On the Complexity of Fragments of Nonmonotonic Logics

Von der Fakultät für Elektrotechnik und Informatik
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des Grades

Doktor der Naturwissenschaften
Dr. rer. nat.

genehmigte Dissertation
von

M. Sc. Michael Thomas

geboren am 23. April 1981 in Hannover

2010

Referent:	Heribert Vollmer, Leibniz Universität Hannover
Koreferent:	Nadia Creignou, Université d'Aix-Marseille II
Tag der Promotion:	04. November 2010

Zwei Dinge sind zu unserer Arbeit nötig: Unermüdliche Ausdauer und die Bereitschaft, etwas, in das man viel Zeit und Arbeit gesteckt hat, wieder wegzuwerfen.

Albert Einstein

Danksagung

Ich möchte meinem Doktorvater Heribert Vollmer für die Betreuung und Unterstützung in meiner Zeit als Doktorand an der Universität Hannover danken. Er hat während meines Studiums mein Interesse an der theoretischen Informatik geweckt und bot mir anschließend die Möglichkeit, in diesem Gebiet zu arbeiten und zu forschen. Ich danke auch meinen Kollegen Arne Meier und Peter Lohmann für die vielen hilfreichen Diskussionen und Anmerkungen zu meiner Arbeit. Außerdem möchte ich mich bei meinen Koautoren für die gemeinsame Forschung bedanken, besonders bei Nadia Creignou für die Zeit in Marseille und viele hilfreiche Kommentare bezüglich des auf der Konferenz LPNMR 2009 erschienenen Teils dieser Arbeit.

Vor allem aber möchte ich mich bei meiner Familie und insbesondere meiner Frau Annika für die Liebe und Unterstützung bedanken.

Acknowledgements

I want to thank my doctoral advisor Heribert Vollmer for the supervision and support during my time as a PhD student at the University of Hanover. It was he who introduced me to theoretical computer science as a student and offered me the possibility to work in this field. I am also grateful to my colleagues Arne Meier and Peter Lohmann for the fruitful discussions and suggestions. Moreover, I wish to thank my coauthors for our joint research, in particular Nadia Creignou for the invitation to Marseille and many helpful comments on the parts of this thesis that were published at the conference LPNMR 2009.

Most of all, I want to thank my family and particularly my wife Annika for the love and support she gave me.

ZUSAMMENFASSUNG

Nichtmonotones Schließen ist eine der wichtigsten Aufgaben im Bereich der Wissensrepräsentation und künstlichen Intelligenz. Verschiedenste Logiken wurden entwickelt, um nichtmonotones Schließen zu formalisieren. In dieser Arbeit betrachten wir drei solcher Logiken, die die Nichtmonotonie auf unterschiedliche Weise modellieren: Default Logik, Autoepistemische Logik und Circumscription.

Wir untersuchen die Komplexität verschiedener Konsistenz-, Folgerungs- und Zählprobleme für Fragmente dieser Logiken, die durch die Beschränkung der verfügbaren Booleschen Operatoren entstehen, sowie die Möglichkeit zwischen Fragmenten dieser Logiken zu übersetzen. Zu diesem Zweck verallgemeinern wir die oben genannten Logiken, indem wir allgemeine Boolesche Operatoren anstelle der üblichen Standardbasis zulassen, und betrachten die Komplexität der Probleme und Existenz von Übersetzungen für alle endlichen Mengen von Booleschen Operatoren.

Unsere Resultate zeigen, dass in allen Fällen die Komplexität der betrachteten Probleme nicht von der speziellen Menge von erlaubten Operatoren B abhängt, sondern von der Menge der Funktionen, die sich aus B mit Hilfe von Projektionen und Komposition bilden lässt. Darüber hinaus nimmt die Komplexität der untersuchten Entscheidungsprobleme für alle möglichen Operatormengen nur endlich viele Komplexitätsgrade an. Die Komplexität der Zählprobleme wiederum ist bis auf eine interessante Ausnahme trichotom, wobei die auftretenden Komplexitätsgrade die untersten drei Stufen der Zählhierarchie umfassen.

Schließlich untersuchen wir die Existenz von Übersetzungen zwischen Fragmenten der genannten Logiken, für die die Menge der logischen Schlußfolgerungen invariant ist. Wir zeigen, welche Fragmente der Default Logik, der Autoepistemischen Logik und von Circumscription sich unter dem gewählten Übersetzungsbegriff in Fragmente der jeweils anderen zwei Logiken übersetzen lassen. Diese Ergebnisse werden komplettiert durch die Feststellung, dass in fast allen Fällen, in denen keine Übersetzungen angegeben sind, Übersetzungen nur dann existieren können, wenn die Polynomialzeithierarchie kollabiert.

SCHLAGWORTE: Nichtmonotone Logik, Komplexität, Post'scher Verbund

ABSTRACT

Nonmonotonic reasoning is one of the most important tasks in the area of knowledge representation and reasoning. Several logics have been developed to formalize nonmonotonic reasoning. In this thesis we consider three well-known logics that facilitate nonmonotonic reasoning by different means: default logic, autoepistemic logic and circumscription. We study the computational complexity of consistency, reasoning and counting problems for fragments of these logics obtained by restricting the available Boolean connectives, as well as the possibility to translate between these fragments. For this we generalize the logics to allow for arbitrary connectives rather than the Boolean standard base and study the complexity of the problems and possibility of translations for all finite sets of allowed Boolean connectives.

Our results show that in all cases the complexity of the problems does not depend on the particular set B of available connectives but on the set of functions expressible by projections and arbitrary compositions from B. We obtain polytomous complexity classifications (that is, into a finite number of complexity degrees) for all decision problems studied herein ranging from completeness for classes in the second level of the polynomial hierarchy down to membership in AC^0. Furthermore, the counting problems are with one interesting exception shown to be trichotomous with complexity degrees spanning the first three levels of the counting hierarchy. To the best of our knowledge, the counting complexity of default logic is addressed here for the first time.

Finally, we consider translations between fragments of these logics that leave the set of propositional consequences of the input invariant. We show which fragments of default logic, autoepistemic logic and circumscription can, under the chosen notion of translations, be embedded into fragments of the other two logics. We complete this picture by showing that in almost all cases in which no translation is given, no translation preserving the set of propositional consequences may exist unless the polynomial hierarchy collapses.

KEYWORDS: nonmonotonic logic, computational complexity, Post's lattice

CONTENTS

List of Figures

Chapter 1

Introduction

1.1 Complexity Theory

Suppose that you are given a set of villages connected via dirt roads and are asked to tarmac a set of streets such that each pair of villages is connected via asphalted roads. Your budget is limited, hence you want to know whether this task can be solved with a given amount of money. This problem is fairly easy to solve: starting from an arbitrary village v_1, tarmac the shortest dirt road connecting v_1 to some not yet accessible village v_2. Now again choose the shortest dirt road as above starting from either v_1 or v_2, and so forth until all villages are connected.

This algorithm, also known as Prim's algorithm [Jar30, Pri57], will provide you with a minimal cost solution which you can compare to your budget. Moreover, the resources required to solve the problem are quite limited: one only needs to keep track of the set of villages already connected to each other and find the shortest dirt road leading from these to some not yet accessible village. But what if you are instead asked to tarmac a round trip that visits each village exactly once rather than an arbitrary set of streets. For this modification, the above strategy will no longer work. Indeed, no one has yet found an algorithm running in subexponential-time that answers the question whether you can tarmac a round trip. But can we be sure that no such algorithm exists? And in which way does the additional restriction make the problem computationally more involved?

These questions are typically studied in an area of theoretical computer science called *(computational) complexity theory*. This area analyzes the resources required to solve a computational problem and classifies these according to their inherent difficulty. One of the main goals of this area is to understand which problems are easy to solve, which problems are computationally hard, and of course, why. The class of easy decision problems is denoted by P and comprises those problems that are efficiently (that is, polynomial-time) solvable. The first of the above problems belongs to this class. For the second problem no polynomial-time algorithm is known; however, given a solution we can easily verify its correctness. Such problems are called *efficiently verifiable*, and the class of all such problems is denoted NP. As any efficiently solvable problem is also efficiently verifiable, we have $\mathrm{P} \subseteq \mathrm{NP}$. And while the question whether $\mathrm{P} = \mathrm{NP}$ or $\mathrm{P} \subsetneq \mathrm{NP}$ is one of the most important open problems in computer science, the inability to prove or refute P = NP led to the development of a rich theory of computational complexity.

An important role in this context play the hardest problems in NP in the sense that an algorithm for any such problem can be transformed into one for any problem in NP. These problems are called NP-*complete*. An efficient algorithm for an NP-complete problem would thus allow for the efficient solution of all problems in NP, that is, P = NP. The first problem shown to be NP-complete was SAT, the satisfiability problem for propositional formulae [Coo71].

In this thesis, we will encounter problems that do not fall into the classes P or NP, for example, problems whose complement lies in NP. This class of problems for which the absence of solutions can be verified in polynomial-time, is known as coNP. We also require classes for problems that are harder to solve than SAT in the sense that they are only known to be efficiently verifiable if provided with an oracle that is able to instantaneously answer queries to a language in NP. These problems are called efficient verifiable relative to an NP-oracle. For example, the problem to determine whether the lexicographic smallest assignment of a formula sets to *true* a certain proposition is known to be efficiently verifiable relative to an NP-oracle but not known to be in NP or coNP. One can now consider problems that are efficiently verifiable relative to such problems, and so on. The concept of efficient verification relative to an oracle thus naturally leads to a hierarchy of complexity classes known as the *polynomial hierarchy*. The $(i+1)$th level of this hierarchy comprises the class $\Sigma^{\mathrm{p}}_{i+1}$ of problems known to be efficiently verifiable given an oracle for the ith level and the class Π^{p}_{i+1} of their complements, where the Σ^{p}_0 and Π^{p}_0 are defined as P.

We will use this rich framework of complexity theory to classify the complexity of computational problems connected to logics for knowledge representation and commonsense reasoning.

1.2 Nonmonotonic Logic

One of the most intriguing aspects of human reasoning is its flexibility and speed. Despite the fact that in most situations we do not have all relevant knowledge at hand, commonsense enables one to draw conclusions by virtue of plausible assumptions. These assumptions might be invalidated by new information about the world; therefore human reasoning is said to be *nonmonotonic*.

For example, suppose that you need some advice from a colleague. As his office is empty and it is noon, you conclude that he is gone for lunch; a conclusion derived from an assumption about his usual behaviour. Learning that he is on a business trip now invalidates your old conclusion.

From the very beginning of knowledge representation and reasoning, it has been argued that classical logic is not suited to formalize the process of human reasoning, mainly for its inherent monotonicity: once a statement is derivable it may never be invalidated regardless of whatever knowledge one might gain.

To overcome this deficiency, nonmonotonic logics have been introduced around 1980 [McC80, MD80, Rei80]. These logic can be distinguished by the way they facilitate nonmonotonic behaviour:

1. by extension with new inference rules,
2. by extension with modal operators,
3. by modification of the semantics.

In this thesis, we will examine one logic from each of the above approaches and study the complexity of natural problems arising in these. In particular, we focus on the following well-known logics.

Default logic has been introduced by Reiter [Rei80] and extends classical (first-order or propositional) logic with inference rules of the form $\frac{\alpha:\beta}{\gamma}$, called *default rules*. The default rule $\frac{\alpha:\beta}{\gamma}$ allows to conclude γ if the premise α is derivable and the justification β can consistently be assumed.

Autoepistemic logic has been introduced by Moore [Moo85] and extends classical logic with a unary "introspective" operator L expressing belief. For a formula φ, $L\varphi$ states that an ideally rational agent can derive φ.

Circumscription has been introduced by McCarthy [McC80]. Rather than extending classical logic, it restricts the notion of satisfiability and inference to consider the minimal model of a formula only. It has been shown that circumscription as defined by Lifschitz [Lif85] is equivalent to reasoning under the *extended closed world assumption*, which for a designated set P allows to assume $\neg p$ whenever $p \in P$ is not derivable [GPP89].

The extensions introduced by default or autoepistemic logic condition the derivable knowledge on a set of beliefs. Therefore maximal stable sets of knowledge supersede the traditional deductive closure. For default logic these are called *stable extensions*; for autoepistemic logic, *stable expansions*. A default or an autoepistemic theory may possess multiple or no such maximal stable sets of knowledge. Thus the following questions naturally arise: Does a given set of formulae admit a maximal stable set of knowledge? A lack thereof would correspond to the case that for all possible sets of beliefs one eventually arrives at contradictory information. The problem hence asks whether one can obtain consistent knowledge of the world. This problem is a rough analogue of the satisfiability problem in propositional logics and will henceforth be referred to as the *extension* (respectively *expansion*) *existence* problem.

Beyond, the potential presence of multiple maximal stable sets of knowledge leads to two different interpretations for the question whether a certain information is derivable: the first, *credulous reasoning* (also referred to as *brave reasoning*), asks whether a formula is contained in at least one stable extension (respectively expansion) of the knowledge base; the second, *skeptical reasoning* (also referred to as *cautious reasoning*), asks whether the formula is contained in all stable extensions (respectively expansions). On an intuitive level, credulously entailed knowledge can be considered "possible", while skeptically entailed knowledge is "certain" in the sense that any possible interpretation of the world entails it. The associated decision problems are natural generalizations of the propositional implication problem and will henceforth be referred to as the *credulous reasoning* problem and the *skeptical reasoning* problem.

In the restricted semantics of minimal models no corresponding notion of maximal stable sets of knowledge exists. The corresponding notion in circumscription are minimal (or *circumscriptive*) models, which exist if and only if the given knowledge base is satisfiable. Therefore the problem of determining their existence of is equal to SAT. For circumscription we are hence restricted to the study of the skeptical reasoning problem, that is, to decide whether for a given set of formulae Γ and a formula φ, whether φ is true in all minimal models of Γ.

1.3 Results

While for extensions of first-order logic all of the above decision problems are undecidable, they are decidable for extensions of propositional logic—but presumably harder than the traditional satisfiability or implication problem: they are known to be complete for the second level of the polynomial hierarchy [Nie90, CL90, Got92, EG93]. For this reason, several semantic restrictions and parameterizations of these problems have been studied in the literature (see [CL90, KS91, NR94, KK03, Nor04, CHS07], amongst others).

In this thesis, we take a different approach and perform a systematic study of the complexity of the above extension (respectively expansion) existence and reasoning problems obtained by restricting the set of allowed Boolean connectives. To this end, we generalize the underlying problems to allow for arbitrary Boolean connectives rather than the Boolean standard base $\{\wedge, \vee, \neg\}$ and classify the complexity of these problems parameterized by the set of allowed Boolean connectives for all possible finite sets of Boolean connectives.

This approach has first been taken by Lewis [Lew79], who showed that the satisfiability problem is NP-complete if and only if the negation of the implication ($x \nrightarrow y$) can be composed from the given Boolean connectives. Such a dichotomous behaviour cannot be taken for granted due to Ladner's theorem: if $\mathrm{P} \neq \mathrm{NP}$ then there exists infinitely many degrees of complexity between P and NP-completeness [Lad75]. Since then, Lewis' approach has been applied to a wide range of problems including equivalence and implication problems [Rei03, BMTV09a], satisfiability and model checking in modal and temporal logics [BHSS06, BSS$^+$08, BMS$^+$09, MMTV09, MMS$^+$09], and abduction [CST10].

Herein we study whether a similarly polytomous complexity classification is possible for the extension (respectively expansion) existence and reasoning problems mentioned above. Our goal is to exhibit fragments of lower complexity which might lead to better algorithms for cases in which the set of Boolean connectives can be restricted. Furthermore we aim to understand the sources of hardness and to provide an understanding which connectives take the role of $x \nrightarrow y$ in the context of the nonmonotonic logics mentioned above, that is, which connectives account for jumps in the complexity of the problems. These connectives may help to identify candidates for parameters in the study of parameterized complexity of nonmonotonic logics.

To be more precise, let B denote the finite set of available Boolean connectives. Although at first sight, an infinite number of sets B of allowed Boolean connectives has to be examined, we prove, making use of results from universal algebra, that for all considered problems the complexity does not depend on the particular set but rather on the *clone* $[B]$ of B, that is, the set of functions which can be implemented from B using projections and arbitrary composition.

Decision Problems

We show that both the complexity of the extension existence problem in default logic and the complexity of the expansion existence problem in autoepistemic logic are polytomous (see Theorems 4.1.1 and 4.2.1):

the extension existence problem remains Σ_2^p-complete for all sets B such that $[B \cup \{1\}] = \mathsf{BF}$; becomes Δ_2^p-complete for monotone sets B that contain conjunctions, disjunctions and the constant 0; is NP-complete if $[B \cup \{1\}]$ contains $\neg$ and comprises affine functions only; and becomes tractable in all other cases (with this case splitting into P-complete, NL-complete, and trivial sub-cases). The expansion existence problem for autoepistemic logic, on the other hand, remains Σ_2^p-complete for all B such that $[B \cup \{0, 1\}]$ includes the Boolean functions $\wedge$ and $\vee$, is NP-complete if $[B]$ contains $\vee$ and the Boolean constants only, and becomes polynomial-time decidable in all other cases (with this case splitting into three different complexity degrees inside P).

For the credulous and skeptical reasoning problems in default logic and autoepistemic logic, the situation is more diverse as there are two sources for the complexity: On the one hand, we need to determine a finite characterization of a candidate for a stable extension (respectively expansion). And, on the other hand, we have to verify that this candidate is indeed a finite characterization as desired—a task that requires to test for formula implication. Depending on the Boolean connectives allowed, one or both tasks can be performed in polynomial time or even become trivial. In principle, this yields five possible cases for the complexity of the problems, and we will see that all five cases actually occur. In principle, this yields five possible cases for the complexity of the problems, and we will see that all four cases actually occur.

We obtain Σ_2^p-completeness for the skeptical reasoning problems and Π_2^p-completeness for the credulous reasoning problems for all clones where both the stable extension and the implication problem attain their highest complexity.

For default logic, the complexity of the reasoning problems reduces to Δ_2^p for clones that allow for an efficient computation of stable extensions but whose implication problem remains coNP-complete. More precisely, these problems are Δ_2^p-complete if a stable extension may not exist and becomes coNP-complete otherwise. Conversely, if the implication problem becomes easy but determining an extension candidate is hard, then the credulous reasoning problem is NP-complete, while the skeptical reasoning problems is coNP-complete. Similarly for autoepistemic logic, the credulous and skeptical reasoning problems become complete for respectively NP and coNP if the implication problem is tractable but determining an expansion candidate is hard. Finally, for clones that allow for solving both tasks in polynomial time all reasoning problems become tractable (with these cases splitting up into different complexity degrees ranging from membership in AC^0 to completeness for P). We hence obtain polytomous classifications of the computational complexity of the problems, where for the credulous reasoning problem in default logic, notably, complete fragments for all classes of the polynomial hierarchy below Σ_2^p occur. In contrast to this, the complexity of credulous and skeptical reasoning in autoepistemic logic decreases in coarser steps. These results are presented in Theorems 5.1.1, 5.1.5, 5.2.1 and 5.2.4.

As for circumscription, the complexity of the skeptical reasoning problem is Π_2^p-complete for all clones such that the implication problem and the problem to determine the minimality of models are intractable. If all available functions are affine or monotone, then the complexity of the problem is contained in coNP, where it is coNP-complete in the former case as long as $\vee$ remains expressible using the available functions and the constant 1. This decrease in the complexity comes from different sources: for monotone functions the test for minimality of models becomes tractable, while for affine functions the implication problem becomes tractable. Finally, if the set of available functions is further restricted to contain either only negations or only conjunctions, then the problem becomes polynomial-time solvable (its complexity drops to respectively $\mathrm{AC}^0[2]$-completeness or membership in AC^0). This is summarized in Theorem 5.3.1. We point out that the implication problem and the problem to determine the minimality of models do not completely determine the complexity of the skeptical reasoning problem: for all sets B such that $[B \cup \{0,1\}]$ contains $\vee$ and the Boolean constants only, the latter problem remains coNP-complete whereas the implication problem and minimality of models can be decided in polynomial time.

Counting Problems

Besides the decision variants, another natural question is concerned with the number of stable extensions (respectively expansions) or the number of minimal models. This question refers to *counting problems*. Recently, counting problems have gained quite a lot of attention in nonmonotonic logics. For circumscription, the counting problem (that is, determining the number of minimal models of a propositional formula) has been studied in [DHK05, DH08]. For propositional abduction, a nonmonotonic formalism for computing explanations, some complexity results on the problem of counting the number of "solutions" to a propositional abduction problem were presented in [HP07, CST10]. Algorithms based on bounded treewidth have been proposed in [JPRW08] for the counting problems in abduction and circumscription. Here, we consider the complexity of the problem to count the number of stable extensions, stable expansions and minimal models of a given knowledge base. To the best of our knowledge, the first problem is addressed here for the first time.

In particular, we show in Theorem 6.1.1 that for sets B of Boolean connectives such that $[B \cup \{1\}]$ is functional complete counting the number of stable extensions is complete for the second level of the counting hierarchy; becomes Δ_2^{p}-complete for all monotone sets B such that $[B \cup \{1\}] = \mathsf{M}$; is #P-complete for affine sets B such that $\neg$ can be implemented from $B \cup \{1\}$; and becomes efficiently computable in all other cases. In autoepistemic logic, the complexity of counting the number of stable expansions is trichotomous and decreases analogously to the complexity of the stable expansion problem, see Theorem 6.2.1.

We think it is important to note that for our classification of the two counting problems above the conceptually simple parsimonious reductions are sufficient, while for related classifications in the literature less restrictive (and more complicated) reductions such as subtractive or complementive reductions had to be used (see, for example, [DHK05, DH08, BBC$^+$09] and some of the results of [HP07]). Parsimonious reductions are not only the conceptually simplest ones since they are direct analogues of the usual many-one reductions among languages. They also form the strongest (or strictest) type of reduction with a number of good properties, for example, all relevant counting classes are closed under parsimonious reductions.

Lastly, the complexity of counting the number of minimal models is classified in Theorem 6.3.1. Unlike the preceding counting problems, here we have sets of Boolean functions for which the problem to decide whether a given assignment is a circumscriptive model is tractable while the corresponding counting problem is #P-complete (namely affine sets of Boolean functions that implement the ternary exclusive-or). In all remaining cases, its complexity can be derived from the complexity of the skeptical reasoning problem in circumscription in the way that completeness for the second level of the polynomial hierarchy translates to #·coNP-completeness, completeness for the first level translates to #P-completeness, and membership in P translates to membership in FP. However, mind that the decision problem underlying the circumscriptive model counting problem is the question whether there exists a minimal model for the given formula—a problem equivalent to the satisfiability problem for propositional formulae. It thus represents a counting problem whose underlying decision problem is, though intractable, supposedly easier to solve than the decision problems underlying the generic complete problem for #·coNP.

TRANSLATIONS

On the basis of these results, we finally examine the possibility of translations preserving the derivability of propositional formulae between fragments of the nonmonotonic logics introduced above.

We prove that, with respect to equality of the set of skeptically entailed formulae, not only default logic can be embedded into autoepistemic logic but that the latter can also be embedded into the former (Theorems 7.2.1 and 7.2.7). Thus, in case one is interested in the set of consequences of the given theory only, one may switch between default and autoepistemic logic. This complements results of Janhunen [Jan99], who proves that with respect to translations preserving stable extensions (respectively expansions), default logic is strictly more expressive. In addition to that, we prove that monotone autoepistemic logic embeds monotone default logic and, quite remarkably, that autoepistemic logic of disjunctions can be embedded into the fragments of default logic containing negations as the sole Boolean connective.

Concerning translations of circumscription into the above two logics, we show in Theorems 7.3.1 and 7.4.1 that, although translations into both full default logic as well as full autoepistemic logic are possible, the results for fragments of this logic differ significantly. While circumscription restricted to Boolean functions from B can be modularly embedded to default logic whenever $\neg$ can be implemented in default logic and all functions from B can be simulated, the analogous statement for autoepistemic logic is more restrictive: a translation from circumscription into a not functional complete fragment of autoepistemic logic exists only if the circumscriptive theory is equivalent to a set of literals. Thus, while both autoepistemic logic and default logic are capable of embedding circumscription, in default logic the concept of default rules allows a translation that separately translates the knowledge base and the nonmonotonic features of circumscription.

For the converse direction, translations from default logic or autoepistemic logic to circumscription are only possible for very restrictive sets of Boolean functions, namely those for which the skeptical reasoning problem is tractable. These results confirm the intuition that circumscription is less expressive than autoepistemic logic or default logic, not only for the full fragment but also the fragments obtained by restricting the set of available Boolean functions (Theorems 7.3.5 and 7.4.6).

Beyond these translatability results, we prove that for almost all remaining pairs of fragments for which no translation is given, no translation is possible unless the polynomial hierarchy collapses to its first or second level.

1.4 PUBLICATIONS

Section 2.5 was previously published in [BMTV09a]. The results on the complexity of the extension existence and reasoning problems for default logic in

Sections 4.1 and 5.1 previously appeared in [BMTV09b]; similarly, the corresponding results on autoepistemic logic in Sections 4.2, 5.2 and 6.2 have been published in [CMTV10]. The results from Section 5.3 on the complexity of reasoning in circumscription appeared in [Tho09]. The remaining sections of Chapter 6 and Chapter 7 contain unpublished results.

CHAPTER 2

PRELIMINARIES

This chapter introduces the basic definitions and concepts relevant throughout this thesis and states the relevant results from complexity theory. Section 2.5 contains own results on the complexity of the propositional implication problem which will be used several times in the subsequent chapters.

2.1 BASIC NOTATIONS

We assume that the reader is familiar with basic mathematical structures like sets, functions, partial orders, and the basic notions from theoretical computer science. We will also use without explanation the terms positive and negative literal, clause, conjunctive normal form and disjunctive normal form from mathematical logic.

The set of natural numbers $\{0, 1, 2, \ldots\}$ is denoted by $\mathbb{N}$, the set of integers by $\mathbb{Z}$. A *lattice* is a partially ordered set $(A, \leq)$ such that for any two elements $a, b \in A$, there exist a greatest lower bounded and a least upper bound in $(A, \leq)$. We use $|\cdot|$ to denote both the length of strings and the cardinality of sets. As usual, we identify decision problems with languages, that is, with the set of its "yes"-instances.

The symbols 0 and 1 represent the Boolean constants *false* and *true*. A *Boolean function* is a function $f\colon \{0,1\}^n \to \{0,1\}$ for some $n \in \mathbb{N}$. We identify the n-ary logical connective c with the n-ary Boolean function f defined by $f(a_1, \ldots, a_n) :=$ 1 if and only if the formula $c(x_1, \ldots, x_n)$ evaluates to true when assigning a_i to x_i for all $1 \leq i \leq n$. The symbols $\wedge$, $\vee$ and $\neg$ are used to denote the logical conjunction, disjunction and negation, respectively. The symbol $\oplus$ denotes the logical exclusive-or, the symbol $\rightarrow$ logical implication, and the symbol $\leftrightarrow$ logical equivalence.

2.2 COMPLEXITY THEORY

In order to determine the resources necessary to solve a computational problem, we use the common terminology from complexity theory. An introduction to this terminology can, for example, be found in [Pap94] or [AB09]. We give here a brief recollection of the relevant classes and results.

2.2.1 Machine Based Complexity Classes

The class P (respectively NP) comprises those problems solvable in polynomial time on a deterministic (respectively nondeterministic) Turing machine. Similarly, L (respectively NL) is defined as the class of problems that can be computed in logarithmic space on a deterministic (respectively nondeterministic) Turing machine. Obviously, $L \subseteq NL \subseteq P \subseteq NP$; yet none of these inclusions is known to be strict.

The class NP can equivalently be characterized as the class of problems that can be *efficiently verified*, that is, the class of problems for which there exists a $B \in P$ and $k \in \mathbb{N}$ such that for any input x

$$x \in A \iff \exists y, |y| \leq |x|^k \colon (x, y) \in B.$$

While all problems encountered in this thesis are solvable in polynomial space, some of them do not fall into the above classes. For example, problems whose complement lies in NP. This class of problems for which the absence of solutions can be verified in polynomial time, is known as coNP.

Similarly, there exist problems that are harder to decide than problems in NP in the sense that an algorithm deciding the former could also decide any problem in NP or coNP. In order to capture the complexity of these problems, the notion of *oracle Turing machines* is helpful. An oracle Turing machine M is an ordinary deterministic or nondeterministic Turing machine with an additional query tape and three distinguished states $q_?, q_+, q_-$. The operation of M on a given input is determined relative to an arbitrary language A, the *oracle (language)*. Whenever M reaches the state $q_?$, M enters q_+ if the word on the query tape belongs to A; otherwise M enters q_-. This allows for the study of the complexity of a problem relative to a given oracle: by giving access to the oracle we essentially ignore the resources needed to decide it. This naturally leads to a hierarchy of complexity classes: For a complexity class $\mathcal{C}$, let $\mathcal{C}^A$ denote the class of problems decidable on a $\mathcal{C}$-machine with access to the oracle A, and let $\mathcal{C}^{\mathcal{D}} := \bigcup_{A \in \mathcal{D}} \mathcal{C}^A$. The *polynomial hierarchy* (PH) [MS72] is defined to consist of the classes

$$\begin{aligned} \Sigma_0^p &:= P, & \Pi_0^p &:= P, & \Delta_0^p &:= P, \\ \Sigma_{k+1}^p &:= NP^{\Sigma_k^p}, & \Pi_{k+1}^p &:= coNP^{\Sigma_k^p}, & \Delta_{k+1}^p &:= P^{\Sigma_k^p}, \end{aligned}$$

for $k \in \mathbb{N}$. Furthermore, define $PH := \bigcup_{k \in \mathbb{N}} (\Sigma_k^p \cup \Pi_k^p \cup \Delta_k^p)$.

Finally, the class $\oplus L$ is defined as the class of problems A for which there exists a nondeterministic logspace Turing machine M such that, for all x, M exhibits an odd number of accepting paths if and only if $x \in A$ [BDHM92]. It holds that $L \subseteq \oplus L \subseteq P$.

2.2.2 Circuit Complexity Classes

To define the complexity classes below L, we introduce a different model of computation, namely *Boolean circuits*. Let B be a set of Boolean functions. A

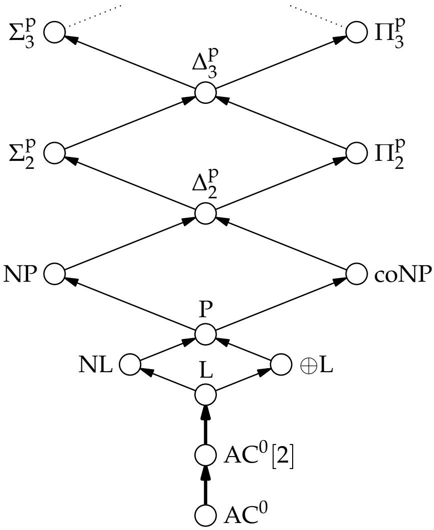

Figure 2.1: Complexity classes

Boolean circuit C over B is a finite directed acyclic graph with node labels from B and an order on the edges. The nodes of C are called *gates*. Gates of fan-out 0 are called *output gates*, gates of fan-in 0 *input gates*. A circuit with n input gates and m output gates computes a function $f_C\colon \{0,1\}^n \to \{0,1\}^m$ in the obvious way. A *family of Boolean circuits* is a sequence $\mathcal{C} = (C_n)_{n\in\mathbb{N}}$ such that C_n is a circuit with exactly n input gates. $\mathcal{C}$ is said to decide a problem A if, for all $n \in \mathbb{N}$, f_{C_n} computes the characteristic function of $A \cap \{0,1\}^n$. A circuit family is *logtime-uniform* if there exists a Turing machine working in logarithmic time that, for all $n \in \mathbb{N}$, outputs a description of C_n on inputs of length n.

The class AC^0 is defined to contain all problems decidable by logtime-uniform Boolean circuits of constant depth and polynomial size over $\{\wedge, \vee, \neg\}$, where the fan-in of gates of the first two types is not bounded. The class $AC^0[2]$ is defined similarly as AC^0, but in addition to $\{\wedge, \vee, \neg\}$ we also allow $\oplus$-gates of unbounded fan-in. It is known that $AC^0 \subsetneq AC^0[2] \subsetneq L$ [FSS84, Smo87]. For a more detailed introduction on circuit complexity, the reader is referred to [Vol99].

The inclusion structure of the classes introduced thus far is depicted in Figure 2.1, where thick arrows represent strict inclusion.

2.2.3 Counting Complexity Classes

All problems considered until now were decision problems, that is, problems for which the answer is either “yes” or “no”. However, in many contexts, one might be interested in counting the number of “yes”-instances instead. Such problems are represented using a *witness function* f, which for every input x returns a finite set $f(x)$ of witnesses. This witness function gives rise to the following *counting problem*: given an instance x, compute the cardinality $|f(x)|$ of the witness set.

The counting problems computable in polynomial time on a deterministic Turing machine are captured by the class FP. The analogue of NP is the class #P, introduced by Valiant [Val79b]. A function f is in #P if there exists a nondeterministic polynomial-time Turing machine M which, on input x, has exactly $f(x)$ accepting computation paths.

To deal with counting problems outside of #P, we follow [HV95] and define $\#\cdot\mathcal{C}$ for a class $\mathcal{C}$ of decision problems to be the class of functions f such that for some binary relation $A \in \mathcal{C}$ and some polynomial p, for all x,

$$f(x) := |\{y \mid |y| \leq p(|x|) \text{ and } A(x,y)\}|$$

In particular, we will make use of the class #·coNP, which can equivalently be characterized as $\#\cdot\mathrm{P}^{\mathrm{NP}}$ [HV95]. We hence obtain the following chain of inclusions:

$$\mathrm{FP} \subseteq \#\mathrm{P} = \#\cdot\mathrm{P} \subseteq \#\mathrm{NP} = \#\cdot\mathrm{P}^{\mathrm{NP}} = \#\cdot\mathrm{coNP}.$$

2.2.4 Reductions

Reductions are an important tool for classifying the complexity of the considered problems. The intention of reductions is to compare problems according to their computational complexity such that a problem A *reduces* to a problem B if “A is not harder to solve than B”. By imposing restrictions on the computational power of f, we obtain reducibilities suitable for comparing the complexity of arbitrary problems. One of the most prominent such is the *polynomial-time many-one reducibility*. A problem $A \subseteq \Sigma^\star$ is said to be *polynomial-time many-one reducible* to a problem $B \subseteq \Delta^\star$ (written: $A \leq^{\mathrm{p}}_{\mathrm{m}} B$) if there exists a polynomial-time computable function $f\colon \Sigma^\star \to \Delta^\star$ such that, for all $x \in \Sigma^\star$, $x \in A \iff f(x) \in B$. However, polynomial-time many-one reductions are too coarse for our purpose, because we wish to provide a fine complexity classification of the decision problems down to AC^0.

We will hence resort to *constant-depth reductions*. A problem A is said to be *constant-depth reducible* to a problem B (written: $A \leq_{\mathrm{cd}} B$) if there exists a logtime-uniform AC^0-circuit family $(C_n)_{n\in\mathbb{N}}$ with unbounded fan-in $\{\wedge, \vee, \neg\}$-gates and oracle gates for B such that for all x, $f_{C_{|x|}}(x) = 1$ if and only if $x \in A$ [CSV84]. We also write $A \equiv_{\mathrm{cd}} B$ if $A \leq_{\mathrm{cd}} B$ and $B \leq_{\mathrm{cd}} A$. It is easy to verify that all complexity classes of decision problems introduced above are closed under constant-depth reductions. Note that, unlike polynomial-time many-one reductions, constant-depth reductions are allowed to query B more than once. This is relevant as several classes close to AC^0 lack complete problems under the more restrictive AC^0 *many-one reducibility*, under which problem A reduces to problem B (written: $A \leq_{\mathrm{m}}^{\mathrm{AC}^0} B$) if there exists an AC^0-computable function f such that $x \in A \iff f(x) \in B$. Still, the reader may notice that, except for Corollary 4.1.5, all reductions between decision problems given in this thesis are AC^0 many-one reductions indeed. It is an easy exercise to show that Corollary 4.1.5 continues to hold for AC^0 many-one reductions, too.

In the context of counting problems, we require a natural generalization of the above reductions. For our results on default and autoepistemic logic, we will use *parsimonious reductions* while for circumscription we require two less restrictive notions, namely *subtractive reductions* and *weakly parsimonious reductions* (also referred to as *counting reductions*, confer [Zan91]).

Let $\#A$ and $\#B$ be the counting problems associated with the witness functions $f_A \colon \Sigma_A^\star \to \wp(\Delta_A^\star)$ and $f_B \colon \Sigma_B^\star \to \wp(\Delta_B^\star)$. A *weakly parsimonious reduction* from $\#A$ to $\#B$ consists of a pair of polynomial-time computable functions $g \colon \Sigma_A^\star \to \Sigma_B^\star$ and $h \colon \mathbb{N} \to \mathbb{N}$ such that for all $x \in \Sigma_A^\star$, $|f_A(x)| = h\big(|f_B(g(x))|\big)$. A *parsimonious reduction* is a weakly parsimonious reduction such that h is the identity. Finally, say that $\#A$ reduces to $\#B$ via strong subtractive reduction if there exists a pair of polynomial-time computable functions $g, h \colon \Sigma_A^\star \to \Sigma_B^\star$ such that, for all $x \in \Sigma_A^\star$, $f_B(g(x)) \subseteq f_B(h(x))$ and $|f_A(x)| = |f_B(h(x))| - |f_B(g(x))|$. A *subtractive reduction* from $\#A$ to $\#B$ is the transitive closure of strong subtractive reductions: $\#A$ reduces to $\#B$ via subtractive reduction if there exists an $n \in \mathbb{N}$ and a sequence $(\#A_i)_{1 \leq i \leq n}$ such that $\#A_1 = \#A$, $\#A_n = \#B$ and $\#A_i$ reduces to $\#A_{i+1}$ via strong subtractive reduction for all $1 \leq i < n$. Clearly, each parsimonious reduction is also a subtractive reduction.

While #P and #·coNP are closed under parsimonious and subtractive reductions, Toda and Wanatabe observed that this is not the case for weakly parsimonious reductions unless the counting hierarchy collapses: For every problem in #·PH there exists a weakly parsimonious reductions to a #P-complete problem [TW92]. However, we will use weakly parsimonious reductions to prove #P-hardness only, which still provides sufficient evidence that the considered problems are not contained in FP: If $\#A$ is #P-complete via weakly parsimonious reduction, then $\#A \in \mathrm{FP}$ if and only if $\mathrm{FP} = \#\mathrm{P}$.

All of the above reducibilities are both transitive and reflexive, and thus induce a preorder on problems. Problems that are maximal with respect to this preorder in a complexity class play an important role in the classification of the complexity of computational problems, as they can be regarded the “most difficult”. Given a reducibility $\leq$, we say that a problem A is *hard* (respectively *complete*) for a class $\mathcal{C}$ with respect to $\leq$-reductions if $B \leq A$ for all $B \in \mathcal{C}$ (respectively if $B \leq A$ for all $B \in \mathcal{C}$ and $A \in \mathcal{C}$). We will also write $\mathcal{C}$-hard (respectively $\mathcal{C}$-complete) if $\leq$ is clear from the context.

2.2.5 Complete Problems

We already mentioned that Cook [Coo71] was the first to show that the *satisfiability problem* for propositional formulae, SAT, is NP-complete with respect to polynomial-time many-one reductions (this was independently shown by Levin in 1973 [Lev73]). This hardness result indeed holds for the weaker constant-depth reductions as well. A slight modification of the construction used in the proof of this result can be used to show that the problem restricted to formulae in conjunctive normal form with exactly three literals per clause,

Problem: 3SAT
Input: A formula φ in conjunctive normal form with exactly three literals per clause
Question: Is φ satisfiable?

remains NP-complete with respect to constant-depth reductions. It follows that the *tautology problem* for propositional formulae in disjunctive normal form with exactly three literals per term,

Problem: 3TAUT
Input: A formula φ in disjunctive normal form with exactly three literals per term
Question: Is φ tautological?

is coNP-complete. The following theorem summarizes the discussed completeness results:

Theorem 2.2.1 ([Coo71])

1. SAT *and* 3SAT *are* NP-*complete with respect to constant-depth reductions.*
2. 3TAUT *is* coNP-*complete with respect to constant-depth reductions.*

A canonical generalization leads to complete problems for the classes Σ_k^{p} and Π_k^{p} for $k > 1$ [Wra76]: define *quantified (Boolean) formulae* as the extension of propositional formulae with the operators $\exists x \varphi(x) := \varphi(0) \vee \varphi(1)$ and $\forall x \varphi(x) := \varphi(0) \wedge \varphi(1)$, where x is a variable and φ is a quantified Boolean formula. Say that an occurrence of the variable x is *bound* if it appears in the scope of $\exists x$ or $\forall x$. A formula is *closed* if all occurrences of variables are bound. Let $Q_k := \exists$ if k is odd and $Q_k := \forall$ if k is even. Then, for $k \geq 1$, the problem

Problem: $\mathrm{QBF}_{\exists,k}$
Input: A closed quantified Boolean formula of the form $\varphi = \exists x_{11} \cdots \exists x_{1n_1} \forall x_{21} \cdots \forall x_{2n_2} \cdots Q_k x_{k1} \cdots Q_k x_{kn_k} \psi$ with ψ in conjunctive normal form if k is odd and ψ in disjunctive normal form if k is even
Question: Is φ valid?

is Σ_k^{p}-complete with respect to constant-depth reductions. Analogously, the dual problem $\textsc{Qbf}_{\forall,k}$ is Π_k^{p}-complete for all $k \geq 1$.

Theorem 2.2.2 ([Wra76]) *Let $k \geq 1$.*

1. $\textsc{Qbf}_{\exists,k}$ *is Σ_k^{p}-complete with respect to constant-depth reductions.*
2. $\textsc{Qbf}_{\forall,k}$ *is Π_k^{p}-complete with respect to constant-depth reductions.*

In the counting complexity context, the analogous complete problems are

Problem: $\#\Pi_k\textsc{Sat}$
Input: A quantified Boolean formula of the form
$\varphi = \forall x_{11} \cdots \forall x_{1n_1} \exists x_{21} \cdots \exists x_{2n_2} \cdots Q_k x_{k1} \cdots Q_k x_{kn_k} \psi$
with ψ in conjunctive normal form if k is even and
ψ in disjunctive normal form if k is odd
Output: The number of assignments satisfying φ,

where $k \geq 0$. Notice that the decision problem underlying $\#\Pi_k\textsc{Sat}$ is $\textsc{Qbf}_{\exists,k+1}$ and that its definition subsumes $\#\textsc{Sat}$, the counting problem associated with the propositional satisfiability problem.

Theorem 2.2.3 ([Val79b, DHK05]) *For all $k \geq 0$, $\#\Pi_k\textsc{Sat}$ is $\#\cdot\Pi_k^{\mathrm{p}}$-complete with respect to parsimonious reductions.*

Coming back to the polynomial hierarchy, we moreover require the *sequentially nested satisfiability problem* defined as

Problem: $\textsc{SNSat}$
Input: A sequence $(\varphi^i)_{1\leq i\leq n}$ of formulae such that φ^i contains the propositions $x_1, \ldots, x_{i-1}$ and $z_{i1}, \ldots, z_{im_i}$
Question: Is $c_n = 1$, where c_i is recursively defined via $c_i := 1$ if and only if φ^i is satisfiable by an assignment σ such that $\sigma(x_j) = c_j$ for all $1 \leq j < i$?

The problem $\textsc{SNSat}$ was incidentally identified to be Δ_2^{p}-complete in [Got95a, Theorem 3.4] (see also [LMS01]).

Theorem 2.2.4 ([Got95a]) $\textsc{SNSat}$ *is Δ_2^{p}-complete with respect to constant-depth reductions.*

For the classes NL and P, we introduce respectively the complete *directed graph accessibility problem* and the *directed hypergraph accessibility problem*, where a directed hypergraph is a hypergraph $H = (V, E)$ whose hyperedges $e \in E$ consist of a set of source nodes $\mathrm{src}(e) \subseteq V$ and a destination $\mathrm{dest}(e) \in V$. A node $t \in V$ is said to be *reachable* from a set of nodes $S \subseteq V$ in H if there exists a sequence $(S_i)_{0\leq i\leq n}$ of node sets such that $S_0 = S$, $t \in S_n$, and for all $0 \leq i < n$, $S_{i+1} = S_i \cup \{\mathrm{dest}(e)\}$ for some hyperedge $e \in E$ with $\mathrm{src}(e) \subseteq S_i$. Although both problems were originally shown to be complete with respect to logspace many-one reductions only, one can verify that this still holds for constant-depth and AC^0 many-one reductions.

Problem: GAP
Input: A directed graph $G = (V, E)$ and nodes $s, t \in V$
Question: Is there a path from s to t in G?

Theorem 2.2.5 ([Sav70, Jon75]) GAP *is* NL*-complete with respect to constant-depth reductions.*

Problem: HGAP
Input: A directed hypergraph $H = (V, E)$, a set of nodes $S \subseteq V$, and a node $t \in V$
Question: Is t reachable from S in H?

Theorem 2.2.6 ([SI90]) HGAP *is* P*-complete with respect to constant-depth reductions, even if all edges of the given hypergraph are allowed to contain at most two source nodes.*

The next problem is complete for the class $\oplus$L:

Problem: MODGAP$_2$
Input: A directed acyclic graph G with nodes s and t
Question: Is there an odd number of simple paths leading from s to t?

Theorem 2.2.7 ([BDHM92]) MODGAP$_2$ *is* $\oplus$L*-complete with respect to constant-depth reductions.*

Finally, for a string $x \in \{0,1\}^\star$, let $|x|_c$ with $c \in \{0,1\}$ denote the number of occurrences of the symbol c in x. It is easy to see that the problem

Problem: MOD$_2$
Input: $x \in \{0,1\}^\star$
Question: Is $|x|_1 \equiv 1 \pmod 2$?

is complete for the class $\mathrm{AC}^0[2]$ with respect to constant-depth reductions, for $\mathrm{AC}^0[2]$ merely extends AC^0 with oracle gates for MOD$_2$.

Theorem 2.2.8 MOD$_2$ *is* $\mathrm{AC}^0[2]$*-complete with respect to constant-depth reductions.*

2.3 Propositional Logic

Propositional logic is usually defined in terms of the functional complete set $\{\wedge, \neg\}$. As we are going to study the problems parameterized by restricted sets of Boolean functions, we give a more general definition.

Let Φ be the set of *propositions* or *variables*. Let B be a set of Boolean functions. Then the set $\mathcal{L}(B)$ of *(propositional) B-formulae* is inductively defined as follows: Each proposition $x \in \Phi$ is a B-formula. If $f \in B$ is an n-ary Boolean function and $\varphi_1, \ldots, \varphi_n$ are B-formulae, then $f(\varphi_1, \ldots, \varphi_n)$ is a B-formula. The set of *atomic B-formulae* can hence be defined as the set of propositions and nullary functions from B. If B is the set of all Boolean functions or the meaning is clear from the context, we omit the prefix "B-". Occasionally we will also refer to sets of formulae as *theories*. For a formula φ, we denote by $\mathrm{Vars}(\varphi)$ the set of propositions occurring in φ and write $\varphi(x_1, \ldots, x_n)$ to indicate that $\mathrm{Vars}(\varphi) \subseteq \{x_1, \ldots, x_n\}$. The set of subformulae of φ is denoted by $\mathrm{SF}(\varphi)$. Further, let $\varphi_{[\alpha/\beta]}$ denote φ with all occurrences of the formula α replaced by the formula β.

An *assignment* is a mapping from propositions to the Boolean constants $\{0,1\}$. For ease of notation, we will sometimes identify assignments with the set of propositions mapped to 1. For a formula φ and an assignment $\sigma\colon \mathrm{Vars}(\varphi) \to \{0,1\}$, the value of φ under σ is defined as the value of φ under the extension $\hat{\sigma}$ of σ to $\mathcal{L}$ inductively defined by

$$\hat{\sigma}(0) := 0,\ \hat{\sigma}(1) := 1, \text{ and } \hat{\sigma}(f(\varphi_1, \ldots, \varphi_n)) := f(\hat{\sigma}(\varphi_1), \ldots, \hat{\sigma}(\varphi_n)).$$

If the value of φ under σ is 1, we say that σ is a *model* of φ and write $\sigma \models \varphi$. A formula φ that possesses a model is said to be *satisfiable*. We say that two formulae φ and ψ are *equivalent* (written: $\varphi \equiv \psi$) if and only if $\hat{\sigma}(\varphi) = \hat{\sigma}(\psi)$ for all assignments $\sigma\colon \mathrm{Vars}(\varphi) \cup \mathrm{Vars}(\psi) \to \{0,1\}$. A formula φ is said to *imply* a formula ψ (written: $\varphi \models \psi$) if ψ is satisfied in all models of φ. The set of formulae implied by φ is denoted by $\mathrm{Th}(\varphi) := \{\psi \mid \varphi \models \psi\}$. The above notions are extended to sets of formulae in the obvious way. We moreover identify finite sets of formulae with their conjunction.

2.4 Clones and Post's Lattice

This thesis studies the complexity of problems parameterized by the set of allowed Boolean connectives. Here we introduce the algebraic tools to handle the infinite sets of problems arising from this parameterization.

A *clone* is a set of Boolean functions, which is closed under superposition, that is, B contains all projections (the functions $f(x_1, \ldots, x_n) = x_k$ for $1 \leq k \leq n$) and is closed under composition (for all $f \in B$ of arity n and $g_1, \ldots, g_n \in B$, $f\big(g_1(x_1, \ldots, x_{m_1}), \ldots, g_n(x_1, \ldots, x_{m_n})\big) \in B$) [Pip97]. For a set B of Boolean functions, we denote with $[B]$ the smallest clone containing B and call B a *base* for $[B]$. In [Pos41], Post showed that the set of all clones ordered by inclusion together with the operations $[B \cup B']$ and $[B \cap B']$ forms a lattice and found a finite base for each clone, see Figure 2.2.

To introduce the clones, we define the following properties. Say that a set $A \subseteq \{0,1\}^n$ is *c-separating*, $c \in \{0,1\}$, if there exists an $i \in \{1, \ldots, n\}$ such that $(a_1, \ldots, a_n) \in A$ implies $a_i = c$. Let f be an n-ary Boolean function and define the dual of f to be the Boolean function $\mathrm{dual}(f)(x_1, \ldots, x_n) := \neg f(\neg x_1, \ldots, \neg x_n)$. We say that

- f is *c-reproducing* if $f(c, \ldots, c) = c$, $c \in \{0,1\}$.
- f is *c-separating* if $f^{-1}(c)$ is c-separating, $c \in \{0,1\}$.

- f is *c-separating of degree m* if all $A \subseteq f^{-1}(c)$ with $|A| = m$ are c-separating.
- f is *monotone* if $a_1 \leq b_1, \ldots, a_n \leq b_n$ implies $f(a_1, \ldots, a_n) \leq f(b_1, \ldots, b_n)$.
- f is *self-dual* if $f \equiv \mathrm{dual}(f)$.
- f is *affine* if $f(x_1, \ldots, x_n) \equiv x_1 \oplus \cdots \oplus x_n \oplus c$ with $c \in \{0, 1\}$.
- f is *essentially unary* if f depends on at most one variable.

Finally, say that B is *functional complete* if $[B] = \mathsf{BF}$. The list of all clones is given in Table 2.1, where id denotes the identity and t_n^{n+1} denotes the $(n+1)$-ary threshold function that evaluates to 1 if at least n of its inputs are set to 1:

$$\mathrm{t}_n^{n+1}(x_0, \ldots, x_{n+1}) := \bigvee_{i=0}^{n} (x_0 \wedge \cdots \wedge x_{i-1} \wedge x_{i+1} \wedge \cdots \wedge x_{n+1}).$$

To see how restrictions on the set of Boolean connectives affect the complexity of problems, let $\Pi(B)$ be a nontrivial computational problem defined over B-formulae (that is, a set of B-formulae). One would certainly expect the complexity of $\Pi(B)$ to drop, the less expressive B is. As an example, recall that Lewis showed that the satisfiability problem

Problem: $\textsc{Sat}(B)$
Input: A propositional B-formula $\varphi \in \mathcal{L}(B)$
Question: Is φ satisfiable?

is NP-complete if $x \nrightarrow y \in [B]$, and contained in P otherwise [Lew79]. The proof of his result uses two important properties:

- The complexity of $\textsc{Sat}(B)$ depends only on the clone $[B]$ and not on the particular B itself.
- For sets B, B' of Boolean functions with $B \subseteq [B']$, we have $\textsc{Sat}(B) \leq \textsc{Sat}(B')$, where $\leq$ is any reducibility capable of substituting B-functions by B'-equivalents.

These properties, which facilitate a complete classification of all possible sets of Boolean functions, hinge on the existence of small representations regardless of the given base of a clone. To be more precise, let f be an n-ary Boolean function and B be a set of Boolean functions. A B-formula g is called *B-representation* of f if $f \equiv g$. It is clear that B-representations exist for every $f \in [B]$. Yet, it may happen that the B-representation of a Boolean function contains some input variables more than once.

Example 2.4.1 *Let $g(x, y) := \neg(x \wedge y)$. The shortest $\{g\}$-representation of the function $x \wedge y$ is $g(g(x, y), g(x, y))$. Expressing $x_1 \wedge \cdots \wedge x_n$ using g leads to an explosion of the formula size of the $\{g\}$-representation of f—its size is exponential in n.*

Clone	Definition	Base
BF	All Boolean functions	$\{x \wedge y, \neg x\}$
R_0	$\{f \in \mathsf{BF} \mid f$ is 0-reproducing$\}$	$\{x \wedge y, x \oplus y\}$
R_1	$\{f \in \mathsf{BF} \mid f$ is 1-reproducing$\}$	$\{x \vee y, x \leftrightarrow y\}$
R_2	$\mathsf{R}_0 \cap \mathsf{R}_1$	$\{x \vee y, x \wedge (y \leftrightarrow z)\}$
M	$\{f \in \mathsf{BF} \mid f$ is monotone$\}$	$\{x \wedge y, x \vee y, 0, 1\}$
M_0	$\mathsf{M} \cap \mathsf{R}_0$	$\{x \wedge y, x \vee y, 0\}$
M_1	$\mathsf{M} \cap \mathsf{R}_1$	$\{x \wedge y, x \vee y, 1\}$
M_2	$\mathsf{M} \cap \mathsf{R}_2$	$\{x \wedge y, x \vee y\}$
S_0	$\{f \in \mathsf{BF} \mid f$ is 0-separating$\}$	$\{x \rightarrow y\}$
S_0^n	$\{f \in \mathsf{BF} \mid f$ is 0-separating of degree $n\}$	$\{x \rightarrow y, \mathrm{dual}(\mathrm{t}_n^{n+1})\}$
S_1	$\{f \in \mathsf{BF} \mid f$ is 1-separating$\}$	$\{x \nrightarrow y\}$
S_1^n	$\{f \in \mathsf{BF} \mid f$ is 1-separating of degree $n\}$	$\{x \nrightarrow y, \mathrm{t}_n^{n+1}\}$
S_{02}^n	$\mathsf{S}_0^n \cap \mathsf{R}_2$	$\{x \vee (y \wedge \neg z), \mathrm{dual}(\mathrm{t}_n^{n+1})\}$
S_{02}	$\mathsf{S}_0 \cap \mathsf{R}_2$	$\{x \vee (y \wedge \neg z)\}$
S_{01}^n	$\mathsf{S}_0^n \cap \mathsf{M}$	$\{\mathrm{dual}(\mathrm{t}_n^{n+1}), 1\}$
S_{01}	$\mathsf{S}_0 \cap \mathsf{M}$	$\{x \vee (y \wedge z), 1\}$
S_{00}^n	$\mathsf{S}_0^n \cap \mathsf{R}_2 \cap \mathsf{M}$	$\{x \vee (y \wedge z), \mathrm{dual}(\mathrm{t}_n^{n+1})\}$
S_{00}	$\mathsf{S}_0 \cap \mathsf{R}_2 \cap \mathsf{M}$	$\{x \vee (y \wedge z)\}$
S_{12}^n	$\mathsf{S}_1^n \cap \mathsf{R}_2$	$\{x \wedge (y \vee \neg z), \mathrm{t}_n^{n+1}\}$
S_{12}	$\mathsf{S}_1 \cap \mathsf{R}_2$	$\{x \wedge (y \vee \neg z)\}$
S_{11}^n	$\mathsf{S}_1^n \cap \mathsf{M}$	$\{\mathrm{t}_n^{n+1}, 0\}$
S_{11}	$\mathsf{S}_1 \cap \mathsf{M}$	$\{x \wedge (y \vee z), 0\}$
S_{10}^n	$\mathsf{S}_1^n \cap \mathsf{R}_2 \cap \mathsf{M}$	$\{x \wedge (y \vee z), \mathrm{t}_n^{n+1}\}$
S_{10}	$\mathsf{S}_1 \cap \mathsf{R}_2 \cap \mathsf{M}$	$\{x \wedge (y \vee z)\}$
D	$\{f \in \mathsf{BF} \mid f$ is self-dual$\}$	$\{(x \wedge y) \vee (x \wedge \neg z) \vee (\neg y \wedge \neg z)\}$
D_1	$\mathsf{D} \cap \mathsf{R}_2$	$\{(x \wedge y) \vee (x \wedge \neg z) \vee (y \wedge \neg z)\}$
D_2	$\mathsf{D} \cap \mathsf{M}$	$\{(x \wedge y) \vee (x \wedge z) \vee (y \wedge z)\}$
L	$\{f \in \mathsf{BF} \mid f$ is affine$\}$	$\{x \oplus y, 1\}$
L_0	$\mathsf{L} \cap \mathsf{R}_0$	$\{x \oplus y\}$
L_1	$\mathsf{L} \cap \mathsf{R}_1$	$\{x \leftrightarrow y\}$
L_2	$\mathsf{L} \cap \mathsf{R}_2$	$\{x \oplus y \oplus z\}$
L_3	$\mathsf{L} \cap \mathsf{D}$	$\{x \oplus y \oplus z \oplus 1\}$
E	$\{f \in \mathsf{BF} \mid f$ is constant or a conjunction$\}$	$\{x \wedge y, 0, 1\}$
E_0	$\mathsf{E} \cap \mathsf{R}_0$	$\{x \wedge y, 0\}$
E_1	$\mathsf{E} \cap \mathsf{R}_1$	$\{x \wedge y, 1\}$
E_2	$\mathsf{E} \cap \mathsf{R}_2$	$\{x \wedge y\}$
V	$\{f \in \mathsf{BF} \mid f$ is constant or a disjunction$\}$	$\{x \vee y, 0, 1\}$
V_0	$\mathsf{V} \cap \mathsf{R}_0$	$\{x \vee y, 0\}$
V_1	$\mathsf{V} \cap \mathsf{R}_1$	$\{x \vee y, 1\}$
V_2	$\mathsf{V} \cap \mathsf{R}_2$	$\{x \vee y\}$
N	$\{f \in \mathsf{BF} \mid f$ is essentially unary$\}$	$\{\neg x, 0, 1\}$
N_2	$\mathsf{N} \cap \mathsf{D}$	$\{\neg x\}$
I	$\{f \in \mathsf{BF} \mid f$ is constant or a projection$\}$	$\{\mathrm{id}, 0, 1\}$
I_0	$\mathsf{I} \cap \mathsf{R}_0$	$\{\mathrm{id}, 0\}$
I_1	$\mathsf{I} \cap \mathsf{R}_1$	$\{\mathrm{id}, 1\}$
I_2	$\mathsf{I} \cap \mathsf{R}_2$	$\{\mathrm{id}\}$

Table 2.1: List of all clones with definition and bases

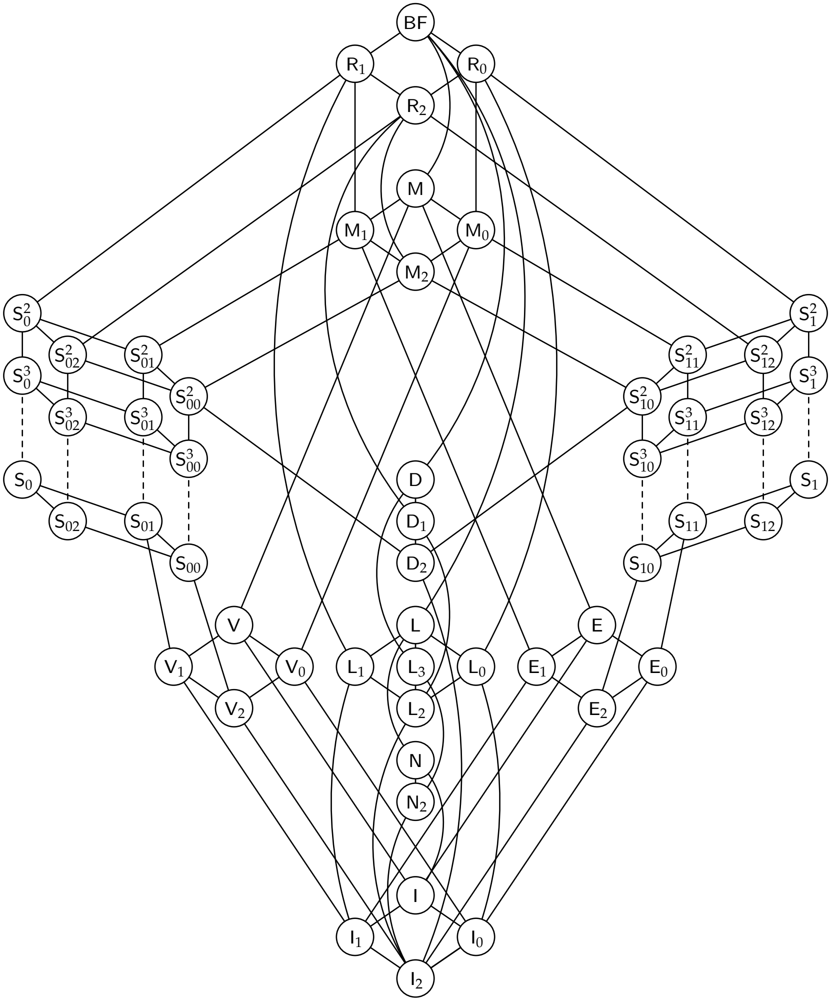

Figure 2.2: Post's lattice

To circumvent this problem, we introduce the notion of efficient implementation. Say that B *efficiently implements* f if there exists a B-formula g such that $f \equiv g$ and each variable in f occurs at most once in the body of g. Given that all $f \in B$ can be efficiently implemented in B', we are able to conclude that $\textsc{Sat}(B) \leq \textsc{Sat}(B')$. The following lemma summarizes the results on efficient implementation required in the subsequent chapters.

Lemma 2.4.2 *Let B be a finite set of Boolean functions.*

1. *If $[B] = \mathsf{BF}$ then B efficiently implements $\{\wedge, \vee, \neg\}$.*
2. *If $\mathsf{M} \subseteq [B]$ then B efficiently implements $\{\wedge, \vee\}$.*
3. *If $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$ then B efficiently implements $x \oplus y \oplus z$.*
4. *If $\mathsf{N} \subseteq [B]$ then B efficiently implements $\neg$.*

Proof. The first and fourth claim are due to Lewis [Lew79]. Although he only proves the efficient implementation of $\neg$ for $[B] = \mathsf{BF}$, it is easy to verify that he merely requires $\neg \in [B]$ and the Boolean constants $\{0, 1\}$.

The second claim is due to [Sch10].

For the third claim, we have to show that $x \oplus y \oplus z$ can be efficiently implemented in any set B such that $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$. Let B be such that $\mathsf{L}_2 \subseteq [B]$ and let $g(x, y, z)$ be a function from $[B]$ depending on three variables. Such a function g exists because $x \oplus y \oplus z \in [B]$. As g is affine, replacing two occurrences of any variable with a fresh variable t does not change g modulo logical equivalence. Let n denote the number of occurrences of x in g and assume that n is even. Replacing all occurrences of x with t yields a formula $g'(y, z, t) \equiv y \oplus z \notin \mathsf{L}_2$, which gives a contradiction. Analogous arguments hold for the number of occurrences of y and z. Hence, each of the variables x, y, and z occurs an odd number of times, and replacing all but one occurrence of each x, y, and z with t yields a function $g'(x, y, z, t) \equiv x \oplus y \oplus z$ in which each of the variables x, y, and z occurs exactly once. $\square$

2.5 The Complexity of Implication

The last section of this chapter is dedicated to the complexity of the implication problem for B-formulae. This problem is of fundamental importance for the results in this thesis, as the implication of B-formulae has to be tested in various contexts ranging from extension existence to credulous and skeptical reasoning.

Hence, let B be a finite set of Boolean functions. We define the *implication problem* for B-formulae as

Problem:	$\textsc{Imp}(B)$
Input:	A finite set Γ of B-formulae and a B-formula φ
Question:	Does $\Gamma \models \varphi$ hold?

The following theorem classifies the complexity of the implication problem for all possible sets B.

Theorem 2.5.1 *Let B be a finite set of Boolean functions. Then the implication problem for propositional B-formulae, $\text{IMP}(B)$, is*

1. coNP-*complete if* $\mathsf{S}_{00} \subseteq [B]$ *or* $\mathsf{S}_{10} \subseteq [B]$ *or* $\mathsf{D}_2 \subseteq [B]$,
2. $\oplus$L-*complete if* $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$,
3. $\mathrm{AC}^0[2]$-*complete if* $\mathsf{N}_2 \subseteq [B] \subseteq \mathsf{N}$, *and*
4. *in* AC^0 *in all other cases,*

with respect to constant-depth reductions.

Remark 2.5.2 *In [BMTV09a] it is shown that the complexity of* $\text{IMP}(B)$ *restricted to instances with* $|\Gamma| = 1$ *remains unchanged for all B except the case that* $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$. *In this case, the complexity drops to* $\mathrm{AC}^0[2]$*-completeness.*

We split the proof of Theorem 2.5.1 into several lemmas.

Lemma 2.5.3 *Let B be a finite set of Boolean functions such that* $\mathsf{S}_{00} \subseteq [B]$ *or* $\mathsf{S}_{10} \subseteq [B]$. *Then* $\text{IMP}(B)$ *is* coNP*-complete with respect to constant-depth reductions.*

Proof. Membership in coNP is apparent, because given Γ and φ, we just have to check that for all assignments σ to the variables of Γ and φ, either $\sigma \not\models \Gamma$ or $\sigma \models \varphi$.

The hardness proof is inspired by [Rei03]. Observe that $\text{IMP}(B) \equiv_{\mathrm{cd}} \text{IMP}(B \cup \{1\})$ if $\wedge \in [B]$, and that $\text{IMP}(B) \equiv_{\mathrm{cd}} \text{IMP}(B \cup \{0\})$ if $\vee \in [B]$ (because $\varphi \models \psi \iff \varphi_{[1/t]} \wedge t \models \psi_{[1/t]}$ and $\varphi \models \psi \iff \varphi_{[0/f]} \models \psi_{[0/f]} \vee f$, where t, f are new variables). It hence suffices to show that $\text{IMP}(B)$ is coNP-hard for M, because $[\mathsf{S}_{00} \cup \{0\}] = \mathsf{M}_0$, $[\mathsf{S}_{10} \cup \{1\}] = \mathsf{M}_1$ and $[\mathsf{M}_0 \cup \{1\}] = [\mathsf{M}_1 \cup \{0\}] = \mathsf{M}$. We claim that $\text{IMP}(B)$ is coNP-hard for $B = \{\wedge, \vee\}$. The proof concludes by appealing to Lemma 2.4.2 (2.). To prove the claim, we will provide a reduction from 3TAUT.

Let φ be a propositional formula in disjunctive normal form over the propositions $X = \{x_1, \ldots, x_k\}$. Then $\varphi = \bigvee_{i=1}^{n} \bigwedge_{j=1}^{3} \ell_{ij}$, where ℓ_{ij} are literals over X. Take new variables $Y = \{y_1, \ldots, y_k\}$ and replace in φ each negative literal $\ell_{ij} = \neg x_l$ by y_l. Define the resulting formula as ψ_2 and let $\psi_1 := \bigwedge_{i=1}^{k} (x_i \vee y_i)$. We claim that $\varphi \in \text{3TAUT} \iff \psi_1 \models \psi_2$.

Let us first assume $\varphi \in \text{3TAUT}$ and let $\sigma \colon X \cup Y \to \{0, 1\}$ be an assignment such that $\sigma \models \psi_1$. As φ is a tautology, $\sigma \models \varphi$. But also $\sigma \models \psi_2$, as we simply replaced the negated variables in φ by positive ones and ψ_2 is monotone. It follows that $\psi_1 \models \psi_2$, since σ was arbitrarily chosen for $\sigma \models \psi_1$.

For the opposite direction, let $\varphi \notin \text{3TAUT}$. Then there exists an assignment $\sigma \colon X \to \{0, 1\}$ such that $\sigma \not\models \varphi$. We extend σ to an assignment $\sigma' \colon X \cup Y \to \{0, 1\}$ by setting $\sigma'(y_i) = 1 - \sigma(x_i)$ for $i = 1, \ldots, k$. Then $\sigma'(x_i) = 0$ if and only if $\sigma'(y_i) = 1$, and consequently σ' simulates σ on ψ_2. As a result, $\sigma' \not\models \psi_2$. Yet, either $\sigma'(x_i) = 1$ or $\sigma'(y_i) = 1$ for $i = 1, \ldots, k$. Thus $\sigma' \models \psi_1$, yielding $\psi_1 \not\models \psi_2$. $\square$

Lemma 2.5.4 *Let B be a finite set of Boolean functions such that $\mathsf{D}_2 \subseteq [B]$. Then* $\textsc{Imp}(B)$ *is* coNP*-complete with respect to constant-depth reductions.*

Proof. Again we just have to argue for coNP-hardness of $\textsc{Imp}(B)$. We give a reduction from $3\textsc{Taut}$ to $\textsc{Imp}(B)$ for $\mathsf{D}_2 \subseteq [B]$ by modifying the reduction given in the proof of Lemma 2.5.3.

Given a formula φ in disjunctive normal form, we define the formulae ψ_1 and ψ_2 as above. As $\mathsf{D}_2 \subseteq [B]$, we know that $g(x,y,z) := (x \wedge y) \vee (y \wedge z) \vee (x \wedge z) \in [B]$. Clearly, $g(x,y,0) \equiv x \wedge y$ and $g(x,y,1) \equiv x \vee y$. Denote by $\psi_i^B(t,f)$, $i \in \{1,2\}$, the formula ψ_i with all occurrences of $x \wedge y$ and $x \vee y$ replaced by a B-representation of $g(x,y,f)$ and $g(x,y,t)$, respectively, where t and f are new propositional variables. Then $\psi_i^B(1,0) \equiv \psi_i$ and $\psi_i^B(0,1) \equiv \mathrm{dual}(\psi_i)$. The variables x and y may occur several times in the B-representation of g, hence $\psi_1^B(t,f)$ and $\psi_2^B(t,f)$ might be exponential in the length of φ (recall that ψ_2 is φ with all negative literals replaced by new variables). That this is not the case follows from the associativity of $\wedge$ and $\vee$: we insert parentheses in such a way that ψ_i is transformed into a binary tree of logarithmic depth; the size of ψ_i^B is exponential in the depth of this tree and therefore polynomial.

We now map the pair (ψ_1, ψ_2) to (ψ_1', ψ_2'), where

$$\psi_1' := g(\psi_1^B(t,f), t, f) \text{ and } \psi_2' := g(g(\psi_1^B(t,f), \psi_2^B(t,f), f), t, f).$$

We claim that $\psi_1 \models \psi_2 \iff \psi_1' \models \psi_2'$. To verify this claim, let σ be an arbitrary assignment for $X \cup Y$. Then σ may be extended to $\{t,f\}$ in the following ways:

$\sigma(t) = 1$ and $\sigma(f) = 0$: This is the intended interpretation. In this case, we have $g(\psi_1^B(1,0),1,0) \equiv \psi_1 \wedge 1 \equiv \psi_1$ and $g(g(\psi_1^B(1,0), \psi_2^B(1,0),0),1,0) \equiv (\psi_1 \wedge \psi_2) \wedge 1 \equiv \psi_1 \wedge \psi_2$. Hence, $\psi_1' \models \psi_2'$ if and only if $\psi_1 \models \psi_1 \wedge \psi_2$.

$\sigma(t) = 0$ and $\sigma(f) = 1$: In this case, we obtain $g(\psi_1^B(0,1),0,1) \equiv \mathrm{dual}(\psi_1) \vee 0 \equiv \mathrm{dual}(\psi_1)$ and $g(g(\psi_1^B(0,1), \psi_2^B(0,1),1),0,1) \equiv (\mathrm{dual}(\psi_1) \vee \mathrm{dual}(\psi_2)) \vee 0 \equiv \mathrm{dual}(\psi_1) \vee \mathrm{dual}(\psi_2)$. As $\mathrm{dual}(\psi_1) \models \mathrm{dual}(\psi_1) \vee \mathrm{dual}(\psi_2)$ is always valid, we conclude that $\psi_1' \models \psi_2'$ in this case.

$\sigma(t) = \sigma(f) = c$ with $c \in \{0,1\}$: Then both ψ_1' and ψ_2' are equivalent to c. Thus, as in the previous case, $\psi_1' \models \psi_2'$.

From the above case distinction, it follows that $\psi_1 \models \psi_2$ if and only if $\psi_1' \models \psi_2'$. Hence, $3\textsc{Taut} \leq_{\mathrm{cd}} \textsc{Imp}(B)$ via the reduction $\varphi \mapsto (\psi_1', \psi_2')$. □

Lemma 2.5.5 *Let B be a finite set of Boolean functions such that $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$. Then* $\textsc{Imp}(B)$ *is* $\oplus$L*-complete with respect to constant-depth reductions.*

Proof. Let B be a finite set of Boolean functions such that $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$, let Γ be a set of B-formulae over $\mathrm{Vars}(\Gamma) = \{x_1, \ldots, x_n\}$, and let φ be a B-formula. Observe that $\Gamma \models \varphi$ if and only if $\Gamma \cup \{\varphi \oplus t, t\}$ is inconsistent, where t is a fresh variable. Let Γ' denote $\Gamma \cup \{\varphi \oplus t, t\}$ rewritten such that for all $\psi \in \Gamma'$, $\psi = c_0 \oplus c_1x_1 \oplus \cdots \oplus c_nx_n$, where $c_0, \ldots, c_n \in \{0,1\}$ and c_ix_i is a shorthand for $(c_i \wedge x_i)$. Γ' is logspace constructible, since $c_0 = 1$ if and only if $\psi(0,\ldots,0) = 1$, for $1 \leq i \leq n$, $c_i = 1$ if and only if

$$\psi(0,\ldots,0) \not\equiv \psi(\underbrace{0,\ldots,0}_{i-1},1,0,\ldots,0),$$

and affine formulae can be evaluated in logarithmic space [Sch10]. Γ' can now be transformed into a system of linear equations S via

$$c_0 \oplus c_1x_1 \oplus \cdots \oplus c_nx_n \mapsto c_0 + c_1x_1 + \cdots + c_nx_n = 1 \pmod 2.$$

Clearly, the resulting system of linear equations has a solution if and only if Γ' is consistent. The equations are furthermore defined over the field $\mathbb{Z}_2$, hence existence of a solution can be decided in $\oplus$L, as shown in [BDHM92].

For the $\oplus$L-hardness, Buntrock *et al.* [BDHM92] give an NC^1-reduction from $\textsc{ModGap}_2$ to the problem whether a given matrix over $\mathbb{Z}_2$ is non-singular. The given reduction is actually an AC^0 many-one reduction. We reduce the latter problem to the complement of $\textsc{Imp}(\{x \oplus y \oplus z\})$. The lower bound then follows from $\oplus$L being closed under complement and Lemma 2.4.2 (3.).

First map the given matrix $A = (a_{ij})_{1\leq i,j\leq n}$ over $\mathbb{Z}_2$ to a system of linear equations defined as

$$S := \{a_{i1}x_1 + \cdots + a_{in}x_n = 0 \mid 1 \leq i \leq n\} \cup \{x_1 = 1\}.$$

It clearly holds that A is non-singular if and only if S has no solutions. Next, map S into a set of affine formulae Γ via

$$c_1x_1 + \cdots + c_nx_n = c \pmod 2 \;\mapsto\; c' \oplus c_1x_1 \oplus \cdots \oplus c_nx_n,$$

where $c' = 1 - c$. Finally, replace the constant 1 with a fresh variable t, pad all formulae having an even number of variables with another fresh variable f, and let $\Gamma' := \Gamma \cup \{t\}$. We claim that S has a solution if and only if $\Gamma' \not\models f$.

Suppose that S has no solutions. If Γ' is inconsistent, then $\Gamma' \models f$. Otherwise, Γ' has a satisfying assignment σ. Clearly, $\sigma(t) = 1$. If $\sigma(f) = 0$, then $\Gamma'_{[t/1,f/0]}$ is equivalent to Γ; hence the transformation of $\Gamma'_{[t/1,f/0]}$ yields a system of linear equations S' that is equivalent to S and that has a solution corresponding to σ—a contradiction to our assumption. Thus $\sigma(f) = 1$ and, consequently, $\Gamma' \models f$.

On the other hand, if S has a solution, then Γ possesses a model σ that can be extended to a model σ' of Γ' by setting $\sigma'(t) = 1$ and $\sigma'(f) = 0$. Concluding, $\Gamma' \not\models f$. $\square$

Lemma 2.5.6 *Let B be a finite set of Boolean functions such that $\mathsf{N}_2 \subseteq [B] \subseteq \mathsf{N}$. Then $\textsc{Imp}(B)$ is $\mathrm{AC}^0[2]$-complete with respect to constant-depth reductions.*

Proof. Let B be a finite set of Boolean functions such that $\mathsf{N}_2 \subseteq [B] \subseteq \mathsf{N}$. Let Γ be a set of B-formulae and let φ be a B-formula, both over the set of propositions $\{x_1, \ldots, x_n\}$.

We will argue on membership in $\mathrm{AC}^0[2]$ first. For all $\psi \in \Gamma$, ψ is equivalent to some literal or a constant. Let Γ' denote this set of literals and constants. Γ' is computable from Γ using an AC^0-circuit with oracle gates for MOD_2: for each formula in Γ, we determine the atom and count the number of preceding negations modulo 2. In the case that Γ is unsatisfiable, then either $0 \in \Gamma'$ or there exist $\ell_1, \ell_2 \in \Gamma'$ with $\ell_1 \equiv \neg\ell_2$. Both conditions can be checked in AC^0, hence we may without loss of generality assume that Γ is satisfiable. Similarly, φ is either equivalent to a literal or a constant. In the former case, it holds that

$$\Gamma \models \varphi \iff \varphi \equiv \ell \text{ for some } \ell \in \Gamma'.$$

In the latter case, $\Gamma \models \varphi$ if and only if $\Gamma \equiv 1$ and $\varphi \equiv 1$. It is easy to see that both of these conditions can again be checked in $\mathrm{AC}^0[2]$, as $\Gamma \equiv 1$ if and only if $\Gamma' = \{1\}$. Thus we conclude $\mathrm{IMP}(B) \in \mathrm{AC}^0[2]$.

For $\mathrm{MOD}_2 \leq_{\mathrm{cd}} \mathrm{IMP}(B)$, we claim that, for $x = x_1 \cdots x_n \in \{0,1\}^n$, $x \in \mathrm{MOD}_2$ if and only if $t \models \neg^{x_1}\neg^{x_2} \cdots \neg^{x_n}(\neg t)$, where $\neg^1 := \neg$, $\neg^0 := \mathrm{id}$, and t is a variable.

First observe that $t \models \neg^{x_1} \cdots \neg^{x_n}(\neg t)$ if and only if $\sigma \models t$ implies $\sigma \models \neg^{x_1} \cdots \neg^{x_n}(\neg t)$ for all assignments $\sigma\colon t \to \{0,1\}$. Now, if $\sigma(t) = 0$ then $t \models \neg^{x_1} \cdots \neg^{x_n}(\neg t)$ is satisfied for all x; whereas if $\sigma(t) = 1$ then $t \models \neg^{x_1} \cdots \neg^{x_n}(\neg t)$ if and only if $1 \models \neg^{x_1} \cdots \neg^{x_n} 0$ if and only if an odd number of x_i's is equal to 1. Summarizing, $\mathrm{MOD}_2 \leq_{\mathrm{cd}} \mathrm{IMP}(B)$. The claim follows from Lemma 2.4.2 (4.). □

Lemma 2.5.7 *Let B be a finite set of Boolean functions such that $[B] \subseteq \mathsf{V}$ or $[B] \subseteq \mathsf{E}$. Then $\mathrm{IMP}(B)$ is in AC^0.*

Proof. We prove the claim for $[B] \subseteq \mathsf{V}$ only. The case $[B] \subseteq \mathsf{E}$ follows analogously.

Let B be a finite set of Boolean functions such that $[B] \subseteq \mathsf{V}$. Further, let Γ be a finite set of B-formulae and let φ be a B-formula, both over the set of propositions $\{x_1, \ldots, x_n\}$. Then, $\varphi \equiv c_0 \vee c_1x_1 \vee \cdots \vee c_nx_n$ for some $c_0, \ldots, c_n \in \{0,1\}$, where c_ix_i abbreviates $(c_i \wedge x_i)$. As B-formulae can be evaluated in AC^0 by guessing a position and verifying that it contains the constant 1 or an input value that is set to 1 [Sch10], the values of the coefficients can be determined: $c_0 = 1$ if and only if $\varphi(0, \ldots, 0)$ evaluates to 1, and $c_i = 0$ if and only if $c_0 = 0$ and

$$\varphi(\underbrace{0, \ldots, 0}_{i-1}, 1, 0, \ldots, 0) = 0.$$

Equally, every formula from Γ is equivalent to an expression of the form $c'_0 \vee c'_1x_1 \vee \cdots \vee c'_nx_n$ with $c'_i \in \{0,1\}$, whose coefficients can be computed analogously to those of φ. It now holds that $\Gamma \models \varphi$ if and only if either $c_0 = 1$ or there exists a formula $\psi \equiv c''_0 \vee c''_1x_1 \vee \cdots \vee c''_nx_n$ from Γ such that $c''_i \leq c_i$ for all $0 \leq i \leq n$. Thus, $\mathrm{IMP}(B)$ can be computed in constant depth using oracle gates for B-formula evaluation. □

CHAPTER 3

NONMONOTONIC LOGICS

This chapter defines the nonmonotonic logics default logic, autoepistemic logic, and circumscription, and states the relevant results from the literature. To preserve consistency we will largely stick to the naming conventions from the respective logics. We begin with Reiter's default logic.

3.1 DEFAULT LOGIC

Among the formalisms that introduce common sense into formal logic, Reiter's default logic [Rei80] is one of the best known and most successful formalisms for modelling of common-sense reasoning. Default logic extends classical logical (first-order or propositional) derivations by patterns for default assumptions. These are of the form "in the absence of contrary information, assume ..." and seem to be very well-suited for the representation of a world, in which "things that are commonly true" outnumber "absolute thruths" [Bes89]. Reiter argued that his logic is an adequate formalization of the human reasoning under the *closed world assumption*, which allows one to assume the negation of facts not derivable from the knowledge base. In fact, nowadays default logic is widely used in artificial intelligence and computational logic.

Definition 3.1.1 (Default rules and default theories) *Fix some finite set B of Boolean functions and let α, β, γ be propositional B-formulae. A B-*default (rule) *is an expression $d = \frac{\alpha:\beta}{\gamma}$; α is called* prerequisite, *β is called* justification, *and γ is called* consequent *of d. A B-*default theory *is a pair (W, D), where W is a set of propositional B-formulae and D is a set of B-default rules.*

If $[B] = \mathsf{BF}$ or the meaning is clear from the context, we omit the prefix "B-".

What makes default logic presumably harder than propositional logic is the fact that the semantics (that is, the set of consequences) of a given set of premises are defined in terms of a fixed-point equation. The different fixed points correspond to different views of the world, based on the given premises. Formally, these are defined by means of an operator which derives all possible consequences of a given set of formulae both with the help of default rules and in the classical sense.

Definition 3.1.2 (Stable extensions) *For a given B-default theory (W, D) and a set E of formulae, let $\Gamma(E)$ be the smallest set of formulae such that*

1. *$W \subseteq \Gamma(E)$,*
2. *$\Gamma(E)$ is deductively closed, that is, $\Gamma(E) = \mathrm{Th}(\Gamma(E))$, and*
3. *for all defaults $\frac{\alpha:\beta}{\gamma} \in D$ with $\alpha \in \Gamma(E)$ and $\neg\beta \notin E$, it holds that $\gamma \in \Gamma(E)$ (in this case, we also say that the default $\frac{\alpha:\beta}{\gamma}$ is* applicable*).*

A (stable) extension *of (W, D) is a fixed point of Γ, that is, a set E such that $E = \Gamma(E)$.*

The following theorem by Reiter provides a finite characterization of stable extensions.

Theorem 3.1.3 ([Rei80]) *Let (W, D) be a B-default theory and E be a set of formulae.*

1. *Let $E_0 := W$ and $E_{i+1} := \mathrm{Th}(E_i) \cup \left\{\gamma \,\middle|\, \frac{\alpha:\beta}{\gamma} \in D, \alpha \in E_i \text{ and } \neg\beta \notin E\right\}$. Then E is a stable extension of (W, D) if and only if $E = \bigcup_{i\in\mathbb{N}} E_i$.*
2. *Let $\mathrm{GD}(E) := \left\{\frac{\alpha:\beta}{\gamma} \in D \,\middle|\, \alpha \in E \text{ and } \neg\beta \notin E\right\}$. If E is a stable extension of (W, D), then*
$$E = \mathrm{Th}\Big(W \cup \left\{\gamma \,\middle|\, \tfrac{\alpha:\beta}{\gamma} \in \mathrm{GD}(E)\right\}\Big).$$
In this case, $\mathrm{GD}(E)$ is also called the set of generating defaults *for E.*

Note that stable extensions need not be consistent. However, the following proposition shows that this only occurs if the set W is inconsistent.

Proposition 3.1.4 ([MT93]) *Let (W, D) be a B-default theory. Then $\mathcal{L}$ is a stable extension of (W, D) if and only if W is inconsistent.*

As a consequence we obtain:

Corollary 3.1.5 *Let (W, D) be a B-default theory.*

- *If W is consistent, then every stable extension of (W, D) is consistent.*
- *If W is inconsistent, then (W, D) has a stable extension.*

Now that we have defined the notion of a stable extension, we are ready to demonstrate the nonmonotonic behaviour of default logic in an example.

Example 3.1.6 *The default theory $(\emptyset, D)$ with $D := \{\frac{1:x}{\neg x}\}$ has no stable extension. On the other hand, $(\emptyset, D)$ with $D := \{\frac{1:x}{x}, \frac{1:\neg x}{\neg x}\}$ has two stable extensions, namely $E' := \mathrm{Th}(x)$ and $E'' := \mathrm{Th}(\neg x)$, corresponding to applications of respectively the first or second default in D.*

Next, consider (W, D) with $W := \{x\}$, $D := \{\frac{x:\neg y}{z}\}$. Then (W, D) has a unique stable extension which contains z. However, if W is extended by the proposition y, then the unique stable extension of $(W \cup \{y\}, D)$ does no longer contain z; the newly added fact y makes the justification of $\frac{x:\neg y}{z}$ inconsistent with its stable extension.

As seen in Example 3.1.6 above, default theories may posses one stable extension, multiple stable extensions, or not allow for stable extensions at all. Hence, the *extension existence problem* for default logic arises naturally.

Problem: $\textsc{Ext}(B)$
Input: A B-default theory (W, D)
Question: Does (W, D) have a stable extension?

Informally, multiple stable extensions correspond to different "interpretations" of the world, whereas a lack of stable extensions corresponds to the case that for all possible sets of assumptions one eventually arrives at contradictory information. The extension existence problem thus asks whether one can obtain consistent knowledge of the world. Similarly, the problem of deciding whether a certain information is derivable gives rise to two possible modes of inference: the first, *credulous reasoning*, is to determine whether a given formula φ appears in at least one stable extension of a given default theory (W, D) (written: $(W, D) \models^{\text{cred}} \varphi$); the second, *skeptical reasoning*, consists of deciding whether φ is contained in all stable extensions of (W, D) (written: $(W, D) \models^{\text{skep}} \varphi$). The associated decision problems are the *credulous reasoning problem* and the *skeptical reasoning problem*:

Problem: $\textsc{Cred}_{\text{DL}}(B)$
Input: A B-formula φ and a B-default theory (W, D)
Question: Does $(W, D) \models^{\text{cred}} \varphi$ hold?

Problem: $\textsc{Skep}_{\text{DL}}(B)$
Input: A B-formula φ and a B-default theory (W, D)
Question: Does $(W, D) \models^{\text{skep}} \varphi$ hold?

They can be interpreted as the problems to determine whether φ is respectively possible or certain under Γ. The complexity of these problems has been settled by Gottlob [Got92], who showed that for the Boolean standard base $B = \{\wedge, \vee, \neg\}$ all three problems are complete for classes in the second level of the polynomial hierarchy. It follows from Lemma 2.4.2 (1.) that the lower bound holds for arbitrary functional complete B. As does the upper bound: the given algorithm is independent of the particular choice of B and simply guesses and verifies a set of generating defaults.

Theorem 3.1.7 ([Got92]) *Let B be a finite set of Boolean functions such that $[B] = \mathsf{BF}$. Then $\textsc{Ext}(B)$ and $\textsc{Cred}_{\text{DL}}(B)$ are Σ_2^{p}-complete, whereas $\textsc{Skep}_{\text{DL}}(B)$ is Π_2^{p}-complete.*

We refine this result and classify the complexity of the above problems for all finite sets of Boolean functions in Sections 4.1 and 5.1. Beyond these decision problems, Section 6.1 is devoted to the study of the complexity of counting the actual number of stable extensions of a given default theory. The possibility of translating default logic into the other nonmonotonic logics considered in this thesis, and vice versa, is finally studied in Sections 7.2 and 7.3.

3.2 Autoepistemic Logic

Autoepistemic logic was introduced by Moore [Moo85] in 1985. Albeit originally created to overcome difficulties present in the nonmonotonic modal logics proposed by McDermott and Doyle [MD80], it was soon shown to embed several prominent nonmonotonic formalisms such as Reiter's default logic [Rei80] or McCarthy's circumscription [McC80]. Thereby autoepistemic logic can be regarded a unified base for nonmonotonic reasoning [Nie93].

Autoepistemic logic extends propositional logic with a unary modal operator L expressing the beliefs of an ideally rational agent, by what we mean an agent that believes in exactly the logical consequences of his knowledge and beliefs.

Definition 3.2.1 (Autoepistemic formulae) *Let B be a finite set of Boolean functions. Then the set of* autoepistemic B-formulae $\mathcal{L}_{\mathrm{ae}}(B)$ *is defined as*

$$\varphi ::= \psi \mid f(\varphi, \dots, \varphi) \mid L\varphi,$$

where f is a Boolean function from B and ψ is a propositional B-formula. The consequence relation $\models$ of propositional logic is extended to $\mathcal{L}_{\mathrm{ae}}(B)$ by simply treating $L\varphi$ as an atomic formula.

As above, we will drop the prefix "B-" if $[B] = \mathsf{BF}$ or the meaning is clear from the context.

The formula $L\varphi$ means that the agent can deduce φ. For example, the intuitive meaning of the formula $Lx \to y$ is that if the agent believes in x, then y holds. An agent with knowledge base $\Sigma_1 := \{x, Lx \to y\}$ would thus believe in x (that is, Lx is true) and deduce that y is a fact; while for $\Sigma_1 \setminus \{x\} = \{Lx \to y\}$ the agent has no reason to believe in x (that is, Lx is false).

To formally capture the set of beliefs of an agent, we introduce the notion of *stable expansions*. Similar to stable extensions, stable expansions are defined as the fixed points of an operator deriving the logical consequences of the agent's knowledge and belief.

Definition 3.2.2 (Stable Expansions) *Let B be a finite set of Boolean functions and let $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ be a set of autoepistemic B-formulae. A set $\Delta \subseteq \mathcal{L}_{\mathrm{ae}}$ is called* stable expansion *of Σ if it satisfies the condition*

$$\Delta = \mathrm{Th}(\Sigma \cup L(\Delta) \cup \neg L(\overline{\Delta})),$$

where $L(\Delta) := \{L\varphi \mid \varphi \in \Delta\}$ and $\neg L(\overline{\Delta}) := \{\neg L\varphi \mid \varphi \notin \Delta\}$.

We give an example to provide a better understanding of the semantics and to highlight differences with default logic.

Example 3.2.3 *Let x and y be propositions and consider the set of autoepistemic formulae $\Sigma_1 = \{x, Lx \to y\}$ defined above. Σ_1 has a unique stable expansion containing both propositions x and y. The set $\Sigma_2 := \{Lx\}$, on the other hand, admits no stable expansion: Assume that Δ was a stable expansion of Σ_2. Of course, $Lx \in \Delta$.*

Furthermore either $x \in \Delta$ *or* $x \notin \Delta$. *In the former case, x cannot be derived from* $\mathrm{Th}(\{Lx\} \cup L(\Delta) \cup \neg L(\overline{\Delta}))$*—contradictory to* Δ *being a stable expansion. In the latter case, we obtain* $\neg Lx \in \mathrm{Th}(\{Lx\} \cup L(\Delta) \cup \neg L(\overline{\Delta}))$ *from* $x \in \overline{\Delta}$. *Consequently,* $\{Lx, \neg Lx\} \subseteq \Delta$ *and* $\Delta = \mathcal{L}_{\mathrm{ae}}$. *This contradicts* $x \notin \Delta$.

By contrast, $\Sigma_3 := \{Lx \to x\}$ *has two stable expansions: one containing x and the other not containing x. Note that the stable expansion containing x is self-justified in that only the belief in x allows its derivation.*

The theory Σ_3 emphasizes an essential difference in the semantics of default logic and autoepistemic logic. Stable extensions are minimal deductively closed sets with respect to the given knowledge base and given default rules. The definition of stable expansions does not require this "groundedness" of beliefs, which may lead to peculiarities in some situations and has spawned a dicussion on alternate definitions of stable expansions (confer, for example, [Kon88, MT89, MT90]).

As an autoepistemic theory may admit no or several stable expansions, the *expansion existence problem*, the *credulous reasoning problem* and the *skeptical reasoning problem* also arise in the context of autoepistemic reasoning. Let B be a finite set of Boolean formulae, $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$, and $\varphi \in \mathcal{L}_{\mathrm{ae}}(B)$. We write $\Sigma \models^{\mathrm{cred}} \varphi$ if φ is contained in any stable expansion of Σ, and $\Sigma \models^{\mathrm{skep}} \varphi$ if φ is contained in all stable expansions of Σ.

Problem: $\textsc{Exp}(B)$
Input: A set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$
Question: Does Σ have a stable expansion?

Problem: $\textsc{Cred}_{\mathrm{AE}}(B)$
Input: A set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ and a formula $\varphi \in \mathcal{L}_{\mathrm{ae}}(B)$
Question: Does $\Sigma \models^{\mathrm{cred}} \varphi$ hold?

Problem: $\textsc{Skep}_{\mathrm{AE}}(B)$
Input: A set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ and a formula $\varphi \in \mathcal{L}_{\mathrm{ae}}(B)$
Question: Does $\Sigma \models^{\mathrm{skep}} \varphi$ hold?

Gottlob proved that for $B = \{\wedge, \vee, \neg\}$ these problems are complete for the second level of the polynomial hierarchy [Got92]. His result generalizes to arbitrary functional complete sets B analogously to the discussion preceding Theorem 3.1.7. We hence obtain:

Theorem 3.2.4 ([Got92]) *Let B be a finite set of Boolean functions such that* $[B] = \mathsf{BF}$. *Then* $\textsc{Exp}(B)$ *and* $\textsc{Cred}_{\mathrm{AE}}(B)$ *are* Σ_2^{p}*-complete, whereas* $\textsc{Skep}_{\mathrm{AE}}(B)$ *is* Π_2^{p}*-complete.*

Our results in Sections 4.2 and 5.2 extend this theorem to all finite sets of allowed Boolean functions. Section 6.2 then complements these results by classifying the complexity of counting the number of stable expansions of a given set of autoepistemic formulae. The results on translations from and to fragments of autoepistemic logic are presented in Sections 7.2 and 7.4.

A central tool for the study of the computational complexity of the above problems will be the following finite characterization of stable expansions given by Niemelä [Nie90]. For $\varphi \in \mathcal{L}_{\mathrm{ae}}(B)$, let $\mathrm{SF}^L(\varphi) := \{L\psi \mid L\psi \in \mathrm{SF}(\varphi)\}$ be the set of its L-prefixed subformulae.

Definition 3.2.5 (Full sets) *Let B be a finite set of Boolean functions. For a set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$, a set $\Lambda \subseteq \mathrm{SF}^L(\Sigma) \cup \neg\mathrm{SF}^L(\Sigma)$ is* Σ-full *if for each $L\varphi \in \mathrm{SF}^L(\Sigma)$,*

1. *$\Sigma \cup \Lambda \models \varphi$ if and only if $L\varphi \in \Lambda$, and*
2. *$\Sigma \cup \Lambda \not\models \varphi$ if and only if $\neg L\varphi \in \Lambda$,*

where $\neg\mathrm{SF}^L(\Sigma) := \{\neg\varphi \mid \varphi \in \mathrm{SF}^L(\Sigma)\}$.

Lemma 3.2.6 ([Nie90]) *Let B be a finite set of Boolean functions and $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$.*

- *Let Λ be a Σ-full set. Then for every $L\varphi \in \mathrm{SF}^L(\Sigma)$ either $L\varphi \in \Lambda$ or $\neg L\varphi \in \Lambda$.*
- *There is a one-to-one correspondence of Σ-full sets and stable expansions of Σ.*

To make this one-to-one correspondence more precise, say that a formula is *quasi-atomic* if it is atomic or else begins with an L. Denote by $\mathrm{SF}^q(\varphi)$ the set of all maximal quasi-atomic subformulae of φ (in the sense that a quasi-atomic subformula is maximal if it is not a subformula of another quasi-atomic subformula of φ). For example, we have $\mathrm{SF}^q(\neg L(\neg x \wedge Lz) \vee y) = \{L(\neg x \wedge Lz), y\}$ and $\mathrm{SF}^q(LLx) = \{LLx\}$. Further write $\mathrm{SE}_\Sigma(\Lambda)$ for the stable expansion of $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}$ corresponding to Λ and say that Λ is its *kernel*.

Definition 3.2.7 ($\models_L$) *Let $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}$ and let $\varphi \in \mathcal{L}_{\mathrm{ae}}$. We define the consequence relation $\models_L$ recursively as*

$$\Sigma \models_L \varphi \iff \Sigma \cup \mathrm{SB}_\Sigma(\varphi) \models \varphi,$$

where $\mathrm{SB}_\Sigma(\varphi) := \{L\chi \in \mathrm{SF}^q(\varphi) \mid \Sigma \models_L \chi\} \cup \{\neg L\chi \mid L\chi \in \mathrm{SF}^q(\varphi), \Sigma \not\models_L \chi\}$.

The point in defining the consequence relation $\models_L$ is that, once a Σ-full set has been determined, it describes membership in the stable expansion corresponding to Λ:

Lemma 3.2.8 ([Nie90]) *Let $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}$, let Λ be a Σ-full set, and let $\varphi \in \mathcal{L}_{\mathrm{ae}}$. It holds that $\Sigma \cup \Lambda \models_L \varphi$ if and only if $\varphi \in \mathrm{SE}_\Sigma(\Lambda)$.*

3.3 Circumscription

We now turn to the last nonmonotonic logic considered in this thesis.

Like any other knowledge representation formalism, logic has to deal with the *qualification problem* that denotes the impossibility of listing all conditions required for a real-world action to have its intended effect. To overcome this problem, McCarthy introduced the nonmonotonic logic circumscription [McC80]. Circumscription allows to conclude that the objects that can be shown to have a certain property P by reasoning from a given knowledge base Γ are all objects that satisfy P. We consider propositional circumscription as defined by Lifschitz [Lif85], which is known to coincide with reasoning under the *extended closed world assumption* [GPP89], in which all formulae involving only propositions from P that cannot be derived from Γ are assumed to be false

Given a first-order theory Γ that contains a predicate P, circumscribing P amounts to selecting only the models of Γ in which P is assigned the value true on a minimal set of tuples. The key intuition behind this rationale is that minimal models have as few exceptions as possible and, thus, embody common sense. In propositional logic, P is simply a set of propositions; whence propositional circumscription asks for the minimal models of Γ with respect to the coordinatewise partial order induced on P by $0 < 1$. The remaining propositions are partitioned into sets Q and Z, where the propositions in Q are fixed and the propositions in Z are allowed to vary in minimizing the extent of P.

Definition 3.3.1 ($\leq_{(P,Q,Z)}$ and (P,Q,Z)-minimal models) *Let (P,Q,Z) be a partition of the set of propositions and B be a finite set of Boolean functions. The preorder $\leq_{(P,Q,Z)}$ on assignments $\sigma, \sigma' \colon P \cup Q \cup Z \to \{0,1\}$ is defined via*

$$\sigma \leq_{(P,Q,Z)} \sigma' :\Longleftrightarrow \sigma \cap P \subseteq \sigma' \cap P \text{ and } \sigma \cap Q = \sigma' \cap Q.$$

(Recall that we identify assignments with the set of propositions set to 1*).*

Further, write $\sigma <_{(P,Q,Z)} \sigma'$ if $\sigma \leq_{(P,Q,Z)} \sigma'$ and $\sigma \cap P \neq \sigma' \cap P$. A model σ of a B-formula φ is minimal with respect to (P,Q,Z) *(or (P,Q,Z)-*minimal*) if there is no model σ' of φ such that $\sigma' <_{(P,Q,Z)} \sigma$.*

Circumscription has also been studied in a restricted form, where all propositions are subject to minimization (that is, $Q = Z = \emptyset$). Following [Nor04], we will call this restricted form *basic circumscription* and write $\leq$ instead of $\leq_{(P,\emptyset,\emptyset)}$ (note that $\leq_{(P,\emptyset,\emptyset)}$ coincides with the coordinatewise partial order on assignments induced by $0 < 1$).

Definition 3.3.2 (Circumscriptive models and inference) *Let (P,Q,Z) be a partition of the set of propositions and let B, B' be finite sets of Boolean functions. An assignment σ is a* circumscriptive model *of the B-formula φ (written: $\sigma \models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$) if σ is a (P,Q,Z)-minimal model of φ. A B'-formula ψ can be* circumscriptively inferred *from φ (written: $\varphi \models^{\mathrm{circ}}_{(P,Q,Z)} \psi$) if ψ holds in all (P,Q,Z)-minimal models of φ.*

Example 3.3.3 *Let $P := \{x\}$, $Q := \{y\}$, and $Z := \{z\}$ partition the set of propositions. Then the formula $y \to (x \wedge z)$ has three (P,Q,Z)-minimal models: $\emptyset$, $\{z\}$, and $\{x,y,z\}$. It hence holds that $y \to (x \wedge z) \models^{\mathrm{circ}}_{(P,Q,Z)} \neg x \vee z$, whereas*

$y \to (x \land z) \not\models \neg x \lor z$ as witnessed by the assignment that sets to true x and false y and z.

Circumscription differs from the previously introduced nonmonotonic logics in that it restricts the semantics of propositional logics to minimal models instead of introducing new concepts. Hence, the question of consistency of the knowledge base is the same as for its satisfiability.

Remark 3.3.4 *One might consider the model checking problem instead. Let B be a finite set of Boolean functions. The model checking problem for circumscription for B-formulae, $\textsc{CircMC}(B)$, is defined as the task to decide whether a given assignment σ is (P, Q, Z)-minimal for a given set $\Gamma \subseteq \mathcal{L}(B)$. The complexity of the basic version $\textsc{CircMC}_{Q,Z=\emptyset}(B)$ of this problem has been resolved for all finite sets B by Kirousis and Kolaitis in an unpublished note [KK01a]. They prove that under polynomial-time many-one reductions $\textsc{CircMC}_{Q,Z=\emptyset}(B)$ is* NP*-complete if $\mathsf{D}_1 \subseteq [B]$ or $\mathsf{S}_{12} \subseteq [B]$ or $\mathsf{S}_{02} \subseteq [B]$, and contained in* P *in all other cases. It is easy to see that their result generalizes to $\textsc{CircMC}(B)$ and continues to hold for constant-depth reductions.*

We are thus left to consider the problem of deciding whether a formula can be inferred from the circumscription of a given knowledge base. We define this problem as the *inference problem* for B-formulae in propositional circumscription.

Problem:	$\textsc{CircInf}(B)$
Input:	A B-formula φ, a set $\Gamma \subseteq \mathcal{L}(B)$, and a partition (P, Q, Z) of the propositions
Question:	Does $\Gamma \models^{\text{circ}}_{(P,Q,Z)} \varphi$ hold?

Moreover, define $\textsc{CircInf}_{Q,Z=\emptyset}$ as the restriction of the above problem to basic circumscription, that is, instances satisfying $Q = Z = \emptyset$.

An upper bound on the complexity of these problems has first been given by Cadoli and Lenzerini [CL90], who showed that $\textsc{CircInf}(\{\land, \lor, \neg\})$ is contained in Π^{P}_2. Soon afterwards, a matching lower bound for $\textsc{CircInf}_{Q,Z=\emptyset}$ was provided by Eiter and Gottlob [EG93]. It can be verified that the upper bound generalizes to arbitrary functional complete sets of Boolean functions. Lemma 2.4.2 (1.) hence yields the following theorem.

Theorem 3.3.5 *Let B be a finite set of Boolean functions such that $[B] = \mathsf{BF}$. Then $\textsc{CircInf}(B)$ and $\textsc{CircInf}_{Q,Z=\emptyset}(B)$ are Π^{P}_2-complete.*

We extend Theorem 3.3.5 to all finite sets B of Boolean functions in Section 5.3. Beyond, we also consider the problem of counting the number of minimal models of a theory Γ with respect to the (P, Q, Z)-preorder. These results are presented in Section 6.3. Finally, Sections 7.3 and 7.4 contain the results on the ability to translate from circumscription into default logic or autoepistemic logic and vice versa.

CHAPTER 4

EXISTENCE

We start by analyzing the computational complexity of the problem to decide the consistency of a given knowledge base, that is, the extension existence problem for default logic and the expansion existence problem for autoepistemic logic.

4.1 EXISTENCE OF STABLE EXTENSIONS

We first classify the complexity of the extension existence problem for B-default theories, $\text{EXT}(B)$, for all possible sets B of Boolean functions. The following theorem proves that the complexity of $\text{EXT}(B)$ forms a tetrachotomy: it remains Σ_2^{p}-complete only if $[B \cup \{1\}]$ comprises all Boolean functions; lowers to Δ_2^{p}-completeness for monotone sets B that comprise the constant 0, conjunctions and disjunctions; and becomes tractable for all remaining monotone sets B, in particular being P-complete (if $[B]$ contains 0 and either conjunctions or disjunctions), NL-complete (if $[B \cup \{1\}]$ contains 0 and 1 only) or trivial (if B is 1-reproducing); lastly, if B consists of affine functions and is not 1-reproducing then the complexity of $\text{EXT}(B)$ drops by one level of the polynomial hierarchy to NP-completeness.

The decrease to Δ_2^{p} stems from the fact that for monotone functions there exists at most one stable extension, whose generating defaults can be iteratively computed. If furthermore the implication problem is polynomial-time decidable or a stable extension is guaranteed to exist then the problem becomes tractable. Finally, the decrease to NP-completeness for affine functions that are not 1-reproducing is due to the possibility of efficiently verifying a set of generating defaults. The classification is illustrated in Figure 4.1 on page 53.

Theorem 4.1.1 *Let B be a finite set of Boolean functions. Then* $\text{EXT}(B)$ *is*

1. Σ_2^{p}*-complete if* $\mathsf{S}_1 \subseteq [B]$ *or* $\mathsf{D} \subseteq [B]$,
2. Δ_2^{p}*-complete if* $\mathsf{S}_{11} \subseteq [B] \subseteq \mathsf{M}$,
3. NP*-complete if* $[B] \in \{\mathsf{N}, \mathsf{N}_2, \mathsf{L}, \mathsf{L}_0, \mathsf{L}_3\}$,
4. P*-complete if* $[B] \in \{\mathsf{V}, \mathsf{V}_0, \mathsf{E}, \mathsf{E}_0\}$,
5. NL*-complete if* $[B] \in \{\mathsf{I}, \mathsf{I}_0\}$, *and*
6. *trivial in all other cases (that is, if* $[B] \subseteq \mathsf{R}_1$*),*

with respect to constant-depth reductions.

Notice also that the complexity of $\textsc{Ext}(B)$ is asymmetric in the sense that there exist sets of functions B such that $\textsc{Ext}(B)$ is trivial while $\textsc{Ext}(\mathrm{dual}(B))$ is Σ_2^{p}-complete, where $\mathrm{dual}(B) := \{\mathrm{dual}(f) \mid f \in B\}$. An informal explanation of this asymmetry is that default logic only allows for "negative introspection": a default can be applied only if the *negation of its justification is not contained in the stable extension*. For example, the negation of 1-reproducing functions cannot be 1-reproducing, whence this introspection becomes meaningless.

The proof of Theorem 4.1.1 will be established from the lemmas in this section. We commence with an auxiliary lemma that substantially reduces the number of cases to be considered.

Lemma 4.1.2 $\textsc{Ext}(B) \equiv_{\mathrm{cd}} \textsc{Ext}(B \cup \{1\})$ *for each finite set B of Boolean functions.*

Proof. It suffices to show that $\textsc{Ext}(B \cup \{1\}) \leq_{\mathrm{cd}} \textsc{Ext}(B)$. We essentially substitute the constant 1 by a new variable t that is forced to be true (a trick that goes back to Lewis [Lew79]).

Given a B-default theory (W, D), the reduction maps $(W, D) \mapsto (W', D')$, where $W' := W_{[1/t]} \cup \{t\}$, $D' := D_{[1/t]}$ and t is a new variable not occurring in (W, D). If (W', D') possesses a stable extension E', then $t \in E'$. Hence, $E'_{[t/1]}$ is a stable extension of (W, D). On the other hand, if E is a stable extension of (W, D), then $\mathrm{Th}(E_{[1/t]} \cup \{t\}) = E_{[1/t]}$ is a stable extension of (W', D'). Therefore, each extension E of (W, D) corresponds to the extension $E_{[1/t]}$ of (W', D'), and vice versa. □

The next lemma holds the key to the complexity of the extension existence problem for 1-reproducing and monotone functions.

Lemma 4.1.3 *Let B be a finite set of Boolean functions. Let (W, D) be a B-default theory. If $[B] \subseteq \mathsf{R}_1$ then (W, D) has a unique stable extension. If $[B] \subseteq \mathsf{M}$ then (W, D) has at most one stable extension.*

Proof. For $[B] \subseteq \mathsf{R}_1$, every premise, justification and consequent is 1-reproducing. As all consequences of 1-reproducing functions are again 1-reproducing and the negation of a 1-reproducing function is not 1-reproducing, the justifications in D become irrelevant. Hence, the characterization of stable extensions from the first item in Theorem 3.1.3 simplifies to the following iterative construction: $E_0 := W$ and $E_{i+1} := \mathrm{Th}(E_i) \cup \{\gamma \mid \frac{\alpha:\beta}{\gamma} \in D, \alpha \in E_i\}$. Then $E = \bigcup_{i \in \mathbb{N}} E_i$ is the unique stable extension of (W, D). For a similar result confer [BO02, Theorem 4.6].

For $[B] \subseteq \mathsf{M}$, every formula is either 1-reproducing or equivalent to 0. As rules with justification equivalent to 0 are never applicable, each B-default theory (W, D) with finite D has at most one stable extension by the same argument as above. □

Remark 4.1.4 *There exist monotone default theories that do not admit a stable extension. Consider, for example, the default theory* $(W, \{\frac{\psi:1}{0}\})$ *with* $\psi \in \mathcal{L}(\{\wedge, \vee\})$ *and* $W \models \psi$.

As an immediate corollary of Lemma 4.1.3, the credulous and the skeptical reasoning problem are equivalent for the above choices of the Boolean functions.

Corollary 4.1.5 *Let B be a finite set of Boolean functions such that* $[B] \subseteq \mathsf{R}_1$ *or* $[B] \subseteq \mathsf{M}$. *Then* $\textsc{Cred}_{\mathrm{DL}}(B) \equiv_{\mathrm{cd}} \textsc{Skep}_{\mathrm{DL}}(B)$.

Proof. Let B be a finite set of Boolean functions. If $[B] \subseteq \mathsf{R}_1$, then the claim trivially holds. On the other hand, if $[B] \subseteq \mathsf{M}$, then a given B-default theory (W, D) possesses a stable extension if and only if $(W, D) \models^{\mathrm{cred}} 1$, a consistent stable extension if and only if $(W, D) \not\models^{\mathrm{skep}} 0$, and the inconsistent stable extension if and only if $(W, \emptyset) \models^{\mathrm{skep}} 0$.

Hence, $(W, D) \models^{\mathrm{skep}} \varphi$ if and only if $(W, D) \models^{\mathrm{cred}} \varphi$ or $(W, D) \not\models^{\mathrm{cred}} 1$; and, similarly, $(W, D) \models^{\mathrm{cred}} \varphi$ if and only if $\big((W, D) \models^{\mathrm{skep}} \varphi$ and $(W, D) \not\models^{\mathrm{skep}} p)\big)$ or $(W, \emptyset) \models^{\mathrm{skep}} p$, where p is a fresh proposition. □

Lemma 4.1.6 *Let B be a finite set of Boolean functions such that* $[B] = \mathsf{M}$. *Then* $\textsc{Ext}(B)$ *is* Δ_2^{p}*-complete with respect to constant-depth reductions.*

Proof. We start by showing $\textsc{Ext}(B) \in \Delta_2^{\mathrm{p}}$. Let B be a finite set of Boolean functions such that $[B] = \mathsf{M}$. Let (W, D) be a B-default theory. If W is inconsistent, then (W, D) has a stable extension. Hence assume that W is consistent. As the negated justification $\neg\beta$ of every default rule $\frac{\alpha:\beta}{\gamma} \in D$ is either equivalent to the constant 1 or not 1-reproducing, it holds that in the former case $\neg\beta$ is contained in any stable extension, whereas in the latter $\neg\beta$ cannot be contained in a consistent stable extension of (W, D). We can distinguish between those two cases in polynomial time. Therefore, using the characterization of Theorem 3.1.3 (1.), we can iteratively compute the applicable defaults and test whether the premise of any default with consistent justification and unsatisfiable conclusion can be derived. Algorithm 4.1 implements these steps on a deterministic Turing machine using a coNP-oracle to test for implication of B-formulae. Clearly, Algorithm 4.1 terminates after a polynomial number of steps. Hence, $\textsc{Ext}(B)$ is contained in Δ_2^{p}.

To show the Δ_2^{p}-hardness of $\textsc{Ext}(B)$, we reduce from SNSAT. To this end, let $(\varphi^i)_{1 \leq i \leq n}$ be the given sequence of propositional formulae and assume without loss of generality that φ^i is in conjunctive normal form for all $1 \leq i \leq n$. For every proposition x_j or z_{ij} occurring in $(\varphi^i)_{1 \leq i \leq n}$, let x'_j respectively z'_{ij} be a fresh proposition, and define

$$\psi^i := \varphi^i_{[\neg x_1/x'_1, \ldots, \neg x_{i-1}/x'_{i-1}, \neg z_{i1}/z'_{i1}, \ldots, \neg z_{im_i}/z'_{i,m_i}]} \wedge \bigwedge_{j=1}^{i-1} (x_j \vee x'_j) \wedge \bigwedge_{j=1}^{m_i} (z_{ij} \vee z'_{ij}).$$

Algorithm 4.1 Determining the existence of a stable extension

Require: (W, D)
if $W \not\equiv 0$ **then**
  $G_{\text{new}} \leftarrow W$
  repeat
    $G_{\text{old}} \leftarrow G_{\text{new}}$
    for all $\frac{\alpha:\beta}{\gamma} \in D$ **do**
      if $G_{\text{old}} \models \alpha$ **and** $\beta \not\equiv 0$ **then**
        if $\gamma \equiv 0$ **then**
          return false
        end if
        $G_{\text{new}} \leftarrow G_{\text{new}} \cup \{\gamma\}$
      end if
    end for
  until $G_{\text{new}} = G_{\text{old}}$
end if
return true

The key observation in the relationship of φ^i and ψ^i is that, for all $c_1, \ldots, c_{i-1} \in \{0,1\}$, $\varphi^i_{[x_1/c_1,\ldots,x_{i-1}/c_{i-1}]}$ is unsatisfiable if and only if for each model σ of $\psi^i_{[x_1/c_1,\ldots,x_{i-1}/c_{i-1},x'_1/\neg c_1,\ldots,x'_{i-1}/\neg c_{i-1}]}$ there exists an index $1 \leq j \leq m_i$ such that σ sets to 1 both z_{ij} and z'_{ij}. We will use this observation to show that the B-default theory (W, D) defined below has a stable extension if and only if $(\varphi^i)_{1 \leq i \leq n}$ is an instance of SNSAT, that is, if and only if $\varphi^n_{[x_1/c_1,\ldots,x_{i-1}/c_{i-1}]}$ is satisfiable for $c_1, \ldots, c_{i-1}$ recursively defined via

$$c_i := 1 \iff \varphi^i_{[x_1/c_1,\ldots,x_{i-1}/c_{i-1}]} \text{ is satisfiable.} \tag{4.1}$$

Define $W := \{\psi^1, \ldots, \psi^n\}$ and

$$D := \left\{ \frac{\bigvee_{j=1}^{m_i}(z_{ij} \wedge z'_{ij}) \vee \bigvee_{j=1}^{i-1}(x_j \wedge x'_j) : 1}{x'_i} \;\middle|\; 1 \leq i < n \right\} \cup \left\{ \frac{\bigvee_{j=1}^{m_n}(z_{nj} \wedge z'_{nj}) \vee \bigvee_{j=1}^{n}(x_j \wedge x'_j) : 1}{0} \right\}.$$

We will prove the claim appealing to the characterization of stable extensions from Theorem 3.1.3 (1.). Let $E_0 := W$. If φ^1 is unsatisfiable then $\frac{\bigvee_{j=1}^{m_1}(z_{1j} \wedge z'_{1j}):1}{x'_1}$ is applicable and thus x'_1 is added to E_1. On the other hand, if φ^1 is satisfiable then there exists a model σ of φ^1. Define $\hat{\sigma}$ as the extension of σ defined as

$\hat{\sigma}(z'_{1j}) = \neg\sigma(z_{1j})$ for all $1 \leq j \leq m_1$. By virtue of $\sigma \models \varphi^1$ and the construction of $\hat{\sigma}$, we obtain that $\hat{\sigma} \models \psi^1$ while $\hat{\sigma} \not\models \bigvee_{j=1}^{m_1}(z_{1j} \wedge z'_{1j})$. Summarizing, φ^1 is unsatisfiable if and only if $\frac{\bigvee_{j=1}^{m_1}(z_{1j} \wedge z'_{1j}):1}{x'_1}$ is applicable.

Now suppose that E_i is such that for all $j < i$ the proposition x'_j is contained in E_i if and only if $\varphi^j_{[x_1/c_1,\ldots,x_{j-1}/c_{j-1}]}$ with $c_1, \ldots, c_{j-1}$ defined as in (4.1) is unsatisfiable. If $\varphi^i_{[x_1/c_1,\ldots,x_{i-1}/c_{i-1}]}$ is unsatisfiable then any model of the formula

$$\psi^i \wedge \bigwedge_{\substack{1 \leq j < i, \\ \sigma(c_j)=1}} x_j \wedge \bigwedge_{\substack{1 \leq j < i, \\ \sigma(c_j)=0}} x'_j \tag{4.2}$$

sets to 1 both z_{ij} and z'_{ij} for some $1 \leq j \leq m_i$. From (4.2) and the monotonicity of ψ^i, we obtain that for each model σ' of $\psi^i \wedge \bigwedge_{1 \leq j < i, \sigma(c_j)=0} x'_j$ there must exist either an index $1 \leq j < i$ such that σ' sets x_j and x'_j to 1, or an index $1 \leq j \leq m_i$ such that σ' sets z_{ij} and z'_{ij} to 1. Consequently, $\frac{\bigvee_{j=1}^{m_i}(z_{ij} \wedge z'_{ij}) \vee \bigvee_{j=1}^{i-1}(x_j \wedge x'_j):1}{x'_i}$ is applicable and $x'_i \in E_{i+1}$. On the other hand, if $\varphi^i_{[x_1/c_1,\ldots,x_{i-1}/c_{i-1}]}$ is satisfiable then there exists a model σ that can be extended to $\hat{\sigma}$ by $\hat{\sigma}(z'_{ij}) = \neg\sigma(z_{ij})$ for all $1 \leq j \leq m_i$ and $\hat{\sigma}(x'_j) = \neg\sigma(x_j)$ for all $1 \leq j < i$ such that $\hat{\sigma} \models \psi^i$ and $\hat{\sigma} \not\models \bigvee_{j=1}^{m_i}(z_{ij} \wedge z'_{ij}) \vee \bigvee_{j=1}^{i-1}(x_j \wedge x'_j)$. Summarizing, φ^i is unsatisfiable if and only if $\frac{\bigvee_{j=1}^{m_i}(z_{ij} \wedge z'_{ij}) \vee \bigvee_{j=1}^{i-1}(x_j \wedge x'_j):1}{x'_i}$ is applicable.

The direction from right to left now follows from the fact that φ_n is satisfiable if and only if $\frac{\bigvee_{j=1}^{m_n}(z_{ij} \wedge z'_{ij}) \vee \bigvee_{j=1}^{n}(x_i \wedge x'_i):1}{0}$ is not applicable, which in turn implies that (W, D) has a stable extension. Conversely, if $\varphi^n_{[x_1/c_1,\ldots,x_{n-1}/c_{n-1}]}$ is unsatisfiable with $c_1, \ldots, c_{n-1}$ defined as in (4.1), then any model of $\psi^i \wedge \bigwedge_{1 \leq j < i, \sigma(c_j)=0} x'_j$ sets to true either x_j and x'_j for some $1 \leq j < i$ or z_{ij} and z'_{ij} for some $1 \leq j \leq m_i$. As a result, the default $\frac{\bigvee_{j=1}^{m_n}(z_{ij} \wedge z'_{ij}) \vee \bigvee_{j=1}^{n}(x_j \wedge x'_j):1}{0}$ is applicable and (W, D) does not possess a stable extension.

Finally, observe that all formulae contained in (W, D) are monotone. Hence, (W, D) is a $\{\wedge, \vee, 0, 1\}$-default theory. As $\wedge$ and $\vee$ are efficient implementable in any set B such that $[B] = \mathsf{M}$ by Lemma 2.4.2 (2.) and the constant 1 can be eliminated by Lemma 4.1.2, the lemma is established. $\square$

Lemma 4.1.7 *Let B be a finite set of Boolean functions such that $\mathsf{N} \subseteq [B] \subseteq \mathsf{L}$. Then* $\text{E}\textsc{xt}(B)$ *is* NP*-complete with respect to constant-depth reductions.*

Proof. Let B be a finite set of Boolean functions. We start by showing $\text{E}\textsc{xt}(B) \in$ NP if $[B] \subseteq \mathsf{L}$. Given a B-default theory (W, D), we first guess a set $G \subseteq D$ which will serve as the set of generating defaults for a stable extension. Let

$G' := W \cup \{\gamma \mid \frac{\alpha:\beta}{\gamma} \in G\}$. We use Theorem 3.1.3 to verify whether $\mathrm{Th}(G')$ is indeed a stable extension of (W, D). For this we inductively compute generators G_i for the sets E_i from Theorem 3.1.3 (1.), until eventually $E_i = E_{i+1}$ (note that, because D is finite, this always occurs). We start by setting $G_0 = W$. Given G_i, we check for each rule $\frac{\alpha:\beta}{\gamma} \in D$, whether $G_i \models \alpha$ and $G' \not\models \neg\beta$ (as all formulae belong to $\mathcal{L}(B)$, this is possible by Theorem 2.5.1 (2.)). If so, then γ is put into G_{i+1}. If this process terminates (that is, if $G_i = G_{i+1}$), we check whether $G' = G_i$. By Theorem 3.1.3 (1.), this test is positive if and only if G generates a stable extension of (W, D).

To show the NP-hardness of $\textsc{Ext}(B)$ for $\mathsf{N} \subseteq [B]$, we establish a constant-depth reduction from $3\textsc{Sat}$ to $\textsc{Ext}(B)$. Let $\varphi = \bigwedge_{i=1}^{n}(\ell_{i1} \vee \ell_{i2} \vee \ell_{i3})$ and ℓ_{ij} be literals over the set of propositions $\{x_1, \ldots, x_m\}$ for $1 \le i \le n$, $1 \le j \le 3$. We transform φ to the B-default theory (W, D_φ) with $W := \emptyset$ and

$$D_\varphi := \left\{ \frac{1 : x_i}{x_i}, \frac{1 : \neg x_i}{\neg x_i} \,\middle|\, 1 \le i \le m \right\} \cup \left\{ \frac{\overline{\ell_{i1}} : \overline{\ell_{i2}}}{\ell_{i3}} \,\middle|\, 1 \le i \le n \right\},$$

where $\overline{\ell}$ denotes the literal of opposite polarity: $\overline{\ell} := \neg x$ if $\ell = x$ is a positive literal, and $\overline{\ell} := x$ if $\ell = \neg x$ is a negative literal.

To prove correctness of the reduction, first assume φ to be satisfiable. For each satisfying assignment $\sigma\colon \{x_1, \ldots, x_m\} \to \{0, 1\}$ of φ, we claim that

$$E := \mathrm{Th}(\{x_i \mid \sigma(x_i) = 1\} \cup \{\neg x_i \mid \sigma(x_i) = 0\})$$

is a stable extension of (W, D_φ). We will verify this claim with the help of Theorem 3.1.3 (1.). Starting with $E_0 = \emptyset$, we already get $\mathrm{Th}(E_1) = E$ by the default rules $\frac{1:x_i}{x_i}$ and $\frac{1:\neg x_i}{\neg x_i}$ in D_φ. As σ is a satisfying assignment for φ, each consequent of a default rule in D_φ is already in E. Hence, $E_3 = E_2$ and therefore $E = \bigcup_{i \in \mathbb{N}} E_i$ is a stable extension of (W, D_φ).

Conversely, assume that E is a stable extension of (W, D_φ). Because of the default rules $\frac{1:x_i}{x_i}$ and $\frac{1:\neg x_i}{\neg x_i}$, we either get $x_i \in E$ or $\neg x_i \in E$ for all $i = 1, \ldots, m$. The rules of the type $\frac{\overline{\ell_{i1}}:\overline{\ell_{i2}}}{\ell_{i3}}$ ensure that E contains at least one literal from each clause $\ell_{i1} \vee \ell_{i2} \vee \ell_{i3}$ in φ: if all $\overline{\ell_{i1}}, \overline{\ell_{i2}}, \overline{\ell_{i3}}$ were contained in E, then $\frac{\overline{\ell_{i1}}:\overline{\ell_{i2}}}{\ell_{i3}}$ would be applicable and so E would have to be inconsistent—contradictory to Corollary 3.1.5. As E is deductively closed, E contains φ. Therefore, φ is satisfiable. $\square$

Lemma 4.1.8 *Let B be a finite set of Boolean functions such that $[B] \in \{\mathsf{E}, \mathsf{E}_0, \mathsf{V}, \mathsf{V}_0\}$. Then* $\textsc{Ext}(B)$ *is* P*-complete with respect to constant-depth reductions.*

Proof. Let B be a finite set of Boolean functions such that $[B] \in \{\mathsf{E}, \mathsf{E}_0, \mathsf{V}, \mathsf{V}_0\}$. Membership in P is is obtained from Algorithm 4.1, as for these types of B-formulae, we have an efficient test for implication (Theorem 2.5.1).

To prove P-hardness for $\mathsf{E}_0 \subseteq [B]$, we provide a reduction from the complement of $\textsc{HGap}$ restricted to hypergraphs whose edges contain at most two source nodes. To this end, we transform a given instance (H, S, t) to the $\textsc{Ext}(\{\wedge, 0, 1\})$-instance (W, D) with

$$W := \{p_s \mid s \in S\}, \;\; D := \left\{ \frac{\bigwedge_{v \in \mathrm{src}(e)} p_v : 1}{p_{\mathrm{dest}(e)}} \,\middle|\, e \in E \right\} \cup \left\{ \frac{p_t : 1}{0} \right\}$$

with pairwise distinct propositions p_v for $v \in V$. It is easy to verify that $(H, s, t) \in \textsc{HGap} \iff (W, D) \notin \textsc{Ext}(\{\wedge, 0, 1\})$. Using Lemma 4.1.2 and replacing $\wedge$ by its B-representation, we obtain $\overline{\textsc{HGap}} \leq_{\mathrm{cd}} \textsc{Ext}(B)$ for all finite sets B such that $\mathsf{E}_0 \subseteq [B]$. As P is closed under complementation, $\textsc{Ext}(B)$ is P-complete.

For $\mathsf{V}_0 \subseteq [B]$, set

$$W := \left\{ \bigvee_{s \notin S} p_s \right\}, \; D := \left\{ \frac{\bigvee_{v \in V \setminus \mathrm{src}(e)} p_v : 1}{\bigvee_{v \in V \setminus (\mathrm{src}(e) \cup \{\mathrm{dest}(e)\})} p_v} \,\middle|\, e \in E \right\} \cup \left\{ \frac{\bigvee_{v \in V \setminus \{t\}} p_v : 1}{0} \right\}.$$

We claim that this mapping realizes the reduction $\overline{\textsc{HGap}} \leq_{\mathrm{cd}} \textsc{Ext}(\{\vee, 0, 1\})$. First suppose that t can be reached from S in H. Then there exists a sequence $(S_i)_{0 \leq i \leq n}$ of sets of nodes such that $S_0 = S$, $t \in S_n$, and for all $0 \leq i < n$, S_{i+1} is obtained from S_i by adding the destination $\mathrm{dest}(e)$ of a hyperedge $e \in E$ with $\mathrm{src}(e) \subseteq S_i$. Let $(e_i)_{0 \leq i < n}$ denote the corresponding sequence of hyperedges used to obtain S_{i+1} from S_i. Then, for all $0 \leq i < n$, the following holds:

$$\mathrm{src}(e_i) \subseteq S_i \iff \frac{\bigvee_{v \in V \setminus \mathrm{src}(e)} p_v : 1}{\bigvee_{v \in V \setminus (\mathrm{src}(e) \cup \{\mathrm{dest}(e)\})} p_v} \text{ is applicable in } E_i,$$

where $(E_i)_{i \in \mathbb{N}}$ is the sequence from Theorem 3.1.3 (1.). As $\bigcup_{i \in \mathbb{N}} E_i$ is guaranteed to be unique by Lemma 4.1.3 and $t \in S_n$, we obtain that $0 \in E_{n+1}$. Consequently, (W, D) does not possess a stable extension.

Conversely, if (W, D) does not admit a stable extension, then 0 has to be derivable. Accordingly, there exists a sequence of defaults $(d_i)_{0 \leq i \leq n}$ such that the premise of d_i can be derived from $W \cup \left\{ \gamma \,\middle|\, d_j = \frac{\alpha : \beta}{\gamma}, 0 \leq j < i \right\}$ and $d_n = \frac{\bigvee_{v \in V \setminus \{t\}} p_v : 1}{0}$. By construction of (W, D), this sequence can be translated into a sequence $(S_i)_{0 \leq i \leq n}$ of node sets in the hypergraph such that $S_0 = S$, $t \in S_n$, and for all $0 \leq i < n$, S_{i+1} is obtained from S_i by adding the destination $\mathrm{dest}(e)$ of a hyperedge $e \in E$ with $\mathrm{src}(e) \subseteq S_i$. Consequently, t is reachable from S in H and we conclude that $\overline{\textsc{HGap}} \leq_{\mathrm{cd}} \textsc{Ext}(\{\vee, 0, 1\})$. Using Lemma 5.1.2, we get $\overline{\textsc{HGap}} \leq_{\mathrm{cd}} \textsc{Ext}(\{\vee, 0\})$.

To see that $\textsc{Ext}(\{\vee, 0\}) \leq_{\mathrm{cd}} \textsc{Ext}(B)$ for all finite sets B such that $\mathsf{V}_0 \subseteq [B]$, let f be the B-representation of $x \vee y$. Replace all occurrences of $\vee$ in W, D and φ with f and call the result W^B, D^B and φ^B. The variables x or y may occur several times in the body of f, hence the $\textsc{Ext}(B)$-instance $((W^B, D^B), \varphi^B)$ might be exponential in the length of the original input. To avoid this blowup, we exploit the associativity of $\vee$: we insert parentheses such that the disjunctions in each of the above formulae are transformed into tree of logarithmic depth. Concluding, $\textsc{Ext}(B)$ is P-complete. $\square$

Lemma 4.1.9 *Let B be a finite set of Boolean functions such that $[B] \in \{\mathsf{I}, \mathsf{I}_0\}$. Then* $\textsc{Ext}(B)$ *is* NL*-complete with respect to constant-depth reductions.*

Proof. Let B be a finite set of Boolean functions such that $[B] \in \{\mathsf{I}, \mathsf{I}_0\}$. We will first show membership in NL by giving a reduction to the complement of the graph accessibility problem, $\overline{\textsc{Gap}}$. Let (W, D) be a B-default theory. Analogously to the proof of Lemma 4.1.6, it holds that (W, D) has a stable extension if and only if either W is inconsistent or the conclusions of all applicable defaults are consistent. Assume that W is consistent and denote by $D' \subseteq D$ those defaults $\frac{\alpha:\beta}{\gamma} \in D$ with $\beta \not\equiv 0$. Then a B-default rule $\frac{\alpha:\beta}{\gamma} \in D'$ is applicable if and only if the proposition α is contained in W or itself the conclusion of an applicable default. Therefore, testing whether the conclusions of all applicable defaults are consistent is essentially equivalent to solving a reachability problem in a directed graph. Define $G_{(W,D)}$ as the directed graph (V, E) with

$$\begin{aligned} V &:= \{0,1\} \cup W \cup \left\{\alpha, \gamma \,\middle|\, \tfrac{\alpha:\beta}{\gamma} \in D\right\}, \\ E &:= \{(1,x) \mid x \in W\} \cup \left\{(\alpha,\gamma) \,\middle|\, \tfrac{\alpha:\beta}{\gamma} \in D, \beta \not\equiv 0\right\} \end{aligned}$$

if W is consistent, and

$$\begin{aligned} V &:= \{0,1\}, \\ E &:= \emptyset \end{aligned}$$

otherwise. It is easy to see that (W, D) has a stable extension if and only if there is no path from 1 to 0 in $G_{(W,D)}$. Thus the function mapping the given B-default theory (W, D) to the $\textsc{Gap}$-instance $(G_{(W,D)}, 1, 0)$ constitutes a reduction from $\textsc{Ext}(B)$ to $\overline{\textsc{Gap}}$. As the consistency of W can be determined in AC^0, the reduction can be computed using constant-depth circuits. Membership in NL follows from the closure of NL under complementation.

To show NL-hardness, we establish a constant-depth reduction in the converse direction. For a directed graph $G = (V, E)$ and two nodes $s, t \in V$, we transform the given $\textsc{Gap}$-instance (G, s, t) to (W, D) with

$$W := \{p_s\}, \;\; D := \left\{ \tfrac{p_u:p_u}{p_v} \,\middle|\, (u,v) \in E \right\} \cup \left\{ \tfrac{p_t:p_t}{0} \right\}$$

Clearly, $(G, s, t) \in \textsc{Gap}$ if and only if (W, D) does not have a stable extension. As NL is closed under complementation, the lemma is established. □

Proof of Theorem 4.1.1. Let B be a finite set of Boolean functions.

If $\mathsf{S}_1 \subseteq [B]$ or $[B] \subseteq \mathsf{D}$, then in both cases $\mathsf{BF} = [B \cup \{1\}]$. The first claim hence follows from Theorem 3.1.7 and Lemma 4.1.2. The second claim follows similarly from Lemmas 4.1.2 and 4.1.6.

For the third claim, it suffices to prove the NP-completeness of $\textsc{Ext}(B)$ for every finite set B such that $\mathsf{N} \subseteq [B] \subseteq \mathsf{L}$. This has been shown in Lemma 4.1.7.

The remaining cases $[B] \in \{\mathsf{N}_2, \mathsf{L}_0, \mathsf{L}_3\}$ follow from Lemma 4.1.2, because $[\mathsf{N}_2 \cup \{1\}] = \mathsf{N}$ and $[\mathsf{L}_0 \cup \{1\}] = [\mathsf{L}_3 \cup \{1\}] = \mathsf{L}$.

The fourth, fifth, and sixth claim follow directly from respectively Lemma 4.1.8, Lemma 4.1.9, and Lemma 4.1.3. □

One might also be interested in a variant of the extension existence problem that considers consistent stable extensions only. Define this problem as

Problem:	$\text{EXT}'(B)$
Input:	A B-default theory (W, D)
Question:	Does (W, D) have a consistent stable extension?

It is easily observed that the complexity classification of $\text{EXT}'(B)$ is the identical to that of $\text{EXT}(B)$:

Corollary 4.1.10 *Let B be a finite set of Boolean functions. Then $\text{EXT}'(B)$ is equivalent to $\text{EXT}(B)$ with respect to constant-depth reductions.*

Proof. The proof of the corollary follows directly from the proof of Theorem 4.1.1 and the complexity of the satisfiability problem for B-formulae. Indeed, the reduction used to prove the Σ_2^{p}-hardness in [Got92] and the reductions used in the proofs of Lemmas 4.1.2 and 4.1.6 to 4.1.9 all map their given input to consistent B-default theories. The lower bounds hence carry over to $\text{EXT}'(B)$. Similarly, the upper bounds continue to hold as apparent from Corollary 3.1.5: test whether the given knowledge base is satisfiable, and if so, verify the existence of a stable extension as above. □

4.2 Existence of Stable Expansions

Our main result in this section is the following theorem, that summarizes the complexity of the expansion existence problem for all finite sets B of Boolean functions. The complexity of the problem substantially differs from that of $\text{EXT}(B)$: $\text{EXP}(B)$ remains Σ_2^{p}-complete for all B such that $[B \cup \{0,1\}]$ includes the Boolean functions $\wedge$ and $\vee$. If only $\vee$ is contained in $[B \cup \{0,1\}]$, then the complexity drops to completeness for NP. On the other hand, if only $\wedge$ is contained in $[B \cup \{0,1\}]$, then the complexity drops to AC^0. In case of an affine B, the complexity drops to membership in P or completeness for $\mathrm{AC}^0[2]$, depending on whether B contains connectives of arity > 1. Altogether the complexity can thus be divided into five complexity degrees. See also Figure 4.2 on page 54.

Moreover, notice that in the upper part the complexity of $\text{EXP}(B)$ is symmetric with respect to the duality implicit in Post's lattice. Unlike default logic, autoepistemic logic allows for both positive and negative introspection; therefore, no simplifications emerge for sets of 1-reproducing or sets of monotone autoepistemic formulae.

Theorem 4.2.1 *Let B be a finite set of Boolean functions. Then $\text{EXP}(B)$ is*

1. Σ_2^p*-complete if* $\mathsf{D}_2 \subseteq [B]$ *or* $\mathsf{S}_{00} \subseteq [B]$ *or* $\mathsf{S}_{10} \subseteq [B]$*,*
2. NP*-complete if* $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$*,*
3. $\oplus$L*-hard and contained in* P *if* $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$*,*
4. $\mathrm{AC}^0[2]$*-complete if* $\mathsf{N}_2 \subseteq [B] \subseteq \mathsf{N}$*, and*
5. *in* AC^0 *in all other cases (that is, if* $[B] \subseteq \mathsf{E}$*),*

with respect to constant-depth reductions.

The proof of Theorem 4.2.1 will be established from the lemmas in this section. To begin with, observe that we may without loss of generality assume the availability of the Boolean constants.

Lemma 4.2.2 $\textsc{Exp}(B) \equiv_{\mathrm{cd}} \textsc{Exp}(B \cup \{0,1\})$ *for all finite sets B of Boolean functions.*

Proof. Let B be a finite set of Boolean functions and $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$. For the non-trivial direction $\textsc{Exp}(B \cup \{0,1\}) \leq \textsc{Exp}(B)$, we map Σ to $\Sigma' := \Sigma_{[1/t, 0/Lf]} \cup \{t\}$, where t and f are fresh propositions. Then the stable expansions of Σ' and Σ are in one-to-one correspondence, as any expansion of Σ' includes t and $\neg Lf$. $\square$

As a consequence of Lemma 4.2.2, it suffices to consider sets B such that $\{0,1\} \in [B]$. This is equivalent to requiring that B is a base for one of the clones I, N, V, E, L, M, or BF (see Table 2.1 on page 21). Provided that the necessary Boolean functions can be efficiently implemented in the respective sets B, all other cases follow from these seven clones. Before we start proving our classification, we give one further observation.

Lemma 4.2.3 *For all finite sets B of Boolean functions and every set* $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$, $\mathcal{L}_{\mathrm{ae}}$ *is a stable expansion of* Σ *if and only if* $\Sigma \cup \mathrm{SF}^L(\Sigma)$ *is inconsistent.*

Proof. Suppose that $\mathcal{L}_{\mathrm{ae}}$ is a stable expansion of Σ and let Λ denote its kernel. Then $\Sigma \cup \Lambda \models_L 0$ by virtue of Lemma 3.2.8. As $\Sigma \cup \Lambda \models_L 0$ if and only if $\Sigma \cup \Lambda \models 0$, we obtain $\Lambda = \mathrm{SF}^L(\Sigma)$ (notice that $\{L\chi \mid L\chi \in \mathrm{SF}^q(0)\} = \emptyset$, see Definition 3.2.7). In conclusion, $\Sigma \cup \mathrm{SF}^L(\Sigma)$ must be inconsistent. Conversely suppose that $\Sigma \cup \mathrm{SF}^L(\Sigma)$ is inconsistent. Then so is $\mathrm{Th}(\Sigma \cup L(\mathcal{L}_{\mathrm{ae}}))$. Consequently, $\mathcal{L}_{\mathrm{ae}}$ is a stable expansion of Σ. $\square$

Lemma 4.2.4 *Let B be a finite set of Boolean functions such that* $\mathsf{M} \subseteq [B]$*. Then* $\textsc{Exp}(B)$ *is* Σ_2^p*-complete with respect to constant-depth reductions.*

Proof. Let B be a finite set of Boolean functions as required. We have to prove hardness, membership in Σ_2^p follows from Theorem 3.2.4.

We reduce from $\textsc{Qbf}_{\exists,2}$. Let $\varphi := \exists x_1 \cdots \exists x_n \forall y_1 \cdots \forall y_m \psi$ be a quantified Boolean formula in disjunctive normal form. In [Got92], Gottlob shows that φ is valid if and only if the set $\Sigma := \{L\psi, Lx_1 \leftrightarrow x_1, \ldots, Lx_n \leftrightarrow x_n\}$ has a stable expansion. The idea of our proof is to modify the given reduction to use monotone connectives only, thus proving that $\textsc{Exp}(B)$ is Σ_2^p-hard for every finite set $\mathsf{M} \subseteq [B]$. More precisely, we define

$$\psi' := \psi_{[\neg x_1/x'_1,\dots,\neg x_n/x'_n,\neg y_1/y'_1,\dots,\neg y_m/y'_m]}$$

and

$$\Sigma' := \{L\psi'\} \cup \{y_j \vee y'_j \mid 1 \leq j \leq m\} \cup \{x_i \vee Lx'_i, Lx_i \vee x'_i \mid 1 \leq i \leq n\}.$$

Clearly, $\Sigma' \subseteq \mathcal{L}_{\mathrm{ae}}(\{\wedge,\vee\})$. Moreover, for every $1 \leq i \leq n$, we have that any stable expansion of Σ contains either Lx_i or Lx'_i but not both: assume that Λ is a Σ'-full set such that $Lx_i \in \Lambda$ and $Lx'_i \in \Lambda$. Then, by definition of Σ', $\Sigma' \cup \Lambda \not\models x_i$ and $\Sigma' \cup \Lambda \not\models x'_i$, although $Lx_i, Lx'_i \in \Lambda$; a contradiction to Λ being Σ'-full. Otherwise, if Λ were a Σ'-full set such that $\neg Lx_i \in \Lambda$ and $\neg Lx'_i \in \Lambda$, then $\Sigma' \cup \Lambda \models x_i$ and $\Sigma' \cup \Lambda \models x'_i$, a contradiction to Λ being Σ'-full, because $Lx_i, Lx'_i \notin \Lambda$. In conclusion, any Σ'-full set and equivalently any stable expansion contains either Lx_i or Lx'_i but not both.

We show that Σ' has a stable expansion if and only if φ is valid. First suppose that Σ' has a stable expansion Δ. Let Λ denote its kernel. As $\Sigma' \cup \mathrm{SF}^L(\Sigma')$ is consistent, we obtain that $\Delta \neq \mathcal{L}_{\mathrm{ae}}$ from Lemma 4.2.3. By the argument above, either $Lx_i \in \Delta$ or $Lx'_i \in \Delta$, but not both. Moreover, $L\psi' \in \Delta$, whence ψ' must be derivable from $\Sigma' \cup \Lambda$ by Definition 3.2.5. Note that this implies that ψ' is satisfied by all assignments setting either y_i or y'_i to 1; in particular, by all assignments that assign a complementary value to y_i and y'_i for every i. Define a truth assignment $\sigma\colon \{x_i \mid 1 \leq i \leq n\} \to \{0,1\}$ from Λ such that $\sigma(x_i) := 1$ if $Lx_i \in \Lambda$, and $\sigma(x_i) := 0$ otherwise. It follows that $\sigma \models \forall y_1 \cdots \forall y_m \psi$, thus φ is valid.

Now suppose that φ is valid. Then there exists an assignment $\sigma\colon \{x_i \mid 1 \leq i \leq n\} \to \{0,1\}$ such that any extension of σ to $y_1,\dots,y_m$ satisfies ψ. Let $\Lambda := \{Lx_i, \neg Lx'_i \mid \sigma(x_i) = 1\} \cup \{\neg Lx_i, Lx'_i \mid \sigma(x_i) = 0\} \cup \{L\psi'\}$. We claim that Λ is Σ'-full. If $Lx_i \in \Lambda$, then $\neg Lx'_i \in \Lambda$; hence $\{Lx'_i \vee x_i, \neg Lx'_i\}$ implies x_i. Conversely, if $\Sigma' \cup \Lambda \models x_i$ then $\neg Lx'_i$ has to be in Λ, because x_i occurs in $L\psi'$ and the clause $Lx'_i \vee x_i$ only. From this, we obtain $Lx_i \in \Lambda$. Therefore, $\Sigma' \cup \Lambda \models x_i$ if and only if $Lx_i \in \Lambda$. From the definition of Λ now follows that $\Sigma' \cup \Lambda \not\models x_i$ if and only if $\neg Lx_i \in \Lambda$. The same holds for x'_i for each i. Due to the construction of Λ, the fact that the clause $y_i \vee y'_i$ enforces y'_i to be assigned a value equal to or bigger than the one assigned to $\neg y_i$, the definition of ψ' and its monotonicity, we also have $\Sigma' \cup \Lambda \models \psi'$. Hence, following Definition 3.2.5, Λ is a Σ'-full set and, by Lemma 3.2.6, Σ' has a stable expansion.

It remains to show that $\wedge$ and $\vee$ are efficiently implementable in any finite set B such that $\mathsf{M} \subseteq [B]$. As $\mathsf{M} \subseteq [B]$ only if $[B] = \mathsf{M}$ or $[B] = \mathsf{BF}$, this follows from Lemma 2.4.2. □

We cannot transfer the above result to $\textsc{Exp}(B)$ for $[B] = \mathsf{V}$, because we may not assume ψ to be in conjunctive normal form. But, using a similar idea, we can show that the problem is NP-complete.

Lemma 4.2.5 *Let B be a finite set of Boolean functions such that $[B] = \mathsf{V}$. Then* $\text{Exp}(B)$ *is* NP*-complete with respect to constant-depth reductions.*

Proof. We first show that $\text{Exp}(B)$ is efficiently verifiable, thus proving membership in NP. Let B be a finite set of Boolean functions such that $[B] = \mathsf{V}$. Given a set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ and a candidate Λ for a Σ-full set, substitute $L\varphi$ by the Boolean value assigned to by Λ and call the resulting set Σ'. Note that Σ' is still equivalent to a set of disjunctions and that $\text{Imp}(B) \in \mathrm{P}$ by Theorem 2.5.1. Therefore, the conditions $\Sigma' \models \varphi$ if and only if $L\varphi \in \Lambda$, and $\Sigma' \not\models \varphi$ if and only if $\neg L\varphi \in \Lambda$ can be verified in polynomial time.

To show NP-hardness, we reduce 3SAT to $\text{Exp}(B)$ as follows. Let $\varphi := \bigwedge_{1 \leq i \leq n} c_i$ with clauses $c_i = \ell_{i1} \vee \ell_{i2} \vee \ell_{i3}$, $1 \leq i \leq n$, be given and let $x_1, \ldots, x_m$ enumerate the propositions occurring in φ. From φ we construct the set

$$\Sigma := \{Lc'_i \mid 1 \leq i \leq n\} \cup \{x_i \vee Lx'_i, Lx_i \vee x'_i \mid 1 \leq i \leq m\},$$

where $c'_i = c_i[\neg x_1/x'_1, \ldots, \neg x_m/x'_m]$ for $1 \leq i \leq n$. Analogously to Lemma 4.2.4, we obtain that for any stable expansion Δ of Σ either $x_i \in \Delta$ or $x'_i \in \Delta$, but not both. We claim that φ is satisfiable if and only if Σ has a stable expansion.

First, suppose that Δ is a stable expansion of Σ. It is easily observed that $\Sigma \cup \mathrm{SF}^L(\Sigma)$ is consistent, therefore $\Delta \neq \mathcal{L}_{\mathrm{ae}}$. Let Λ be the kernel of Δ. As $\Delta \neq \mathcal{L}_{\mathrm{ae}}$ and $Lc'_i \in \Sigma$ for all $1 \leq i \leq n$, Definition 3.2.5 implies that $\Sigma \cup \Lambda \models c'_i$ and hence $\Sigma \cup \Lambda \models_L c_i$ for all $1 \leq i \leq n$. We thus conclude from Lemma 3.2.8 that $\varphi \in \Delta$.

Conversely, suppose that φ is satisfied by the assignment σ. Define the set $\Lambda := \{Lx_i, \neg Lx'_i \mid \sigma(x_i) = 1\} \cup \{Lx'_i, \neg Lx_i \mid \sigma(x_i) = 0\} \cup \{Lc'_i \mid 1 \leq i \leq n\}$. As $\sigma \models c_i$ for any $1 \leq i \leq n$, we obtain that $\Sigma \cup \Lambda \models c'_i$. Concluding, Λ is a Σ-full set. $\square$

Next, we turn to the case $[B] = \mathsf{L}$. Say that an L-prefixed formula is *L-atomic* if it is of the form $L\varphi$ for some atomic formula φ.

Lemma 4.2.6 *Let $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(\{\oplus, 1\})$. If $\mathrm{SF}^L(\Sigma)$ contains only L-atomic formulae, then one can decide in polynomial time whether Σ has a stable expansion.*

Proof. The idea is to use Gaussian elimination twice. Let Σ be as required and suppose that Σ consists of m formulae. Then the set Σ can be seen as a system of linear equations and thus written as $A\mathbf{x} = B\mathbf{y} + C$, where $\mathbf{x} = (x_1, \ldots, x_n)^{\mathrm{T}}$, $\mathbf{y} = (Lx_1, \ldots, Lx_n)^{\mathrm{T}}$, A and B are Boolean matrices having m rows, and C is a Boolean vector.

By applying Gaussian elimination to A we obtain an equivalent system $A'\mathbf{x} = B'\mathbf{y} + C'$ with an upper triangular matrix A'. Let r denote the number of free variables in $A'\mathbf{x}$ and suppose without loss of generality that these are $x_1, \ldots, x_r$. By subsequently eliminating the variables $x_{r+1}, \ldots, x_n$, we arrive at a system T

equivalent to Σ of the form

$$\{x_i = f_i(x_1,\ldots,x_r) + g_i(Lx_1,\ldots,Lx_n) + c_i \mid r<i\leq n\} \cup \\ \{0 = g_i(Lx_1,\ldots,Lx_n) + c_i \mid n<i\leq m+r\},$$

where for each i the functions f_i and g_i are linear.

Observe that $\Sigma \cup \mathrm{SF}^L(\Sigma)$ is inconsistent if and only if $T[Lx_1/1,\ldots Lx_n/1]$ has no solution. In this case Σ has $\mathcal{L}_{\mathrm{ae}}$ as a stable expansion. Hence assume that $\Sigma \cup \mathrm{SF}^L(\Sigma)$ is consistent. We will now show how to construct a Σ-full set Λ.

Since the variables $x_1,\ldots,x_r$ are free, they cannot be derived from $\Sigma \cup \Lambda$ whatever Λ is. The same occurs for every $i \geq r+1$ such that $f_i(x_1,\ldots,x_r)$ is not a constant function. Suppose this is the case for $r+1 \leq i \leq s$. Then any Σ-full set has to contain $\neg Lx_j$ for $1 \leq j \leq s$. Let T' be the system obtained by considering all remaining equations while replacing Lx_i with 0 for each $1 \leq i \leq s$. For each equation in T', the function f_i (if present) is a constant function ε_i. Therefore T' consists of the following equations:

$$\{x_i = \varepsilon_i + g_i'(Lx_{s+1},\ldots,Lx_n) + c_i \mid s < i \leq n\} \cup \\ \{0 = g_i'(Lx_{s+1},\ldots,Lx_n) + c_i \mid n < i \leq m+r\}$$

with $g_i'(Lx_{s+1},\ldots,Lx_n) := g_i(0,\ldots,0,Lx_{s+1},\ldots,Lx_n)$ for $s < i \leq m+r$. For every $\Lambda \subseteq \mathrm{SF}^L(\Sigma) \cup \neg\mathrm{SF}^L(\Sigma)$ such that $\{\neg Lx_1,\ldots,\neg Lx_s\} \subseteq \Lambda$, and every i, $\Sigma \cup \Lambda \models x_i$ (respectively $\Sigma \cup \Lambda \not\models x_i$) if and only if $T' \cup \Lambda \models x_i$ (respectively $T' \cup \Lambda \not\models x_i$).

Claim. Let I and J form a partition of $\{s+1,\ldots,n\}$. Then $(Lx_{s+1},\ldots,Lx_n)$ with $Lx_i = 0$ if $i \in I$ and $Lx_j = 1$ if $j \in J$ is a solution of the system $T'[x_{s+1}/Lx_{s+1},\ldots, x_n/Lx_n]$ if and only if $\Lambda = \{\neg Lx_1,\ldots,\neg Lx_s\} \cup \{\neg Lx_i \mid i \in I\} \cup \{Lx_j \mid j \in J\}$ is a Σ-full set.

To prove the claim, let $\Lambda = \{\neg Lx_1,\ldots,\neg Lx_s\} \cup \{\neg Lx_i \mid i \in I\} \cup \{Lx_j \mid j \in J\}$ be a Σ-full set. Observe that $\Sigma \cup \Lambda$ is consistent and that either $T' \cup \Lambda \models x_i$ or $T' \cup \Lambda \models \neg x_i$, for all $1 \leq i \leq n$. Denote by λ the truth assignment induced by Λ on $\mathrm{SF}^L(\Sigma)$. Then, for every $i > s$, $Lx_i \in \Lambda$ if and only if $\lambda(Lx_i) = 1$ if and only if $T' \cup \Lambda \models x_i$ if and only if $\varepsilon_i + g_i'\big(\lambda(Lx_{s+1}),\ldots,\lambda(Lx_n)\big) + c_i = 1$; and $\neg Lx_i \in \Lambda$ if and only if $\lambda(Lx_i) = 0$ if and only if $T' \cup \Lambda \models \neg x_i$ if and only if $\varepsilon_i + g_i'\big(\lambda(Lx_{s+1}),\ldots,\lambda(Lx_n)\big) + c_i = 0$. This means that for every i, we have $\varepsilon_i + g_i'\big(\lambda(Lx_{s+1}),\ldots,\lambda(Lx_n)\big) + c_i = \lambda(Lx_i)$. Therefore λ is a solution of the system $\{Lx_i = \varepsilon_i + g_i(0,\ldots,0,\, Lx_{s+1},\ldots,Lx_n) + c_i \mid s < i \leq n\}$, and hence of the system $T'[x_{s+1}/Lx_{s+1},\ldots,x_n/Lx_n]$.

Conversely, suppose that λ is a solution of $T'[x_{s+1}/Lx_{s+1},\ldots,x_n/Lx_n]$. In particular, λ satisfies

$$\{Lx_i = \varepsilon_i + g_i(0,\ldots,0,Lx_{s+1},\ldots,Lx_n) + c_i \mid s+1 \leq i \leq n\}. \tag{4.3}$$

Set $\Lambda := \{\neg Lx_1, \dots, \neg Lx_s\} \cup \{\neg Lx_i \mid s+1 \leq i \leq n, \lambda(x_i) = 0\} \cup \{Lx_i \mid s+1 \leq i \leq n, \lambda(x_i) = 1\}$. Then $T' \cup \Lambda$ is equivalent to

$$\{x_i = \varepsilon_i + g_i'\big(\lambda(Lx_{s+1}), \dots, \lambda(Lx_n)\big) + c_i \mid s < i \leq n\} \cup \\ \{0 = g_i'\big(\lambda(Lx_{s+1}), \dots, \lambda(Lx_n)\big) + c_i \mid n < i \leq m + r\}.$$

Therefore, by (4.3), $T' \cup \Lambda \models x_i$ if and only if $\lambda(Lx_i) = 1$ and $T' \cup \Lambda \models \neg x_i$ if and only if $\lambda(Lx_i) = 0$. Hence, Λ is a Σ-full set. This proves the claim.

We conclude that Σ has a stable expansion if and only if $T'[x_{s+1}/Lx_{s+1}, \dots, x_n/Lx_n]$ has a solution. □

Remark 4.2.7 *Note that solving this last system by Gaussian elimination also gives the total number of possible Σ-full sets: the number of consistent stable expansions is equal to the number of solutions of $T'[x_{s+1}/Lx_{s+1}, \dots, x_n/Lx_n]$; testing for the inconsistent stable expansion can be accomplished using Lemma 4.2.3 and the fact that the satisfiability of sets of affine formulae is polynomial-time decidable.*

Lemma 4.2.8 *Let B be a finite set of Boolean functions such that $[B] = \mathsf{L}$. Then* $\text{E}\scriptstyle\text{XP}$$(B)$ *is contained in* P *and* $\oplus$L*-hard with respect to constant-depth reductions.*

Proof. Let B be as required and Σ be a set of autoepistemic B-formulae. Then Σ can be written in polynomial time as a set of $\{\oplus\}$-formulae (see the proof of Lemma 2.5.5). We transform this set to Σ' as follows: introduce a fresh variable y_φ for every non-atomic formula φ such that $L\varphi \in \Sigma$; add the equation $y_\varphi \leftrightarrow \varphi$; and replace all occurrences of $L\varphi$ by Ly_φ. We claim that the Σ-full sets and the Σ'-full sets are in one-to-one correspondence. This establishes the upper bound, because Σ' satisfies the conditions of Lemma 4.2.6.

To prove our claim, let $\Lambda \subseteq \mathrm{SF}^L(\Sigma) \cup \neg\mathrm{SF}^L(\Sigma)$. We give an inductive argument on the number of L-prefixed formulae in Σ. To this end, choose an $L\varphi \in \mathrm{SF}^L(\Sigma)$ such that φ does not contain any L-prefixed subformulae. Define

$$\Sigma_\varphi := \Sigma_{[L\varphi/Ly_\varphi]} \cup \{y_\varphi \leftrightarrow \varphi\},$$
$$\Lambda_\varphi := (\Lambda \setminus \{L\varphi, \neg L\varphi\}) \cup \{Ly_\varphi \mid L\varphi \in \Lambda\} \cup \{\neg Ly_\varphi \mid \neg L\varphi \in \Lambda\}.$$

That is, Σ_φ differs from Σ in that we substituted a non-L-atomic subformula $L\varphi$ by Ly_φ and added the equation $y_\varphi \leftrightarrow \varphi$.

Observe that $\Sigma \cup \Lambda \models \varphi$ if and only if $\Sigma_\varphi \cup \Lambda_\varphi \models y_\varphi$. Therefore, since $L\varphi \in \Lambda$ if and only if $Ly_\varphi \in \Lambda_\varphi$, and $\neg L\varphi \in \Lambda$ if and only if $\neg Ly_\varphi \in \Lambda_\varphi$, it holds that Λ is Σ-full if and only if Λ_φ is Σ_φ-full. Repeating the above argument eventually yields Σ', for which the existence of stable expansions can be tested in polynomial time by Lemma 4.2.6.

It hence remains to establish $\oplus$L-hardness. We give a reduction from $\text{I}\scriptstyle\text{MP}$$(B)$ for $[B \cup \{0,1\}] = \mathsf{L}$. Given an instance (Γ, ψ) of $\text{I}\scriptstyle\text{MP}$$(B)$, let $\Sigma := \Gamma \cup \{L\psi\}$. Indeed, if $\Gamma \models \psi$, then $\Lambda := \{L\psi\}$ is Σ-full; and if $\Lambda := \{L\psi\}$ is Σ-full, then $\Gamma \models \psi$. Thus, $\text{I}\scriptstyle\text{MP}$$(B) \leq_{\mathrm{cd}}$ $\text{E}\scriptstyle\text{XP}$$(B)$ via the mapping $(\Gamma, \psi) \mapsto \Sigma$. □

Lemma 4.2.9 *Let B be a finite set of Boolean functions such that $[B] = \mathsf{N}$. Then* $\textsc{Exp}(B)$ *is* $\mathrm{AC}^0[2]$*-complete with respect to constant-depth reductions. It moreover holds that, for every set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$, there is at most one consistent stable expansion.*

Proof. Let B be a finite set of Boolean functions such that $[B] = \mathsf{N}$ and let $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ be given. Denote by Σ' the set of formulae obtained from the $\{\neg\}$-representation of Σ by repeatedly eliminating all occurrences of $\neg\neg$, replacing all occurrences of LL by L, and replacing all occurrences of $L\neg L$ by $\neg L$. Then Σ' is a set of literals and formulae of the form $L\ell$ or $\neg L\ell$, where ℓ is again a literal. As $LL\varphi$ is true if and only if $L\varphi$ is true, and $L\neg L\varphi$ is true if and only if $L\varphi \vee L0$ is true, it holds that Σ has a consistent stable expansion if and only if $\varphi \in \Sigma'$ for all $L\varphi \in \Sigma'$ and $\varphi \notin \Sigma'$ for all $\neg L\varphi \in \Sigma'$. This can be tested using an AC^0-circuit. On the other hand, existence of the inconsistent stable expansion can be tested using Lemma 4.2.3, which requires an AC^0-circuit with oracle gates for B-formula evaluation. As B-formulae can be evaluated in $\mathrm{AC}^0[2]$ [Sch10] and Σ' can be constructed from Σ using an AC^0-circuit with oracle gates for B-formula evaluation, we conclude that $\textsc{Exp}(B) \in \mathrm{AC}^0[2]$.

As for the $\mathrm{AC}^0[2]$-completeness, observe that $\textsc{Imp}(B)$ is $\mathrm{AC}^0[2]$-complete and that $\textsc{Imp}(B) \leq_{\mathrm{cd}} \textsc{Exp}(B)$ via the reduction $(\Gamma, \psi) \mapsto \Sigma := \Gamma \cup \{L\psi\}$. □

Lemma 4.2.10 *Let B be a finite set of Boolean functions such that $[B] \subseteq \mathsf{E}$. Then* $\textsc{Exp}(B)$ *is solvable in* AC^0*. It moreover holds that, for every set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$, there is at most one consistent stable expansion.*

Proof. As any given set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ is equivalent to a set of propositions, the result follows from the proof of Lemma 4.2.9 together with the fact that B-formula evaluation can be performed in AC^0 (see the proof of Lemma 2.5.7). □

Proof of Theorem 4.2.1. According to Lemma 4.2.2, $\textsc{Exp}(B) \equiv \textsc{Exp}(B \cup \{0,1\})$. Note that $[\mathsf{D}_1 \cup \{0,1\}] = [\mathsf{S}_{02} \cup \{0,1\}] = [\mathsf{S}_{12} \cup \{0,1\}] = \mathsf{BF}$ and $[\mathsf{D}_2 \cup \{0,1\}] = [\mathsf{S}_{00} \cup \{0,1\}] = [\mathsf{S}_{10} \cup \{0,1\}] = \mathsf{M}$. Moreover, if $[B] \subseteq \mathsf{V}$ or $[B] \subseteq \mathsf{L}$ or $[B] \subseteq \mathsf{E}$, then either $[B \cup \{0,1\}] \in \{\mathsf{V}, \mathsf{L}, \mathsf{N}\}$ or $[B] \subseteq \mathsf{E}$. Therefore, Lemmas 4.2.4, 4.2.5 and 4.2.8 to 4.2.10 cover all cases of the theorem. □

In many applications, one is interested in the consistent stable expansions of an autoepistemic theory only. Denote this alternative formalization of the expansion existence problem as $\textsc{Exp}'(B)$:

Problem: $\textsc{Exp}'(B)$
Input: A set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$
Question: Does Σ have a consistent stable expansion?

From Theorem 4.2.1 and its proof one can easily settle the complexity of $\textsc{Exp}'(B)$ for all finite sets B of Boolean functions:

Corollary 4.2.11 $\textsc{Exp}(B) \equiv \textsc{Exp}'(B)$ *for all finite sets B of Boolean functions,* *Proof.* The corollary follows immediately from the proof of Theorem 4.2.1. Indeed, in each hardness proof (see Lemmas 4.2.4, 4.2.5 and 4.2.8) we have shown that the set of autoepistemic B-formulae constructed in that proof, Σ or Σ', does not admit $\mathcal{L}_{\text{ae}}$ as a stable expansion. Therefore, Σ or Σ' have a stable expansion if and only if Σ or Σ' have a consistent stable expansion. This proves all the hardness results. As for the upper bounds, Lemmas 4.2.4 and 4.2.5 are easily seen to extend to the existence of a consistent stable expansion. And for the tractable cases $[B] \subseteq \mathsf{E}$ and $[B] \subseteq \mathsf{N}$, one can decide the existence of a consistent stable expansion in $\mathrm{AC}^0[2]$. This follows from the proof of Lemmas 4.2.9 and 4.2.10. Finally, for $[B] \subseteq \mathsf{L}$, observe that the proof of Lemma 4.2.8 actually allows to compute full sets corresponding to consistent stable expansions in polynomial time. $\square$

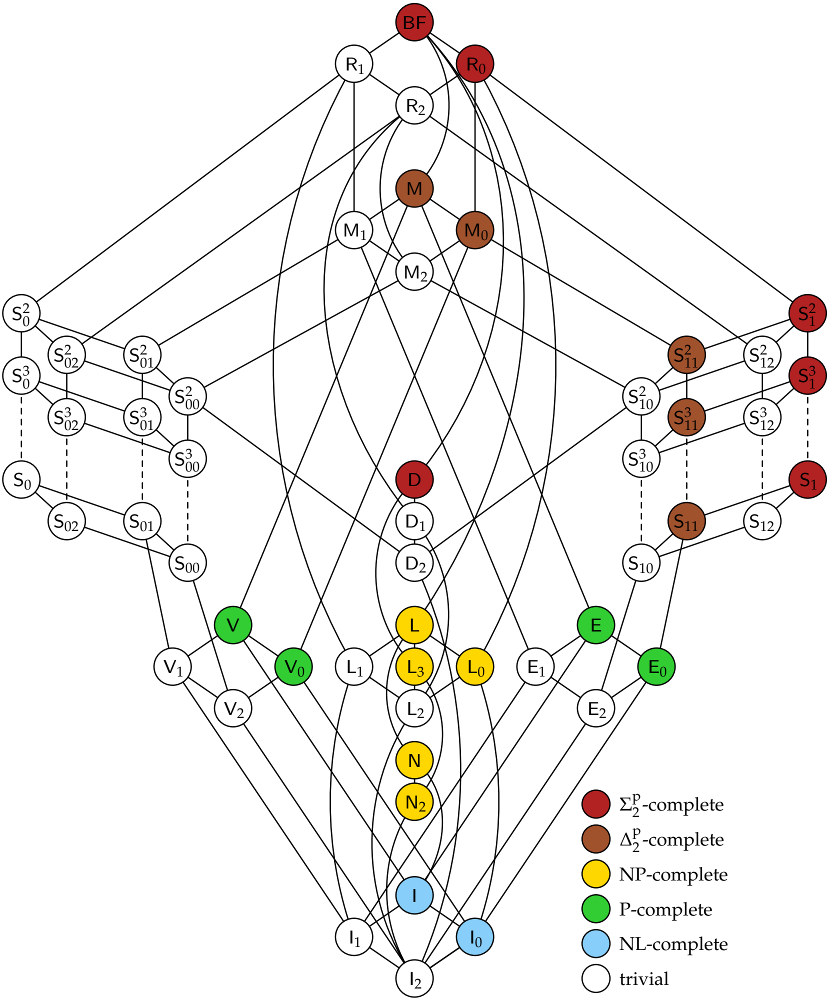

Figure 4.1: The complexity of $\textsc{Ext}(B)$

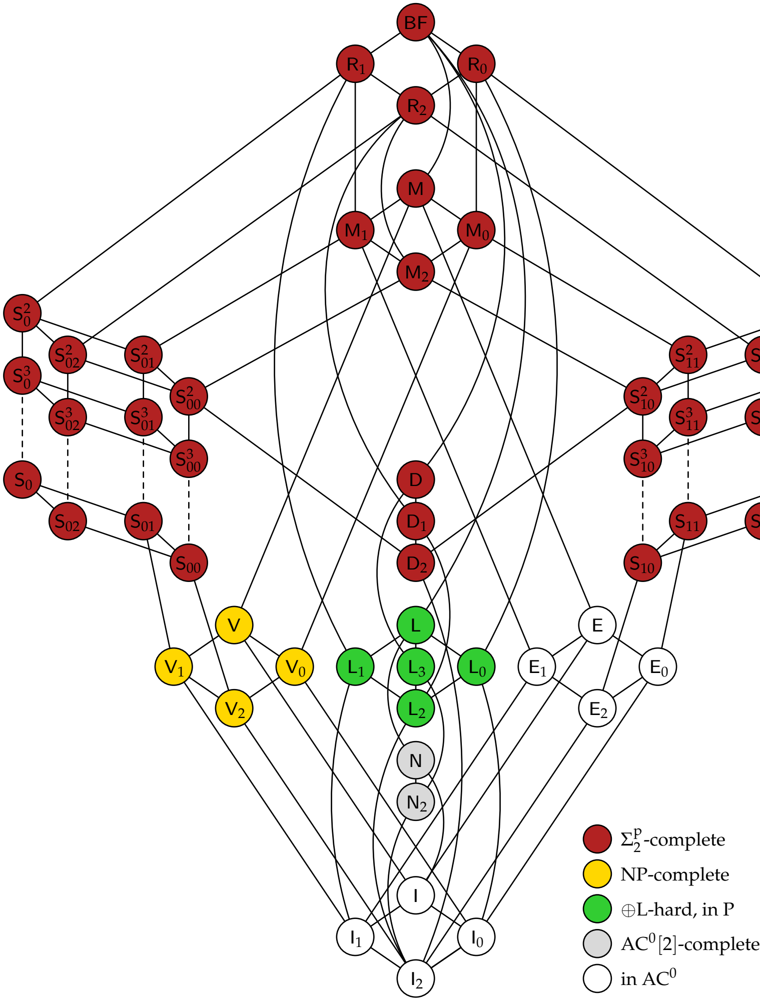

Figure 4.2: The complexity of $\textsc{Exp}(B)$

Chapter 5

Reasoning

This chapter analyzes the computational complexity of credulous and skeptical reasoning in the nonmonotonic logic introduced in Chapter 3. This task, to decide whether a given statement can be derived from a given knowledge base, is central to knowledge representation systems. Here we extend the know complexity results to all fragments obtained from restrictions on the set of available Boolean functions.

We will stick to the order of the previous chapter and study the computational complexity of credulous and skeptical reasoning in default logic first, proceed with autoepistemic logic, and finally analyze the complexity of skeptical reasoning in circumscription.

5.1 Reasoning in Default Logic

We will first analyze the credulous reasoning problem and the skeptical reasoning problem in default logic. For these problems, there are two sources of complexity. On the one hand, we need to determine a candidate for a stable extension. On the other hand, we have to verify that this candidate is indeed a finite characterization of some stable extension—a task that requires to test for formula implication. Depending on the Boolean connectives allowed, one or both tasks can be performed in polynomial time or even become trivial. In principle, this yields five possible cases for the complexity of $\textsc{Cred}_{\mathrm{DL}}(B)$, namely the classes of the polynomial hierarchy below and including Σ_2^{p}. We will see that all five cases actually occur, with the easy case splitting further into two sub-cases.

We obtain the full complexity of $\textsc{Cred}_{\mathrm{DL}}(B)$, that is, Σ_2^{p}-completeness, for all clones B where both problems $\textsc{Ext}(B)$ and $\textsc{Imp}(B)$ attain their highest complexity (compare Theorems 2.5.1 and 4.1.1). The complexity reduces to Δ_2^{p} for clones that allow for an efficient computation of stable extensions but whose implication problem remains coNP-complete. More precisely, the problem is complete for this class if a stable extension may not exist ($\mathsf{S}_{11} \subseteq [B] \subseteq \mathsf{M}$) and becomes coNP-complete otherwise ($X \subseteq [B] \subseteq \mathsf{R}_1$ for $X \in \{\mathsf{S}_{00}, \mathsf{S}_{10}, \mathsf{D}_2\}$). Conversely, if the implication problem becomes easy but determining an extension candidate is hard, then $\textsc{Cred}_{\mathrm{DL}}(B)$ is NP-complete, while the dual reasoning problem $\textsc{Skep}_{\mathrm{DL}}(B)$ has to test for all extensions and is coNP-complete. This is the case for the clones $[B] \in \{\mathsf{N}, \mathsf{N}_2, \mathsf{L}, \mathsf{L}_0, \mathsf{L}_3\}$. Finally, for clones B that allow for solving both tasks in polynomial time, both $\textsc{Cred}_{\mathrm{DL}}(B)$ and $\textsc{Skep}_{\mathrm{DL}}(B)$ are in P. The complete classification of $\textsc{Cred}_{\mathrm{DL}}(B)$ is given in the following theorem. It is also depicted in Figure 5.1 on page 69.

Theorem 5.1.1 *Let B be a finite set of Boolean functions. Then $\textsc{Cred}_{\mathrm{DL}}(B)$ is*

1. Σ_2^{p}*-complete if* $\mathsf{S}_1 \subseteq [B]$ *or* $\mathsf{D} \subseteq [B]$,
2. Δ_2^{p}*-complete if* $\mathsf{S}_{11} \subseteq [B] \subseteq \mathsf{M}$,
3. coNP*-complete if* $\mathsf{S}_{00} \subseteq [B] \subseteq \mathsf{R}_1$ *or* $\mathsf{S}_{10} \subseteq [B] \subseteq \mathsf{R}_1$ *or* $\mathsf{D}_2 \subseteq [B] \subseteq \mathsf{R}_1$,
4. NP*-complete if* $[B] \in \{\mathsf{N}, \mathsf{N}_2, \mathsf{L}, \mathsf{L}_0, \mathsf{L}_3\}$,
5. P*-complete if* $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$ *or* $\mathsf{E}_2 \subseteq [B] \subseteq \mathsf{E}$ *or* $[B] \in \{\mathsf{L}_1, \mathsf{L}_2\}$, *and*
6. NL*-complete in all other cases (that is, if* $[B] \subseteq \mathsf{I}$),

with respect to constant-depth reductions.

The proof of Theorem 5.1.1 follows from Theorem 3.1.7 and the upper and lower bounds given in Lemma 5.1.3 and Lemma 5.1.4 below. We start with a result analogous to Lemma 4.1.2 for both the credulous and the skeptical reasoning problem.

Lemma 5.1.2 $\textsc{Cred}_{\mathrm{DL}}(B) \equiv_{\mathrm{cd}} \textsc{Cred}_{\mathrm{DL}}(B \cup \{1\})$ *and* $\textsc{Skep}_{\mathrm{DL}}(B) \equiv_{\mathrm{cd}} \textsc{Skep}_{\mathrm{DL}}(B \cup \{1\})$ *for each finite set B of Boolean functions.*

Proof. Observe that the reduction $(W, D) \mapsto (W', D')$ given in Lemma 4.1.2 has the additional property that for each formula φ and each extension E of (W, D), $\varphi \in E$ if and only if $\varphi_{[1/t]} \in E_{[1/t]}$. □

Lemma 5.1.3 *Let B be a finite set of Boolean functions. Then $\textsc{Cred}_{\mathrm{DL}}(B)$ is contained*

1. *in* Δ_2^{p} *if* $[B] \subseteq \mathsf{M}$,
2. *in* coNP *if* $[B] \subseteq \mathsf{R}_1$,
3. *in* NP *if* $[B] \subseteq \mathsf{L}$,
4. *in* P *if* $[B] \subseteq \mathsf{V}$ *or* $[B] \subseteq \mathsf{E}$ *or* $[B] \subseteq \mathsf{L}_1$, *and*
5. *in* NL *if* $[B] \subseteq \mathsf{I}$.

Proof. Let B be a finite set of Boolean functions, let (W, D) be a B-default theory, and let $\varphi \in \mathcal{L}(B)$.

For $[B] \subseteq \mathsf{M}$, membership in Δ_2^{p} is obtained from a straightforward extension of Algorithm 4.1: We first iteratively compute the applicable defaults G while asserting that (W, D) has a stable extension using Algorithm 4.1, and eventually verify that φ is implied by W and the conclusions in G.

For $[B] \subseteq \mathsf{R}_1$, the justifications β are irrelevant for computing a stable extension, as for every default rule $\frac{\alpha:\beta}{\gamma} \in D$ we cannot derive $\neg\beta$ ($\neg\beta$ is not 1-reproducing). Thence, a unique consistent stable extension E is guaranteed to exist by Lemma 4.1.3. Using Algorithm 4.1 we can iteratively compute the generating defaults of E of the unique consistent stable extension of (W, D) and eventually check whether φ is implied by W and the conclusions in $\mathrm{GD}(E)$.

The described algorithm is a monotone Turing reduction from $\textsc{Cred}_{\mathrm{DL}}(B)$ to $\textsc{Imp}(B)$ in the sense that for any deterministic oracle Turing machine M that executes it, $A \subseteq A'$ implies that the language recognized by M with oracle A is a subset of the language recognized by M with oracle A'. As coNP is closed under monotone Turing reductions [Sel82], $\textsc{Cred}_{\mathrm{DL}}(B) \in \mathrm{coNP}$.

For $[B] \subseteq \mathsf{L}$, we proceed similarly as in the proof of Theorem 4.1.1 (3.). First, we guess a set G of generating defaults and subsequently verify that both $\mathrm{Th}\big(W \cup \{\gamma \mid \frac{\alpha:\beta}{\gamma} \in G\}\big)$ is a stable extension and that $W \cup \{\gamma \mid \frac{\alpha:\beta}{\gamma} \in G\} \models \varphi$. Using Theorem 2.5.1, both conditions may be verified in polynomial time.

For $[B] \subseteq \mathsf{V}$, $[B] \subseteq \mathsf{E}$, and $[B] \subseteq \mathsf{L}_1$, we proceed as for $[B] \subseteq \mathsf{M}$. However, for these types of B-formulae we have an efficient test for implication (see Theorem 2.5.1). Hence, $\textsc{Cred}_{\mathrm{DL}}(B) \in \mathrm{P}$.

For $[B] \subseteq \mathsf{I}$, observe that NL is closed under intersections. Hence, given a B-default theory (W, D) and a B-formula φ we can first test whether (W, D) has a stable extension E using Lemma 4.1.9 and subsequently assert that $\varphi \in E$ by reusing the graph $G_{(W,D)}$ constructed from (W, D): it holds that $\varphi \in E$ if and only if the node corresponding to φ is contained in $G_{(W,D)}$ and reachable from the node 1. Thus, $\textsc{Cred}_{\mathrm{DL}}(B) \in \mathrm{NL}$. □

We will now establish the lower bounds required to complete the proof of Theorem 5.1.1.

Lemma 5.1.4 *Let B be a finite set of Boolean functions. Then* $\textsc{Cred}_{\mathrm{DL}}(B)$ *is*

1. Σ_2^{P}-*hard if* $\mathsf{S}_1 \subseteq [B]$ *or* $\mathsf{D} \subseteq [B]$,
2. Δ_2^{P}-*hard if* $\mathsf{S}_{11} \subseteq [B]$,
3. coNP-*hard if* $\mathsf{S}_{00} \subseteq [B]$ *or* $\mathsf{S}_{10} \subseteq [B]$ *or* $\mathsf{D}_2 \subseteq [B]$,
4. NP-*hard if* $\mathsf{N}_2 \subseteq [B]$ *or* $\mathsf{L}_0 \subseteq [B]$,
5. P-*hard if* $\mathsf{V}_2 \subseteq [B]$, $\mathsf{E}_2 \subseteq [B]$ *or* $\mathsf{L}_2 \subseteq [B]$, *and*
6. NL-*hard in all other clones.*

Proof. The first part follows from Theorem 3.1.7 and Lemma 5.1.2.

For the second part, observe that the constant 1 is contained in any stable extension. The second part thus follows from Lemmas 4.1.6 and 5.1.2.

For $\mathsf{S}_{00} \subseteq [B]$, $\mathsf{S}_{10} \subseteq [B]$, and $\mathsf{D}_2 \subseteq [B]$, coNP-hardness is established by a constant-depth reduction from $\textsc{Imp}(B)$. Let $\Gamma \subseteq \mathcal{L}(B)$ and $\varphi \in \mathcal{L}(B)$. Then the default theory $(\Gamma, \emptyset)$ has the unique stable extension $\mathrm{Th}(\Gamma)$, and hence $\Gamma \models \varphi$ if and only if $((\Gamma, \emptyset), \varphi) \in \textsc{Cred}_{\mathrm{DL}}(B)$. Therefore, $\textsc{Imp}(B) \leq_{\mathrm{cd}} \textsc{Cred}_{\mathrm{DL}}(B)$, and the claim follows with Theorem 2.5.1.

For the fourth part, it suffices to prove NP-hardness for $\mathsf{N}_2 \subseteq [B]$; the case $\mathsf{L}_0 \subseteq [B]$ then follows from $\textsc{Cred}_{\mathrm{DL}}(\mathsf{N}_2) \leq_{\mathrm{cd}} \textsc{Cred}_{\mathrm{DL}}(\mathsf{L}) \equiv_{\mathrm{cd}} \textsc{Cred}_{\mathrm{DL}}(\mathsf{L}_0 \cup \{1\})$ and Lemma 5.1.2. We obtain the desired hardness result by adjusting the reduction given in the proof of Lemma 4.1.7. Consider the mapping $\varphi \mapsto ((\{\psi\}, D_\varphi), \psi)$, where D_φ is the set of default rules constructed from φ in Lemma 4.1.7, and ψ is a satisfiable B-formula such that φ and ψ do not use common variables. By Lemma 4.1.7, $\varphi \in 3\textsc{Sat}$ if and only if $(\{\psi\}, D_\varphi)$ has a stable extension. As any extension of $(\{\psi\}, D_\varphi)$ contains ψ, we obtain $3\textsc{Sat} \leq_{\mathrm{cd}} \textsc{Cred}_{\mathrm{DL}}(B)$ via the above reduction.

For the fifth part, it suffices to prove the P-hardness for $[B] \in \{\mathsf{L}_1, \mathsf{L}_2\}$. The cases $\mathsf{E}_2 \subseteq [B]$ and $\mathsf{V}_2 \subseteq [B]$ follow analogously to the second part from Lemmas 4.1.8 and 5.1.2. We again provide a reduction from $\textsc{HGap}$ restricted to hypergraphs whose edges contain at most two source nodes. To this end, we transform a given instance (H, S, t) to the $\textsc{Cred}_{\mathrm{DL}}(\{x \oplus y \oplus z, 1\})$-instance $((W, D), \varphi)$ with

$$\begin{aligned}
W &:= \{p_s \mid s \in S\},\\
D &:= \left\{ \frac{p_{\mathrm{src}(e)} : 1}{p_{\mathrm{dest}(e)}} \;\middle|\; e \in E, |\mathrm{src}(e)| = 1 \right\} \cup \\
&\quad \left\{ \frac{p_{\mathrm{src}_1(e)} : 1}{p_e}, \frac{p_{\mathrm{src}_2(e)} : 1}{p_e}, \frac{p_{\mathrm{src}_1(e)} \oplus p_{\mathrm{src}_2(e)} \oplus p_e : 1}{p_{\mathrm{dest}(e)}} \;\middle|\; e \in E, |\mathrm{src}(e)| = 2 \right\},\\
\varphi &:= p_t,
\end{aligned}$$

and $\{\mathrm{src}_1(e), \mathrm{src}_2(e)\}$ denote the source nodes of e. As for the correctness, observe that if for some $e \in E$ with $|\mathrm{src}(e)| = 2$ both variables $p_{\mathrm{src}_1(e)}$ and $p_{\mathrm{src}_2(e)}$ can be derived from the stable extension of (W, D), then p_e and consequently $p_{\mathrm{dest}(e)}$ can be derived. Conversely, if $\mathrm{src}_1(e)$ or $\mathrm{src}_2(e)$ cannot be derived, then either none or two of the propositions in $p_{\mathrm{src}_1(e)} \oplus p_{\mathrm{src}_2(e)} \oplus p_e$ are satisfied. Thus $p_{\mathrm{dest}(e)}$ cannot be derived from the defaults corresponding to e.

Finally, it remains to show NL-hardness for $\mathsf{I}_2 \subseteq [B]$. We give a constant-depth reduction from $\textsc{Gap}$ to $\textsc{Cred}_{\mathrm{DL}}(\{\mathrm{id}\})$ similar to that in the proof of Lemma 4.1.9. For a directed graph $G = (V, E)$ and two nodes $s, t \in V$, we transform the given $\textsc{Gap}$-instance (G, s, t) to $((W, D), \varphi)$, where

$$W := \{p_s\}, \;\; D := \left\{ \frac{p_u : p_u}{p_v} \;\middle|\; (u, v) \in E \right\}, \;\; \varphi := p_t.$$

Clearly, $(G, s, t) \in \textsc{Gap}$ if and only if φ is contained in all stable extensions of (W, D). □

This completes the proof of Theorem 5.1.1.

We will next classify the complexity of the skeptical reasoning problem. The analysis as well as the result are similar to the classification of the credulous reasoning problem.

Theorem 5.1.5 *Let B be a finite set of Boolean functions. Then* $\text{SKEP}_{\text{DL}}(B)$ *is*

1. Π_2^{p}*-complete if* $\mathsf{S}_1 \subseteq [B]$ *or* $\mathsf{D} \subseteq [B]$*,*
2. Δ_2^{p}*-complete if* $\mathsf{S}_{11} \subseteq [B] \subseteq \mathsf{M}$*,*
3. coNP*-complete if* $X \subseteq [B] \subseteq Y$ *for* $X \in \{\mathsf{S}_{00}, \mathsf{S}_{10}, \mathsf{D}_2, \mathsf{N}_2, \mathsf{L}_0\}$ *and* $Y \in \{\mathsf{R}_1, \mathsf{L}\}$*,*
4. P*-complete if* $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$ *or* $\mathsf{E}_2 \subseteq [B] \subseteq \mathsf{E}$ *or* $[B] \in \{\mathsf{L}_1, \mathsf{L}_2\}$*, and*
5. NL*-complete in all other cases (that is, if* $[B] \subseteq \mathsf{I}$*),*

with respect to constant-depth reductions.

Proof. The first part again follows from Theorem 3.1.7 and Lemma 5.1.2.

For $[B] \in \{\mathsf{N}, \mathsf{N}_2, \mathsf{L}, \mathsf{L}_0, \mathsf{L}_3\}$, we guess similarly as in Theorem 4.1.1 a set G of defaults and then verify in the same way whether W and G generate a stable extension E. If not, then we accept. Otherwise, we check if $E \models \varphi$ and answer according to this test. This yields a coNP-algorithm for $\text{SKEP}_{\text{DL}}(B)$. Hardness for coNP is achieved by modifying the reduction from Theorem 4.1.1 (see also the proof of Lemma 5.1.4): map φ to $((\emptyset, D_\varphi), \psi)$, where D_φ is defined as in the proof of Theorem 4.1.1, and ψ is a B-formula such that φ and ψ do not share variables. Then $\varphi \notin 3\text{SAT}$ if and only if $(\emptyset, D_\varphi)$ does not have a stable extension. The latter is true if and only if ψ is in all extensions of $(\emptyset, D_\varphi)$. Hence $\overline{3\text{SAT}} \leq_{\text{cd}} \text{SKEP}_{\text{DL}}(B)$, establishing the claim.

For all remaining clones B, it holds that $[B] \subseteq \mathsf{R}_1$ or $[B] \subseteq \mathsf{M}$. Hence, Corollary 4.1.5 and Theorem 5.1.1 imply the claimed results. □

5.2 Reasoning in Autoepistemic Logic

In this section, we analyze the credulous and the skeptical reasoning problem for autoepistemic logic. Similar to default logic, there are two sources for the complexity of these problems: we need to determine a candidate for a full set and to verify that this candidate is indeed the finite characterization of a stable expansion.

As determining whether an autoepistemic theory admits a stable expansions alone is Σ_2^{p}-complete for all clones above E, V, and L, both the credulous and skeptical reasoning problems remain complete for respectively Σ_2^{p} and Π_2^{p} for these clones, too. For the clones E, V, L and below, the implication problem and hence the task of verifying a candidate for a stable expansion become tractable. As a result, the complexity drops by at least one level of the polynomial hierarchy: we obtain respectively NP- or coNP-completeness if $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$ and polynomial-time solvable fragments if $[B] \subseteq \mathsf{L}$ or $[B] \subseteq \mathsf{E}$, where for the latter fragments even the set of candidates can be computed efficiently. The complete classification is established in Theorems 5.2.1 and 5.2.4 below and depicted in Figure 5.2 on 70.

Theorem 5.2.1 *Let B be a finite set of Boolean functions. Then* $\textsc{Cred}_{\text{AE}}(B)$ *is*

1. Σ_2^{p}*-complete if* $\mathsf{D}_2 \subseteq [B]$ *or* $\mathsf{S}_{00} \subseteq [B]$ *or* $\mathsf{S}_{10} \subseteq [B]$*,*
2. NP*-complete if* $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$*,*
3. $\oplus$L*-hard and contained in* P *if* $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$*,*
4. $\mathrm{AC}^0[2]$*-complete if* $\mathsf{N}_2 \subseteq [B] \subseteq \mathsf{N}$*, and*
5. *in* AC^0 *in all other cases (that is, if* $[B] \subseteq \mathsf{E}$*),*

with respect to constant-depth reductions.

The proof of Theorem 5.2.1 requires two auxiliary lemmas: the first shows that we may restrict our attention to the clones containing both Boolean constants; the second provides an upper bounds on the complexity of $\textsc{Cred}_{\text{AE}}(B)$ via reduction to the expansion existence problem.

Lemma 5.2.2 *For all finite sets B of Boolean functions, it holds that* $\textsc{Cred}_{\text{AE}}(B) \equiv_{\text{cd}} \textsc{Cred}_{\text{AE}}(B \cup \{0,1\})$ *and* $\textsc{Skep}_{\text{AE}}(B) \equiv_{\text{cd}} \textsc{Skep}_{\text{AE}}(B \cup \{0,1\})$*.*

Proof. The proof is analogous to the proof of Lemma 4.2.2. For the nontrivial directions, we map the given pair (Σ, φ) to (Σ', φ'), where $\Sigma' := \Sigma_{[1/t,0/Lf]} \cup \{t\}$, $\varphi' := \varphi_{[1/t,0/Lf]}$, and t and f are new propositions. Correctness of these reductions follows from the one-to-one correspondence of the stable expansions of Σ and the stable expansions of Σ'. ☐

Lemma 5.2.3 *For all finite sets B of Boolean functions, it holds that* $\textsc{Cred}_{\text{AE}}(B) \leq_{\text{cd}} \textsc{Exp}(B \cup \{\leftrightarrow\})$*.*

Proof. Let B be a finite set of Boolean functions. Given $\Sigma \subseteq \mathcal{L}_{\text{ae}}(B)$ and $\varphi \in \mathcal{L}_{\text{ae}}(B)$, map the pair (Σ, φ) to $\Sigma' := \Sigma \cup \{L\varphi \leftrightarrow p, Lp\}$, where p is a fresh proposition. We claim that φ is contained in a stable expansion of Σ if and only if $\Sigma' \in \textsc{Exp}(B)$.

First suppose that φ is contained in a stable expansion Δ of Σ. Let Λ denote the kernel of Δ. As $\Sigma \cup \Lambda \models_L \varphi$, there has to exist a set $\mathrm{SB}_{\Sigma\cup\Lambda}(\varphi) = \{L\chi \in \mathrm{SF}^q(\varphi) \mid \Sigma \cup \Lambda \models_L \chi\} \cup \{\neg L\chi \mid L\chi \in \mathrm{SF}^q(\varphi), \Sigma \cup \Lambda \not\models_L \chi\}$ such that $\Sigma \cup \Lambda \cup \mathrm{SB}_{\Sigma\cup\Lambda}(\varphi) \models \varphi$. We claim that $\Lambda' := \Lambda \cup \mathrm{SB}_{\Sigma\cup\Lambda}(\varphi) \cup \{L\varphi, Lp\}$ is Σ'-full:

- $\Sigma' \cup \Lambda' \models \varphi$, because $\Sigma \cup \Lambda \cup \mathrm{SB}_{\Sigma\cup\Lambda}(\varphi) \models \varphi$;
- $\Sigma' \cup \Lambda' \models p$, because $\Sigma \cup \{L\varphi, L\varphi \leftrightarrow p\} \models p$;
- for all $L\psi \in \Lambda$, we have $\Sigma' \cup \Lambda' \equiv \Sigma \cup \Lambda \cup \mathrm{SB}_{\Sigma\cup\Lambda}(\varphi) \cup \{L\varphi, L\varphi \leftrightarrow p, Lp\} \models_L \psi$ using the same derivation as for $\Sigma \cup \Lambda$; whereas for all $\neg L\psi \in \Lambda$, we still have $\Sigma' \cup \Lambda' \equiv \Sigma \cup \Lambda \cup \mathrm{SB}_{\Sigma\cup\Lambda}(\varphi) \cup \{L\varphi, L\varphi \leftrightarrow p, Lp\} \not\models_L \psi$.

Hence, Σ' has a stable expansion.

Conversely, suppose that φ is not credulously entailed. Hence Σ does not have $\mathcal{L}_{\text{ae}}$ as a stable expansion and $\neg L\varphi \in \Delta$ for all stable expansions Δ of

Σ. Observe that $\Sigma' \cup \mathrm{SF}^L(\Sigma') = \Sigma \cup \mathrm{SF}^L(\Sigma) \cup \{L\varphi \leftrightarrow p, Lp\} \cup \{L\varphi, Lp\}$ is consistent, therefore $\mathcal{L}_{\mathrm{ae}}$ is not a stable expansion of Σ' either.

Hence, assume that Δ' is a consistent stable expansion of Σ'. Then either $Lp \in \Delta'$ or $\neg Lp \in \Delta'$. In the former case, Δ' would also have to contain $L\varphi$, while φ cannot be derived. A contradiction to Δ' being a stable expansion of Σ'. In the latter case, we have that $\mathrm{Th}(\Sigma' \cup L(\Delta') \cup \neg L(\overline{\Delta'})) \supseteq \{\neg Lp, Lp\}$. Thus, $\mathrm{Th}(\Sigma' \cup L(\Delta') \cup \neg L(\overline{\Delta'})) = \mathcal{L}_{\mathrm{ae}} \supsetneq \Delta'$—a contradiction to Δ' being a stable expansion. We conclude that Σ' does not posses any stable expansions. □

Proof of Theorem 5.2.1. Let B be a finite set of Boolean functions. According to Lemma 5.2.2 one can suppose without loss of generality that B contains the constant 1. Since 1 belongs to all stable expansions, a set Σ of autoepistemic B-formulae has a stable expansion if and only if 1 belongs to some stable expansion of Σ. Therefore, the lower bounds follow from Theorem 4.2.1.

As for the upper bounds, membership of $\textsc{Cred}_{\mathrm{AE}}(B)$ in Σ_2^{p} for $[B] = \mathsf{BF}$ follows from Theorem 3.2.4.

For $[B] \subseteq \mathsf{V}$, the proof of Lemma 4.2.5 shows that, given $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$, we can compute a Σ-full set Λ in NP. By Lemma 3.2.8, it remains to check whether $\Sigma \cup \Lambda \models_L \varphi$. To this end, we nondeterministically guess a set $T \subseteq \mathrm{SF}^q(()\varphi) \cap \mathrm{SF}^L(()\varphi)$, verify that $\Sigma \cup \Lambda \cup T \cup \{\neg L\chi \mid L\chi \in \mathrm{SF}^q(()\varphi) \cap \mathrm{SF}^L(()\varphi) \setminus T\} \models \varphi$, and recursively check that

- $\Sigma \cup \Lambda \models_L \chi$ for all $L\chi \in T$,
- $\Sigma \cup \Lambda \not\models_L \chi$ for all $L\chi \in \mathrm{SF}^q(()\varphi) \cap \mathrm{SF}^L(()\varphi) \setminus T$.

This recursion terminates after at most $|\varphi|$ steps as $|\mathrm{SF}^q(\varphi) \cap \mathrm{SF}^L(\varphi)| \leq |\mathrm{SF}(\varphi)| \leq |\varphi|$ and $\Sigma \cup \Lambda \models_L \chi$ if and only if $\Sigma \cup \Lambda \models \chi$ for all propositional formulae χ. The above algorithm hence constitutes a polynomial-time Turing reduction to the implication problem for propositional B-formulae. Using Theorem 2.5.1, we now obtain that $\Sigma \cup \Lambda \models_L \varphi$ is polynomial-time decidable; thence, $\textsc{Cred}_{\mathrm{AE}}(B) \in \mathrm{NP}$.

For $[B] \subseteq \mathsf{N}$ and $[B] \subseteq \mathsf{E}$, the proofs of Lemmas 4.2.9 and 4.2.10 show that, given $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$, computation of a Σ-full set Λ can be performed in respectively $\mathrm{AC}^0[2]$ and AC^0, while deciding $\Sigma \cup \Lambda \models_L \varphi$ reduces to testing whether $\Sigma \cup \Lambda \models \psi$ for the (unique) atomic subformula $\psi \in \mathrm{SF}(\varphi)$.

Finally, for $[B \cup \{0,1\}] = \mathsf{L}$, the claim follows from Lemmas 4.2.8 and 5.2.3. □

We will now give the complexity classification of the skeptical reasoning problem. The analysis is similar to the credulous reasoning problem.

Theorem 5.2.4 *Let B be a finite set of Boolean functions. Then $\textsc{Skep}_{\mathrm{AE}}(B)$ is*

1. Π_2^{p}*-complete if* $\mathsf{D}_2 \subseteq [B]$ *or* $\mathsf{S}_{00} \subseteq [B]$ *or* $\mathsf{S}_{10} \subseteq [B]$,
2. coNP*-complete if* $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$,
3. $\oplus$L*-hard and contained in* P *if* $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$,
4. $\mathrm{AC}^0[2]$*-complete if* $\mathsf{N}_2 \subseteq [B] \subseteq \mathsf{N}$, *and*

5. *in* AC^0 *in all other cases (that is, if* $[B] \subseteq \mathsf{E}$*),*

with respect to constant-depth reductions.

To complete the proof of Theorem 5.2.4, we require a result analogous to Lemma 5.2.3.

Lemma 5.2.5 $\overline{\mathrm{SKEP}_{\mathrm{AE}}(B)} \leq_{\mathrm{cd}} \mathrm{EXP}'(B \cup \{\oplus\})$ *for all finite sets* B *of Boolean functions, where* EXP' *denotes the problem of deciding the existence of a consistent stable expansion.*

Proof. Let B be a finite set of Boolean functions. Given $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ and $\varphi \in \mathcal{L}_{\mathrm{ae}}(B)$, map the pair (Σ, φ) to $\Sigma' := \Sigma \cup \{L\varphi \oplus p, Lp\}$, where p is a fresh proposition. We claim that φ is contained in any stable expansion of Σ if and only if $\Sigma' \notin \mathrm{EXP}(B)$.

First suppose that there exists a stable expansion Δ of Σ that does contain φ. Let Λ denote its kernel. Then, for the same arguments as in the proof of Lemma 5.2.3, $\Lambda' := \Lambda \cup \mathrm{SB}_{\Sigma\cup\Lambda}(\varphi) \cup \{\neg L\varphi, Lp\}$ is a Σ'-full set.

Conversely, suppose that φ is contained in all stable expansions Δ of Σ and let Δ' denote a consistent stable expansion of Σ'. Then either $Lp \in \Delta'$ or $\neg Lp \in \Delta'$. If $Lp \in \Delta'$, then Δ' would also have to contain $\neg L\varphi$, while φ can be derived. A contradiction to Δ' being a stable expansion of Σ'. Otherwise, if $\neg Lp \in \Delta'$, then $\Sigma' \cup L(\Delta') \cup \neg L(\overline{\Delta'})$ is inconsistent—contradictory to Δ' being a consistent stable expansion. We conclude that Σ' does not posses any consistent stable expansion. □

Proof of Theorem 5.2.4. We proceed analogous to the proof of Theorem 5.2.1. According to Lemma 5.2.2 one can suppose without loss of generality that B contains the constant 0. Since 0 does not belong to any consistent stable expansion, a set Σ of autoepistemic B-formulae has no consistent stable expansion if and only if 0 belongs to all stable expansions of Σ. The lower bounds thus follow from Corollary 4.2.11, while the upper bounds can be derived from Lemmas 4.2.5, 4.2.8 to 4.2.10 and 5.2.5 as in the proof of Theorem 5.2.1. □

5.3 Reasoning in Circumscription

In the last section of this chapter, we analyze the complexity of reasoning in circumscription.

In general the problem has been shown to be complete for Π_2^{p} in Theorem 3.3.5; and this completeness continues to hold for all sets of Boolean functions that are neither monotone nor affine. If all available functions are affine or monotone, then the complexity of the problem is contained in coNP, whereby it is coNP-complete in the latter case as long as $\vee$ remains expressible using the available Boolean functions and the constant 1. This decrease in the complexity comes from different sources: for monotone functions the test for minimality of models becomes tractable, while for affine functions the implication problem becomes tractable. If the set of available Boolean functions is further restricted to contain either only negations or only conjunctions, then the problem becomes polynomial-time solvable (that is, its complexity drops to respectively $\mathrm{AC}^0[2]$-completeness or membership in AC^0). The classification is shown in Figure 5.3 on page 71.

Theorem 5.3.1 *Let B be a finite set of Boolean functions. Then* $\textsc{CircInf}(B)$ *is*

1. Π_2^{p}*-complete if* $\mathsf{S}_{02} \subseteq [B]$ *or* $\mathsf{S}_{12} \subseteq [B]$ *or* $\mathsf{D}_1 \subseteq [B]$*,*
2. coNP*-complete if* $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{M}$ *or* $\mathsf{S}_{10} \subseteq [B] \subseteq \mathsf{M}$ *or* $\mathsf{D}_2 \subseteq [B] \subseteq \mathsf{M}$*,*
3. $\oplus$L*-hard and contained in* coNP *if* $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$*,*
4. $\mathrm{AC}^0[2]$*-complete if* $\mathsf{N}_2 \subseteq [B] \subseteq \mathsf{N}$*, and*
5. *in* AC^0 *in all other cases (that is, if* $[B] \subseteq \mathsf{E}$*),*

with respect to constant-depth reductions.

Similar to Theorem 5.1.5, Theorem 5.3.1 is asymmetric in the sense that the complexity of circumscriptive inference differs for sets of Boolean functions being dual to each other, which is not the case for inference in propositional logic: the inference problem for both V- and E-formulae in propositional logic is contained in AC^0, whereas $\textsc{CircInf}(B)$ is coNP-hard if $\mathsf{V}_2 \subseteq [B]$. This stems from the identification of $\Gamma \subseteq \mathcal{L}(\{\vee\})$ with the conjunction of the contained formulae, while the (P, Q, Z)-minimality of models for Γ allows for the simulation of atomic negations. Therefore, $\textsc{CircInf}(B)$ with $\mathsf{V}_2 \subseteq [B]$ is as hard as the implication problem for formulae in conjunctive normal form. On the other hand, if $B \subseteq \mathsf{E}$ then Γ is equivalent to a conjunction of propositions, the complexity of the circumscriptive inference problem hence remains in AC^0.

This asymmetry neatly contrasts with the complexity of inference for basic circumscription:

Theorem 5.3.2 *Let B be a finite set of Boolean functions. Then* $\textsc{CircInf}_{Q,Z=\emptyset}(B)$ *is contained in* AC^0 *for all B such that* $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$ *and equivalent to* $\textsc{CircInf}(B)$ *with respect to constant-depth reductions in all other cases.*

Remark 5.3.3 *Another possible restriction of* $\textsc{CircInf}(B)$ *is to require Γ to be a singleton, that is, that Γ is a formula. Denote this modified version of the problem as* $\textsc{CircInf}^1(B)$*. In [Tho09], the author proves that* $\textsc{CircInf}^1(B)$ *is* $\mathrm{AC}^0[2]$*-complete for all B such that* $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$*, contained in* AC^0 *for all B such that* $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$*, and* $\leq_{\mathrm{cd}}$*-equivalent to* $\textsc{CircInf}(B)$ *in all other cases.*

The proof of Theorems 5.3.1 and 5.3.2 will be established from the lemmas in this section. To begin with, the following lemma reduces the number of clones to be considered.

Lemma 5.3.4 *Let B be a finite sets of Boolean functions. It holds that* $\textsc{CircInf}(B) \equiv_{\mathrm{cd}} \textsc{CircInf}(B \cup \{1\})$. *Additionally, if $\vee \in [B]$ then* $\textsc{CircInf}(B) \equiv_{\mathrm{cd}} \textsc{CircInf}(B \cup \{0\})$. *The equivalences hold even if $Z = Q = \emptyset$ is assumed.*

Proof. We will first show that $\textsc{CircInf}(B \cup \{1\}) \leq_{\mathrm{cd}} \textsc{CircInf}(B)$. We map the given $\textsc{CircInf}(B \cup \{1\})$-instance $(\Gamma, \psi, (P, Q, Z))$ to $(\Gamma_{[1/t]} \cup \{t\}, \psi_{[1/t]}, (P \cup \{t\}, Q, Z))$. Now, for all models σ of $\Gamma_{[1/t]} \cup \{t\}$, we have that $\sigma(t) = 1$. Moreover, there is a bijection between the models of Γ and $\Gamma_{[1/t]} \cup \{t\}$. Hence, $(\Gamma, \psi, (P, Q, Z)) \in \textsc{CircInf}(B \cup \{1\}) \iff (\Gamma_{[1/t]} \cup \{t\}, \psi_{[1/t]}, (P \cup \{t\}, Q, Z)) \in \textsc{CircInf}(B)$.

It remains to show that $\textsc{CircInf}(B \cup \{0\}) \leq_{\mathrm{cd}} \textsc{CircInf}(B)$ if $\vee \in [B]$. For the nontrivial direction, observe that $\Gamma \models^{\mathrm{circ}}_{(P,Q,Z)} \psi$ if and only if $\Gamma_{[0/f]} \models^{\mathrm{circ}}_{(P \cup \{f\}, Q, Z)} \psi \vee f$. □

Lemma 5.3.5 *Let B be a finite set of Boolean functions such that $[B] = \mathsf{M}$. Then* $\textsc{CircInf}(B)$ *is* coNP*-complete with respect to constant-depth reductions, even if $Q = Z = \emptyset$.*

Proof. Let B be a finite set of Boolean functions such that $[B] = \mathsf{M}$. For membership in coNP, let $\Gamma \subseteq \mathcal{L}(B)$, $\varphi \in \mathcal{L}(B)$ and (P, Q, Z) be a partition of the set of propositions. It holds that $\Gamma \models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$ if and only if for all (P, Q, Z)-minimal models σ of Γ, $\sigma \models \Gamma$ implies $\sigma \models \varphi$.

Due to the monotonicity of the functions in B, we have that $\sigma \models \psi$ implies that $\sigma \cup \{x\} \models \psi$ for all B-formulae ψ and all propositions x. Thus, a model σ of Γ is (P, Q, Z)-minimal if and only if $(\sigma \cup Z) \setminus \{x\} \not\models \Gamma$ for all $x \in P$ with $\sigma(x) = 1$. One can hence check in polynomial time whether σ is a (P, Q, Z)-minimal model of Γ. Consequently, to prove that $\Gamma \not\models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$ it suffices to guess an assignment σ and check (in polynomial time according to the discussion above) that σ is a minimal model of Γ falsifying φ. This shows that $\textsc{CircInf}(B) \in$ coNP.

As for coNP-hardness, we give a reduction from $3\textsc{Taut}$. Let $\varphi \in \mathcal{L}$ be in disjunctive normal form with exactly three literals per term. Assume without loss of generality that $\mathrm{Vars}(\varphi) = \{x_1, \ldots, x_n\}$ and denote this set by X. Let $Y = \{y_1, \ldots, y_n\}$ be a set of propositions disjoint from X. Denote by φ' the formula derived from φ by replacing all negative literals $\neg x_i$ by y_i. By virtue of Lemma 2.4.2 (2.), we may assume that $\wedge$ and $\vee$ can be efficiently implemented in B. We can hence define the reduction function f as

$$f \colon \varphi \mapsto \Big(\Big\{ \bigwedge_{1 \leq i \leq n} (x_i \vee y_i) \Big\}, \varphi', (X \cup Y, \emptyset, \emptyset)\Big).$$

The formulae $\bigwedge_{1 \leq i \leq n}(x_i \vee y_i)$ and φ' are obviously monotone and can furthermore be constructed using AC^0-circuits. We show that $\varphi \in 3\textsc{Taut}$ if and only if $f(\varphi) \in \textsc{CircInf}(B)$.

First assume that $\varphi \in$ 3TAUT. Then $\sigma \models \varphi(X)$ for any assignment $\sigma\colon X \to \{0,1\}$. We define σ' as the extension of σ to $X \cup Y$ defined by $\sigma'(y_i) = 1 - \sigma(x_i)$ for all $1 \leq i \leq n$. As a result, for any two different models $\sigma_1, \sigma_2\colon X \to \{0,1\}$ of φ, the corresponding assignments σ_1', σ_2' are incomparable under $\leq_{(X\cup Y,\emptyset,\emptyset)}$.

Now assume that there exists a $(X \cup Y, \emptyset, \emptyset)$-minimal model σ' of the premise such that $\sigma'(x_i) = \sigma'(y_i) = 1$ for some $1 \leq i \leq n$. Then both assignments $\sigma' \setminus \{x_i\}$ and $\sigma' \setminus \{y_i\}$ still satisfy $\bigwedge_{1\leq i\leq n}(x_i \vee y_i)$, a contradiction to σ' being minimal. Hence, any $(X \cup Y, \emptyset, \emptyset)$-minimal model σ' of the premise satisfies $\sigma'(x_i) \neq \sigma'(y_i)$ for all $1 \leq i \leq n$. As a result, any such model is an extension of some assignment $\sigma\colon X \to \{0,1\}$ as defined above. From the assumption $\sigma \models \varphi$, we finally obtain $\sigma' \models \varphi_{[\neg x_1/y_1,\dots\neg x_n/y_n]} = \varphi'$.

As for $\varphi \notin$ 3TAUT, there exists an assignment $\sigma\colon X \to \{0,1\}$ falsifying $\sigma \models \varphi$. Let σ' again be defined as the extension of σ to $X \cup Y$ satisfying $\sigma'(y_i) = 1 - \sigma(x_i)$. According to the above discussion, σ' is a $(X \cup Y, \emptyset, \emptyset)$-minimal model of $\bigwedge_{1\leq i\leq n}(x_i \vee y_i)$ such that $\sigma' \not\models \varphi'$. Hence, σ' witnesses $\{\bigwedge_{1\leq i\leq n}(x_i \vee y_i)\} \not\models^{\text{circ}}_{(P,Q,Z)} \varphi'$. □

The following proposition re-proves coNP-hardness for a clone properly contained in M. However, the proof of Lemma 5.3.5 is required to establish the coNP-hardness of $\text{CIRCINF}_{Q,Z=\emptyset}(B)$ for $[B] = \mathsf{M}$ (see Theorem 5.3.2).

Lemma 5.3.6 *Let B be a finite set of Boolean functions such that $[B] = \mathsf{V}$. Then* $\text{CIRCINF}(B)$ *is* coNP*-complete with respect to constant-depth reductions, while* $\text{CIRCINF}_{Q,Z=\emptyset}(B)$ *is contained in* AC^0.

Proof. Let B be a finite set of Boolean functions such that $[B] = \mathsf{V}$. Membership of $\text{CIRCINF}(B)$ in coNP is immediate from Lemma 5.3.5. To prove its coNP-hardness, we again reduce from 3TAUT to $\text{CIRCINF}(B)$.

Given $\varphi = \bigvee_{i=1}^{m}(\ell_{i1} \wedge \ell_{i2} \wedge \ell_{i3})$ with $\text{Vars}(\varphi) = \{x_1, \dots, x_n\}$, we map $\varphi \mapsto (\Gamma, z, (P, Q, Z))$, where $P := \text{Vars}(\varphi) \cup \{\hat{x} : x \in \text{Vars}(\varphi)\} \cup \{t_i : 1 \leq i \leq m\}$, $Q := \emptyset$, $Z := \{z\}$ with z being a fresh proposition and Γ defined as follows:

1. For each proposition $x \in \text{Vars}(\varphi)$, Γ contains the formula $x \vee \hat{x}$.
2. For each term $(\ell_{i1} \wedge \ell_{i2} \wedge \ell_{i3})$ of φ, Γ contains the four formulae
$$t_i \vee \ell'_{i1}, \quad t_i \vee \ell'_{i2}, \quad t_i \vee \ell'_{i3}, \quad t_i \vee z,$$
where $\ell'_{ij} := x$ if $\ell_{ij} = x$ and $\ell'_{ij} := \hat{x}$ if $\ell_{ij} = \neg x$, for some $x \in \text{Vars}(\varphi)$.

We claim that $\varphi \in$ 3TAUT if and only if $\Gamma \models^{\text{circ}}_{(P,Q,Z)} z$.

Suppose that φ is tautological. In this case, every assignment $\sigma\colon \text{Vars}(\varphi) \to \{0,1\}$ satisfies at least one term $(\ell_{i1} \wedge \ell_{i2} \wedge \ell_{i3})$ of φ. As every model of Γ sets to 1 either x or $\hat{x}$ for all $x \in \text{Vars}(\varphi)$, for every model σ' of Γ there exists a model σ'' of Γ such that $\sigma'' \leq_{(P,Q,Z)} \sigma'$ and that is an extension of a model of φ. Any model σ' of Γ thus sets $\sigma'(\ell'_{ij}) = 1$ for $1 \leq j \leq 3$. As the corresponding t_i occurs only in the formulae $t_i \vee \ell'_{i1}, t_i \vee \ell'_{i2}, t_i \vee \ell'_{i3}, t_i \vee z$, it follows that any (P, Q, Z)-minimal model σ' sets $\sigma'(t_i) = 0$ and $\sigma'(z) = 1$. We conclude that $\Gamma \models^{\mathrm{circ}}_{(P,Q,Z)} z$.

On the other hand, suppose that $\sigma'(z) = 1$ for all (P, Q, Z)-minimal models σ' of Γ. Then, any such model falsifies at least one t_i, because otherwise $\sigma' \setminus \{z\}$ would also be a (P, Q, Z)-minimal model of Γ, in contradiction to the assumption. This in turn implies the existence of a term $(\ell_{i1} \wedge \ell_{i2} \wedge \ell_{i3})$ such that $\sigma \models \ell_{ij}$ for all $1 \leq j \leq 3$, where $\sigma := \sigma' \cap \mathrm{Vars}(\varphi)$. We conclude that $\sigma \models \varphi$.

As the propositions t_i need not be numbered by i, but can rather be indexed according to their position in the input string, we conclude that the reduction is implementable in an AC^0-circuit. Whence the coNP-hardness of $\mathrm{CIRCINF}(B)$ under constant-depth reductions follows.

We will now prove that $\mathrm{CIRCINF}(B)$ is contained in AC^0 if the problem is restricted to $Q = Z = \emptyset$. Let $\Gamma \subseteq \mathcal{L}(B)$, let $\varphi \in \mathcal{L}(B)$, and assume without loss of generality that $\mathrm{Vars}(\Gamma \cup \{\varphi\}) = \{x_1, \ldots, x_n\}$. Moreover, let $\Gamma = \{\psi_i \mid 1 \leq i \leq m\}$. Then $\Gamma \equiv c_1 \vee \bigwedge_{j=1}^{m} \bigvee_{i \in I_j} x_i$ and $\varphi \equiv c_2 \vee \bigvee_{i \in J} x_i$, where $c_1, c_2 \in \{0, 1\}$ and $I_1, \ldots, I_m, J \subseteq \{1, \ldots, n\}$. These representations can be computed using AC^0-circuits, since $c_1 \equiv 1$ ($c_2 \equiv 1$) if and only if $\emptyset \models \Gamma$ (respectively $\emptyset \models \varphi$) and $i \in I_k$, $1 \leq k \leq m$, ($i \in J$) if and only if $c_1 \equiv 0$ and $\{x_i\} \models \psi_k$ (respectively $i \in J$ if and only if $c_2 \equiv 0$ and $\{x_i\} \models \varphi$). Henceforth assume without loss of generality that $\varphi \not\equiv 1$ (otherwise, $\Gamma \models^{\mathrm{circ}} \varphi$ trivially holds). Define $X_J := \{x_i : i \in J\}$ to be the set of propositions occurring in X_J and let $\sigma \colon X_J \to \{0, 1\}$ be the partial assignment defined by $\sigma(x_i) = 0$ for all $i \in J$. Then $\Gamma \models^{\mathrm{circ}} \varphi$ if and only if σ cannot be extended to a minimal model of Γ.

We show that σ cannot be extended to a minimal model of Γ if and only if $\Gamma_{[X_J/0]} := c_1 \vee \bigwedge_{j=1}^{m} \bigvee_{i \in I_j \setminus J} x_i$ is unsatisfiable. Suppose that σ cannot be extended to a minimal model of Γ. Either σ cannot be extended to a model of Γ or for all extensions σ' of σ satisfying Γ, there exists a model ρ of Γ such that $\rho < \sigma'$. In the former case, we obtain that $\Gamma \cup \{\neg x_i \mid \sigma(x_i) = 0\} \equiv \Gamma_{[X_J/0]}$ is unsatisfiable; whereas in the latter case, there has to exist some minimal model ρ' of Γ with $\rho' < \sigma'$ for σ' as above. But then $\rho' \cap \{x_i \mid i \in J\} = \emptyset$, because $\sigma' \cap \{x_i \mid i \in J\} = \sigma \cap \{x_i \mid i \in J\}$ and $\rho' < \sigma'$—contradictory to the assumption that σ cannot be extended to a minimal model of Γ. We conclude that for all extensions σ' of σ, $\sigma' \not\models \Gamma$ and, in particular, that $\Gamma_{[X_J/0]}$ is unsatisfiable. Conversely, suppose that σ can be extended to a minimal model σ' of Γ. Then, clearly, $\sigma' \models \Gamma_{[X_J/0]}$.

To decide whether $\Gamma \models^{\mathrm{circ}} \varphi$, it thus suffices to check whether $\Gamma_{[X_J/0]}$ is satisfiable. As $\Gamma_{[X_J/0]}$ is satisfiable if and only if it is satisfiable by the all-1 assignment, membership of $\mathrm{CIRCINF}_{Q,Z=\emptyset}(B)$ in AC^0 follows. □

An argument similar to the above can now be used to show that $\mathrm{CIRCINF}(B) \in \mathrm{AC}^0$ for all B such that $[B] \subseteq \mathsf{E}$.

Lemma 5.3.7 *Let B be a finite set of Boolean functions such that $[B] = \mathsf{E}$. Then $\mathrm{CIRCINF}(B)$ is contained in AC^0.*

Proof. Let B be a finite set of Boolean functions such that $[B] = \mathsf{E}$. Let $\Gamma \subseteq \mathcal{L}(B)$, let $\varphi \in \mathcal{L}(B)$ and let (P, Q, Z) partition the set of propositions. Then Γ is equivalent to $\bigwedge_{x \in X} x$ for some set of propositions X. Let σ be the partial assignment on $P \cup X$ defined as $\sigma(x) := 1$ if and only if $x \in X$. Let σ' be an arbitrary (P, Q, Z)-minimal model of Γ. Then, $\sigma'(x) = \sigma(x)$ for all $x \in P \cup X$. Therefore, $\sigma' \models \varphi$ if and only if $\sigma \models \varphi$ for all such σ'. We conclude that $\Gamma \models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$ if and only if $\sigma \models \varphi$, which can be verified in AC^0.

The claim now follows from the fact that the conjunctive representations of Γ and φ can be computed using AC^0-circuits: for $\psi \in \mathcal{L}(B)$ with $\psi \equiv \bigwedge_{x \in X} x$, we have $x \in X$ if and only if $\mathrm{Vars}(\psi) \models \psi$ and $\mathrm{Vars}(\psi) \setminus \{x\} \not\models \psi$. □

Lemma 5.3.8 *Let B be a finite set of Boolean functions such that $[B] = \mathsf{L}$ or $[B] = \mathsf{L}_1$. Then* CIRCINF(B) *is contained in* coNP *and* $\oplus$L*-hard with respect to constant-depth reductions, even if $Q = Z = \emptyset$.*

Proof. Let B be a finite set of Boolean functions such that $[B] = \mathsf{L}$

We will first show membership in coNP. Let $\Gamma \subseteq \mathcal{L}(B)$, $\varphi \in \mathcal{L}(B)$, and a partition (P, Q, Z) of the set of propositions be given. To prove that $\Gamma \not\models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$, we guess an assignment σ and verify that it is a (P, Q, Z)-minimal model of Γ that falsifies φ. The test for (P, Q, Z)-minimality proceeds as follows:

Let $\sigma \colon \mathrm{Vars}(\Gamma \cup \{\varphi\}) \to \{0, 1\}$ be an assignment satisfying Γ and let $p_1, \ldots, p_k$ and $q_1 \ldots, q_m$ enumerate the propositions in $P \setminus \sigma$ and in Q respectively. From the definition of $\leq_{(P,Q,Z)}$, it follows that Γ has a model smaller than σ with respect to $\leq_{(P,Q,Z)}$ if and only if $\Gamma_{[p_1/0,\ldots,p_k/0,q_1/\sigma(q_1),\ldots,q_m/\sigma(q_m)]}$ is satisfiable by an assignment $\sigma' \colon (P \cap \sigma) \cup Z \to \{0, 1\}$ setting at least one proposition in $P \cap \sigma$ to 0. This can be tested in polynomial time, see [KK01b].

For the $\oplus$L-hardness, recall that the classical inference problem for affine 1-reproducing formulae is hard for $\oplus$L under constant-depth reductions (see Theorem 2.5.1). Hence, mapping an instance (Γ, φ) over propositions $X = \{x_1, \ldots, x_n\}$ to $(\Gamma \cup \Delta, \varphi, (X \cup Y \cup Z, \emptyset, \emptyset))$ with $\Delta := \{x_i \oplus y_i \oplus z_i : 1 \leq i \leq n\}$, $Y := \{y_1, \ldots, y_n\}$, and $Z := \{z_1, \ldots, z_n\}$ yields the desired reduction: the models of Γ extend to pairwise incomparable and therefore minimal models of $\Gamma \cup \Delta$.□

Lemma 5.3.9 *Let B be a finite set of Boolean functions such that $[B] = \mathsf{N}$. Then* CIRCINF(B) *is* $\mathrm{AC}^0[2]$*-complete with respect to constant-depth reductions, even if $Q = Z = \emptyset$.*

Proof. Let B be a finite set of Boolean functions such that $[B] = \mathsf{N}$ and let (P, Q, Z) be a partition of the set of propositions. As $[\{\neg\}] = \mathsf{N}_2$, any formula φ is equivalent to some literal. Denote this literal by ℓ_φ. Then, for $\Gamma \subseteq \mathcal{L}(B)$, $\Gamma \equiv \bigwedge_{\varphi \in \Gamma} \ell_\varphi$. Accordingly, all (P, Q, Z)-minimal models σ of Γ satisfy, for all $x \in P \cup \mathrm{Vars}(\Gamma)$, $\sigma(x) = 1$ if and only if $\ell_\varphi = x$ for some $\varphi \in \Gamma$. Hence all (P, Q, Z)-minimal models of Γ coincide on $P \cup \mathrm{Vars}(\Gamma)$.

Thus, given $\Gamma \subseteq \mathcal{L}(B)$, $\varphi \in \mathcal{L}(B)$ and a partition (P, Q, Z) of the set of propositions, we can compute the above representation of Γ and accept if and only if $\psi \equiv \ell_\varphi$ for some $\varphi \in \Gamma$: if ψ is equivalent to an ℓ_φ for some $\varphi \in \Gamma$ then any (P, Q, Z)-minimal model evaluates ψ to 1; otherwise, if $\psi \not\equiv \ell_\varphi$ for all $\varphi \in \Gamma$ then the assignment σ defined as

$$\sigma(x) := \begin{cases} 1 & \text{if } \psi \equiv \neg x, \\ 1 & \text{if } x \in P \cup \mathrm{Vars}(\Gamma) \text{ and } x = \ell_\varphi \text{ for some } \varphi \in \Gamma, \\ 0 & \text{for all remaining } x \in \mathrm{Vars}(\Gamma) \end{cases}$$

witnesses $\Gamma \not\models^{\mathrm{circ}}_{(P,Q,Z)} \psi$. Membership in $\mathrm{AC}^0[2]$ follows from the fact that N-formulae can be evaluated in $\mathrm{AC}^0[2]$ (see [Sch10]).

To establish $\mathrm{AC}^0[2]$-hardness, we give a reduction from MOD_2. Let $w = w_1 \cdots w_n$ with $w_i \in \{0, 1\}$, $1 \leq i \leq n$, be given and let $f_\neg(x)$ be the B-representation of $\neg$. As the value of all functions in N depends on at most one variable and $[B] \subseteq \mathsf{L}$, we may without loss of generality assume that x is the last symbol in $f_\neg$. We transform w into the formula $\varphi := f_1 f_2 \cdots f_n f_\neg t$, where $f_i := f_\neg$ if $w_i = 1$, and $f_i :=$ id otherwise. Clearly, $w \in \mathrm{MOD}_2$ if and only if $t \models^{\mathrm{circ}}_{(\{t\},\emptyset,\emptyset)} f_1 f_2 \cdots f_n f_\neg t$. □

We are now ready to give the proof of Theorems 5.3.1 and 5.3.2.

Proof of Theorems 5.3.1 and 5.3.2. According to Lemma 5.3.4, we may without loss of generality assume that B includes the constant 1 and that B includes the constant 0 if $\vee \in [B]$. Now observe that $[\mathsf{D}_1 \cup \{1\}] \supseteq \mathsf{S}_{02}$, $[\mathsf{S}_{12} \cup \{1\}] \supseteq \mathsf{S}_{02}$, and $[\mathsf{S}_{02} \cup \{0, 1\}] = \mathsf{BF}$. The Π^{p}_2-complete cases hence follow from Theorem 3.3.5 and Lemma 5.3.4. It analogously holds that $[\mathsf{D}_2 \cup \{1\}] \supseteq \mathsf{S}_{00}$, $[\mathsf{S}_{10} \cup \{1\}] \supseteq \mathsf{S}_{00}$, and $[\mathsf{S}_{00} \cup \{0, 1\}] = \mathsf{M}$. Therefore, the coNP-complete cases are established from Lemmas 5.3.4 to 5.3.6. The remaining cases follow from Lemmas 5.3.4 and 5.3.7 to 5.3.9 using identical arguments. □

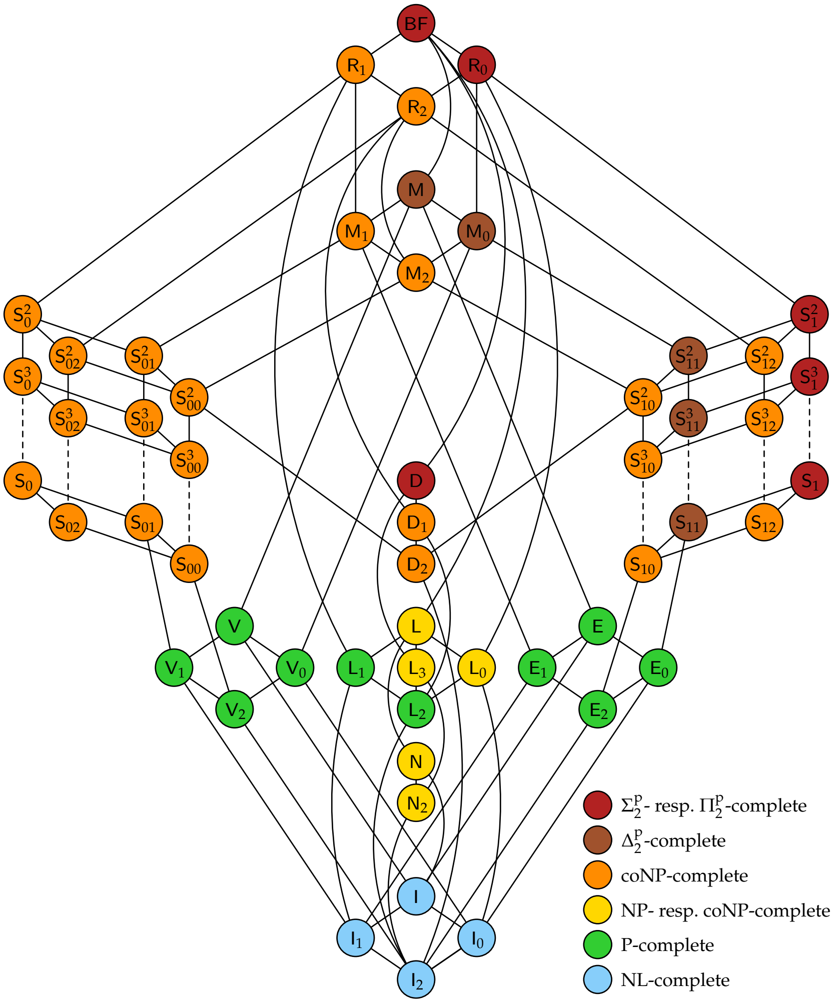

Figure 5.1: The complexity of $\textsc{Cred}_{\mathrm{DL}}(B)$ and $\textsc{Skep}_{\mathrm{DL}}(B)$

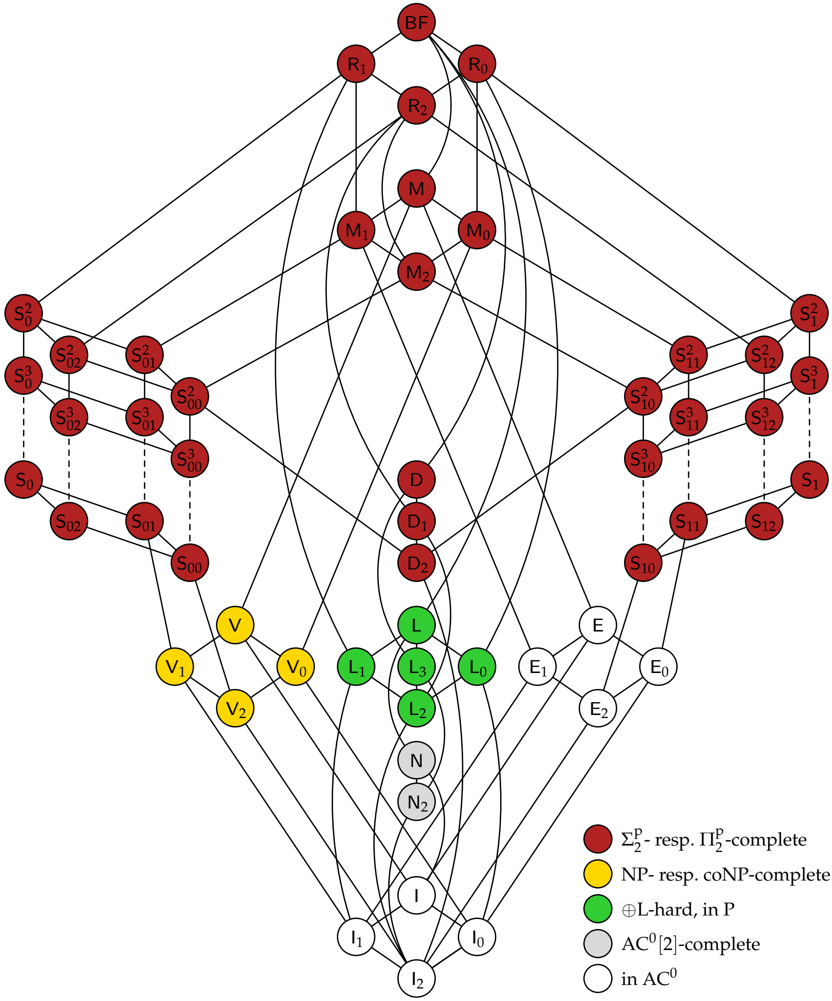

Figure 5.2: The complexity of $\textsc{Cred}_{\mathrm{AE}}(B)$ and $\textsc{Skep}_{\mathrm{AE}}(B)$

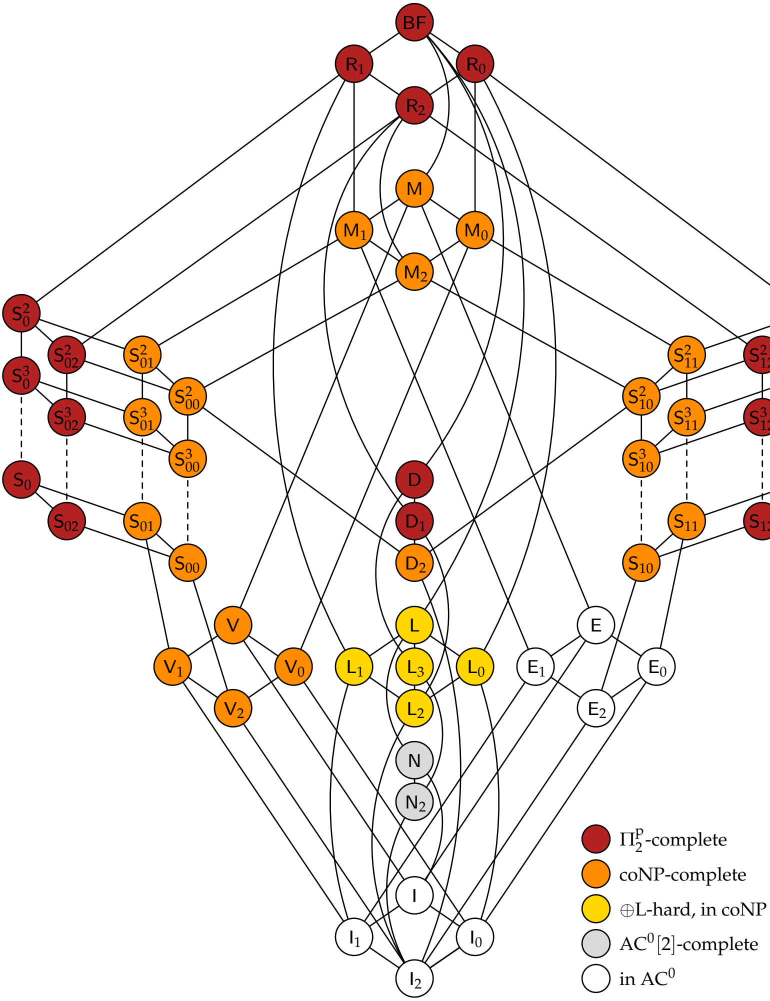

Figure 5.3: The complexity of CIRCINF(B)

Chapter 6

Counting

The problems considered until now were decision problems, that is, problems whose answer is either "yes" or "no". However, in many situations one might not only be interested in the existence of a solution but their number. Although for many problems in P this number is computable in polynomial time (the problem to count the number of minimum spanning trees mentioned in Chapter 1 belongs to this class), there are numerous problems in P whose counting variant is known to be complete for #P, the class of functions counting the number of accepting paths of NP machines. The prime example of these is the problem to count the number of perfect matchings in a bipartite graph, or equivalently, computing the permanent of a square matrix [Val79a]. This demonstrates that there can be a dramatic gap between the complexity of a counting problem and its underlying decision problem: a deterministic polynomial-time computation using a single call to an oracle in #P suffices to decide any language in the polynomial hierarchy [Tod91]

In this chapter, we study the complexity of counting the number of stable extensions of a default theory, the complexity of counting the number of stable expansions of an autoepistemic theory, and the complexity of counting the number of circumscriptive (that is, (P, Q, Z)-minimal) models of a given set of formulae. We provide a full classification for each of these problems for all finite sets of allowed Boolean functions. In particular, we prove that with one remarkable exception the complexity of all three problems forms a trichotomy: it is either #·coNP-complete and thus presumably harder than computing the permanent, #P-complete, or contained in FP. The exceptional case concerns the problem to count the number of stable extensions for B-default theories with $[B \cup \{1\}] = \mathsf{M}$. These theories may possess either no or exactly one stable extension, whence the counting problem is equivalent to deciding the existence of stable extensions and therefore Δ_2^{P}-complete.

We point out that for our classification of the problems to count the number of stable extensions and to count the number of stable expansions, the conceptually simple parsimonious reductions are sufficient, while for related classifications in the literature less restrictive (and more complicated) reductions such as subtractive or complementive reductions had to be used (see, for example, [DHK05, DH08, BBC$^+$09] and some of the results of [HP07]).

6.1 The Number of Stable Extensions

We start our study of counting problems with the problem to count the number of stable extensions of a given default theory. We formally define the *stable extension counting problem* as follows:

Problem: $\#\text{Ext}(B)$
Input: A B-default theory (W, D)
Output: The number of stable extensions of (W, D)

The following theorem proves that the complexity of this counting problem is tetrachotomous and decreases analogously to the complexity of $\text{Ext}(B)$: it remains $\#\cdot$coNP-complete for all finite sets B such that $[B \cup \{1\}] = \mathsf{BF}$; becomes Δ_2^p-complete for all monotone sets B such that $[B \cup \{1\}] = \mathsf{M}$; is #P-complete for affine sets B implementing $\neg$; and becomes efficiently computable in all other cases (with the theorem additionally distinguishing between trivial and nontrivial cases). The classification is illustrated in Figure 6.4 on page 84.

Observe that here we do not distinguish between decision problems and their characteristic functions: For $\mathsf{S}_{11} \subseteq [B] \subseteq \mathsf{M}$, any B-default theory has zero or one stable extension by Lemma 4.1.3. The counting problem $\#\text{Ext}(B)$ thus coincides with the characteristic function of $\text{Ext}(B)$, whose computation is Δ_2^p-complete.

Theorem 6.1.1 *Let B be a finite set of Boolean functions. Then $\#\text{Ext}(B)$ is*

1. $\#\cdot$coNP*-complete if* $\mathsf{S}_1 \subseteq [B]$ *or* $\mathsf{D} \subseteq [B]$,
2. Δ_2^p*-complete if* $\mathsf{S}_{11} \subseteq [B] \subseteq \mathsf{M}$,
3. #P*-complete if* $[B] \in \{\mathsf{N}, \mathsf{N}_2, \mathsf{L}, \mathsf{L}_0, \mathsf{L}_3\}$,
4. *in* FP *if* $[B] \in \{\mathsf{V}, \mathsf{V}_0, \mathsf{E}, \mathsf{E}_0, \mathsf{I}, \mathsf{I}_0\}$, *and*
5. *trivial in all other cases (that is, if* $[B] \subseteq \mathsf{R}_1$*),*

with respect to parsimonious reductions.

Proof. We first prove the $\#\cdot$coNP-complete cases, then consider the Δ_2^p-complete and #P-complete ones, and finally prove that $\#\text{Ext}(B)$ is contained in FP for $[B] \in \{\mathsf{V}, \mathsf{V}_0, \mathsf{E}, \mathsf{E}_0, \mathsf{I}, \mathsf{I}_0\}$ and trivial in all remaining cases.

To begin with, let B be an arbitrary finite set of Boolean functions. Membership in $\#\cdot$coNP in the general case is obtained from the fact that we can construct a nondeterministic oracle Turing machine M with an NP-oracle whose accepting computation paths are in one-to-one correspondence with the stable extensions of its input. Given the B-default theory (W, D), M nondeterministically guesses a set $G \subseteq D$ and accepts if and only if G is a set of generating defaults for some stable extension, where the test for G being a set of generating defaults proceeds as follows. The set $G = \{\frac{\alpha_i : \beta_i}{\gamma_i} \mid 1 \leq i \leq k\}$ is a set of generating set for some stable extension of (W, D) if and only if there exists a permutation $\pi \in S_k$ such that

1. $W \cup \{\gamma_{\pi(i)} \mid 1 \leq i \leq j-1\} \models \alpha_{\pi(j)}$ for all $1 \leq j \leq k$,
2. $W \cup \{\gamma_i \mid 1 \leq i \leq k\} \not\models \neg\beta_j$ for all $1 \leq j \leq k$, and
3. $W \cup \{\gamma_i \mid 1 \leq i \leq k\} \not\models \alpha$ or $W \cup \{\gamma_i \mid 1 \leq i \leq k\} \models \neg\beta$ for all $\frac{\alpha:\beta}{\gamma} \in D \setminus G$.

Using these conditions, we can iteratively construct π in deterministic polynomial time using an NP-oracle. As the set of generating defaults uniquely characterizes a stable extension, we obtain that the accepting paths of M are in one-to-one correspondence with the stable extensions of (W, D). Hence, $\#\text{E}\textsc{xt}(B) \in \#\cdot\text{P}^{\text{NP}} = \#\cdot\text{coNP}$ for all finite sets B of Boolean functions.

To establish the #·coNP-hardness, let φ be a quantified Boolean formula of the form $\exists x_1 \cdots \exists x_n \forall y_1 \cdots \forall y_m \psi(x_1, \ldots, x_n, y_1, \ldots, y_m)$ with ψ in disjunctive normal form. Then the reduction $f\colon \varphi \mapsto (\emptyset, D)$ with

$$D := \left\{ \frac{1 : x_1}{x_1}, \frac{1 : \neg x_1}{\neg x_1}, \frac{1 : x_2}{x_2}, \frac{1 : \neg x_2}{\neg x_2}, \ldots \frac{1 : x_n}{x_n}, \frac{1 : \neg x_n}{\neg x_n}, \frac{1 : \neg\psi}{0} \right\},$$

given in [Got92, Theorem 5.1], establishes a bijection between the assignments $\sigma\colon \{x_i \mid 1 \leq i \leq n\} \to \{0, 1\}$ that satisfy $\forall y_1 \cdots \forall y_n \psi(x_1, \ldots, x_n, y_1, \ldots, y_m)$ and the stable extensions of $(\emptyset, D)$. Thus, f indeed constitutes a parsimonious reduction from $\#\Pi_1\text{S}\textsc{at}$, that is, the problem of counting the number of satisfying assignments of a quantified Boolean formula $\forall y_1 \cdots \forall y_m \psi(x_1, \ldots, x_n, y_1, \ldots, y_m)$ with ψ in disjunctive normal form. The #·coNP-hardness of $\text{E}\textsc{xt}(B)$ for all finite sets B such that $\mathsf{S}_1 \subseteq [B]$ or $\mathsf{D} \subseteq [B]$ follows from the fact that the reduction given in the proof of Lemma 4.1.2 is parsimonious.

The Δ_2^{p}-completeness of $\#\text{E}\textsc{xt}(B)$ for $\mathsf{S}_{11} \subseteq [B] \subseteq \mathsf{M}$ follows from the fact that by Lemma 4.1.3 the number of stable extensions of a B-default theory coincides with the characteristic function of $\text{E}\textsc{xt}(B)$.

As for the #P-complete cases, Lemma 4.1.7 constitutes a parsimonious reduction from $\#\text{S}\textsc{at}$. Hence, Lemma 4.1.2 implies the claim.

To establish the membership of $\#\text{E}\textsc{xt}(B)$ in FP for all B such that $[B] \subseteq \mathsf{V}$ or $[B] \subseteq \mathsf{E}$, notice that any B-default theory possesses at most one stable extension, whose existence can be verified in polynomial time.

Finally, the triviality for all finite sets B such that $[B] \subseteq \mathsf{R}_1$ follows from Lemma 4.1.3. □

6.2 The Number of Stable Expansions

As regards the number of stable expansions, the situation is similar to that in default logic. Define the *stable expansion counting problem* as:

Problem: $\#\text{E}\textsc{xp}(B)$
Input: A set $\Sigma \subseteq \mathcal{L}$
Output: The number of stable expansions of Σ

The complexity of $\#\textsc{Exp}(B)$ is trichotomous and decreases analogously to the complexity of $\textsc{Exp}(B)$. In other words, it is $\#\cdot$coNP-complete for exactly those cases, where $\textsc{Exp}(B)$ is Σ_2^p-complete; becomes #P-complete for those, where $\textsc{Exp}(B)$ is NP-complete; and drops to membership in FP in all cases for which $\textsc{Exp}(B)$ is tractable. This is summarized in the following theorem (see also Figure 6.5 on page 85).

Theorem 6.2.1 *Let B be a finite set of Boolean functions. Then $\#\textsc{Exp}(B)$ is*

1. $\#\cdot$coNP-*complete if* $\mathsf{D}_2 \subseteq [B]$ *or* $\mathsf{S}_{00} \subseteq [B]$ *or* $\mathsf{S}_{10} \subseteq [B]$,
2. #P-*complete if* $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$,
3. *in* FP *in all other cases (that is, if* $[B] \subseteq \mathsf{L}$ *or* $[B] \subseteq \mathsf{E}$*),*

with respect to parsimonious reductions.

Proof. Let B be a finite set of Boolean functions.

First, suppose that $\mathsf{D}_2 \subseteq [B]$ or $\mathsf{S}_{00} \subseteq [B]$ or $\mathsf{S}_{10} \subseteq [B]$. Then membership in $\#\cdot$coNP follows from the one-to-one correspondence of full sets and stable expansions (Lemma 3.2.6), since a nondeterministic oracle Turing machine with an NP-oracle may guess a candidate for a full set and verify its fullness using Definition 3.2.5. The $\#\cdot$coNP-hardness, on the other hand, follows from Lemma 4.2.4 analogously to the proof of Theorem 6.1.1: Let φ be a quantified Boolean formula of the form $\forall x_1 \cdots \forall x_n \psi(x_1, \ldots, x_n, y_1, \ldots, y_m)$ with ψ in disjunctive normal form. Then the reduction $f\colon \varphi \mapsto \Sigma$ with

$$\Sigma := \{L\psi'\} \cup \{y_j \vee y_j' \mid 1 \leq j \leq m\} \cup \{x_i \vee Lx_i', Lx_i \vee x_i' \mid 1 \leq i \leq n\}$$

and

$$\psi' := \psi_{[\neg x_1/x_1',\ldots,\neg x_n/x_n',\neg y_1/y_1',\ldots,\neg y_m/y_m']}$$

is a parsimonious reduction from $\#\Pi_1\textsc{Sat}$ to $\#\textsc{Exp}(B)$ for all finite sets B with $\mathsf{M} \subseteq [B]$. Finally observe that the reduction given in the proof of Lemma 4.2.2 is parsimonious, whence $\#\textsc{Exp}(B)$ is $\#\cdot$coNP-complete for all B such that $\mathsf{D}_2 \subseteq [B]$ or $\mathsf{S}_{00} \subseteq [B]$ or $\mathsf{S}_{10} \subseteq [B]$.

Second, suppose that $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$. In this case, membership in #P is straightforward from Lemma 4.2.5, while for the #P-hardness it suffices to note that the reduction given in the proof of Lemma 4.2.5 actually establishes a parsimonious reduction from $\#\textsc{Sat}$.

Third, suppose that $[B] \subseteq \mathsf{L}$. Given $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$, it is easy to verify that the transformation to Σ' provided in the proof of Lemma 4.2.8 preserves the number of stable expansions. The number of consistent stable expansions of Σ' is in turn equal to the number of solutions of the system $T'[x_{s+1}/Lx_{s+1}, \ldots, x_n/Lx_n]$ from the proof of Lemma 4.2.6: namely 2^t with t being the number of free variables in this system of linear equations. Moreover, $\mathcal{L}_{\mathrm{ae}}$ is a stable expansion of Σ if and only if $\Sigma \cup \mathrm{SF}^L(\Sigma)$ is inconsistent, which is polynomial-time decidable (see Remark 4.2.7). Thus, $\#\textsc{Exp}(B) \in$ FP.

Fourth, suppose that $[B] \subseteq \mathsf{E}$. Then there exist at most two stable expansions; existence for both of which can be checked in polynomial time (see Lemma 4.2.10). Hence the claim applies. □

6.3 The Number of Minimal Models

The last section of this chapter studies the complexity of counting the number of circumscriptive (that is, (P, Q, Z)-minimal) models of a given set of formulae. Formally, the *circumscriptive model counting problem* is defined as follows:

Problem:	#CIRC
Input:	A set $\Gamma \subseteq \mathcal{L}(B)$ and a partition (P, Q, Z) of the propositions
Output:	The number of (P, Q, Z)-minimal models of Γ

Unlike the preceding counting problems, #CIRC involves sets of Boolean functions for which the problem to decide whether a given assignment is a circumscriptive model is tractable while the counting problem is #P-complete (namely those sets B that satisfy $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$). In all the remaining cases, the complexity of #CIRC(B) can be derived from the complexity of CIRCINF(B) in that completeness for the second level of the polynomial hierarchy translates to #·coNP-completeness, completeness for the first level translates to #P-completeness, and membership in P translates to membership in FP. However, mind that the decision problem underlying the circumscriptive model counting problem is the question whether there exists a (P, Q, Z)-minimal model for the given formula—a problem equivalent to the satisfiability problem for propositional formulae. For $\mathsf{S}_{02} \subseteq [B]$ or $\mathsf{S}_{12} \subseteq [B]$ or $\mathsf{D}_1 \subseteq [B]$, #CIRC$(B)$ thus represents a problem whose underlying decision problem is, though intractable, supposedly easier to solve than the decision problems underlying the generic complete problem for #·coNP.

The complexity of #CIRC is summarized in the theorem below and shown in Figure 6.6 on page 86.

Theorem 6.3.1 *Let B be a finite set of Boolean functions. Then* #CIRC(B) *is*

1. #·coNP-*complete with respect to subtractive reductions if* $\mathsf{S}_{02} \subseteq [B]$ *or* $\mathsf{S}_{12} \subseteq [B]$ *or* $\mathsf{D}_1 \subseteq [B]$,
2. #P-*complete with respect to subtractive reductions if* $\mathsf{S}_{00} \subseteq [B] \subseteq \mathsf{M}$ *or* $\mathsf{S}_{10} \subseteq [B] \subseteq \mathsf{M}$ *or* $\mathsf{D}_2 \subseteq [B] \subseteq \mathsf{M}$,
3. #P-*complete with respect to weakly parsimonious reductions if* $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$ *or* $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$, *and*
4. *in* FP *in all other cases (that is, if* $[B] \subseteq \mathsf{N}$ *or* $[B] \subseteq \mathsf{E}$*).*

The proof of Theorem 6.3.1, given at the end of this section, requires some auxiliary lemmas. Before we proceed, observe that the third item improves a result established by Durand and Hermann in [DH08]. They give a parsimonious reduction from the counting problem

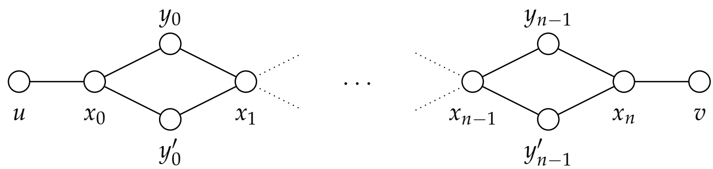

Figure 6.1: The subgraph in G' constructed from $(u, v) \in E$.

Problem: #S-T PATHS
Input: A directed graph $G = (V, E)$ and nodes $s, t \in V$
Output: The number of simple paths from s to t in G,

which is #P-complete under Turing reductions [Val79b]. Durand and Hermann thus establish the #P-hardness of #CIRC(B) for $\mathsf{L}_2 \subseteq [B]$ with respect to Turing reductions. We improve this result by showing that #S-T PATHS is indeed #P-complete with respect to weakly parsimonious reductions.

Lemma 6.3.2 #S-T PATHS *is* #P*-complete with respect to weakly parsimonious reductions.*

Proof. Let #UHAMPATH denote the problem to count the number of Hamiltonian paths from some node s to some node t in an undirected graph. #UHAMPATH is #P-complete with respect to parsimonious reductions [Val79b]. This completeness result continues to hold if the problem is restricted to graphs of degree ≤ 4: the reduction of #SAT to #UHAMPATH given in [Sip05, Theorems 7.35 and 7.36] with the input formula restricted to exactly three literals per clause yields a graph of degree 4 whose number of Hamiltonian paths coincides with the number of satisfying assignments of the original formula.

We prove that there exists a weakly parsimonious reduction from #UHAMPATH restricted to graphs of degree ≤ 4 to #S-T PATHS. To this end, let $G = (V, E)$ be an undirected graph, let $s \neq t$ be nodes in V, and assume without loss of generality that the degree of G is bounded by 4. Fix $n := 3|V|$. We define $G' = (V', E')$ as the graph obtained from G by adding a node t' to V and edges $\{s, t'\}$, $\{t, t'\}$ to E. Denote by $p(G, s, t)$ the number of simple paths from s to t in G and by $p_k(G, s, t)$ the number of simple paths from s to t of length k. Then $p_1(G', s, t') = 1$ and $p_{k+1}(G', s, t') = p_k(G, s, t)$ for all $k > 0$. Finally, transform G' to G'' by replacing each edge $\{u, v\} \in E'$ with a copy of the graph $G_{exp} := (V_{exp}, E_{exp})$ and the connecting edges $\{u, x_0\}$, $\{x_n, v\}$, where

$$V_{exp} := \{x_i \mid 0 \leq i \leq n\} \cup \{y_i, y'_i \mid 0 \leq i < n\},$$
$$E_{exp} := \big\{\{x_i, y_i\}, \{y_i, x_{i+1}\}, \{x_i, y'_i\}, \{y'_i, x_{i+1}\} \,\big|\, 1 \leq i < n\big\}.$$

This substitution is indicated in Figure 6.1. Each path from s to t' of length k in

G' corresponds to $2^{k\cdot n}$ simple paths from s to t' in G''. Thus the number of path from s to t' in G'' is

$$p(G'',s,t') = \sum_{k=1}^{|V'|-1} 2^{k\cdot n} p_k(G',s,t') = \sum_{k=1}^{|V|-1} 2^{(k+1)\cdot n} p_k(G,s,t) + 2^n.$$

As the degree of G is bounded by 4, we obtain that $p_k(G,s,t) \leq 4^{k-1}$. Hence, for each $1 < i \leq |V|$,

$$\sum_{k=1}^{i-1} 2^{k\cdot n} p_k(G',s,t') < 2^{(i-1)\cdot n} 4^{|V|} < 2^{(i-1)\cdot n} \cdot 2^n = 2^{i\cdot n}.$$

Therefore the number of Hamiltonian paths in G from s to t is can be computed from $p(G',s,t')$ by dividing with $2^{(|V|+1)\cdot n} = 2^{\frac{1}{3}n^2+n}$, where n equals the number of zeros from the least significant bit to the position of the first 1 in the binary encoding of $p(G'',s,t')$. $\square$

Lemma 6.3.3 *Let B be a finite set of Boolean functions. Then* #CIRC$(B \cup \{1\})$ *reduces to* #CIRC(B) *via parsimonious reductions for all B, and* #CIRC$(B \cup \{0\})$ *reduces to* #CIRC(B) *via subtractive reductions for all B such that $\vee \in [B]$.*

Proof. Let B be a finite set of Boolean functions. Let $\Gamma \subseteq \mathcal{L}(B)$ and a partition (P,Q,Z) of the set of propositions be given. For the first claim, transform Γ to $\Gamma' := \Gamma_{[1/t]} \cup \{t\}$ and map (P,Q,Z) to $(P \cup \{t\},Q,Z)$ as in the proof of Lemma 5.3.4. Then each (P,Q,Z)-minimal model σ of Γ corresponds to the $(P \cup \{t\},Q,Z)$-minimal model $\sigma' := \sigma \cup \{t\}$ and vice versa. Thus #CIRC$(B \cup \{1\})$ parsimoniously reduces to #CIRC(B).

For the second claim, define Γ' as $\{\varphi \vee f \mid \varphi \in \Gamma\}$. Then an assignment σ is a minimal model of Γ' if and only if either $\sigma(f) = 0$ and σ is a minimal model of Γ or $\sigma(f) = 1$ and $\sigma(x) = 0$ for all $x \in P$. Thus the functions

$$\begin{aligned} g\big((\Gamma,(P,Q,Z))\big) &:= \big(\Gamma',(P \cup \{f\},Q,Z)\big), \\ h\big((\Gamma,(P,Q,Z))\big) &:= \big(\{f\},(\{f\},Q,Z)\big) \end{aligned}$$

constitute a subtractive reduction from #CIRC$(B \cup \{0\})$ to #CIRC(B): any assignment that satisfies $h\big((\Gamma,(P,Q,Z))\big)$ also satisfies $g\big((\Gamma,(P,Q,Z))\big)$, and

$$\begin{aligned} |MM(\Gamma',(P \cup \{f\},Q,Z))| &= 2^{|Q|+|Z|} + |MM(\Gamma,(P,Q,Z))|, \\ |MM\big(\{f\},(\{f\},Q,Z)\big)| &= 2^{|Q|+|Z|}, \end{aligned}$$

where $MM\big(\Gamma,(P,Q,Z)\big)$ denotes the set of (P,Q,Z)-minimal models of Γ. $\square$

Lemma 6.3.4 *Let B be a finite set of Boolean functions such that $[B] = \mathsf{M}$. Then* #CIRC(B) *is* #P*-complete with respect to subtractive reductions.*

Proof. Membership in #P is obvious from the fact that one can check in polynomial time whether an assignment is a minimal model (see Lemma 5.3.5).

As for the #P-hardness, we give a subtractive reduction from #SAT to #CIRC(B) for all B such that $[B] = \mathsf{M}$. Hence, let φ be a propositional formula in conjunctive normal form. Assume without loss of generality that $\mathrm{Vars}(\varphi) = \{x_1, \dots, x_n\}$ and denote this set by X. Let $Y = \{y_1, \dots, y_m\}$ be a set of propositions disjoint from X. Now define φ' to be the formula derived from φ by replacing all negative literals $\neg x_i$ by y_i, and let

$$\Gamma := \Big\{\varphi', \bigwedge_{1 \le i \le n} (x_i \vee y_i)\Big\}, \qquad \Gamma' := \Gamma \cup \Big\{ \bigvee_{1 \le i \le n} (x_i \wedge y_i)\Big\}.$$

Then the set of assignments $\sigma\colon X \to \{0,1\}$ satisfying φ can be characterized as

$$\begin{aligned}\{\sigma \mid \sigma \models \varphi\} &= \{\sigma\colon X \cup Y \to \{0,1\} \mid \sigma \models \varphi' \wedge \textstyle\bigwedge_{1 \le i \le n}(x_i \oplus y_i)\} \\ &= \{\sigma\colon X \cup Y \to \{0,1\} \mid \sigma \models \Gamma, \sigma \not\models \Gamma'\} \\ &= \{\sigma\colon X \cup Y \to \{0,1\} \mid \sigma \models \Gamma\} \setminus \{\sigma\colon X \cup Y \to \{0,1\} \mid \sigma \models \Gamma'\}.\end{aligned}$$

Define the functions

$$f(\varphi) := (\Gamma, (X \cup Y, \emptyset, \emptyset)), \qquad g(\varphi) := (\Gamma', (X \cup Y, \emptyset, \emptyset)).$$

We claim that f and g constitute a subtractive reduction from #SAT to #CIRC(B) for any finite set B such that $[B] = \mathsf{M}$.

Suppose that $\sigma\colon X \to \{0,1\}$ is a model of φ. Define σ' as the extension of σ setting $\sigma(y_i) = 1 - \sigma(x_i)$. Clearly, $\sigma' \models \Gamma$ and $\sigma' \not\models \Gamma'$. As all models of Γ that set to 1 exactly one of x_j and y_j for all $1 \le j \le m$ are mutually incomparable with respect to $\le_{(P,Q,Z)}$, we obtain that $\sigma \models^{\mathrm{circ}}_{(P,Q,Z)} \Gamma$. Thus the mapping $\sigma \mapsto \sigma'$ is an injective embedding from the models of φ to the minimal models of Γ that do not satisfy Γ'. Now if σ is a (P, Q, Z)-minimal model of Γ that sets to 1 both x_i and y_i for some $1 \le i \le n$, then $\sigma \models^{\mathrm{circ}}_{(P,Q,Z)} \Gamma'$. Consequently, $\sigma \mapsto \sigma'$ is onto the set of (P, Q, Z)-minimal models of Γ that do not satisfy Γ'. This proves the claim.

Finally, Lemma 2.4.2 (2.) yields the #P-hardness of #CIRC(B) for all finite sets B such that $[B] = \mathsf{M}$. $\square$

Lemma 6.3.5 *Let B be a finite set of Boolean functions such that $[B] = \mathsf{V}$. Then* #CIRC(B) *is* #P*-complete with respect to weakly parsimonious reductions.*

Proof. Let #PERFECTMATCHING denote the problem to count the number of perfect matchings (that is, sets of edges such that each node is incident to exactly one edge in the set) in a bipartite graph. It is well known that #PERFECTMATCHING is #P-complete via parsimonious reductions [Val79a]. We give a weakly parsimonious reduction from #PERFECTMATCHING to #CIRC$(\{\vee\})$ such that the arity of all disjunctions is bounded by a constant. This establishes the lemma.

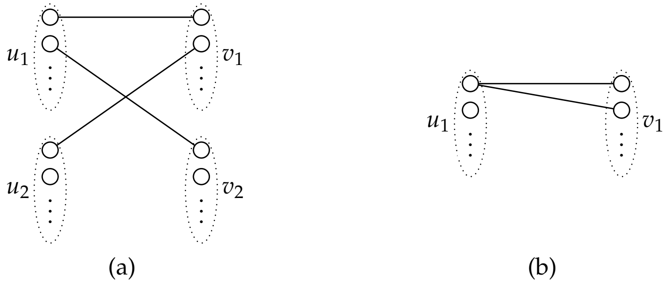

Figure 6.2: Forbidden patterns in matchings of G_1.

Let $G = (V, E)$ be the given graph with $V = \{u_1, \ldots, u_n, v_1 \ldots, v_n\}$. Assume without loss of generality that $n \geq 1$. We transform G in five steps to a set Γ' of disjunctions such that the number of perfect matchings in G can be computed from the number of minimal models of Γ', where all propositions are subject to minimization (that is, $P := \mathrm{Vars}(\Gamma)$, $Q := \emptyset$, $Z := \emptyset$). This transformation first modifies G such that the number of maximal matchings with k edges is multiplied with a factor $c(k)$, and next computes from the result its corresponding edge graph whose minimal vertex covers correspond to the maximal matchings in G. Into that edge graph, we then encode the number of vertices of G and finally transform the outcome to a set of disjunctions.

In the first step, construct $G_1 = (V_1, E_1)$ with

$$\begin{aligned} V_1 &:= \{u_i^j, v_i^j \mid 1 \leq i \leq n, 1 \leq j \leq 2n\}, \\ E_1 &:= \{(u_i^k, v_j^\ell) \mid (u_i, v_j) \in E, 1 \leq k, \ell \leq 2n\}, \end{aligned}$$

analogously to the reduction from #PERFECTMATCHING to #PRIMEIMPLICANT in [Val79b]. Note that there exist matchings in G_1 that do not correspond to matchings in G (see Figure 6.2 (a)); these matchings will be taken care of in the next step.

Secondly, transform G_1 into a graph $G_2 = (V_2, E_2)$, whose nodes represent the edges of G_1 and that contains an edge between two nodes if their corresponding edges in G_1 share a node (as shown in Figure 6.2 (b)) or they arise from distinct edges in G_1 that share a node in G (as shown in Figure 6.2 (a)); that is,

$$\begin{aligned} V_2 &:= \{x_{(u_i^k, v_j^\ell)} \mid (u_i^k, v_j^\ell) \in E_1\}, \\ E_2 &:= \Big\{ \{x_{(u_{i_1}^{k_1}, v_{j_1}^{\ell_1})}, x_{(u_{i_2}^{k_2}, v_{j_2}^{\ell_2})}\} \,\Big|\, u_{i_1}^{k_1} = u_{i_2}^{k_2} \text{ or } v_{j_1}^{\ell_1} = v_{j_2}^{\ell_2} \Big\} \cup \\ &\qquad \Big\{ \{x_{(u_{i_1}^{k_1}, v_{j_1}^{\ell_1})}, x_{(u_{i_2}^{k_2}, v_{j_2}^{\ell_2})}\} \,\Big|\, (i_1 = i_2 \text{ and } j_1 \neq j_2) \text{ or } (i_1 \neq i_2 \text{ and } j_1 = j_2) \Big\}. \end{aligned}$$

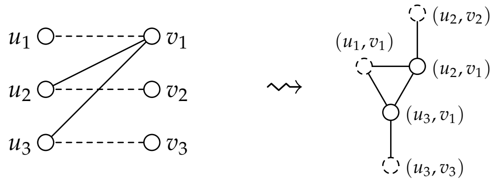

Figure 6.3: Bipartite graph G and its corresponding edge graph G'. Dashed edges in G form a perfect matching, solid vertices in G' a minimal vertex cover.

Say that a matching is *maximal* if no edge can be added to it without violating the matching property. Suppose that M is a matching in G_1 that corresponds to a maximal matching in G. Then the set of nodes in G_2 whose corresponding edges do *not* belong to M yield a minimal vertex cover, that is, $C := V_2 \setminus M$ is a minimal set of nodes such that $C \cap e \neq \emptyset$ for all edges $e \in E_2$ (see Figure 6.3 for an example). Conversely, any minimal vertex cover C of G_2 is the image of a matching in G_1 that corresponds to a maximal matching in G. As for each edge in a maximal matching in G we may in G_1 choose from $(2n)!$ possible pairs of end-points, the number of minimal vertex covers of size i in G_2 is equal to $\big((2n)!\big)^i$ times the number of maximal matchings in G.

Thirdly, add to G_2 nodes w_1, w_2, w_3 and edges $\{w_1, w_2\}, \{w_2, w_3\}, \{w_3, w_1\}$. Call the resulting graph $G_3 = (V_3, E_3)$. Each vertex cover of size i in G_2 corresponds to three vertex covers of size $i + 2$ is G_3.

Fourthly, transform G_3 to the set Γ of $\{\vee\}$-formulae defined as

$$\Gamma := \big\{x_{\{u,v\}} \vee x_{\{u',v'\}} \vee y \,\big|\, \{x_{\{u,v\}}, x_{\{u',v'\}}\} \in E_3\big\}.$$

Any minimal model of Γ is either a minimal vertex cover of G_3 or it is an assignment setting to 1 only y. In particular, any minimal model of Γ aside the assignment $\{y\}$ sets to 1 at least two propositions, as any vertex cover of G_3 contains at least two nodes.

Finally, define Γ' as

$$\Gamma' := \Gamma \cup \{z_i \vee z_i' \mid 1 \leq i \leq n\}.$$

As a result, to each minimal model of Γ there correspond 2^n minimal models Γ'. This concludes the transformation of the input.

To summarize the above, let $m_i(\Gamma)$ denote the number of minimal models of Γ that have exactly i propositions set to 1, let $v_i(G)$ denote the number of minimal vertex covers of size i in the graph G, and let $t_i(G)$ denote the number of maximal matchings of size i in the graph G. Then the number of minimal models of Γ is equal to

$$\sum_{i=0}^{k} m_i(\Gamma') = 2^n \cdot \Big(\sum_{i=0}^{k} m_i(\Gamma) \Big) = 2^n \cdot \Big(1 + \sum_{i=2}^{k} 3 \cdot v_{i-2}(G_2) \Big)$$
$$= 2^n \cdot \Big(1 + \sum_{i=0}^{n} 3 \cdot \big((2n)!\big)^i \cdot t_i(G) \Big),$$

where $k \leq n^c$ for some $c \in \mathbb{N}$.

Given the number x of minimal models of Γ', we can thus obtain n from the number of zeros from the least significant bit to the position of the first 1 in the binary encoding of x. We are then able to compute from x the value of

$$\sum_{i=0}^{n} \big((2n)!\big)^i \cdot t_i(G). \tag{6.1}$$

As $(2n)! \geq 2(j-1)!$ for all $1 \leq j \leq n$, we moreover know that

$$\sum_{i=0}^{j-1} \big((2n)!\big)^i \cdot t_i(G) \leq \big((2n)!\big)^{j-1} \cdot (j-1)! \cdot \sum_{i=0}^{j-1} \frac{1}{2^i}$$
$$< 2 \cdot \big((2n)!\big)^{j-1} \cdot (j-1)!$$
$$\leq \big((2n)!\big)^j.$$

Hence, the number $t_n(G)$ of perfect matchings in G can be obtained as the integer part of the quotient of equation (6.1) and $\big((2n)!\big)^n$. □

Proof of Theorem 6.3.1 For $B = \{\wedge, \vee, \neg\}$, the #·coNP-completeness of $\#\text{CIRC}(B)$ has been shown in [DHK05]. As $[\mathsf{D}_1 \cup \{1\}] \supseteq \mathsf{S}_{02}$, $[\mathsf{S}_{12} \cup \{1\}] \supseteq \mathsf{S}_{02}$, and $[\mathsf{S}_{02} \cup \{0,1\}] = \mathsf{BF}$, the #·coNP-completeness of $\#\text{CIRC}(B)$ for $\mathsf{S}_{02} \subseteq [B]$ or $\mathsf{S}_{12} \subseteq [B]$ or $\mathsf{D}_1 \subseteq [B]$ follows from Lemma 6.3.3.

Similarly, to establish the #P-completeness of $\#\text{CIRC}(B)$ in the cases $\mathsf{S}_{00} \subseteq [B] \subseteq \mathsf{M}$, $\mathsf{S}_{10} \subseteq [B] \subseteq \mathsf{M}$, and $\mathsf{D}_2 \subseteq [B] \subseteq \mathsf{M}$, it suffices to prove #P-completeness for all B such that $[B] = \mathsf{M}$. This has been shown in Lemma 6.3.4.

As for the #P-completeness of $\#\text{CIRC}(B)$ in the cases $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$ and $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$, it remains to show the #P-hardness via weakly parsimonious reductions. In the former case,the result follows from Lemma 6.3.5, whereas in the latter it follows from Lemma 6.3.2 and [DH08, Theorem 4].

Finally, if B is such that $[B] \subseteq \mathsf{N}$ or $[B] \subseteq \mathsf{E}$, then the (P, Q, Z)-minimal models are uniquely determined on the set of propositions $\text{Vars}(\Gamma) \cup P$. Hence, the number of (P, Q, Z)-minimal models of a given set $\Gamma \subseteq \mathcal{L}(B)$ is $2^{|(Q \cup Z) \setminus \text{Vars}(\Gamma)|}$. □

Remark 6.3.6 *It is apparent from the above proofs that the complexity of* $\#\text{CIRC}(B)$ *remains unchanged if its input is restricted to* $Q = Z = \emptyset$*. This exhibits another case, where the counting problem is* #P*-complete while underlying decision problem is tractable, namely for the clones* $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$.

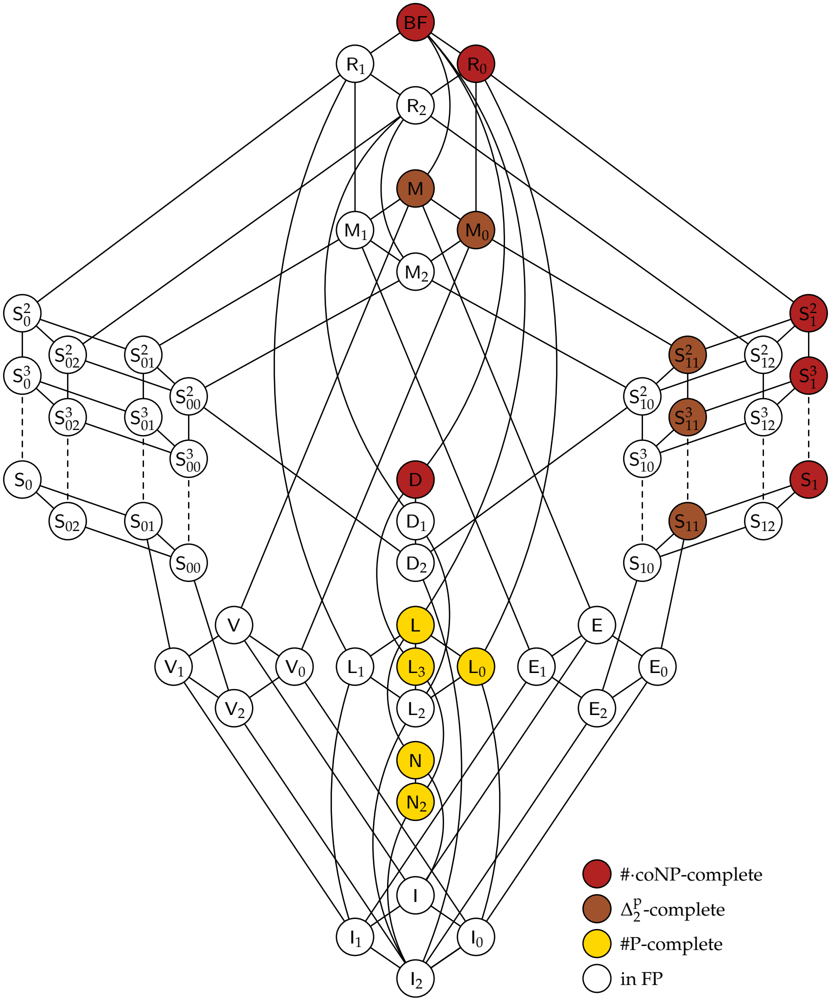

Figure 6.4: The complexity of #EXT(B)

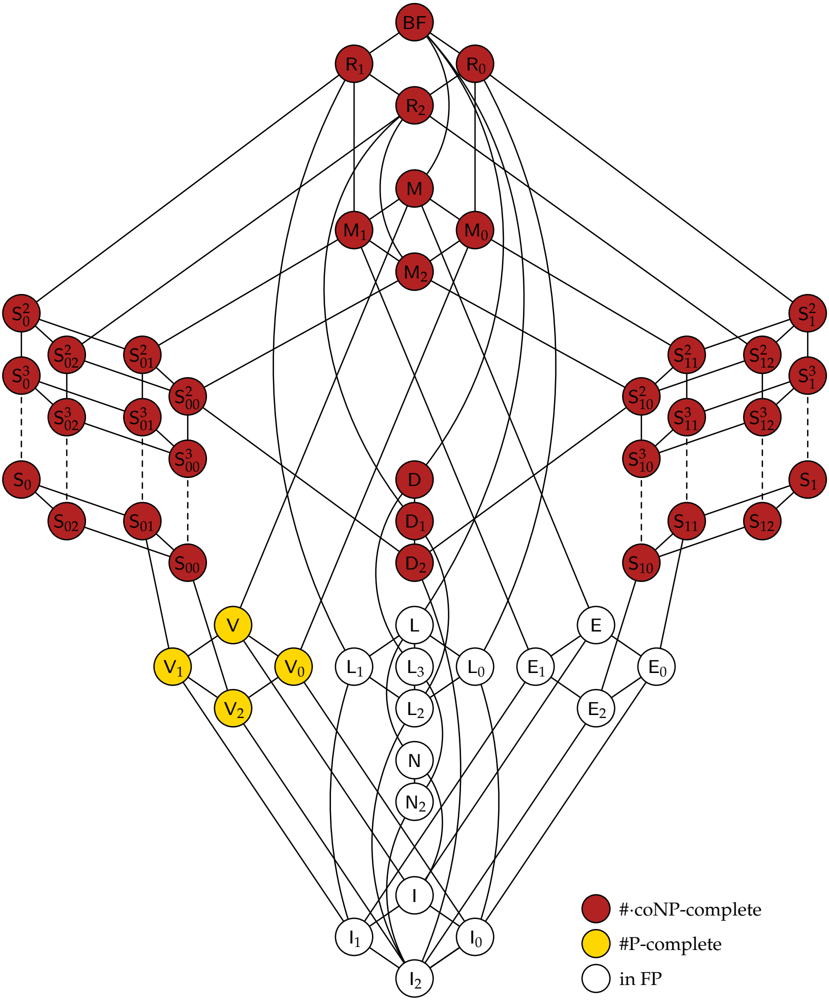

Figure 6.5: The complexity of #EXP(B)

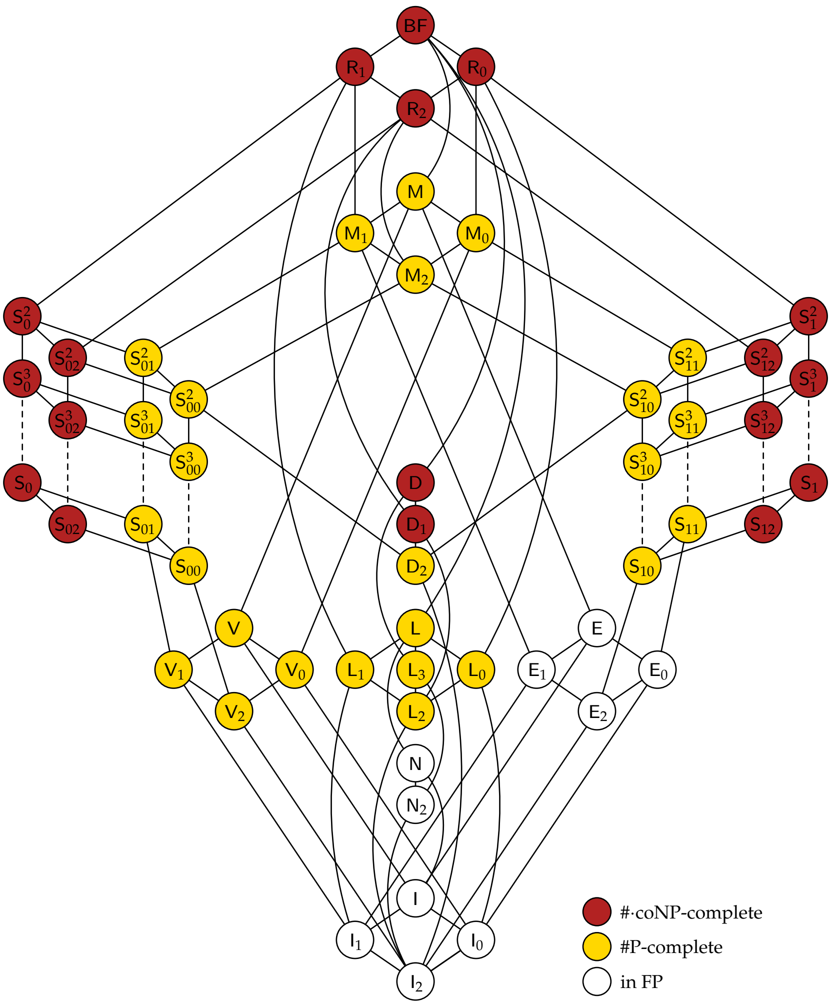

Figure 6.6: The complexity of #CIRC(B)

CHAPTER 7

TRANSLATIONS

Having determined the computational complexity of the decision and counting problems arising in the context of default logic, autoepistemic logic, and circumscription, we will now investigate their relationship. In particular, we will study the possibility of translating between fragments of these logics.

Default and autoepistemic logic—though semantically different—are both based on the principle of defining nonmonotonicity by means of a fixed-point equation that describes possible sets of knowledge or beliefs. Circumscription contrasts this approach by restricting the semantics of classical logic to minimal models. Owing to these differences, out notion of translations will be based on the set of skeptically entailed formulae. Although for translations between default logic and autoepistemic logic, one could also consider the set of credulously entailed formulae for translations between default and autoepistemic logic (called *faithful translations*): while the picture for translations from default logic to autoepistemic remains almost unchanged, differences regarding the inconsistent stable expansions yield non-translatability results in the converse direction. This is due to the ability of an autoepistemic theory to possess the inconsistent stable expansion aside several consistent ones. For default logic admitting the inconsistent stable extension rules out the existence of consistent ones.

A property worth mentioning is that of *modular translations* [Imi87]. A translation is called *modular* if the addition of facts to the knowledge base does not require the recomputation of its image, instead the image of the modified theory is obtained by adjoining to the old image the translation of the added fact. Under the assumption that the "nonmonotonic part" of a given knowledge base is largely invariant while the objective part may be subject to more frequent changes, modular translations are highly desirable from the computational point of view.

A first investigation of the relationship between default and autoepistemic logic has been carried out by Konolige [Kon88], who showed that default logic and *strongly grounded autoepistemic logic*, a more restrictive variant of autoepistemic logic, are equivalent. The second major approach of relating default logic and autoepistemic logic has been taken by Marek and Truszczyński [MT89]. They showed that default logic can be embedded into a wide range of nonmonotonic modal logics that are based on an approach introduced by McDermott and Doyle [MD80, McD82]. Finally, Gottlob [Got95b] showed that if no additional propositional atoms are to be used then no modular translation from default logic to autoepistemic logic is possible, but that there exists a faithful translation via the nonmonotonic version of the *pure logic necessitation* [MT90]. This result was reproven shortly afterwards using a purely model-theoretic approach by Schwarz [Sch96].

As regards translations to or from circumscription, less results can be found in the literature. Konolige [Kon89] showed that there exists a faithful and modular translation from circumscription to autoepistemic logic. Imielinski [Imi87] proved that default logic cannot be embedded into circumscription by a faithful and modular translation but that there exist restricted types of default theories that admit such translations. These results, among others, are also summarized in [Eth87].

Lastly, Janhunen studied the intertranslatability of these three logics in [Jan99]. Disregarding the inconsistent stable extension, the inconsistent stable expansion, and allowing the use of new propositions, he is able to show that with respect to polynomial-time, faithful and modular translations propositional logic is strictly less expressive than circumscription, which is strictly less expressive than autoepistemic logic, which is again strictly less expressive than default logic, which is equivalent to strongly grounded autoepistemic logic. At first sight, this result seems to contradict the result by Gottlob, but it crucially relies on his weaker notion of faithfulness.

Here, we take the opposite approach and study the intertranslatability of these nonmonotonic logics with respect to a very weak notion of translations, namely polynomial-time transformations that leave invariant the set of skeptically entailed formulae. We prove that with respect to this notion, default logic and autoepistemic logic are equally expressive. To be more precise, autoepistemic logic and default logic admit translations into each other for functional complete sets of Boolean functions and for fragments that admit efficient computation of respectively stable extensions and stable expansions. In addition to that, monotone autoepistemic logic embeds monotone default logic and, quite remarkably, disjunctive autoepistemic logic can be embedded into the fragments of default logic containing negation as the sole Boolean connective.

Concerning translations of circumscription into the above two logics, we show that, though translations into both full default as well as full autoepistemic logic are possible, the results for fragments of these logic differ significantly. While default logic modularly embeds circumscription whenever $\neg$ is available and all Boolean functions used in the source logic can be simulated, the analogous statement for autoepistemic is more restrictive: a translation from circumscription into a not functional complete fragment of autoepistemic logic exists only if any circumscriptive theory is equivalent to a set of literals. Therefore, while both autoepistemic logic and default logic are capable of embedding circumscription, the presence of negation alone is enough for default logic to subsume the corresponding fragment of circumscription.

In the converse direction, translations from default logic or autoepistemic logic to circumscription are only possible for very restricted sets of Boolean functions, namely those for which the skeptical reasoning problem is tractable. These results confirm the intuition that—even under the weak notion of translations considered herein—circumscription is less expressive than autoepistemic logic or default logic, not only for the full fragment but also the fragments obtained by restricting the set of available Boolean functions.

Finally, for almost all fragments for which no translation is given, we show that no translation is possible unless the polynomial hierarchy collapses.

The rest of this chapter is structured as follows. The first section defines the notion of translations. In the remaining three sections, we study for each pair of the considered logics the possibility of translating between them. Each of these sections contains two theorems, one for each direction; the proofs of these theorems will be established from the lemma following them.

7.1 Preliminaries

For a finite set B of Boolean functions, write *B-default logic* to denote default logic restricted to B-default theories. Analogously, define *B-autoepistemic logic* and *B-circumscription* as respectively autoepistemic logic restricted to sets of autoepistemic B-formulae and circumscription restricted to sets of B-formulae.

Definition 7.1.1 (Translations)

1. *A* translation from default logic to autoepistemic logic *is a function f mapping any finite default theory (W,D) to a finite set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}$ such that $(W,D) \models^{\mathrm{skep}} \varphi$ if and only if $\Sigma \models^{\mathrm{skep}} \varphi$ for all $\varphi \in \mathcal{L}$ over variables from (W,D).*
2. *A* translation from autoepistemic logic to default logic *is a function f mapping any finite set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}$ to a finite default theory (W,D) such that $\Sigma \models^{\mathrm{skep}} \varphi$ if and only if $(W,D) \models^{\mathrm{skep}} \varphi$ for all $\varphi \in \mathcal{L}$ over variables from Σ.*
3. *A* translation from circumscription to default logic (respectively autoepistemic logic) *is a function f mapping any pair of a finite set Γ of formulae and disjoint sets (P,Q,Z) of propositions such that* $\mathrm{Vars}(\Gamma) \subseteq P \cup Q \cup Z$ *to a finite default theory (respectively finite set of autoepistemic formulae) such that $\Gamma \models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$ if and only if $f\big((\Gamma,(P,Q,Z))\big) \models^{\mathrm{skep}} \varphi$ for all $\varphi \in \mathcal{L}$ over variables from $P \cup Q \cup Z$.*
4. *A* translation from default logic (respectively autoepistemic logic) to circumscription *is a function f mapping any finite default theory (W,D) (respectively finite set Σ of autoepistemic formulae) to a finite set Γ of formulae and a disjoint sets (P,Q,Z) of propositions such that* $\mathrm{Vars}(\Gamma) \subseteq P \cup Q \cup Z$ *and $(W,D) \models^{\mathrm{skep}} \varphi$ (respectively $\Sigma \models^{\mathrm{skep}} \varphi$) if and only if $\Gamma \models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$ for all $\varphi \in \mathcal{L}$ over variables from (W,D) (respectively Σ).*

That is, translations are mappings between nonmonotonic logics, that preserve the skeptical semantics of the given input. Please note that the above definition allows the usage of new variables in translations. It is hence a relaxation of the notion used in [Got95b, Sch96, Jan99] and subsumes the notion of translations used in [Imi87, Kon89]. In what follows, we will study the possibility of translating between fragments of the nonmonotonic formalisms considered in this thesis.

7.2 Default Logic and Autoepistemic Logic

We will first study the relationship between fragments of default logic and fragments of autoepistemic logic. Our first theorem proves that translations from B-default logic to B'-autoepistemic logic are, with the exception of one open case, only possible either if the set $B' \cup \{0,1\}$ is functional complete, or if $B' \cup \{0,1\}$ contains all monotone functions and implements all functions in B, or if any B-default theory possesses at most one efficiently computable stable extension.

Theorem 7.2.1 *Let B and B' be finite sets of Boolean functions such that $[B \cup \{0,1\}] \neq \mathsf{V}$ or $[B' \cup \{0,1\}] \neq \mathsf{L}$. Then there exists translation from B-default logic to B'-autoepistemic logic*

1. *if $[B' \cup \{0,1\}] = \mathsf{BF}$, or*
2. *if $[B] \subseteq \mathsf{M}$ and $[B] \subseteq [B' \cup \{0,1\}]$, or*
3. *if $[B] \subseteq \mathsf{L}_1$ and $[B] \subseteq [B' \cup \{0,1\}]$, or*
4. *if $[B] \subseteq \mathsf{E}$;*

unless $\mathrm{P} = \mathrm{NP}$, *no other translations are possible.*

The proof of Theorem 7.2.1 will be established from the following four lemmas. The first of which follows from [Got95b] (or equivalently from [Sch96]) together with Lemma 2.4.2 (1.) and the fact that any Boolean formula over $\{\wedge, \vee, \neg\}$ can be restructured to be of polynomial size and logarithmic depth [Spi71].

Lemma 7.2.2 ([Got95b]) *Let B and B' be finite sets of Boolean functions such that $[B' \cup \{0,1\}] = \mathsf{BF}$. Then there exists a translation from B-default logic to B'-autoepistemic logic.*

Lemma 7.2.3 *Let B and B' be finite sets of Boolean functions such that $\mathsf{S}_{00} \subseteq [B \cup \{1\}] \subseteq \mathsf{M}$ and $[B] \subseteq [B' \cup \{0,1\}]$. Then there exists a translation from B-default logic to B'-autoepistemic logic.*

Proof. Let B and B' be finite sets of Boolean functions satisfying the requirements of the lemma, and denote by (W, D) the given B-default theory. By Lemma 4.1.3, (W, D) possesses at most one stable extension. Define

$$\Sigma := W \cup \left\{ L\alpha \vee p_\alpha, Lp_\alpha \vee \gamma \;\middle|\; \frac{\alpha : \beta}{\gamma} \in D \text{ and } \beta \text{ is satisfiable} \right\}$$

for fresh, mutually different propositions p_α. We define the translation function f to be the mapping $(W, D) \mapsto \Sigma'$, where Σ' denotes the B'-representation of Σ.

To see that f is indeed polynomial-time computable, recall that the consistency of a set of monotone formulae is decidable in polynomial time and that, for $\mathsf{M} \subseteq [B' \cup \{0,1\}]$, B efficiently implements $\wedge$ and $\vee$ by Lemma 2.4.2.

As for the correctness of the translation, first suppose that (W, D) has a stable extension E. From Theorem 3.1.3, it follows that there exists an ordering $\delta_1, \ldots, \delta_n$ of the defaults in $\mathrm{GD}(E)$ such that for all $0 \leq i \leq n$ and

$$\Lambda_i := \left\{ L\alpha_j, \neg Lp_{\alpha_j} \,\middle|\, \delta_j = \frac{\alpha_j : \beta_j}{\gamma_j} \text{ and } j \leq i \right\}$$

we obtain $\Sigma \cup \Lambda_{i-1} \models \alpha_i$ and $\Sigma \cup \Lambda_{i-1} \cup \{L\alpha_i\} \not\models p_{\alpha_i}$. As the p_{α_i}'s do not occur in any formula except $L\alpha_i \vee p_{\alpha_i}$ and $Lp_{\alpha_i} \vee \gamma$, this eventually leads to $\Sigma \cup \Lambda_n \models \alpha$ if and only if $L\alpha \in \Lambda_n$ and, $\Sigma \cup \Lambda_n \cup \{L\alpha\} \not\models p_\alpha$ if and only if $\neg Lp_\alpha \in \Lambda_n$, for all premises α in $\mathrm{GD}(E)$. As $W \cup \mathrm{GD}(E) \not\models \alpha$ for all $\frac{\alpha:\beta}{\gamma} \in D \setminus \mathrm{GD}(E)$, setting

$$\Lambda := \Lambda_n \cup \left\{ \neg L\alpha, Lp_\alpha \,\middle|\, \frac{\alpha : \beta}{\gamma} \in D \setminus \mathrm{GD}(E) \right\}.$$

we obtain $\Sigma \cup \Lambda \models \alpha$ if and only if $L\alpha \in \Lambda$ and, $\Sigma \cup \Lambda \cup \{L\alpha\} \not\models p_\alpha$ if and only if $\neg Lp_\alpha \in \Lambda$. Thus, Λ is a Σ-full set. Although Σ may possess more than one Σ-full set, the presence of Λ suffices to establish the translation: On the one hand, for all $\varphi \in \mathcal{L}$ over propositions from (W, D) it holds that

$$\begin{aligned}
&(W, D) \models^{\text{skep}} \varphi \\
\iff &W \cup \left\{ \gamma \,\middle|\, \tfrac{\alpha:\beta}{\gamma} \in \mathrm{GD}(E) \right\} \models \varphi \\
\iff &\Sigma \cup \left\{ L\alpha, \neg Lp_\alpha \,\middle|\, \tfrac{\alpha:\beta}{\gamma} \in \mathrm{GD}(E) \right\} \cup \left\{ \neg L\alpha, Lp_\alpha \,\middle|\, \tfrac{\alpha:\beta}{\gamma} \in D \setminus \mathrm{GD}(E) \right\} \models \varphi \\
\iff &\Sigma \cup \Lambda \models \varphi.
\end{aligned}$$

On the other hand, any Σ-full set Λ' has to satisfy $\Lambda_n \subseteq \Lambda'$. Thus $\Sigma \cup \Lambda \models \varphi \implies \Sigma \cup \Lambda' \models \varphi$ for all $\varphi \in \mathcal{L}$ over propositions from (W, D), due to the monotonicity of $\models$ and $p_\alpha \notin \mathrm{Vars}(\varphi)$ for all such propositions p_α. Therefore, $(W, D) \models^{\text{skep}} \varphi$ if and only if $f((W, D)) \models^{\text{skep}} \varphi$ for all $\varphi \in \mathcal{L}$ over propositions from (W, D).

Finally, suppose that (W, D) does not have a stable extension. Then there has to exist an applicable default $\frac{\alpha:\beta}{\gamma} \in D$, whose conclusion is equivalent to 0. By construction, Σ then contains a formula that is equivalent to Lp_α. Since the only other occurrence of p_α in Σ is in $L\alpha \vee p_\alpha$, $L\alpha$ has to be 0. However, the applicability of $\frac{\alpha:\beta}{\gamma}$ implies that α can be derived. Therefore, any Σ-full set has to contain $L\alpha$. Hence, no consistent stable expansion of Σ may exist. □

Lemma 7.2.4 *Let B and B' be finite sets of Boolean functions such that $[B] \subseteq \mathsf{V}$ and $[B] \subseteq [B' \cup \{0,1\}]$, or such that $[B] \subseteq \mathsf{L}_1$ and $[B] \subseteq [B' \cup \{0,1\}]$, or such that $[B] \subseteq \mathsf{E}$. Then there exists a translation from B-default logic to B'-autoepistemic logic.*

Proof. Let B and B' be finite sets of Boolean functions satisfying the requirements of the lemma. Then any B-default theory possesses at most one stable extension (see Lemma 4.1.3), whose existence can be efficiently decided by virtue of Theorem 4.1.1. Consequently, the function

$$f((W,D)) := \begin{cases} W \cup \left\{ \gamma \,\middle|\, \frac{\alpha:\beta}{\gamma} \in \mathrm{GD}(E) \right\}, & \text{if } (W,D) \text{ has a stable extension } E, \\ W & \text{otherwise,} \end{cases}$$

can be computed in polynomial time. Moreover, all Boolean functions in $[B]$ are associative. We may thus insert parentheses into any set V of B-formulae to obtain a set V' such that the nesting depth of all contained formulae is logarithmic. Let g be the function that maps V to V' and subsequently replaces all Boolean functions from B with their B'-representations (in particular, replacing $\varphi_i \wedge \cdots \wedge \varphi_n$ with $\varphi_1, \ldots, \varphi_n$). Then $f \circ g$ is the desired translation. □

Lemma 7.2.5 *Let B and B' be finite sets of Boolean functions such that*

1. $[B' \cup \{0,1\}] \in \{\mathsf{M}, \mathsf{E}, \mathsf{V}, \mathsf{L}, \mathsf{N}, \mathsf{I}\}$,
2. $[B] \subseteq \mathsf{M}$ *implies* $[B] \nsubseteq [B' \cup \{0,1\}]$,
3. $[B] \subseteq \mathsf{V}$ *implies* $[B' \cup \{0,1\}] \neq \mathsf{L}$,
4. $[B] \subseteq \mathsf{L}_1$ *implies* $[B] \nsubseteq [B' \cup \{0,1\}]$, *and*
5. $[B] \nsubseteq \mathsf{E}$.

Then there exists no translation from B-default logic to B'-autoepistemic logic unless $\mathrm{P} = \mathrm{NP}$.

Proof. Let B and B' be as in the statement of the lemma. We distinguish between the possible clones of $[B' \cup \{0,1\}]$.

If $[B' \cup \{0,1\}] = \mathsf{M}$, then we have to prove that no translation exists for $\mathsf{S}_{02} \subseteq [B \cup \{1\}]$ and $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$. In the first case, $(W,D) := (\{x \vee (t \wedge \neg y), t\}, \emptyset)$ cannot be translated to a set Σ of autoepistemic B'-formulae. Suppose it could, and let Λ denote the kernel of a consistent stable expansion of Σ. If $\Sigma \cup \Lambda \equiv 1$, then $\Sigma \models^{\mathrm{skep}} \varphi$ if and only if φ is tautological. Hence, we may without loss of generality assume that $\Sigma \cup \Lambda \not\equiv 1$. Then, $\Sigma \cup \Lambda \models_L \varphi$ if φ is contained in a stable extension E of (W,D). As φ is propositional, this simplifies to $\Sigma \cup \Lambda \models \varphi$. But $\Sigma \cup \Lambda$ is equivalent to a monotone formula and therefore 1-reproducing, while $\{x \vee (t \wedge \neg y), t\} \equiv \{x \rightarrow y\}$ is not. Consequently, $\Sigma \not\models^{\mathrm{skep}} x \rightarrow y$; contradictory to $(W,D) \models^{\mathrm{skep}} x \rightarrow y$.

In the second case, $(W,D) := (\{x_1 \oplus x_2 \oplus x_3\}, \emptyset)$ cannot be translated to a set Σ of autoepistemic B'-formulae either. By similar arguments, there have to exist (not necessarily distinct) stable expansions Δ_1, Δ_2, Δ_3 of Σ such that no Δ_i is satisfied by the all-0 assignment and, for $1 \leq i \leq 3$, Δ_i is satisfied by the assignment setting to 1 only x_i. From the monotonicity of Σ it now follows that $x_1 \vee x_2 \vee x_3 \in \Delta_1 \cap \Delta_2 \cap \Delta_3$. Hence, $\Sigma \models^{\mathrm{skep}} x_1 \vee x_2 \vee x_3$—a contradiction.

If $[B' \cup \{0,1\}] = \mathsf{L}$, then B satisfies either $\mathsf{S}_{00} \subseteq [B \cup \{1\}]$ or $\mathsf{N}_2 \subseteq [B \cup \{1\}]$. In both cases, $\text{SKEP}_{\text{DL}}(B)$ is coNP-hard by Theorem 5.1.5, whereas $\text{SKEP}_{\text{AE}}(B') \in \text{P}$ by Theorem 5.2.4. Hence, unless $\text{P} = \text{NP}$, no translation from B-default logic to B'-autoepistemic logic is possible.

If $[B' \cup \{0,1\}] = \mathsf{V}$, then either $\mathsf{N}_2 \subseteq [B \cup \{1\}]$ or $\mathsf{L}_2 \subseteq [B \cup \{1\}]$ or $\mathsf{S}_{00} \subseteq [B \cup \{1\}]$. In any case, there exists a k-ary Boolean function $f \in [B' \cup \{0,1\}]$ such that $f \notin \mathsf{V}$. Consequently, the B-representation of $(\{f(x_1, \ldots, x_k)\}, \emptyset)$ cannot be translated to a set of autoepistemic B'-formulae by an argument analogous to the case $[B' \cup \{0,1\}] = \mathsf{M}$.

If $[B' \cup \{0,1\}] = \mathsf{N}$, then $\mathsf{V}_2 \subseteq [B \cup \{1\}]$ or $\mathsf{L}_2 \subseteq [B \cup \{1\}]$. In both cases, there exists a binary function $f \in [B \cup \{1\}] \setminus \mathsf{N}$, while any set of autoepistemic B'-formulae is equivalent to a set of literals and has at most one consistent stable expansion. Consequently, $(\{f(x_1, x_2)\}, \emptyset)$ cannnot be translated to a set of autoepistemic B'-formulae.

The same arguments apply for $[B' \cup \{0,1\}] \subseteq \mathsf{E}$. □

Observe that Lemmas 7.2.2 to 7.2.5 leave open only the case $[B \cup \{0,1\}] = \mathsf{V}$ and $[B' \cup \{0,1\}] = \mathsf{L}$. Thus, Theorem 7.2.1 is established.

Remark 7.2.6 *For the stricter notion of faithful translations (that is, translations that constitute a bijection between the stable extensions and the stable expansions that preserves membership on the set of objective formulae), Theorem 7.2.1 can be stated unconditional and without open cases:*

If $[B' \cup \{0,1\}] = \mathsf{L}$ and $\mathsf{N}_2 \subseteq [B \cup \{1\}]$, then we may without loss of generality assume that $\neg \in [B]$. Define a set D of default rules as

$$D := \left\{ \frac{1 : x_i}{x_i}, \frac{1 : \neg x_i}{\neg x_i} \;\middle|\; 1 \leq i \leq 3 \right\} \cup \left\{ \frac{x_1 : x_2}{x_3}, \frac{x_1 : x_2}{\neg x_3} \right\}.$$

Then corresponding $(B \cup \{1\})$-default theory $(\emptyset, D)$ has six stable extensions, each corresponding to a model of the formula $(\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_2 \vee x_3)$ (see the proof of Lemma 4.1.7). By Remark 4.2.7, the number of stable expansions of any set of autoepistemic B'-formulae is either 2^k or $2^k + 1$ for $k \in \mathbb{N}$. We thus conclude that no faithful translation is possible.

If $[B' \cup \{0,1\}] = \mathsf{L}$ and $\mathsf{V}_2 \subseteq [B \cup \{1\}]$, then the B-representation of $(\{x \vee y\}, \emptyset)$ cannot be translated into a set of autoepistemic B'-formulae: if there were a set $\Sigma \subseteq \mathcal{L}_{\text{ae}}(B')$ such that Σ had a unique stable expansion Δ satisfying $\Delta \cap \mathcal{L} = \text{Th}(x \vee y)$, then $x \vee y$ were expressible as a conjunction of affine formulae—contradictory to a result by Schaefer [Sch78, Lemma 3.1A].

We will now turn to translations in the converse direction. Although the full monotone fragment of autoepistemic logic does not admit a translation to default logic, translations exist if $B' \cup \{1\}$ is functional complete, if B comprises disjunctions only and $\neg \in [B']$, or if $\text{SKEP}_{\text{AE}}(B) \in \text{P}$ and all functions from B and both Boolean constants can be simulated using B'-default logic. Remarkably, the second condition does not require $\vee$ to be available in B'; disjunctions are instead simulated using the nonmonotonic features of default logic.

Theorem 7.2.7 *Let B and B' be finite sets of Boolean functions such that (a) $[B \cup \{0,1\}] = \mathsf{L}$ implies $[B' \cup \{1\}] \neq \mathsf{N}$, and (b) $[B \cup \{0,1\}] = \mathsf{V}$ implies $[B' \cup \{1\}] \neq \mathsf{M}$. Then there exists a translation from B-autoepistemic logic to B'-default logic*

1. *if $[B' \cup \{1\}] = \mathsf{BF}$, or*
2. *if $[B] \subseteq \mathsf{V}$ and $\neg \in [B']$, or*
3. *if $[B] \subseteq \mathsf{L}$ and $[B] \cup \{0\} \subseteq [B' \cup \{1\}]$, or*
4. *if $[B] \subseteq \mathsf{E}$ and $[B] \cup \{0\} \subseteq [B' \cup \{1\}]$;*

unless $\Sigma_2^{\mathrm{p}} = \Pi_2^{\mathrm{p}}$, no other translations are possible.

The proof of Theorem 7.2.7 will be established from Lemmas 7.2.8 to 7.2.12.

The first of these lemmas establishes the translation of arbitrary fragments of autoepistemic logic into full default logic. The existence of this translation independently follows from [Jan99, Propositions 4.9 and 4.10].

Lemma 7.2.8 *Let B and B' be finite sets of Boolean functions such that $[B' \cup \{1\}] = \mathsf{BF}$. Then there exists a translation from B-autoepistemic logic to B'-default logic.*

Proof. Let B and B' be finite sets of Boolean functions such that $[B' \cup \{1\}] = \mathsf{BF}$. Let $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ be the given set of autoepistemic formulae, let $\mathrm{SF}^L(\Sigma) = \{L\psi_1, \ldots, L\psi_n\}$, and let $t \notin \mathrm{Vars}(\Sigma)$. We map Σ to the B'-representation of the default theory (W, D), where $W := \Sigma_{[L\psi_1/p_{\psi_1}, \ldots, L\psi_n/p_{\psi_n}, 1/t]} \cup \{t\}$ and

$$D := \left\{ \frac{1 : p_{\psi_i}}{p_{\psi_i}}, \frac{1 : \neg p_{\psi_i}}{\neg p_{\psi_i}} \;\middle|\; 1 \leq i \leq n \right\} \cup \left\{ \frac{\psi_i : \neg p_{\psi_i}}{0}, \frac{p_{\psi_i} : \neg \psi_i}{0}, \;\middle|\; 1 \leq i \leq n \right\}.$$

Observe that the B'-representation of W and D can without loss of generality be assumed to be polynomial in the size of (W, D), as $B' \cup \{1\}$ efficiently implements $\{\wedge, \vee, \neg\}$, which implies that all formulae in Σ can be restructured to be of at most logarithmic depth [Spi71].

We proceed in two steps to show that f constitutes a translation from B-autoepistemic logic to B'-default logic. First we prove that any consistent stable expansion Δ with kernel $\Lambda \subseteq \mathrm{SF}^L(\Sigma) \cup \neg\mathrm{SF}^L(\Sigma)$ is a Σ-full set if and only if

$$G_\Lambda := \left\{ \frac{1 : p_{\psi_i}}{p_{\psi_i}} \;\middle|\; L\psi_i \in \Lambda \right\} \cup \left\{ \frac{1 : \neg p_{\psi_i}}{\neg p_{\psi_i}} \;\middle|\; \neg L\psi_i \in \Lambda \right\} \tag{7.1}$$

is the set of generating defaults for some consistent stable extension of (W, D). And second, we show that for all $\varphi \in \mathcal{L}$ over variables from Σ and all consistent stable expansions Δ, $\varphi \in \Delta \iff W \cup G_\Lambda \models \varphi$, where Λ is the kernel of Δ.

Observe that we may without loss of generality restrict our attention to consistent stable expansions: the (non-)existence of the inconsistent stable expansion does not alter the set of skeptical consequences of a given set $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$.

So suppose that Λ is a Σ-full set such that $\Sigma \cup \Lambda$ is consistent. Then, for each $L\psi_i \in \mathrm{SF}^L(\Sigma)$, $\Sigma \cup \Lambda \models \psi_i$ if and only if $L\psi_i \in \Lambda$, and $\Sigma \cup \Lambda \not\models \psi_i$ if and only if $\neg L\psi_i \in \Lambda$. With $p_{\psi_i} := 1$ if $\mathrm{L}\psi_i \in \Lambda$ and $p_{\psi_i} := 0$ otherwise, this is equivalent to $\Sigma_{[L\psi_1/p_{\psi_1},\ldots,L\psi_n/p_{\psi_n}]} \models \psi_i$ if and only if $p_{\psi_i} \in \Lambda$ and $\Sigma_{[L\psi_1/p_{\psi_1},\ldots,L\psi_n/p_{\psi_n}]} \not\models \psi_i$ if and only if $\neg p_{\psi_i} \in \Lambda$. For a set G of defaults, let $c(G) := \left\{\gamma \,\middle|\, \frac{\alpha:\beta}{\gamma} \in G\right\}$. Then, substituting p_{ψ_i} for $L\psi_i$ in Σ, we obtain

$$W \cup c(G_\Lambda) \models \psi_i \iff \frac{1 : p_{\psi_i}}{p_{\psi_i}} \in G_\Lambda \tag{7.2}$$

and

$$W \cup c(G_\Lambda) \not\models \psi_i \iff \frac{1 : \neg p_{\psi_i}}{\neg p_{\psi_i}} \in G_\Lambda, \tag{7.3}$$

for all $1 \le i \le n$. Accordingly, it follows that for all $1 \le i \le n$ neither $\frac{\psi_i:\neg p_{\psi_i}}{0}$ nor $\frac{p_{\psi_i}:\neg\psi_i}{0}$ are applicable: If $\frac{1:p_{\psi_i}}{p_{\psi_i}} \in G_\Lambda$, then both justifications $\neg p_{\psi_i}$ and $\neg\psi_i$ are inconsistent with $\mathrm{Th}\big(W \cup c(G_\Lambda)\big)$ by (7.2). If, on the other hand, $\frac{1:\neg p_{\psi_i}}{\neg p_{\psi_i}} \in G_\Lambda$, then their premises cannot be derived by (7.3). Therefore G_Λ is a maximal set of applicable default rules whose justifications are consistent with W. This allows us to conclude that G_Λ is a set of generating defaults for some stable expansion of (W, D).

Conversely, let G be the set of generating defaults for some consistent stable extension of (W, D). Then, by Proposition 3.1.4, W is consistent and for all $1 \le i \le n$ neither $\frac{\psi_i:\neg p_{\psi_i}}{0}$ nor $\frac{p_{\psi_i}:\neg\psi_i}{0}$ can be contained in G. As a result of this, if ψ_i can be derived, then $\neg p_{\psi_i}$ cannot. This in turn implies that $\frac{1:p_{\psi_i}}{p_{\psi_i}}$ is applicable and has to be contained in G. Furthermore, if p_{ψ_i} can be derived, then $\neg\psi_i$ has to be inconsistent with $\mathrm{Th}(W \cup c(G))$, because $\frac{p_{\psi_i}:\neg\psi_i}{0}$ is not applicable. Hence, $W \cup c(G) \models \psi_i$. Summarizing, $\frac{1:p_{\psi_i}}{p_{\psi_i}} \in G$ if and only if $W \cup c(G) \models \psi_i$. For the remaining default rules $\frac{1:\neg p_{\psi_i}}{\neg p_{\psi_i}}$ in G, this implies $W \cup c(G) \not\models \psi_i$. As either $\frac{1:p_{\psi_i}}{p_{\psi_i}}$ or $\frac{1:\neg p_{\psi_i}}{\neg p_{\psi_i}}$ is always applicable, we obtain that G has to contain exactly one of these default rules for each $1 \le i \le n$. Setting

$$\Lambda := \left\{L\psi_i \,\middle|\, \frac{1 : p_{\psi_i}}{p_{\psi_i}} \in G\right\} \cup \left\{\neg L\psi_i \,\middle|\, \frac{1 : \neg p_{\psi_i}}{\neg p_{\psi_i}} \in G\right\}$$

thus yields $\big(W \cup c(G)\big)_{[p_{\psi_1}/L\psi_1,\ldots,p_{\psi_n}/L\psi_n]} \equiv \Sigma \cup \Lambda$ as well as the equivalences $\Sigma \cup \Lambda \models \psi_i$ if and only if $L\psi_i \in \Lambda$, and $\Sigma \cup \Lambda \not\models \psi_i$ if and only if $\neg L\psi_i \in \Lambda$. Therefore, Λ is a Σ-full set and $G = G_\Lambda$ as defined in (7.1).

To conclude the proof, let $\varphi \in \mathcal{L}$ be over variables from Σ and Δ be a consistent stable expansion with kernel Λ. As φ is propositional, it holds that $\varphi \in \Delta \iff$

$\Sigma \cup \Lambda \models_L \varphi \iff \Sigma \cup \Lambda \models \varphi \iff W \cup G_\Lambda \models \varphi$. This concludes the proof, because $W \cup G_\Lambda \models \varphi$ if and only if φ is contained in the stable extension of (W, D) with generating defaults G_Λ. □

The following lemma presents the translation from the disjunctive fragment of autoepistemic logic to default logic restricted to essentially unary functions. This particularly emphasizes the expressive power inherent to default rules, since this constitutes the only translation eliminating Boolean connectives other than $\wedge$ by exploiting the nonmonotonic features of the target logic.

Lemma 7.2.9 *Let B and B' be finite sets of Boolean functions such that $[B] \subseteq \mathsf{V}$ and $\mathsf{N}_2 \subseteq [B']$. Then there exists a translation from B-autoepistemic logic to B'-default logic.*

Proof. Let B and B' be as in the statement of the lemma. Let $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ be the given set of autoepistemic formulae. The idea is to construct a stable extension for each model of a stable expansion of Σ.

Autoepistemic logic restricted to consistent stable expansions can be characterized as the nonmonotonic modal logic KD45 [Shv90, MT93]. Using the KD45-equivalences $LL\varphi \equiv L\varphi$ and $L(L\varphi \vee \psi) \equiv L\varphi \vee L\psi$, we can hence transform Σ to a set Σ' whose consistent stable expansions are identical to those of Σ and whose formulae are of the form $\varphi = L\beta_1 \vee \ldots \vee L\beta_k \vee \gamma$, where $k \geq 0$ and $\beta_1, \ldots, \beta_k, \gamma$ are disjunctions of propositions. The set Σ' is the image of a default theory (W, D) under Konolige's translation scheme [Kon88], where

$$W := \{\gamma \mid \varphi = \gamma \in \Sigma' \cap \mathcal{L}\} \text{ and}$$
$$D := \left\{ \frac{1 : \neg\beta_1 \wedge \cdots \wedge \neg\beta_k}{\gamma} \;\middle|\; \varphi = L\beta_1 \vee \ldots \vee L\beta_k \vee \gamma \in \Sigma', k > 0 \right\}.$$

As this translation is known to be faithful for for prerequisite-free default logic [MT89], it follows that the stable extensions of (W, D) coincide with the objective parts of the consistent stable expansions of Σ' and hence those of Σ. Therefore, $\Sigma \models^{\mathrm{skep}} \varphi$ iff $(W, D) \models \varphi$ for all propositional formulae φ over $\mathrm{Vars}(\Sigma)$.

We conclude the proof by eliminating all connectives except $\neg$ from (W, D). To this end, first map (W, D) to the equivalent default theory $(\emptyset, D')$ with

$$D' := D \cup \left\{ \frac{1 : 1}{\gamma} \;\middle|\; \gamma \in W \right\}.$$

Next, let $d = \frac{1:\beta}{\gamma}$ be some default from D'. By converting β to negation normal form, we can write β as a conjunction of negative literals and constants $\beta \equiv \bigwedge_{1 \leq i \leq n} \beta'_i$. Moreover, $\gamma = \gamma_1 \vee \cdots \vee \gamma_m$ is a disjunction of propositions or constants. Let $b_1, \ldots, b_n, c_1, \ldots, c_{m-1}$ be fresh and pairwise distinct propositions. Define

$$D_{just}(d) := \left\{ \frac{1 : \beta'_1}{b_1} \right\} \cup \left\{ \frac{b_{i-1} : \beta'_i}{b_i} \;\middle|\; 1 < i \leq n \right\}$$

and

$$D_{concl}(d) := \begin{cases} \left\{ \frac{b_n : \neg\gamma_1}{0} \right\} & \text{if } m = 1, \\ \left\{ \frac{b_n : \neg\gamma_1}{c_1} \right\} \cup \left\{ \frac{c_{i-1} : \neg\gamma_i}{c_i} \,\middle|\, 1 < i < m \right\} \cup \left\{ \frac{c_{m-1} : \neg\gamma_m}{0} \right\} & \text{if } m > 1. \end{cases}$$

Let $(\emptyset, D'')$ be obtained from $(\emptyset, D')$ by adding the default rules $D_{\mathrm{Vars}(\Sigma)} := \left\{ \frac{1:x}{x}, \frac{1:\neg x}{\neg x} \,\middle|\, x \in \mathrm{Vars}(\Sigma) \right\}$, replacing all $d \in D'$ with the default contained in $D_{just}(d) \cup D_{concl}(d)$ (using a fresh set of propositions for each default), and substituting 0 (respectively 1) with f (respectively t) while adding t and $\neg f$ to W. Then, by virtue of Lemma 2.4.2 (4.), $(\emptyset, D'')$ can clearly be rewritten as a B'-default theory.

As for the correctness of the transformation from (W, D) to $(\emptyset, D'')$, let E be a consistent stable extension of (W, D), let $d \in \mathrm{GD}(E)$ be a generating default of E, and let σ be a model of E. Define $G' := \left\{ \frac{1:x}{x} \,\middle|\, \sigma(x) = 1 \right\} \cup \left\{ \frac{1:\neg x}{\neg x} \,\middle|\, \sigma(x) = 0 \right\}$ to be the set of defaults from $D_{\mathrm{Vars}(\Sigma)}$ corresponding to σ. Using the iterative construction from Theorem 3.1.3 (1.), it is not hard to see that, for $E'_0 = \emptyset$ and $E'_1 = \mathrm{Th}(E'_0) \cup \left\{ \gamma \,\middle|\, \frac{\alpha:\beta}{\gamma} \in G' \right\}$, $W \subseteq \mathrm{Th}(E'_2)$ and all defaults in $D_{just}(d)$ may be successively applied to obtain a set E'_i such that $b_n \in E'_i$. As $\gamma = \gamma_1 \vee \cdots \vee \gamma_m \in E$, there exists an index i such that the default from $D_{concl}(d)$ with justification $\neg\gamma_i$ may not be applied. Similarly, for all defaults d not applicable in E, there exists a negated literal in the justification of d such that the corresponding default in $D_{just}(d)$ is not applicable. Consequently, 0 cannot be derived and the iterative construction will eventually converge to a set E' such that all justifications of default applied in its construction are consistent with E'. Hence, E' is a stable extension of $(\emptyset, D'')$ such that E' contains exactly those formulae satisfied by σ.

On the other hand, if (W, D) does not possess a consistent stable extension, then W is inconsistent by Proposition 3.1.4. To any assignment of $\sigma\colon W \to \{0, 1\}$ there hence exists a formula $\gamma \in W$ such that $\sigma \not\models \gamma$. Assume for simplicity that γ is equivalent to a disjunction $\bigvee_{1 \le i \le m} \gamma_i$ of propositions (the case $\varphi \equiv 0$ is analogous). Then, $\sigma \models \neg\gamma_i$ for all $1 \le i \le m$, which implies that 0 can be derived from the defaults in $D_{concl}(\frac{1:1}{\gamma})$. Therefore, $(\emptyset, D'')$ does not possess a stable extension.

From this we obtain that, for all $\varphi \in \mathcal{L}$ over $\mathrm{Vars}(\Sigma)$,

$$\begin{aligned} \Sigma \models^{\mathrm{skep}} \varphi \quad &\iff \quad (W, D) \models^{\mathrm{skep}} \varphi \\ &\iff \quad \sigma \in E \text{ for all stable extensions } E \text{ of } (W, D) \\ &\iff \quad \sigma \models \varphi \text{ for all models } \sigma \text{ of all stable extensions of } (W, D) \\ &\iff \quad \varphi \in E \text{ for all stable extensions } E \text{ of } (\emptyset, D'') \\ &\iff \quad (W', D') \models^{\mathrm{skep}} \varphi. \end{aligned}$$

□

Lemma 7.2.10 *Let B and B' be finite sets of Boolean functions such that $[B] \subseteq \mathsf{L}$ and $[B] \cup \{0\} \subseteq [B' \cup \{1\}]$, or such that $[B] \subseteq \mathsf{E}$ and $[B] \cup \{0\} \subseteq [B' \cup \{1\}]$. Then there exists a translation from B-autoepistemic logic to B'-default logic.*

Proof. Let B and B' be as in the statement of the lemma and denote by $\Sigma \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ the given set of autoepistemic B-formulae.

Suppose first that $[B \cup \{0,1\}] = \mathsf{L}$ and $[B] \cup \{0\} \subseteq [B' \cup \{1\}]$. For $B \subseteq \mathsf{L}$, the consistent Σ-full sets can be described as the solutions of the system $T'[x_{s+1}/Lx_{s+1}, \ldots, x_n/Lx_n]$ of linear equations from the proof of Lemma 4.2.6. Denote by T'' the system of linear equations obtained by applying Gaussian elimination to $T'[x_{s+1}/Lx_{s+1}, \ldots, x_n/Lx_n]$ and assume without loss of generality that the variables $Lx_{t+1}, \ldots, Lx_n$, $s \leq t$, are free in T''. Then the set of Σ-full sets is $\mathfrak{L} := \{\Lambda_I \mid I \subseteq \{t+1, \ldots, n\}\}$, where

$$\begin{aligned}
\Lambda_I := \, & \{Lx_i \mid t < i, i \in I\} \cup && (7.4)\\
& \{\neg Lx_i \mid t < i, i \notin I\} \cup && (7.5)\\
& \{Lx_i \mid s < i \leq t, f(Lx_{t+1}, \ldots, Lx_n) = 1\} \cup && (7.6)\\
& \{\neg Lx_i \mid s < i \leq t, f(Lx_{t+1}, \ldots, Lx_n) = 0\} \cup && (7.7)\\
& \{\neg Lx_i \mid i \leq s\}. && (7.8)
\end{aligned}$$

Observe that (7.6)–(7.8) do not depend on I. Hence, there is a Σ-full set for any set of beliefs $Lx_{t+1}, \ldots, Lx_n$. We define the translation as $f(\Sigma) := (W', D')$, where W' and D' are the $(B' \cup \{1\})$-representation of

$$W := (\Sigma' \cup \Pi)_{[Lx_1/0, \ldots, Lx_s/0, Lx_{s+1}/p_{x_{s+1}}, \ldots, Lx_n/p_{x_n}]}$$

and

$$D := \left\{ \frac{1 : p_{x_i}}{p_{x_i}}, \frac{1 : \neg p_{x_i}}{\neg p_{x_i}} \;\middle|\; t < i \leq n \right\},$$

where $\Sigma' \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ is the set of autoepistemic formulae constructed in the proof of Lemma 4.2.8 and Π being derived from the equations in T''. Notice that all connectives in $[B]$ are associative, thus W' and D' may without loss of generality be assumed to be of size polynomial in $|W|$ and $|D|$ by a simple restructuring argument (compare the proof of Lemma 7.2.4).

It is easy to see that the stable expansions of Σ are in one-to-one correspondence with the stable extensions of (W, D). So it remains to show that the set of skeptically entailed propositional formulae over variables from Σ remains unaltered. To this end, let φ be a propositional formulae over variables from Σ, let Δ be a consistent stable expansion, and let Λ be its kernel. Then $\varphi \in \Delta \iff \Sigma \cup \Lambda \models_L \varphi \iff \Sigma' \cup \Lambda' \models \varphi \iff W \cup G \models \varphi \iff E \models \varphi$, where Λ' is the Σ'-full set corresponding to Λ (as constructed in the proof of Lemma 4.2.8), $G := \left\{ \frac{1:p_{x_i}}{p_{x_i}} \;\middle|\; Lx_i \in \Lambda' \right\} \cup \left\{ \frac{1:\neg p_{x_i}}{\neg p_{x_i}} \;\middle|\; \neg Lx_i \in \Lambda' \right\}$, and E the stable extension corresponding to the set G of generating defaults. It is obvious that this construction also works for the case that $[B \cup \{0,1\}] = \mathsf{N}$ and

$[B] \cup \{0\} \subseteq [B' \cup \{1\}]$, as in this case there exists at most one consistent stable expansion, whence Π can be written using negations only. This concludes the proof of the first part.

Hence, suppose that $[B] \subseteq \mathsf{E}$ and $[B] \cup \{0\} \subseteq [B' \cup \{1\}]$. Again, Σ is known to have at most one consistent stable expansion whose kernel can be computed in polynomial time using the proof of Lemma 4.2.10. Suppose without loss of generality that $\mathrm{SF}^L(\Sigma) = \{L\varphi_1, \dots, L\varphi_n\}$ and define, for a Σ-full set Λ, $c_i(\Lambda) := 0$ if $\neg L\varphi_i \in \Lambda$ and $c_i(\Lambda) := 1$ if $L\varphi_i \in \Lambda$. Now set $W := \Sigma'_{[L\varphi_1/c_1(\Lambda),\dots,L\varphi_n/c_n(\Lambda)]}$ if Σ has a consistent stable expansion with kernel Λ, and $W := \{0\}$ otherwise. Then the function f mapping Σ to the B'-representation of W and an empty set of default rules yields the desired translation. $\square$

Lemma 7.2.11 *Let B and B' be finite sets of Boolean functions such that $0 \notin [B' \cup \{1\}]$. Then there exists no translation from B-autoepistemic logic to B'-default logic.*

Proof. Let B and B' be finite sets of Boolean functions such that $0 \notin [B' \cup \{1\}]$. Then $[B' \cup \{1\}] \subseteq \mathsf{R}_1$ and any B'-default theory is guaranteed to have a consistent stable extension by Lemma 4.1.3. Therefore $\Sigma := \{Lf\}$ cannot be translated into an equivalent B'-default theory. $\square$

Lemma 7.2.12 *Let B and B' be finite sets of Boolean functions such that*

1. $[B' \cup \{1\}] \in \{\mathsf{M}, \mathsf{E}, \mathsf{V}, \mathsf{L}, \mathsf{N}, \mathsf{I}\}$,
2. $[B \cup \{0,1\}] = \mathsf{V}$ *implies* $[B' \cup \{1\}] \subsetneq \mathsf{M}$,
3. $[B \cup \{0,1\}] \subseteq \mathsf{L}$ *implies* $[B' \cup \{1\}] \subseteq \mathsf{M}$,
4. $[B] \nsubseteq \mathsf{E}$.

Then there exists no translation from B-autoepistemic logic to B'-default logic unless $\Sigma_2^{\mathrm{p}} = \Pi_2^{\mathrm{p}}$.

Proof. Let B and B' be finite sets of Boolean functions as in the statement of the lemma. We will to distinguish between the possible clones of $[B' \cup \{1\}]$.

For $[B' \cup \{1\}] = \mathsf{M}$, we distinguish two possible cases. First, if $[B] \nsubseteq \mathsf{M}$, then this induces the existence of a k-ary $f \in [B \cup \{0,1\}]$ such that $f \notin [B' \cup \{1\}]$. Hence, $\{f(x_1, \dots, x_k), t\}_{[1/t, 0/Lf]} \subseteq \mathcal{L}_{\mathrm{ae}}(B)$ cannot be translated to a B'-default theory: any B'-default theory has at most one stable extension that is equivalent to a monotone formula. Second, if $[B] \subseteq \mathsf{M}$ and $[B] \subsetneq \mathsf{V}$, then $\textsc{Exp}(B)$ is Σ_2^{p}-complete (by conditions (2.) and (4.) of the lemma), while $\textsc{Ext}(B')$ is contained in Δ_2^{p}. Consequently, there cannot be a translation from B-autoepistemic logic to B'-default logic unless $\Sigma_2^{\mathrm{p}} = \Pi_2^{\mathrm{p}}$.

Similarly, if $[B' \cup \{1\}] \subseteq \mathsf{V}$ or $[B' \cup \{1\}] \subseteq \mathsf{E}$, then either $\mathsf{V} \subseteq [B \cup \{0,1\}]$ or $\mathsf{N} \subseteq [B \cup \{0,1\}]$. In the latter case, $\Sigma := \{\neg x\}$ cannot be translated to B'-autoepistemic logic. In the former, $\textsc{Exp}(B)$ is at least NP-hard, while $\textsc{Ext}(B) \in \mathrm{P}$. Consequently, no translation from B-autoepistemic logic to B'-default logic exists unless $\mathrm{P} = \mathrm{NP}$.

Next, if $[B' \cup \{1\}] = \mathsf{L}$, then $\mathsf{M} \subseteq [B \cup \{0,1\}]$. As a result $\textsc{Exp}(B)$ is Σ_2^{p}-complete, whereas $\textsc{Ext}(B')$ is contained in NP. Thus a translation from B-autoepistemic logic to B'-default logic exists only if $\mathrm{NP} = \Sigma_2^{\mathrm{p}}$. This is equivalent to the condition NP = coNP.

Lastly, if $[B' \cup \{1\}] = \mathsf{N}$, then again $\mathsf{M} \subseteq [B \cup \{0,1\}]$. Thus, no translations is possible unless NP = coNP. □

Concluding, Lemmas 7.2.8 to 7.2.10 cover all cases in which a transition was claimed to exist, while Lemmas 7.2.11 and 7.2.12 refute the existence of translations in all remaining cases except (a) $[B \cup \{0,1\}] = \mathsf{L}$ and $[B' \cup \{1\}] = \mathsf{N}$, and (b) $[B \cup \{0,1\}] = \mathsf{V}$ and $[B' \cup \{1\}] = \mathsf{M}$. Hence, Theorem 7.2.7 is established.

Remark 7.2.13 *Theorem 7.2.7 differs significantly from the analogue statement for faithful translations. For finite sets B and B' of Boolean functions, there exists a faithful translation from B-autoepistemic logic to B'-default logic if and only if $[B] \subseteq \mathsf{E}$ or $[B] \subseteq \mathsf{N} \subseteq [B' \cup \{1\}]$. This is due to $\Sigma_1 := \{t, p \oplus Lp \oplus t, p \oplus Lq \oplus t\}$ and $\Sigma_2 := \{Lp \vee q, Lq \vee p\}$.*

- *Σ_1 admits two stable expansions: a consistent one containing $\{\neg Lp, \neg Lq\}$ and the inconsistent stable expansion. Thence, by Proposition 3.1.4, there is no faithful translation for all B such that $\mathsf{L}_2 \subseteq [B]$.*
- *The set Σ_2 admits two consistent stable expansions. Lemma 4.1.3 hence yields the inexistence of a faithful translation for all B' such that $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{M}$ and $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{R}_1$.*

7.3 Default Logic and Circumscription

In the last section, we have seen that default logic and autoepistemic logic admit translations to each other in presence of functional complete sets of Boolean functions, from the monotone fragment of default logic to the monotone fragment of autoepistemic logic, from the disjunctive fragment autoepistemic logic to default logic restricted to essentially unary functions, and if the set of stable extensions or stable expansions can be efficiently computed.

For translations from circumscription to default logic the situation is different: with the exception of the open cases, translations from B-circumscription to B'-default logic exist only if B'-default logic is able to express all functions from $B \cup \{\neg\}$. That is, negation is enough to subsume the corresponding fragment of circumscription in default logic. We point out that these translations are modular.

Theorem 7.3.1 *Let B and B' be finite sets of Boolean functions such that (a) $[B] \subseteq \mathsf{M}$ implies $[B' \cup \{1\}] \notin \{\mathsf{N}, \mathsf{L}\}$, and (b) $[B \cup \{0,1\}] = \mathsf{L}$ implies $[B' \cup \{1\}] \neq \mathsf{N}$. Then there exists a translation from B-circumscription to B'-default logic*

1. *if $\neg \in [B' \cup \{1\}]$ and $[B] \subseteq [B' \cup \{1\}]$, or*
2. *if $\neg \in [B' \cup \{1\}]$ and $[B] \subseteq \mathsf{E}$;*

unless NP $=$ coNP*, no other translations are possible.*

The proof of Theorem 7.3.1 is established from the following three lemmas. The first asserts the existence of a translation in the above mentioned cases. The second and third lemma show that in all other cases, no translation may exist.

Lemma 7.3.2 *Let B and B' be finite sets of Boolean functions such that $\neg \in [B' \cup \{1\}]$ and $[B] \subseteq [B' \cup \{1\}]$, or such that $\neg \in [B' \cup \{1\}]$ and $[B] \subseteq \mathsf{E}$. Then there exists a translation from B-circumscription to B'-default logic.*

Proof. Let B and B' be finite sets of Boolean functions such that $\neg \in [B' \cup \{1\}]$ and $[B] \subseteq [B' \cup \{1\}]$ or such that $\neg \in [B' \cup \{1\}]$ and $[B] \subseteq \mathsf{E}$. Following ideas presented in [Eth87] (see also [Nie93]), we translate the given pair $\big(\Gamma, (P, Q, Z)\big)$ with $\Gamma \subseteq \mathcal{L}(B)$ and (P, Q, Z) being a partition of $\mathrm{Vars}(\Gamma)$, to (W, D), where W is the B'-representation of $\Gamma_{[1/t]} \cup \{t\}$ and

$$D := \left\{ \frac{: \neg p}{\neg p} \,\middle|\, p \in P \right\} \cup \left\{ \frac{: \neg q}{\neg q}, \frac{: q}{q} \,\middle|\, q \in Q \right\}.$$

Any (P, Q, Z)-minimal model σ of Γ clearly corresponds to a stable extension, with generating defaults

$$G_\sigma := \left\{ \frac{: \neg p}{\neg p} \,\middle|\, p \in P \setminus \sigma \right\} \cup \left\{ \frac{: \neg q}{\neg q} \,\middle|\, q \in Q \setminus \sigma \right\} \cup \left\{ \frac{: q}{q} \,\middle|\, q \in Q \cap \sigma \right\}.$$

Vice versa, any stable extension of (W, D) corresponds to a (P, Q, Z)-minimal model of Γ. This yields

$$\begin{aligned} \Gamma \models^{\mathrm{circ}}_{(P,Q,Z)} \varphi \quad &\iff \quad \varphi \text{ is satisfied by all } (P, Q, Z)\text{-minimal models } \sigma \text{ of } \Gamma \\ &\iff \quad W \cup G_\sigma \models \varphi \text{ for all } (P, Q, Z)\text{-minimal models } \sigma \text{ of } \Gamma \\ &\iff \quad (W, D) \models^{\mathrm{skep}} \varphi. \end{aligned}$$

As for the computation of the translation, note that $[B' \cup \{1\}]$ is among the clones N, L, BF. In the first two cases, the B'-representation of $\Gamma_{[1/t]} \cup \{t\}$ can be computed in polynomial time as in the proof of Lemma 2.5.5. Lastly, if $[B' \cup \{1\}] = \mathsf{BF}$, then $B' \cup \{1\}$ efficiently implements the Boolean standard base. Therefore, $\Gamma_{[1/t]} \cup \{t\}$ can be rewritten as a set of formulae of logarithmic depth in polynomial time [Spi71]. Consequently, space and time needed to compute its B'-representation can be bounded by a polynomial. □

Lemma 7.3.3 *Let B and B' be finite sets of Boolean functions such that $\neg \notin [B' \cup \{1\}]$. Then there exists no translation from B-circumscription to B'-default logic.*

Proof. Let B and B' be finite sets of Boolean functions such that $\neg \notin [B' \cup \{1\}]$. Then $\emptyset \models^{\mathrm{circ}}_{(\{x\},\emptyset,\emptyset)} \neg x$. As $\neg \notin [B' \cup \{1\}]$, either $[B' \cup \{1\}] \subseteq \mathsf{R}_1$ or $[B' \cup \{1\}] \subseteq \mathsf{M}$ or both. Therefore, any consistent stable extension of a B'-default theory is 1-reproducing and no translation from B-circumscription to B'-default logic can exist. □

Lemma 7.3.4 *Let B and B' be finite sets of Boolean functions such that*

1. $[B] \nsubseteq [B' \cup \{1\}]$,
2. $[B \cup \{0,1\}] = \mathsf{BF}$ *or* $[B' \cup \{1\}] \notin \{\mathsf{N}, \mathsf{L}\}$, *and*
3. $[B] \nsubseteq \mathsf{E}$.

Then there exists no translation from B-circumscription to B'-default logic unless NP $=$ coNP.

Proof. Let B and B' be finite sets of Boolean functions such that $[B] \nsubseteq [B' \cup \{1\}]$ and $[B] \nsubseteq \mathsf{E}$. We distinguish among the possible clones of $[B' \cup \{1\}]$.

If $[B' \cup \{1\}] \subseteq \mathsf{R}_1$ or $[B' \cup \{1\}] \subseteq \mathsf{M}$, then any consistent stable extension of a B'-default theory is 1-reproducing. As regards B, we know that $x \vee y \in [B \cup \{1\}]$ or $x \oplus y \oplus z \in [B \cup \{1\}]$. Define $\Gamma := \{x \vee y \vee z\}$, $\Gamma' := \{x \oplus y \oplus z\}$, and $P := \{x, y, z\}$, $Q := \emptyset$, $Z := \emptyset$. Clearly, $\Gamma \models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$ (respectively $\Gamma' \models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$) if and only if $f_{1\mathrm{in}3}(x,y,z) \models \varphi$, where $f_{1\mathrm{in}3}$ denotes the Boolean function that evaluates to true if exactly one of its arguments is set to true: $f_{1\mathrm{in}3}(x,y,z) := (x \wedge \neg y \wedge \neg z) \vee (\neg x \wedge y \wedge \neg z) \vee (\neg x \wedge \neg y \wedge z)$. Clearly, $f_{1\mathrm{in}3} \notin \mathsf{R}_1 \cup \mathsf{M}$. Therefore, no translation from B-circumscription to B'-default logic may exist.

If $[B' \cup \{1\}] \subseteq \mathsf{L}$, then $\mathsf{S}_{02} \subseteq [B]$ or $\mathsf{S}_{12} \subseteq [B]$ or $\mathsf{D}_1 \subseteq [B]$ by the second condition in the statement of the lemma. For all such sets B, $\textsc{CircInf}(B)$ is known to be Π^{p}_2-complete, while $\textsc{Skep}_{\mathrm{DL}}(B') \in$ coNP. Hence, no translation may exist unless NP $=$ coNP. □

Observe that the second condition in Lemma 7.3.4 is equivalent to $[B] \nsubseteq \mathsf{M}$ or $[B] \nsubseteq \mathsf{L}$ or $[B' \cup \{1\}] \notin \{\mathsf{N}, \mathsf{L}\}$. Lemmas 7.3.3 and 7.3.4 hence refute the existence of translations in all cases except those covered by Lemma 7.3.2 and those excluded in Theorem 7.3.1. This concludes the proof of Theorem 7.3.1.

As for the converse direction, we show that full default logic cannot be embedded into circumscription unless PH collapses to its first level. Hence, even under the weak notion of translations considered herein, circumscription is in general less expressive than default logic. This statement carries over to almost all fragments of default logic such that $\textsc{Skep}_{\mathrm{DL}}(B)$ is intractable: conditioned on NP $\neq$ coNP and with the exception of the open cases, B-default logic can be translated to B'-circumscription if and only if $\textsc{Skep}_{\mathrm{DL}}(B) \in$ P and $[B] \subseteq [B' \cup \{1\}]$. Key to these translations is that for these fragments both the implication problem is tractable and that stable extensions are guaranteed to be unique.

Theorem 7.3.5 *Let B and B' be finite sets of Boolean functions such that (a) $[B \cup \{1\}] = \mathsf{V}$ implies $[B' \cup \{0,1\}] \neq \mathsf{L}$ or $0 \in [B \cup \{1\}] \setminus [B' \cup \{1\}]$, and (b) $[B \cup \{1\}] \in \{\mathsf{N}, \mathsf{L}\}$ or $\mathsf{S}_{00} \subseteq [B \cup \{1\}] \subseteq \mathsf{R}_1$ implies $\mathsf{S}_1 \nsubseteq [B']$. Then there exists a translation from B-default logic to B'-circumscription*

1. *if* $[B] \subseteq \mathsf{V}$ *and* $[B] \subseteq [B' \cup \{1\}]$, *or*
2. *if* $[B] \subseteq \mathsf{L}_1$ *and* $[B] \subseteq [B' \cup \{1\}]$, *or*

3. *if* $[B] \subseteq \mathsf{E}$;

unless NP = coNP, *no other translations are possible.*

Similar to the previous theorems, the proof of Theorem 7.3.5 will be established from the lemmas in the rest of this section.

Lemma 7.3.6 *Let B and B' be finite sets of Boolean functions such that $[B] \subseteq \mathsf{V}$ and $[B] \subseteq [B' \cup \{1\}]$, or such that $[B] \subseteq \mathsf{L}_1$ and $[B] \subseteq [B' \cup \{1\}]$, or such that $[B] \subseteq \mathsf{E}$. Then there exists a translation from B-default logic to B'-circumscription.*

Proof. Let B and B' be finite sets of Boolean functions as in the statement of the lemma. Let (W, D) denote the given B-default theory. By Lemma 4.1.3, (W, D) has at most one stable extension E whose generating defaults can be computed efficiently. We can hence define the translation as the mapping of (W, D) to the B'-representation of $\big(\Gamma, (P, Q, Z)\big)$, where $\Gamma := W \cup \big\{\gamma \mid \frac{\alpha:\beta}{\gamma} \in \mathrm{GD}(E)\big\}$, $P := \emptyset$, $Q := \emptyset$, $Z := \mathrm{Vars}(\Gamma)$. It clearly holds that $(W, D) \models^{\mathrm{skep}} \varphi$ if and only if $\Gamma \models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$ for all $\varphi \in \mathcal{L}$ over propositions from (W, D). We conclude by observing that, by the associativity of $\vee$ and $\leftrightarrow$, and the fact that $[B] \subseteq \mathsf{E}$ or $[B] \subseteq [B' \cup \{1\}]$, the B'-representation of Γ is guaranteed to be polynomial-time computable. ☐

Lemma 7.3.7 *Let B and B' be finite sets of Boolean functions such that* $\textsc{Skep}_{\mathrm{DL}}(B)$ *is* coNP*-hard and* $\textsc{Sat}(B')$ *is in* P*. Then there exists no translation from B-default logic to circumscription unless* P = NP*.*

Proof. Assume that P $\neq$ NP and suppose for a contradiction that there exists a translation from B-default logic to B'-circumscription such that $\textsc{Skep}_{\mathrm{DL}}(B)$ is coNP-hard and $\textsc{Sat}(B')$ is in P. Let (W, D) be some B-default theory, let $\varphi \in \mathcal{L}(B)$, and define $D' := D \cup \{\frac{\varphi:x}{y}\}$ for fresh propositions x and y. Denote by $\big(\Gamma, (P, Q, Z)\big)$ the translation of (W, D'). It now holds that $(W, D) \models^{\mathrm{skep}} \varphi$ if and only if $(W, D') \models^{\mathrm{skep}} y$ if and only if $\Gamma \models^{\mathrm{circ}}_{(P,Q,Z)} y$ if and only if there exists no minimal model σ of Γ such that $\sigma(y) = 0$. However, deciding $(W, D) \models^{\mathrm{skep}} \varphi$ is coNP-hard while deciding the existence of a minimal model of Γ with $\sigma(y) = 0$ is contained in P; a contradiction to the existence of a translation. ☐

Lemma 7.3.8 *Let B and B' be finite sets of Boolean functions such that* $\textsc{Skep}_{\mathrm{DL}}(B)$ *is* Δ_2^{P}*-hard and* $\textsc{Sat}(B')$ *is in* NP*. Then there exists no translation from B-default logic to circumscription unless* NP = coNP*.*

Proof. Analogous to Lemma 7.3.7. ☐

Lemma 7.3.9 *Let B and B' be finite sets of Boolean functions such that*

1. $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$ *or* $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}_1$,
2. $[B] \nsubseteq [B' \cup \{1\}]$, *and*

3. $[B \cup \{0,1\}] = \mathsf{V}$ *implies* $[B' \cup \{0,1\}] \neq \mathsf{L}$ *or* $0 \in [B \cup \{1\}] \setminus [B' \cup \{1\}]$.

Then there exists no translation from B-default logic to B'-circumscription.

Proof. Let B and B' be finite sets of Boolean functions as in the statement of the lemma. First consider the case that $0 \in [B \cup \{1\}] \setminus [B' \cup \{1\}]$. Then $(\{0\}, \emptyset)$ cannot be translated into B'-circumscription, because any set of B'-formulae is trivially satisfiable. We may thus assume that $0 \in [B \cup \{1\}]$ implies $0 \in [B' \cup \{1\}]$.

If $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$ then, by the given conditions on B and B' and the fact that $0 \in [B \cup \{1\}]$ implies $0 \in [B' \cup \{1\}]$, $[B' \cup \{1\}] \subseteq \mathsf{N}$ or $[B' \cup \{1\}] \subseteq \mathsf{E}$. In both cases the (P,Q,Z)-minimal models of any given circumscriptive theory Γ are uniquely determined on $\mathrm{Vars}(\Gamma) \cup P$. From this, it is easy to see that $x \vee y$ cannot be translated to B'-circumscription.

On the other hand, if $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}_1$ then $[B'] \subseteq \mathsf{M}$ or $[B'] \subseteq \mathsf{N}$. In the latter case, the same arguments as for the case $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$ apply. In the former, we have $x \leftrightarrow y \notin [B' \cup \{1\}]$. Thus a translation $\big(\Gamma, (P,Q,Z)\big)$ of $(W,D) := (\{x \leftrightarrow y\}, \emptyset)$ has to circumscribe at least one proposition (that is, $P \neq \emptyset$). Moreover, the restriction of the (P,Q,Z)-minimal models of Γ to $\{x,y\}$ has to coincide with the set of models of $x \leftrightarrow y$. Let σ and σ' denote two models of Γ such that $\sigma \cap \{x,y\} = \emptyset$ and $\sigma' \cap \{x,y\} = \{x,y\}$. For σ and σ' to be (P,Q,Z)-minimal, there have to exist propositions $p,q \in P$ satisfying $\sigma(p) \neq \sigma'(p)$, $\sigma(q) \neq \sigma'(q)$ and $\sigma(p) \neq \sigma(q)$. But this contradicts the monotonicity of Γ, because $\sigma \models \Gamma$ and $\sigma \cup \{x\} \not\models \Gamma$. $\square$

This completes the proof of Theorem 7.3.5. Indeed, the existence of the claimed translations follows from Lemma 7.3.6. On the other hand, Lemma 7.3.7 shows that for $\mathsf{N} \subseteq [B \cup \{1\}]$ or $\mathsf{S}_{00} \subseteq [B \cup \{1\}] \subseteq \mathsf{R}_1$ translations from B-default logic to B'-circumscription may exist only if $\mathsf{S}_1 \subseteq [B']$ or $\mathrm{P} = \mathrm{NP}$; Lemma 7.3.8 shows that for $[B \cup \{1\}] \in \{\mathsf{M}, \mathsf{BF}\}$ translations from B-default logic to B'-circumscription may exist only if $\mathrm{NP} = \mathrm{coNP}$; and Lemma 7.3.9 shows that for $[B] \nsubseteq [B' \cup \{1\}]$ and either $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$ or $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}_1$ translations from B-default logic to B'-circumscription may exist only if $\mathsf{V}_2 \subseteq [B] \subseteq \mathsf{V}$, $\mathsf{L}_2 \subseteq [B'] \subseteq \mathsf{L}$, and $\{0\} \cap [B \cup \{1\}] \subseteq \{0\} \cap [B' \cup \{1\}]$. Concluding, Lemmas 7.3.7 to 7.3.9 cover all remaining cases except those excluded in the statement of the theorem.

7.4 Autoepistemic Logic and Circumscription

In the last section of this chapter, we study the possibility of translating circumscription into autoepistemic logic and vice versa. Commencing with the former direction, we prove that translations from B-autoepistemic logic to B'-circumscription are possible if $B' \cup \{0,1\}$ is functional complete or if $\mathrm{SKEP}_{\mathrm{AE}}(B)$ is polynomial-time decidable and $\neg \in [B']$. Accordingly, the fragments of autoepistemic logic that allow for translations to circumscription are a strict subset of the fragments of default logic that do so. In all remaining cases except $[B \cup \{0,1\}] = \mathsf{L}$ and $[B' \cup \{0,1\}] = \mathsf{L}$, no translation is possible unless $\mathrm{P} = \mathrm{NP}$.

Theorem 7.4.1 *Let B and B' be finite sets of Boolean functions such that $[B \cup \{0,1\}] \neq \mathsf{L}$ or $[B' \cup \{0,1\}] \neq \mathsf{L}$. Then there exists a translation from B-circumscription to B'-autoepistemic logic*

1. *if $[B' \cup \{0,1\}] = \mathsf{BF}$, or*
2. *if $\neg \in [B']$ and $[B] \subseteq \mathsf{E}$, or*
3. *if $\neg \in [B']$ and $[B] \subseteq \mathsf{N}$;*

unless $\mathrm{P} = \mathrm{NP}$, no other translations are possible.

We prove Theorem 7.4.1 from the four lemmas below.

Lemma 7.4.2 *Let B and B' be finite sets of Boolean functions such that $\neg \in [B']$ and $[B] \subseteq \mathsf{E}$, or such that $[B] \subseteq \mathsf{N}$. Then there exists a translation from B-circumscription to B'-autoepistemic logic.*

Proof. The key idea is that for $[B] \subseteq \mathsf{E}$ or $[B] \subseteq \mathsf{N}$, the (P, Q, Z)-minimal models of a given $\Gamma \subseteq \mathcal{L}(B)$ are determined on $\mathrm{Vars}(\Gamma) \cup P$: $\sigma\colon \mathrm{Vars}(\Gamma) \to \{0,1\}$ is (P, Q, Z)-minimal if and only if $\sigma(x) = 1$ for all $x \in \mathrm{Vars}(\Gamma)$ such that $\Gamma \models x$, and $\sigma(x) = 0$ for all $x \in P \setminus \mathrm{Vars}(\Gamma)$ and all $x \in \mathrm{Vars}(\Gamma)$ such that $\Gamma \models \neg x$. We can therefore map the tuple $\big(\Gamma, (P, Q, Z)\big)$ to the B'-representation of the autoepistemic theory Σ defined as $\Gamma \cup \{\neg p \mid p \in P \setminus \mathrm{Vars}(\Gamma)\}$. The correctness of this translation follows from the fact that for all assignments σ, $\sigma \models \Sigma$ if and only if σ is a (P, Q, Z)-minimal model of Γ. ☐

Lemma 7.4.3 *Let B and B' be finite sets of Boolean functions such that $\neg \notin [B']$. Then there exists no translation from B-circumscription to B'-autoepistemic logic.*

Proof. Let B and B' be finite sets of Boolean functions such that $\neg \notin [B']$. Then $[B' \cup \{0,1\}] \subseteq \mathsf{M}$. Consequently, $\big(\emptyset, (\{x\}, \emptyset, \emptyset)\big)$ cannot be translated into a set of autoepistemic formulae analogously to the proof of Lemma 7.3.3. ☐

Lemma 7.4.4 *Let B and B' be finite sets of Boolean functions such that $[B] \not\subseteq \mathsf{E}$, $[B] \not\subseteq \mathsf{L}$, and $[B' \cup \{0,1\}] = \mathsf{L}$. Then there exists no translation from B-circumscription to B'-autoepistemic logic unless $\mathrm{P} = \mathrm{NP}$.*

Proof. Let B and B' be finite sets of Boolean functions such that $[B] \not\subseteq \mathsf{E}$ and $[B] \not\subseteq \mathsf{L}$ and $[B' \cup \{0,1\}] = \mathsf{L}$. The conditions on B are equivalent to $\mathsf{V}_2 \subseteq [B \cup \{1\}]$. In this case, $\textsc{CircInf}(B)$ is coNP-hard, whereas $\textsc{Skep}_{\mathrm{AE}}(B') \in \mathrm{P}$. Hence, if there exists a translation that preserved the set of skeptical consequences of the given circumscriptive theory, then $\mathrm{P} = \mathrm{NP}$. ☐

Lemma 7.4.5 *Let B and B' be finite sets of Boolean functions such that $[B] \not\subseteq \mathsf{E}$, $[B] \not\subseteq \mathsf{N}$, and $[B' \cup \{0,1\}] = \mathsf{N}$. Then there exists no translation from B-circumscription to B'-autoepistemic logic.*

Proof. Let B and B' be as in the statement of the lemma. We distinguish two cases. First suppose that $[B] \subseteq \mathsf{L}$. Consider $\Gamma := \{x \oplus y \oplus z\}$ and $P := \{x, y, z\}$, $Q := \emptyset$, $Z := \emptyset$. Analogously to the proof of Lemma 7.3.4, it holds that $\Gamma \models^{\text{circ}}_{(P,Q,Z)} \varphi$ if and only if $f_{1\text{in}3}(x, y, z) \models \varphi$ for all $\varphi \in \mathcal{L}$. On the other hand, any set $\Sigma \subseteq \mathcal{L}_{\text{ae}}(B')$ has at most one consistent stable expansion (see Lemma 4.2.9). Hence, if there existed a translation f from B-circumscription to B'-autoepistemic logic, then either $f\big((\Gamma, (P, Q, Z))\big) \models^{\text{skep}} \varphi$ for at least one of $\varphi \in \{x, y, z\}$ or $f\big((\Gamma, (P, Q, Z))\big) \not\models^{\text{skep}} f_{1\text{in}3}(x, y, z)$, both of which yield a contradiction.

Hence suppose that $[B] \nsubseteq \mathsf{L}$. The proof of this case is completely analogous to the proof of Lemma 7.4.4 except for the fact that $\textsc{Skep}_{\text{AE}}(B')$ is contained in $\mathrm{AC}^0[2] \subsetneq \mathrm{coNP}$. From this we obtain an unconditional result. □

To conclude the proof of Theorem 7.4.1, note that Theorems 7.2.1 and 7.3.1 together assert the existence of a translation from B-circumscription to B'-autoepistemic logic for all B' such that $[B' \cup \{0, 1\}] = \mathsf{BF}$ (see also [Jan99] for an alternative proof). Moreover, $\neg \in [B']$ if and only if $[B' \cup \{0, 1\}]$ is among the clones N, L, BF. Hence, Lemmas 7.4.2 to 7.4.5 cover all cases except $[B \cup \{0, 1\}] = [B' \cup \{0, 1\}] = \mathsf{L}$. This establishes the theorem.

Lastly, it remains to discuss translations from autoepistemic logic to circumscription. Similar to the case of translations from default logic, full autoepistemic logic cannot be translated to circumscription unless $\mathrm{P} = \mathrm{NP}$. This continues to hold even for monotone autoepistemic logic. Indeed, with the exception of the case $[B \cup \{0, 1\}] = \mathsf{V}$ and $\mathsf{S}_1 \subseteq [B']$, B-autoepistemic logic can only be translated into B'-circumscription if $\textsc{Skep}_{\text{AE}}(B) \in \mathrm{P}$ and the necessary Boolean functions can be simulated using circumscription.

Theorem 7.4.6 *Let B and B' be finite sets of Boolean functions such that $[B \cup \{0, 1\}] = \mathsf{V}$ implies $\mathsf{S}_1 \nsubseteq [B']$. Then there is a translation from B-autoepistemic logic to B'-circumscription*

1. *if $[B] \subseteq \mathsf{L}$ and $[B] \cup \{0\} \subseteq [B' \cup \{1\}]$, or*
2. *if $[B] \subseteq \mathsf{N}$ and $0 \in [B' \cup \{1\}]$, or*
3. *if $[B] \subseteq \mathsf{E}$ and $0 \in [B' \cup \{1\}]$;*

unless $\mathrm{NP} = \mathrm{coNP}$, *no other translations are possible.*

The proof of Theorem 7.4.6 will be established from the lemmas in the remainder of this section.

Lemma 7.4.7 *Let B and B' be finite sets of Boolean functions such that $0 \in [B' \cup \{1\}]$ and $[B] \subseteq \mathsf{N}$, or such that $0 \in [B' \cup \{1\}]$ and $[B] \subseteq \mathsf{E}$. Then there exists a translation from B-autoepistemic logic to B'-circumscription.*

Proof. Let B and B' be finite sets of Boolean functions such that $0 \in [B' \cup \{1\}]$ and $[B] \subseteq \mathsf{N}$. Denote by $\Sigma \subseteq \mathcal{L}_{\text{ae}}(B)$ the given autoepistemic theory. Then Σ is equivalent to a set $\Sigma^+ \cup \Sigma^- \cup \Sigma^L$, where Σ^+ is a set of positive literals, Σ^- is a set of negative literals and Σ^L is a set of L-prefixed formulae and negations of L-prefixed formulae. This representation can be computed efficiently, as can be seen from the proof of Lemma 2.5.6. We map Σ to $\big(\Gamma, (P, Q, Z)\big)$, where $P := \Sigma^-$, $Q := \emptyset$, $Z := \Sigma^+$, and

$$\Gamma := \begin{cases} \Sigma^+ & \text{if } \Sigma \text{ has a consistent stable expansion,} \\ \{0\} & \text{otherwise.} \end{cases}$$

If Σ has a consistent stable expansion, then any (P, Q, Z)-minimal model of Γ sets to 1 all propositions occurring in Σ^+ and to 0 all propositions occurring Σ^-. Hence, $\Sigma \models^{\text{skep}} \varphi$ implies $\Gamma \models^{\text{circ}}_{(P,Q,Z)} \varphi$, for all propositional formulae φ over propositions from Σ. It is easy to see that the converse direction also holds. On the other hand, if Σ does not admit a consistent stable expansion, then all propositional formulae are skeptically implied by both Σ and Γ. This concludes the proof of the first part of the lemma. The second part follows from the observation that any set of autoepistemic B-formulae can be rewritten as a set of autoepistemic formulae without using conjunctions. □

Lemma 7.4.8 *Let B and B' be finite sets of Boolean functions such that $[B] \subseteq \mathsf{L}$ and $[B] \cup \{0\} \subseteq [B' \cup \{1\}]$. Then there exists a translation from B-autoepistemic logic to B'-circumscription.*

Proof. Let B and B' be finite sets of Boolean functions such that $[B] \subseteq \mathsf{L}$ and $[B] \cup \{0\} \subseteq [B' \cup \{1\}]$. Let $\Sigma \subseteq \mathcal{L}_{\text{ae}}(B)$ be the given autoepistemic theory.

Using Lemma 4.2.8, we can transform Σ to a set $\Sigma' \subseteq \mathcal{L}_{\text{ae}}(B)$ such that all L-prefixed formulae in Σ are L-atomic and $\Sigma \models^{\text{skep}} \varphi \iff \Sigma' \models^{\text{skep}} \varphi$ for all $\varphi \in \mathcal{L}$ over propositions from Σ. Moreover, the consistent Σ'-full sets of $\Sigma' \subseteq \mathcal{L}_{\text{ae}}(B)$ can be described as solutions of the system T' from the proof of Lemma 4.2.6 (see also Lemma 7.2.10). Denote by T'' the system of linear equations obtained by applying Gaussian elimination to $T'[x_{s+1}/Lx_{s+1}, \ldots, x_n/Lx_n]$ and let Π be the set of autoepistemic formulae equivalent to the equations in T''. We define the translation to map Σ to $\big(\Gamma', (P, Q, Z)\big)$, where $P := \emptyset$, $Q := \emptyset$, $Z := \mathrm{Vars}(\Gamma)$, and Γ' is the B'-representation of

$$\Gamma := \begin{cases} (\Sigma' \cup \Pi)_{[Lx_1/0,\ldots,Lx_s/0,Lx_{s+1}/p_{s+1},\ldots,Lx_n/p_n]} & \text{if } \Sigma \text{ has a consistent expansion,} \\ \{0\} & \text{otherwise.} \end{cases}$$

To prove the correctness of this translation, first suppose that Σ possesses no consistent stable expansions. Consequently, all propositional formulae are skeptically entailed by Σ, and all propositional formulae are circumscriptively entailed by Γ. Hence, suppose that Σ admits at least one consistent stable expansion. Let $\varphi \in \mathcal{L}$ be over propositions from Σ. If $\Sigma \models^{\text{skep}} \varphi$, then $\Sigma' \cup \Lambda \models \varphi$ for each Σ'-full set Λ. This can be rewritten as $\Sigma' \cup \Pi \models \varphi$ by appealing to the fact that the assignments σ satisfying the set Π and the Σ'-full sets Λ of Σ' are in one-to-one correspondence: to Λ associate the unique assignment satisfying Λ restricted to $\{Lx_i \mid s < i \leq n\}$. Replacing Lx_i by p_{x_i} for all $s < i \leq n$ hence yields $\Gamma \models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$.

As for the converse direction, suppose that Σ does not skeptically entail φ. Then there exists a Σ'-full set Λ such that $\Sigma' \cup \Lambda$ is consistent and $\Sigma' \cup \Lambda \not\models \varphi$. Let σ be the assignment witnessing this fact, that is, $\sigma \models \Sigma' \cup \Lambda$ and $\sigma \not\models \varphi$. By the above argument, we also obtain $\sigma \models \Sigma' \cup \Pi$. Therefore, the assignment σ' defined by $\sigma'(x) := \sigma(x)$ for all $x \in \mathrm{Vars}(\Sigma)$ and $\sigma'(p_{x_i}) := \sigma(Lx_i)$ for all $1 \leq i \leq s$ witnesses $\Gamma \not\models^{\mathrm{circ}}_{(P,Q,Z)} \varphi$.

This concludes the proof, since the B'-representation of $\Gamma \subseteq \mathcal{L}(\{\oplus\})$ can be efficiently computed using Lemma 2.4.2 (3.) and the associativity of $x \oplus y$. $\square$

Lemmas 7.4.7 and 7.4.8 verify the existence of all translations claimed in Theorem 7.4.6. To conclude the proof, we will show that, conditioned on NP $\neq$ coNP, no translations from B-autoepistemic logic to B'-circumscription may exist if $0 \notin [B' \cup \{1\}]$ (Lemma 7.4.9), or if $[B] \not\subseteq \mathsf{N}$, $[B] \not\subseteq \mathsf{E}$, $[B] \not\subseteq \mathsf{L}$ (Lemma 7.4.10), or if $[B] \not\subseteq \mathsf{N}$, $[B] \not\subseteq \mathsf{E}$, $[B] \not\subseteq [B' \cup \{1\}]$ (Lemma 7.4.11).

Lemma 7.4.9 *Let B and B' be finite sets of Boolean functions such that $0 \notin [B' \cup \{1\}]$. Then there exists no translation from B-autoepistemic logic to B'-circumscription.*

Proof. The set $\Sigma := \{Lf\}$ has the unique stable expansion $\mathcal{L}_{\mathrm{ae}}$. However, if $0 \notin [B' \cup \{1\}]$ then $[B] \subseteq \mathsf{R}_1$ or $[B] \subseteq \mathsf{D}$; hence any circumscriptive theory built from B'-formulae is satisfiable. $\square$

Lemma 7.4.10 *Let B and B' be finite sets of Boolean functions such that $\mathsf{M} \subseteq [B \cup \{0,1\}]$. Then there exists no translation from B-autoepistemic logic to B'-circumscription unless* NP $=$ coNP.

Proof. Let B and B' be as in the statement of the lemma. Consequently, $\textsc{Exp}'(B)$, the problem to decide whether a given set of autoepistemic B-formulae has a consistent stable expansion, is Σ_2^{p}-complete. Assume that there exists a translation f from B-autoepistemic logic to B'-circumscription. Then $\textsc{Exp}'(B) \leq_{\mathrm{cd}} \textsc{Sat}(B')$ via $\Sigma \in \textsc{Exp}'(B) \iff \Sigma \not\models^{\mathrm{skep}} 0 \iff \Gamma \not\models^{\mathrm{circ}}_{(P,Q,Z)} 0 \iff \Gamma \in \textsc{Sat}(B')$, where $f(\Sigma) = \big(\Gamma, (P,Q,Z)\big)$. Consequently, NP $= \Sigma_2^{\mathrm{p}}$ and NP $=$ coNP. $\square$

Lemma 7.4.11 *Let B and B' be finite sets of Boolean functions such that $[B \cup \{0,1\}] =$ V and $\mathsf{S}_1 \not\subseteq [B']$, or such that $[B \cup \{0,1\}] = \mathsf{L}$ and $[B] \not\subseteq [B' \cup \{1\}]$ Then there exists no translation from B-autoepistemic logic to B'-circumscription unless* P $=$ NP.

Proof. First, let B and B' be finite sets of Boolean functions such that $[B \cup \{0,1\}] =$ V and $[B' \cup \{1\}] \neq \mathsf{BF}$. Then $[B' \cup \{1\}]$ is situated below R_1, M, or L. The first case follows from Lemma 7.4.9. For the second and third, consider some formulae φ in conjunctive normal form with exactly three literals per clause. Let Σ denote the set of autoepistemic B-formulae constructed in the proof of Lemma 4.2.5 and suppose there exists a translation $\big(\Gamma, (P, Q, Z)\big)$ of Σ to B'-circumscription. Then φ is satisfiable if and only if $\Sigma \not\models^{\mathrm{skep}} 0$ if and only if $\Gamma \not\models^{\mathrm{circ}}_{(P,Q,Z)} 0$ if and only if Γ is satisfiable. But the satisfiability of $\Gamma \subseteq \mathcal{L}(B')$ can be determined in polynomial time for all B' such that $\mathsf{S}_1 \not\subseteq [B']$, see [Lew79]. As $\textsc{Exp}(B)$ is NP-complete by Theorem 4.2.1, $\mathrm{P} = \mathrm{NP}$.

Second, let B and B' be finite sets of Boolean functions $[B \cup \{0,1\}] = \mathsf{L}$ and $[B] \not\subseteq [B' \cup \{1\}]$. Then $[B' \cup \{1\}]$ is situated below R_1, E, or V. The first case again follows from Lemma 7.4.9. For the second and third, we have $x \leftrightarrow y \notin [B' \cup \{1\}]$. The claim now follows from the last paragraph in the proof of Lemma 7.3.9: any translation of $x \leftrightarrow y$ has to circumscribe at least one proposition, from which we eventually obtain a contradiction to its monotonicity. $\square$

Eventually, using Lemmas 7.4.7 to 7.4.11, Theorem 7.4.6 is established.

Chapter 8

Epilogue

In this thesis, we systematically studied the computational complexity of consistency and reasoning problems for fragments of nonmonotonic logics obtained by restricting the available Boolean connectives. We hope that the results presented contribute to a better understanding of the sources of complexity for these problems and lead to better algorithms in situations where the occurring formulae are written over a restricted set of Boolean functions.

Future research in this direction should aim at determining the exact complexity of the problems $\text{CIRCINF}(B)$ and $\text{EXP}(B)$ for $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$, for which we were only able to obtain $\oplus$L-hardness and membership in respectively NP and P. While the latter gap is rather small, it is yet unknown whether the former one is tractable. It can alternatively be rephrased as the question whether in all minimal models of a given knowledge base an odd number of propositions from a given set A is set to 1. When A is restricted to be a singleton set, the problem is known to be polynomial-time solvable, whereas the related question whether some assignment belongs to the set of minimal models is known to be coNP-complete [DH03].

Further, a finer classification of the complexity beyond the usual worst-case measures would be of interest. Despite the fact that the considered consistency and reasoning problems are complete for the second level of the polynomial hierarchy, knowledge representation and reasoning systems are widely used in practice. The question arises why these systems remain manageable. Insights into this question might be gained from the analysis of the structure of typical instances (that is, its average case complexity) or the parameterized complexity of the problems. Here, connectives accounting for jumps in the complexity might lead to parameters that, in connection with other restrictions, yield fixed-parameter tractability. As a simple example, the following "small model credulous reasoning" problem is easily seen to be fixed-parameter tractable:

Input:	Formulae $\varphi, \psi \in \mathcal{L}_{\mathrm{ae}}(\{\wedge, \vee\})$ in conjunctive normal form
Parameter:	The number of (binary) $\wedge$-connectives in φ and $k \in \mathbb{N}$
Question:	Does φ admit a stable expansion Δ with $\lvert\Delta \cap \mathrm{SF}^L(\varphi)\rvert = k$ such that $\psi \in \Delta$?

Given the number l of conjunctions in φ, it follows from the monotonicity of φ that to any model σ of φ there exists a model σ' such that $\lvert\sigma'\rvert \leq l+1$ and $\sigma' \models \varphi$.

To test whether ψ is contained in a stable expansion Δ with $|\Delta \cap \mathrm{SF}^L(\varphi)| = k$, it hence suffices to verify that ψ is satisfied in all models of φ setting to 1 at most l propositions. This combined with the restriction of the search space of the stable expansions yields a fixed-parameter tractable algorithm. But if the number of $\wedge$-connectives is dropped from the parameter, the problem becomes fixed-parameter intractable. Indeed, we can reduce the A[2]-complete problem p-AWSAT$_2(\Delta^+_{1,2})$ to the above problem parameterized by the number of L-operators only (for the definition of the problem p-AWSAT$_2(\Delta^+_{1,2})$ and the class A[2], see [FG06]).

In addition to the consistency and reasoning problems, we presented new counting problems from the area of knowledge representation and reasoning that are complete for the first and second level of the counting hierarchy. These problems may prove helpful in deriving the counting complexity of similar problems in the area of knowledge representation and nonmonotonic reasoning.

Finally, we examined the existence of translations between the above mentioned fragments of nonmonotonic logics and exhibited those that admit translations to other nonmonotonic logics. These results were complemented by showing that, conditioned on the strictness of the polynomial hierarchy, in almost all cases in which no translation was given indeed no translation may exist. The open cases can be summarized as follows:

1. Translations from B-default logic
   a. for $[B \cup \{0,1\}] = \mathsf{V}$ to B'-autoepistemic logic for $[B \cup \{0,1\}] = \mathsf{L}$
   b. for $[B \cup \{0,1\}] = \mathsf{V}$ to B'-circumscription for $[B \cup \{0,1\}] = \mathsf{L}$
   c. for $[B \cup \{0,1\}] \in \{\mathsf{N}, \mathsf{L}, \mathsf{M}\}$ to B'-circumscription for $[B' \cup \{1\}] = \mathsf{BF}$
2. Translations from B-autoepistemic logic
   a. for $[B \cup \{0,1\}] = \mathsf{L}$ to B-default logic for $[B \cup \{1\}] = \mathsf{N}$
   b. for $[B \cup \{0,1\}] = \mathsf{V}$ to B'-default logic for $[B' \cup \{1\}] = \mathsf{M}$
   c. for $[B \cup \{0,1\}] = \mathsf{V}$ to B'-circumscription for $[B' \cup \{1\}] = \mathsf{BF}$
3. Translations from B-circumscription
   a. for $[B \cup \{0,1\}] \in \{\mathsf{V}, \mathsf{M}\}$ to B'-default logic for $[B \cup \{0,1\}] \in \{\mathsf{N}, \mathsf{L}\}$
   b. for $[B \cup \{0,1\}] = \mathsf{L}$ to B'-default logic for $[B \cup \{1\}] = \mathsf{N}$
   c. for $[B \cup \{0,1\}] = \mathsf{L}$ to B'-autoepistemic logic for $[B \cup \{0,1\}] = \mathsf{L}$

The difficulty in proving or refuting the existence of a translation in these cases arises from different sources.

First, the disjunctive fragments of default logic are no more expressive than the disjunctive fragments of propositional logic. The existence of a translation consequently reduces to the question whether there exists an affine autoepistemic (respectively an affine circumscriptive theory) that can be computed in polynomial time and is equivalent to the propositional representation of the given default theory. This pertains the cases (1.a.) and (1.b.).

Second, nonmonotonicity might allow for the simulation of connectives not present in the set of available Boolean connectives. To refute the existence of a translation, counterexamples would hence have to take the extended capabilities of the target logic into account. On the other hand, there seems to be no obvious construction for translations. This subsumes (1.c.), (2.b.), (2.c.), and (3.a.).

Third, for (2.a.), (3.b.), and (3.c.), the expressive power of the fragment (or its exact computational complexity) are not known. In particular, establishing a translation for (3.c.) would lead to a polynomial-time upper bound for the circumscriptive inference problem for affine B-formulae, $\textsc{CircInf}(B)$ for $\mathsf{L}_2 \subseteq [B] \subseteq \mathsf{L}$, and also resolve (3.c.). Closing these gaps would hence lead to insights into the expressiveness and the computational complexity of the corresponding fragments.

The results on the intertranslatability of nonmonotonic logics are particularly interesting in light of their connections to logic programming using the stable set semantics: Bidoit and Froidevaux showed that general logic programs coincide with the fragment of default logic obtained by restricting the knowledge base, prerequisites and conclusions to positive literals, and justifications to negative literals [BF87]. In particular, extended logic programs coincide with B-default logic for $[B] = \mathsf{N}$ [GL91]. For extended disjunctive logic programs, similar connections exist to autoepistemic logic [LS93]. Thus, Theorem 7.2.1 reproves parts of the embedding of extended logic programs to autoepistemic logic found in the latter article. Further investigating these connections and the resulting consequences would be very interesting.

Bibliography

[AB09] S. Arora and B. Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.

[BBC$^+$09] M. Bauland, E. Böhler, N. Creignou, S. Reith, H. Schnoor, and H. Vollmer. The complexity of problems for quantified constraints. *Theory Comput. Syst.*, 2009. Electronically available at http://dx.doi.org/10.1007/s00224-009-9194-6.

[BDHM92] G. Buntrock, C. Damm, U. Hertrampf, and C. Meinel. Structure and importance of logspace-mod class. *Mathematical Systems Theory*, 25(3):223–237, 1992.

[Bes89] P. Besnard. *Introduction to Default Logic*. Springer, 1989.

[BF87] N. Bidoit and C. Froidevaux. Minimalism subsumes default logic and circumscription in stratified logic programming. In *Proc. 2nd Symposium on Logic in Computer Science*, pages 89–97. IEEE Computer Society, 1987.

[BHSS06] M. Bauland, E. Hemaspaandra, H. Schnoor, and I. Schnoor. Generalized modal satisfiability. In *Proc. 23rd Annual Symposium on Theoretical Aspects of Computer Science*, volume 3884 of *Lecture Notes in Computer Science*, pages 500–511. Springer, 2006.

[BMS$^+$09] M. Bauland, M. Mundhenk, T. Schneider, H. Schnoor, I. Schnoor, and H. Vollmer. The tractability of model-checking for LTL: The good, the bad, and the ugly fragments. *Electr. Notes Theor. Comput. Sci.*, 231:277–292, 2009.

[BMTV09a] O. Beyersdorff, A. Meier, M. Thomas, and H. Vollmer. The complexity of propositional implication. *Inf. Process. Lett.*, 109(18):1071–1077, 2009.

[BMTV09b] O. Beyersdorff, A. Meier, M. Thomas, and H. Vollmer. The complexity of reasoning for fragments of default logic. In *Proc. 12th International Conference on Theory and Applications of Satisfiability Testing*, volume 5584 of *Lecture Notes in Computer Science*, pages 51–64. Springer, 2009.

[BO02] P. A. Bonatti and N. Olivetti. Sequent calculi for propositional nonmonotonic logics. *ACM Trans. Comput. Log.*, 3(2):226–278, 2002.

[BSS+08] M. Bauland, T. Schneider, H. Schnoor, I. Schnoor, and H. Vollmer. The complexity of generalized satisfiability for linear temporal logic. *Logical Methods in Computer Science*, 5(1), 2008.

[CHS07] P. Chapdelaine, M. Hermann, and I. Schnoor. Complexity of default logic on generalized conjunctive queries. In *Proc. 9th International Conference on Logic Programming and Nonmonotonic Reasoning*, volume 4483 of *Lecture Notes in Computer Science*, pages 58–70. Springer, 2007.

[CL90] M. Cadoli and M. Lenzerini. The complexity of closed world reasoning and circumscription. In *Proc. 8th National Conference on Artificial Intelligence*, pages 550–555. AAAI Press, 1990.

[CMTV10] N. Creignou, A. Meier, M. Thomas, and H. Vollmer. The complexity of reasoning for fragments of autoepistemic logic. In *Circuits, Logic, and Games*, volume 10061 of *Dagstuhl Seminar Proceedings*, 2010.

[Coo71] S. A. Cook. The complexity of theorem-proving procedures. In *Proc. 3rd Annual ACM Symposium on Theory of Computing*, pages 151–158. ACM, 1971.

[CST10] N. Creignou, J. Schmidt, and M. Thomas. Complexity of propositional abduction for restricted sets of Boolean functions. In *Proc. 12th International Conference on Principles of Knowledge Representation and Reasoning*. AAAI, 2010.

[CSV84] A. K. Chandra, L. J. Stockmeyer, and U. Vishkin. Constant depth reducibility. *SIAM J. Comput.*, 13(2):423–439, 1984.

[DH03] A. Durand and M. Hermann. The inference problem for propositional circumscription of affine formulas is coNP-complete. In *Proc. 20th Annual Symposium on Theoretical Aspects of Computer Science*, volume 2607 of *Lecture Notes in Computer Science*, pages 451–462. Springer, 2003.

[DH08] A. Durand and M. Hermann. On the counting complexity of propositional circumscription. *Inf. Process. Lett.*, 106(4):164–170, 2008.

[DHK05] A. Durand, M. Hermann, and P. G. Kolaitis. Subtractive reductions and complete problems for counting complexity classes. *Theor. Comput. Sci.*, 340(3):496–513, 2005.

[EG93] T. Eiter and G. Gottlob. Propositional circumscription and extended closed-world reasoning are Π_2^p-complete. *Theor. Comput. Sci.*, 114(2):231–245, 1993.

[Eth87] D. W. Etherington. Relating default logic and circumscription. In *Proc. 10th International Joint Conference on Artificial Intelligence*, pages 489–494. Morgan Kaufman, 1987.

[FG06] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer, 2006.

[FSS84] M. L. Furst, J. B. Saxe, and M. Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*, 17(1):13–27, 1984.

[GL91] M. Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Comput.*, 9(3/4):365–386, 1991.

[Got92] G. Gottlob. Complexity results for nonmonotonic logics. *J. Log. Comput.*, 2(3):397–425, 1992.

[Got95a] G. Gottlob. NP trees and Carnap's modal logic. *J. ACM*, 42(2):421–457, 1995.

[Got95b] G. Gottlob. Translating default logic into standard autoepistemic logic. *J. ACM*, 42(4):711–740, 1995.

[GPP89] M. Gelfond, H. Przymusinska, and T. C. Przymusinski. On the relationship between circumscription and negation as failure. *Artif. Intell.*, 38(1):75–94, 1989.

[HP07] M. Hermann and R. Pichler. Counting complexity of propositional abduction. In *Proc. 20th International Joint Conference on Artificial Intelligence*, pages 417–422, 2007.

[HV95] L. Hemaspaandra and H. Vollmer. The satanic notations: counting classes beyond #P and other definitional adventures. *Complexity Theory Column 8, ACM-SIGACT News*, 26(1):2–13, 1995.

[Imi87] T. Imielinski. Results on translating defaults to circumscription. *Artif. Intell.*, 32(1):131–146, 1987.

[Jan99] T. Janhunen. On the intertranslatability of non-monotonic logics. *Ann. Math. Artif. Intell.*, 27(1-4):79–128, 1999.

[Jar30] V. Jarník. O jistém problému minimálním. *Práce Mor. Přírodověd. Spol. v Brně*, 6:57–63, 1930. English translation: About a certain minimal problem.

[Jon75] N. D. Jones. Space-bounded reducibility among combinatorial problems. *J. Comput. Syst. Sci.*, 11(1):68–85, 1975.

[JPRW08] M. Jakl, R. Pichler, S. Rümmele, and S. Woltran. Fast counting with bounded treewidth. In *Proc. 15th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning*, volume 5330 of *Lecture Notes in Computer Science*, pages 436–450. Springer, 2008.

[KK01a] L. M. Kirousis and P. G. Kolaitis. The complexity of minimal satisfiability in Post's lattice. Unpublished note, 2001.

[KK01b] L. M. Kirousis and P. G. Kolaitis. The complexity of minimal satisfiability problems. In *Proc. 18th Annual Symposium on Theoretical Aspects of Computer Science*, volume 2010 of *Lecture Notes in Computer Science*, pages 407–418. Springer, 2001.

[KK03] L. M. Kirousis and P. G. Kolaitis. The complexity of minimal satisfiability problems. *Inf. Comput.*, 187(1):20–39, 2003.

[Kon88] K. Konolige. On the relation between default and autoepistemic logic. *Artif. Intell.*, 35(3):343–382, 1988.

[Kon89] K. Konolige. On the relation between autoepistemic logic and circumscription. In *Proc. 11th International Joint Conference on Artificial Intelligence*, pages 1213–1218. Morgan Kaufman, 1989.

[KS91] H. A. Kautz and B. Selman. Hard problems for simple default logics. *Artif. Intell.*, 49(1-3):243–279, 1991.

[Lad75] R. E. Ladner. On the structure of polynomial time reducibility. *J. ACM*, 22(1):155–171, 1975.

[Lev73] L. A. Levin. Universal sequential search problems. *Problemi Peredachi Informatsii*, 9(3):115–116, 1973. English translation: *Problems of Information Transmission*, 9(3):265–266.

[Lew79] H. R. Lewis. Satisfiability problems for propositional calculi. *Mathematical Systems Theory*, 13:45–53, 1979.

[Lif85] V. Lifschitz. Computing circumscription. In *Proc. 9th International Joint Conference on Artificial Intelligence*, pages 121–127. Morgan Kaufman, 1985.

[LMS01] F. Laroussinie, N. Markey, and P. Schnoebelen. Model checking CTL$^+$ and FCTL is hard. In *Proc. 4th International Conference on Foundations of Software Science and Computation Structures*, volume 2030 of *Lecture Notes in Computer Science*, pages 318–331. Springer, 2001.

[LS93] V. Lifschitz and G. Schwarz. Extended logic programs as autoepistemic theories. In *Proc. 2nd International Conference on Logic Programming and Nonmonotonic Reasoning*, pages 101–114, 1993.

[McC80] J. McCarthy. Circumscription – a form of non-monotonic reasoning. *Artif. Intell.*, 13(1-2):27–39, 1980.

[McD82] D. V. McDermott. Nonmonotonic logic II: Nonmonotonic modal theories. *J. ACM*, 29(1):33–57, 1982.

[MD80] D. V. McDermott and J. Doyle. Non-monotonic logic I. *Artif. Intell.*, 13(1-2):41–72, 1980.

[MMS$^+$09] A. Meier, M. Mundhenk, T. Schneider, M. Thomas, V. Weber, and F. Weiss. The complexity of satisfiability for fragments of hybrid logic—part I. In *Proc. 34th International Symposium on Mathematical Foundations of Computer Science*, volume 5734 of *Lecture Notes in Computer Science*, pages 587–599. Springer, 2009.

[MMTV09] A. Meier, M. Mundhenk, M. Thomas, and H. Vollmer. The complexity of satisfiability for fragments of CTL and CTL*. *Int. J. Found. Comput. Sci.*, 20(5):901–918, 2009.

[Moo85] R. C. Moore. Semantical considerations on nonmonotonic logic. *Artif. Intell.*, 25(1):75–94, 1985.

[MS72] A. R. Meyer and L. J. Stockmeyer. The equivalence problem for regular expressions with squaring requires exponential space. In *Proc. 13th Annual Symposium on Switching and Automata Theory*, pages 125–129. IEEE, 1972.

[MT89] V. W. Marek and M. Truszczynski. Relating autoepistemic and default logics. In *Proc. 1st International Conference on Principles of Knowledge Representation and Reasoning*, pages 276–288. Morgan Kaufmann, 1989.

[MT90] V. W. Marek and M. Truszczynski. Modal logic for default reasoning. *Ann. Math. Artif. Intell.*, 1, 1990.

[MT93] V. W. Marek and M. Truszczyński. *Nonmonotonic Logic*. Springer, 1993.

[Nie90] I. Niemelä. Towards automatic autoepistemic reasoning. In *Proc. 2th European Conference on Logics in Artificial Intelligence*, volume 478 of *Lecture Notes in Computer Science*, pages 428–443. Springer, 1990.

[Nie93] I. Niemelä. *Autoepistemic Logic as a Unified Basis for Nonmonotonic Reasoning*. PhD thesis, Helsinki University of Technology, Digital Systems Laboratory, Espoo, Finland, August 1993.

[Nor04] G. Nordh. A trichotomy in the complexity of propositional circumscription. In *Proc. 11th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning*, volume 3452 of *Lecture Notes in Computer Science*, pages 257–269. Springer, 2004.

[NR94] I. Niemelä and J. Rintanen. On the impact of stratification on the complexity of nonmonotonic reasoning. *Journal of Applied Non-Classical Logics*, 4(2), 1994.

[Pap94] C. M. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[Pip97] N. Pippenger. *Theories of Computability*. Cambridge University Press, 1997.

[Pos41] E. Post. The two-valued iterative systems of mathematical logic. *Annals of Mathematical Studies*, 5:1–122, 1941.

[Pri57] R. C. Prim. Shortest connection networks and some generalizations. *Bell System Technical Journal*, 36:1389–1401, 1957.

[Rei80] R. Reiter. A logic for default reasoning. *Artif. Intell.*, 13(1-2):81–132, 1980.

[Rei03] S. Reith. On the complexity of some equivalence problems for propositional calculi. In *Proc. 28th International Symposium on Mathematical Foundations of Computer Science*, volume 2747 of *Lecture Notes in Computer Science*, pages 632–641. Springer, 2003.

[Sav70] W. J. Savitch. Relationships between nondeterministic and deterministic tape complexities. *J. Comput. Syst. Sci.*, 4(2):177–192, 1970.

[Sch78] T. J. Schaefer. The complexity of satisfiability problems. In *Proc. 10th Annual ACM Symposium on Theory of Computing*, pages 216–226. ACM, 1978.

[Sch96] G. Schwarz. On embedding default logic into moore's autoepistemic logic. *Artif. Intell.*, 80(1-2):349–359, 1996.

[Sch10] H. Schnoor. The complexity of model checking for boolean formulas. *Int. J. Found. Comput. Sci.*, 21(3):289–309, 2010.

[Sel82] A. L. Selman. Reductions on np and p-selective sets. *Theor. Comput. Sci.*, 19:287–304, 1982.

[Shv90] G. F. Shvarts. Autoepistemic modal logics. In *Proc. 3rd Conference on Theoretical Aspects of Reasoning about Knowledge*, pages 97–109. Morgan Kaufmann, 1990.

[SI90] R. Sridhar and S. S. Iyengar. Efficient parallel algorithms for functional dependency manipulations. In *Proc. 2nd International Symposium on Databases in Parallel and Distributed Systems*, pages 126–137. IEEE, 1990.

[Sip05] M. Sipser. *Introduction to the Theory of Computation*. Course Technology, 2005.

[Smo87] R. Smolensky. Algebraic methods in the theory of lower bounds for Boolean circuit complexity. In *Proc. 19th Annual ACM Symposium on Theory of Computing*, pages 77–82. ACM, 1987.

[Spi71] P. M. Spira. On time-hardware complexity tradeoffs for Boolean functions. In *Proc. 4th Hawaii International Symposium on System Sciences*, pages 525–527, 1971.

[Tho09] M. Thomas. The complexity of circumscriptive inference in Post's lattice. In *Proc. 10th International Conference on Logic Programming and Nonmonotonic Reasoning*, volume 5753 of *Lecture Notes in Computer Science*, pages 290–302. Springer, 2009.

[Tod91] S. Toda. PP is as hard as the polynomial-time hierarchy. *SIAM J. Comput.*, 20(5):865–877, 1991.

[TW92] S. Toda and O. Watanabe. Polynomial time 1-Turing reductions from #PH to #P. *Theor. Comput. Sci.*, 100(1):205–221, 1992.

[Val79a] L. G. Valiant. The complexity of computing the permanent. *Theor. Comput. Sci.*, 8:189–201, 1979.

[Val79b] L. G. Valiant. The complexity of enumeration and reliability problems. *SIAM J. Comput.*, 8(3):410–421, 1979.

[Vol99] H. Vollmer. *Introduction to Circuit Complexity – A Uniform Approach*. Springer, 1999.

[Wra76] C. Wrathall. Complete sets and the polynomial-time hierarchy. *Theor. Comput. Sci.*, 3(1):23–33, 1976.

[Zan91] V. Zanko. #P-completeness via many-one reductions. *Int. J. Found. Comput. Sci.*, 2(1):77–82, 1991.

Index

Lebenslauf

Persönliche Daten

Name	Michael Thomas
Geburt	23. April 1981 in Hannover

Schulausbildung

08/1987–06/1991	Grundschule Greifswalder Straße in Hildesheim
08/1991–06/1993	Orientierungsstufe Ost in Hildesheim
08/1993–06/2000	Scharnhorst Gymnasiums in Hildesheim
06/2000	Abitur

Studium

10/2001–07/2004	Studium des Fachs „Angewandte Informatik" an der Leibniz Universität Hannover
07/2004	Bachelor of Science Auszeichnung für die Bachelorarbeit
10/2004–08/2006	Studium des Fachs „Informatik" an der Leibniz Universität Hannover
08/2006	Master of Science

Akademische Laufbahn

08/2006–07/2007	Wissenschaftlicher Mitarbeiter am Institut für Systems Engineering an der Leibniz Universität Hannover
seit 08/2007	Wissenschaftlicher Mitarbeiter am Institut für Theoretische Informatik an der Leibniz Universität Hannover

Ausserberufliches

08/2000–06/2001	Zivildienst bei der Johanniter Unfall Hilfe Hildesheim
07.09.2007	Hochzeit mit Annika Thomas, geb. Biener