$, but not from the translation $\mathrm{Tr}_{\mathrm{BFK}}(P) = \{\mathbf{B}\mathrm{b} \rightarrow \mathrm{a}\}$. On the other hand, the atom a is not $\mathbf{B}$-provable from either translation. We note also that this result can be directly established for the non-monotonic systems $\mathfrak{H}_P$ and $\mathfrak{I}_P$ introduced in Definition 10.4, but restriction to the atoms of $\mathrm{Hb}(P)$ is still necessary.

Proposition 10.5 *If P is a normal logic program, $\mathcal{A} = \mathrm{Hb}(P)$, $N \subseteq \mathcal{L}_{\mathrm{BD}}$ and $\phi \in \mathcal{A}$, then $\mathrm{Tr}_{\mathrm{GL}}(P) \cup \mathbf{D}N \vdash_{\mathbf{B}} \phi$ if and only if $\mathrm{Tr}_{\mathrm{BFK}}(P) \cup \mathbf{D}N \vdash_{\mathbf{B}} \phi$.*

Proof. It follows by Lemma 3.16 and the Definition of $\mathbf{B}$-provability that $\mathrm{Tr}_{\mathrm{GL}}(P) \cup \mathbf{D}N \models \phi$ if and only if $\mathrm{Tr}_{\mathrm{GL}}(P) \cup \mathbf{D}N \vdash_{\mathbf{B}} \phi$, since $\mathrm{Tr}_{\mathrm{GL}}(P) \cup \mathbf{D}N$

is free of **B**-atoms. It is then shown for $\phi \in \mathcal{A}$ that $\mathrm{Tr}_{\mathrm{GL}}(P) \cup \mathbf{D}N \models \phi$ if and only if $\mathrm{Tr}_{\mathrm{BFK}}(P) \cup \mathbf{D}N \vdash_{\mathbf{B}} \phi$. ($\Leftarrow$) We use induction on the length of the **B**-proof of ϕ. For the base case, assume that ϕ is **B**-provable from $\mathrm{Tr}_{\mathrm{BFK}}(P) \cup \mathbf{D}N$ in 0 steps. This implies that $\mathrm{Tr}_{\mathrm{BFK}}(P) \cup \mathbf{D}N \models \phi$ and $\mathrm{Hds}(\mathrm{Tr}_{\mathrm{BFK}}(P), \mathbf{D}N) \models \phi$ by Lemma 3.30. Note that $\mathrm{Hds}(\mathrm{Tr}_{\mathrm{BFK}}(P), \mathbf{D}N)$ is a set of atoms to which ϕ belongs necessarily. Then there is a rule $\phi \leftarrow {\sim}\psi_1, ..., {\sim}\psi_m$ of P without positive subgoals such that $\{\mathbf{D}\psi_1, ..., \mathbf{D}\psi_m\} \subseteq \mathbf{D}N$. Then $\mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \rightarrow \phi$ is also included in the translation $\mathrm{Tr}_{\mathrm{GL}}(P)$. It follows that $\mathrm{Tr}_{\mathrm{GL}}(P) \cup \mathbf{D}N \models \phi$. Induction step follows. Now ϕ has a **B**-proof $\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n$ from $\mathrm{Tr}_{\mathrm{BFK}}(P) \cup \mathbf{D}N$ so that $\mathrm{Tr}_{\mathrm{BFK}}(P) \cup \mathbf{D}N \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\} \models \phi$ holds. Moreover, we may assume that each ϕ_i is a member of $\mathcal{A} = \mathrm{Hb}(P)$ by Lemma 3.16 and the definition of $\mathrm{Tr}_{\mathrm{BFK}}(P)$. It follows by Lemma 3.30 that $\mathrm{Hds}(\mathrm{Tr}_{\mathrm{BFK}}(P), \mathbf{D}N \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\}) \models \phi$. Again, the atom ϕ belongs necessarily to the set of atoms $\mathrm{Hds}(\mathrm{Tr}_{\mathrm{BFK}}(P), \mathbf{D}N \cup \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\})$. Then there is a rule $\phi \leftarrow \chi_1, ..., \chi_l, {\sim}\psi_1, ..., {\sim}\psi_m$ in P such that $\{\mathbf{D}\psi_1, ..., \mathbf{D}\psi_m\} \subseteq \mathbf{D}N$ and $\{\mathbf{B}\chi_1, ..., \mathbf{B}\chi_l\} \subseteq \{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\}$. Now each ϕ_i and thus also χ_i is an atom **B**-provable from $\mathrm{Tr}_{\mathrm{BFK}}(P) \cup \mathbf{D}N$ in less than n steps. Therefore $\mathrm{Tr}_{\mathrm{GL}}(P) \cup \mathbf{D}N \models \chi_1 \wedge ... \wedge \chi_l$ holds by the inductive hypothesis. Since $\mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \chi_1 \wedge ... \wedge \chi_l \rightarrow \phi$ is included in $\mathrm{Tr}_{\mathrm{GL}}(P)$ and $\{\mathbf{D}\psi_1, ..., \mathbf{D}\psi_m\} \subseteq \mathbf{D}N$, it follows that also $\mathrm{Tr}_{\mathrm{GL}}(P) \cup \mathbf{D}N \models \phi$ holds.

($\Rightarrow$) Let us then assume that $\phi \notin \mathrm{Cn}_{\mathbf{B}}(\mathrm{Tr}_{\mathrm{BFK}}(P) \cup \mathbf{D}N)$ holds and define $T = \mathrm{Cn}_{\mathbf{B}}(\mathrm{Tr}_{\mathrm{BFK}}(P) \cup \mathbf{D}N) \cap \mathcal{A}$ and the model $\mathcal{M} = T \cup \mathbf{D}N$. Thus $\phi \notin T$ and $\mathcal{M} \not\models \phi$. Then assume that $\mathcal{M}$ is not a model of $\mathrm{Tr}_{\mathrm{GL}}(P)$. So there is an implication $\mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \phi_1 \wedge ... \wedge \phi_n \rightarrow \phi'$ in $\mathrm{Tr}_{\mathrm{GL}}(P)$ which is false in $\mathcal{M}$, i.e. $\mathcal{M} \models \mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \phi_1 \wedge ... \wedge \phi_n$ and $\mathcal{M} \not\models \phi'$. It follows that $\{\mathbf{D}\psi_1, ..., \mathbf{D}\psi_m\} \subseteq \mathbf{D}N$ and $\{\phi_1, ..., \phi_n\} \subseteq T$ and $\phi' \notin T$ by the definition of $\mathcal{M}$. On the other hand, the sentence $\mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \mathbf{B}\phi_1 \wedge ... \wedge \mathbf{B}\phi_n \rightarrow \phi'$ belongs to $\mathrm{Tr}_{\mathrm{BFK}}(P)$. But then $\phi' \in \mathrm{Cn}_{\mathbf{B}}(\mathrm{Tr}_{\mathrm{BFK}}(P) \cup \mathbf{D}N)$, because $\{\mathbf{D}\psi_1, ..., \mathbf{D}\psi_m\} \subseteq \mathbf{D}N$ and $\{\mathbf{B}\phi_1, ..., \mathbf{B}\phi_n\} \subseteq \mathbf{B}T$ are also contained in this closure. But this implies $\phi' \in T$ by the definition of T, a contradiction. Hence $\mathcal{M}$ is a model of $\mathrm{Tr}_{\mathrm{GL}}(P)$ and $\mathrm{Tr}_{\mathrm{GL}}(P) \cup \mathbf{D}N$. Then $\mathcal{M} \not\models \phi$ implies $\mathrm{Tr}_{\mathrm{GL}}(P) \cup \mathbf{D}N \not\models \phi$. $\square$

Let us now discuss what kinds of semantics are obtained for a normal logic program P in terms of the translations $\mathrm{Tr}_{\mathrm{GL}}(P)$ and $\mathrm{Tr}_{\mathrm{BFK}}(P)$ using autoepistemic systems. First of all, the stable semantics of the system $\boldsymbol{\mathcal{A}}_{\mathrm{Tr}_{\mathrm{GL}}(P)}$ (restricted to the set of atoms $\mathrm{Hb}(P)$) coincides with the stable semantics of a normal logic program P proposed by Gelfond and Lifschitz [32]. It can be shown using Proposition 10.5 that the systems $\boldsymbol{\mathcal{A}}_{\mathrm{Tr}_{\mathrm{GL}}(P)}$ and $\boldsymbol{\mathcal{A}}_{\mathrm{Tr}_{\mathrm{BFK}}(P)}$ yield the same semantics. Quite similarly, the stationary semantics of $\boldsymbol{\mathcal{A}}_{\mathrm{Tr}_{\mathrm{GL}}(P)}$ as well as $\boldsymbol{\mathcal{A}}_{\mathrm{Tr}_{\mathrm{BFK}}(P)}$ coincide with the well-founded semantics of P proposed by Van Gelder et al. [115]. As discussed already in Section 4.3 this is a particular novelty of the bimodal autoepistemic logic proposed in this work. Moreover, the semantics considered in Section 8.2 and 8.4 give rise to a variety of semantics for normal logic programs including regular semantics [118], stable class semantics [3], codiagonal semantics [110] and cautious semantics proposed in this work. Again, it does not matter which one of the translations $\mathrm{Tr}_{\mathrm{GL}}(P)$ and $\mathrm{Tr}_{\mathrm{BFK}}(P)$ is used: the semantics of P remains unaffected.

Particularly, cautious semantics is a semantics of normal logic programs that (i) refines the well-founded semantics, but is sound with respect to stable semantics and (ii) is cumulatively monotonic. Since the translations $\mathrm{Tr}_{\mathrm{GL}}(P)$ and $\mathrm{Tr}_{\mathrm{BFK}}(P)$ contain sentences that are effectively Horn-clauses, the decision problem corresponding to the cautious semantics of normal logic programs is $\mathbf{\Pi}^{\mathrm{p}}_{1}$-complete (i.e. **co-NP**-complete). This is because the satisfiability of a set of Horn-clauses can be tested in linear time using the Dowling-Gallier procedure [21]. Thus Proposition 9.3 gives the membership in $\mathbf{\Pi}^{\mathrm{p}}_{1}$, since the problem $\mathbf{TH}_{\mathrm{L}}$ belongs to $\mathbf{P}$ in this particular case. The $\mathbf{\Pi}^{\mathrm{p}}_{1}$-hardness is obtained by representing the satisfiability problem [14] as a normal logic program. The author [44] presents an encoding of the satisfiability problem to prove a predecessor of cautious semantics to be $\mathbf{\Pi}^{\mathrm{p}}_{1}$-complete for normal logic programs.

10.1.3 Program Transformations

Dix [18, 20] presents a variety of *transformations* for logic programs. The idea behind his transformation approach is to establish invariance results for various semantics of logic programs. More precisely, a logic program P and the transformed program P' should have the same semantics. Dix considers invariance under particular transformations such as *elimination of tautologies*, *elimination of non-minimal rules* and *partial evaluation* as a merit for a semantics. Dix has used his framework for two main purposes. Firstly, he has *classified* various semantics of logic programs [20] by identifying under which transformations a particular semantics is invariant. Moreover, Brass and Dix [9] have taken a step further: they *define* a semantics for logic programs as the weakest semantics that is invariant under a set of transformations that they consider to be appropriate. To keep things simple, we present the above mentioned three transformations formulated for a *normal logic program P* in Definitions 10.6-10.8.

Definition 10.6 (Elimination of Tautologies) *Remove a rule*

$$\phi \leftarrow \phi_1, ..., \phi_n, {\sim}\psi_1, ..., {\sim}\psi_m$$

from P if the head $\phi \in \{\phi_1, ..., \phi_n\}$.

Definition 10.7 (Elimination of Non-Minimal Rules) *Remove a rule*

$$\phi \leftarrow \phi_1, ..., \phi_n, {\sim}\psi_1, ..., {\sim}\psi_m$$

from P if there is another rule $\phi \leftarrow \phi'_1, ..., \phi'_{n'}, \sim\psi'_1, ..., \sim\psi'_{m'}$ in P such that the subgoals $\{\phi'_1, ..., \phi'_{n'}\} \subseteq \{\phi_1, ..., \phi_n\}$ and $\{\psi'_1, ..., \psi'_{m'}\} \subseteq \{\psi_1, ..., \psi_m\}$.

Definition 10.8 (Partial Evaluation) *For a rule*

$$\phi \leftarrow \phi_1, ..., \phi_n, \sim\psi_1, ..., \sim\psi_m$$

and a subgoal ϕ_i and for every rule $\phi_i \leftarrow \phi'_1, ..., \phi'_{n'}, \sim\psi'_1, ..., \sim\psi'_{m'}$ in P add the rule $\phi \leftarrow \phi_1, ..., \phi_{i-1}, \phi_{i+1}, ..., \phi_n, \phi'_1, ..., \phi'_{n'}, \sim\psi_1, ..., \sim\psi_m, \sim\psi'_1, ..., \sim\psi'_{m'}$.

The partial evaluation principle of Definition 10.8 is slightly different from that proposed by Dix [20]: it only adds rules and the rule being partially evaluated is not removed. This modification is necessary to guarantee that the propositional models of the program as well as the autoepistemic interpretation $\mathrm{Tr}_{\mathrm{GL}}(P)$ are preserved. This aspect of partial evaluation is demonstrated further in Example 10.9.

Example 10.9 Consider a normal logic program $P = \{\mathrm{a} \leftarrow \mathrm{b}\ ;\ \mathrm{b} \leftarrow \mathrm{c}\}$[42]. A Dix-style [20] partial evaluation of the first rule with respect to the subgoal b yields the program $P_1 = \{\mathrm{a} \leftarrow \mathrm{c}\ ;\ \mathrm{b} \leftarrow \mathrm{c}\}$ whereas its partial evaluation corresponding to Definition 10.8 gives $P_2 = \{\mathrm{a} \leftarrow \mathrm{b}\ ;\ \mathrm{a} \leftarrow \mathrm{c}\ ;\ \mathrm{b} \leftarrow \mathrm{c}\}$. Now $\mathrm{Tr}_{\mathrm{GL}}(P)$ and $\mathrm{Tr}_{\mathrm{GL}}(P_2)$ are propositionally equivalent (have the same propositional models) whereas $\mathrm{Tr}_{\mathrm{GL}}(P)$ and $\mathrm{Tr}_{\mathrm{GL}}(P_1)$ are not. To see this, note that $\mathrm{Tr}_{\mathrm{GL}}(P_2)$ does not imply $\mathrm{b} \rightarrow \mathrm{a}$. However, all of translations $\mathrm{Tr}_{\mathrm{GL}}(P)$, $\mathrm{Tr}_{\mathrm{GL}}(P_1)$ and $\mathrm{Tr}_{\mathrm{GL}}(P_2)$ have a unique minimal model $\mathcal{M} = \emptyset$. □

As suggested by this example, our goal is to show that the transformations introduced in the preceding definitions do not affect the propositional consequences of the translations. This property is shown for each of the transformations in Lemmas 10.10, 10.11 and 10.12, respectively. Note that the programs under consideration are normal logic programs, i.e. only propositional atoms may appear in the rules, and the Gelfond-Lifschitz-style translation of the program rules is considered (see Definition 10.2).

Lemma 10.10 *Assume that P' is a transformation of P obtained by the elimination of a tautology. Then* $\mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P)) = \mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P'))$.

Proof. Assume that the rule $\phi \leftarrow \phi_1, ..., \phi_n, \sim\psi_1, ..., \sim\psi_m$ was deleted when P' was derived from P. The difference between $\mathrm{Tr}_{\mathrm{GL}}(P)$ and $\mathrm{Tr}_{\mathrm{GL}}(P')$ is the implication $\mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \phi_1 \wedge ... \wedge \phi_n \rightarrow \phi$ which is a propositional tautology, since ϕ is one of the subgoals $\phi_1, ..., \phi_n$. Therefore, the removal of the implication does not affect the propositional models of $\mathrm{Tr}_{\mathrm{GL}}(P)$. Hence $\mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P)) = \mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P'))$. □

[42]Since subgoals are separated by commas, we use semicolons to separate program rules.

Lemma 10.11 *Assume that P' is a transformation of P obtained by the elimination of a non-minimal rule. Then* $\mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P)) = \mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P'))$.

Proof. Assume that the rule $\phi \leftarrow \phi_1, ..., \phi_n, {\sim}\psi_1, ..., {\sim}\psi_m$ was deleted when P' was derived from P. Thus the only difference between $\mathrm{Tr}_{\mathrm{GL}}(P)$ and $\mathrm{Tr}_{\mathrm{GL}}(P')$ is the implication (i) $\mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \phi_1 \wedge ... \wedge \phi_n \rightarrow \phi$ which is missing from $\mathrm{Tr}_{\mathrm{GL}}(P')$. Thus $\mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P')) \subseteq \mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P))$ directly by the monotonicity of the operator Cn. To show the converse inclusion, recall that by Definition 10.7 there is another rule $\phi \leftarrow \phi'_1, ..., \phi'_{n'}, {\sim}\psi'_1, ..., {\sim}\psi'_{m'}$ such that $\{\phi'_1, ..., \phi'_{n'}\} \subseteq \{\phi_1, ..., \phi_n\}$ and $\{\psi'_1, ..., \psi'_{m'}\} \subseteq \{\psi_1, ..., \psi_m\}$. Let (ii) $\mathbf{D}\psi'_1 \wedge ... \wedge \mathbf{D}\psi'_{m'} \wedge \phi'_1 \wedge ... \wedge \phi'_{n'} \rightarrow \phi'$ be the corresponding autoepistemic translation which appears in $\mathrm{Tr}_{\mathrm{GL}}(P)$ as well as $\mathrm{Tr}_{\mathrm{GL}}(P')$ by definition. Let us then show that the implication (i) is a logical consequence of the implication (ii). So assume this is not the case. Then there is a model $\mathcal{M}$ of (ii) in which (i) is false, i.e. $\mathcal{M} \models \mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \phi_1 \wedge ... \wedge \phi_n$ and $\mathcal{M} \not\models \phi$. Since $\{\psi'_1, ..., \psi'_{m'}\} \subseteq \{\psi_1, ..., \psi_m\}$ and $\{\phi'_1, ..., \phi'_{n'}\} \subseteq \{\phi_1, ..., \phi_n\}$, it follows that $\mathcal{M} \models \mathbf{D}\psi'_1 \wedge ... \wedge \mathbf{D}\psi'_{m'} \wedge \phi'_1 \wedge ... \wedge \phi'_{n'}$ and $\mathcal{M} \not\models \phi$. Thus (ii) is false in $\mathcal{M}$, a contradiction. Hence the implication (i) is a logical consequence of (ii) as well as $\mathrm{Tr}_{\mathrm{GL}}(P')$. It follows that $\mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P)) \subseteq \mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P'))$. □

Lemma 10.12 *Assume P' is a transformation of P obtained by a partial evaluation of a rule $\phi \leftarrow \phi_1, ..., \phi_n, {\sim}\psi_1, ..., {\sim}\psi_m$ with respect to a positive subgoal ϕ_i. Then* $\mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P)) = \mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P'))$.

Proof. ($\subseteq$) This holds by the monotonicity of Cn, since $\mathrm{Tr}_{\mathrm{GL}}(P) \subseteq \mathrm{Tr}_{\mathrm{GL}}(P')$. ($\supseteq$) Let $\phi \leftarrow \phi_1, ..., \phi_{i-1}, \phi_{i+1}, ..., \phi_n, \phi'_1, ..., \phi'_{n'}, {\sim}\psi_1, ..., {\sim}\psi_m, {\sim}\psi'_1, ..., {\sim}\psi'_{m'}$ be a rule obtained from the rule $\phi \leftarrow \phi_1, ..., \phi_n, {\sim}\psi_1, ..., {\sim}\psi_m$ by partial evaluation using the rule $\phi_i \leftarrow \phi'_1, ..., \phi'_{n'}, {\sim}\psi'_1, ..., {\sim}\psi'_{m'}$. We refer by (i), (ii) and (iii) to the autoepistemic translations of these sentences, respectively. Let us show that (i) is a logical consequence of (ii) and (iii). For the sake of contradiction, we assume that there is a model $\mathcal{M}$ of (ii) and (iii) such that (i) is false in $\mathcal{M}$. It follows that the conjunction $\mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \mathbf{D}\psi'_1 \wedge ... \wedge \mathbf{D}\psi'_{m'} \wedge \phi_1 \wedge ... \wedge \phi_{i-1} \wedge \phi_{i+1} \wedge ... \wedge \phi_n \wedge \phi'_1 \wedge ... \wedge \phi'_{n'}$ is true in $\mathcal{M}$, but $\mathcal{M} \not\models \phi$. The former implies that $\mathcal{M} \models \mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \phi_1 \wedge ... \wedge \phi_{i-1} \wedge \phi_{i+1} \wedge ... \wedge \phi_n$ and $\mathcal{M} \models \mathbf{D}\psi'_1 \wedge ... \wedge \mathbf{D}\psi'_{m'} \wedge \phi'_1 \wedge ... \wedge \phi'_{n'}$. There are two cases to consider. (A) If $\mathcal{M} \models \phi_i$, we obtain $\mathcal{M} \models \mathbf{D}\psi_1 \wedge ... \wedge \mathbf{D}\psi_m \wedge \phi_1 \wedge ... \wedge \phi_n$. Since also $\mathcal{M} \not\models \phi$ holds, it follows that (ii) is false in $\mathcal{M}$, a contradiction. (B) If $\mathcal{M} \not\models \phi_i$, then (iii) is false in $\mathcal{M}$, a contradiction. By (A) and (B), we have a contradiction. Hence (i) is a logical consequence of (ii) and (iii) for all rules obtained by partial evaluation. Thus $\mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P')) \subseteq \mathrm{Cn}(\mathrm{Tr}_{\mathrm{GL}}(P))$ holds. □

Lemma 10.13 *If* $\mathrm{Cn}(\Sigma) = \mathrm{Cn}(\Sigma')$ *for* $\Sigma \subseteq \mathcal{L}_{\mathrm{BD}}$ *and* $\Sigma' \subseteq \mathcal{L}_{\mathrm{BD}}$, *then*

$$\mathrm{Cn}_{\mathbf{B}}(\Sigma) = \mathrm{Cn}_{\mathbf{B}}(\Sigma')$$

Proof sketch. Let $\mathrm{Cn}(\Sigma) = \mathrm{Cn}(\Sigma')$. ($\subseteq$) This follows easily by induction on the lengths of $\mathbf{B}$-proofs. ($\supseteq$) This holds by symmetry. $\square$

Theorem 10.14 *Let P a normal logic program and P' its transformation obtained by one of the transformations given in Definitions 10.6-10.8. Let $T = \mathrm{Tr}_{\mathrm{GL}}(P)$ and $T' = \mathrm{Tr}_{\mathrm{GL}}(P')$ and $\mathcal{S}$ an autoepistemic system $\mathcal{A}_\emptyset$ such that T and T' are subsets of $\mathcal{L}^+(\mathcal{S}) = \mathcal{L}_{\mathrm{BD}}$. Then T and T' are $\mathcal{S}$-equivalent.*

Proof. Let N be any subset of $\mathcal{L}^-(\mathcal{S})$. We know by Lemmas 10.10-10.12 that $\mathrm{Cn}(T) = \mathrm{Cn}(T')$. It follows that $\mathrm{Cn}(T \cup \mathbf{D}N) = \mathrm{Cn}(T' \cup \mathbf{D}N)$ by the properties of Cn. Therefore, the sets of theorems $\mathcal{S}(T, N) = \mathrm{Cn}_{\mathbf{B}}(T \cup \mathbf{D}N)$ and $\mathcal{S}(T', N) = \mathrm{Cn}_{\mathbf{B}}(T' \cup \mathbf{D}N)$ coincide by Lemma 10.13. Consequently, the sets of sentences T and T' are $\mathcal{S}$-equivalent by Proposition 5.7. $\square$

Corollary 10.15 *For T, T' and $\mathcal{S}$ given in Theorem 10.14, it holds that*

$$\begin{array}{rl} \textit{(i)} & \mathrm{STAT}_{\mathcal{S}}(T) = \mathrm{STAT}_{\mathcal{S}}(T'), \\ \textit{(ii)} & \mathrm{STAB}_{\mathcal{S}}(T) = \mathrm{STAB}_{\mathcal{S}}(T') \textit{ and} \\ \textit{(iii)} & \mathrm{CAUT}_{\mathcal{S}}(T) = \mathrm{CAUT}_{\mathcal{S}}(T'). \end{array}$$

Proof sketch. This is a corollary of Theorem 10.14 and Theorem 5.8 if stationary semantics and stable semantics are considered. For cautious semantics, it can be shown that B is an $\mathcal{S}[T]$-bias if and only if B is an $\mathcal{S}[T']$-bias and that the fixed points $N = \mathrm{lfp}(\overline{\mathcal{S}[T; B]}^2, \emptyset)$ and $N' = \mathrm{lfp}(\overline{\mathcal{S}[T'; B]}^2, \emptyset)$ coincide. Then it follows by Proposition 5.7 that a sentence $\phi \in \mathcal{L}^+(\mathcal{S})$ belongs to $\phi \in \mathcal{S}(T, N)$ if and only if $\phi \in \mathcal{S}(T', N')$. $\square$

From the point of view of normal logic programs, Corollary 10.15 means that the stable semantics [32], the well-founded semantics [115] and the cautious semantics of normal logic programs are immune to the three transformations under consideration. This result was first established by Dix for stable semantics and well-founded semantics up to the slight difference in the notions of partial evaluation. Our contribution is establishing such an invariance result for the cautious semantics under the three transformations of normal logic programs introduced in Definitions 10.6, 10.7 and 10.8.

10.1.4 Other Characterizations of Well-Founded Semantics

Since our autoepistemic interpretation captures the well-founded semantics of normal logic programs, we provide a brief comparison with similar approaches. Przymusinski [92] has studied a variant of autoepistemic logic which is based on three-valued propositional logic. He introduces a notion of a three-valued autoepistemic expansion which captures the well-founded model of a normal logic program [115] under a suitable translation. Przymusinski's characterization resorts to a three-valued implication which is closer to two-valued implication than the three-valued implication based on Kleene's strong tables [51]. In contrast, our interpretation is based on standard two-valued propositional logic where implication has a unique, well-understood interpretation. The three-valued character of well-founded models is captured in terms of $\boldsymbol{\mathcal{A}}_{\mathrm{Tr}_{\mathrm{GL}}(P)}$-stationary sets of non-theorems $N \subseteq \mathcal{L}_{\mathrm{BD}}$ each of which divides the language $\mathcal{L}_{\mathrm{BD}}$ in three parts: non-theorems included in N, theorems included in $\boldsymbol{\mathcal{A}}_{\mathrm{Tr}_{\mathrm{GL}}(P)}(N)$ and potential non-theorems included in $\overline{\boldsymbol{\mathcal{A}}_{\mathrm{Tr}_{\mathrm{GL}}(P)}(N)} - N$. Thus our autoepistemic interpretation is three-valued at the level of beliefs while the underlying monotonic logic (i.e. propositional logic) is two-valued.

Przymusinski has proposed also further derivatives of autoepistemic logic [90, 91] by replacing negative introspection by a new primitive corresponding to Lifschitz's parallel circumscription [59]. This leads to the notion of *static semantics* of disjunctive logic programs. For normal logic programs, this semantics coincides with well-founded semantics. In this way, Przymusinski's approach provides a further autoepistemic characterization of the well-founded model though parallel circumscription is used for negative introspection. By our results, it is not necessary to resort to minimal models in negative introspection to capture well-founded models. We obtain the same effect by introducing the modality **D** to represent the results of negative introspection and by using an autoepistemic counterpart of stationary extensions [89].

Bonatti [6] has studied the semantics of logic programs in terms of a three-valued autoepistemic logic. His system captures various semantics of logic programs including stable models [32], well-founded models [115] and *supported* models [1]. To achieve this, Bonatti employs *different* translations of the program rules. Marek and Truszczyński [67] have shown that Reiter's extensions and weak extensions capture the stable and supported models of a normal logic programs, respectively. The same effect is achieved with stable/iterative expansions introduced in Definition 2.31. Thus our system captures various models of normal logic programs (including stable, well-founded and supported) using the *same translation function* but different kinds of autoepistemic expansions (iterative, stationary [47] and stable). This speaks for a uniform view that $\mathrm{Tr}_{\mathrm{GL}}$ *preserves* these semantics of normal logic programs.

10.1.5 Classical Negation

There are also other classes of logic programs that seem interesting from knowledge representation perspective. One of the first ideas to extend normal logic programs was to introduce classical negation in the program rules, i.e. replace bare atoms by literals. The resulting class of logic programs is known as *extended logic programs* [33] as proposed by Gelfond and Lifschitz. They generalize the notion of stable semantics of normal logic program by introducing *answer sets* for extended normal logic programs. A further result is that an extended logic program P can be rewritten as normal logic programs P', by replacing negative literals $\neg\mathrm{a}$ with new atoms a'. Then consistent answer sets of P are obtained as stable models of P' in which a and a' are not simultaneously true for any atom $\mathrm{a} \in \mathrm{Hb}(P)$. It is clear that the notion of hyper logic programs covers also extended logic programs. However, a Gelfond-Lifschitz-style translation of program rules is not sound with respect to answer set semantics, since the contrapositives of implications in the translation are activated. This phenomenon is demonstrated in Example 10.16.

Example 10.16 Consider an extended logic program $P = \{\neg\mathrm{a}\ ;\ \mathrm{a} \leftarrow \mathrm{b}\}$ and its autoepistemic translations $\mathrm{Tr}_{\mathrm{GL}}(P) = \{\neg\mathrm{a}, \mathrm{b} \rightarrow \mathrm{a}\}$ and $\mathrm{Tr}_{\mathrm{BFK}}(P) = \{\neg\mathrm{a}, \mathbf{B}\mathrm{b} \rightarrow \mathrm{a}\}$. There is only one answer set $\{\neg\mathrm{a}\}$, but the translation $\mathrm{Tr}_{\mathrm{GL}}(P)$ has only one iterative expansion which contains both $\neg\mathrm{a}$ and $\neg\mathrm{b}$. However, the only iterative expansion of the translation $\mathrm{Tr}_{\mathrm{BFK}}(P)$ contains just $\neg\mathrm{a}$. □

There is also an alternative solution to this problem. It is possible to use the rewriting technique proposed by Gelfond and Lifschitz [33] and then apply the translation $\mathrm{Tr}_{\mathrm{GL}}$. The consistency of answer sets can be enforced under this translation by adding for each $\mathrm{a} \in \mathrm{Hb}(P)$ the sentence $\mathrm{a} \wedge \mathrm{a}' \rightarrow \bot$. It is also possible to address *paraconsistent reasoning*[43] within this approach by dropping out these constraints and using stable semantics as it is. Note that adding the sentences $\mathrm{a} \wedge \mathrm{a}' \rightarrow \bot$ and $\mathrm{b} \wedge \mathrm{b}' \rightarrow \bot$ to the resulting translation $\{\mathrm{a}', \mathrm{b} \rightarrow \mathrm{a}\}$ of P above does not make b' a logical consequence of the translation. Finally, we note that stationary semantics is probably very weak if classical negation is allowed in the program rules, e.g., under the translation $\mathrm{Tr}_{\mathrm{BFK}}(P)$. This is because the set of potential theorems becomes easily propositionally inconsistent. One possibility to go round this problem is to use the paraconsistent approach described above. Nevertheless, our results imply that cautious semantics provides a semantics for extended logic programs that is stronger than stationary semantics, but sound with respect to stable semantics independently of our attitude to paraconsistency.

10.1.6 Disjunctions

Another generalization of normal logic programs is obtained if disjunctions of propositional atoms are allowed as heads and subgoals in the rules. This gives rise to the class of *disjunctive logic programs*. Our definition is somewhat looser

[43] This means reasoning in presence of inconsistency, i.e. when both a and $\neg\mathrm{a}$ are derivable for some atom a.

than the conventional one (see e.g. [9]) which allows disjunctions to appear in the heads of rules only. However, we feel that this restriction is unnecessary, and therefore allow disjunctions to appear as subgoals of the rules. The benefit of this is additional succinctness in knowledge representation: similar rules can be combined into one. For instance, two rules $\mathrm{a} \leftarrow \mathrm{b}$ and $\mathrm{a} \leftarrow \mathrm{c}$ can be merged into $\mathrm{a} \leftarrow \mathrm{b} \vee \mathrm{c}$. On the other hand, it is possible to remove disjunctions from positive subgoals if a Gelfond-Lifschitz-style interpretation of the program rules is considered. Thus only disjunctions in the heads as well as negative subgoals of rules make a real difference with respect to normal logic programs.

By our generalized definition, we take the following view in disjunctive logic programming: disjunctions of propositional atoms take the same role as propositional atoms have in normal logic programs. In this way, disjunctions of atoms $\mathrm{a}_1 \vee ... \vee \mathrm{a}_n$ – representing a choice between atoms $\mathrm{a}_1, ..., \mathrm{a}_n$ – become basic elements of disjunctive logic programs instead of single atoms. The same principle is applied to queries: we would like know whether any given disjunction of atoms can be derived by the program rules or not. For a disjunctive logic program P, we let $\mathrm{Db}(P)$ denote the *disjunction base* of P, i.e the set of all disjunctions of propositional atoms that are constructible from the atoms in $\mathrm{Hb}(P)$. In particular, we distinguish the *empty disjunction* $\bot$[44] which represents a logical contradiction as one would expect. Example 10.17 illustrates in the monotonic case how we would like to enrich our inferences in the presence of disjunctions.

Example 10.17 Consider a disjunctive logic program

$$P = \{\mathrm{a} \vee \mathrm{b}\ ;\ \mathrm{c} \leftarrow \mathrm{a}\ ;\ \mathrm{d} \leftarrow \mathrm{b}\ ;\ \mathrm{e} \leftarrow \mathrm{e}\}.$$

Given these rules, we would like to infer $\mathrm{c} \vee \mathrm{d}$ as well as $\mathrm{c} \vee \mathrm{d} \vee \mathrm{e}$. The former disjunction of these is obtained as a result of *case-analysis* while the latter disjunction can be considered as a *weakening* of the former. These conclusions are obtained as direct propositional consequences from a Gelfond-Lifschitz-style translation $\mathrm{Tr}_{\mathrm{GL}}(P) = \{\mathrm{a} \vee \mathrm{b}, \mathrm{a} \rightarrow \mathrm{c}, \mathrm{b} \rightarrow \mathrm{d}, \mathrm{e} \rightarrow \mathrm{e}\}$ of the above program. However, this is not the case for the alternative translation $\mathrm{Tr}_{\mathrm{BFK}}(P) = \{\mathrm{a} \vee \mathrm{b}, \mathbf{B}\mathrm{a} \rightarrow \mathrm{c}, \mathbf{B}\mathrm{b} \rightarrow \mathrm{d}, \mathbf{B}\mathrm{e} \rightarrow \mathrm{e}\}$ of the program even if **B**-provability is considered instead of propositional consequence. □

Our example shows that disjunctive programs get a different interpretation under the translation functions $\mathrm{Tr}_{\mathrm{GL}}$ and $\mathrm{Tr}_{\mathrm{BFK}}$. It seems possible to eliminate this difference by introducing suitable axiom schemas for the modality **B**. To keep things simple, we resort to a Gelfond-Lifschitz-style translation of the program rules in the disjunctive case and take the full advantage of the

[44]Recall that $\bot$ is the identity element for $\vee$, i.e. the sentences $\bot \vee \phi$ and $\phi \vee \bot$ and ϕ are propositionally equivalent for all sentences $\phi \in \mathcal{L}$.

machinery of propositional logic. Let us then look at what kind of stationary semantics is obtained for disjunctive logic programs under this interpretation. Example 10.18 demonstrates how disjunctive inferences are naturally incorporated to the stationary construction.

Example 10.18 Consider a disjunctive logic program

$$P = \{\mathrm{b} \vee \mathrm{c} \leftarrow {\sim}\mathrm{a}\ ;\ \mathrm{p} \leftarrow \mathrm{b}\ ;\ \mathrm{p} \leftarrow \mathrm{c}\ ;\ \mathrm{q} \leftarrow {\sim}(\mathrm{p} \vee \mathrm{q})\ ;\ \mathrm{s} \leftarrow {\sim}\mathrm{b}, {\sim}\mathrm{c}\ ;\ \mathrm{r} \leftarrow {\sim}\mathrm{q}\}.$$

This is interpreted as a set of autoepistemic sentences

$$\begin{aligned}\mathrm{Tr_{GL}}(P) \ = \{ \ & \mathbf{D}\mathrm{a} \rightarrow \mathrm{b} \vee \mathrm{c}, \\ & \mathrm{b} \rightarrow \mathrm{p},\ \mathrm{c} \rightarrow \mathrm{p}, \\ & \mathbf{D}(\mathrm{p} \vee \mathrm{q}) \rightarrow \mathrm{q}, \\ & \mathbf{D}\mathrm{b} \wedge \mathbf{D}\mathrm{c} \rightarrow \mathrm{s}, \\ & \mathbf{D}\mathrm{q} \rightarrow \mathrm{r} \qquad\qquad \}.\end{aligned}$$

Let Σ denote this translation. We use the subsystem $\mathcal{S}$ of $\boldsymbol{\mathcal{A}}_\Sigma$ with $\mathcal{L}^+(\mathcal{S}) = \mathrm{Db}(P)$ and $\mathcal{L}^-(\mathcal{S}) = \mathrm{SS}^*_{\mathbf{D}}(\Sigma) = \{\mathrm{a}, \mathrm{p}\vee\mathrm{q}, \mathrm{b}, \mathrm{c}, \mathrm{q}\}$ to illustrate how the stationary semantics of the program is determined.[45] Note that we need not **B**-provability in this case, since the translation is free of **B**-atoms and therefore it suffices to consider only propositional consequence. We begin with the set of non-theorems $N_0 = \emptyset$. None of sentences $\mathrm{a}, \mathrm{p}\vee\mathrm{q}, \mathrm{b}, \mathrm{c}$ and q follows from $\Sigma \cup \mathbf{D}N_0$ so that the set of potential non-theorems N_0' is $\{\mathrm{a}, \mathrm{p}\vee\mathrm{q}, \mathrm{b}, \mathrm{c}, \mathrm{q}\}$. Then $\mathrm{b}\vee\mathrm{c}, \mathrm{p}, \mathrm{p}\vee\mathrm{q}$ and q follow from $\Sigma \cup \mathbf{D}N_0'$. Thus we obtain the set of non-theorems $N_1 = \{\mathrm{a}, \mathrm{b}, \mathrm{c}\}$. Now p and $\mathrm{p} \vee \mathrm{q}$ are still consequences of $\Sigma \cup \mathbf{D}N_1$ so that $N_1' = \{\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{q}\}$ is the set of potential non-theorems. By the same arguments, we obtain the set of non-theorems $N_2 = \{\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{q}\}$. This is the least fixed point $\mathrm{lfp}(\overline{\mathcal{S}}^2, \emptyset)$ which is also the unique $\mathcal{S}$-stable set of non-theorems. Therefore, the sentences $\mathrm{b} \vee \mathrm{c}$, p, $\mathrm{p} \vee \mathrm{q}$, s and r belong to $\mathrm{STAT}_{\mathcal{S}}(\emptyset) = \mathrm{STAB}_{\mathcal{S}}(\emptyset)$ while a, b, c and q do not. □

A particular novelty of our approach is that the computational complexity of stationary semantics becomes strictly lower than that of stable semantics as implied by Theorems 7.18 and 7.20 on the complexity of the variant of autoepistemic logic proposed in this work. The problem whether a given disjunction $\mathrm{a}_1 \vee ... \vee \mathrm{a}_n$ is a conclusion under stationary semantics is $\mathbf{\Delta}^{\mathrm{p}}_2$-complete while the problem whether a given disjunction $\mathrm{a}_1 \vee ... \vee \mathrm{a}_n$ is a conclusion under stable semantics is $\mathbf{\Pi}^{\mathrm{p}}_2$-complete. This differs from earlier approaches (e.g. [9, 10]) where the decision problem of stationary semantics

[45] A subsystem $\mathcal{S}$ with a finite language $\mathcal{L}(\mathcal{S})$ is used, because then the sets of non-theorems under consideration are finite and thus easily representable. However, this restriction causes no loss of generality by Theorem 6.18: the same conclusions are drawn if the unrestricted system $\boldsymbol{\mathcal{A}}_\Sigma$ is considered.

is at least $\mathbf{\Pi}_2^{\mathrm{p}}$-hard. This is because these variants of stationary semantics insist on the minimal models of the program clauses already when there are no negative subgoals present, i.e. when *positive* disjunctive logic programs are considered. The results of Gottlob and Eiter [25, 24] imply that in the presence of minimization the complexity of reasoning with disjunctive program clauses becomes $\mathbf{\Pi}_2^{\mathrm{p}}$-hard. Since minimization is not *enforced* in our approach, the notions of stable semantics and stationary semantics differ from standard approaches such as one by Gelfond and Lifschitz [34].

However, minimization is also possible within our approach, if it is appropriate for knowledge representation purposes. This is illustrated by the following.

Example 10.19 Consider a logic program $P = \{\mathrm{a} \vee \mathrm{b}\ ;\ \mathrm{c} \leftarrow {\sim}\mathrm{a}, {\sim}\mathrm{b}\}$. A typical approach is to check minimal models of $\mathrm{a} \vee \mathrm{b}$, i.e. $\mathcal{M}_1 = \{\mathrm{a}\}$ and $\mathcal{M}_2 = \{\mathrm{b}\}$, and evaluate the body ${\sim}\mathrm{a}, {\sim}\mathrm{b}$ of the latter rule with respect to these. Then the body is not satisfied and c is not a conclusion under stable semantics. Our interpretation of the program above is $\Sigma = \mathrm{Tr}_{\mathrm{GL}}(P) = \{\mathrm{a} \vee \mathrm{b}, \mathbf{D}\mathrm{a} \wedge \mathbf{D}\mathrm{b} \to \mathrm{c}\}$. Since a and b are not propositional consequences of Σ, the set of non-theorems $N = \{\mathrm{a}, \mathrm{b}\}$ is obtained as the least $\mathcal{S}$-stationary set of non-theorems (for a subsystem $\mathcal{S}$ of $\boldsymbol{\mathcal{A}}_{\Sigma}$ such that $\mathcal{L}^{-}(\mathcal{S}) = \{\mathrm{a}, \mathrm{b}\}$). This is also the unique $\mathcal{S}$-stable set of non-theorems. Therefore, the atom c can be concluded under both the stationary and the stable semantics of Σ, in contrast to minimization-based approaches (such as [34]) to disjunctive logic programming where $\neg\mathrm{c}$ is concluded.

However, minimization can be expressed within our approach. What is most pleasant, we achieve this effect by just introducing additional program rules $\perp \leftarrow \mathrm{a}, {\sim}\mathrm{a}$, $\perp \leftarrow \mathrm{b}, {\sim}\mathrm{b}$ and $\perp \leftarrow \mathrm{c}, {\sim}\mathrm{c}$ that enforce minimization of a, b and c as follows. Let P' be the enlarged program. Note that these new rules are interpreted as an autoepistemic sentences $\mathbf{D}\mathrm{a} \wedge \mathrm{a} \to \perp$, $\mathbf{D}\mathrm{b} \wedge \mathrm{b} \to \perp$ and $\mathbf{D}\mathrm{c} \wedge \mathrm{c} \to \perp$ which are propositionally equivalent to $\mathbf{D}\mathrm{a} \to \neg\mathrm{a}$, $\mathbf{D}\mathrm{b} \to \neg\mathrm{b}$ and $\mathbf{D}\mathrm{c} \to \neg\mathrm{c}$, respectively. These implications express effectively that the atoms a, b and c are false *by default*. The effect of these sentences is that the stationary semantics of the extended translation Σ' becomes weaker: the least $\mathcal{S}'$-stationary set of non-theorems is $N_0 = \emptyset$ (for a subsystem $\mathcal{S}'$ of $\boldsymbol{\mathcal{A}}_{\Sigma'}$ with $\mathcal{L}^{-}(\mathcal{S}') = \{\mathrm{a}, \mathrm{b}, \mathrm{c}\}$), since the set of potential non-theorems $N_0' = \{\mathrm{a}, \mathrm{b}, \mathrm{c}\}$ gives rise to a propositionally inconsistent theory $\Sigma \cup \mathbf{D}N_0'$ (setting both a and b false in the presence of $\mathrm{a} \vee \mathrm{b}$ causes a contradiction). Hence, only $\mathrm{a} \vee \mathrm{b}$ can be concluded under stationary semantics. On the other hand, stable semantics is successful: there are two $\mathcal{S}'$-stable sets of non-theorems $N = \{\mathrm{a}, \mathrm{c}\}$ and $N' = \{\mathrm{b}, \mathrm{c}\}$. But now, there is no way of concluding c, because the sets of sentences $\Sigma' \cup \mathbf{D}N$ and $\Sigma' \cup \mathbf{D}N'$ imply $\neg\mathrm{c}$ rather than c. In this way, the status of c has changed to agree with the minimization-based approaches.

Since the stationary semantics and the stable semantics of P' differ under our autoepistemic interpretation, this gives us an opportunity to apply cautious semantics. Since a and b cannot be concluded under stationary semantics, we are entitled to fix either one as a non-theorem. Both of these turn out to be successful. The non-monotonic systems $\mathcal{S}'[\{\mathrm{a}\}]$ and and $\mathcal{S}'[\{\mathrm{b}\}]$ obtained in this way give rise to the least stationary sets of non-theorems $N = \{\mathrm{a}, \mathrm{c}\}$ and $N' = \{\mathrm{b}, \mathrm{c}\}$, respectively. This is how cautious semantics coincides with stable semantics for this particular program. This is quite understandable, since the equality of cautious semantics and stable semantics was established for minimal model systems in Theorem 8.23. However, this is not the case in general if disjunctive programs are considered. □

Few remarks on this lengthy example are necessary. Firstly, it shows that the cautious semantics of the translation $\mathrm{Tr}_{\mathrm{GL}}(P)$ leads to a further semantics for disjunctive logic programs. By our results presented in Section 8, this is a semantics between stationary semantics and stable semantics that refines stationary semantics in many cases. It is also cumulatively monotonic and of the same computational complexity as stable semantics. Secondly, it is possible to minimize atoms if necessary by introducing new rules that produce the sentences of the form $\mathbf{D}\mathrm{a} \rightarrow \neg\mathrm{a}$ in the translation. Note that such implications are effectively obtained through the translation functions Tr_{E} and Tr_{K} proposed by Etherington [26] and Konolige [52] when $\neg\mathbf{B}$ is replaced with $\mathbf{D}$. Practically the same translation is obtained as a composition of the translation by Sakama and Inoue [98] and Tr_{K}. They translate rules $\mathrm{a}_1 \vee \ldots \vee \mathrm{a}_l \leftarrow \mathrm{b}_1, \ldots, \mathrm{b}_m, {\sim}\mathrm{c}_1, \ldots, {\sim}\mathrm{c}_n$ of a disjunctive logic program P as defaults $\frac{:\neg\mathrm{c}_1,\ldots,\neg\mathrm{c}_n}{\mathrm{b}_1\wedge\ldots\wedge\mathrm{b}_m\rightarrow\mathrm{a}_1\vee\ldots\vee\mathrm{a}_l}$ and introduce defaults $\frac{:\neg\mathrm{a}}{\neg\mathrm{a}}$ to minimize each $\mathrm{a} \in \mathrm{Hb}(P)$. Under Tr_{K} this leads to a translation equivalent to $\mathrm{Tr}_{\mathrm{GL}}(P)$ given that minimization defaults are ignored. By the results of Sakama and Inoue as well as Corollary 6.23, our notion of stable semantics coincides with that of Gelfond and Lifschitz [34] if we set all atoms subject to minimization.

In our opinion, a blind minimization of all propositional atoms may become a source of complications that restricts the applicability of disjunctive logic programs for knowledge representation. To get an idea of the possible difficulties, we consider a diagnosis example Example 10.20. The rules of the program P are derived from the natural representations of digital components to be introduced in Section 10.2 in further detail. The rules describe the normal behaviour of a digital circuit consisting of two buffers with a joint input d connected to the inputs a and b of an or-gate with an output c. The circuit is depicted in Figure 18. Moreover, the output of the or-gate has been measured to be 1, and therefore the fact $\mathrm{c} \leftarrow$ been introduced in the program.

Example 10.20 Consider a logic program $P = \{\mathrm{c}\vee\mathrm{ab}_1 \leftarrow \mathrm{a}\vee\mathrm{b}\ ;\ \mathrm{a}\vee\mathrm{b}\vee\mathrm{ab}_1 \leftarrow \mathrm{c}\ ;\ \mathrm{c} \leftarrow\ ;\ \mathrm{a} \vee \mathrm{ab}_2 \leftarrow \mathrm{d}\ ;\ \mathrm{d} \vee \mathrm{ab}_2 \leftarrow \mathrm{a}\ ;\ \mathrm{b} \vee \mathrm{ab}_3 \leftarrow \mathrm{d}\ ;\ \mathrm{d} \vee \mathrm{ab}_3 \leftarrow \mathrm{b}\}$.

If we set all propositional atoms subject to minimization (in accordance with Gelfond and Lifschitz's approach [34]), we obtain six minimal models for P, namely $\mathcal{M}_1 = \{\mathrm{c}, \mathrm{ab}_1\}$, $\mathcal{M}_2 = \{\mathrm{c}, \mathrm{b}, \mathrm{ab}_3\}$, $\mathcal{M}_3 = \{\mathrm{c}, \mathrm{b}, \mathrm{d}, \mathrm{ab}_2\}$, $\mathcal{M}_4 = \{\mathrm{c}, \mathrm{a}, \mathrm{ab}_2\}$, $\mathcal{M}_5 = \{\mathrm{c}, \mathrm{a}, \mathrm{d}, \mathrm{ab}_2\}$ and $\mathcal{M}_6 = \{\mathrm{c}, \mathrm{a}, \mathrm{d}, \mathrm{b}\}$. The first five minimal models describe situations where one of the components is broken (i.e. abnormal) while $\mathcal{M}_6$ gives the intended situation where none of the components is broken, as suggested by our observations on the circuit so far (only the output has been measured while the inputs are still unknown).

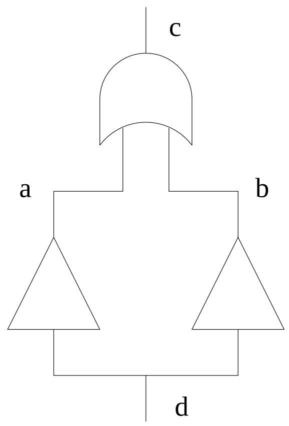


Figure 18: A simple digital circuit

The above anomaly is avoided if we *restrict* minimization to the abnormality atoms ab_1, ab_2 and ab_3 solely. In our approach, we need just to introduce the rules $\perp \leftarrow \mathrm{ab}_i, {\sim}\mathrm{ab}_i$ for $i \in \{1, 2, 3\}$. Consequently, we obtain a unique stable set of non-theorems containing ab_1, ab_2 and ab_3. This describes a situation where all the three components are functioning correctly and all inputs have the value 1. This matches perfectly with our intuitions (and $\mathcal{M}_6$ above). $\square$

On the basis of this example, we conclude that our approach brings additional flexibility into disjunctive logic programming, as it allows us to specify which atoms are to be minimized. Recall also Example 10.18 where minimal models do not have any role, but still reasonable disjunctive inferences are made in conjunction with the stationary construction. This kind of flexibility differentiates our approach from typical approaches to disjunctive logic programming. To end our account on disjunctive logic programming, we provide a brief comparison with two recent approaches, namely *disjunctive well-founded semantics* proposed by Dix and Brass and *static semantics* proposed by Przymusinski et al. [10]. The former semantics is obtained in terms of a set of program transformations. The latter semantics is not restricted to disjunctive logic programs, it applies also to the class of super logic programs. Przymusinski uses a modal non-monotonic logic with a modal operator "not" and effectively considers minimal models to define the semantics of this operator (this logic was originally proposed as an autoepistemic logic of closed beliefs [91]). Static semantics involves also modal schemas for Przymusinski's not-operator.

Brass and Dix [9] argue that the two semantics coincide if disjunctive logic programs are considered and only purely disjunctive queries are taken into account. Moreover, a normal logic program P is assigned to its standard well-founded semantics [115] under these semantics. As argued earlier, the same result is deducible for our approach, i.e. the stationary semantics of the translation $\mathrm{Tr}_{\mathrm{GL}}(P)$ yields also the well-founded semantics of P. Differences arise when disjunctive programs are considered and the arguments presented in Example 10.19 apply. According to disjunctive well-founded semantics and static semantics c is not a conclusion of the program P analyzed in the example. Brass and Dix use a program transformation named as *negative reduction* to remove the rule concerning c while Przymusinski et al. obtain the same effect from the minimal models of $\mathrm{a} \vee \mathrm{b}$. Finally, we present the main example by Przymusinski et al. [10] proposed in conjunction with static semantics.

Example 10.21 Consider a super/hyper logic program

$$P = \{\mathrm{e} \vee \mathrm{a}\ ;\ \mathrm{h} \leftarrow \mathrm{e} \vee \mathrm{a}\ ;\ \mathrm{b} \leftarrow \mathrm{e} \wedge \mathrm{a}\ ;\ \mathrm{p} \leftarrow {\sim}(\mathrm{e} \wedge \mathrm{a})\ ;\ \mathrm{d} \leftarrow {\sim}(\mathrm{e} \vee \mathrm{a})\}.$$

The meanings of the atoms e, a, h, b, p and d are "one visits Europe", "one visits Australia", "happy", "bankrupt", "prudent" and "disappointed", respectively. Przymusinski et al. argue that h and p are plausible conclusions while b and d are not. The translation $\mathrm{Tr}_{\mathrm{GL}}(P)$ is $\Sigma = \{\mathrm{e} \vee \mathrm{a}, \mathrm{e} \vee \mathrm{a} \rightarrow \mathrm{h}, \mathrm{e} \wedge \mathrm{a} \rightarrow \mathrm{b}, \mathbf{D}(\mathrm{e} \wedge \mathrm{a}) \rightarrow \mathrm{p}, \mathbf{D}(\mathrm{e} \vee \mathrm{a}) \rightarrow \mathrm{d}\}$. For a subsystem $\mathcal{S}$ of $\mathcal{A}_{\Sigma}$ with $\mathcal{L}^{-}(\mathcal{S}) = \{\mathrm{e} \wedge \mathrm{a}, \mathrm{e} \vee \mathrm{a}\}$, the least $\mathcal{S}$-stationary set of non-theorems is $N = \{\mathrm{e} \wedge \mathrm{a}\}$. It is easy to see that h and p are $\mathbf{B}$-provable from $\Sigma \cup \mathbf{D}N$ while b and d are not. □

The program P above is not a disjunctive program, but still very interesting to compare our approach with that of Przymusinski's. It shows that our autoepistemic interpretation of the program rules yields exactly the conclusions wanted by Przymusinski and al. already if the stationary semantics of $\mathrm{Tr}_{\mathrm{GL}}(P)$ is considered. This indicates that our approach based on bimodal autoepistemic logic yields reasonable conclusions, even though it is based on much simpler constructs than Przymusinski's approach.

10.2 Consistency-Based Diagnosis

The second application of non-monotonic reasoning to be considered is the *diagnosis* problem of technical devices. Basically, a technical device can be understood to consist of a set of interacting components. The diagnosis problem is to identify which components of the device are broken (if any) given some *observations* on the device state/behaviour. Generally speaking, there are two leading approaches to this problem: *abductive* [54] and *consistency-based* [94]. The abductive approach tries to *explain* the observed misbehaviour by assuming that some components are faulty. This kind of reasoning requires inherently information on how the components of the device behave when they malfunction. In contrast to this, only the correct behaviour of the components is modeled in the other approach. Every component is assumed either to operate according to its specification or to be faulty. Then any discrepancy between the observations and the predictions given by the model is taken as misbehaviour. In other words, if our observations are not *consistent* with predictions obtained from the model by the assumption that all components are operating normally, then some components have to be faulty. In the following, we resort to a consistency-based definition of a diagnosis as proposed by Reiter [94]. We assume that a *system description* S , which describes the correct behaviour of the components, has been given as a set of propositional sentences. For each component c in the given set of components C, a special atom ab_c is used to denote that the component c is *abnormal* (i.e. broken). Observations on the device behaviour are also represented as a set of propositional sentences O. A *candidate* $D \subseteq C$ is a set of components (assumed to be abnormal simultaneously) that is sufficient to remove the contradiction between the observations and the model. A *diagnosis* D is a minimal set of components with this property. This definition stems from an assumption that typically only few components of a device are malfunctioning.

Definition 10.22 *Let C be a set of components, S a system description and O a set of observations. A set $D \subseteq C$ is a candidate if and only if $S \cup O \cup \{\neg\mathrm{ab}_c \mid c \in C - D\} \cup \{\mathrm{ab}_c \mid c \in D\}$ is consistent. A candidate $D \subseteq C$ is a diagnosis if there is no other candidate D' such that $D' \subset D$.*

We rely on the close relationship of diagnoses and minimal models:

Proposition 10.23 *Let C be a set of components, S a system description and O a set of observations. Define the set $P = \{\mathrm{ab}_c \mid c \in C\}$ of atoms to be minimized. Then the P-extensions of $S \cup O$ determine the diagnoses of the system S given observations O.*

Proof sketch. By Reiter's results [94] it holds for a diagnosis D and for all components $c \in D$ that $S \cup O \cup \{\neg\mathrm{ab}_c \mid c \in C - D\} \models \mathrm{ab}_c$. □

In the sequel, we will consider the diagnosis problem in the context of digital circuits consisting of various types of components such as *and-gates*, *or-gates*, *inverters* and *buffers*. In Table 5, we have modeled these components in terms of propositional sentences. Note that a wire state 0 and 1 is represented by a single atom in the descriptions. This is an obvious improvement compared to the representation used in Examples 2.35-2.37 where two atoms were used. A similar approach is adopted by Baumgartner and al. [4] recently. The descriptions of the components concern their normal behaviour and they have been written such that they can be seen as Gelfond-Lifschitz-style translations of rules of a disjunctive logic program.

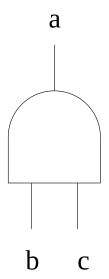
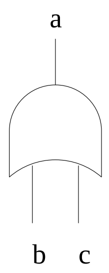
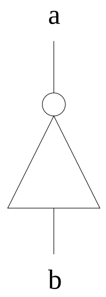
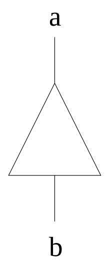
$\mathrm{b} \wedge \mathrm{c} \rightarrow \mathrm{a} \vee \mathrm{ab}$ $\mathrm{a} \rightarrow \mathrm{b} \vee \mathrm{ab}$ $\mathrm{a} \rightarrow \mathrm{c} \vee \mathrm{ab}$	$\mathrm{b} \vee \mathrm{c} \rightarrow \mathrm{a} \vee \mathrm{ab}$ $\mathrm{a} \rightarrow \mathrm{b} \vee \mathrm{c} \vee \mathrm{ab}$	$\mathrm{a} \vee \mathrm{b} \vee \mathrm{ab}$ $\mathrm{a} \wedge \mathrm{b} \rightarrow \mathrm{ab}$	$\mathrm{a} \rightarrow \mathrm{b} \vee \mathrm{ab}$ $\mathrm{b} \rightarrow \mathrm{a} \vee \mathrm{ab}$
			
an and-gate	an or-gate	an inverter	a buffer

Table 5: Descriptions of digital components

In addition to the component descriptions, we have to represent our intention to minimize abnormality. This is expressed as a sentence $\mathbf{D}\mathrm{ab}_c \wedge \mathrm{ab}_c \rightarrow \bot$ for each component c (or $\bot \leftarrow \mathrm{ab}_c, \sim\!\mathrm{ab}_c$ in terms of disjunctive logic programs). Thus every component c is assumed to behave normally by default. In particular, we emphasize that the atoms representing wire states are not minimized and the complications of unrestricted minimization are avoided. It was illustrated in Example 10.20 how *counter-intuitive diagnoses are obtained if the atoms representing wire states are also minimized.* The rules in this example are compatible with our representation of the gates in Table 5.

Our system description is a set of autoepistemic sentences, but we argue by Proposition 10.23 and Corollaries 6.22 and 6.23 that our representation of the problem captures correctly the minimization insisted by Definition 10.22. By the following example, we demonstrate how disjunctive inferences are needed to solve the diagnosis problem properly.

Example 10.24 Consider a digital circuit with two or-gates (numbered 1 and 4), two inverters (numbered 2 and 3) and two buffers (numbered 5 and 6) depicted in Figure 19. If we apply the representation given in Table 5, we obtain a logical description of the whole circuit given in Table 6. Let Σ denote this set of sentences. Suppose that we have also measured the outputs a = 1 and g = 1 and the status of the input d is not known. This information can be encoded by adding the atomic propositions a and g in our description.

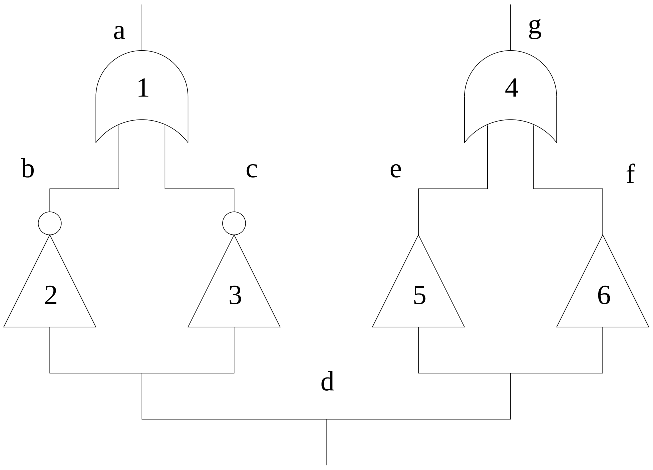


Figure 19: A more complex digital circuit

$\mathrm{b} \vee \mathrm{c} \rightarrow \mathrm{a} \vee \mathrm{ab}_1$	$\mathbf{D}\mathrm{ab}_1 \wedge \mathrm{ab}_1 \rightarrow \bot$	$\mathrm{e} \vee \mathrm{f} \rightarrow \mathrm{g} \vee \mathrm{ab}_4$
$\mathrm{a} \rightarrow \mathrm{b} \vee \mathrm{c} \vee \mathrm{ab}_1$	$\mathbf{D}\mathrm{ab}_4 \wedge \mathrm{ab}_4 \rightarrow \bot$	$\mathrm{g} \rightarrow \mathrm{e} \vee \mathrm{f} \vee \mathrm{ab}_4$
$\mathrm{b} \vee \mathrm{d} \vee \mathrm{ab}_2$	$\mathbf{D}\mathrm{ab}_2 \wedge \mathrm{ab}_2 \rightarrow \bot$	$\mathrm{e} \rightarrow \mathrm{d} \vee \mathrm{ab}_5$
$\mathrm{d} \wedge \mathrm{b} \rightarrow \mathrm{ab}_2$	$\mathbf{D}\mathrm{ab}_5 \wedge \mathrm{ab}_5 \rightarrow \bot$	$\mathrm{d} \rightarrow \mathrm{e} \vee \mathrm{ab}_5$
$\mathrm{c} \vee \mathrm{d} \vee \mathrm{ab}_3$	$\mathbf{D}\mathrm{ab}_3 \wedge \mathrm{ab}_3 \rightarrow \bot$	$\mathrm{f} \rightarrow \mathrm{d} \vee \mathrm{ab}_6$
$\mathrm{d} \wedge \mathrm{c} \rightarrow \mathrm{ab}_3$	$\mathbf{D}\mathrm{ab}_6 \wedge \mathrm{ab}_6 \rightarrow \bot$	$\mathrm{d} \rightarrow \mathrm{f} \vee \mathrm{ab}_6$

Table 6: Description of a digital circuit

Let us then consider what kind of inferences are performed in this example if the stationary semantics of the autoepistemic representation is adopted. First of all, none of the atoms ab_i is a logical consequence of $\Sigma' = \Sigma \cup \{\mathrm{a}, \mathrm{g}\}$. Therefore a Reiter-style closed world assumption [93] "all components are functioning normally" is introduced by stationary semantics: the set of potential non-theorems $N = \{\mathrm{ab}_1, ..., \mathrm{ab}_6\}$ is considered. Then atoms $\neg\mathrm{ab}_1, ..., \neg\mathrm{ab}_6$ become propositional consequences of $\Sigma' \cup \mathbf{D}N$. Using the sentences in Σ', we can infer (i) $\mathrm{b} \vee \mathrm{c}$ and (ii) $\neg\mathrm{d}$ (a disjunctive inference) from $\Sigma' \cup \mathbf{D}N$. For similar reasons, the sentences $\mathrm{e} \vee \mathrm{f}$ and d are propositional consequences of $\Sigma' \cup \mathbf{D}N$ which is therefore inconsistent. Thus $N = \emptyset$ is the least stationary set of non-theorems. This indicates that the assumption that all components are functioning correctly is wrong, i.e. at least one component is broken.

If we use cautious semantics, the computations described above for stationary semantics are performed repeatedly. If the propositional atoms $\mathrm{ab}_1, ..., \mathrm{ab}_6$ are considered in this order and the algorithm of function $\mathsf{FindBias}_X$ is used, a bias $B = \{\mathrm{ab}_1, ..., \mathrm{ab}_5\}$ is gradually constructed. The bias B gives rise to the least stationary set of non-theorems $N = B$. Note that ab_6 is a logical consequence of $\Sigma' \cup \mathbf{D}N$ and $\Sigma' \cup \mathbf{D}N$ remains propositionally consistent. This indicates that N determines one diagnosis. The buffer 6 is broken: the output $\mathrm{f} = 1$ even though the input $\mathrm{d} = 0$, since $\neg\mathrm{d}$ and f follow from $\Sigma' \cup \mathbf{D}N$. □

Some remarks on the example and the role of cautious semantics follows. (i) We note that other diagnoses are obtained by changing order in which sentences are used for biasing. For instance, we could change the relative order of ab_5 and ab_6 and use the atoms in the order $\mathrm{ab}_1, \ldots, \mathrm{ab}_4, \mathrm{ab}_6, \mathrm{ab}_5$ for biasing. This order gives rise to the least stationary set of non-theorems $N' = \{\mathrm{ab}_1, ..., \mathrm{ab}_4, \mathrm{ab}_6\}$. The theory $\Sigma' \cup \mathbf{D}N'$ determines a diagnosis by which the buffer 5 is faulty. This indicates that the order in which sentences are used for biasing induces priorities on diagnoses. The components that end of this ordering are most probably faulty, as suggested by the two diagnoses computed above. (ii) Secondly, cautious semantics and stable semantics coincide for our representation of the diagnosis problem. This follows effectively from Corollaries 6.22 and 6.23 which imply that our autoepistemic representation captures minimal models as well as Theorem 8.23 where we showed that the two semantics coincide for minimal model systems. However, we feel that one advantage of cautious semantics is that it gives a refined structure for the diagnosis problem while stable semantics gives just a fixed point condition for diagnoses. (iii) Note also that diagnostic reasoning is cumulative under our representation, since the two semantics coincide and we know by Theorems 5.13 and 8.27 that stable semantics is cumulatively closed and cautious semantics is cumulatively monotonic. This implies that if we can deduce that a component c is necessarily faulty under these semantics (i.e. it is faulty in all diagnoses), then adding ab_c to the system description will not change the diagnoses. (iv) Finally, the algorithm encoded in function $\mathsf{FindBias}_X$ improves the algorithm proposed by Friedrich, Gottlob and Nejdl [30] for computing the *first diagnosis* (any of the diagnoses). Their algorithm is based on a similar idea in the sense that components of the device are marked to be normal step by step (this amounts to fixing ab_c as a non-theorem in our approach). A difference is that the algorithm of the function $\mathsf{FindBias}_X$ uses stationary semantics as an approximation repeatedly: (i) it is tested whether the remaining *undetermined* components (i.e. that have not yet been assumed to normal nor known to be faulty) can be assumed to be normal in one step. If this is the case, we have obtained a diagnosis. Otherwise, (ii) our algorithm picks one of the undetermined components c and assumes the component c to be normal (by fixing ab_c as non-theorem) and returns to step (i).

11 Conclusions

A framework of non-monotonic systems is proposed in this work to provide a basis for studying non-monotonic logics, their semantics and abstract properties. The inference operators associated with leading formalizations of non-monotonic reasoning – circumscription, default logic and autoepistemic logic – are shown to fit our framework. This indicates a deep similarity in these logics despite they are based on relatively different primitives: minimal models, default rules and belief operators, respectively. Moreover, it is demonstrated that various semantics proposed for non-monotonic logics can be lifted to the level of non-monotonic systems. More importantly, many interesting properties of these semantics – such as cumulativity and computational complexity – are analyzed by starting directly from the specification of non-monotonic systems and independently of particular non-monotonic logic. This provides new insight to how far certain semantical ideas can be applied and where are the limits of their knowledge representation capability. For instance, we formulate conditions on which stationary semantics collapses.

In particular, we want to emphasize the role of the bimodal autoepistemic logic proposed in this work. The following advantages over unimodal autoepistemic logic are established. (i) A reasonable notion of stationary semantics is obtained for autoepistemic logic. This is by no means straightforward for conventional autoepistemic logic and its predecessors as illustrated in this work. (ii) Very intuitive translations functions are presented between default theories and autoepistemic theories. This involves solving a fundamental problem related with Konolige's translation from default theories into autoepistemic theories as well as inventing a completely new translation technique for the converse direction. An interesting result is that the presented translations are modular and they preserve the semantics considered in the work. We take this as a strong indication that default logic and (bimodal) autoepistemic logic are of equal expressive power under the semantics considered.

Also new ideas concerning the semantics of non-monotonic logics are brought to the scene: cautious semantics is proposed as a systematic refinement of stationary semantics. It is also established that cautious semantics is sound with respect to stable semantics. In fact, cautious semantics is a relatively strong semantics, because these semantics are of the same computational complexity. An interesting difference is also established: stable semantics is cumulatively closed while cautious semantics is cumulatively monotonic. It remains as an open problem whether cautious semantics can be still refined such that it becomes also cumulatively closed. Nevertheless, we feel that the biasing principle behind cautious semantics forms a promising step towards cumulative forms of non-monotonic inference.

Two applications domains of non-monotonic reasoning, namely logic programming and consistency-based diagnosis are considered. The relationship to various semantics of logic programs is studied via an interpretation of logic programs as theories in the bimodal autoepistemic logic proposed in this work. Basically, three classes of logic programs are considered: normal logic programs and the extensions obtained by allowing classical negation and disjunction in the program rules. To conclude, cautious semantics leads to a new semantics in all of these three classes. For normal logic programs, the notions of stable and stationary semantics coincide with the standard notions. In the disjunctive case, noticeable differences are obtained even for stationary and stable semantics. In particular, the scope of minimization is identified to cause important differences. Finally, the diagnosis problem of digital circuits is formulated in terms of the bimodal autoepistemic logic. An example demonstrating the essence of disjunctions in representing this problem is given. The role of cautious semantics in this context is discussed: it turns out that cautious semantics provides means to incorporate priorities in diagnostic reasoning. Also, the possibility of using stationary semantics for approximation purposes in this context is pointed out.

11.1 Future Work

The results presented in this work suggest the following topics for further research. (i) It is to be expected that non-monotonic systems can be extracted for other formalizations of non-monotonic reasoning than circumscription, default logic and autoepistemic logic. For instance, the *priority logic* proposed by Wang et al. [116] and the *argumentation-theoretic* approach proposed by Bondarenko et al. [7] seem interesting in this respect. Once new non-monotonic systems are established, it is interesting to investigate what role the semantics considered in this work have in conjunction with these systems. For instance, we could analyze whether a non-trivial notions of stable and stationary semantics result for a given system. The latter is a natural prerequisite in order to apply cautious semantics to the system. Moreover, our framework provides a platform for generalizing a semantics proposed for a particular non-monotonic system to cover all non-monotonic systems. This is actually done in Section 8.4 where a number of semantics – proposed mainly for normal logic programs – are lifted to the level of non-monotonic systems. This kind of abstraction gives a unified basis for analysing and comparing semantics.

(ii) In Section 2.3.4, we discuss some possibilities of modifying minimal model systems, default systems and autoepistemic systems such that the axioms of non-monotonic systems still hold. The ideas presented provide a starting point to devising new variants of existing systems. These variants are called for to enhance the knowledge representation capabilities of existing systems. Also, one can expect that the understanding gained from non-monotonic systems is indeed valuable if one considers equipping a logic of interest with a non-monotonic reasoning mechanism. To give an example, a hybrid of temporal logic [36] and autoepistemic logic might be a suitable combination if one wants to formalize the beliefs of an agent that depend on time. Generally speaking, the non-monotonic logics considered in this work lack an explicit concept of time. For example, autoepistemic logic tries to capture the possible sets of beliefs of an agent at one point in time (unless we introduce time by selecting atoms of the language suitably). Nevertheless, the way that the beliefs of an agent change in time, i.e. *dynamics* of beliefs, is an issue worth attention.

(iii) Cautious semantics is an interesting semantics which is cumulatively monotonic but not cumulatively closed, as shown in Section 8.3.3. It remains open whether the biasing principle of cautious semantics can be strengthened such that the resulting semantics is also cumulatively closed. In light of our experiences, this is a non-trivial result. Finding a solution to this matter would lead us to take the algorithms and approximation techniques of Section 9 into reconsideration, since our aim is to utilize maximally the stationary construction as an approximation. In the presence of cumulativity, it is also possible to "compile" a knowledge base, i.e. add non-monotonic conclusions in the knowledge base, in order to speed up forthcoming query processing. Moreover, the weaker notions of cumulativity which are addressed in Section 5.6 seem interesting and need attention. For example, Schaub's approach [99] where inference rules are added as "lemmas" in a default theory should be kept in mind when the weaker notions of cumulativity are reconsidered.

(iv) Cautious semantics seems to provide a reasonable set of conclusions in many cases where stable semantics is inconsistent (there are no stable sets of non-theorems) and this suggests that cautious semantics could be useful in *debugging* a knowledge base, i.e. to find out reasons for which a stable state cannot be reached. A further application of cautious semantics is to use it as an *approximation* of stable semantics which is sound by Theorem 8.21. Basically, there are two strategies, depending on whether we are interested in a single sets of theorems induced by a stable set of non-theorems or the intersection of such sets of theorems, i.e. the full stable semantics. If the full stable semantics is of interest, a possibility is to employ the range of approximations introduced for cautious semantics in Section 9.3. If we want a single set of theorems, we could compute one complete bias and use the resulting set of theorems as a first approximation – to provide preliminary answers to queries. Our representation of the diagnosis problem in Section 10.2 is such that a complete bias gives already rise to a diagnosis (in the sense of Definition 10.22) which can be subsequently used as the first diagnosis. However, if additional constraints are introduced in the representation this might not be the case. In this setting, a complete bias seems to provide a notion of a *partial diagnosis* which should be studied further. A partial diagnosis can be extended to a complete diagnosis, or alternatively it becomes a diagnosis given some extra observations. Another promising application to be investigated is the *configuration* problem of technical devices [71, 117]. In this domain, we expect that a complete bias gives rise to a *preliminary configuration* of the device, which can be refined to a full configuration.

(v) The bimodal autoepistemic logic proposed in this work provides an interesting platform for addressing the semantics of negation as failure. An advantage of this approach that it is not restricted to a particular class of logic programs, i.e. covers the class of *hyper logic programs* considered in the work. In particular, the semantic basis of disjunctive logic programs is promising from the point of view of the examples presented in Section 10.1.6. So far, the approaches in disjunctive logic programming have been very diverse and we believe that the approach based on bimodal autoepistemic logic could clarify the situation and lead to a reasonable semantics for disjunctive logic programs. The properties of disjunctive logic programs as suggested by Brass and Dix [9] provide means to assess the success of our approach.

(vi) Niemelä and Simons [85] have developed an efficient implementation of the stable semantics [32] for normal logic programs. It is to be expected the techniques used in their system `smodels` lead to an efficient implementation of the algorithm FindBias for computing a single complete bias (see Figure 14 on page 158) for a normal logic program. As suggested in the end of Section 10.2, the order in which the algorithm FindBias considers sentences for biasing provides a way of prioritising among conclusions (the sentences that are most preferably non-theorems should be considered first). Given such an ordering on the atoms of a normal logic program, the algorithm FindBias finds the first bias in polynomial time with respect to the size of the program. This observation suggests that non-monotonic reasoning could be *easier* in cases where a priority ordering is available, since all orderings should be taken into account otherwise. In any case, the way in which priorities are incorporated into non-monotonic reasoning is already a research problem of its own.

A Preliminaries (Appendix)

We assume that the reader is familiar with the basic notions of set theory such as the relations $\in$ (membership) and $\subseteq$ (subset) as well as the basic operations $\cap$ (intersection), $\cup$ (union) and $-$ (difference). A detailed introduction and treatment is given by Jech [49], for instance. In the following, we briefly review the basic notions of propositional logic as well as ordinal and cardinal numbers and transfinite induction, since they are continuously needed in this work.

A.1 Propositional Logic

Propositional logic is based on *propositional language* $\mathcal{L}$ which is a set of *sentences.* A definition of sentences is given below. To enhance readability, parenthesis are typically dropped from sentences as long as they remain unambiguous by the precedence of *connectives* (the negation "$\neg$" is the strongest one, the conjunction "$\wedge$" and the disjunction "$\vee$" are weaker than "$\neg$" whereas the implication "$\rightarrow$" and the equivalence "$\leftrightarrow$" are the weakest ones). Moreover, chains of conjunctions $\phi_1 \wedge ... \wedge \phi_n$ and disjunctions $\phi_1 \vee ... \vee \phi_n$ are often written without parenthesis. The set of atoms $\mathcal{A}$ below is assumed to be at most *denumerable*, i.e. of the same cardinality as the set of natural numbers, which implies that the resulting propositional language $\mathcal{L}$ is also denumerable.

Definition A.1 *Given a set of* atomic propositions *(or simply* atoms*) $\mathcal{A}$, the propositional language $\mathcal{L}$ is the least set which is closed under the following rules: (i) if $\phi \in \mathcal{A}$, then $\phi \in \mathcal{L}$ and (ii) if $\phi \in \mathcal{L}$ and $\psi \in \mathcal{L}$, then also $(\neg\phi)$, $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$, $(\phi \leftrightarrow \psi)$ belong to $\mathcal{L}$.*

The *semantics* of propositional logic is obtained by assigning *truth values* for the sentences of a propositional language $\mathcal{L}$. A *model* $\mathcal{M}$ for a language $\mathcal{L}$ is simply a subset of $\mathcal{A}$ such that $\mathcal{M}$ contains the atoms of $\mathcal{A}$ that considered to be *true* while the atoms of $\mathcal{A} - \mathcal{M}$ are considered to be *false.* Then the truth value of any sentence $\phi \in \mathcal{L}$ with respect to $\mathcal{M}$ can be determined by the structural rules given in Definition A.2. We write $\mathcal{M} \models \phi$ ($\mathcal{M} \not\models \phi$) to denote that ϕ is *true* (*false*) given a model $\mathcal{M}$ for the language $\mathcal{L}$.

Definition A.2 *Let $\mathcal{M} \subseteq \mathcal{A}$ be a model of $\mathcal{L}$. For $\phi \in \mathcal{A}$, define $\mathcal{M} \models \phi$ iff $\phi \in \mathcal{M}$. For $\phi \in \mathcal{L}$ and $\psi \in \mathcal{L}$, define*

(i)	$\mathcal{M} \models \neg\phi$	*iff*	$\mathcal{M} \not\models \phi$,
(ii)	$\mathcal{M} \models \phi \wedge \psi$	*iff*	$\mathcal{M} \models \phi$ *and* $\mathcal{M} \models \psi$,
(iii)	$\mathcal{M} \models \phi \vee \psi$	*iff*	$\mathcal{M} \models \phi$ *or* $\mathcal{M} \models \psi$,
(iv)	$\mathcal{M} \models \phi \rightarrow \psi$	*iff*	$\mathcal{M} \not\models \phi$ *or* $\mathcal{M} \models \psi$ *and*
(iv)	$\mathcal{M} \models \phi \leftrightarrow \psi$	*iff*	$\mathcal{M} \models \phi \rightarrow \psi$ *and* $\mathcal{M} \models \psi \rightarrow \phi$.

Sometimes special atoms $\top$ and $\bot$ are used to represent truth and falsity (respectively). A possibility is to consider these as shorthands for sentences $\mathrm{p} \vee \neg\mathrm{p}$ and $\mathrm{p} \wedge \neg\mathrm{p}$ for some atom $\mathrm{p} \in \mathcal{A}$ while another is to include $\top$ and $\bot$ as special atoms in $\mathcal{L}$ (but not in $\mathcal{A}$). Given a model $\mathcal{M}$, the truth values of $\top$ and $\bot$ are then defined by $\mathcal{M} \models \top$ and $\mathcal{M} \not\models \bot$ in the latter case.

A model $\mathcal{M} \subseteq \mathcal{A}$ for $\mathcal{L}$ is a model *of* $\phi \in \mathcal{L}$ if and only if $\mathcal{M} \models \phi$. A sentence is *valid* (or equivalently, *tautology*) if and only if every model for $\mathcal{L}$ is a model of ϕ. Similarly, a model $\mathcal{M} \subseteq \mathcal{A}$ for $\mathcal{L}$ is a model *of* $T \subseteq \mathcal{L}$ (denoted by $\mathcal{M} \models T$) if and only if $\mathcal{M} \models \phi$ for all $\phi \in T$. We let $\mathrm{True}(\mathcal{M})$ denote the set $\{\phi \in \mathcal{L} \,|\, \mathcal{M} \models \phi\}$, i.e. the set of sentences that are true in a model $\mathcal{M}$. Thus the set of valid sentences coincides with the intersection of $\mathrm{True}(\mathcal{M})$ for all models $\mathcal{M}$ for $\mathcal{L}$. A set of sentences $T \subseteq \mathcal{L}$ is *satisfiable* (or equivalently, propositionally *consistent*) if it has at least one model. Otherwise, the set T is *unsatisfiable* (or equivalently, propositionally *inconsistent*).

A sentence ϕ is a *logical consequence* of $T \subseteq \mathcal{L}$ (denoted by $T \models \phi$) if and only if $\mathcal{M} \models \phi$ holds for all models $\mathcal{M}$ of T. We let $\mathrm{Cn}(T)$ denote the set of logical consequences $\{\phi \in \mathcal{L} \,|\, T \models \phi\}$. Then it is straightforward to establish that $\mathrm{Cn}(T) = \bigcap\{\mathrm{True}(\mathcal{M}) \,|\, \mathcal{M} \models T\}$ so that $\mathrm{Cn}(\emptyset)$ is the set of all tautologies. A set of sentences $T \subseteq \mathcal{L}$ is *closed* under propositional consequence if and only if $\mathrm{Cn}(T) \subseteq T$. If $T \subseteq \mathcal{L}$ is inconsistent, then $\mathrm{Cn}(T) = \mathcal{L}$. Finally, let us address one important property of propositional logic, namely *compactness*. Nerode and Shore [74] apply König's lemma[46] to show that an infinite set of propositional sentences $T \subseteq \mathcal{L}$ is satisfiable if and only if every finite subset T' of T is satisfiable. This implies the result of Proposition A.3, since $T \models \phi$ holds if and only if $T \cup \{\neg\phi\}$ is unsatisfiable.

Proposition A.3 *The set of logical consequences* $\mathrm{Cn}(T)$ *coincides with the union of the sets of logical consequences* $\mathrm{Cn}(T')$ *for all finite subsets* T' *of* T.

A.2 Ordinal Numbers

To introduce ordinal numbers, we follow the presentation by Jech [49] to the extent necessary for this work. Firstly, we need some notions on *orderings* of sets. A *binary relation* $<$ over a set S is a *partial ordering* of S if and only if (i) $e \not< e$ for all elements $e \in S$ and (ii) $e_1 < e_2$ and $e_2 < e_3$ imply $e_1 < e_3$ for all elements e_1, e_2 and e_3 of S. If in addition (iii) $e_1 < e_2$ or $e_1 = e_2$ or $e_2 < e_1$ for all elements $e_1$and e_2 of S, then $<$ is a *linear ordering* of S. If a set S is partially ordered by $<$ and $T \subseteq S$, then an element e of T is the least element of T if and only if $e \in T$ and $e \leq e'$ for all elements $e' \in T$. Here $e \leq e'$ denotes that $e = e'$ or $e < e'$ (sometimes the relation $\leq$ is also called

[46] If a finitely branching tree is infinite, it has an infinite path.

partial ordering and $<$ is called *strict* ordering). An element $e \in S$ is the least element of S if and only if $e \leq e'$ holds for all $e' \in S$. A linear ordering $<$ of a set S is a *well-ordering* if and only if every nonempty subset X of S has the least element. Thus any well-ordered set S has necessarily a least element.

A set S is *transitive* if every element of S is a subset of S. A set is an *ordinal number* (or *ordinal* for short) if it is transitive and well-ordered by $\in$. The class of ordinals is linearly ordered as follows: $\alpha < \beta$ if and only if $\alpha \in \beta$. Note that $\emptyset$ (also denoted by 0) is trivially an ordinal. Moreover, if α and β are ordinals, then either $\alpha \subseteq \beta$ or $\beta \subseteq \alpha$. The *successor* of α, denoted by $\alpha + 1$, is defined as $\alpha \cup \{\alpha\}$. If $\alpha = \beta + 1$, then alpha is a *successor ordinal.* If α is not a successor ordinal, then α is a *limit ordinal.* Thus $0 = \emptyset$ is also a limit ordinal. *Natural numbers* can be defined as finite ordinals, i.e. $0 = \emptyset$, $1 = 0+1 = \{\emptyset\}$, $2 = 1+1 = \{\emptyset, \{\emptyset\}\}$ and so on. The first infinite limit ordinal is ω, i.e. the set of all natural numbers. The sum $\alpha+\beta$ of two ordinals α and β can be understood as a concatenation of the orders corresponding to α and β. Therefore the ordinal sums $2 + \omega = \omega$ and $\omega + 2$ differ. An important feature of ordinals is that every well-ordered set S is *isomorphic* to a unique ordinal number α, i.e. there is a one-to-one and order-preserving mapping between S and α. Thus ordinals can be seen as canonical well-orderings, or *order-types.*

A.3 Transfinite Induction

Let P be a property defined for ordinals α. We write $P(\alpha)$ to denote that an ordinal α has the property P. The principle of *transfinite induction* is as follows. Assume that it has been established that (i) $P(0)$ holds, (ii) if $P(\alpha)$ holds, then $P(\alpha + 1)$ holds and (ii) if α is an infinite limit ordinal and $P(\beta)$ holds for all $\beta < \alpha$, then $P(\alpha)$ holds. Then it can be concluded that $P(\alpha)$ holds for all ordinals α. In fact, any definition for ordinals has to be written in analogy to the three steps above in order to cover all ordinals α.

A.4 Cardinal Numbers

Two sets S and T have the same *cardinality*, denoted by $|S| = |T|$, if and only if there exists a one-to-one mapping between S and T. The measures of cardinality are called *cardinal numbers* (or *cardinals* for short). An ordinal α is a cardinal number if and only if $|\alpha| \neq |\beta|$ for all $\beta < \alpha$. Every natural number is a cardinal (a *finite cardinal*) and if S is a finite set, then $|S| = n$ for some natural number n. The ordinal ω is the least infinite cardinal, also denoted by $\aleph_0$ in the literature. All infinite cardinals are necessarily limit ordinals.

References

[1] K.R. Apt, H.A. Blair, and A. Walker. Towards a theory of declarative knowledge. In J. Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 89–148. Morgan Kaufmann Publishers, Los Altos, 1988.

[2] J.L. Balcázar, I. Díaz, and J. Gabarró. *Structural Complexity I.* Springer-Verlag, Berlin, 1988.

[3] C.R. Baral and V.S. Subrahmanian. Stable and extension class theory of logic programs and default logics. *Journal of Automated Reasoning*, 8:345–366, 1992.

[4] P. Baumgartner, P. Frölich, U. Furbach, and W. Nejdl. Semantically guided theorem proving for diagnosis applications. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence*, Japan, August 1997. Morgan Kaufmann.

[5] N. Bidoit and C. Froidevaux. General logical databases and programs: Default logic semantics and stratification. *Information and Computation*, 91:15–54, 1991.

[6] P. Bonatti. Autoepistemic logics as a unifying framework for the semantics of logic programs. *Journal of Logic Programming*, 22(2):91–149, 1995.

[7] A. Bondarenko, P.M. Dung, R.A Kowalski, and F. Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence*, 93:63–101, 1997.

[8] G. Boolos. *The Logic of Provability.* Cambridge University Press, 1993.

[9] S. Brass and J. Dix. Disjunctive semantics based on partial and bottom-up evaluation. In *Proceedings of the 12th International Conference on Logic Programming*, pages 199–213, Tokyo, June 1995. The MIT Press.

[10] S. Brass, J. Dix, and T. Przymusinski. Super logic programs. In L.C. Aiello and S. Doyle, J. Shapiro, editors, *Proceedings of the 5th International Conference on the Principles of Knowledge Representation and Reasoning*, pages 529–540. Morgan Kaufmann, 1996.

[11] G. Brewka. Cumulative default logic: in defense of nonmonotonic inference rules. *Artificial Intelligence*, 50:183–205, 1991.

[12] G. Brewka and G. Gottlob. Well-founded semantics for default logic. *Fundamenta Informaticae*, 31:221–236, 1997.

[13] K.L. Clark. Negation as failure. In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 293–322. Plenum Press, New York, 1978.

[14] S. A. Cook. The complexity of theorem proving procedures. In *Proceedings of the third Annual ACM Symposium on Theory of Computing*, pages 151–158, 1971.

[15] B.A. Davey and H.A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, 1990.

[16] J. de Kleer and K. Konolige. Eliminating the fixed predicates from a circumscription. *Artificial Intelligence*, 39(3):391–398, July 1989.

[17] J.P. Delgrande, T. Schaub, and W.K. Jackson. Alternative approaches to default logic. *Artificial Intelligence*, 70:167–237, 1994.

[18] J. Dix. A framework for representing and characterizing semantics of logic programs. In B. Nebel, V. Rich, and W. Swartout, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference (KR'92)*, pages 591–602, San Mateo, CA, 1992. Morgan Kaufmann.

[19] J. Dix. A classification theory of semantics of normal logic programs: I. strong properties. *Fundamenta Informaticae*, 22:227–255, 1995.

[20] J. Dix. A classification theory of semantics of normal logic programs: II. weak properties. *Fundamenta Informaticae*, 22:257–288, 1995.

[21] W.F. Dowling and J.H. Gallier. Linear-time algorithms for testing the satisfiability of propositional Horn formulae. *Journal of Logic Programming*, 3:267–284, 1984.

[22] J. Doyle. A truth maintenance system. *Artificial Intelligence*, 12:231–272, 1979.

[23] P. Dung. Negation as hypotheses: An abductive foundation for logic programming. In *the 8th International Conference on Logic Programming*, pages 3–17, 1991.

[24] T. Eiter and G. Gottlob. Complexity aspects of various semantics for disjunctive databases. In *Proceedings of the 12th Symposium on Principles of Database Systems*, pages 158–167, 1993.

[25] T. Eiter and G. Gottlob. Propositional circumscription and extended closed world reasoning are Π_2^p-complete. *Theoretical Computer Science*, 114:231–245, 1993.

[26] D.W. Etherington. Relating default logic and circumscription. In *Proceedings of the 10th International Joint Conference on Artificial Intelligence*, pages 489–494, Milan, Italy, August 1987. Morgan Kaufmann Publishers.

[27] M. Fitting. *First-Order Logic and Automated Theorem Proving.* Springer-Verlag, New York, 1990.

[28] M. Fitting. *Basic Modal Logic*, volume 1 of *Handbook of Logic in Artificial Intelligence and Logic Programming.* Oxford Science Puclications, 1993.

[29] M. Fitting, W. Marek, and M. Truszczyński. The pure logic of necessitation. *Journal of Logic and Computation*, 2(3):349–373, 1992.

[30] G. Friedrich, G. Gottlob, and W. Nejdl. Physical impossibility instead of fault models. In W. Hamscher, L. Console, and J. de Kleer, editors, *Readings in Model-based Diagnosis*, chapter 4, pages 159–164. Morgan Kaufmann, San Mateo, CA, 1992.

[31] D. Gabbay. Theoretical foundations for non-monotonic reasoning in expert systems. In *Logics and Models of Concurrent Systems*, pages 429–457. Springer-Verlag, 1985.

[32] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In *Proceedings of the 5th International Conference on Logic Programming*, pages 1070–1080, Seattle, USA, August 1988. The MIT Press.

[33] M. Gelfond and V. Lifschitz. Logic programs with classical negation. In *Proceedings of the 7th International Conference on Logic Programming*, pages 579–597, Jerusalem, Israel, June 1990. The MIT Press.

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

[35] M. Gelfond and V. Lifschitz. Representing action and change by logic programs. *Journal of Logic Programming*, 17, 1993.

[36] R. Goldblatt. *Logics of Time and Computation.* CSLI, 1987.

[37] G. Gottlob. Complexity results for nonmonotonic logics. *Journal of Logic and Computation*, 2(3):397–425, June 1992.

[38] G. Gottlob. The complexity of default reasoning under the stationary fixed point semantics. *Information and Computation*, 121(1):81–92, 1995.

[39] G. Gottlob. Translating default logic into standard autoepistemic logic. *Journal of the Association for Computing Machinery*, 42(2):711–740, 1995.

[40] G. Gottlob and M. Truszczyński. Approximating the stable model semantics is hard. *Fundamenta Informaticae 28*, 28(1-2):123–128, 1996.

[41] G.E. Hughes and M.J. Cresswell. *A Companion to Modal Logic*. Methuen and Co., London, 1984.

[42] T. Imielinski. Results on translating defaults to circumscription. *Artificial Intelligence*, 32:131–146, 1987.

[43] T. Janhunen. Investigations on cautious autoepistemic reasoning. Research report A31, Helsinki University of Technology, Digital Systems Laboratory, Espoo, Finland, December 1994.

[44] T. Janhunen. Cautious models for general logic programs. In H. Decker, U. Geske, T. Kakas, C. Sakama, and T. Seipel, D. Urpi, editors, *Proceedings of the Third International Workshop on Deductive Databases and Logic Programming*, pages 25–39, Kanagawa, Japan, June 1995. Gesellschaft für Mathematik und Datenverarbeitung. Studien Nr. 266.

[45] T. Janhunen. On abstracting decision procedures for non-monotonic reasoning. In I. Niemelä, editor, *Proceedings of the ECAI'96 Workshop on Integrating Nonmonotonicity into Automated Reasoning Systems*, pages 29–31, August 1996. Universität Koblenz-Landau, Institut für Informatik, Fachberichte Informatik, 18–96.

[46] T. Janhunen. Representing autoepistemic introspection in terms of default rules. In *Proceedings of the European Conference on Artificial Intelligence*, pages 70–74, Budapest, Hungary, August 1996. John Wiley.

[47] T. Janhunen. Separating disbeliefs from beliefs in autoepistemic reasoning. In J. Dix, U. Furbach, and A. Nerode, editors, *Proceedings of the 4th International Conference on Logic Programming and Non-Monotonic Reasoning*, pages 132–151, Dagstuhl, Germany, July 1997. Springer-Verlag. LNAI 1265.

[48] T. Janhunen and I. Niemelä. A scheme for weakened negative introspection in autoepistemic reasoning. In *Proceedings of the 3rd Kurt Gödel Colloquium on Computational Logic and Proof Theory*, pages 211–222, Brno, Czech Republic, August 1993. Springer-Verlag. LNCS 713.

[49] T. Jech. *Set Theory*. Pure and Applied Mathematics. Academic Press, 1978.

[50] D.S. Johnson. A catalog of complexity classes. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume A. Algorithms and Complexity, pages 67–161. Elsevier, 1990.

[51] S. C. Kleene. *Introduction to Metamathematics*, volume 1 of *Bibliotecha mathematica*. North-Holland, 1971.

[52] K. Konolige. On the relation between default and autoepistemic logic. *Artificial Intelligence*, 35:343–382, 1988.

[53] K. Konolige. Errata: On the relation between default and autoepistemic logic [Artificial Intelligence 35 (1988) 343-382]. *Artificial Intelligence*, 41:115, 1989.

[54] K. Konolige. Abduction versus closure in causal theories. *Artificial Intelligence*, 53:255–272, 1992.

[55] K. Konolige. Autoepistemic logic. In D.M. Gabbay, C.J. Hogger, and J.A. Robinson, editors, *Handbook of Logic and Artificial Intelligence and Logic Programming, Volume 3: Nonmonotonic Reasoning and Uncertain Reasoning*, pages 217–295. Oxford University Press, 1994.

[56] S. Kraus, D. Lehmann, and M. Magidor. Nonmonotonic reasoning, preferential models and cumulative logics. *Artificial Intelligence*, 44:167–207, 1990.

[57] G. Lakemeyer. All they know: A study in multi-agent autoepistemic reasoning. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, pages 376–381, Chambéry, France, August 1993. Morgan Kaufmann.

[58] D. Lehmann and M. Magidor. What does a conditional knowledge base entail. *Artificial Intelligence*, 55(1):1–60, 1992.

[59] V. Lifschitz. Computing circumscription. In *Proceedings of the 9th International Joint Conference on Artificial Intelligence*, pages 121–127, Los Angeles, California, USA, August 1985. Morgan Kaufmann Publishers.

[60] F. Lin and Y. Shoham. A logic of knowledge and justified assumptions. *Artificial Intelligence*, 57:271–289, 1992.

[61] J.W. Lloyd. *Foundations of Logic Programming.* Springer-Verlag, Berlin, 1984.

[62] J. Lobo, J. Minker, and A. Rajasekar. *Foundations of Disjunctive Logic Programming.* The MIT Press, 1992.

[63] D. Makinson. A general theory of cumulative inference. In *Proceedings of the 2nd Workshop on Non-Monotonic Reasoning*, pages 1–18, Grassau, FRG, June 1988. Springer-Verlag.

[64] W. Marek, G.F. Schwarz, and M. Truszczyński. Modal nonmonotonic logics: Ranges, characterization, computation. *Journal of the ACM*, 40(4):963–990, 1993.

[65] W. Marek and M. Truszczyński. Relating autoepistemic and default logics. In *Proceedings of the 1st International Conference on Principles of Knowledge Representation and Reasoning*, pages 276–288, Toronto, Canada, May 1989.

[66] W. Marek and M. Truszczyński. Modal logic for default reasoning. *Annals of Mathematics and Artificial Intelligence*, 1:275–302, 1990.

[67] W. Marek and M. Truszczyński. *Nonmonotonic Logic: Context-Dependent Reasoning.* Springer-Verlag, Berlin, 1993.

[68] J. McCarthy. Circumscription—a form of non-monotonic reasoning. *Artificial Intelligence*, 13:27–39, 1980.

[69] D. McDermott. Non-monotonic logic II. *Journal of the ACM*, 29:33–57, 1982.

[70] D. McDermott and J. Doyle. Non-monotonic logic I. *Artificial Intelligence*, 13:41–72, 1980.

[71] S. Mittal and F. Frayman. Towards a generic model of configuration tasks. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, pages 1395–1401, Detroit, Michigan, 1989. Kaufmann.

[72] R.C. Moore. Semantical considerations on nonmonotonic logic. In *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, pages 272–279, Karlsruhe, FRG, August 1983. Morgan Kaufmann Publishers.

[73] R.C. Moore. Semantical considerations on nonmonotonic logic. *Artificial Intelligence*, 25:75–94, 1985.

[74] A. Nerode and R. Shore. *Logic for Applications.* Springer-Verlag, 1993.

[75] I. Niemelä. Towards automatic autoepistemic reasoning. In *Proceedings of the European Workshop on Logics in Artificial Intelligence—JELIA'90*, pages 428–443, Amsterdam, The Netherlands, September 1990. Springer-Verlag.

[76] I. Niemelä. Constructive tightly grounded autoepistemic reasoning. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, pages 399–404, Sydney, Australia, August 1991. Morgan Kaufmann Publishers.

[77] I. Niemelä. On the decidability and complexity of autoepistemic reasoning. *Fundamenta Informaticae*, 17(1,2):117–155, 1992.

[78] I. Niemelä. A unifying framework for nonmonotonic reasoning. In *Proceedings of the 10th European Conference on Artificial Intelligence*, pages 334–338, Vienna, Austria, August 1992. John Wiley.

[79] I. Niemelä. Autoepistemic logic as a unified basis for nonmonotonic reasoning. Doctoral dissertation. Research report A24, Helsinki University of Technology, Digital Systems Laboratory, Espoo, Finland, August 1993.

[80] I. Niemelä. A decision method for nonmonotonic reasoning based on autoepistemic reasoning. *Journal of Automated Reasoning*, 14:3–42, 1995.

[81] I. Niemelä. Towards efficient default reasoning. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 312–318, Montreal, Canada, August 1995. Morgan Kaufmann Publishers.

[82] I. Niemelä. Implementing circumscription using a tableau method. In *Proceedings of the European Conference on Artificial Intelligence*, pages 80–84, Budapest, Hungary, August 1996. John Wiley.

[83] I. Niemelä. A tableau calculus for minimal model reasoning. In *Proceedings of the Fifth Workshop on Theorem Proving with Analytic Tableaux and Related Methods*, pages 278–294, Terrasini, Italy, May 1996. Springer-Verlag.

[84] I. Niemelä and J. Rintanen. On the impact of stratification on the complexity of nonmonotonic reasoning. *Journal of Applied Non-Classical Logics*, 4(2):141–179, 1994.

[85] I. Niemelä and P. Simons. Efficient implementation of the well-founded and stable model semantics. In M. Maher, editor, *Proceedings of the Joint International Conference and Symposium on Logic Programming*, pages 289–303, Bonn, Germany, September 1996. The MIT Press.

[86] C.H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[87] J. Pearl. System Z: Natural ordering of defaults with tractable applications to nonmonotonic reasoning. In R. Parikh, editor, *Theoretical Aspects of Reasoning about Knowledge*, pages 121–135. Morgan Kaufmann, 1990.

[88] L. M. Pereira, J. J. Alferes, and J. N. Aparício. Adding closed world assumptions to well-founded semantics. *Theoretical Computer Science*, 122:49–68, 1994.

[89] H. Przymusinska and T.C. Przymusinski. Stationary default extensions. In *Working Notes of the 4th International Workshop on on Nonmonotonic Reasoning*, pages 179–193, Plymouth, Vermont, USA, May 1992.

[90] T. C. Przymusinski. Static semantics for normal and disjunctive logic programs. *Annals of Mathematics and Artificial Intelligence*, 14:323–357, 1995.

[91] T.C. Przymusinski. Autoepistemic logics of closed beliefs and logic programming. In *Proceedings of the 1st International Workshop on Logic Programming and Non-monotonic Reasoning*, pages 3–20, Washington, D.C., USA, July 1991. The MIT Press.

[92] T.C. Przymusinski. Three-valued nonmonotonic formalisms and semantics of logic programming. *Artificial Intelligence*, 49:309–343, 1991.

[93] R. Reiter. On closed world data bases. In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 55–76. Plenum Press, New York, 1978.

[94] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13:81–132, 1980.

[95] R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32:57–95, 1987.

[96] V. Risch. Cumulative default theories vs non-cumulative default logics. In *In Working Notes of the 6th International Workshop on Nonmonotonic Reasoning*, pages 116–122, Timberline, Oregon, June 1996.

[97] D. Saccà and C. Zaniolo. Partial models and three-valued stable models in logic programs with negation. In *the Workshop on Nonmonotonic Reasoning and Logic Programming*, pages 87–101, 1991.

[98] C. Sakama and K. Inoue. Relating disjunctive logic programs to default theories. In *Proceedings of the 2nd International Conference on Logic Programming and Non-Monotonic Reasoning*, pages 266–282. MIT Press, 1993.

[99] T. Schaub. On commitment and cumulativity in default logics. In *Proceedings of European Conference on Symbolic and Quantitative Approaches to Uncertainty*, pages 304–309, Marseilles, France, 1991. Springer-Verlag.

[100] T. Schaub. On constrained default theories. In *Proceedings of the 10th European Conference on Artificial Intelligence*, pages 304–308, Vienna, Austria, August 1992. John Wiley.

[101] G. Schwarz. On embedding default logic into Moore's autoepistemic logic. *Artificial Intelligence*, 80:349–359, 1996.

[102] G. Schwarz and M. Truszczyński. Nonmonotonic reasoning is sometimes simpler. In *Proceedings of the 3rd Kurt Gödel Colloquium on Computational Logic and Proof Theory*, pages 313–324, Brno, Czech Republic, August 1993. Springer-Verlag.

[103] R.M. Smullyan. *First-Order Logic*. Springer-Verlag, Berlin, 1968.

[104] R. Stalnaker. A note on non-monotonic modal logic. *Artificial Intelligence*, 64:183–196, 1993.

[105] L. Sterling and E. Shapiro. *The Art of Prolog*. MIT Series in logic programming. The MIT Press, London, 1986.

[106] J. Stillman. The complexity of propositional default logics. In *Proceedings of the 10th National Conference on Artificial Intelligence*, pages 794–800, San Jose, California, USA, July 1992. The MIT Press.

[107] P. Struss and O. Dressler. Integrating fault models into the general diagnostic engine. In W. Hamscher, L. Console, and J. de Kleer, editors, *Readings in Model-based Diagnosis*, chapter 4, pages 153–158. Morgan Kaufmann, San Mateo, CA, 1992.

[108] V.S. Subrahmanian and C. Zaniolo. Database updates and AI planning domains. LPNMR-archive at al.cs.engr.uky.edu, 1995.

[109] A. Tarski. *Logic, Semantics, Metamathematics: Papers From 1923 to 1938*. Clarendon Press, Oxford, 1956.

[110] H. Thiele. On generation of cumulative inference operators by default deduction rules. In *LNAI 543*, pages 100–137, 1991.

[111] 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 10th International Joint Conference on Artificial Intelligence*, pages 476–482, Milan, Italy, August 1987. Morgan Kaufmann Publishers.

[112] M. Truszczyński. Embedding default logic into modal nonmonotonic logics. In *Proceedings of the 1st International Workshop on Logic Programming and Non-monotonic Reasoning*, pages 151–165, Washington, D.C., USA, July 1991. The MIT Press.

[113] M. Truszczyński. Modal interpretations of default logic. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, pages 393–398, Sydney, Australia, August 1991. Morgan Kaufmann Publishers.

[114] M.H. van Emden and R.A. Kowalski. The semantics of predicate logic as a programming language. *Journal of the ACM*, 23:733–742, 1976.

[115] A. Van Gelder, K.A. Ross, and J.S. Schlipf. The well-founded semantics for general logic programs. *Journal of the ACM*, 38(3):620–650, July 1991.

[116] X. Wang, J.-H. You, and L.Y. Yuan. Nonmonotonic reasoning by monotonic inferences with priority constraints. In *Proceedings of the 2nd International Workshop on Non-Monotonic Extensions of Logic Programming*, pages 91–109. Springer, 1996. LNAI 1216.

[117] B. Wielinga and G. Schreiber. Configuration-design problem solving. *IEEE Expert*, 1997.

[118] J. You and L. Yuan. On the equivalence of semantics for normal logic programs. In *Proceedings of Workshop on Logic Programming with Incomplete Information*, Vancouver, October 1993.

[119] J. You and L. Yuan. A three-valued semantics for deductive databases and logic programs. *Journal of Computer and System Sciences*, 49:334–361, 1994.

[120] L. Yuan. Autoepistemic logic of first order and its expressive power. *Journal of Automated Reasoning*, 13:69–82, 1994.

Index