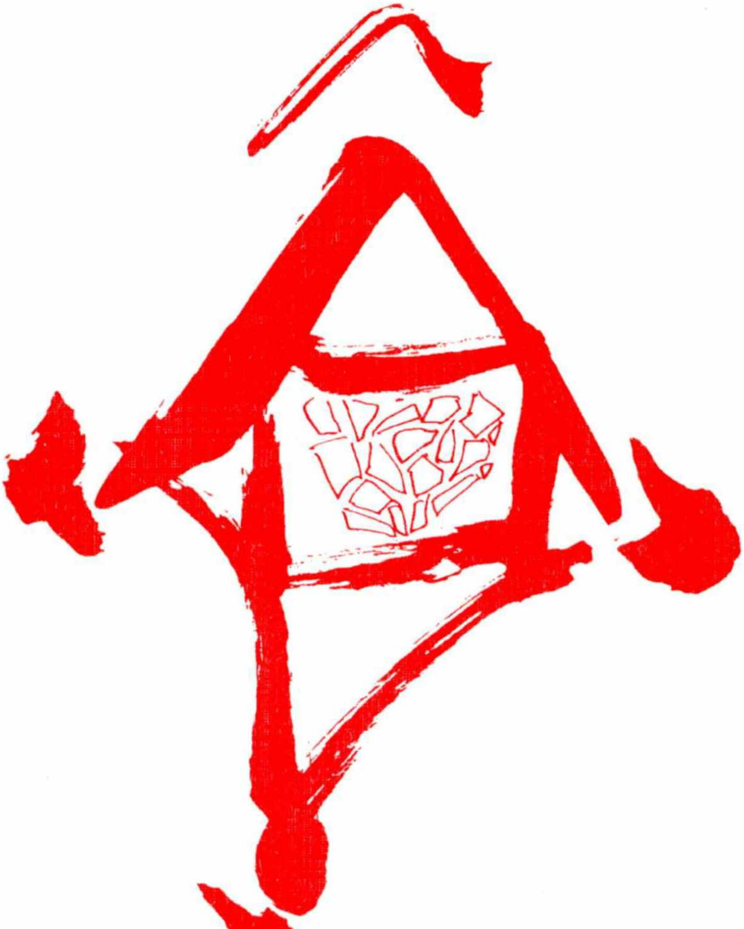
Extending Modal Logic



Maarten de Rijke

Extending Modal Logic

ILLC Dissertation Series 1993-4

institute for logic, language and computation

For further information about ILLC-publications, please contact:

Institute for Logic, Language and Computation
Universiteit van Amsterdam
Plantage Muidergracht 24
1018 TV Amsterdam
phone: +31-20-5256090
fax: +31-20-5255101
e-mail: illc@fwi.uva.nl

Extending Modal Logic

Academisch Proefschrift

ter verkrijging van de graad van doctor aan de
Universiteit van Amsterdam,
op gezag van de Rector Magnificus
Prof. Dr. P.W.M. de Meijer
in het openbaar te verdedigen in de
Aula der Universiteit
(Oude Lutherse Kerk, ingang Singel 411, hoek Spui)
op vrijdag 3 december 1993 te 15.00 uur

door

Maarten de Rijke

geboren te Vlissingen.

Promotor: Prof.dr. J.F.A.K. van Benthem
Faculteit Wiskunde en Informatica
Universiteit van Amsterdam
Plantage Muidergracht 24
1018 TV Amsterdam

The investigations were supported by the Philosophy Research Foundation (SWON), which is subsidized by the Netherlands Organization for Scientific Research (NWO).

Copyright © 1993 by Maarten de Rijke

Cover design by Han Koster.
Typeset in LaTeX by the author. Camera-ready copy produced at the Faculteit der Wiskunde en Informatica, Vrije Universiteit Amsterdam.
Printed and bound by Haveka B.V., Alblasserdam.

ISBN: 90–800769–9–6

Contents

Acknowledgments

It takes four ingredients to cook an agreeable PhD project: (i) the right supervision, (ii) the right co-authors and colleagues, (iii) sufficiently many projects and side activities running in parallel, and (iv) plenty of trips abroad. I would like to thank several people and organizations for providing these ingredients and making my four years as a PhD student a valuable experience.

When I arrived as a PhD student it was agreed that Dick de Jongh would be my supervisor for the first six months. With his questions and insights Dick helped me deliver my first reports (which, by the way, are still among my favorite 'children'). While Dick was, so to say, the person who taught me how to fly, Johan van Benthem, who took over from Dick, showed me exotic destinations I could fly to. His special style of 'long-distance supervision' was the right thing for me at the right time, leaving me enough elbow-room to follow side routes, but also making sure that I didn't forget about the main route: writing this dissertation. Thank you Dick and Johan!

One of the most rewarding academic things must be co-working on one project or another. I had the pleasure of co-writing a paper and/or book with a number of people. Thank you Andrei Arsov, Patrick Blackburn, Claire Gardent, Edith Hemaspaandra, Wiebe van der Hoek, Dick de Jongh, Yde Venema, and Roel Wieringa! Further people I would like to thank for contributing to my scientific well-being during the past four years include Valentin Goranko, David Israel, Marcus Kracht, Valentin Shehtman, Dimiter Vakarelov, Frank Veltman, Heinrich Wansing, and all the colleagues from the Department of Mathematics and Computer Science of the University of Amsterdam.

An important maxim for PhD students that is often overlooked is that the best way to focus on a large project is to have plenty of side activities for distraction. One of my main and most rewarding side routes has been the organization of the Seminar on Intensional Logic. I would like to thank the many people that attended the Seminar, the people that presented a paper before the Seminar, and the people that were involved with one of the "spin-offs" of the Seminar. In addition I would like to thank the Dutch Network for Logic, Language and Information (TLI), the European Community, and the Department of Mathematics and Computer Science of the University of Amsterdam for making it all financially possible.

As for the fourth essential ingredient for enjoyable PhD projects, I am very grateful to the Department of Mathematics and Computer Science of the University of Amsterdam, the ERASMUS project, NWO, SWON, the TEMPUS project, and TLI for financing my trips to Belgium, France, Sweden, England, the USA, Hungary, Bulgaria, Spain, Portugal, and Germany.

This dissertation has benefitted from valuable input by Johan van Benthem, Patrick Blackburn, Peter van Emde Boas, Wiebe van der Hoek (who generously read and re-read the entire manuscript), and Yde Venema. A special thanks to Vincent van Oostrom for his help in printing the camera-ready copy.

At a more personal level I owe a lot to a small number of people. It's best that I try and settle my large debt to them in a private setting.

Amstelveen
October, 1993.

Maarten de Rijke

Part I

Introduction

1

What this dissertation is about

This dissertation is about extending modal logic. It tells you what a system of extended modal logic is, it gives you three case studies of systems of modal logic, and it gives you very general approaches to two important themes in modal logic.

This Chapter offers a brief overview of the dissertation.

1.1 Modal logic

According to the general framework presented in Chapter 2 below, a modal language is first and foremost a description language for relational structures. Its distinguishing features are

1. the use of simple restricted relational patterns to describe the truth conditions of its operators,
2. the use of multiple sorts,
3. a concern for the fine structure of model theory.

To illustrate these points, recall that the *standard modal language* $\mathcal{ML}(\Diamond)$ has its formulas built up from proposition letters p, the usual Boolean connectives $\neg$ and $\wedge$, and a unary modal operator $\Diamond$. Models for the language $\mathcal{ML}(\Diamond)$ assign *subsets* P of the universe to proposition letters, and use a binary *relation* R to interpret the modal operator: $x \models \Diamond p$ iff for some y both Rxy and Py hold. Observe that this truth definition for the diamond $\Diamond$ involves two sorts of objects (a relation and a set), and that it involves only two individual variables. Indeed, as was first observed by Dov Gabbay, when interpreted on models the standard modal language is essentially a restricted fragment of a first-order language employing only two variables.

Moreover, due to the *restricted* quantification in the truth definition of $\Diamond$, the satisfiability problem for $\mathcal{ML}(\Diamond)$ is decidable, and the model theory of the language can be analyzed quite elegantly with bisimulations. This restricted quantification also makes sure that we have a decent proof theory for $\mathcal{ML}(\Diamond)$.

1.2 Extending modal logic

Quite often the need arises to go beyond a given description language. In the case of modal logic viewed as a tool for talking about relational structures this is usually because of one of the following two reasons:

- it may be necessary to capture more aspects of the structures we work with than we are able to with our current modal language: we need to increase the expressive power of our description language,
- as interests and research directions change new kinds of structures come into focus, calling for novel languages that employ appropriate vocabularies.

An extensive list of examples illustrating both of the above phenomena is given in Chapter 2. Chapters 3, 4 and 5 below contain case studies of formalisms whose *raison d'être* may be traced back to one (or both) of the above points.

The standard modal language cannot express several natural properties of its underlying semantic structures. Chapter 3 analyzes a simple proposal to overcome some of the most striking of these defects: the addition of a *difference operator* D whose semantic definition reads: $D\phi$ is true at a state iff ϕ is true at a *different* state.

In recent years modal logic has found novel applications in artificial intelligence, natural language analysis, knowledge representation, and theoretical computer science at large. Several general themes that form rich sources of fresh and extended modal formalisms go by such names as 'change,' 'dynamics,' and 'information.' Chapters 4 and 5 below both deal with modal languages originating in, or relevant to, areas subsumed by those themes.

With the emergence of every new modal language the usual set of research topics and questions concerning, for example, expressive power and axiomatic aspects, carries over from our old formalisms. However, new questions arise as well. Chapter 2 describes a number of those new issues in extended modal logic, one of which has to do with *generalizations* of results and techniques known from traditional modal languages. In Chapters 6 and 7 two themes ('Modal Logic and Bisimulations' and 'Correspondence Theory') from standard modal logic are generalized, and studied in an abstract setting which allows for application of the results to large classes of modal languages.

1.3 A LOOK AHEAD

I now give a brief chronological survey of things to come.

As new and extended modal formalisms, some differing vastly from others, emerge in a variety of research areas, it becomes imperative to ask what it is that constitutes a modal logic. Chapter 2 offers a unifying framework for modal logic and identifies research lines, thus setting the stage for the rest of the dissertation.

Chapters 3, 4 and 5 contain case studies of three extended modal languages. Chapter 3 introduces one of the main characters of the dissertation: the D-operator. Its basic theory is developed, answering most of the usual questions one is interested in for any system of modal logic: axiomatics, decidability, and expressive power. In addition, Chapter 3 serves as an introduction to the basic terminology and techniques used in the dissertation.

A certain system of dynamic modal logic is analyzed in Chapter 4. According to the view advocated there, one of the most interesting aspects of dynamics is the *interaction* between static and dynamic information. Hence, the dynamic modal language of Chapter 4 has explicit operators for reasoning about such interactions. We briefly mention some of the uses of the language, and then develop its basic theory by investigating its expressive power, the complexity of its satisfiability problem, and its axiomatic aspects; to arrive at a complete axiomatization we call in the help of the D-operator from Chapter 3.

The modal algebras underlying the dynamic modal logic of Chapter 4 are the topic of Chapter 5. Those algebras, called Peirce algebras, have two sorts, sets and relations, and various operators to take you from one sort to the other. The bulk of the Chapter is devoted to finding a complete axiomatization for those algebras — this is done almost entirely inside modal logic, in a fully two-sorted extension of the earlier dynamic modal logic. In this Chapter too we use the D-operator from Chapter 3 in an essential way to arrive at our completeness results.

Chapters 6 and 7 are both concerned with generalizations of themes from standard modal logic. Chapter 6 develops the model theory of a class of basic modal languages using bisimulations as the fundamental tool. This results in modal analogues of many results from basic first-order model theory: a Keisler-Shelah style theorem, definability results, a Lindström type result, and various preservation results.

Chapter 7 then views modal formulas as special kinds of higher-order conditions on the underlying semantic structures, or frames; it formulates several abstract and very general algorithms that in many important cases reduce such modal higher-order conditions to simpler ones. The classes of formulas suitable as input for those reduction algorithms are described semantically and syntactically, and the algorithms are shown to be applicable to a wide variety of modal logics.

Finally, the appendix contains a few facts and notions from classical, modal and algebraic logic that are used, but not explained elsewhere in the dissertation.

Before starting off, a brief word about the origin of the remaining Chapters.

A version of Chapter 2 is to appear as 'What is Modal Logic?' in M. Masuch & L. Pólos, eds, 'Arrow Logic and Multi-modal Logic.'

Chapter 3 is an updated and expanded version of 'The Modal Logic of Inequality,' *Journal of Symbolic Logic* **57**, 566–584, 1992.

A version of Chapter 4 is to appear as 'A System of Dynamic Modal Logic,' in the *Journal of Philosophical Logic*.

Chapters 5, 6 and 7 were written especially for this occasion.

2

What is Modal Logic?

2.1 Introduction

This Chapter contains no results. Instead, it is concerned with methodological issues in what may be called *extended (propositional) modal logic*, a rapidly expanding and active field that comprises of modal formalisms that differ in important aspects from the traditional format by extending or restricting it in a variety of ways. The Chapter surveys the parameters along which extensions of the standard modal format have been carried out, it proposes a general framework for modal logic, and formulates research topics that arise naturally in this setting. Thus this Chapter provides the setting for the analysis of three modal languages in Chapters 3, 4 and 5, and for the general themes addressed in Chapters 6 and 7.

It has been a long time since modal logic (ML) dealt with just two operators $\Diamond$ and $\Box$. Nowadays every possible way of deviating from the syntactic, semantic and algebraic notions pertaining to this familiar duo seems to be explored. The creation of such new, or extended modal logics is largely application driven. In many applications ML is used as a description language for relational structures. This connection with relational structures makes ML into a powerful tool — besides ML they occur naturally in many parts of linguistics, mathematics, computer science and artificial intelligence. As new (aspects of) structures become important the need arises to go beyond existing modal formalisms to more powerful ones. Or, on the other hand, it may be necessary to consider weaker or restricted versions of earlier languages because of computational concerns.

This Chapter offers a framework for modal logic that is meant to deal with the current diversity of the field.[1] The benefits of this general approach are manifold:

- The framework offers a systematic view of modal logic, identifying systems of ML as description languages for relational structures, concerned with fine-structural aspects of model theory. This will help avoid future 'explosions' of techniques serving essentially similar techniques.
- The framework identifies new possibilities and indicates new questions; section 2.4 below contains examples. They have to do with

[1] Although this is an important issue in itself, the framework is not meant to address more philosophical issues such as which notions are amenable for analysis by means of modal operators.

1. exploring the universe of modal logics and languages,
2. generalizing known results,
3. classification of modal logics and languages,
4. combining modal logics.

- Just as Abstract Model Theory is a large industry of interesting results at a high level of abstraction, so may an abstract approach to modal logic reveal general properties of and connections between modal logics.

The rest of this Chapter is organized as follows. Section 2.2 introduces the framework. §2.3 contains a series of examples of systems of ML that fit inside this framework. Section 2.4 discusses the framework and research topics it naturally gives rise to. The fifth section has some concluding remarks.

2.2 A FRAMEWORK FOR MODAL LOGIC

To have an example at hand, recall that the standard modal language $\mathcal{ML}(\Diamond)$ has its formulas built up from proposition letters p taken from a set Φ, according to the rule $\phi ::= p \mid \bot \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \Diamond\phi \mid \Box\phi$. A model for $\mathcal{ML}(\Diamond)$ is a tuple $\mathfrak{M} = (W, R, V)$ with $W \neq \emptyset$, $R \subseteq W^2$, and V a valuation, that is a function $\Phi \to 2^W$. Truth is defined by $\mathfrak{M}, x \models p$ iff $x \in V(p)$, and the usual clauses for the Booleans, while $\mathfrak{M}, x \models \Diamond\phi$ iff for some y, Rxy and $\mathfrak{M}, y \models \phi$, and $\mathfrak{M}, x \models \Box\phi$ iff $\mathfrak{M}, x \models \neg\Diamond\neg\phi$.

$\mathcal{ML}(\Diamond)$ expresses (first-order) properties of relational structures via its truth conditions. These properties don't exhaust the first-order spectrum — only simple graphical *patterns* of the form "there is an R-successor with a property P" (and combinations thereof) are considered. This will be the hall-mark of all systems of ML: to focus only on restricted aspects of relational structures via truth conditions that encode simple geometric patterns involving multiple sorts.

To present a system of ML we need to give its syntax, its semantic structures, and an interpretation linking the two.

SYNTAX

A general syntax for ML should be able to handle multiple sorts, as well as variables, constants and connectives of each sort, plus, of course, modal operators.

2.2.1. DEFINITION. The *syntax* of a system of modal logic is given by a vocabulary $(\mathcal{S}, \mathcal{V}, \mathcal{C}, \mathcal{O}, \mathcal{F})$ where

- $\mathcal{S}$ is a non-empty set of sort symbols;
- $\mathcal{V}_s$ $(s \in \mathcal{S})$ is a set of (propositional) variables;
- $\mathcal{C}_s$ $(s \in \mathcal{S})$ is a set of constants;
- $\mathcal{O}_s$ $(s \in \mathcal{S})$ is a set of connectives;
- $\mathcal{F}$ is a set of function symbols.

The elements of $\mathcal{F}$ are *modal operators*; via the semantics these will encode simple patterns in the structures in which the modal language is interpreted.

Each (propositional) variable and each constant is assumed to be equipped with a sort symbol as are the argument places of the modal operators and connectives. Further, connectives of sort s return formulas of sort s, and modal operators are assumed to be marked with the sort they return.

The formulas $Form_s$ of sort s ($s \in \mathcal{S}$) are built up as follows.

Connectives:	$\bullet$	Atomic formulas:	$p_s \in \mathcal{V}_s \cup \mathcal{C}_s$,
Modal operators:	$\#$,	Formulas:	$\phi \in Form_s$,

$$\phi ::= p_s \mid \bullet(\phi_{1,s}, \ldots, \phi_{n,s}) \mid \#(\phi_{s_1}, \ldots, \phi_{s_n}),$$

where it is assumed that $\bullet$ and $\#$ return values of sort s. (A side remark: according to the above set-up there is little difference between connectives and modal operators. But this is what we encounter in many systems of ML; Definition 2.2.5 and §2.4 contain further remarks on this issue.)

2.2.2. EXAMPLE. In the standard modal language we have two sorts: p (propositions) and r (relations), the usual set of propositional variables ($p_0, p_1, \ldots$) and constants ($\bot, \top$), and only one constant but no variables of the relational sort (R); the connectives are the usual Booleans, while there is only one modal operator, $\langle\cdot\rangle\cdot$, taking a first argument of the relational sort, a second argument of the propositional sort, and returning a formula of the propositional sort.

SEMANTICS

A system of modal logic not only specifies the syntax of legal formulas, it also provides a semantics to interpret these formulas. We need multiple domains, a uniform approach to dealing with the semantics of the modal operators, and a flexible way of incorporating side-conditions on the interpretations of the symbols in our language.

Generalizing our intuitions from the standard modal format, we interpret modal operators as describing simple patterns in the relational structures underlying our modal languages. Such patterns are given as formulas of a classical language $\mathcal{L}$. A classical logic is any logic in the sense of (Barwise & Feferman 1985, Chapter 2); often one can simply think of first-order logic when we write classical logic (cf. also the Appendix).

2.2.3. DEFINITION. Let $\mathcal{F}$ be as in 2.2.1, and let $\# \in \mathcal{F}$. A *pattern* or *$\mathcal{L}$-pattern* $\delta_\#$ for $\#$ is a formula in a classical logic $\mathcal{L}$ that specifies the semantics of $\#$. A pattern typically has the form $\lambda x_{s_1} \ldots \lambda x_{s_n} . \phi(x_{s_1}, \ldots, x_{s_n}; x_{s_{n+1}}; \ldots; x_{s_m})$, where $x_{s_1}, \ldots, x_{s_n}, x_{s_{n+1}}, \ldots, x_{s_m}$ are variables of sort $s_1, \ldots, s_n; s_{n+1}, \ldots, s_m$, respectively, the variables $x_{s_{n+1}}, \ldots, x_{s_m}$ are free variables, and all non-logical symbols occurring in ϕ are either among these variables or constants from $\mathcal{C}$.

The semantic structures for a system of ML consist of a number of domains W_s (for s a sort symbol in $\mathcal{S}$) with the modal operators describing configurations of elements of these domains, and formulas of sort s being interpreted as subsets of W_s. Formally, there is an interpretation function I assigning a subset of W_s to every formula in $Form_s$ as follows. First, there is a valuation V that assigns subsets of W_s to atomic symbols, that is, to elements of $\mathcal{V}_s \cup \mathcal{C}_s$. I will tacitly assume that the function I 'knows' how to deal with the elements in $\mathcal{O}_s$. Truly modal aspects pop up with formulas of the form $\#(\phi_1, \ldots, \phi_n)$, for $\# \in \mathcal{F}$: to compute the value of such a formula at a tuple $\vec{x}$ one checks whether the pattern for $\#$ applied to $\phi_1, \ldots, \phi_n$, with $\vec{x}$ assigned to its free variables is satisfied.

But this is not quite good enough. We may have to let our interpretation know that we have special things in mind for some of the symbols in our syntax.

2.2.4. EXAMPLE. Consider the standard modal language with its syntax as given in Example 2.2.2. The intended interpretation of the symbol R is not just $V(R) \subseteq W_r$, but rather $V(R) \subseteq (W_p)^2$: R is to be interpreted as a binary relation on W_p. With the standard modal format as in Example 2.2.2, and the constraint as above, the pattern for $\langle\cdot\rangle\cdot$ is $\lambda x_p.\,\exists y\,((x, y) \in V(R) \wedge y \in V(x_p))$.

Example 2.2.4 motivates the following. A *constraint on an interpretation* I is a formula expressing some meta-level condition on I, and on the way I interacts with the domains W_s. Familiar constraints include 'non-empty domain,' and 'persistence of proposition letters' in intuitionistic logic.

Summing up we arrive at the following definition of the semantics for an ML.

2.2.5. DEFINITION. Let $\tau = (\mathcal{S}, \mathcal{V}, \mathcal{C}, \mathcal{O}, \mathcal{F})$ be a modal vocabulary, and I an interpretation based on a valuation V and a set of patterns $\delta_{\#}$ ($\# \in \mathcal{F}$). A τ*-structure* is a tuple $\mathfrak{M} = (\{ W_s \}_{s \in \mathcal{S}}, I, \Gamma)$, where W_s is a non-empty domain corresponding to the sort s, and Γ is a set of constraints on I and satisfied by I. The value in $\mathfrak{M}$ of a formula ϕ (of sort s) at an element $x \in W_s$ is computed as follows:

$$\begin{aligned}
\mathfrak{M}, x \models p_s \quad &\text{iff} \quad x \in V(p_s),\\
\mathfrak{M}, x \models \bullet(\phi_1, \ldots, \phi_n) \quad &\text{iff} \quad x \in I(\bullet)(I(\phi_1), \ldots, I(\phi_n)),\\
\mathfrak{M}, x \models \#(\phi_1, \ldots, \phi_n) \quad &\text{iff} \quad \mathfrak{M}, x \models \delta_{\#}(I(\phi_1), \ldots, I(\phi_n)).
\end{aligned}$$

Just as there was little difference between modal operators and connectives at the syntactic level, it is hard to identify a difference at the semantic level. According to one opinion evaluating modal operators typically involves multiple states; but several modal operators involve just a single state (Gargov et al. 1987), and the evaluation of intuitionistic connectives in Kripke models involves multiple states. Another claim is that modal formulas typically involve (non-logical) relations between states when they are evaluated — but cf. the D-logic of Chapter 3, and the modal logics of (representable) relation algebras (Venema 1991); §2.4 below contains a further comment on this issue.

The upshot of Definitions 2.2.1 and 2.2.5 is that modal logic is a formalism with expressions of multiple sorts whose distinguishing feature is the presence of functions between sorts, and the use of restricted relational patterns as truth definitions to describe simple geometric aspects of the underlying structures.

At this point one may introduce *general* properties a system of ML should satisfy, as is done for classical logics in Abstract Model Theory. We will not do so; instead we refer the reader to Chapter 6 for more on this point.

MODAL LOGIC AS A DESCRIPTION LANGUAGE

Consider Definition 2.2.5 again. Let $\mathcal{ML}$ be a modal language. If for some classical language $\mathcal{L}$, all connectives and patterns of $\mathcal{ML}$ live in $\mathcal{L}$, then $\mathcal{ML}$ may be seen as a fragment of $\mathcal{L}$. Formally, this connection is given by a *standard translation* that takes $\mathcal{ML}$-formulas to $\mathcal{L}$-formulas by simply following the truth definition of $\mathcal{ML}$. For each sort s we fix sufficiently many individual variables x_s, and assume there are predicate symbols P_s corresponding to the propositional variables p_s. Then $ST(p_s) = P_s x_s$, ST commutes with connectives $\bullet$, and $ST(\#(\phi_1, \ldots, \phi_n)) = \delta_\#(ST(\phi_1), \ldots, ST(\phi_n))$, where some provisos have to be made to perform substitutions in $\delta_\#$. For $\mathcal{ML}(\Diamond)$ this instantiates to: $ST(p) = Px$, $ST(\neg\phi) = \neg ST(\phi)$, $ST(\phi \wedge \psi) = ST(\phi) \wedge ST(\psi)$, and $ST(\Diamond\phi) = \exists y\,(Rxy \wedge [y/x]ST(\phi))$, where $[y/x]\chi$ is χ with y substituted for x.

Hence, every modal language 'is' a restricted fragment of a classical language that precisely contains all descriptions of (combinations of) the geometric patterns in relational structures that underlie its modal operators. But then: which fragments of classical languages are modal fragments in the above sense? Chapter 6 answers this question for a variety of modal languages.

There is a second level at which modal languages are used as description languages. A *frame* $\mathfrak{F}$ is a τ-structure in which we 'forget' the valuation V; modal formulas are interpreted on frames by quantifying over all valuations: $\mathfrak{F}, x \models \phi$ iff for all models $\mathfrak{M} = (\mathfrak{F}, V)$ on $\mathfrak{F}$, $\mathfrak{M}, x \models \phi$. If $\mathcal{ML}$ and $\mathcal{L}$ are such that the standard translation maps $\mathcal{ML}$ into $\mathcal{L}$, then, on frames, $\mathcal{ML}$-formulas are equivalent to Π^1_1-conditions over $\mathcal{L}$. E.g. on frames the standard modal formula $p \to \Diamond p$ corresponds to $\forall P\,(P(x) \to \exists y\,(Rxy \wedge P(y)))$. Often such second-order conditions can be reduced to first-order ones; the above formula is equivalent to Rxx. Chapter 7 explores the mathematics of such reductions.

To conclude this section, the distinguishing aspects of modal languages are: multiple sorts, restricted relational truth definitions describing simple geometric patterns of models, and a concern with the fine-structure of model theory.

2.3 EXAMPLES

This section contains examples of modal logics fitting the framework of §2.2. Our examples differ from the standard modal format in having more operators, alternative modes of evaluation, more sorts, more structure, more or alternative constraints, or .... The examples all fit the framework of §2.2 rather easily, although we won't actually describe *how* they fit the framework — this is left to the reader. We first describe the standard modal format.

THE STANDARD MODAL FORMAT. The standard modal format has two sort symbols p (propositions) and r (relations). The variables of sort p are $\mathcal{V}_p = \{p_0, p_1, \ldots\}$; the constants in $\mathcal{C}_p$ are $\bot$, $\top$; the connectives $\mathcal{O}_p$ of sort p are $\neg$, $\wedge$. For the relational sort we have $\mathcal{V}_r = \emptyset$, $\mathcal{C}_r = \{R\}$, $\mathcal{O}_r = \emptyset$. $\mathcal{F} = \{\langle\cdot\rangle\cdot\}$, where the first argument place is marked for symbols of sort r, the second for symbols of sort p, and the result sort is p. The sole constraint is that $W_r \subseteq (W_p)^2$ The pattern for $\langle\cdot\rangle\cdot$ is $\lambda x_r . \lambda x_p . \exists y((x, y) \in V(x_r) \wedge y \in V(x_p))$. As there is only one possible input relation for this pattern we often use the 'old' notation $\Diamond\cdot$.

MORE OPERATORS. A common motivation for extending a modal formalism is the need to increase its expressive power. The most obvious way to go about things is to add extra operators; such additions can easily be accounted for in the framework of §2.2. One can simply add an extra operator to the existing stock, and give its pattern in terms of the material already present. One of the main characters of this dissertation is the *D-operator.* The pattern of this unary operator reads $\lambda x_p . \exists y\,(y \neq x \wedge y \in V(x_p))$. Adding it to $\mathcal{ML}(\Diamond)$ overcomes important deficiencies in expressive power; Chapters 3, 4 and 5 contain numerous examples to this effect. Alternatively, new operators and their patterns can be defined in terms of new relations. Provability logic, for example, where the dual $[\cdot]\cdot$ of $\langle\cdot\rangle\cdot$ simulates provability in an arithmetical theory, has been expanded with modal operators simulating (relative) interpretability whose pattern is based on an additional ternary relation, cf. (Berarducci 1990, De Rijke 1992*d*).

MORE COMPLEX MODES OF EVALUATION. In the above example formalisms one evaluates formulas by 'starting from a single state in a model, and looking for a configuration described by the connectives and modal operators.' There may be a need to switch to other modes of evaluation, involving more complex starting points for evaluation than just a single state. Concerns from linguistics, philosophy and computer science have led to the invention of modal formalisms in which evaluation takes place at *pairs* of points (Gabbay & Guenthner 1982), *intervals* (Van Benthem 1991*d*), and *computation paths* (Manna & Pnueli 1992).

MORE SORTS. In the above formalisms the relations whose properties are described by modal operators, are not visible at the syntactic level. *Propositional Dynamic Logic* (**PDL**) has explicit reference to relations by means of syntactic items occurring inside modal operators $\langle\alpha\rangle\cdot$, although these items aren't evaluated directly. A system of *Dynamic Modal Logic* (**DML**) that has been introduced by Van Benthem (1989*a*) is more explicitly *many-sorted*; it is analyzed in Chapter 4. **DML** too has no direct evaluation at relations; but a truly 2-sorted extension of **DML** arising from Peirce algebras is studied in Chapter 5.

MORE STRUCTURE, 1. A further mode of extending modal logic concerns the need to add structure to a sort already present, rather than add a new one. **PDL** provides an example. The big difference between $\mathcal{ML}(\Diamond)$ and **PDL** is not just that **PDL** has a larger stock of relation symbols, but that **PDL** provides means to add structure in the relational component. The standard version of **PDL** has composition and iteration of relations, but many more relational constructs have been studied, cf. (Harel 1984, Blackburn 1993*a*).

MORE STRUCTURE, 2. Instead of adding more structure *amongst* elements of a sort, it has also been proposed to add *internal* structure to elements of a sort. The point is this. In $\mathcal{ML}(\Diamond)$ we are interested in transitions between states and the way they affect truth values; we are *not* interested in the nature of these states. But in many applications it may be necessary or convenient to be more specific about their nature. Finger & Gabbay (1992) present a canonical way to amalgamate two modal languages $\mathcal{ML}_1$ and $\mathcal{ML}_2$ into one system $\mathcal{ML}_1(\mathcal{ML}_2)$ with a *global* component to reason about transitions between structured states, and a *local* one to handle their internal aspects.

CHANGING THE LOGIC WITHIN A SORT. One important mode of varying one of the components in a modal logic is changing the logic within a sort. Most proposals to this effect have been concerned with changing the logic within the propositional sort: from classical to partial (Thijsse 1992), many-valued (Fitting 1992), relevance (Fuhrmann 1990*a*), to intuitionistic (Božić & Došen 1984), and even infinitary logic.

RESTRICTING THE LANGUAGE. An adaptation of the standard modal format that also fits the framework of §2.2 easily is restricting the language. Examples of modal languages in which the *relational* repertoire is restricted for 'meta-purposes' are abundant. Because of computational concerns Van der Hoek & De Rijke (1992) restrict the propositional part of modal languages arising in AI. And in the Lambek Calculus the Booleans are left out, resulting in a decidable system with an associative operation of composition (Van Benthem 1991*a*).

ADDING AND CHANGING VARIABLES AND CONSTANTS. The final mode of altering the standard modal format that I mention here is restricting the admissible valuations on (some of) the variables. To increase the expressive power of $\mathcal{ML}(\diamond)$ Blackburn (1993*b*) discusses the addition of special variables whose value in a model should always be a singleton. In interval logic Venema (1990) restricts valuations to make his languages *less* expressive, as a result of which some meta-properties are regained (like admitting a finite axiomatization).

To prevent this section from becoming purely taxonomical, further ways of altering the standard modal format will be left out here. In particular, I will leave out some extensions that were inspired not by external, or application driven concerns, but by stimuli from within modal logic. But it should be clear that according to the framework of §2.2 a modal logic is a many-sorted formalism that is concerned with the fine-structure of models, one that has multiple components each of which may be varied, and which has functions going from sorts to sorts that describe simple relational patterns of the underlying structures.

2.4 QUESTIONS AND COMMENTS

With the examples behind us we address broader issues our framework for modal logic raises.

Q: Many modal logics do indeed count as modal logics according to the framework of §2.2. But isn't the framework so general that any formalism counts as a modal logic?

A: What is important is that the framework captures and clarifies our intuitions about ML, that it offers a unifying approach to ML, and that it pays off in terms of new insights and questions — I think it does.

Q: How does the framework of §2.2 relate to other general approaches to logic, like *Abstract Model-Theoretic Logic*?
A: Two important aspects of abstract model-theoretic logic are

1. the isolation and study of specific logics for the analysis of various (mathematical) properties, and
2. the investigation into the properties of and the relations between such logics.

Within a general framework for ML similar research issues become clearly visible. As a tool for reasoning about relational structures modal logic has long been occupied with the first aspect above — but at a more fine-structural level than abstract model theory: most modal systems have just a few simple patterns with low quantifier rank for their operators; there is an interest in finite variable fragments, and finite model property or low complexity results rather than in Löwenheim-Skolem numbers or definability in the analytic hierarchy.

The second aspect is only now gaining attention in modal logic; general analyses of properties of modal systems are now being developed, resulting, among others, in modal versions of theorems from abstract model theory: Chapter 6 below has a Lindström-style characterization of so-called basic modal languages.

ML is a part of abstract model-theoretic logic with a very special status; being many-sorted by nature and paying special attention to simple relational truth-definitions, it is concerned with the fine-structure of model theory.

Q: Which research topics arise naturally in the framework of §2.2?
A: Many questions arise. They can be divided somewhat roughly into four groups, most of them overlapping with the above items 1 and 2: *exploring* the modal universe, *classifying*, *generalizing* and *combining*.[2]

EXPLORING. Just as a large part of abstract model theory is devoted to studying relations between logics, so should modal logic. Here are two examples of the thing I have in mind. There is a wide gap in expressive power between ordinary tense logic with F and P as its modal operators, and *Until*, *Since*-logic. For one thing, the operators *Until*, *Since* are known to be irreducibly binary — are there elegant *unary* extensions of F, P-logic approximating *Until*, *Since*-logic in expressive power? Put more generally, how can this gap be filled?

Another example: according to general results in abstract model theory, a classical logic enjoys interpolation only if it has the Beth definability property — can similar general implications be found in modal logic?

[2]A side remark: of course one can pose the 'old questions' for every system of ML — whether it is formulated according to the scheme of §2.2. or not — like questions concerning completeness, expressive power, definability, decidability (or its refinement complexity) and truth preserving relations. But these old questions are not my prime concern here.

CLASSIFYING. In a large universe of objects insight is often gained through classification. In the present setting patterns underlying modal operators need classifying. Which are the 'nice' ones? Which yield 'nice' modal logics? Patterns may be organized according to their quantificational structure. Let a basic modal pattern be one of the form $\lambda R.\, \lambda\vec{p}.\, \exists\vec{y}\, (Rx\vec{y} \wedge \bigwedge_i p_i(y_i))$. Modal logics built on basic patterns admit a decent sequent-style axiomatization, are decidable, and enjoy interpolation (Van Benthem 1993); by using bisimulations their model theory may be developed in parallel with basic first-order model theory (Chapter 6 below). These results may fail for patterns with a more complex structure.

Modal patterns may also be classified according to their behavior with respect to relations between models. In addition to the proposals mentioned in Definition 2.2.5, this is another way of distinguishing modal operators from connectives: connectives (and their patterns) are not sensitive to the relational structure of models, while modal operators and their patterns are.

More generally, broad criteria for classifying modal operators and their patterns have yet to be invented, although certain case studies have been carried out.[3] One desideratum here is an analysis of the fine-structure of patterns that reveals how axiomatic and complexity theoretic aspects of the resulting ML are determined by the patterns.

GENERALIZING. How do the 'old' results from the standard modal logic generalize to novel systems of ML? If generalizations aren't possible, can natural counterexamples be given? This dissertation provides two examples of generalizations of results in the standard modal language. Chapter 7 presents a vast generalization of Sahlqvist's Correspondence Theorem. In its original form the result describes a class of $\mathcal{ML}(\Diamond)$-formulas that reduce to first-order formulas when interpreted on frames. This is generalized to arbitrary modal languages below, while examples beyond the generalized result are also given. In Chapter 6 I generalize and extend the model theory of the standard modal language $\mathcal{ML}(\Diamond)$ with bisimulations as its central tool, to a large class of modal languages.

The benefit of striving towards such generalizations may not just be achieving greater generality, but also gaining a better understanding of what made the 'old' result work in the first place.

COMBINING. This has to do with the general architecture of ML, and arises naturally in a setting with multiple components and various links between them. How do those aspects *combine*? How do the components influence each other? What kind of communication is there between them? How do properties of the parts *transfer* to the larger system? Some instances of this question are studied by Kracht & Wolter (1991), Fine & Schurz (1991), and Goranko & Passy (1992); they all consider the case of 'independently axiomatized' polymodal logics, and show how properties like completeness and the finite model property do or do not transfer. The results of (Finger & Gabbay 1992) mentioned in §2.3 provide a further example, as does the work of (Spaan 1993) on complexity of modal

[3] An example: Van Benthem (1991*a*) classifies functions between certain sorts with respect to their being a homomorphism or not.

logics. Blackburn & De Rijke (1993) develop a general perspective on combining logics in their *zooming in*, *zooming out* framework.

Another question here is: when can we do without some of the components of an ML? For example, when can we do away with some of the sorts? The relational sort may in some sense be eliminated from the basic modal format (Brink 1981), but we know from universal algebra that in general this is impossible .

Q: What about proof theory? Your framework does not mention it at all.
A: Yes, although the position of this Chapter has its roots in the Amsterdam school of modal logic which has emphasized the semantic aspects of the enterprise, I feel that a framework for modal logic should also address the issue of proof theory. The proof theory of ML has not kept pace with its model theory, mainly due to the fact that innovating ideas in ML often arise from its semantic use, where proof theory is not the most obvious research topic.

Should one demand that an ML have a complete proof procedure? In order to answer this, one should keep in mind the role ML is supposed to play. As many systems of ML are designed with applications in mind, a complete proof theory seems desirable. But completeness is not just another property a system might have or not have. It may be that completeness is too stringent a requirement; even when using ML as a deductive machine completeness might be sacrificed for other advantages, such as greater expressive power. And even when one does strive for and obtain completeness results in ML a lot still remains to be done. Sequent calculi seem to be needed for most practical purposes, and in this respect the approach advocated by Wansing (1993) seems promising, as his sequent style proof theory seems to allow generalizations to arbitrary modal languages.

2.5 Concluding remarks

At the risk of overdoing it, let me repeat the picture of modal logic presented here once more. Several influential presentations of modal logic have had a rather 'metaphysical' flavour, obscuring the real use and power of modal languages by paying (too) much attention to notions like 'possible world' and 'transworld identity.' A modal language is a restricted description language for relational structures; it is a many-sorted formalism in which the modal operators emerge as functions from sorts to sorts that describe simple, restricted patterns in the underlying relational structures. Thus, the main concern of modal logic is best described as 'the fine-structure of many-sorted model theory.'

Although the examples of 'truly' many-sorted systems of modal logic given in this Chapter are still quite traditional, I think that we will see the development of lots of many-sorted modal formalisms in the near future, especially with the growing interest in richer, mixed calculi that allow for more subtle notions of transition than just sets of ordered pairs, and in which multiple cognitive activities can be adequately represented alongside programs or actions. Further examples of multi-sorted modal formalisms are bound to arise in areas where one works with different kinds of information, some of which may influence others.

Finally, the global perspective outlined here may not be the best approach to actually *work* in or with systems of modal logic. But the ideology presented here is meant only as a framework and unifying approach to modal logic in which important research lines can be identified — not as a technical tool.

Part II

Three Case Studies

3

The Modal Logic of Inequality

3.1 Introduction

Basic (propositional) modal and temporal logic cannot define all the natural assumptions one would like to make on the accessibility relation. An obvious move to try and overcome this lack of expressive power, is to extend the languages of modal and temporal logic with new operators. One particular such extension consists of adding the operator D mentioned earlier whose semantics is based on the relation of inequality. Segerberg (1976) appears to be the first to have studied the D-operator. Some 10 years later the operator was re-invented independently by Sain (1988), Gargov, Passy & Tinchev (1987) and Koymans (1992). Extending the standard modal languages with the D-operator is of interest for various reasons. First, it shows that the most striking deficiencies in expressive power of the standard modal and temporal languages may be removed with relatively simple means. Second, several recent proposals to enhance the expressive power of the standard language naturally give rise to considering the D-operator; thus the language with the operators $\Diamond$ and D appears as a kind of fixed point amongst the wide range of recently introduced extensions of the standard language, including logics analyzed in Chapters 4 and 5 below. And third, many of the interesting logical phenomena that one encounters in the study of enriched modal languages are illustrated by this particular extension.

Applications of the D-operator can be found in (Gargov & Goranko 1991), where it has been used in the study of various enriched modal languages, and in (Koymans 1992), where it is applied in the specification of message passing and time-critical systems. Sain (1988) uses the D-operator in a comparative study of program verification methods.

The main topic of this Chapter is the modal language $\mathcal{ML}(\Diamond, D)$ whose modal operators are $\Diamond$ and D. The remainder of §3.1 introduces the basic notions, and examines which of the (anti-) preservation results known from standard modal logic remain valid in the extended formalism. Next, §3.2 compares the expressive power of modal languages that contain the D-operator to that of other modal languages. In §3.3 we present the basic logics in some languages with the D-operator, and we give complete axiomatizations for several special structures; in it we prove that, given the right basic logic in $\mathcal{ML}(D)$, all finite extensions thereof are complete; we also discuss complexity and decidability issues as well as analogues of the Sahlqvist Theorem for $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$. §3.4

then deals with definability — both of classes of frames and of classes of models. §3.5 completes the Chapter with general comments and open questions.

Basics

The modal languages we consider in this Chapter usually have two sorts: propositions and relations. We have an infinite supply $\Phi = \{ p_0, p_1, \dots \}$ of proposition letters. Our modal operator $\langle\cdot\rangle\cdot$ takes a relation and a proposition, and returns a proposition; its pattern reads $\lambda R.\, \lambda p.\, \exists y(Rxy \wedge Py)$.

In the standard modal language $\mathcal{ML}(\Diamond)$ we only consider one symbol R of the relational sort, and write $\Diamond$ for $\langle R\rangle\cdot$; $\Box$ is short for $\neg\Diamond\neg$. $\mathcal{ML}(D)$ has $\neq$ as its relation symbol; we write D for $\langle\neq\rangle\cdot$, and $\overline{D}$ for $\neg D\neg$. In temporal languages we have both R and $R^{\smile}$, the converse of R. In such languages we write F for $\Diamond$, G for $\Box$, and P for $\langle R^{\smile}\rangle\cdot$, H for $\neg P\neg$. In languages extending $\mathcal{ML}(D)$ $E\phi$ abbreviates $(\phi \vee D\phi)$ (there *exists* a point at which ϕ holds); $A\phi$ is short for $\neg E\neg\phi$ (ϕ holds at *all* points; $O\phi$ is short for $(\phi \wedge \neg D\phi)$ (ϕ holds at *only* here). Rather than using the general notation of Chapter 2, and carrying along redundant information, we write $\mathcal{ML}(O_1, \dots, O_n)$ for the modal language with operators $O_1, \dots, O_n$.

Following the general set-up of Chapter 2, the models for our languages are tuples (W, V) or (W, R, V), depending on whether R is among the relational elements or not. From these, frames are defined in the usual way by leaving out the valuation. We assume that $\mathfrak{M}$ denotes the model $(\mathfrak{F}, V)$. It may be worthwhile instantiating the general truth definition (Definition 2.2.5) for the modal operators of this Chapter:

- $\mathfrak{M}, w \models \Diamond\phi$ iff for some $v \in W$, Rwv and $\mathfrak{M}, v \models \phi$,
- $\mathfrak{M}, w \models D\phi$ iff for some $v \in W$, $v \neq w$ and $\mathfrak{M}, v \models \phi$,
- $\mathfrak{M}, w \models F\phi$ iff for some $v \in W$, Rwv and $\mathfrak{M}, v \models \phi$,
- $\mathfrak{M}, w \models P\phi$ iff for some $v \in W$, Rvw and $\mathfrak{M}, v \models \phi$.

Notions like '$\mathfrak{M} \models \phi$,' '$\mathfrak{F}, w \models \phi$,' and '$\mathfrak{F} \models \phi$' are defined by universally quantifying over the missing component(s); however, '$x \models \phi$' will simply mean $\mathfrak{M}, x \models \phi$, where $\mathfrak{M}$ is provided by the context.

The fact that some notions are sensitive to the language we are working with, is reflected in our notation: e.g. we write $\mathfrak{F} \equiv_{\Diamond,D} \mathfrak{G}$ for $\mathfrak{F}$ and $\mathfrak{G}$ validate the same $\phi \in \mathcal{ML}(\Diamond, D)$, and $\mathrm{Th}_{\Diamond,D}(\mathfrak{F})$ for the set of formulas in $\mathcal{ML}(\Diamond, D)$ that are valid on $\mathfrak{F}$.

We sometimes refer to the first-order languages $\mathcal{L}(\boldsymbol{\tau}_0)$ and $\mathcal{L}(\boldsymbol{\tau}_1)$, where $\boldsymbol{\tau}_0$ has one binary predicate symbol R as well as identity; in addition $\boldsymbol{\tau}_1 \supseteq \boldsymbol{\tau}_0$ has unary predicate symbols P_0, $P_1, \dots$ corresponding to the proposition letters in Φ. First-order formulas will be denoted by α, β, $\gamma, \dots$. First-order truth of a formula $\alpha(x)$ in one free variable is written as $\mathfrak{M} \models \alpha[w]$, or $\mathfrak{F} \models \alpha[w]$. We call α (*locally*) *definable* in $\mathcal{ML}(O_1, \dots, O_n)$ if for some $\phi \in \mathcal{ML}(O_1, \dots, O_n)$, for all $\mathfrak{F}$, and all $w \in W$, $\mathfrak{F} \models \alpha[w]$ iff $\mathfrak{F}, w \models \phi$; it is called *globally definable* in $\mathcal{ML}(O_1, \dots, O_n)$ if for some $\phi \in \mathcal{ML}(O_1, \dots, O_n)$, for all $\mathfrak{F}$, $\mathfrak{F} \models \alpha$ iff $\mathfrak{F} \models \phi$.

(ANTI-) PRESERVATION AND FILTRATIONS

It is well-known that standard modal formulas are preserved under surjective p-morphisms, disjoint unions and generated subframes:

3.1.1. DEFINITION. 1. A function f from a frame $\mathfrak{F}_1$ to a frame $\mathfrak{F}_2$ is called a *p-morphism* if (i) for all $w, v \in W_1$, if R_1wv then $R_2f(w)f(v)$; and (ii) for all $w \in W_1$, $v \in W_2$, if $R_2f(w)v$ then there is a $u \in W_1$ such that R_1wu and $f(u) = v$.

2. $\mathfrak{F}_1$ is called a *generated subframe* of a frame $\mathfrak{F}_2$ if (i) $W_1 \subseteq W_2$; (ii) $R_1 = R_2 \cap (W_1 \times W_1)$ (the restriction of R_2 to W_1); and (iii) for all $w \in W_1$, $v \in W_2$, if R_2wv then $v \in W_1$.
3. Let $\mathfrak{F}_i$ $(i \in I)$ be a collection of disjoint frames. Then the *disjoint union* $\biguplus_{i\in I} \mathfrak{F}_i$ is the frame $(\bigcup\{ W_i : i \in I \}, \bigcup\{ R_i : i \in I \})$.

Here are some examples showing that adding the D-operator to $\mathcal{ML}(\Diamond)$ gives an increase in expressive power over standard modal logic:

1. $\Diamond p \to Dp$ defines $\forall x \neg Rxx$, but the latter is not definable in $\mathcal{ML}(\Diamond)$;
2. $\Diamond\top \vee D\Diamond\top$ defines $R \neq \emptyset$, and, again, the latter is not definable in $\mathcal{ML}(\Diamond)$;
3. $p \vee Dp \to \Diamond p$ defines $R = W^2$, but the latter is not definable in $\mathcal{ML}(\Diamond)$.

We prove the first of these claims, leaving the others to the reader. The proof involves two implications: if $\mathfrak{F} \models \forall x \neg Rxx$ then $\mathfrak{F} \models \Diamond p \to Dp$, and conversely. Assume $\mathfrak{F} \models \forall x \neg Rxx$, and $(\mathfrak{F}, V), w \models \Diamond p$. Then there exists w with Rwv and $(\mathfrak{F}, V), v \models p$; as R is irreflexive $w \neq v$, hence $(\mathfrak{F}, V), w \models Dp$. Conversely, if $\mathfrak{F} \not\models \forall x \neg Rxx$, there exists w in $\mathfrak{F}$ with Rww; putting $V(p) = \{ w \}$ defines a valuation that refutes $\Diamond p \to Dp$ at w.

Using the above preservation results it is easily verified that none of the above three conditions is definable in $\mathcal{ML}(\Diamond)$. And conversely, the fact that they are definable in $\mathcal{ML}(\Diamond, D)$ implies that we no longer have these preservation results in $\mathcal{ML}(\Diamond, D)$. Moreover, they can be restored only at the cost of trivializing the constructions concerned.

A fourth important construction in standard modal logic is the following:

3.1.2. DEFINITION. Let $\mathfrak{F}$ be a frame, and $X \subseteq W$. Then $L_R(X) = \{ w \in W : \forall v \in W\, (Rwv \to v \in X) \}$. The *ultrafilter extension* $ue(\mathfrak{F})$ is the frame $(W_{\mathfrak{F}}, R_{\mathfrak{F}})$, where $W_{\mathfrak{F}}$ is the set of ultrafilters on W, and $R_{\mathfrak{F}}U_1U_2$ holds if for all $X \subseteq W$, $L_R(X) \in U_1$ implies $X \in U_2$.

Standard modal formulas are *anti-preserved* under ultrafilter extension, i.e. if $ue(\mathfrak{F}) \models \phi$ then $\mathfrak{F} \models \phi$, cf. Van Benthem (1983, Lemma 2.25). Perhaps surprisingly, for formulas $\phi \in \mathcal{ML}(\Diamond, D)$ this results still holds good — as one easily deduces from the following result.

3.1.3. PROPOSITION. *Let V be a valuation on $\mathfrak{F}$. Define the valuation $V_{\mathfrak{F}}$ on $ue(\mathfrak{F})$ by putting $V_{\mathfrak{F}}(p) = \{ U : V(p) \in U \}$. Then, for all ultrafilters U on W, and all formulas $\phi \in \mathcal{ML}(\Diamond, D)$ we have $(ue(\mathfrak{F}), V_{\mathfrak{F}}), U \models \phi$ iff $V(\phi) \in U$.*

Proof. This is by induction on ϕ. The cases $\phi \equiv p$, $\neg\psi$, $\psi \wedge \chi$, $\Diamond\psi$ are proved by Van Benthem (1983, Lemma 2.25). The only new case is $\phi \equiv D\psi$. Suppose $V(D\psi) = \{\, w : \exists v \neq w\, (v \in V(\psi)) \,\} \in U$. We must find an ultrafilter $U_1 \neq U$ such that $(ue(\mathfrak{F}),\ V_{\mathfrak{F}}),\ U_1 \models \psi$. First assume that U contains a singleton — say, $U = \{\, X \subseteq W : X \supseteq \{\, w_0 \,\} \,\}$. Then $w_0 \in V(D\psi)$, so there exists a $v \neq w_0$ with $v \in V(\psi)$. Since $v \neq w_0$, we must have $\{\, v \,\} \notin U$. Let U_1 be the ultrafilter generated by $\{\, v \,\}$; then $U \neq U_1$. Furthermore, $v \in V(\psi)$ implies $V(\psi) \in U_1$, and hence $(ue(\mathfrak{F}),\ V_{\mathfrak{F}}),\ U_1 \models \psi$, by the induction hypothesis. It follows that $(ue(\mathfrak{F}),\ V_{\mathfrak{F}}),\ U \models D\psi$. Next, suppose that U does not contain a singleton. Since $V(D\psi) \in U$, we find some $w_0 \in V(D\psi)$. Let v be a point such that $v \neq w_0$ and $v \in V(\psi)$. Then $\{\, v \,\} \notin U$ — and we can proceed as in the previous case.

Conversely, assume $V(D\psi) \notin U$. We have to show that $(ue(\mathfrak{F}),\ V_{\mathfrak{F}}),\ U \not\models D\psi$. Since $V(D\psi) \notin U$, we have that $X = \{\, w : \forall v\, (v \neq w \rightarrow v \notin V(\psi)) \,\} \in U$, and hence $X \neq \emptyset$. Let $w_0 \in X$. Clearly, if $w_0 \notin V(\psi)$, then $X = W$ and $V(\psi) = \emptyset$. Consequently, for all ultrafilters $U_1 \neq U$ we have $V(\psi) \notin U_1$. So, by the induction hypothesis, $(ue(\mathfrak{F}),\ V_{\mathfrak{F}}),\ U_1 \not\models \psi$, and hence $(ue(\mathfrak{F}),\ V_{\mathfrak{F}}),\ U \not\models D\psi$ — as required. If, on the other hand $w_0 \in V(\psi)$, then $X = \{\, w_0 \,\} = V(\psi)$, and U is generated by X. It follows that for any ultrafilter $U_1 \neq U$, $X = V(\psi) \notin U_1$. So by the induction hypothesis $(ue(\mathfrak{F}),\ V_{\mathfrak{F}}),\ U_1 \not\models \psi$, for such U_1. This implies $(ue(\mathfrak{F}),\ V_{\mathfrak{F}}),\ U \not\models D\psi$. $\dashv$

3.1.4. THEOREM. *For any frame $\mathfrak{F}$ and all $\phi \in \mathcal{ML}(\Diamond, D)$, if $ue(\mathfrak{F}) \models \phi$ then $\mathfrak{F} \models \phi$.*

3.1.5. COROLLARY. *$\exists x\, Rxx$ is not definable in $\mathcal{ML}(\Diamond, D)$.*

Proof. Evidently, $\mathfrak{F} = (\mathbb{N}, <) \not\models \exists x\, Rxx$. By an elementary argument any non-principal ultrafilter U on $\mathbb{N}$ has $R_{\mathfrak{F}}UU$. Hence, $ue(\mathfrak{F}) \models \exists x\, Rxx$. Now apply 3.1.4. $\dashv$

Another important notion from standard modal logic is that of a *filtration*. It has a straightforward adaptation to $\mathcal{ML}(\Diamond, D)$:

3.1.6. DEFINITION. Let $\mathfrak{M}_1$, $\mathfrak{M}_2$ be models, and let Σ be a set of formulas $\phi \in \mathcal{ML}(\Diamond, D)$ closed under subformulas and single negations. A surjective function $g : \mathfrak{M}_1 \rightarrow \mathfrak{M}_2$ is an *extended filtration with respect to* Σ, if

1. for all w, $v \in W_1$, if R_1wv then $R_2g(w)g(v)$,
2. for all $w \in W_1$, and all proposition letters p in Σ, $w \in V_1(p)$ iff $g(w) \in V_2(p)$,
3. for all $w \in W_1$, and all $\Delta\phi \in \Sigma$, if $\mathfrak{M}_2, g(w) \models \Delta\phi$ then $\mathfrak{M}_1, w \models \Delta\phi$, where $\Delta \in \{\, \Diamond, D \,\}$, and conversely for $\Delta \equiv D$.

In $\mathcal{ML}(F, P, D)$ we replace the clause for $\Diamond$ by two similar clauses for F and P.

3.1.7. PROPOSITION. *If g is an extended filtration w.r.t. Σ from $\mathfrak{M}_1$ to $\mathfrak{M}_2$, then for all $w \in W_1$, and all $\phi \in \Sigma$, $\mathfrak{M}_1, w \models \phi$ iff $\mathfrak{M}_2, g(w) \models \phi$.*

Recall that the standard example of a filtration in ordinary modal logic is the *modal collapse*: given a model $\mathfrak{M}$ and a set Σ that is closed under subformulas, it is defined as the model $\mathfrak{M}'$, where for $g(w) = \{\phi \in \Sigma : \mathfrak{M}, w \models \phi\}$, $W' = g[W]$, $R'g(w)g(v)$ holds iff for all $\Box\phi \in \Sigma$, $\Box\phi \in g(w)$ implies $\phi \in g(v)$, and $V'(p) = \{g(w) : p \in g(w)\}$. To obtain an analogue of the modal collapse for $\mathcal{ML}(\Diamond, D)$, take the ordinary modal collapse and double points that correspond to more than one point in the original model. A simple inductive proof then shows that corresponding (doubled) points verify the same formulas. Likewise, for $\mathcal{ML}(F, P, D)$ we take $R'g(w)g(v)$ iff for all $G\phi \in \Sigma$, $G\phi \in g(w)$ implies $\phi \in g(v)$ *and* for all $H\phi \in \Sigma$, $H\phi \in g(v)$ implies $\phi \in g(w)$.

Using the extended collapse one may show in a standard way that formulas $\phi \in \mathcal{ML}(\Diamond, D)$ satisfy the finite model property (and, hence, that the validities in $\mathcal{ML}(\Diamond, D)$ form a recursive set): such a $\phi \in \mathcal{ML}(\Diamond, D)$ has a model iff it has a model with at most $2 \cdot 2^n$ elements, where n is the length of ϕ. Further decidability and complexity issues concerning the D-operator are contained in §3.3 below.

3.2 Some comparisons

In this section we compare modal languages containing the D-operator to some languages without it. It is not our aim to give a complete description of all aspects in which languages of the former kind differ from, or are the same as, languages of the latter kind, but merely to highlight some of the features of the former languages.

The language $\mathcal{ML}(D)$

The general standard translation ST defined in §2.2 specializes to $\mathcal{ML}(D)$, $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$ in the following way. Let x be a fixed variable.

$$\begin{aligned}
ST(p) &= Px \\
ST(\neg\psi) &= \neg ST(\psi) \\
ST(\phi \wedge \psi) &= ST(\phi) \wedge ST(\psi) \\
ST(\Diamond\psi) = ST(F\psi) &= \exists y\,(Rxy \wedge [y/x]ST(\psi)) \\
ST(P\psi) &= \exists y\,(Ryx \wedge [y/x]ST(\psi)) \\
ST(D\psi) &= \exists y\,(x \neq y \wedge [y/x]ST(\psi)),
\end{aligned}$$

where $[y/x]\chi$ denotes the result of substituting y for all free occurrences of x in χ, and y is a variable not occurring free in $ST(\psi)$. Observe that ST maps our modal languages into fragments of first-order logic that need contain only *two* variables. For example, we can translate $\Diamond Dp$ as $\exists y\,(Rxy \wedge \exists x\,(y \neq x \wedge Px))$. Also, we have $\mathfrak{M}, w \models \phi$ iff $\mathfrak{M} \models ST(\phi)[w]$, and $\mathfrak{F}, w \models \phi$ iff $\mathfrak{F} \models \forall P_0 \ldots P_n\, ST(\phi)[w]$, where $P_0, \ldots, P_n$ correspond to the proposition letters in ϕ.

3.2.1. Proposition. *On frames all formulas $\phi \in \mathcal{ML}(D)$ define first-order conditions.*

Proof. Using the ST-translation such formulas can be translated into equivalent second-order formulas containing only monadic predicate variables and $=$. By a result in (Ackermann 1954, Chapter IV) these formulas are equivalent to first-order ones; a proof of the Ackermann result may be found in §7.7. $\dashv$

Proposition 3.2.1 marks a considerable difference with $\mathcal{ML}(\Diamond)$: as is well-known, on frames not all $\mathcal{ML}(\Diamond)$-formulas express first-order conditions. In the opposite direction, there are also some natural conditions undefinable in $\mathcal{ML}(\Diamond)$ that are definable in $\mathcal{ML}(D)$. For example, using the preservation of standard modal formulas under generated subframes and disjoint unions, it is easily verified that no finite cardinality is definable in $\mathcal{ML}(\Diamond)$; on the other hand, although 3.2.1 implies that 'infinity' is not definable in $\mathcal{ML}(D)$, we do have

3.2.2. PROPOSITION. (Koymans) *On frames all finite cardinalities are definable in $\mathcal{ML}(D)$.*

Proof. For $n \in \mathbb{N}$, $|W| \leq n$ is defined by $\bigwedge_{1\leq i\leq n+1} EOp_i \longrightarrow \bigvee_{1\leq i<j\leq n+1} E(p_i \wedge p_j)$, while $|W| > n$ is defined by $A(\bigvee_{1\leq i\leq n} p_i) \longrightarrow E\bigvee_{1\leq i\leq n}(p_i \wedge \bar{D}p_i)$. $\dashv$

3.2.3. THEOREM. (Expressive Completeness) *On frames $\mathcal{ML}(D)$ is equivalent with the language of first-order logic over $=$.*

Proof. All first-order formulas over $=$ are equivalent to Boolean combinations of formulas of the form 'there exist at least n objects' (Chang & Keisler 1973, Corollary 1.5.8). By 3.2.2 these formulas are definable in $\mathcal{ML}(D)$. The converse follows from 3.2.1. $\dashv$

THE LANGUAGES $\mathcal{ML}(\Diamond, D)$ AND $\mathcal{ML}(\Diamond)$

One way to compare the expressive powers of two languages is to examine their ability to discriminate between special (read: well-known) structures. For example, unlike $\mathcal{ML}(\Diamond)$, $\mathcal{ML}(\Diamond, D)$ can distinguish between $\mathbb{Z}$ and $\mathbb{N}$: $(\mathbb{N}, <) \not\equiv_{\Diamond,D} (\mathbb{Z}, <)$. This follows from the fact that the existence of a (different) predecessor is expressible in $\mathcal{ML}(\Diamond, D)$ by means of the formula $p \rightarrow D\Diamond p$.

So $\forall x \exists y\, (x \neq y \wedge Ryx)$ is an $\mathcal{L}(\tau_0)$-condition on frames which is definable in $\mathcal{ML}(\Diamond, D)$, but not in $\mathcal{ML}(\Diamond)$. Other well-known $\mathcal{L}(\tau_0)$-conditions undefinable in $\mathcal{ML}(\Diamond)$ are irreflexivity and anti-symmetry. By the next result, these conditions do have an $\mathcal{ML}(\Diamond, D)$-equivalent:

3.2.4. PROPOSITION. *All Π^1_1-sentences in R, $=$ of the purely universal form*

$$\forall P_1 \ldots \forall P_m \forall x_1 \ldots \forall x_n\, BOOL(P_i x_j, Rx_i x_j, x_i = x_j)$$

are definable in $\mathcal{ML}(\Diamond, D)$.

Proof. Let $p_1, \ldots, p_m$, $q_1, \ldots, q_n$ be proposition letters such that each of $p_1, \ldots, p_m$ is different from each of $q_1, \ldots, q_n$. Now take

$$EOq_1 \wedge \ldots \wedge EOq_n \rightarrow BOOL\Big(E(q_i \wedge p_j), E(q_i \wedge \Diamond q_j), E(q_i \wedge q_j)\Big). \quad \dashv$$

It is well-known that two finite, rooted frames that validate the same formulas $\phi \in \mathcal{ML}(\Diamond)$, are isomorphic. This is improved upon in $\mathcal{ML}(\Diamond, D)$:

3.2.5. COROLLARY. *If $\mathfrak{F}_1$ and $\mathfrak{F}_2$ are finite frames, then $\mathfrak{F}_1 \equiv_{\Diamond,D} \mathfrak{F}_2$ iff $\mathfrak{F}_1 \cong \mathfrak{F}_2$.*

Proof. Finite frames are isomorphic iff they have the same universal first-order theory. So from 3.2.4 the result follows. Alternatively, one may give, for each finite frame $\mathfrak{F}$, a 'characteristic formula' $\chi_{\mathfrak{F}}$ such that for all $\mathfrak{G}$, $\mathfrak{G} \not\models \neg\chi_{\mathfrak{F}}$ iff $\mathfrak{G} \cong \mathfrak{F}$ (cf. §3.4). $\dashv$

Let us call a set T of (multi-) modal formulas *(frame) categorical* if, up to isomorphism, there is only one frame validating T; T is λ*-categorical* if, up to isomorphism, T has only one frame of power λ validating it. (λ-) categoricity is an important notion in first-order logic that is meaningless in standard modal languages: by some elementary manipulations one easily establishes that if $\mathfrak{F} \models T$ for some $\mathfrak{F}$, where T is a theory in either $\mathcal{ML}(\Diamond)$ or $\mathcal{ML}(F, P)$, and if I is a set of indices, then for each $i \in I$ there is a frame $\mathfrak{F}_i \models T$ such that $\mathfrak{F}_i \not\cong \mathfrak{F}_j$ if $i \neq j$. In contrast, for any *finite* frame $\mathfrak{F}$ the complete $\Diamond, D$-theory $\mathrm{Th}_{\Diamond,D}(\mathfrak{F})$ is easily seen to be categorical by 3.2.4.

The classical example of an ω-categorical theory in first-order logic is the complete theory of the rationals. By standard techniques one can show that $\mathrm{Th}_{\Diamond}(\mathbb{Q})$ is not ω-categorical; but $\mathrm{Th}_{\Diamond,D}(\mathbb{Q})$ *is* ω-categorical:

3.2.6. PROPOSITION. *The complete $\Diamond, D$-theory of $\mathbb{Q}$ is ω-categorical.*

Proof. It suffices to give formulas $\phi \in \mathcal{ML}(\Diamond, D)$ which are equivalent to the axioms for the theory of dense linear order without endpoints:

$$\begin{array}{ll}
\forall xyz\,(x < y \wedge y < z \rightarrow x < z) & \Diamond\Diamond p \rightarrow \Diamond p \\
\forall xy\,(x < y \wedge y < x \rightarrow x = y) & p \wedge \neg Dp \rightarrow \Box(\Diamond p \rightarrow p) \\
\forall x\,\neg(x < x) & \Diamond p \rightarrow Dp \\
\forall xy\,(x = y \vee x < y \vee y < x) & p \rightarrow \Diamond q \vee \overline{D}(q \rightarrow \Diamond p) \\
\forall xy\exists z\,(x < y \rightarrow x < z \wedge z < y) & \Diamond p \rightarrow \Diamond\Diamond p \\
\exists xy\,(x \neq y) & D\top \\
\forall x\exists y\,(x < y) & \Diamond\top \\
\forall x\exists y\,(y < x) & p \rightarrow E\Diamond p. \ \dashv
\end{array}$$

The special form of the antecedent of the second $\Diamond, D$-formula in the above list is worth noting. When such an antecedent is true it forces p to act as a so-called *nominal*, i.e., to hold at precisely one point; this then enables us to describe the behavior of $<$ locally, at the unique point at which p holds. (See below for more on nominals.)

Recall that a *modal sequent* is a pair $\sigma = (\Gamma_0, \Theta_0)$ where Γ_0 and Θ_0 are finite sets of modal formulas; $\mathfrak{F} \models \sigma$ if for every V, if $(\mathfrak{F}, V) \models \Gamma_0$ then there is a $\theta \in \Theta_0$ with $(\mathfrak{F}, V) \models \theta$. A class K of frames is *sequentially definable* if there is a set L of modal sequents such that $\mathsf{K} = \{\, \mathfrak{F} : \forall\sigma \in L\,(\mathfrak{F} \models \sigma)\,\}$. Kapron (1987) shows that in $\mathcal{ML}(\Diamond)$ sequential definability is strictly stronger than ordinary definability. By our remarks in §3.1 and the fact that validity of sequents in

$\mathcal{ML}(\Diamond)$ is preserved under p-morphisms (Kapron 1987), it follows that definability in $\mathcal{ML}(\Diamond, D)$ is still stronger. Furthermore, in $\mathcal{ML}(\Diamond, D)$ the notions of ordinary and sequential definability coincide; as is pointed out by Goranko & Passy (1992) this is due to the fact that we can define the 'universal modality' A in $\mathcal{ML}(\Diamond, D)$:

3.2.7. PROPOSITION. *Let* K *be a class of frames.* K *is sequentially definable in* $\mathcal{ML}(\Diamond, D)$ *iff it is definable in* $\mathcal{ML}(\Diamond, D)$.

Proof. One direction is clear. To prove the other one, assume that K is defined by a set L of sequents. For each $\sigma = (\{\phi_0, \ldots, \phi_n\}, \{\psi_0, \ldots, \psi_m\}) \in L$ put $\sigma^* := \bigwedge_{0\leq i\leq n} A\phi_i \to \bigvee_{0\leq i\leq m} A\psi_i$. Then K is defined by $\{\sigma^* : \sigma \in L\}$. ⊣

It should be clear by now that adding the D-operator to $\mathcal{ML}(\Diamond)$ greatly increases the expressive power. Limitations are easily found, however. As we have seen, $\exists x\, Rxx$ is still not definable in $\mathcal{ML}(\Diamond, D)$. And, as with the standard modal language we find that on well-orders a sort of 'stabilization of discriminatory power' occurs at a relatively early stage; cf. (Van Benthem 1989*b*) for a proof of this result for the standard modal language. To prove this, we recall that the *clusters* of a transitive frame $\mathfrak{F} = (W, R)$ are the equivalence classes of W under the relation $x \sim y$ iff $(Rxy \wedge Ryx) \vee x = y$. Clusters are divided into three kinds: *proper*, with at least two elements, all reflexive; *simple*, with one reflexive element; and *degenerate*, with one irreflexive element.

3.2.8. THEOREM. *If* $\phi \in \mathcal{ML}(\Diamond, D)$, *and* $\mathfrak{F}$ *is a well-ordered frame with* $\mathfrak{F} \not\models \phi$, *then there is a well-ordered frame* $\mathfrak{G}$ *such that* $\mathfrak{G} < \omega^2$ *and* $\mathfrak{G} \not\models \phi$.

Proof. Suppose that for some V, $w \in W$, $\mathfrak{M} = (\mathfrak{F}, V), w \models \neg\phi$. Let Σ^- be the set of subformulas of $\neg\phi$, and define $\Sigma := \Sigma^- \cup \{\Diamond\psi : D\psi \in \Sigma^-\}$. Let $\mathfrak{M}_1$ be the (extended) collapse of $\mathfrak{M}$ w.r.t. Σ. Then $\mathfrak{M}_1$ is transitive and linear. Consequently, $\mathfrak{M}_1$ may be viewed as a finite linear sequence of clusters.

Next, $\mathfrak{M}_1$ will be blown up into a well-ordered model $\mathfrak{M}_2$ by replacing each cluster with an appropriate well-order. If $C = \{w\}$ is a degenerate cluster, then C is itself a well-order, and we do nothing. Non-degenerated clusters $\{w_1, \ldots, w_k\}$ are replaced with a copy of ω; the valuation is adapted by verifying p in a newly added n iff $n = i \bmod k$ and $w_i \in V_1(p)$. The resulting model is a well-order, and since $\mathfrak{M}_1$ is finite it will have order type $< \omega^2$.

If $w \in W_1$, we write $\overline{w}$ for (a) point(s) corresponding to w in $\mathfrak{M}_2$. Then, for all $\psi \in \Sigma$, and $w \in W_1$, $\mathfrak{M}_1, w \models \psi$ iff $\mathfrak{M}_2, \overline{w} \models \psi$. This equivalence is proved by induction on ψ. The only non-trivial case is when $\psi \equiv D\chi$, and $\mathfrak{M}_2, \overline{w} \models D\chi$. In that case one uses the fact that $D\chi \in \Sigma$ implies $\Diamond\chi \in \Sigma$. ⊣

From 3.2.8 and (Van Benthem 1989*b*, Theorem 5.2) it follows that well-orders of type $< \omega \cdot k + n$ ($k \leq \omega$, $n < \omega$) all have distinct $\Diamond, D$-theories, while for $k \geq \omega$, $\omega \cdot k + n \equiv_{\Diamond, D} \omega \cdot \omega + n$. As a result, on well-orders $\mathcal{ML}(\Diamond)$ and $\mathcal{ML}(\Diamond, D)$ have the same discriminatory power. It seems that to be able to get stabilization of discriminatory power beyond $\omega \cdot \omega$ one needs to move to modal languages containing the *Until* and *Since* operators (cf. §4.4 for their definition).

On well-orders *Until* and *Since* can express every monadic Π^1_1-condition (in $<$, $=$) as a consequence of Kamp (1968)'s Theorem. By the proof of Rosenstein (1982, Theorem 6.22) one has that the discriminatory power of the monadic Π^1_1-fragment on well-orders stabilizes at ω^ω (and not before that), hence the same holds for the discriminatory power of *Until*, *Since*-logic.

THE LANGUAGES $\mathcal{ML}(\Diamond, D)$ AND $\mathcal{ML}(F, P)$

On strict linear orders the D-operator becomes definable in $\mathcal{ML}(F, P)$: on such frames we have $\mathfrak{F} \models (P\phi \vee F\phi) \leftrightarrow D\phi$. In fact, this may be generalized somewhat; call a frame $\mathfrak{F}$ *n-connected* ($n > 0$) if for any w, $v \in W$ with $w \neq v$, there exists a sequence $w_1, \ldots, w_n$ such that $w_1 = w$, $w_n = v$ and for each j ($1 \leq j < n$) either Rw_jw_{j+1} or $Rw_{j+1}w_j$. Then, using a suitable translation, one may show that on irreflexive, n-connected frames every $\Diamond, D$-formula is equivalent to one in $\mathcal{ML}(F, P)$. This shows that new results about standard modal languages may be obtained by studying extended ones: for it follows from 3.2.4 that on the class of irreflexive, n-connected frames every purely universal Π^1_1-sentence in $R, =$ is definable in $\mathcal{ML}(F, P)$.

By the next result there is no converse to our previous remarks: P is not definable in $\mathcal{ML}(\Diamond, D)$ — not even on strict linear orders.

3.2.9. THEOREM. $(\mathbb{Q}, <) \not\equiv_{F,P} (\mathbb{R}, <)$ *and* $(\mathbb{Q}, <) \equiv_{\Diamond,D} (\mathbb{R}, <)$.

Proof. The first part is well-known. To prove the second part, assume first that for some $\phi \in \mathcal{ML}(\Diamond, D)$ and valuation V, $(\mathbb{R}, <, V) \not\models \phi$. Using the ST-translation we find that $(\mathbb{R}, <, V) \models \exists x\, ST(\neg\phi)$. Hence by the Löwenheim-Skolem Theorem, $(\mathbb{Q}, <, V') \models \exists x\, ST(\neg\phi)$, where $V'(p) = V(p) \restriction \mathbb{Q}$, for all proposition letters p. It follows that $(\mathbb{Q}, <) \not\models \phi$.

Conversely, assume that for some $\phi \in \mathcal{ML}(\Diamond, D)$ and a valuation V, $(\mathbb{Q}, <, V) \not\models \phi$. Define Σ and $\mathfrak{M}_1$ as in the proof of 3.2.8. Then $\mathfrak{M}_1$ is transitive, linear and successive — both to the right and to the left. A model $\mathfrak{M}_2$ may then be constructed by replacing each cluster with an ordering of type λ if it is the left-most cluster, and otherwise, if it is degenerated it and its *non-degenerated* successor (by Segerberg (1970, Lemma 1.1) $\mathfrak{M}_1$ does not contain adjacent degenerated clusters) are replaced in one go with an ordering of type $1 + \lambda$; after that, the remaining non-degenerated clusters are also replaced by $1 + \lambda$. The valuation may then be extended to newly added points in such a way that an induction similar to the one in the proof of 3.2.8 yields $\mathfrak{M}_2 \not\models \phi$. $\dashv$

FURTHER COMPARISONS

Blackburn (1993*b*) presents a simple method of incorporating reference into modal logic by introducing a new sort of atomic symbols — *nominals* — to the modal language. These new symbols combine with other symbols of the language in the usual way to form formulas. Their only non-standard feature is that they are true at exactly *one* point in a model. Let $\mathcal{ML}_n(\Diamond)$ denote the language $\mathcal{ML}(\Diamond)$ with nominals added to it. From (Blackburn 1993*b*) we know that $\mathcal{ML}_n(\Diamond)$ is much more expressive than $\mathcal{ML}(\Diamond)$: important classes of frames undefinable in $\mathcal{ML}(\Diamond)$ become definable in $\mathcal{ML}_n(\Diamond)$. But it turns out that $\mathcal{ML}(\Diamond, D)$ is even more expressive than $\mathcal{ML}_n(\Diamond)$. To see this, let $n_0, n_1, n_2, \ldots$ range over nominals; let $p_0, p_1, p_2, \ldots$ denote the proposition letters in $\mathcal{ML}_n(\Diamond)$ and $\mathcal{ML}(\Diamond, D)$, and define $\tau : \mathcal{ML}_n(\Diamond) \to \mathcal{ML}(\Diamond, D)$ by putting $\tau(p_i) = p_{2i}$ and $\tau(n_i) = p_{2i+1}$, and by letting τ commute with the connectives and operators. Given a formula $\phi \in \mathcal{ML}_n(\Diamond)$, let $n_1, \ldots, n_k$ be the nominals occurring in ϕ, and define $(\phi)^* \in \mathcal{ML}(\Diamond, D)$ to be $O\tau(n_1) \wedge \ldots \wedge O\tau(n_k) \to \tau(\phi)$.

3.2.10. PROPOSITION. *Every class of frames that is definable in $\mathcal{ML}_n(\Diamond)$ is definable in $\mathcal{ML}(\Diamond, D)$, but not conversely.*

Proof. The first part follows from the observation that for any formula $\phi \in \mathcal{ML}_n(\Diamond)$, and any model (W, R, V), $(W, R, V), w \models \phi$ iff $(W, R, V^*), w \models \phi^*$, where $V^*(p) = V(\tau^{-1}(p))$. The second part follows from 3.2.2 and the fact that 1 is the only cardinality definable in $\mathcal{ML}_n(\Diamond)$ (cf. (Blackburn 1993*b*)). ⊣

Both Blackburn (1993*b*) and Gargov & Goranko (1991) study the extension $\mathcal{ML}_n(\Diamond, A)$ of $\mathcal{ML}_n(\Diamond)$ — here A is the operator defined in §3.1, whose semantics is given by $\mathfrak{M}, w \models A\phi$ iff for all $v \in W$, $\mathfrak{M}, v \models \phi$; it is sometimes called the *shifter* (by Blackburn (1993*b*)), or the *universal modality* by Gargov & Goranko (1991). By the above observations $\mathcal{ML}_n(\Diamond, A)$ is no more expressive than $\mathcal{ML}(\Diamond, D)$. Moreover, by a nice result in (Gargov & Goranko 1991) the converse holds as well:

3.2.11. THEOREM. *A class of frames is definable in $\mathcal{ML}_n(\Diamond, A)$ iff it is definable in $\mathcal{ML}(\Diamond, D)$.*

Writing $\mathcal{ML}_1 \leq \mathcal{ML}_2$ for: every class of frames definable in $\mathcal{ML}_1$ is definable in $\mathcal{ML}_2$, the combination of results from this section and earlier ones together with results from (Gargov & Goranko 1991) and (Goranko & Passy 1992), yields the picture in Figure 3.1. (In Figure 3.1 $\mathcal{ML}(\Diamond)^{seq}$ is $\mathcal{ML}(\Diamond)$ with sequential definability.)

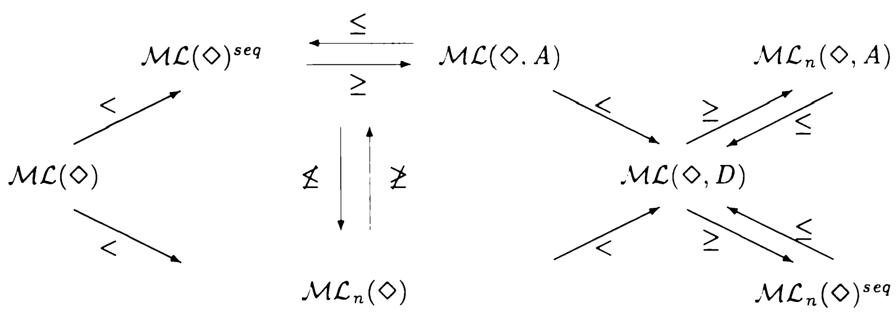


Figure 3.1: $\mathcal{ML}(\Diamond, D)$ and its neighbours.

3.3 Axiomatics

After recalling some basic facts, we first study logics in the language $\mathcal{ML}(D)$; we show that, given the right basic logic in $\mathcal{ML}(D)$, all its extensions are complete. After that we consider the basic logics in $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$. Finally, we discuss decidability issues and Sahlqvist Theorems for $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$.

Basic facts

We assume that the reader has a basic understanding of what an axiom, a derivation rule and a substitution is. Given those assumptions, a *derivation system* consists of a set of derivation rules and a set of axioms. A set of formulas Σ is *closed under a derivation rule* $\Delta \,/\, \phi$ if any instantiation of ϕ is in Σ whenever the corresponding instantiations of Δ are in Σ. A logic $\mathbf{L}$ over a derivation system is the least set of formulas containing the axioms and closed under the derivation rules. For ϕ a formula, $\mathbf{L}\phi$ denotes the logic $\mathbf{L}$ extended with ϕ as an axiom.

A *derivation* in a derivation system is a list $\phi_1, \ldots, \phi_n$ such that every ϕ_i is either an axiom, or obtained from $\phi_1, \ldots, \phi_{n-1}$ by application of a derivation rule. A *theorem* is any formula that can appear as the last item of a derivation. We write $\vdash_{\mathbf{L}} \phi$ to denote that ϕ is a theorem of $\mathbf{L}$; $\Sigma \vdash_{\mathbf{L}} \phi$ (read: Σ derives ϕ in $\mathbf{L}$) if there are $\sigma_1, \ldots, \sigma_n \in \Sigma$ such that $\vdash_{\mathbf{L}} \bigwedge_i \sigma_i \to \phi$. Finally, a set of formulas Σ is $\mathbf{L}$-*consistent* if $\Sigma \not\vdash_{\mathbf{L}} \bot$; it is *maximal consistent* if it is consistent and no consistent set (in the same language) properly extends it.

3.3.1. Definition. $\mathbf{K}$ is the basic modal logic in $\mathcal{ML}(\Diamond)$. It extends propositional logic with the following distribution axiom:

(K1) $\Box(p \to q) \to (\Box p \to \Box q)$,

As rules of inference it has Modus Ponens, Substitution, and a 'Necessitation Rule'

(MP) $\phi, \phi \to \psi \,/\, \psi$
(SUB) $\phi \,/\, \sigma\phi$, for any substitution instance $\sigma\phi$ of ϕ
(NEC) $\phi \,/\, \Box\phi$.

$\mathbf{K}_t$ is the basic logic in the temporal language $\mathcal{ML}(F, P)$. It extends propositional logic with two distribution axioms and two axioms relating F and P:

(K_t1a) $G(p \to q) \to (Gp \to Gq)$
(K_t1b) $H(p \to q) \to (Hp \to Hq)$
(K_t2a) $p \to GPp$
(K_t2a) $p \to HFp$.

Its rules of inference are Modus Ponens, Substitution, and two Necessitation Rules, one for G, and for H.

For K a class of frames, Σ a set of formulas, we write $\Sigma \models_{\mathsf{K}} \phi$ (read ϕ is a *consequence* of Σ over K) if for every model $\mathfrak{M}$ based on a frame in K, and every x in $\mathfrak{M}$ we have that $\mathfrak{M}, x \models \sigma$, for all $\sigma \in \Sigma$, implies $\mathfrak{M}, x \models \phi$.

THE LOGICS IN $\mathcal{ML}(D)$

3.3.2. DEFINITION. $\mathbf{DL}^-$ is propositional logic plus the following schemata:

(D1) $\overline{D}(p \to q) \to (\overline{D}p \to \overline{D}q)$,
(D2) $p \to \overline{D}Dp$ (symmetry),
(D3) $DDp \to (p \vee Dp)$ (pseudo-transitivity).

As rules of inference it has Modus Ponens, Substitution, and a 'Necessitation Rule' for $\overline{D}$: $\phi \,/\, \overline{D}\phi$.

3.3.3. THEOREM. *Let* $\Sigma \cup \{\phi\} \subseteq \mathcal{ML}(D)$. *Then* $\Sigma \vdash_{\mathbf{DL}^-} \phi$ *iff* $\Sigma \models \phi$.

Proof. Soundness is immediate. To prove completeness, assume $\Sigma \nvdash_{\mathbf{DL}^-} \phi$, and let $\Delta \supseteq \Sigma \cup \{\neg\phi\}$ be a maximal $\mathbf{DL}^-$-consistent set. Consider $W_\Delta := \{\Gamma : \exists n\,(R_D)^n \Delta\Gamma\}$, where Γ ranges over maximal $\mathbf{DL}^-$-consistent sets and R_D is the canonical relation defined by: $R_D\Gamma_1\Gamma_2$ iff for all $\overline{D}\psi \in \Gamma_1$, $\psi \in \Gamma_2$. Then $\forall xy\,(R_Dxy \to R_Dyx)$ and $\forall xyz\,(R_Dzy \wedge R_Dyz \to R_Dxz \vee x = z)$. If there are any R_D-reflexive points, let c be such a point; replace it with *two* points c_1, c_2, and adapt R_D by putting $R_Dc_1c_2$, and conversely, and by putting R_Dc_iw (R_Dwc_i) if R_Dcw (R_Dwc) ($i = 1, 2$). In the resulting structure R_D is real inequality, and ϕ is refuted somewhere. ⊣

One may be inclined to think that $\mathbf{DL}^-$ is the basic logic in $\mathcal{ML}(D)$ — just like $\mathbf{K}$ is the basic logic in $\mathcal{ML}(\Diamond)$. Recalling that a logic is *incomplete* whenever it cannot derive some of its consequences, $\mathbf{DL}^-$ is, so to speak, not as stable as $\mathbf{K}$ since in $\mathcal{ML}(\Diamond)$ incompleteness phenomena occur only with more exotic extensions of $\mathbf{K}$ (Van Benthem 1979), while, in contrast, here's a very simple incomplete extension of $\mathbf{DL}^-$:

3.3.4. EXAMPLE. Consider the system $\mathbf{DL}^- + (p \to Dp)$. Then $\mathbf{DL}^- + (p \to Dp) \models \bot$, since no frame validates $\mathbf{DL}^- + (p \to Dp)$. On the other hand, $\mathbf{DL}^- + (p \to Dp) \nvdash \bot$. To see this, recall that a *general frame* is a triple $\mathfrak{F} = (W, R, \mathcal{W})$, where $\mathcal{W} \subseteq P(W)$ contains $\emptyset$, and is closed under the Boolean operations as well as the operator L_R (cf. 3.1.2); valuations on a general frame should take their values inside $\mathcal{W}$. Now, let $\mathfrak{F} = (W, R, \mathcal{W})$, where $W = \{0, 1\}$, $R = \emptyset$ and $\mathcal{W} = \{\emptyset, \{0, 1\}\}$ (so D is interpreted using the relation $R = \emptyset$). Then $\mathfrak{F} \models \mathbf{DL}^- + (p \to Dp)$. Therefore, $\mathbf{DL}^- + (p \to Dp)$ is incomplete.

To avoid incompleteness phenomena as those sketched above, we follow some suggestions by Yde Venema and Valentin Goranko, and add the following *irreflexivity rule* to $\mathbf{DL}^-$:

(IR_D) $p \wedge \neg Dp \to \phi \,/\, \phi$, provided p does not occur in ϕ.

The idea to use special kinds of derivation rules to obtain completeness results originates with Dov Gabbay who used an irreflexivity rule to axiomatize the set of $\Diamond$-formulas valid on irreflexive frames, cf. (Gabbay 1981). A rule analogous to (IR_D) has been used to obtain completeness results for axiom systems in languages with nominals, cf. (Blackburn 1993*b*, Gargov & Goranko 1991, Gargov et al. 1987).

Let **DL** denote $\mathbf{DL}^-$ plus the rule (IR_D). Our next aim is to prove that in terms of general consequence, **DL** has no effects over $\mathbf{DL}^-$. To this end it suffices to show that **DL** precisely axiomatizes the basic logic in $\mathcal{ML}(D)$. The material below will be familiar to many. Nevertheless, for future applications (in Chapters 4 and 5) the proofs below will be more general and given in greater detail than needed for our present purposes.

We will use the (IR_D) rule to build a model consisting of maximal consistent sets Δ, each containing a unique proposition letter p_Δ such that $p \wedge \neg Dp \in \Delta$; hence this p_Δ serves as a 'name' for Δ. To achieve this we 'paste' names on appropriate spots inside the formulas that we come across.

The construction below is an adaptation of constructions given by Roorda (1993) and Venema (1991, Chapter 2), while earlier versions of the argument can be found in (Gabbay 1981) and (Gabbay & Hodkinson 1991).

3.3.5. DEFINITION. We write $\psi \trianglelefteq \phi$ for 'ψ occurs as a subformula in ϕ.' *We don't identify different occurrences of ψ in ϕ.*

Define a function $\mathtt{Paste}(\nu, \psi, \phi)$ (paste ν (the name) next to the occurrence of ψ in ϕ) by induction on ϕ, treating ψ as an atomic symbol in ϕ.

$$\begin{array}{rcl}
\mathtt{Paste}(\nu, \psi, p) & = & p \text{ if } \psi \not\equiv p \\
\mathtt{Paste}(\nu, \psi, \psi) & = & \nu \wedge \psi \\
\mathtt{Paste}(\nu, \psi, \neg\phi) & = & \neg\phi \\
\mathtt{Paste}(\nu, \psi, \phi \wedge \chi) & = & \mathtt{Paste}(\nu, \psi, \phi) \wedge \mathtt{Paste}(\nu, \psi, \chi) \\
\mathtt{Paste}(\nu, \psi, D\phi) & = & D\mathtt{Paste}(\nu, \psi, \phi)
\end{array}$$

As an example, we have

$$\begin{array}{ll}
\multicolumn{2}{l}{\mathtt{Paste}(Op, r \wedge \neg q, r \wedge D(r \wedge \neg q))} \\
= & \mathtt{Paste}(Op, r \wedge \neg q, r) \wedge \mathtt{Paste}(Op, r \wedge \neg q, D(r \wedge \neg q)) \\
= & r \wedge D\mathtt{Paste}(Op, r \wedge \neg q, r \wedge \neg q) \\
= & r \wedge D(Op \wedge r \wedge \neg q).
\end{array}$$

Pasting names into formulas does not hurt in that no new consequences arise. To prove this we need the following.

3.3.6. LEMMA. (Switching Lemma) *Let $\vdash$ denote $\vdash_{\mathbf{DL}}$. Then $\vdash D\phi \rightarrow \psi$ iff $\vdash \phi \rightarrow \overline{D}\psi$.*

Proof.

$$\begin{array}{rcl}
\vdash D\phi \rightarrow \psi & \Rightarrow & \vdash \overline{D}D\phi \rightarrow \overline{D}\psi \\
& \Rightarrow & \vdash \phi \rightarrow \overline{D}\psi \\
& \Rightarrow & \vdash D\neg\psi \rightarrow \neg\phi \\
& \Rightarrow & \vdash \overline{D}D\neg\psi \rightarrow \overline{D}\neg\phi \\
& \Rightarrow & \vdash \neg\psi \rightarrow \overline{D}\neg\phi \\
& \Rightarrow & \vdash D\phi \rightarrow \psi. \quad \dashv
\end{array}$$

3.3.7. LEMMA. (Pasting Lemma) *Assume that Op has no proposition letters in common with ϕ and θ. For any subformula occurrence $\psi \trianglelefteq \phi$ we have that*

$$\vdash \texttt{Paste}(Op, \psi, \phi) \to \theta \quad \textit{implies} \quad \vdash \phi \to \theta.$$

Proof. This is by induction on ϕ, treating ψ as an atomic symbol. There are two atomic cases. If $\phi \equiv q \not\equiv \psi$, then $\texttt{Paste}(Op, \psi, \phi) = q$, and we're done. If $\phi \equiv \psi$, then $\texttt{Paste}(Op, \psi, \phi) = (Op \wedge \phi)$, and we find

$$\begin{aligned} \vdash Op \wedge \phi \to \theta \quad &\Rightarrow \quad \vdash Op \to (\phi \to \theta) \\ &\Rightarrow \quad \vdash \phi \to \theta, \quad \text{by } (\mathrm{IR}_D). \end{aligned}$$

For the induction step we again distinguish several cases. The Boolean cases are straightforward. For the modal case we have $\texttt{Paste}(Op, \psi, D\phi) = D\texttt{Paste}(Op, \psi, \phi)$. Observe

$$\begin{aligned} &\vdash D\texttt{Paste}(Op, \psi, \phi) \to \theta \\ &\quad \Rightarrow \quad \vdash \texttt{Paste}(Op, \psi, \phi) \to \overline{D}\theta, \quad \text{by the Switching Lemma} \\ &\quad \Rightarrow \quad \vdash \phi \to \overline{D}\theta, \quad \text{by induction hypothesis} \\ &\quad \Rightarrow \quad \vdash D\phi \to \theta, \quad \text{by the Switching Lemma.} \quad \dashv \end{aligned}$$

3.3.8. LEMMA. *If Σ is a consistent set of formulas, p does not occur in Σ, $\phi \in \Sigma$, and $\psi \trianglelefteq \phi$, then $\Sigma \cup \{\texttt{Paste}(Op, \psi, \phi)\}$ is consistent.*

Proof. If $\Sigma \cup \{\texttt{Paste}(Op, \psi, \phi)\}$ is inconsistent, then for some $\theta_1, \ldots, \theta_n \in \Sigma$, we have $\vdash \texttt{Paste}(Op, \psi, \phi) \to (\neg\theta_1 \vee \ldots \vee \neg\theta_n)$. Hence, by the Pasting Lemma, $\vdash \phi \to (\neg\theta_1 \vee \ldots \vee \neg\theta_n)$. But then Σ would be inconsistent. $\dashv$

3.3.9. DEFINITION. A *theory* is a set of formulas. For Φ a set of proposition letters, a theory Δ is a Φ-*theory* if all proposition letters occurring in Δ are in Φ. Δ is a maximal **L**-consistent Φ-theory if no Φ-theory properly extends Δ while being consistent.

Δ is a *distinguishing* Φ-*theory* if for every $\phi \in \Delta$ and every $\psi \trianglelefteq \phi$, there is a proposition letter p such that $\texttt{Paste}(Op, \psi, \phi) \in \Delta$.

3.3.10. LEMMA. (Extension Lemma) *Let Σ be a consistent Φ-theory. Let $\Phi' \supseteq \Phi$ be an extension of Φ by a countably infinite set of propositional variables. Then there is a maximal consistent, distinguishing Φ'-theory Σ' extending Σ.*

Proof. The proof is a variation on the usual construction of maximal consistent sets. Let $\Phi' = \Phi \cup \Phi''$, where $\Phi \cap \Phi'' = \emptyset$ and $\Phi'' = \{p_0, p_1, p_2, \ldots\}$; let $\Phi_n = \Phi \cup \{p_i : i \leq n\}$. Define $\mathcal{ML}_n$ to be $\mathcal{ML}(D)$ with Φ_n as its set of proposition letters; $\mathcal{ML}_\omega$ is $\mathcal{ML}(D)$ with Φ' as its set of proposition letters.

A theory $\Delta \supseteq \Sigma$ is called an *approximation* if for some n it is a consistent Φ_n-theory; for an approximation Δ the atomic symbol $p_{n+1} \in \Phi_{n+1}$ is called the *new atomic symbol for* Δ if n is the least number m such that Δ is a Φ_m-theory. Now, fix an enumeration of all pairs (ϕ, ψ) where $\psi \trianglelefteq \phi$. A *defect* for a theory Δ is a pair (ϕ, ψ) such that $\phi \in \Delta$, while for no proposition letter p $\texttt{Paste}(Op, \psi, \phi) \in \Delta$. Put

$$\Delta^+ = \begin{cases} \Delta \cup \{\texttt{Paste}(Op, \psi, \phi)\} & \text{where } p \text{ is the new atom for } \Delta, \\ & \text{and } (\phi, \psi) \text{ is the } \textit{first} \text{ defect for} \\ & \Delta \text{ (if it exists),} \\ \Delta, & \text{otherwise.} \end{cases}$$

Clearly, by the previous Lemma, if Δ is an approximation, then so is Δ^+.

To define the extension Σ' of Σ that we are after, let $\phi_0, \phi_1, \ldots$ be an enumeration of the formulas in $\mathcal{ML}_\omega$, and define

$$\begin{array}{rcl} \Sigma_0 & = & \Sigma \\ \Sigma_{2n+1} & = & \begin{cases} \Sigma_{2n} \cup \{\phi_n\}, & \text{if this is consistent} \\ \Sigma_{2n} \cup \{\neg\phi_n\}, & \text{otherwise} \end{cases} \\ \Sigma_{2n+2} & = & (\Sigma_{2n+1})^+ \\ \Sigma' & = & \bigcup_{n \in \omega} \Sigma_n. \end{array}$$

Then Σ' is a maximal consistent, distinguishing Φ'-theory extending Σ. $\dashv$

3.3.11. DEFINITION. (Canonical relation) Define a canonical relation R_D^c between distinguishing theories by putting $R_D^c \Delta\Sigma$ if for all ϕ, $\phi \in \Sigma$ implies $D\phi \in \Delta$. We use R_D^c to interpret the D-operator.

3.3.12. LEMMA. (Successor Lemma) *Assume that Δ is a maximal consistent distinguishing theory. If Δ contains a formula of the form $D\phi$, then the required R_D^c-successor exists: if $D\phi \in \Delta$, then there is a maximal consistent distinguishing Σ with $\phi \in \Sigma$ and $R_D^c \Delta\Sigma$.*

Proof. If $D\phi \in \Delta$, then, for some proposition letter p, we have $D(\phi \wedge Op) \in \Delta$. Define $\Sigma := \{\psi : D(Op \wedge \psi) \in \Delta\}$. We check that Σ has all the desired properties. First, Σ is consistent, for otherwise there are $\psi_1, \ldots, \psi_n \in \Sigma$ with $\vdash (\bigwedge_i \psi_i) \rightarrow \bot$. By Theorem 3.3.3 we have

$$\vdash_{\mathbf{DL}} \Big(\bigwedge_i D(Op \wedge \psi_i)\Big) \rightarrow D\Big(Op \wedge \bigwedge_i \psi_i\Big).$$

Hence $D(Op \wedge \bigwedge_i \psi_i) \in \Delta$, and thus $D\bot \in \Delta$ — a contradiction. Second, Σ is maximal: if $\psi \notin \Sigma$, then $D(Op \wedge \psi) \notin \Delta$. But then $D(Op \wedge \neg\psi) \in \Delta$, and $\neg\psi \in \Sigma$. Third, it is obvious that Σ is an R_D^c-successor of Δ. Fourth, to see that Σ is distinguishing, assume that $\chi \in \Sigma$, $\psi \trianglelefteq \chi$; we have to show that $\texttt{Paste}(Oq, \psi, \chi) \in \Sigma$ for some Oq. Now, $\chi \in \Sigma$ implies $D(Op \wedge \chi) \in \Delta$. As Δ is distinguishing this yields an atomic symbol q such that

$$\texttt{Paste}\Big(Oq, \psi, D(Op \wedge \chi)\Big) = D\Big(Op \wedge \texttt{Paste}(Oq, \psi, \chi)\Big) \in \Delta,$$

This implies $\texttt{Paste}(Oq, \psi, \chi) \in \Sigma$. $\dashv$

Here, finally, is the definition of a canonical model.

3.3.13. DEFINITION. We define a *provisional canonical model* $\mathfrak{M}^c$ as follows. Fix a set of proposition symbols Φ.

$$\mathfrak{M}^c = (W^c, R^c_D, V^c),$$

where the relation R^c_D is defined as in Definition 3.3.11, and W^c is the set of all maximal consistent distinguishing theories over Φ. V^c is the *canonical valuation* given by $\Delta \in V^c(p)$ iff $p \in \Delta$. On the provisional canonical model $\mathfrak{M}^c$ we interpret the operator D using the relation R^c_D: $\Delta \models D\phi$ iff for some Σ we have both $R^c_D\Delta\Sigma$ and $\phi \in \Sigma$.

The provisional canonical model $\mathfrak{M}^c$ has almost all the required properties to count as a model for $\mathcal{ML}(D)$. By a simple argument (or by appealing to the fact that the **DL**-axioms are Sahlqvist formulas expressing universal first-order conditions), the axioms (D2), (D3) are canonical. This implies that the relation R^c_D is symmetric (by axiom (D2)) and pseudo-transitive[1] (by axiom (D3)). Moreover, by construction the relation R^c_D is irreflexive. Thus, the only possible shortcoming $\mathfrak{M}^c$ has, is that $(R^c_D \cup =)$ is not the universal relations on W^c. This will be fixed below. But first we state the following.

3.3.14. LEMMA. (Provisional Truth Lemma) *Consider the provisional canonical model $\mathfrak{M}^c$. For all $\Delta \in W^c$ and all formulas ϕ in $\mathcal{ML}(D)$, we have $\mathfrak{M}^c, \Delta \models \phi$ iff $\phi \in \Delta$.*

3.3.15. DEFINITION. (Final Canonical Model) Fix a distinguishing set Δ in the provisional canonical model. Define W^f to be the set of all Σ such that for some n, $(R^c_D)^n\Delta\Sigma$; R^f_D is $R^c_D \cap (W^f \times W^f)$ (the restriction of R^c_D to W^f), and for every proposition letter p, $V^f(p) = V^c(p) \cap W^f$ (the restriction of V^c to W^f).

A *final canonical model* for **DL** is any tuple $\mathfrak{M}^f = (W^f, R^f_D, V^f)$ with W^f, R^f_D, V^f defined as above.

3.3.16. LEMMA. (Structure Lemma) *On $\mathfrak{M}^f$ the relation R^f_D is real inequality: $R^f_D\Delta\Sigma$ iff $\Delta \neq \Sigma$.*

3.3.17. LEMMA. (Final Truth Lemma) *Consider a final canonical model $\mathfrak{M}^f$. For all $\Delta \in W^f$ and all formulas ϕ in $\mathcal{ML}(D)$, we have $\mathfrak{M}^f, \Delta \models \phi$ iff $\phi \in \Delta$.*

3.3.18. THEOREM. (Completeness Theorem) *Let $\Sigma \cup \{\phi\}$ be a set of $\mathcal{ML}(D)$-formulas. Then $\Sigma \vdash_{\mathbf{DL}} \phi$ iff $\Sigma \models \phi$.*

Proof. Soundness is immediate. To prove completeness, assume that $\Sigma \not\vdash_{\mathbf{DL}} \phi$. By the Extension Lemma $\Sigma \cup \{\neg\phi\}$ can be extended to a distinguishing set Σ'. Consider a final canonical model $\mathfrak{M}^f$ such that Σ' is in $\mathfrak{M}^f$. By the Final Truth Lemma $\mathfrak{M}^f, \Sigma' \models \sigma$, for all $\sigma \in \Sigma$, and $\mathfrak{M}^f, \Sigma' \not\models \phi$. $\dashv$

It follows from 3.3.3 and 3.3.18 that the rule (IR_D) is superfluous in the basic logic. Why then, one might ask, did we go through all the trouble to prove Theorem 3.3.18? First, because the above construction of canonical models in

[1]That is: R^c_D satisfies $\forall xyz\,(Rxy \wedge Ryz \rightarrow x = z \vee Rxz)$.

which every element has a unique name lays the groundwork for things to come in Chapters 4 and 5.

Second, having added the (IR_D) rule to our base logic we do obtain new consequences in extensions of **DL**: let **X** be **DL** + $(p \rightarrow Dp)$. Then **X** is inconsistent, and thus complete. (To see that it's inconsistent, note that for any proposition letter q, $\vdash_{\mathbf{X}} (q \wedge \neg Dq \rightarrow \bot)$, hence by the rule ($\mathrm{IR}_D$), $\vdash_{\mathbf{X}} \bot$.)

Better still, if $\mathbf{L} = \mathbf{DL} + \{ \phi_i : i \in I \}$ is any extension of **DL** in $\mathcal{ML}(D)$, then **L** is complete. To prove this we show that every extension **L** is *d-canonical*: if $\mathbf{L} = \mathbf{DL} + \{ \phi_i : i \in I \}$, then every **L**-axiom ϕ_i is valid on every (final) canonical frame for **L**, that is, on every frame underlying a final canonical model for **L**. We need the following.

3.3.19. DEFINITION. A *canonical general frame* for $\mathbf{L} \supseteq \mathbf{DL}$ is a structure $(W, R_D, \mathcal{W})$, where (W, R_D) is the frame underlying a final canonical model as defined in Definition 3.3.15, and $\mathcal{W} = \{ X \subseteq W : \exists\phi\forall\Delta\, (\phi \in \Delta \leftrightarrow \Delta \in X) \}$.

Let $\mathfrak{F}_{\mathbf{L}} = (W, R_D, \mathcal{W})$ be a canonical general frame for $\mathbf{L} \supseteq \mathbf{DL}$. A set $X \subseteq W$ is *definable in* $\mathfrak{F}_{\mathbf{L}}$ if $X \in \mathcal{W}$. A valuation V is *definable in* $\mathfrak{F}_{\mathbf{L}}$ if for every $\phi \in \mathcal{ML}(D)$, $V(\phi)$ is definable.

3.3.20. PROPOSITION. *Let $\mathfrak{F}_{\mathbf{L}} = (W, R_D, \mathcal{W})$ be a canonical general frame for a logic $\mathbf{L} \supseteq \mathbf{DL}$. Let $X \subseteq W$ be finite or co-finite. Then X is definable in $\mathfrak{F}_{\mathbf{L}}$.*

3.3.21. PROPOSITION. *Let $\mathfrak{F}_{\mathbf{L}} = (W, R_D, \mathcal{W})$ be a canonical general frame for a logic $\mathbf{L} \supseteq \mathbf{DL}$. Let V be a valuation. If for all proposition letters p, $V(p)$ is either finite or co-finite, then V is definable in $\mathfrak{F}_{\mathbf{L}}$.*

Proof. This follows from 3.3.20 and the fact that for any ϕ, $V(D\phi)$ is either $\emptyset$, W, or the complement of a singleton. ⊣

We prove canonicity of $\mathbf{L} \supseteq \mathbf{DL}$ by showing that every satisfiable formula is satisfiable using definable valuations.

3.3.22. LEMMA. *Let $\phi \in \mathcal{ML}(D)$. Let W be any non-empty set. If for some valuation V, $(W, V) \not\models \phi$, then there is a valuation V' with $(W, V') \not\models \phi$ and such that $V'(p)$ is either finite or co-finite for all proposition letters p.*

Proof. This is (De Rijke 1992*b*, Lemma 3.11). The proof uses the same techniques as Theorem 7.7.2 below. ⊣

3.3.23. THEOREM. *Let* **L** *be any extension of* **DL**. *Then* **L** *is d-canonical.*

Proof. If **L** is inconsistent, there is nothing to prove. So assume that **L** is consistent. Let ψ be an **L**-axiom. It suffices to show that ψ is d-canonical. Suppose it is not, and let $\mathfrak{F}_{\mathbf{L}} = (W, R_D, \mathcal{W})$ be a final canonical frame for **L** with $(W, R_D, V) \not\models \psi$, for some valuation V. By Lemma 3.3.22 there is a definable valuation V' with $(W, R_D, V') \not\models \psi$. Hence $\mathfrak{F}_{\mathbf{L}} \not\models \psi$ — a contradiction. ⊣

3.3.24. THEOREM. *Let* **L** *be any extension of* **DL**. *Then* **L** *is complete.*

3.3.25. REMARK. Theorem 3.3.24 fully justifies the addition of the irreflexivity rule (IR_D) to the basic modal logic of the D-operator. The result also shows that $\mathcal{ML}(D)$ is a language with very weak expressive power.

What about decidability of the satisfiability problem for the above logics? By a filtration argument (Proposition 3.1.7) over **DL** every satisfiable formula ϕ has a model of size at most $2 \cdot 2^{2n}$, where n is the length of ϕ. De Smit & Van Emde Boas (1990) show that one can do considerably better: a formula in $\mathcal{ML}(D)$ has a model over the base logic **DL** iff it has a model with at most $4n$ elements, where n is the length of ϕ.

3.3.26. THEOREM. *The satisfiability problem for $\mathcal{ML}(D)$ is NP-complete.*

As to the satisfiability problem for finite extensions of **DL** (that is: extensions with only finitely many additional axioms), we will only establish that they are all decidable, leaving the matter of their complexity for further study. From the proof of (De Rijke 1992*b*, Lemma 3.11) it follows that a formula α of monadic first-order logic over identity has a model iff it has a model of size at most $|\alpha| \cdot 2^{|\alpha|}$, where $|\alpha|$ is the length of α. Now, let $\mathbf{L} = \mathbf{DL} + \phi$ be a finite extension of **DL**. By Theorem 3.2.3 ϕ is equivalent to a formula α_ϕ of the form '($\neg$) there exist at least n objects.' For ψ an arbitrary $\mathcal{ML}(D)$-formula, ψ is **L**-satisfiable iff $(\alpha_\phi \wedge ST(\psi))$ is satisfiable in monadic first-order logic over identity. By the above remarks exhaustive search of small models establishes decidability of the latter.

3.3.27. THEOREM. *Let $\mathcal{L}$ be a finite extension of* **DL**. *Then* **L** *is decidable.*

LOGICS IN $\mathcal{ML}(\Diamond, D)$ AND $\mathcal{ML}(F, P, D)$

As with the base logic in $\mathcal{ML}(D)$ we formulate two versions of the base logics in $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$, one version without and one with special derivation rules. The first base logic $\mathbf{DL}_m^-$ in $\mathcal{ML}(\Diamond, D)$ is $\mathbf{DL}^- + \mathbf{K}$ plus the inclusion axiom

(INC$_m$) $(\Diamond p \to p \vee Dp)$.

Its rules of inference are those of $\mathbf{DL}^-$ plus those of **K**. The first base logic $\mathbf{DL}_t^-$ in the language $\mathcal{ML}(F, P, D)$ is $\mathbf{DL}^- + \mathbf{K}_t$ plus the inclusion axioms

(INC$_t$1) $Fp \to p \vee Dp$
(INC$_t$2) $Pp \to p \vee Dp$.

Its rules of inference are those of $\mathbf{DL}^-$ plus those of $\mathbf{K}_t$.

For a logic **L** extending $\mathbf{DL}_m^-$ in $\mathcal{ML}(\Diamond, D)$ the canonical relation R_D is defined as in Definition 3.3.15, and the canonical relation $R_\Diamond$ is defined as $\{ (\Sigma, \Gamma) :$ for all $\psi \in \Gamma$, $\Diamond\psi \in \Sigma \}$. Analogous definitions may be given for extensions of $\mathbf{DL}_t^-$, where the canonical relations are denoted R_F, R_P, R_D.

3.3.28. LEMMA. (Structure Lemma)

1. *Let* **L** *extend* $\mathbf{DL}_m^-$. *Then, for the canonical relations R_D and $R_\Diamond$ we have $R_\Diamond \subseteq (R_D \cup =)$.*
2. *Let* **L** *extend* $\mathbf{DL}_t^-$. *Then, for the canonical relations R_F, R_P and R_D we have $R_F \subseteq (R_D \cup =)$ and $R_P \subseteq (R_D \cup =)$, and R_F is the converse of R_P.*

3.3.29. THEOREM.

1. *Let* $\Sigma \cup \{\phi\} \subseteq \mathcal{ML}(\Diamond, D)$. *Then* $\Sigma \vdash_{\mathbf{DL}_m^-} \phi$ *iff* $\Sigma \models \phi$.
2. *Let* $\Sigma \cup \{\phi\} \subseteq \mathcal{ML}(F, P, D)$. *Then* $\Sigma \vdash_{\mathbf{DL}_t^-} \phi$ *iff* $\Sigma \models \phi$.

Proof. Similar to the proof of 3.3.3. Use Proposition 3.3.28. $\dashv$

As with $\mathbf{DL}^-$, simple incomplete extensions of $\mathbf{DL}_m^-$ and $\mathbf{DL}_t^-$ are easily found, suggesting that the base logics should be beefed up with additional rules. Before defining more powerful base logics in $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$ it will help to recall what it took us to do the proof of Theorem 3.3.18. One of the main things was the Pasting Lemma allowing us to paste names in any formula that we came across, without creating new derivable formulas. The Pasting Lemma, in turn, depended in an essential way on the Switching Lemma:

$$\vdash D\phi \to \psi \;\text{ iff }\; \vdash \phi \to \overline{D}\psi.$$

In modal languages where every modal operator $\#$ comes with a converse $\#^{\smile}$, one automatically has a Switching Lemma by the 'converse axioms' (K_t2a), (K_t2b):

$$\vdash \#\phi \to \psi \;\text{ iff }\; \vdash \phi \to \overline{\#}^{\smile}\psi.$$

(Observe that D is its own converse because of the axiom (D2).) Since in the language $\mathcal{ML}(F, P, D)$ F and P are each others converse, the construction underlying the proof of Theorem 3.3.18 carries through almost without modifications for the temporal language. In $\mathcal{ML}(\Diamond, D)$, however, we lack a converse of $\Diamond$. To overcome this we need to add infinitely many derivation rules.

First, we extend our terminology (Definition 3.3.5). Let $\#$ denote one of $\Diamond$, F or P.

$$\mathtt{Paste}(Op, \psi, \#\phi) \;=\; \#\mathtt{Paste}(Op, \psi, \phi).$$

3.3.30. DEFINITION. The logic $\mathbf{DL}_m$ in $\mathcal{ML}(\Diamond, D)$ extends $\mathbf{DL}_m^-$ with the following collection of derivation rules:

(IR_D^*) $\neg\mathtt{Paste}(Op, \psi, \phi) \,/\, \neg\phi$, provided $\psi \trianglelefteq \phi$ and p does not occur in ϕ.

The logic $\mathbf{DL}_t$ in $\mathcal{ML}(F, P, D)$ extends $\mathbf{DL}_t^-$ with the rule (IR_D).

3.3.31. LEMMA. (Pasting Lemma) *Let* $\vdash$ *denote either* $\vdash_{\mathbf{DL}_m}$ *or* $\vdash_{\mathbf{DL}_t}$. *Assume Op has no proposition letters in common with* ϕ *and* θ. *For any subformula occurrence* $\psi \trianglelefteq \phi$ *we have that*

$$\vdash \mathtt{Paste}(Op, \psi, \phi) \to \theta \;\; \textit{implies} \;\; \vdash \phi \to \theta.$$

Proof. For $\mathbf{DL}_t$ we need only complement the proof of the Pasting Lemma for $\mathbf{DL}$ with the induction cases for F, P. But here we we can use the same argument as for D, since F and P enjoy a Switching Lemma. For $\mathbf{DL}_m$ we reason as follows:

$$\begin{aligned} \vdash \texttt{Paste}(Op, \psi, \phi) \to \theta \quad &\Rightarrow \quad \vdash \neg(\neg\theta \wedge \texttt{Paste}(Op, \psi, \phi)) \\ &\Rightarrow \quad \vdash \neg\texttt{Paste}(Op, \psi, \neg\theta \wedge \phi) \\ &\Rightarrow \quad \vdash \neg(\neg\theta \wedge \phi), \text{ by the rule } (\mathrm{IR}^*_D) \\ &\Rightarrow \quad \vdash \phi \to \theta. \ \dashv \end{aligned}$$

3.3.32. THEOREM.

1. *Let* $\Sigma \cup \{\phi\} \subseteq \mathcal{ML}(\Diamond, D)$. *Then* $\Sigma \vdash_{\mathbf{DL}_m} \phi$ *iff* $\Sigma \models \phi$.
2. *Let* $\Sigma \cup \{\phi\} \subseteq \mathcal{ML}(F, P, D)$. *Then* $\Sigma \vdash_{\mathbf{DL}_t} \phi$ *iff* $\Sigma \models \phi$.

Proof. In both cases the proof is entirely analogous to the proof of the completeness result for **DL** (3.3.18) up to the definition of the provisional canonical models. For $\mathbf{DL}_m$ we add the canonical relation $R^c_\Diamond$, and for $\mathbf{DL}_t$ we add the canonical relation R^c_F (recall that R_P is the converse of R_F by the Structure Lemma 3.3.28). Then, the Provisional Truth Lemma may be proved without further ado. The Final Canonical Model is defined as before, by generating along R^c_D; because of the Structure Lemma 3.3.28 the resulting structure will also be a generated substructure for $R^c_\Diamond$ (in the case of $\mathbf{DL}_m$) and R^c_F (in the case of $\mathbf{DL}_t$). Hence, for all formulas truth is preserved when going from the Provisional to the Final Canonical Model. $\dashv$

The above only establishes completeness of the basic logics in $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$. But the results extended to the counterparts in $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$ of many (individual) well-known logics in those languages. Further, the paper (De Rijke 1992*b*) on which this Chapter is based, contains complete axiomatizations of the $\mathcal{ML}(\Diamond, D)$-formulas valid on certain special structures and familiar classes of frames.

As before, the question arises whether the above logics $\mathbf{DL}_m$ and $\mathbf{DL}_t$ are decidable. Using the extended filtrations of 3.1.7 one easily establishes that for both $\mathbf{DL}_m$ and $\mathbf{DL}_t$ the satisfiability problem is decidable. In fact, from 3.1.7 it follows that if ϕ is a satisfiable $\mathcal{ML}(\Diamond, D)$- or $\mathcal{ML}(F, P, D)$-formula, then ϕ is satisfiable in a model with at most $2 \cdot 2^{2n}$ elements, for n the length of ϕ. Hence the satisfiability problems for $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$ are both decidable in non-deterministic exponential time. But we can improve this, following (Pratt 1979) and (Halpern & Moses 1985).

3.3.33. THEOREM. *The satisfiability problems for* $\mathcal{ML}(\Diamond, D)$ *and* $\mathcal{ML}(F, P, D)$ *are decidable in EXPTIME.*

Proof. Let $Cl(\phi)$ be the closure of $\{\phi\}$ under subformulas and single negations. Define $\mathcal{S} = \{\Gamma \subseteq Cl(\phi) : \Gamma \text{ is maximally propositional consistent}\}$. Suppose ϕ is satisfiable in $\mathfrak{M}$. Define $\mathcal{S}_{\mathfrak{M}} = \{\Gamma \in \mathcal{S} : \mathfrak{M}, x \models \Gamma, \text{ for some } x \text{ in } \mathfrak{M}\}$. Clearly $\mathcal{S}_{\mathfrak{M}} \subseteq \mathcal{S}$. Let $\Sigma \subseteq 2^{\mathcal{S}}$ consist of all maximal sets $\mathcal{S}' \subseteq \mathcal{S}$ such that

1. if $\neg D \neg\psi \in Cl(\phi)$ then for all $\Gamma, \Gamma' \in \mathcal{S}'$: $\psi, \neg D\neg\psi \in \Gamma$ iff $\psi, \neg D\neg\psi \in \Gamma'$,
2. if $\neg D\psi \in Cl(\phi)$, there is at most one set $\Gamma \in \mathcal{S}'$ with $\psi, \neg D\psi \in \Gamma$.

As $Cl(\phi) \leq 2n$, for n the length of ϕ, there exist at most 2^{2n} sets $\mathcal{S}' \subseteq \mathcal{S}$ satisfying 1, and as the number of formulas of the form $\neg D\psi$ in $Cl(\phi)$ is at most n, at most $|\mathcal{S}'|^n$ subsets of $\mathcal{S}'$ occur in Σ. Hence, $|\Sigma|$ is exponential in the size of ϕ.

For every $\mathcal{S}_1$ we define a finite sequence of sets $\mathcal{S}_1 \supsetneq \mathcal{S}_2 \supsetneq \mathcal{S}_3 \supsetneq \cdots$ such that if ϕ is satisfiable in a model $\mathfrak{M}$ and $\mathcal{S}_{\mathfrak{M}} \subseteq \mathcal{S}_1$, then $\mathcal{S}_{\mathfrak{M}} \subseteq \mathcal{S}_i$. Assume $\mathcal{S}_i$ has already been defined. A set $\Gamma \in \mathcal{S}_i$ is called *incoherent* if one of the following occurs:

1. $D\psi \in \Gamma$, but there is no $\Gamma' \in \mathcal{S}_i$ such that for all $D\chi \in Cl(\phi)$, $\chi \in \Gamma'$ implies $D\chi \in \Gamma$
2. $\Diamond\psi \in \Gamma$, but there is no $\Gamma' \in \mathcal{S}_i$ such that for all $\Diamond\chi \in Cl(\phi)$, $\chi \in \Gamma'$ implies $\Diamond\chi \in \Gamma$.

If there are incoherent sets in $\mathcal{S}_i$, let $\mathcal{S}_{i+1}$ be $\mathcal{S}_i$ minus the incoherent sets. Otherwise, ϕ is satisfiable iff $\phi \in \Gamma$ for some $\Gamma \in \mathcal{S}_i$.

Finally, as $\mathcal{S}_i$ is of exponential size, and $|\mathcal{S}_{i+1}| < |\mathcal{S}_i|$, the algorithm terminates after exponentially many steps. Determining incoherence takes polynomial time in the length of the representation of $\mathcal{S}_1$. Thus, for every $\mathcal{S} \in \Sigma$, the algorithm takes at most deterministic polynomial time. Given the exponential size of Σ, we can determine if ϕ is satisfiable in EXPTIME. $\dashv$

3.3.34. THEOREM. *The satisfiability problems for $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$ are both EXPTIME-complete, provided that the languages contain at least one proposition letter.*

Proof. Spaan (1993, Theorems 5.4.5, 5.4.6) proves that the satisfiability problem for the language $\mathcal{ML}(\Diamond, A)$ (with at least one proposition letter) is EXPTIME-hard. This provides the lower bound. The upper bound follows from Theorem 3.3.33. $\dashv$

ON SAHLQVIST THEOREMS FOR $\mathcal{ML}(\Diamond, D)$ AND $\mathcal{ML}(F, P, D)$

As will be shown shortly, we can not hope for analogues of Theorem 3.3.24 for $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$ — there are many incomplete extensions of $\mathbf{DL}_m$ and $\mathbf{DL}_t$ in their respective languages. Nevertheless, we may try and transfer known *general* completeness results from $\mathcal{ML}(\Diamond)$ and $\mathcal{ML}(F, P)$ to $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$, respectively. One important such result is Sahlqvist's Theorem (Sahlqvist 1975). It says that for a large class S of formulas in $\mathcal{ML}(\Diamond)$, any extension $\mathbf{K} + S'$, for $S' \subseteq S$ is complete. In fact, Sahlqvist (1975) proves two things concerning this class S:

1. every ϕ in S corresponds to a first-order condition α_ϕ on frames that is effectively obtainable from ϕ, and
2. for any $S' \subseteq S$, $\mathbf{K} + S'$ is complete with respect to the class of frames satisfying α_ϕ, for $\phi \in S'$.

We now discuss the extent to which the above generalizes to $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$.

3.3.35. DEFINITION. We define Sahlqvist formulas. Let $\# \in \{ \Diamond, F, P, D \}$, and $\overline{\#} \in \{ \Box, G, H, \overline{D} \}$. A *simple Sahlqvist formula* is an implication $\phi \to \pi$ where

1. π is positive (in the usual syntactic sense), and
2. ϕ is built up from
   (a) negative formulas,
   (b) closed formulas, that is formulas without occurrences of proposition letters, and
   (c) 'boxed proposition letters,' that is formulas of the form $\overline{\#}_1 \dots \overline{\#}_n p$,

   using only $\wedge$, $\vee$, and $\Diamond$.

A *Sahlqvist formula* is any formula that can be obtained from simple Sahlqvist formulas by applications of $\wedge$ and $\overline{\#}$.

3.3.36. THEOREM. *Let ϕ be a Sahlqvist formula in $\mathcal{ML}(\Diamond, D)$. Then ϕ corresponds to a first-order condition on frames, effectively obtainable from ϕ.*

Proof. This is a consequence of the general Sahlqvist result in Chapter 7, §§7.5, 7.6. ⊣

3.3.37. EXAMPLE. 1. The $\mathcal{ML}(\Diamond, D)$-formula $\Diamond p \to Dp$ translates into the second-order condition $\forall p\,\big(\exists y\,(Rxy \wedge p(y)) \to \exists z\,(x \neq z \wedge p(z))\big)$, which reduces to $\neg Rxx$.

2. A slightly more complex example: the $\mathcal{ML}(\Diamond, D)$-formula $p \wedge \neg Dp \to A\neg\Diamond p$ translates into $\forall p\,(p(x) \wedge \exists yz\,(Ryz \wedge p(z)) \to \exists u\,(u \neq x \wedge p(u)))$, which reduces to $\neg\exists y\,(Ryx)$.
3. As a final example, here is a Sahlqvist formula in $\mathcal{ML}(F, P, D)$: $Gp \wedge Hp \to \overline{D}p$. This translates into

$$\forall p\,\Big(\forall y\,(Rxy \to p(y)) \wedge \forall z\,(Ryz \to p(z)) \to \forall y\,(y \neq x \to p(y))\Big),$$

which reduces to $\forall y\,(x \neq y \to Rxy \vee Ryx)$.

(The details of these reductions will be supplied in Chapter 7.)

What about the completeness half of a Sahlqvist Theorem for $\mathcal{ML}(\Diamond, D)$? An early version of the paper (De Rijke 1992*b*) on which this Chapter is based, did contain a 'proof' for the completeness half of a Sahlqvist Theorem for $\mathcal{ML}(\Diamond, D)$. However, Yde Venema found a serious mistake in it; he subsequently did prove a full Sahlqvist Theorem for $\mathcal{ML}(F, P, D)$. Unfortunately his proof has no adaptation to the Sahlqvist fragment of $\mathcal{ML}(\Diamond, D)$, for it relies heavily on the fact that if a set X is definable (in the sense of definition 3.3.19, but with some obvious changes) in a canonical general frame for a logic $\mathbf{L}$ in $\mathcal{ML}(F, P, D)$, then so is the cone $\{ y : R_F xy \text{ for some } x \in X \}$. In general, such cones need not be definable in canonical general frames for logics in $\mathcal{ML}(\Diamond, D)$, cf. (Venema 1991). For a special subclass of Sahlqvist forms we do have the following result. Let a *weak* Sahlqvist formula be a Sahlqvist formula built up from simple Sahlqvist formulas $\phi \to \pi$ whose antecedent ϕ is construed without 'boxed proposition letters' (as under 2c above).

3.3.38. THEOREM. *Let ϕ be a weak Sahlqvist form in $\mathcal{ML}(\Diamond, D)$. Then ϕ corresponds to a first-order condition α_ϕ effectively obtainable from ϕ, and $\mathbf{DL}_m + \phi$ is complete w.r.t. the class of frames satisfying α_ϕ.*

Proof. The correspondence half is a subcase of 3.3.36. For a proof of the completeness half we refer the reader to (Venema 1991). $\dashv$

Although the class of weak Sahlqvist forms is strictly smaller than the class of all Sahlqvist forms, it is still a large one, which contains $\mathcal{ML}(\Diamond, D)$-equivalents of many important first-order conditions on binary relations. E.g., by inspecting the proof of 3.2.4 one can see that it contains equivalents of all Horn-like first-order sentences of the form $\forall \vec{x}\, (\alpha \rightarrow \beta)$, where α, β are positive quantifier-free $\mathcal{L}(\boldsymbol{\tau}_0)$-formulas.

ON TRANSFER OF PROPERTIES

One of the general research topics in modal logic that we mentioned in §2.4 had to do with *transferring* properties of a logic to its extensions in, for example, richer languages. In the present setting we briefly consider the following special instances of the *Transfer Problem*. Let $\mathbf{L}$ extend $\mathbf{K}$ in $\mathcal{ML}(\Diamond)$ with axioms $\{\, \phi_i : i \in I \,\}$; the *minimal extension* of $\mathbf{L}$ in $\mathcal{ML}(\Diamond, D)$ is $\mathbf{DL}_m$ plus the axioms ϕ_i read as axioms over $\mathcal{ML}(\Diamond, D)$. The question is: if $\mathbf{L}$ has property P, does its minimal extension have P? Here we consider only four of the many obvious properties one may study: *(un-) decidability* and *(in-) completeness*.

The two 'negative' properties both transfer. As to incompleteness, let $\mathbf{L}$ be a logic in $\mathcal{ML}(\Diamond)$. To show that if $\mathbf{L}$ is incomplete, then so is its minimal extension $\mathbf{L}'$ in $\mathcal{ML}(\Diamond, D)$, it suffices to show that $\mathbf{L}'$ is conservative over $\mathbf{L}$. To this end, assume $\mathbf{L} \nvdash \phi$. Then, by the completeness of $\mathbf{K}$, we find a model $\mathfrak{M}$, and a $w \in \mathfrak{M}$, such that $\mathfrak{M}, w \models \mathbf{L}^* \cup \{\, \neg\phi \,\}$, where $\mathbf{L}^*$ is the set of all $\mathcal{ML}(\Diamond)$-instances of the axioms of $\mathbf{L}$. Now obviously, $\mathfrak{M} \models \mathbf{DL}_m$, and also $\mathfrak{M}, w \models \mathbf{L}^{**}$, where $\mathbf{L}^{**}$ is the set of $\mathcal{ML}(\Diamond, D)$-instances of the axioms of $\mathbf{L}$ (this is because for any set $V(\phi)$, $V(D\phi)$ is either $\emptyset$, $V(\top)$, or the complement of $V(\phi)$). But then $\mathbf{L}' \nvdash \phi$.[2] By essentially the same conservativity argument, it follows that *un*decidability also transfers.

As to the 'positive' properties, decidability does not transfer. This follows from a result due to Edith Spaan: there is a finitely axiomatized logic $\mathbf{L}$ in $\mathcal{ML}(\Diamond)$ with a decidable satisfiability problem, such that its minimal extension in $\mathcal{ML}(\Diamond, A)$ has an undecidable satisfiability problem (Spaan 1993, Theorem 4.2.1). It follows that the minimal extension of $\mathbf{L}$ in $\mathcal{ML}(\Diamond, D)$ also has an undecidable satisfiability problem. The question whether completeness transfers from logics in $\mathcal{ML}(\Diamond)$ to their minimal extension in $\mathcal{ML}(\Diamond, D)$ is still open. For some base languages other than $\mathcal{ML}(\Diamond)$ results are known, however. For example, completeness does *not* transfer from the language of propositional dynamic

[2] As an aside, new and fairly simple incomplete logics occur as well: let $\mathbf{X}$ be $\mathbf{DL}_m + (\Diamond p \rightarrow Dp) + (\Diamond\Diamond p \rightarrow \Diamond p) + (\Box\Diamond p \rightarrow \Diamond\Box p)$. Then $\mathbf{X} \models \bot$ since $\Diamond p \rightarrow Dp$ defines irreflexivity of R, while given $\Diamond\Diamond p \rightarrow \Diamond p$, $\Box\Diamond p \rightarrow \Diamond\Box p$ defines $\forall x \exists y\, (Rxy \rightarrow \forall z\, (Ryz \rightarrow z = y))$. However, by a routine argument involving general frames, $\mathbf{X} \nvdash \bot$.

logic **PDL** to its extension with the D-operator: by results of Passy & Tinchev (1991) **PDL** plus the D-operator has a Σ^1_1-complete satisfiability problem, hence it cannot have a complete axiomatization.

3.4 Definability

Given the perspective of this dissertation of modal languages as description languages for relational structures, the issue of expressive power is a central one. One way to get a grip on the expressive power is to find semantic characterizations of the definable classes of structures.[3]

This issue can be addressed from two angles, corresponding to the two modes of interpreting the modal languages of this Chapter, on models and on frames. Below we consider definability aspects of our languages on both levels.

Definability of classes of models

When interpreted on models our modal languages $\mathcal{ML}(D)$, $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$ end up as fragments of the first-order language $\mathcal{L}(\boldsymbol{\tau}_1)$ via the standard translation ST as defined in §3.2. Those fragments can be characterized by means of appropriate notions of bisimulations.

3.4.1. Definition. A *basic R-bisimulation* is a non-empty binary relation Z between two models $\mathfrak{M}_1 = (W_1, R_1, V_1)$ and $\mathfrak{M}_2 = (W_2, R_2, V_2)$ such that the following holds:

1. if Zwv then w, v verify the same proposition letters,
2. if Zwv, $w' \in W_1$ and R_1ww' then $Zw'v'$ for some $v' \in W_2$ with R_2vv' (*forth*),
3. if Zwv, $v' \in W_2$ and R_2vv' then $Zw'v'$ for some $w' \in W_1$ with R_1ww' (*back*).

3.4.2. Definition. Given a notion of bisimulation, say X-bisimulations, a classical formula $\alpha(x_1, \ldots, x_n)$ is *invariant for X-bisimulations* if, for all models $\mathfrak{M}_1$, $\mathfrak{M}_2$, all X-bisimulations Z between $\mathfrak{M}_1$ and $\mathfrak{M}_2$, and all $w_1, \ldots, w_n \in W_1$, $v_1, \ldots, v_n \in W_2$ such that $Zw_1v_1, \ldots, Zw_nv_n$, we have $\mathfrak{M}_1 \models \alpha[w_1, \ldots, w_n]$ iff $\mathfrak{M}_2 \models \alpha[v_1, \ldots, v_n]$.

3.4.3. Theorem. *Let $\alpha(x)$ be an $\mathcal{L}(\boldsymbol{\tau}_1)$-formula. Then α is (equivalent to) the translation of an $\mathcal{ML}(\Diamond)$-formula iff it is invariant for basic R-bisimulations.*

Proof. This is essentially (Van Benthem 1983, Theorem 3.9). ⊣

According to Chapter 6 the way to understand the relation between bisimulation and modal languages is that both are concerned with certain simple patterns in relational structures. This is clearly reflected in the definition of basic

[3] Another way would be to link the modal languages with automata, as is done for certain temporal languages by Thomas (1989). This won't be pursued here.

R-bisimulations, and it also shows us how we should define bisimulations for $\mathcal{ML}(D)$, $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$.

3.4.4. DEFINITION. We define bisimulations for the modal languages $\mathcal{ML}(D)$, $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$.

A non-empty binary relation Z between two models (W_1, V_1) and (W_2, V_2) is called a *basic $\neq$-bisimulation* if it satisfies item 1 of Definition 3.4.1 plus item 4 below:

4. if Zwv, $w' \in W_1$ and $w \neq w'$ then $Zw'v'$ for some $v' \in W_2$ with $v \neq v'$, and a similar condition going backwards.

(So, basically, Z is a basic $\neq$-bisimulation if it extends the union of two mappings f and g^{-1}, where both $f : W_1 \to W_2$ and $g : W_2 \to W_1$ are injective.)

A non-empty binary relation Z between two models (W_1, R_1, V_1) and (W_2, R_2, V_2) is a *basic $R, \neq$-bisimulation* if it satisfies items 1, 2 and 3 of Definition 3.4.1 plus the above item 4.

A non-empty binary relation Z between two models (W_1, R_1, V_1) and (W_2, R_2, V_2) is a *basic $R, R^{\smile}, \neq$-bisimulation* if it satisfies items 1, 2 and 3 of Definition 3.4.1, the above item 4, and item 5 below:

5. if Zwv, $w' \in W_1$ and $R_1{}^{\smile}ww''$ then $Zw'v'$ for some $v' \in W_2$ with $R_2{}^{\smile}vv'$, and a similar condition going backwards.

As with the standard modal language $\mathcal{ML}(\Diamond)$, the fragments of our classical languages that correspond to our modal languages can now be characterized in terms of their behaviour with respect to the above notions of bisimulation.

3.4.5. THEOREM. *Let τ_1^- be the vocabulary τ_1 with the binary relation symbol R left out.*

1. *Let $\alpha(x)$ be an $\mathcal{L}(\tau_1^-)$-formula. Then α is (equivalent to) the translation of a formula in $\mathcal{ML}(D)$ iff it is invariant for basic $\neq$-bisimulations.*
2. *Let $\alpha(x)$ be an $\mathcal{L}(\tau_1)$-formula. Then α is (equivalent to) the translation of a formula in $\mathcal{ML}(\Diamond, D)$ iff it is invariant for basic $R, \neq$-bisimulations.*
3. *Let $\alpha(x)$ be an $\mathcal{L}(\tau_1)$-formula. Then α is (equivalent to) the translation of a formula in $\mathcal{ML}(F, P, D)$ iff it is invariant for basic $R, R^{\smile}, \neq$-bisimulations.*

Proof. See Example 6.7.2. $\dashv$

Next we apply further general results from Chapter 6 to obtain a definability result for classes of models. To this end we find it convenient to take *pointed models* $(\mathfrak{M}, w)$ with a distinguished world w and $\mathfrak{M}$ as before as the basic notion of model. Evaluation of a formula on a pointed model takes place at the distinguished point.

A class K of pointed models is *definable* by means of a modal formula if for some modal formula ϕ we have $(\mathfrak{M}, w) \in \mathsf{K}$ iff $(\mathfrak{M}, w) \models \phi$.

3.4.6. THEOREM. *Let K be a class of pointed models. Then*

1. K *is definable by an* $\mathcal{ML}(D)$*-formula iff it is closed under* $\neq$*-bisimulations and ultraproducts, while its complement is closed under ultraproducts.*
2. K *is definable by an* $\mathcal{ML}(\Diamond, D)$*-formula iff it is closed under* $R, \neq$*-bisimulations and ultraproducts, while its complement is closed under ultraproducts.*
3. K *is definable by an* $\mathcal{ML}(F, P, D)$*-formula iff it is closed under* R, $R^{\smile}$, $\neq$*-bisimulations and ultraproducts, while its complement is closed under ultraproducts.*

Proof. See Example 6.7.2. ⊣

Using the general approach from Chapter 6 further definability results can be derived from Theorem 3.4.5 for all of the languages considered here, cf. Example 6.7.2.

Definability of classes of frames

On the level of frames a standard way to capture the definable classes is by means of closure under certain 'algebraic' operations; to obtain smooth formulations it is often assumed that the classes considered are closed under elementary equivalence. According to a classic result by Goldblatt & Thomason (1974) the definable classes of frames that are closed under elementary equivalence, are precisely the ones that are closed under generated subframes, p-morphic images and disjoint unions, while their complement is closed under ultrafilter extensions. In this subsection we characterize the definable classes of frames in $\mathcal{ML}(D)$, $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$.

For $\mathcal{ML}(D)$ a more concrete characterization of the definable classes of frames is available than for $\mathcal{ML}(\Diamond)$. Recall that a frame for $\mathcal{ML}(D)$ is simply a non-empty set W. For $\mathcal{ML}(D)$-frames $\mathfrak{F}$, $\mathfrak{G}$, let $\mathfrak{F} \cong \mathfrak{G}$ denote that $\mathfrak{F}$, $\mathfrak{G}$ are isomorphic (that is: there is a bijection $f : \mathfrak{F} \to \mathfrak{G}$). For K a class of frames for $\mathcal{ML}(D)$, K$/\cong$ denotes the subclass of K containing exactly one representative of every $\cong$-class in K; we say that a property holds of K modulo isomorphism if it holds of K$/\cong$.

3.4.7. Theorem. *A class* K *of frames for* $\mathcal{ML}(D)$ *is definable by an* $\mathcal{ML}(D)$*-formula if it is closed under isomorphism and if modulo isomorphism either* K *or the complement of* K *is a finite set of finite frames.*

Proof. The direction from right to left is clear. For the converse, let K be defined by the $\mathcal{ML}(D)$-formula ϕ. Then K is closed under isomorphisms. For the remainder of the proof we need the following auxiliary definition and claim. Let $\mathfrak{F}_m$ denote the frame $(\{1, \ldots, m\})$; (so $\mathfrak{F}_m$ is definable in $\mathcal{ML}(D)$).

Claim. If for all $k \in \mathbb{N}$ there is an $l > k$ such that $\mathfrak{F}_l \in$ K, then for some $n \in \mathbb{N}$, $\mathfrak{F}_m \in$ K for all $m \geq n$.

Proof of the Claim. Assume that K satisfies the antecedent of the Claim. By Proposition 3.2.1 ϕ is equivalent to a first-order sentence α. Let n be the quantifier rank of α; this will turn out to be the n we are looking for. Choose any $\mathfrak{F}_l \in \mathsf{K}$ with $l > n$. By assumption $\mathfrak{F}_l \models \phi$. Now, take any $\mathfrak{F}_m$ with $m \geq n$. By general considerations $\mathfrak{F}_m$ and $\mathfrak{F}_n$ verify the same first-order sentences (in the pure equality language) of quantifier rank at most n. Therefore, $\mathfrak{F}_l \models \phi$ implies $\mathfrak{F}_l \models \alpha$ implies $\mathfrak{F}_m \models \alpha$ implies $\mathfrak{F}_m \models \phi$. So $\mathfrak{F}_m \in \mathsf{K}$. $\dashv_{Claim}$

Returning to the main argument, suppose that $\mathsf{K}/{\cong}$ is infinite. Then, by the Claim $\mathfrak{F}_m \in \mathsf{K}$ for all $m \geq n$, for some n. Then $\mathsf{K}/{\cong}$ must contain all (representatives of) infinite frames as well, by arguments similar to those establishing the Claim. It follows that $\mathsf{K}^c/{\cong}$ can only contain finite frames $\mathfrak{F}_l$ for $l < n$; but then $\mathsf{K}^c/{\cong}$ must be finite. If, on the other hand, $\mathsf{K}/{\cong}$ is finite, then $\mathsf{K}/{\cong}$ cannot contain infinite frames, for otherwise it would contain arbitrarily large finite ones, and thus be infinite. So, if $\mathsf{K}/{\cong}$ is finite, it must be a finite set of finite frames, as required. $\dashv$

The study of definable classes of frames in $\mathcal{ML}(\Diamond, D)$ in the above spirit of Goldblatt & Thomason (1974) has been undertaken in (Goranko 1990) and (Gargov & Goranko 1991).

3.4.8. DEFINITION. We assume that the reader knows what a modal algebra is. Let $(\cdot)^+$ denote the mapping that takes (general) frames to modal algebras (as defined, for example, in (Van Benthem 1983, Chapter 4)).

$\mathfrak{F}' = (W', R', \mathcal{W})$ is a *$\neq$-collapse* of the general frame $\mathfrak{F} = (W, R, \mathcal{W})$ if $\mathfrak{F}'$ is a subframe of $\mathfrak{F}$ and if there exists a subframe $\mathfrak{G}$ of $\mathfrak{F}$ such that $(\mathfrak{F}')^+ \cong (\mathfrak{G})^+$ and for each $x \in W'$, $\{ y : Rxy \} \subseteq [\{ y : xR'y \}]_{\mathfrak{G}^+}$, where $[X]_{\mathfrak{G}^+}$ is the least element of $(\mathfrak{G})^+$ containing X. (Warning: $\neq$-collapses should not be confused with the modal and extended modal collapse from §3.1.)

When we adapt the classic Goldblatt and Thomason result about definability of classes of frames in $\mathcal{ML}(\Diamond)$, closure under p-morphic images, generated subframes and disjoint unions will be left out because of the remarks following Definition 3.1.1. What remains, is closure under $\neq$-collapses:

3.4.9. THEOREM. (Gargov and Goranko) *A class of frames that is closed under elementary equivalence is definable in $\mathcal{ML}(\Diamond, D)$ iff it is closed under $\neq$-collapses.* •

Gargov and Goranko prove Theorem 3.4.9 largely by algebraic means. As finite frames don't have proper collapses, a class of finite frames is definable iff it is closed under isomorphisms; for this special case a purely modal proof may be given:

3.4.10. PROPOSITION. *A class K of finite frames is definable in $\mathcal{ML}(\Diamond, D)$ iff it is closed under isomorphisms.*

Proof. Let $\mathfrak{F}$ be a finite frame with $W = \{ w_1, \ldots, w_n \}$, and $\mathfrak{F} \models \mathrm{Th}_{\Diamond,D}(\mathsf{K})$. Assume $p_1, \ldots, p_n$ are different proposition letters. Define $\chi_{\mathfrak{F}}$ by

$$\bigwedge_{1 \leq i \leq n} Ep_i \wedge A\Big(\bigvee_{1 \leq i \leq n} (p_i \wedge \neg Dp_i)\Big) \wedge$$

$$A\Big(\bigwedge_{1\leq i\neq j\leq n}(p_i \to \neg p_j)\Big) \wedge A\Big(\bigwedge_{1\leq i,j\leq n}(p_i \to \# p_j)\Big),$$

where $\# \equiv \Diamond$ if Rw_iw_j holds, and $\# \equiv \neg\Diamond$ otherwise. Then for any frame $\mathfrak{G}$, there is a valuation V with $(\mathfrak{G}, V) \not\models \neg\chi_{\mathfrak{F}}$ iff $\mathfrak{G} \cong \mathfrak{F}$. In particular $\mathfrak{F} \not\models \neg\chi_{\mathfrak{F}}$. Hence $\neg\chi_{\mathfrak{F}} \notin \mathrm{Th}_{\Diamond,D}(\mathsf{K})$. Thus for some $\mathfrak{G} \in \mathsf{K}$, $\mathfrak{G} \not\models \neg\chi_{\mathfrak{F}}$. So $\mathfrak{F} \in \mathsf{K}$. $\dashv$

For the sake of completeness we conclude by considering the definable classes of frames in $\mathcal{ML}(F, P, D)$; given the above result for $\mathcal{ML}(\Diamond, D)$ this does not bring any surprises.

Recall from (Thomason 1972) that a *temporal algebra* is simply a modal algebra with an additional unary function corresponding to the backward looking modal operator P and satisfying the 'converse axioms' (K_t2a) and (K_t2b). The appropriate notion of $\neq$-collapse here extends the one from Definition 3.4.8 with an additional closure condition: $\{\, y : Ryx \,\} \subseteq [\{\, y : R'yx \,\}]_{\mathfrak{G}^+}$. Then, an elementary class of frames is definable in $\mathcal{ML}(F, P, D)$ iff it is closed under such extended $\neq$-collapses.

3.5 Concluding remarks

To overcome some of the most striking deficiencies in expressive power of the standard modal and temporal languages, we added a difference operator to our languages. With relatively simple means we obtained a boost in expressive power, while the theory of the resulting languages was still quite 'manageable.'

To bring the deductive power of our languages enriched with the D-operator in line with their increased expressive powers, we needed to add special derivation rules. The mechanism for proving completeness in the presence of such rules was developed in quite some detail, especially because of later uses in this dissertation; in Chapters 4 and 5 we will call in the help of the D-operator to establish completeness results. (This is a general strategy: if in a given modal logic some condition is not expressible, or completeness is hard to obtain, add the D-operator to the language and then use the completeness construction of this Chapter or the expressive power of the language; one may subsequently try to remove the additions.)

Finally, we listed some results related to definability aspects of the standard modal and temporal languages equipped with the D-operator; some of those were direct consequences of the general theory developed in Chapters 7 and 6.

To conclude this Chapter, here are some questions.

1. The irreflexivity rule (IR_D) didn't add any consequences to the base logic in $\mathcal{ML}(D)$; it did add new consequences to some extensions of the base logic (Example 3.3.4). Is there a general result saying when the rule does or does not add new consequences? In a recent manuscript Yde Venema shows that this question is closely related to the question whether or not a logic enjoys the Craig Interpolation Property.

2. What is the general proof theory of the D-operator? In particular, what problems (if any) does the irreflexivity rule (IR_D) create for a decent sequent-style axiomatization of logics containing the D-operator?
3. At present it is unknown whether the completeness half of a Sahlqvist Theorem for the *full* Sahlqvist fragment of $\mathcal{ML}(\Diamond, D)$ exists (cf. Theorem 3.3.38), although individual cases of logics in the full fragment on which the present author tried his hands, all turned out complete. Is there a general result covering the full fragment after all? If not, how far can Theorem 3.3.38 be extended?
4. This is about transfer of completeness (cf. page 41). If transfer of completeness of logics in $\mathcal{ML}(\Diamond)$ to their minimal extensions in $\mathcal{ML}(\Diamond, D)$ fails, is there a largest language intermediate between $\mathcal{ML}(\Diamond)$ and $\mathcal{ML}(\Diamond, D)$ to which completeness does transfer? And if completeness does transfer from $\mathcal{ML}(\Diamond)$ to $\mathcal{ML}(\Diamond, D)$, what is the richest language $\mathcal{ML}$ extending $\mathcal{ML}(\Diamond)$ such that completeness transfers from $\mathcal{ML}$ to $\mathcal{ML}$ plus D?
5. We know that all extensions of the basic logic **DL** in $\mathcal{ML}(D)$ are decidable (Theorem 3.3.27)? But what is their complexity?

4

A System of Dynamic Modal Logic

4.1 Introduction

Over the past decade logicians have paid more and more attention to dynamic aspects of reasoning. Motivated by examples taken from such diverse disciplines as natural language semantics, linguistic analysis of discourse, the philosophy of science, artificial intelligence and program semantics, a multitude of logical systems have been proposed, each of them equipped with the predicate 'dynamic'. At present it is not clear at all what it is that makes a logical system a *dynamic* system. One of the very few general perspectives on dynamic matters is due to Van Benthem (1991*b*). This Chapter studies a dynamic modal language ($\mathcal{DML}$) designed within this perspective by Van Benthem (1989*a*, 1991*b*). Before introducing the formal aspects of $\mathcal{DML}$, let me sketch the main ideas underlying it.

Nowadays many logical systems focus on the structure and processing of information. Often these calculi do not aim at dealing with what is true at information *states*, but rather with *transitions* between such states. Cognitive notions, however, have a dual character. Actual inference, for instance, is a mixture of more dynamic short-term effects and long-term static ones. Thus, in a logical analysis of dynamic matters it is desirable to have two levels of propositions co-existing. In addition, the two levels may mutually influence each other; the effects of transitions are often couched in static terms, and the processing of pieces of static information may give rise to instructions as to getting from one cognitive state to another. The general format for $\mathcal{DML}$, then, is one of two levels, of *states* and of *transitions*, plus systematic *interactions* between them.

Given this choice of basic ingredients we are faced with a number of questions, including:

1. what are states and transitions?
2. what are the appropriate connectives?
3. which relations model the interaction between states and transitions?
4. do we evaluate formulas only at states, or also at transitions?

In $\mathcal{DML}$ we opt for the following. We abstract from any particular choice of states, and take them to be primitive objects without further structure. Although recent years have witnessed the emergence of calculi in which transitions are primitive objects too (in (Van Benthem 1991*a*, Venema 1991) they are called *arrows*), here our transitions will simply be ordered pairs of states. In our choice of connectives we will be rather conservative: we use propositions with the usual Boolean operations to talk about states, and we use the usual relation algebraic operations (including converse) to combine procedures that denote sets of transitions. Among our procedures there will be a relation $\sqsubseteq$ denoting an abstract notion of *information growth* or *change*, which we will assume to be a pre-order.

As to the interaction between states and transitions, states are linked to transitions via *modes*, and transitions are linked to states via *projections*, as in Figure 4.1 below. The choice of projections and modes will, of course, depend on the particular application one has in mind; here, we choose a very basic set. The projections we consider return, given a procedure as input, its *domain*, *range* and *fix points*. Given our interests in dynamic matters here, they form a natural choice, expressing, for instance, whether or not in a given state a certain change is at all possible. The modes we consider take a formula ϕ as input, and return the procedure consisting of all moves along the information ordering to states where ϕ holds, or all moves *backwards* along the ordering to states where ϕ fails; in addition there is the simple 'test-for-ϕ' relation.

$$\begin{tikzcd} \begin{array}{c}\text{propositions}\\ (\mathsf{BA})\end{array} \arrow[r, shift left, "\text{modes}"] & \begin{array}{c}\text{procedures}\\ (\mathsf{RA})\end{array} \arrow[l, shift left, "\text{projections}"] \end{tikzcd}$$

Figure 4.1: Propositions and procedures.

The issue whether we evaluate formulas at states, transitions, or both, is a subtle one. As in Propositional Dynamic Logic (**PDL**) our language has syntactic items referring to relations, but the notions of validity and consequence are couched solely in terms of formulas denoting sets of states; thus $\mathcal{DML}$ cannot express the *identity* of two relations directly — only the effects of making transitions can be measured. That is: $\mathcal{DML}$-formulas can only be evaluated at states, not at pairs. Chapter 5 below deals with a truly two-sorted language in which states and transitions have, so to say, equal rights.

I believe $\mathcal{DML}$ is not just another device for reasoning about dynamics and change, but, rather, that it provides a more general framework in which other proposals can be described and compared. A number of such descriptions and comparisons have been given by Van Benthem (1989*a*, 1991*b*) and De Rijke (1992*a*); §4.3 below contains a brief survey.

The main purpose of this Chapter is to study the language $\mathcal{DML}$ in precise and formal detail. After some initial definitions in §4.2, §4.3 contains examples of the uses of the $\mathcal{DML}$; these include Theory Change, Update Semantics, and Dynamic Inference. In §4.4 the expressive power of the language is studied; a precise syntactic description is given of the first-order counterpart of $\mathcal{DML}$, as well as a characterization in terms of bisimulations using general techniques from

Chapters 6 and 7. In §4.5 (un-) decidability results for satisfiability in $\mathcal{DML}$ are given. §4.6 provides a complete axiomatization of validity in the language of $\mathcal{DML}$; the proofs employ the general techniques of Chapter 3. Finally, §4.7 contains concluding remarks and suggestions for further work.

4.2 Preliminaries

4.2.1. Definition. Let Φ be a set of proposition letters. We define the dynamic modal language $\mathcal{ML}(\mathsf{do}, \mathsf{ra}, \mathsf{fix}, \mathsf{exp}, \mathsf{con}; \Phi)$, or $\mathcal{DML}(\Phi)$ or even $\mathcal{DML}$ for short. Its formulas and procedures (typically denoted by ϕ and α, respectively) are built up from proposition letters ($p \in \Phi$) according to the following rules

$$\begin{aligned} \phi &::= p \mid \bot \mid \top \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \mathsf{do}(\alpha) \mid \mathsf{ra}(\alpha) \mid \mathsf{fix}(\alpha), \\ \alpha &::= \mathsf{exp}(\phi) \mid \mathsf{con}(\phi) \mid \alpha_1 \cap \alpha_2 \mid \alpha_1 ; \alpha_2 \mid -\alpha \mid \alpha^{\smallsmile} \mid \phi?. \end{aligned}$$

$\mathcal{DML}$-formulas are assumed to live in a set $Form(\Phi)$, the procedures in a set $Proc(\Phi)$, and the elements of $Form(\Phi) \cup Proc(\Phi)$ are referred to as $\mathcal{DML}$-*expressions*.

At several occasions we will refer to a version of $\mathcal{DML}$ with multiple base relations $\sqsubseteq_i$ taken from a set Ω, together with corresponding modes exp_i; we use $\mathcal{DML}(\Phi, \Omega)$ to refer to this language.

The intended interpretation of the above connectives and operators is the following. A formula $\mathsf{do}(\alpha)$ ($\mathsf{ra}(\alpha)$) is true at a state x iff x is in the domain (range) of α, and $\mathsf{fix}(\alpha)$ is true at x if x is a fixed point of α. The interpretation of $\mathsf{exp}(\phi)$ (read: expand with ϕ) is the set of all moves along the information ordering $\sqsubseteq$ leading to a state where ϕ holds; the interpretation of $\mathsf{con}(\phi)$ (read: contract with ϕ) consists of all moves *backwards* along the ordering to states where ϕ *fails*. As usual, $\phi?$ is the 'test-for-ϕ' relation, while the intended interpretation of the operators left unexplained should be clear.[1]

4.2.2. Definition. The *models* for $\mathcal{DML}$ are structures of the form $\mathfrak{M} = (W, \sqsubseteq, [\![\cdot]\!], V)$, where $\sqsubseteq \subseteq W^2$ is transitive and reflexive (the *information ordering*), $[\![\cdot]\!] : Proc(\Phi) \to 2^{W \times W}$, and $V : \Phi \to 2^W$ is a *valuation* assigning subsets of W to proposition letters.[2] The interpretation of the projections is the following:

$$\begin{aligned} \mathfrak{M}, x \models \mathsf{do}(\alpha) &\quad \text{iff} \quad \exists y\, ((x,y) \in [\![\alpha]\!]), \\ \mathfrak{M}, x \models \mathsf{ra}(\alpha) &\quad \text{iff} \quad \exists y\, ((y,x) \in [\![\alpha]\!]), \\ \mathfrak{M}, x \models \mathsf{fix}(\alpha) &\quad \text{iff} \quad (x,x) \in [\![\alpha]\!]. \end{aligned}$$

A model $\mathfrak{M}$ is *standard* if it interprets the relational part of the language as follows:

[1] This terminology $\mathsf{exp}(\cdot)$ and $\mathsf{con}(\cdot)$ derives from one of the uses of $\mathcal{DML}$, viz. as a setting in which the basic operations studied in Theory Change, *expansions* and *contractions*, are modeled. See §4.3 for some details.

[2] A quick remark about the properties of $\sqsubseteq$. It seems a reasonable minimal requirement to let this abstract relation of information growth or change be a pre-order. Pre-orders have a long tradition as information structures, viz. their use as models for intuitionistic logic. Of the technical results presented below none hinges on $\sqsubseteq$ being a pre-order.

$$
\begin{aligned}
[\![\mathsf{exp}(\phi)]\!] &= \lambda xy.\,(x \sqsubseteq y \wedge \mathfrak{M}, y \models \phi),\\
[\![\mathsf{con}(\phi)]\!] &= \lambda xy.\,(x \sqsupseteq y \wedge \mathfrak{M}, y \not\models \phi),\\
[\![\alpha \cap \beta]\!] &= [\![\alpha]\!] \cap [\![\beta]\!],\\
[\![\alpha ; \beta]\!] &= [\![\alpha]\!] ; [\![\beta]\!],\\
[\![-\alpha]\!] &= -[\![\alpha]\!],\\
[\![\alpha^{\smallsmile}]\!] &= \{\, (x,y) : (y,x) \in [\![\alpha]\!] \,\},\\
[\![\phi ?]\!] &= \{\, (x,x) : \mathfrak{M}, x \models \phi \,\}.
\end{aligned}
$$

As usual, we say that a formula ϕ is a *consequence* of a set of formulas Δ if for every (standard) model $\mathfrak{M}$ and every x in $\mathfrak{M}$, $\mathfrak{M}, x \models \psi$, for all $\psi \in \Delta$, implies $\mathfrak{M}, x \models \phi$.

Observe that ra and fix are definable using the other operators; the contraction mode $\mathsf{con}(\cdot)$ is equivalent to $\mathsf{exp}(\neg\phi)^{\smallsmile}$. Whenever this is convenient we will assume that exp only has $\top$ as its argument; this is justified by the equivalence $[\![\mathsf{exp}(\phi)]\!] = [\![\mathsf{exp}(\top); \phi?]\!]$.

The original definition of $\mathcal{DML}$ as given in (Van Benthem 1989*a*) included the *minimal* projections μ-$\mathsf{exp}(\cdot)$ and μ-$\mathsf{con}(\cdot)$ whose definitions read

$$
\begin{aligned}
[\![\mu\text{-}\mathsf{exp}(\phi)]\!] &= \lambda xy.\Big(x \sqsubseteq y \wedge \mathfrak{M}, y \models \phi \wedge \neg\exists x\,(x \sqsubseteq z \sqsubset y \wedge \mathfrak{M}, z \models \phi)\Big), \quad \text{and}\\
[\![\mu\text{-}\mathsf{con}(\phi)]\!] &= \lambda xy.\Big(x \sqsupseteq y \wedge \mathfrak{M}, y \not\models \phi \wedge \neg\exists x\,(x \sqsupseteq z \sqsupset y \wedge \mathfrak{M}, z \not\models \phi)\Big).
\end{aligned}
$$

They have been left out because they are definable:

$$(x,y) \in [\![\mu\text{-}\mathsf{exp}(\phi)]\!] \ \text{ iff } \ (x,y) \in [\![\mathsf{exp}(\phi) \cap -\big(\mathsf{exp}(\phi); (\mathsf{exp}(\top) \cap -\top?)\big)]\!],$$

and similarly for μ-$\mathsf{con}(\phi)$.

There are obvious connections between $\mathcal{DML}$ and *Propositional Dynamic Logic* (**PDL**, (Harel 1984)). The 'old diamonds' $\langle\alpha\rangle$ from **PDL** can be simulated in $\mathcal{DML}$ by putting $\langle\alpha\rangle\phi := \mathsf{do}(\alpha; \phi?)$. And conversely, the expansion and contraction operators are definable in a particular mutation of **PDL** where taking converses of program relations is allowed and a name for the information ordering is available: $[\![\mathsf{exp}(\phi)]\!] = [\![\sqsubseteq; \phi?]\!]$. The domain operator $\mathsf{do}(\alpha)$ can be simulated in standard **PDL** by $\Diamond\top$. A difference between the two is that (standard) **PDL** only has the regular program operations $\cup$, ; and the Kleene star *, while $\mathcal{DML}$ has the full relational repertoire $\cup$, $-$, $^{\smallsmile}$ and ;, but not *. Another difference is not a technical difference, but one in emphasis: whereas in **PDL** the Boolean part of the language clearly is the primary component of the language and the main concern lies with the *effects* of programs, in $\mathcal{DML}$ one focuses on the *interaction* between the static and dynamic component.

A related formalism whose relational apparatus is more alike that of $\mathcal{DML}$ is the *Boolean Modal Logic* (**BML**) studied by Gargov & Passy (1990). This system has atomic relations $\rho_1, \rho_2, \ldots$, a constant for the universal relation ∇, and relation-forming operators $\cap$, $\cup$ and $-$. Relations are referred to within **BML** by means of the **PDL**-like diamonds $\langle\alpha\rangle$. Since the language of **BML** does *not* allow either ; or $^{\smallsmile}$ as operators on relations, it is a strict subset of $\mathcal{DML}(\Phi; \Omega)$.

4.3 USING $\mathcal{DML}$

THEORY CHANGE

One of the original motivations for the invention of $\mathcal{DML}$ was to obtain a formalism for reasoning about the cognitive moves an agent makes while searching for new knowledge or information; possible moves one should be able to formulate included acquiring new information, and giving up information. It later turned out that along similar lines $\mathcal{DML}$ can be used to model postulates for Theory Change. I will briefly sketch this.

Consider a set of beliefs or a knowledge set T. As our perception of the world as described by T changes, the knowledge set may have to be modified. In the literature on theory change a number of such modifications have been identified (Alchourrón, Gärdenfors & Makinson 1985, Katsuno & Mendelzon 1991), including expansions, contractions and revisions. If we acquire information that does not contradict T, we can simply *expand* our knowledge set with this piece of information. When a sentence ϕ previously believed becomes questionable and has to be abandoned, we *contract* our knowledge with ϕ. Somewhat intermediate between expansion and contraction is the operation of *revision*: the operation of resolving the conflict that arises when the newly acquired information contradicts our old beliefs. The revision of T by a sentence ϕ is often thought of as consisting of first making changes to T, so as to then be able to expand with ϕ. According to general wisdom on theory change, *as little as possible* of the old theory is to be given up in order to accommodate for newly acquired information.

Gärdenfors and others have proposed an influential set of rationality postulates that the revision operation must satisfy. By defining revision and expansion operators inside $\mathcal{DML}$ all of the postulates (except one) can be modeled inside $\mathcal{DML}$. We briefly sketch how this may be done. First, one represents theories T as nodes in a model, and statements of the form '$\phi \in T$' as modal formulas $[\sqsubseteq]\phi$ (i.e. $\neg\mathsf{do}(\mathsf{exp}(\top);\neg\phi?)$). Then, following the above maxim to change as little as possible of the old theory, one defines an expansion operator $[+\phi]\psi$ ('ψ belongs to every theory that results from expanding with ϕ') as

$$[+\phi]\psi := \neg\mathsf{do}\Big(\mu\text{-}\mathsf{exp}([\sqsubseteq]\phi);\neg[\sqsubseteq]\psi?\Big),$$

So, $[+\phi]\psi$ is true at a node x if in every minimal $\sqsubseteq$-successor y of x where $[\sqsubseteq]\phi$ holds (i.e. where ϕ has been added to the theory), the formula $[\sqsubseteq]\psi$ is true (i.e. ψ is in the theory). Next, one defines a revision operator $[*\phi]\psi$ ('ψ belongs to every theory resulting from revising by ϕ') by first minimally removing possible conflicts with ϕ, then minimally adding ϕ, and subsequently testing whether ψ belongs to the result:

$$[*\phi]\psi := \neg\mathsf{do}\Big(\big(\mu\text{-}\mathsf{con}([\sqsubseteq]\neg\phi);\mu\text{-}\mathsf{exp}([\sqsubseteq]\phi)\big);\neg[\sqsubseteq]\psi?\Big).$$

Given this modeling the Gärdenfors postulates can be translated into $\mathcal{DML}$. As an example we consider the 3rd postulate, also known as the inclusion postulate:

'the result of revising T by ϕ is included in the expansion of T with ϕ,' or $T * \phi \subseteq T + \phi$. Its translation reads: $[*\phi]\psi \to [+\phi]\psi$. It is easily verified that this translation is valid on all $\mathcal{DML}$-models. In fact, nearly all of Gärdenfors (1988)'s postulates for revision and contraction come out true in this modeling. The only one that fails is the 8th postulate, also known as 'conjunctive vacuity' (Fuhrmann 1990*b*); its failure is caused by the information ordering $\sqsubseteq$ in $\mathcal{DML}$-models not being a function. De Rijke (1992*a*) provides further details.

UPDATE SEMANTICS

Further formalisms to which $\mathcal{DML}$ has been linked include conditionals and other systems that somehow involve a notion of change. But, whereas the applications to Theory Change and conditionals do not require the states in $\mathcal{DML}$-models to have any particular structure, others do.

For example, one version of Frank Veltman's Update Semantics (Veltman 1991) may be seen as a formalism for reasoning about models of the modal system **S5** (where each **S5**-model represents a possible information state of a single agent) and certain transitions between such models. By imposing the structure of **S5**-models on the individual states in a $\mathcal{DML}$-model, the latter becomes a class of **S5**-models in which the $\mathcal{DML}$-apparatus can be used to reason about *global* transitions between **S5**-models, while the language of **S5** can be used to reason about the *local* structure of the **S5**-models. The global transitions can then be interpreted as various kinds of updates; (De Rijke 1992*a*) shows how Veltman's `might`-operator and sequential conjunction can be accounted for in this way. Furthermore, notions of consequence considered by Veltman for Update Semantics can be modeled using the $\mathcal{DML}$-apparatus (cf. page 54 for related issues).

DYNAMIC CONNECTIVES; DYNAMIC INFERENCE

Many of the dynamic operators that have been proposed in the literature can be defined in $\mathcal{DML}$. The underlying reason for this is that most dynamic proposals have some kind of two-dimensional structures in common as their underlying models, and that the operators considered are usually only concerned with certain pre- and postconditions of transitions in such structures — $\mathcal{DML}$ is strong enough to reason about the pre- and postconditions of all transitions defined by the standard operations on binary relations, and many more besides. For instance, the *residuals* of Vaughan Pratt's action logic (Pratt 1990*a*) can be defined in $\mathcal{DML}$:

$$\begin{array}{lclcl} \alpha \Rightarrow \beta &=& \{ (x,y) : \forall z\,((z,x) \in [\![\alpha]\!] \to (z,y) \in [\![\beta]\!]) \} &=& -(\alpha^{\smallsmile}; -\beta), \\ \alpha \Leftarrow \beta &=& \{ (x,y) : \forall z\,((y,z) \in [\![\alpha]\!] \to (x,z) \in [\![\beta]\!]) \} &=& -(-\beta; \alpha^{\smallsmile}). \end{array}$$

As pointed out in (Van Benthem 1991*b*) the *test negation* proposed in Groenendijk & Stokhof (1991) becomes

$$\sim \alpha \quad = \quad \{ (x,x) : \neg\exists y\,((x,y) \in [\![\alpha]\!] \} \quad = \quad \delta \cap -(\alpha; \top?).$$

A logical system is sometimes dubbed dynamic because it has dynamic connectives as in the above examples, and sometimes because it has a dynamic notion of inference. Quite often the latter can also be simulated in $\mathcal{DML}$. Here are some examples taken from (Van Benthem 1991*b*). The standard notion of inference $\models_1$ ("every state that models all of the premises, should also model the conclusion") may be represented as

$$\phi_1 \wedge \ldots \wedge \phi_n \models_1 \psi \quad \text{iff} \quad \mathsf{fix}(\phi_1?) \wedge \ldots \wedge \mathsf{fix}(\phi_n?) \rightarrow \mathsf{fix}(\psi?).$$

A more dynamic notion $\models_2$ taken from (Groenendijk & Stokhof 1991), which may be paraphrased as "process all premises consecutively, then you should be able to reach a state where the conclusion holds", has the following transcription in $\mathcal{DML}$:

$$\phi_1 \wedge \ldots \wedge \phi_n \models_2 \psi \quad \text{iff} \quad \mathsf{ra}\Big(\mathsf{exp}(\phi_1); \ldots ; \mathsf{exp}(\phi_n)\Big) \rightarrow \mathsf{do}(\mathsf{exp}(\psi)).$$

A third notion of inference, $\models_3$, found for example in Van Eijck & de Vries (1993) which reads "whenever it is possible to consecutively expand with all premises, then it should be possible to expand with the conclusion", can be given the following representation:

$$\phi_1 \wedge \ldots \wedge \phi_n \models_3 \psi \quad \text{iff} \quad \mathsf{do}\Big(\mathsf{exp}(\phi_1); \ldots ; \mathsf{exp}(\phi_n)\Big) \rightarrow \mathsf{do}(\mathsf{exp}(\psi)).$$

4.4 THE EXPRESSIVE POWER OF $\mathcal{DML}$

Recall from §§2.2, 3.4 that modal formulas in $\mathcal{ML}(\Diamond)$ become equivalent to a special kind of first-order formulas when interpreted on models. These first-order counterparts form a restricted 2-variable fragment of the full-first order language, one that can easily be described syntactically, and for which a semantic characterization can be given in terms of bisimulations. Likewise, the first-order transcriptions of modal formalisms used to reason about relation algebras live in a 3-variable fragment of the full first-order language; they too can be given precise syntactic and semantic descriptions.

The above two are special cases of a much more general phenomenon, namely the relation between patterns or important features of structures and bisimulations that precisely preserve these patterns on the one hand, and modal formalisms describing such patterns on the other hand (cf. Chapter 6 for more). In the present case of $\mathcal{DML}$ it is also possible to give a precise syntactic description of its first-order transcriptions, and using the general results of Chapter 6 an appropriate notion of bisimulation can be defined that allows a semantic characterization of these first-order transcriptions.

THE CONNECTION WITH FIRST-ORDER LOGIC

Recall that $\mathcal{DML}$-formulas are evaluated at points, not transitions; as in many other modal formalisms the relational compound is (still) treated as a second class citizen. This is reflected by the fact that $\mathcal{DML}$-formulas end up as first-order formulas in *one* free variable when translated into first-order logic. Roughly speaking, $\mathcal{DML}$-expressions correspond to a 3-variable fragment of first-order logic with up to two free variables; $\mathcal{DML}$-formulas are obtained from these by quantifying over one of the variables. The following makes this precise.

4.4.1. DEFINITION. Fix a vocabulary $\boldsymbol{\tau}$, and fix individual variables $x_1, x_2, \ldots$. For $m \in \mathbb{N}$, X a set of individual variables, $\mathcal{L}_m(X)$ denotes the set of first-order formulas which have at most m variables $x_1, \ldots, x_m$, and whose free variables are in X; $\mathcal{L}_m$ is the set of formulas having at most m variables $x_1, \ldots, x_m$.

4.4.2. DEFINITION. Let X be a set of (first-order) formulas, and let K be a class of models. Then, a modal language $\mathcal{ML}$ is called *expressively complete for X over* K if for all $\chi \in X$ there is an $\mathcal{ML}$-formula ϕ such that $\mathsf{K} \models ST(\phi) \leftrightarrow \chi$. If K is the class of all models the clause 'over K' will be suppressed.

Below we will show that $\mathcal{DML}$ is expressively complete for $\mathcal{L}_3(x_1)$.

The standard translation $ST(\cdot)$ taking modal formulas to first-order ones can easily be adapted to the full $\mathcal{DML}$. However, whereas standard modal formulas translate into formulas having one free variable in a two-variable fragment, formulas in $\mathcal{DML}$ translate into first-order formulas of a three-variable fragment that contain one free variable. (The reader should compare these results to analogous links between algebraic logics of relation and cylindric algebras, and first-order logic, Tarksi & Givant (1987).)

Instead of $\sqsubseteq$ I will use an abstract binary relation symbol R to translate the modal operators and the 'dynamic' constructs.

4.4.3. DEFINITION. Let $\boldsymbol{\tau}$ be the (first-order) vocabulary $\{R, P_1, P_2, \ldots\}$, with R a binary relation symbol, and the P_i's unary relation symbols corresponding to the proposition letters $p_i \in \Phi$. Let $\mathcal{L}(\boldsymbol{\tau})$ be the set of all first-order formulas over $\boldsymbol{\tau}$ (with identity).

Define a translation $ST(\cdot)$ taking $\mathcal{DML}$-formulas to formulas in $\mathcal{L}(\boldsymbol{\tau})$ as in Table 4.1.

$$
\begin{array}{rcl@{\qquad}rcl}
ST(\top) & = & (x_1 = x_1) & ST(p) & = & P(x_1)\\
ST(\neg\phi) & = & \neg ST(\phi) & ST(\phi \wedge \psi) & = & ST(\phi) \wedge ST(\psi)\\
\multicolumn{6}{c}{ST(\mathsf{do}(\alpha)) = \exists x_2\, ST(\alpha)}\\
\hline
ST(-\alpha) & = & \neg ST(\alpha) & ST(\alpha \cap \beta) & = & ST(\alpha) \wedge ST(\beta)\\
ST(\alpha^{\smile}) & = & [x_2/x_1, x_1/x_2]ST(\alpha) & ST(\phi?) & = & (x_1 = x_2) \wedge ST(\phi).\\
\multicolumn{6}{c}{ST(\mathsf{exp}(\phi)) = (Rx_1x_2) \wedge [x_2/x_1]ST(\phi)}\\
\multicolumn{6}{c}{ST(\alpha;\beta) = \exists x_3\,([x_3/x_2]ST(\alpha) \wedge [x_3/x_1]ST(\beta))}
\end{array}
$$

Table 4.1: The standard translation for $\mathcal{DML}$.

Observe that the translations of $\mathcal{DML}$-formulas live in $\mathcal{L}_3(x_1)$, and that the translations of $\mathcal{DML}$-expressions as a whole live in $\mathcal{L}_3(x_1, x_2)$.

4.4.4. PROPOSITION. *Let ϕ be a formula in $\mathcal{DML}(\Phi)$. For any $\mathfrak{M}$, and any x in $\mathfrak{M}$, $\mathfrak{M}, x \models \phi$ iff $\mathfrak{M} \models ST(\phi)[x]$.*

Proof. By induction on $\mathcal{DML}$-expressions θ one shows that for any $\mathfrak{M}$, and for any x, y in $\mathfrak{M}$, $\mathfrak{M}, x \models \theta$ iff $\mathfrak{M} \models ST(\theta)[x]$, if $\theta \in Form(\Phi)$, and $(x, y) \in [\![\theta]\!]_{\mathfrak{M}}$ iff $\mathfrak{M} \models ST(\theta)[x, y]$, in case $\theta \in Proc(\Phi)$. $\dashv$

To show that, conversely, every $\mathcal{L}(x_1)$-formula is equivalent to a $\mathcal{DML}$-formula, we need an intermediate fragment.

4.4.5. DEFINITION. (Cf. (Venema 1991, Definition 3.3.12)) Fix individual variables x_1, x_2, x_3, and let $\tau = \{ R, P_1, P_2, \ldots \}$ be as before. Assume $\{ i, j, k \} = \{ 1, 2, 3 \}$.

We define fragments of $\mathcal{L}_3(x_i, x_j)$ and $\mathcal{L}_3(x_i)$. $\mathcal{L}_3(x_i, x_j)^-$ contains the same atomic formulas as $\mathcal{L}_3(x_i, x_j)$, it is closed under $\wedge$, $\neg$, $\exists x_j$, and if α is in $\mathcal{L}_3(x_i, x_j)^-$ and β is in $\mathcal{L}_3(x_j, x_k)^-$, then $\exists x_k\,(\alpha \wedge \beta)$ is in $\mathcal{L}_3(x_i, x_j)^-$.

The fragment $\mathcal{L}_3(x_i)^-$ is obtained from $\mathcal{L}_3(x_i, x_j)^-$ by prefixing every formula in $\mathcal{L}_3(x_i, x_j)^-$ with $\exists x_j$. (There is an obvious, but harmless sloppiness to this definition; but $\mathcal{L}_3(x_i)^-$ as a fragment of $\mathcal{L}_3(x_i, x_j)^-$ is isomorphic to $\mathcal{L}_3(x_i)^-$ as a fragment of $\mathcal{L}_3(x_i, x_k)^-$, etc.)

4.4.6. PROPOSITION. *Every formula in $\mathcal{L}_3$ is equivalent to a Boolean combination of formulas in $\mathcal{L}_3^{2-}$ with the same free variables. (Here $\mathcal{L}_3^{2-}$ is simply the union of the 3 $\mathcal{L}_3(x_i, x_j)^-$ fragments from 4.4.5.)*

Proof. This is by induction on $\mathcal{L}_3$. The only (mildly) interesting case is if α in $\mathcal{L}_3$ is of the form $\exists x_3\,\beta$. As α is in $\mathcal{L}_3$, so is β. By induction hypothesis β is equivalent to a Boolean combination of $\mathcal{L}_3^{2-}$-formulas. By using results on disjunction normal forms, we may assume that β is equivalent to a conjunction $\alpha_{12} \wedge \alpha_{13} \wedge \alpha_{23}$, where α_{ij} is in $\mathcal{L}_3(x_i, x_j)^-$. Then α is equivalent to $\alpha' = \alpha_{12} \wedge \exists x_3\,(\alpha_{13} \wedge \alpha_{23})$ — this is a Boolean combination of two $\mathcal{L}_3^{2-}$-formulas. Observe that α' has the same free variables as α. $\dashv$

4.4.7. DEFINITION. We now translate the $\mathcal{L}_3(x_i, x_j)^-$ fragments into expressions of $\mathcal{DML}$. As before, let δ abbreviate $\top?$; let $\mathbf{1}$ abbreviate $(\delta \cup -\delta)$.

$$\begin{array}{rcl@{\qquad}rcl}
(Px_i)^{ij} & = & p & (Px_j)^{ij} & = & p \\
(Rx_ix_j)^{ij} & = & \mathsf{exp}(\top) & (Rx_jx_i)^{ij} & = & \mathsf{exp}(\top)^{\smallsmile} \\
(Rx_ix_i)^{ij} & = & \mathsf{exp}(\top) \cap \delta & (Rx_jx_j)^{ij} & = & (\mathbf{1}; (\mathsf{exp}(\top) \cap \delta)) \\
(x_i = x_j)^{ij} & = & \delta & (x_j = x_i)^{ij} & = & \delta \\
(x_i = x_i)^{ij} & = & \mathbf{1} & (x_j = x_j)^{ij} & = & \mathbf{1} \\
(\neg\alpha(x_i))^{ij} & = & \neg\alpha(x_i)^{ij} & (\neg\alpha(x_j))^{ij} & = & \neg\alpha(x_j)^{ij}
\end{array}$$

$$\begin{array}{rcl}
(\neg\alpha(x_i, x_j))^{ij} & = & -\alpha(x_i, x_j)^{ij} \\
(\alpha(x_i) \wedge \beta(x_i))^{ij} & = & \alpha(x_i)^{ij} \wedge \beta(x_i)^{ij} \\
(\alpha(x_j) \wedge \beta(x_j))^{ij} & = & \alpha(x_j)^{ij} \wedge \beta(x_j)^{ij} \\
(\alpha(x_i) \wedge \beta(x_j))^{ij} & = & (\alpha(x_i)^{ij}?; \mathbf{1}) \cap (\mathbf{1}; \beta(x_j)^{ij}?)
\end{array}$$

$$
\begin{array}{rcl}
(\alpha(x_i, x_j) \wedge \beta(x_i))^{ij} & = & \alpha(x_i, x_j)^{ij} \cap (\beta(x_i)^{ij}?; \mathbf{1}) \\
(\alpha(x_i, x_j) \wedge \beta(x_j))^{ij} & = & \alpha(x_i, x_j)^{ij} \cap (\mathbf{1}; \beta(x_j)^{ij}?) \\
(\alpha(x_i, x_j) \wedge \beta(x_i, x_j))^{ij} & = & \alpha(x_i, x_j)^{ij} \cap \beta(x_i, x_j)^{ij} \\
(\exists x_j\, \alpha(x_i))^{ij} & = & \alpha(x_i)^{ij} \\
(\exists x_j\, \alpha(x_j))^{ij} & = & E(\alpha(x_j)^{ij}) \\
(\exists x_j\, \alpha(x_i, x_j))^{ij} & = & \mathsf{do}(\alpha(x_i, x_j)^{ij}) \\
(\exists x_k\, (\alpha(x_i, x_k) \wedge \beta(x_k, x_j)))^{ij} & = & \alpha(x_i, x_k)^{ik}; \beta(x_k, x_j)^{kj}.
\end{array}
$$

4.4.8. PROPOSITION. *Let α be a formula in $\mathcal{L}_3(x_i, x_j)^-$. Then*

1. *if $\alpha \equiv \alpha(x_i)$ or $\alpha \equiv \alpha(x_j)$, then $\mathfrak{M} \models \alpha[a]$ iff $\mathfrak{M}, a \models \alpha^{ij}$,*
2. *if $\alpha \equiv \alpha(x_i, x_j)$, then $\mathfrak{M} \models \alpha[ab]$ iff $(a, b) \in [\![\alpha^{ij}]\!]_{\mathfrak{M}}$.*

4.4.9. THEOREM. *$\mathcal{DML}$ is expressively complete for $\mathcal{L}_3(x_1)$: every first-order formula in $\mathcal{L}_3(x_1)$ is equivalent to a $\mathcal{DML}$-formula.*

Proof. By Proposition 4.4.6 $\alpha(x_1)$ has an equivalent $\alpha'(x_1)$ in $\mathcal{L}_3(x_1)^-$. So by Proposition 4.4.8 $\alpha'(x_1)$, and therefore $\alpha(x_1)$, is equivalent to the $\mathcal{DML}$-formula $\alpha'(x_1)^{12}$. $\dashv$

What about expressive completeness of $\mathcal{DML}$ with respect to the full first-order language? It is known that no modal language with finitely many modal operators is expressively complete for full first-order logic over *all* structures (cf. the Appendix). Over restricted classes of structures, however, expressive completeness results do exist. By a result of Immerman and Kozen, over linear orders every first-order formula $\alpha(x)$ is equivalent to a formula in $\mathcal{L}_3(x)$. Hence, by Theorem 4.4.9 we have the following:

4.4.10. COROLLARY. *$\mathcal{DML}$ is expressively complete for first-order logic over linear orders.*

An alternative way to derive this result is to show that $\mathcal{DML}$ is at least as expressive as systems of modal logic of which it is know that they are expressively complete over some class of linear orders. Let me illustrate this with two examples. The first one of which involves the temporal operator *Until* whose pattern reads:

$$
\mathfrak{M}, x \models \mathit{Until}(p, q) \text{ iff } \exists y \Big(Rxy \wedge Py \wedge \neg\exists z\, (Rxz \wedge Rzy \wedge z \neq y \wedge \neg Qz) \Big).
$$

In $\mathcal{DML}$ this operator can be defined by $\mathsf{do}\big(\mathsf{exp}(p) \cap -(\mathsf{exp}(\neg q); (\mathsf{exp}(\top) \cap -\delta))\big)$. (And similarly for *Since*, the backward-looking version of *Until*.) According to Kamp (1968)'s Theorem, the *Until*, *Since*-language is expressively complete for the full first-order language over all continuous linear orders. Hence, so is $\mathcal{DML}$.

Jonathan Stavi has improved Kamp's Theorem by defining the Stavi connectives *Until'* and *Since'*, and proving those two operators to be expressively complete for first-order logic over *all* linear orders, cf. (Gabbay, Hodkinson & Reynolds 1993). Here, $\mathit{Until}'(p, q)$ is defined by

$$
\exists y \Big(Rxy \wedge \forall z\, (Rxz \wedge Rzy \rightarrow Qz) \Big) \wedge \tag{4.1}
$$

$$\forall y\,\Big(Rxy \wedge \forall z\,(Rxz \wedge Rzy \to Qz) \to$$
$$\Big(Qy \wedge \exists x\,(Ryx \wedge \forall z\,(Ryz \wedge Rzx \to Qz))\Big)\Big) \wedge \tag{4.2}$$
$$\exists y\,\Big(Rxy \wedge \neg Qy \wedge Py \wedge \forall z\,(Rxz \wedge Rzy \wedge \exists y\,(Rxy \wedge Ryz \wedge \neg Qy) \to Pz)\Big), \tag{4.3}$$

and $Since'(\cdot,\cdot)$ is its 'backward-looking' version. By Theorem 4.4.9 *Until'* and *Since'* are definable in $\mathcal{DML}$. Here's an explicit definition; writing R for $\mathsf{exp}(\top)$ we can define the operator $Until'(p,q)$ in $\mathcal{DML}$ as follows:

$$\mathsf{do}\Big(R \cap -(\mathsf{exp}(\neg q); R)\Big) \wedge \tag{4.4}$$
$$\neg\mathsf{do}\Big(R \cap -(\mathsf{exp}(\neg q); R) \cap -\Big(\mathsf{exp}(q) \cap \mathsf{do}\Big(R \cap -(\mathsf{exp}(\neg q); R)\Big)?\Big)\Big) \wedge \tag{4.5}$$
$$\mathsf{do}\Big(\mathsf{exp}(\neg q \wedge p) \cap -\Big(\Big(\mathsf{exp}(\neg p) \cap (\mathsf{exp}(\neg q); R)\Big); R\Big)\Big). \tag{4.6}$$

It is easily verified that (4.1), (4.2) and (4.3) are defined by the $\mathcal{DML}$-formulas (4.4), (4.5) and (4.6), respectively.

Definability issues

As in §3.4 we want to use an appropriate notion of bisimulation to characterize definability of classes of models in $\mathcal{DML}$. According to the perspective of Chapter 6 when looking for such a notion, one should locate the relevant relational patterns for the modal language, and have candidate bisimulations respect these patterns. With $\mathcal{ML}(\Diamond)$ and $\mathcal{ML}(D)$ the relevant patterns were certain configurations involving R or $\neq$ which could be decomposed into simple R-transitions and '$\neq$-transitions,' respectively (cf. 3.4.1 and 3.4.4). In $\mathcal{DML}$ far more complex patterns are relevant; by the analysis of Theorem 4.4.9 we have to take all $\mathcal{L}_3(x_1,x_2)$-definable patterns into account, and there does not seem to be an obvious way to decompose such patterns into simpler ones in such a way that any candidate bisimulation that respects the simpler patterns, also respects arbitrary $\mathcal{L}_3(x_1,x_2)$-definable patterns. This suggests the following definition.

4.4.11. Definition. Let $\mathfrak{A}_1 = (W_1, \sqsubseteq_1, [\![\cdot]\!]_1, V_1)$ and $\mathfrak{A}_2 = (W_2, \sqsubseteq_2, [\![\cdot]\!]_2, V_2)$ be two $\mathcal{DML}$-models; we are not assuming here that $\sqsubseteq$ is a pre-order.

A non-empty relation Z between $\mathfrak{A}_1$ and $\mathfrak{A}_2$ is a *$\mathcal{DML}$-bisimulation* if the following holds:

1. Zwv implies that w, v satisfy the same proposition letters,
2. if Zwv, $w' \in W_1$ and $\mathfrak{A}_1 \models \alpha[ww']$, for $\alpha \in \mathcal{L}_3(x_1,x_2)$, then for some $v' \in W_2$, $Zw'v'$ and $\mathfrak{A}_2 \models \alpha[vv']$,
3. if Zwv, $v' \in W_2$ and $\mathfrak{A}_2 \models \alpha[vv']$, for $\alpha \in \mathcal{L}_3(x_1,x_2)$, then for some $w' \in W_1$, $Zw'v'$ and $\mathfrak{A}_1 \models \alpha[ww']$.

4.4.12. Remark. Two comments are in order. First, the back-and-forth conditions 2 and 3 in Definition 4.4.11 are rather linguistic; for an abstract relation relation between models that is meant to characterize $\mathcal{DML}$-equivalence, this is ugly. However, no natural non-linguistic alternative linking only single states is known to me. Chapter 5 below contains an algebraic description of the fragment $\mathcal{L}_3(x_1, x_2)$ in terms of so-called 2-partial isomorphisms linking sequences of objects of length at most 2; a characterization of the first-order counterpart of $\mathcal{DML}$ may be derived from this (Theorem 5.6.7).

Second, I conjecture that the back-and-forth conditions 2 and 3 have no elegant decomposition into simpler back-and-forth conditions as in the R-bisimulations and $\neq$-bisimulations of Chapter 3.

4.4.13. EXAMPLE. Despite the strength of $\mathcal{DML}$, on the class of finite models $\mathcal{DML}$-bisimilarity and isomorphism do not coincide. Consider $\mathfrak{A}$, $\mathfrak{B}$ in Figure 4.2 below where all states have the same valuation. Clearly $\mathfrak{A}$, $\mathfrak{B}$ are not isomorphic,

Figure 4.2: Bisimilar but not isomorphic.

but putting $Z = A \times B$ defines a $\mathcal{DML}$-bisimulation between $\mathfrak{A}$ and $\mathfrak{B}$. (To see this, it comes in handy to observe that both in $\mathfrak{A}$ and $\mathfrak{B}$ any formula is true in all or in no points, and any procedure coincides with $\emptyset$, ∇, δ, or $-\delta$.)

4.4.14. EXAMPLE. Given two finite models $\mathfrak{A}$, $\mathfrak{B}$ with $x \in \mathfrak{A}$, $y \in \mathfrak{B}$ such that for all $\mathcal{DML}$-formulas ϕ, $\mathfrak{A}, x \models \phi$ iff $\mathfrak{B}, y \models \phi$, one may define a $\mathcal{DML}$-bisimulation between $\mathfrak{A}$ and $\mathfrak{B}$ that connects x and y, by putting Zuv iff for all $\phi \in \mathcal{DML}(\Phi)$, $\mathfrak{A}, u \models \phi$ iff $\mathfrak{B}, v \models \phi$. It follows that two finite $\mathcal{DML}$-models are bisimilar iff they satisfy the same $\mathcal{DML}$-formulas.

Recall that a formula is invariant for bisimulations of it cannot distinguish between bisimilar models (Definition 3.4.2).

4.4.15. PROPOSITION. *$\mathcal{DML}$-formulas are invariant for bisimulations.*

This invariance characterizes the $\mathcal{DML}$-formulas as a first-order fragment.

4.4.16. THEOREM. *Let $\boldsymbol{\tau}$ be the first-order vocabulary with a binary relation symbol R and unary relation symbols P_1, P_2, ... A first-order formula $\alpha(x)$ in $\mathcal{L}(\boldsymbol{\tau})$ is (equivalent to) the translation of a $\mathcal{DML}$-formula iff it is invariant for $\mathcal{DML}$-bisimulations.*

Proof. See Example 6.7.4. $\dashv$

Using $\mathcal{DML}$-bisimulations further definability results can be derived using the general results of Chapter 6; here we will mention just one. For an elegant formulation of the result it is convenient to consider so-called pointed models as our fundamental structures. Here, a *pointed* model is a structure of the form

$(W, \sqsubseteq, [\![\cdot]\!], V, w)$, where $(W, \sqsubseteq, [\![\cdot]\!], V)$ is an ordinary $\mathcal{DML}$-model (in which $\sqsubseteq$ may or may not be a pre-order), and $w \in W$.

4.4.17. THEOREM. *Let* K *be a class of pointed models. Then* K *is definable by a* $\mathcal{DML}$*-formula iff it is closed under* $\mathcal{DML}$*-bisimulations and ultraproducts, while its complement is closed under ultraproducts.*

Proof. Again, see Example 6.7.4. $\dashv$

4.5 DECIDABILITY

The preceding sections contain ample demonstrations of the large expressive power of $\mathcal{DML}$. The main result of this section gives further evidence of the computational power of the language: satisfiability in $\mathcal{DML}$ is not decidable. We also show that decidability may be restored either by restricting the language, or by restricting or liberalizing the class of structures used to interpret it.

THE FULL LANGUAGE INTERPRETED ON PRE-ORDERS

Our language $\mathcal{DML}$ is somewhere in between the language of Boolean Modal Logic (**BML**) and full relation algebra. It is well-known that the latter is undecidable. Since in the intermediate case of $\mathcal{DML}$ we only have the operations of relation algebra on top of a single relation, it may be hoped that we are closer to **BML** than to relation algebra, and hence that the satisfiability problem for $\mathcal{DML}$ is decidable. But here is an important difference between the two. By Gargov et al. (1987) **BML** enjoys the finite model property, while $\mathcal{DML}$ does not. To see this, define

- $R := \mathsf{exp}(\top)$,
- $\infty := \neg E\mathsf{do}((R \cap -\delta) \cap R^{\smile})$.

Then, since ∞ forces the absence of loops, the formula $A\mathsf{do}(R \cap -\delta) \wedge \infty$ is satisfiable only on infinite models for $\mathcal{DML}$. In fact we have the following result:

4.5.1. THEOREM. *The satisfiability problem for* $\mathcal{DML}$ *is* Π^0_1*-hard.*

Proof. This is a reduction of a known Π^0_1-complete problem, a so-called *unbounded tiling problem* (UTP), to satisfiability in $\mathcal{DML}$. The version of the UTP that I will use here is given by the following data. Given a set of tiles $T = \{ d_0, \ldots, d_m \}$, each having 4 sides whose colors are in $C = \{ c_0, \ldots, c_k \}$, is there a tiling of $\mathbb{N} \times \mathbb{N}$? The rules of the tiling game are

1. every point in the grid is associated with a single tile,
2. adjacent edges have the same color.

The version of the UTP presented here is known to be Π^0_1-complete (Robinson 1971, Harel 1983). So to prove the theorem it suffices to define, for a given set of tiles T, a formula ϕ_T in $\mathcal{DML}$ such that

1. its models look like grids,

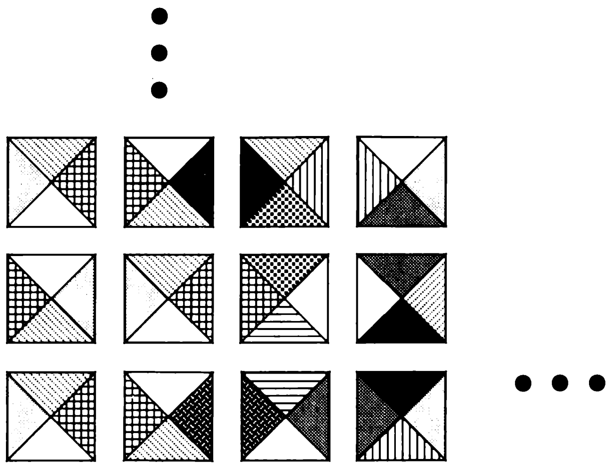


Figure 4.3: An unbounded tiling.

2. it says that every point is covered by a tile from T,
3. and that colors match right and above neighbours,

and show that ϕ_T is satisfiable iff T can tile $\mathbb{N} \times \mathbb{N}$.

To make a grid, define

- LEAVE$(\phi) := (\phi?; R)$,
- ONE $:= (R \cap -\delta) \cap -[(R \cap -\delta); (R \cap -\delta))$; then, for all $\mathfrak{M}$, and for all $x, y \in \mathfrak{M}$,

$$(x, y) \in [\![\text{ONE}]\!]_{\mathfrak{M}} \text{ iff } Rxy \wedge x \neq y \wedge \neg\exists z\,(Rxz \wedge x \neq z \wedge Rzy \wedge z \neq y),$$

- UP $:= (\text{ONE} \cap \text{LEAVE}(p \wedge q) \cap \mathsf{exp}(p \wedge \neg q))$
  $\cup\, (\text{ONE} \cap \text{LEAVE}(p \wedge \neg q) \cap \mathsf{exp}(p \wedge q))$
  $\cup\, (\text{ONE} \cap \text{LEAVE}(\neg p \wedge q) \cap \mathsf{exp}(\neg p \wedge \neg q))$
  $\cup\, (\text{ONE} \cap \text{LEAVE}(\neg p \wedge \neg q) \cap \mathsf{exp}(\neg p \wedge q))$,
- RIGHT $:= (\text{ONE} \cap \text{LEAVE}(p \wedge q) \cap \mathsf{exp}(\neg p \wedge q))$
  $\cup\, (\text{ONE} \cap \text{LEAVE}(\neg p \wedge q) \cap \mathsf{exp}(p \wedge q))$
  $\cup\, (\text{ONE} \cap \text{LEAVE}(p \wedge \neg q) \cap \mathsf{exp}(\neg p \wedge \neg q))$
  $\cup\, (\text{ONE} \cap \text{LEAVE}(\neg p \wedge \neg q) \cap \mathsf{exp}(p \wedge \neg q))$,
- EQUAL$(\alpha, \beta) := \neg E\mathsf{do}(\alpha \cap -\beta) \wedge \neg E\mathsf{do}(\beta \cap -\alpha)$,
- CR $:= \text{EQUAL}\Big((\text{UP}; \text{RIGHT}), (\text{RIGHT}; \text{UP})\Big)$.

Here, finally, is the formula that will force our models to contain a copy of $\mathbb{N} \times \mathbb{N}$:

- GRID $:= (p \wedge q) \wedge A\mathsf{do}(\text{UP}) \wedge A\mathsf{do}(\text{RIGHT}) \wedge \text{CR} \wedge \infty$.

Next we have to define formulas that force items 2 and 3. Let $T = \{ d_0, \ldots, d_m \}$ and $C = \{ c_0, \ldots, c_k \}$ be given. For each color $c_i \in C$ introduce four proposition letters, suggestively denoted by $(up = c_i)$, $(right = c_i)$, $(down = c_i)$, and $(left = c_i)$. Identifying each tile $d \in T$ with its four sides I assume that each tile d is represented as

$$\Big((up = c_{i_1}) \wedge (right = c_{i_2}) \wedge (down = c_{i_3}) \wedge (left = c_{i_4})\Big) \wedge$$
$$\Big(\bigwedge_{c \in T \setminus \{c_{i_1}\}} \neg(up = c) \wedge \ldots \wedge \bigwedge_{c \in T \setminus \{c_{i_4}\}} \neg(left = c)\Big).$$

Abbreviating $\neg\mathsf{do}(\alpha; \neg\phi?)$ as $[\alpha]\phi$, we put

– COVER $:= A\Big(\bigvee_{d \in T} d\Big) \wedge A\Big(\bigwedge_{\{d \neq e \in T\}} \neg(d \wedge e)\Big)$,

– MATCH $:=$

$$A\Big(\bigwedge_{c \in C}((up = c) \to [\mathrm{UP}](down = c)) \wedge$$
$$\bigwedge_{c \in C}((right = c) \to [\mathrm{RIGHT}](left = c))\Big).$$

Put $\phi_T :=$ GRID $\wedge$ COVER $\wedge$ MATCH. Then ϕ_T is satisfiable in a $\mathcal{DML}$-model iff T can tile $\mathbb{N} \times \mathbb{N}$. The *if*-direction is trivial, since if a tiling exists ϕ_T is satisfiable in $\mathbb{N} \times \mathbb{N}$, simply by verifying $(p \wedge q)$ in $(0, 0)$, switching the truth values of p and q while going right and up through the grid, respectively, while the tiling will tell you how to satisfy COVER and MATCH.

Conversely, assume $\mathfrak{M}, x \models \phi_T$. Let $f : \mathbb{N} \times \mathbb{N} \to \mathfrak{M}$ be such that $f(0,0) = x$, $(f(n,m), f(n,m+1)) \in [\![\mathrm{UP}]\!]_{\mathfrak{M}}$, and $(f(n,m), f(n+1,m)) \in [\![\mathrm{RIGHT}]\!]_{\mathfrak{M}}$; such f exist as $x \models$ GRID. Define a tiling on $\mathbb{N} \times \mathbb{N}$ by putting a tile $d \in T$ on (n,m) iff $\mathfrak{M}, f(n,m) \models d$. This tiling is well-defined and total by COVER. Moreover, if d is associated with with (n,m), and e is associated with $(n, m+1)$, then the up side of d and the down side of e must have the same colour by MATCH. Similarly, MATCH ensures that colours match right and left-hand sides. Hence T tiles $\mathbb{N} \times \mathbb{N}$. $\dashv$

4.5.2. COROLLARY. *The satisfiability problem for $\mathcal{DML}$ is Π^0_1-complete.*

4.5.3. REMARK. A few remarks are in order. The undecidability result for $\mathcal{DML}$ may seem to depend on the transitivity of the models for the language, as the formula ∞ only does its work properly on transitive structures. However, this dependency can be avoided; it suffices to have structures satisfying a Church-Rosser like property like $\forall yz\,(Rxy \wedge Rxz \to \exists u\,(Ryu \wedge Rzu))$, while always being able to move UP and RIGHT. In particular it follows that undecidability may already be found in the 'forward looking' $\breve{}$-free part of our language.

Also, having the relation $-\delta$ around, one can do without further complementation of relations by using μ-$\mathsf{exp}(\cdot)$ in the definition of ONE and redefining EQUAL as $\neg E\mathsf{do}((\alpha; -\delta) \cap \beta) \wedge \neg E\mathsf{do}(\alpha \cap (\beta; -\delta))$.

FRAGMENTS AND OTHER MODEL CLASSES

There are three obvious ways to escape from the undecidability result 4.5.1: by restricting the language, or by liberalizing or restricting the classes of relevant models.

To try and find reasonably large fragments of $\mathcal{DML}$ that are decidable, let us reconsider what made the proof of 4.5.1 work. Essentially, we were able to build a grid there, thanks to the availability of ;, $\cap$ and $-$. Thus, when looking for reasonably large *decidable* fragments of $\mathcal{DML}$, giving up some of these three might get us results. Indeed, giving up ; (and $\check{}$, by the way) restricts $\mathcal{DML}$ to the decidable Boolean Modal Logic. Alternatively, giving up $-$ and $\check{}$ yields decidability by Danecki (1985). (It seems that Danecki's proof can be extended to deal with $\check{}$ as well.) Of course, in these fragments operators like D and μ-$\mathsf{exp}(\cdot)$, μ-$\mathsf{con}(\cdot)$ need no longer be definable, so it remains to be seen whether adding any of these to the above fragments preserves decidability.

Another approach towards obtaining decidability is not to restrict the language, but to restrict the structures used to interpret it. As an example I will consider the class of all trees. By a *tree* is meant a structure $(W, \sqsubseteq)$ with $\sqsubset \subseteq W^2$ a transitive, asymmetric relation such that for each $x \in W$ the set of $\sqsubseteq$-predecessors of x is linearly ordered by $\sqsubseteq$. Let $\mathrm{Th}_{\mathcal{DML}}(\mathsf{TREES})$ denote the set of $\mathcal{DML}$-formulas valid on the class TREES of all trees. Observe that $\mathrm{Th}_{\mathcal{DML}}(\mathsf{TREES})$ lacks the finite model property. (To see this consider the earlier formula $A\mathsf{do}(R) \wedge \infty$.) Therefore, to establish decidability of this theory some other tools have to be employed. One obvious candidate is Rabin's Theorem (Rabin 1969); to apply this result the semantics of $\mathrm{Th}_{\mathcal{DML}}(\mathsf{TREES})$ has to be embedded in $S\omega S$, the monadic second-order theory of infinitely many successor functions.

Here, I will take an easier way out by appealing to a result by Gurevich & Shelah (1985). Let $\mathcal{L}_{GS}$ be the language of monadic second-order logic with additional unary predicates, that is, it has individual variables and unary predicate variables (ranging over branches) as well as a binary relation symbol $<$ and unary predicate constants $P_0, P_1, \ldots$. Let $\mathrm{Th}_{GS}(\mathsf{TREES})$ be the set of $\mathcal{L}_{GS}$ formulas valid on all trees. Then obviously, the question whether a given $\mathcal{DML}$ formula ϕ is valid on all trees, boils down to the question whether its standard translation $ST(\phi)$ is a theorem of $\mathrm{Th}_{GS}(\mathsf{TREES})$. But by (Gurevich & Shelah 1985) the latter question is decidable.

4.5.4. THEOREM. *The satisfiability problem for* $\mathrm{Th}_{\mathcal{DML}}(\mathsf{TREES})$ *is decidable.*

Several natural variations on the above still yield decidable theories. They include the set of $\mathcal{DML}$-formulas valid on all trees of finite depth, and the $\mathcal{DML}$-formulas valid on all well-founded trees.

A third way of avoiding the undecidability result Theorem 4.5.1 is to liberalize the class of structures our language deals with. This is what happens in systems of *Arrow Logic*. In one particular system designed by Van Benthem (1992) one has two sorts of objects, *states* and *arrows*, plus systematic connections between the two sorts, in much the same way as in $\mathcal{DML}$. Arrows are abstract versions of ordered pairs that are not necessarily equated with the latter. With this abstraction one regains decidability; even adding the Kleene star does not destroy decidability (Van Benthem 1992, Section 4).

4.6 Completeness

We will first outline how the completeness proofs of Chapter 3 can be used in the present setting. Then this construction is applied to obtain a complete axiomatization of the valid $\mathcal{DML}$-formulas.

How to use the D-operator

We extend the language $\mathcal{DML}$ with the D-operator from Chapter 3 to obtain the language $\mathcal{DML}^+$. Completeness is initially proved for a derivation system in this enriched language, and then transferred to the old language.

Completeness in $\mathcal{DML}^+$ is established using the general techniques of Chapter 3. The construction has to be *localized* for the present setting in the following way. Recall from the discussion following Theorem **3.3.29**, that in order to apply the completeness construction involving the irreflexivity rule (IR_D), we needed to have a Pasting Lemma (3.3.7), and in order to prove that, a Switching Lemma of the form $\vdash \#\phi \to \psi$ iff $\vdash \phi \to \overline{\#}^{\smile}\psi$ was essential; it was also observed that the latter came 'for free' in modal languages in which every (unary) operator came with a converse. We will use the modal operators D and $\mathsf{do}(\alpha; \cdot ?)$, for α a procedure in $\mathcal{DML}$, as input for the construction. The converse of the latter is $\mathsf{do}(\alpha^{\smile}; \cdot ?)$; D is its own converse. Also, what was needed to arrive at the completeness results for $\mathbf{DL}_m$ and $\mathbf{DL}_t$ was the presence of *inclusion axioms* stating that any binary relation is included in ∇, the universal relation; in effect, the inclusion allowed us to generate along the relation R_D in the provisional canonical model to arrive at the final canonical model without destroying the Truth Lemma. We add inclusion axioms for each of our modal operators $\mathsf{do}(\alpha; \cdot ?)$.

With these modifications we can define a Canonical Model for a dynamic modal logic **DML**, and establish a Pasting, Successor, and Structure Lemma, and, finally, a Completeness Theorem for **DML**.

Axioms

4.6.1. Definition. We define the dynamic modal language $\mathcal{DML}^+(\Phi)$ to be the language $\mathcal{DML}(\Phi)$ plus the difference operator D from Chapter **3**.

4.6.2. Definition. We define the derivation system $\mathbf{DML}^+$ in the language $\mathcal{DML}^+$. (From this definition notions like *derivation in* $\mathbf{DML}^+$, $\mathbf{DML}^+$-*theorem*, and $\mathbf{DML}^+$-*consistency* can be defined in the obvious way, cf. §**3.3**.)

Let $[\alpha]\phi?$ abbreviate the formula $\neg\mathsf{do}(\alpha; \neg\phi?)$; $E\phi$ is short for $\phi \vee D\phi$; $\mathsf{con}(\phi)$ abbreviates $(\mathsf{exp}(\neg\phi))^{\smile}$, $\mathsf{exp}(\phi)$ is short for $(\mathsf{exp}(\top); \phi?)$; and δ is short for $\top?$.

Besides enough classical tautologies, and the axioms of **DL** (taken as axioms over $\mathcal{DML}^+(\Phi)$) the system $\mathbf{DML}^+$ has the following axioms:

Definitions

(DML0)	$\mathsf{do}(\alpha) \leftrightarrow \mathsf{do}(\alpha; \delta)$,
(DML1)	$\mathsf{ra}(\alpha) \leftrightarrow \mathsf{do}(\alpha^{\smile})$,
(DML2)	$\mathsf{fix}(\alpha) \leftrightarrow \mathsf{do}(\alpha \cap \delta)$.

Basic axioms

(DML3) $[\alpha](p \to q) \to ([\alpha]p \to [\alpha]q)$,
(DML4) $\mathsf{do}(\alpha; p?) \to p \vee Dp$.

Relation connectives

(DML5) $\mathsf{do}(\alpha \cap \beta; p?) \to \mathsf{do}(\alpha; p?) \wedge \mathsf{do}(\beta; p?)$,
(DML6) $E(p \wedge \neg Dp) \to (\mathsf{do}(\alpha; p?) \wedge \mathsf{do}(\beta; p?) \to \mathsf{do}(\alpha \cap \beta; p?))$,
(DML7) $\mathsf{do}((\alpha; \beta); p?) \leftrightarrow \mathsf{do}(\alpha; \mathsf{do}(\beta; p?)?)$,
(DML8) $E(p \wedge \neg Dp) \to (\mathsf{do}(\alpha; p?) \leftrightarrow \neg\mathsf{do}(-\alpha; p?))$,
(DML9) $p \to [\alpha]\mathsf{do}(\alpha^{\smallsmile}; p?)$,
(DML10) $p \to [\alpha^{\smallsmile}]\mathsf{do}(\alpha; p?)$.

Test

(DML11) $\mathsf{do}(p?; q?) \leftrightarrow (p \wedge q)$.

Structure

(DML12) $\mathsf{do}(\mathsf{exp}(\mathsf{do}(\mathsf{exp}(p)))) \to \mathsf{do}(\mathsf{exp}(p))$,
(DML13) $p \to \mathsf{do}(\mathsf{exp}(p))$.

Besides those of **DL**, the rules of inference of **DML$^+$** are:

(NEC_α) $\phi \,/\, [\alpha]\phi$, for $\alpha \in Proc(\Phi)$.

4.6.3. DEFINITION. We define the derivation system **DML** in the original language $\mathcal{DML}$. First, let $D'\phi$ abbreviate $\mathsf{do}(-\delta; \phi?)$, and let E', O' be defined in terms of D' as E, O are defined in terms of D.

The axioms of **DML** are (DML0)–(DML13) with E' and D' instead of E and D, as well as (DML14) below.

(DML14) $\mathsf{do}((-\delta)^{\smallsmile}; p?) \to \mathsf{do}(-\delta; p?)$.

Its rules are of inference are (MP), (SUB), (NEC_α) and an irreflexivity rule for D':

($\text{IR}_{D'}$) $p \wedge \neg D'p \to \phi \,/\, \phi$, provided p does not occur in ϕ.

So **DML** differs from **DML$^+$** in that it views the difference operator D as a *defined* operator, rather than as a primitive one; **DML** can do without the **DL**-axioms (D1)–(D3) provided we add axiom (DML14).

4.6.4. REMARK. Observe that for every relational connective in our language **DML** has one or two axioms describing its behaviour *implicitly*, that is, in the context of a formula of the form $\mathsf{do}(\cdot; \cdot?)$. As $\mathcal{DML}$-formulas are evaluated at points, not at pairs or transitions, it is impossible to state explicitly how the relational connectives should behave.

COMPLETENESS RESULTS

Now that we have a derivation system we can prove our completeness results; as announced before we will use the construction from §3.3. As input for the construction we will use the modal operators D and $\mathsf{do}(\alpha; \cdot?)$, for α a procedure in $\mathcal{DML}$ in which all occurrences of exp are of the form $\mathsf{exp}(\top)$ only.

We need to make a few minor adjustments to the completeness construction of Chapter 3, mainly having to do with the additional modal operators $\mathsf{do}(\alpha; \cdot ?)$.

THE **Paste** FUNCTION. We extend Definition 3.3.5 with one clause for formulas of the form $\mathsf{do}(\alpha)$. We have to be somewhat careful about where we allow pasting of names in such formulas.

We recursively demand that whenever we select $\psi \trianglelefteq \mathsf{do}(\alpha)$ with the intention to paste a name next to it, then α is of the form $(\alpha'; \phi?)$, and $\psi \trianglelefteq \phi$. This is in full analogy with the pasting conventions for formulas $D\psi$, $\Diamond\psi$, ... in Chapter 3.

$$\mathtt{Paste}(\nu, \psi, \mathsf{do}(\alpha; \phi?)) = \mathsf{do}(\alpha; \mathtt{Paste}(\nu, \psi, \phi)?).$$

SWITCHING LEMMA. Let $\vdash$ denote $\vdash_{\mathbf{DML}}$. Then $\vdash \mathsf{do}(\alpha; \phi?) \to \psi$ iff $\vdash \phi \to [\alpha^{\smile}]\psi$.
Proof.

$$\begin{aligned} \vdash \mathsf{do}(\alpha; \phi?) \to \psi \quad &\Rightarrow \quad \vdash [\alpha^{\smile}]\mathsf{do}(\alpha; \phi?) \to [\alpha^{\smile}]\psi \\ &\Rightarrow \quad \vdash \phi \to [\alpha^{\smile}]\psi, \text{ by (DML9)} \\ &\Rightarrow \quad \vdash [\alpha]\mathsf{do}(\alpha^{\smile}; \neg\psi?) \to [\alpha]\neg\phi \\ &\Rightarrow \quad \vdash \neg\psi \to [\alpha]\neg\phi, \text{ by (DML10)} \\ &\Rightarrow \quad \vdash \mathsf{do}(\alpha; \phi?) \to \psi. \ \dashv \end{aligned}$$

CANONICAL RELATIONS. The canonical relation R^c_D is defined as in 3.3.11. For the modal operators $\mathsf{do}(\alpha; \cdot ?)$ the canonical relation R^c_α is defined by putting $R^c_\alpha \Sigma\Delta$ if for all ϕ, $\phi \in \Delta$ implies $\mathsf{do}(\alpha; \phi?) \in \Sigma$.

SUCCESSOR LEMMA. Let Δ be a maximal consistent distinguishing theory. If Δ a formula of the form $D\phi$ or $\mathsf{do}(\alpha; \phi?)$, then the required R^c_D-successor or R^c_α-successor exists: if $D\phi \in \Delta$, then there is a maximal consistent distinguishing Σ with $\phi \in \Sigma$ and $R^c_D \Delta\Sigma$, and if $\mathsf{do}(\alpha; \phi?) \in \Delta$, then there is a maximal consistent distinguishing Σ with $\phi \in \Sigma$ and $R^c_\alpha \Delta\Sigma$.

PROVISIONAL CANONICAL MODEL. A provisional canonical model $\mathfrak{M}^c$ is defined by putting $\mathfrak{M}^c = (W^c, R^c_D, R^c_{\mathsf{exp}}, [\![\cdot]\!]^c, V^c)$, where W^c is the set of all maximal $\mathbf{DML}^+$-consistent distinguishing theories, R^c_D is as defined before, $R^c_{\mathsf{exp}} = R^c_\alpha$, for $\alpha = \mathsf{exp}(\top)$, is the informational ordering, $[\![\alpha]\!]^c = R^c_\alpha$, and $V^c(p) = \{\, \Delta : p \in \Delta \,\}$.

Observe that the provisional canonical model may still be a *non-standard* model for $\mathbf{DML}^+$: R^c_D need not connect every two different point in W^c, even though it satisfies all the remaining properties: it is symmetric, pseudo-transitive, and irreflexive.

As before (3.3.14) one can prove a Provisional Truth Lemma on this non-standard model, interpreting, for example, $\mathsf{do}(\alpha \cap \beta; \phi?)$ using $R^c_{\alpha\cap\beta}$ rather than $R^c_\alpha \cap R^c_\beta$.

FINAL CANONICAL MODEL. To obtain a final canonical model which is based on a standard frame for **DML**$^+$ we generate along the relation R^c_D. More precisely, take any Δ in W^c, and consider all Σ with $R^c_D \Delta\Sigma$; let W^f be the resulting subset of W^c. For all procedures α, let R^f_α be the restriction R^c_α to W^f.

A *final canonical model* for **DML**$^+$ is a tuple $\mathfrak{M}^f = (W^f, R^f_D, R^f_{\mathsf{exp}}, [\![\cdot]\!]^f, V^f)$, with W^f, R^f_D as above, $R^f_{\mathsf{exp}} = R^f_\alpha$ for $\alpha = \mathsf{exp}(\top)$, $[\![\alpha]\!]^f = R^f_\alpha$, and $V^f(p) = V^c(p) \cap W^f$.

4.6.5. LEMMA. (Structure Lemma) *Any final canonical model* $\mathfrak{M}^f$ *for* **DML**$^+$ *is a standard model for* **DML**$^+$.

Proof. To show that $\mathfrak{M}^f$ is standard, we have to show that R^f_D is real inequality, that the relational connectives behave properly, and that R^f_{exp} is transitive and reflexive. A useful feature of the canonical model that is worth recalling before we start off, is that by construction for any Δ in $\mathfrak{M}^f$ there is a proposition letter p_Δ such that $p \wedge \neg Dp \in \Delta$.

First of all, as before, R^f_D is real inequality in $\mathfrak{M}^f$. As to the relational connectives, consider $\cap$. By (DML5)

$$[\![\alpha \cap \beta]\!]^f = R^f_{\alpha\cap\beta} \subseteq R^f_\alpha \cap R^f_\beta = [\![\alpha^f]\!] \cap [\![\beta]\!]^f.$$

For the converse inclusion, assume that $(\Delta, \Sigma) \in [\![\alpha]\!]^f \cap [\![\beta]\!]^f$. Let p be a unique proposition letter in Σ. Then $E(p \wedge \neg Dp)$, $\mathsf{do}(\alpha; p?)$, $\mathsf{do}(\beta; p?) \in \Delta$. Hence, by axiom (DML6), $\mathsf{do}(\alpha \cap \beta; p?) \in \Delta$. But this is possible only if $(\Delta, \Sigma) \in [\![\alpha \cap \beta]\!]^f$, as required.

By using axiom (DML7) it is easily verified that $[\![\alpha; \beta]\!]^f = [\![\alpha]\!]^f ; [\![\beta]\!]^f$.

To see that $[\![-\alpha]\!]^f = -[\![\alpha]\!]^f$, argue as follows. Assume $(\Delta, \Sigma) \in [\![-\alpha]\!]^f$. Let p be a unique proposition letter with $p \in \Sigma$. Then $E(p \wedge \neg Dp)$, $\mathsf{do}(-\alpha; p?) \in \Delta$. Therefore, $\neg\mathsf{do}(\alpha; p?) \in \Delta$, by axiom (DML8). It follows that $(\Delta, \Sigma) \notin [\![\alpha]\!]^f$. For the converse inclusion, assume $(\Delta, \Sigma) \notin [\![\alpha]\!]^f$. Choose a unique proposition letter p in Σ. Then, by the Successor Lemma, $E(p \wedge \neg Dp)$, $\neg\mathsf{do}(\alpha; p?) \in \Delta$, and, by axiom (DML8), $\mathsf{do}(-\alpha; p?) \in \Delta$. But this is possible only if $(\Delta, \Sigma) \in [\![-\alpha]\!]^f$.

To prove that the converse operation $\breve{}$ is standard, use axioms (DML9) and (DML10). For the test operation ? one uses (DML11).

Finally, to see that R^f_{exp} has the right structural properties, viz. that it is transitive and reflexive, use axioms (DML12) and (DML13). $\dashv$

4.6.6. LEMMA. (Final Truth Lemma) *Let* $\mathfrak{M}^f$ *be a final canonical model. For all* $\Delta \in W^f$ *and all* $\mathcal{DML}^+$*-formulas* ϕ, *we have* $\mathfrak{M}, \Delta \models \phi$ *iff* $\phi \in \Delta$.

Proof. The proof is by induction on ϕ; the only interesting cases are $D\phi$ and $\mathsf{do}(\alpha)$, for α any procedure. We only do the case $\mathsf{do}(\alpha)$.

If $\mathsf{do}(\alpha) \in \Delta$ then, by (DML0), $\mathsf{do}(\alpha; \top?) \in \Delta$. By the Successor Lemma there exists Σ with $(\Delta, \Sigma) \in [\![\alpha]\!]$. Hence, $\Delta \models \mathsf{do}(\alpha)$. Conversely, if $\Delta \models \mathsf{do}(\alpha)$, choose Σ such that $(\Delta, \Sigma) \in [\![\alpha]\!]$. As $\top \in \Sigma$ it follows that $\mathsf{do}(\alpha; \top?) \in \Delta$. So, by (DML0) again, $\mathsf{do}(\alpha) \in \Delta$. $\dashv$

4.6.7. THEOREM. (Completeness Theorem) *Let* $\Sigma \cup \{\phi\}$ *be a set of* $\mathcal{DML}^+$*-formulas. Then* $\Sigma \vdash \phi$ *in* **DML**$^+$ *iff* $\Sigma \models \phi$.

To port Theorem 4.6.7 to the old language $\mathcal{DML}$, we need the following.

4.6.8. DEFINITION. Define a translation $(\cdot)^\ddagger$ from $\mathcal{DML}^+$-formulas to $\mathcal{DML}$-formulas by putting $p^\ddagger = p$, letting $(\cdot)^\ddagger$ commute with the connectives and operators of $\mathcal{DML}$, and putting $(D\phi)^\ddagger = \mathsf{do}(-\delta;\phi^\ddagger?)$ $(= D'\phi^\ddagger)$.

4.6.9. LEMMA. *Let ϕ be a $\mathcal{DML}^+$-formula. Then $\vdash_{\mathbf{DML}^+} \phi$ iff $\vdash_{\mathbf{DML}} \phi^\ddagger$.*

Proof. This is by induction on the length of derivations. It suffices to show that the axioms and rules of the one system are derivable in the other.

As the **DML**-axioms are valid, by the completeness of $\mathbf{DML}^+$ they are derivable in $\mathbf{DML}^+$. The rules of **DML** are derived rules of $\mathbf{DML}^+$.

Conversely, except for the **DL**-axioms (D1)–(D3) it is immediate that the translations of the $\mathbf{DML}^+$-axioms are derivable in **DML**. As to $(\mathrm{D1})^\ddagger$, this is an instance of (DML3). For (D2) we have $(\mathrm{D2})^\ddagger = p \to [-\delta]\mathsf{do}(-\delta;p?)$. Let $\vdash$ denote $\vdash_{\mathbf{DML}}$. Then

$$\begin{aligned} &\vdash \mathsf{do}((-\delta)^\smile;p?) \to \mathsf{do}(-\delta;p?), \quad \text{(DML14)}\\ &\quad\Rightarrow \quad \vdash [-\delta]\big(\mathsf{do}((-\delta)^\smile;p?) \to \mathsf{do}(-\delta;p?)\big)\\ &\quad\Rightarrow \quad \vdash [-\delta]\mathsf{do}((-\delta)^\smile;p?) \to [-\delta]\mathsf{do}(-\delta;p?), \quad \text{by (DML3), (MP)}\\ &\quad\Rightarrow \quad \vdash p \to [-\delta]\mathsf{do}(-\delta;p?), \quad \text{by (DML9).}\end{aligned}$$

As for $(\mathrm{D3})^\ddagger$, observe that $(\mathrm{D3})^\ddagger = \mathsf{do}(-\delta;\mathsf{do}(-\delta;p?)?) \to p \vee \mathsf{do}(-\delta;p?)$. Now, by (DML4) we have $\vdash \mathsf{do}((-\delta;-\delta);p?) \to p \vee \mathsf{do}(-\delta;p?)$. Also,

$$\vdash \mathsf{do}(-\delta;\mathsf{do}(-\delta;p?)?) \to \mathsf{do}((-\delta;-\delta);p?),$$

by (DML7). Hence $\vdash (\mathrm{D3})^\ddagger$. Finally, it is easily verified that the translations of the rules of $\mathbf{DML}^+$ are derived rules in **DML**. $\dashv$

4.6.10. THEOREM. (Completeness Theorem) *Let $\Sigma \cup \{\phi\}$ be a set of $\mathcal{DML}$-formulas. Then $\Sigma \vdash \phi$ in* **DML** *iff $\Sigma \models \phi$.*

Proof. This is immediate from Theorem 4.6.7 and Lemma 4.6.9. $\dashv$

I want to stress that nothing in the proof of Theorem 4.6.10 depends in an essential way on the relation underlying exp being a pre-order. Also, the proof and and result easily generalize to a logic $\mathbf{DML}(\Phi;\Omega)$ in an extension $\mathcal{DML}(\Phi;\Omega)$ of $\mathcal{DML}$, where one has propositions Φ as before, and multiple base relations $\sqsubseteq_i$ $(i \in \Omega)$, none of which needs to be a pre-order.

4.7 CONCLUDING REMARKS

In this Chapter we analyzed a dynamic modal language $\mathcal{DML}$ whose distinctive aspect is its attention for the interplay between static objects and dynamic transitions. The dynamic language turned out to be a powerful one, and to have a number of applications in other areas of logic. The expressive power of

$\mathcal{DML}$ was exemplified by the fact that it coincides with a large fragment of first-order logic, that its satisfiability problem is undecidable, and that we needed a difference operator D and an irreflexivity rule to match its expressive power. Nevertheless, the language of **DML** could still be analyzed with general modal logic tools: using general results from Chapter 6 definability and preservation results could be established.

Several natural extensions of the language studied here present themselves. Given the close connections between **DML** and **PDL**, it may seem natural to add Kleene star * that is present in **PDL** to **DML**. But **DML** with Kleene star has a Σ^1_1-complete satisfiability problem; this may be proved by using a *recurrent tiling problem* (RTP): given a finite set of tiles T, and a tile $d_1 \in T$, can T tile $\mathbb{N} \times \mathbb{N}$ such that d_1 occurs infinitely often on the first row? The RTP is a Σ^1_1-complete problem (Harel 1983). To obtain a reduction of the RTP to satisfiability in $\mathcal{DML}$ plus Kleene star, we define a formula ϕ_{RT} as the conjunction of the formula ϕ_T used in the proof of Theorem 4.5.1 and a formula REC to be defined shortly. We use a new propositional symbol row_0 which can only be true at nodes on the bottom row of a grid; we will ensure that there exists an infinite number of worlds where row_0 holds and the tile d_1 is placed. Now, define REC to be the conjunction of row_0, $A[\mathrm{UP}]\neg row_0$ and

$$[\mathrm{RIGHT}^*](row_0 \to \mathsf{do}(\mathrm{RIGHT}^*; (row_0 \wedge d_1)?)).$$

As in the proof of Theorem 4.5.1 it may be shown that T recurrently tiles $\mathbb{N} \times \mathbb{N}$ iff ϕ_{RT} is satisfiable. This proves a Σ^1_1 lower bound. A Σ^1_1 upper bound is found by observing that that a formula in $\mathcal{DML}$ plus Kleene star is satisfiable iff it is satisfiable on a countable model.

One important issue not dealt with in this Chapter is: what are the modal algebras appropriate for **DML**? In recent years so-called *Peirce algebras* have been invented; these are are two-sorted structures not unlike the *dynamic algebras* underlying **PDL** (cf. Kozen (1981), Pratt (1990*b*)). The following Chapter will be devoted exclusively to Peirce algebras, and their connection with **DML** will be made clear there. Among other things, the next Chapter uses the completeness of **DML** to arrive at an algebraic completeness result; it also studies a fully two-sorted extension of **DML** in which states and transitions are treated on a par.

To conclude, here are some questions (cf. also Remark 4.4.12).

1. In §4.5 we mentioned a number of systems closely related to **DML** with a decidable satisfiable problem. An obvious question is to locate the boundary (in terms of fragments of **DML**) where the satisfiability becomes undecidable more precisely, and to identify as large as possible a decidable fragment of **DML**. In particular, what if we start with as few relational connectives as possible but with the minimal expansion and contraction operators μ-exp and μ-con — will we still have decidability?

2. As with the system **DL** in Chapter 3 one can ask whether the irreflexivity rule (IR_D) is necessary in **DML**. Adding the rule to **DL** made all incompleteness phenomena disappear, although it did not add any new consequences to the base logic ($\mathbf{DL}^-$ and **DL** shared the same theorems according to 3.3.3 and 3.3.18). With **DML** it is not clear whether the IR_D-rule adds anything to **DML** in terms of new consequences; but, as with the logics $\mathbf{DL}_m$ and $\mathbf{DL}_t$, there will still be incomplete extensions of the base system in the language of **DML**.
3. In §4.5 we found several fragments of $\mathcal{DML}$ with a decidable satisfiability problem. What is their complexity?

5

The Logic of Peirce Algebras

5.1 Introduction

Chapter 4 left us with the question what the modal algebras appropriate for the dynamic modal logic **DML** are. Recently Peirce algebras have emerged as the common mathematical structures underlying many of the phenomena being studied in program semantics, AI and natural language analysis; they are also the modal algebras underlying **DML**. Peirce algebras are two-sorted algebras in which sets and relations co-exist together with operations between them modeling their interaction. The most important such operations considered here are the *Peirce product* : that takes a relation and a set, and returns a set

$$R : A = \{\, x : \exists y\, ((x, y) \in R \wedge y \in A) \,\},$$

and *right cylindrification* ^{c} which takes a set and returns a relation

$$A^c = \{\, (x, y) : x \in A \,\}.$$

The main purpose of this Chapter is to use modal formalisms and techniques to axiomatize the representable Peirce algebras.

The next section quickly reviews basic algebraic definitions; it also describes areas where Peirce algebras emerge. §5.3 explains how modal languages may be used to obtain algebraic completeness results for Peirce algebras. §5.4 uses results from Chapter 4 to axiomatize the set equations valid in all representable Peirce algebras; here a set equation is an equation between terms denoting sets. §5.5 contains the main result of this Chapter: an axiomatization of both the set and relation equations valid in all representable Peirce algebras; this work builds in an essential way on Venema (1991), who used non-standard means to arrive at an axiomatization of representable relation algebras. Section six discusses the expressive power of Peirce algebras, and §5.7 concludes with some questions.

5.2 Preliminaries

Basic definitions

5.2.1. Definition. Let U be a set; $Re(U)$ is $\{\, R : R \subseteq U \times U \,\}$. R, S typically denote elements of $Re(U)$, while A, B typically denote elements of 2^U, the power set of U.

Recall the following operations on elements of $Re(U)$.

top	∇	$\{(r,s) \in U \times U : r, s \in U\}$
complement	$-R$	$\{(r,s) \in U \times U : (r,s) \notin R\}$
converse	R^{-1}	$\{(r,s) \in U \times U : (s,r) \in R\}$
diagonal	Id	$\{(r,s) \in U \times U : r = s\}$
composition	$R \mid S$	$\{(r,s) \in U \times U : \exists u\, ((r,u) \in R \wedge (u,s) \in S)\}$

We also consider the following operations from $Re(U)$ and $Re(U) \times 2^U$ to 2^U

domain	$\mathsf{do}(R)$	$\{x \in U : \exists y \in U\, ((x,y) \in R)\}$
range	$\mathsf{ra}(R)$	$\{x \in U : \exists y \in U\, ((y,x) \in R)\}$
Peirce product	$R : A$	$\{x \in U : \exists y \in U\, ((x,y) \in R \wedge y \in A)\}$,

as well as the following operations going from 2^U to $Re(U)$

tests	$A?$	$\{(x,y) \in U \times U : x = y \wedge x \in A)\}$
right cylindrification	A^c	$\{(x,y) \in U \times U : x \in A\}$.

5.2.2. DEFINITION. A *relation type algebra* is a Boolean algebra with a binary operation ;, a unary operation $\breve{}$, and a constant 1'. The class FRA of *full relation algebras* consists of all relation type algebras isomorphic to an algebra of the form $\mathfrak{R}(U) = (Re(U), \cup, -, |, ^{-1}, Id)$. RRA is the class of *representable relation algebras*, that is, RRA = $\mathbf{SP}$(FRA) (= $\mathbf{HSP}$(FRA) by a result due to Birkhoff).

RA is the class of *relation algebras*, that is, of relation type algebras $\mathfrak{A} = (A, +, -, ;, \breve{}, 1')$ satisfying the axioms

(R0) $(A, +, -, \emptyset)$ is a Boolean algebra
(R1) $(x + y); z = x; z + y; z$
(R2) $(x + y)\breve{} = x\breve{} + y\breve{}$
(R4) $(x; y); z = x; (y; z)$
(R5) $x; 1' = x = 1'; x$
(R6) $(x\breve{})\breve{} = x$
(R7) $(x; y)\breve{} = y\breve{}; x\breve{}$
(R8) $x\breve{}; -(x; y) \leq -y$.

The reader is referred to Jónsson (1982, 1991) for the essentials on relation algebra; at this point it suffices to recall that there are relation algebras that are not representable; no finite set of axioms suffices to axiomatize all valid principles concerning binary relations.

5.2.3. DEFINITION. A *Peirce type algebra* is a two-sorted algebra $(\mathfrak{B}, \mathfrak{R}, :\, ^c)$, where $\mathfrak{B}$ is a Boolean algebra, $\mathfrak{R}$ is a relation type algebra, : is a function from $\mathfrak{R} \times \mathfrak{B}$ to $\mathfrak{B}$, and $^c : \mathfrak{B} \to \mathfrak{R}$. The class FPA of *full Peirce algebras* consists of all Peirce type algebras isomorphic to an algebra of the form

$$\mathfrak{P}(U) = \big((2^U, \cup, -, \emptyset), (Re(U), \cup, -, ^{-1}, |, Id), :, ^c\big).$$

The class RPA of *representable Peirce algebras* is defined as RPA = $\mathbf{HSP}$(FPA), the variety generated by FPA.

PA is the class of *Peirce algebras*, that is of all Peirce type algebras $\mathfrak{A} = (\mathfrak{B}, \mathfrak{R}, :, {}^c)$ where $\mathfrak{B}$ is a Boolean algebra, $\mathfrak{R}$ is a relation algebra, : is a mapping $\mathfrak{R} \times \mathfrak{B} \to \mathfrak{B}$ such that

(P1) $r : (a + b) = (r : a) + (r : b)$
(P2) $(r + s) : a = (r : a) + (s : a)$
(P3) $r : (s : a) = (r; s) : a$
(P4) $1' : a = a$
(P5) $0 : a = 0$
(P6) $r^{\smallsmile} : -(r : a) \leq -a$,

while ^{c} is a mapping $\mathfrak{B} \to \mathfrak{R}$ such that

(P7) $a^c : 1 = a$
(P8) $(r : 1)^c = r; 1$.

Algebras of the form $(\mathfrak{B}, \mathfrak{R}, :)$ were introduced by Brink (1981) as *Boolean modules*. Sources for Peirce algebras are (Britz 1988) and (Brink, Britz & Schmidt 1993).

The algebraic language of Peirce algebras has two sorts of terms: one interpreted in $\mathfrak{B}$, the other in $\mathfrak{R}$. Terms of the first sort are called *set terms*, terms of the second sort *relation terms*. Identities between set terms are called *set identities*; identities between relation terms are *relation identities*.

What distinguishes Peirce algebras from relation algebras is that the former have a separate sort of sets plus functions relating the set and relation sorts. By identifying set elements with the right ideal elements of the relational sort, the sort of set elements can be faithfully embedded in the relational sort, and the interactions between the two sorts can be studied at a meta-level (Brink et al. 1993). But, although it is important to know that such reductions exist, Peirce algebras may be the more natural framework for certain applications, especially when these require (equational reasoning about) the two separate sorts as well as equational reasoning about functions modeling their interaction. (This is similar to the relation between ZF set theory and Peano arithmetic: the latter is interpretable in ZF, and hence one may stick to ZF to do arithmetic, but this wouldn't be as natural as working in PA).

Related algebraic structures

A dynamic algebra is a two-sorted algebra of sets and relations. Its relations are organized in a Kleene algebra, and its sets are organized in a Boolean algebra. The following definitions are due to Kozen (1981).

5.2.4. Definition. A *Kleene type algebra* is an algebra $\mathfrak{K} = (K, +, 0, ;, {}^*, 1')$ where $(K, +, 0)$ is an upper semilattice, $(K, ;, 1')$ is a monoid. KA is the class of all Kleene type algebras such that

(KA1) $r; (s + t) = r; s + r; t$
(KA2) $(r + s); t = r; t + s; t$
(KA3) $r; 0 = 0; r = 0$
(KA4) $r; s^*; t = \sum_n (r; s^n; t)$.

5.2.5. Definition. A *dynamic type algebra* is a two-sorted algebra $(\mathfrak{B}, \mathfrak{K}, :)$ where $\mathfrak{B}$ is a Boolean algebra, $\mathfrak{K}$ is a Kleene algebra, and : is a function from $\mathfrak{B} \times \mathfrak{K}$ to $\mathfrak{B}$. DA is the class of all *dynamic algebras*, that is of all dynamic type algebras $\mathfrak{D} = (\mathfrak{B}, \mathfrak{K}, :)$ where $\mathfrak{B}$ is a Boolean algebra, $\mathfrak{K}$ is a Kleene algebra, and : is a function from $\mathfrak{K} \times \mathfrak{B}$ to $\mathfrak{B}$ such that

(DA1) $r : (a + b) = r : a + r : b$
(DA2) $(r + s) : a = r : a + s : a$
(DA3) $r : (s : a) = (r; s) : a$
(DA4) $1' : a = a$
(DA5) $0 : a = a : 0 = 0$
(DA6) $r^* : a = \sum_n (r^n : a)$.

So, dynamic algebras differ from Boolean modules in that their relational component is based on a Kleene algebra rather than a relation algebra. Pratt (1990*b*) discusses the relative merits of Boolean modules and dynamic algebras. A close connection between Peirce algebras and dynamic algebras emerges when we add to the latter a test operation $? : \mathfrak{B} \to \mathfrak{K}$ satisfying

(DA?) $\langle p?\rangle q = p \cdot q$.

Finally, any *join complete* Peirce algebra gives rise to a dynamic algebra.

To equip fragments of natural language with a variable free semantic analysis, Suppes (1976) defines a class of algebras closely related to Peirce algebras called extended relation algebras. The *image* of a relation R from a set A is the set $R\text{ ''}\ A = \{\, y : \exists x\, ((x, y) \in R \wedge x \in A\,\}$, and the *domain restriction* of a relation R to a set A is the relation $R \lceil A = \{\, (x, y) : (x, y) \in R \wedge x \in A\,\}$. For U a non-empty set the *extended relation algebra* $\mathfrak{E}(U)$ over U has $2^U \cup 2^{U\times U}$ as its universe, and $\cup$, $\cap$, $-$, ;, $\breve{}$, image and domain restriction as operations, with complementation on 2^U taken relative to U, and on $2^{U\times U}$ relative to $U \times U$. Clearly, an extended relation algebra gives rise to a Peirce algebra when we explicitly distinguish between the operations on 2^U and those on $2^{U\times U}$, and define $R : A = R\breve{}\text{ ''}\ A$, and $A^c = \nabla \lceil A$. And conversely, putting $R\text{ ''}\ A = R\breve{} : A$, and $R \lceil A = R \cap A^c$, and forgetting about the distinction between the set space and relation space turns a Peirce algebra into an extended relation algebra. (Cf. page 76 below for a brief sketch of the use of such algebras.)

Where Peirce algebras emerge

In a number of areas frameworks are studied that have Peirce algebras in common as their underlying mathematical structures: modal logic, arrow logic, knowledge representation, natural language analysis, and weakest prespecifications.

Modal logic. We can now answer our earlier question asking for the modal algebras appropriate for $\mathcal{DML}$. Recall that models for $\mathcal{DML}$ are structures of the form $\mathfrak{M} = (W, \sqsubseteq, [\![\cdot]\!], V)$, where $\sqsubseteq\ \subseteq W^2$ is a pre-order, $[\![\cdot]\!]$ assigns binary relations to procedures, and V is a valuation assigning subsets of W to proposition letters (Definition 4.2.2). The interpretation of the operator **do** taking relations to propositions is: $\mathfrak{M}, x \models \mathsf{do}(\alpha)$ iff $\exists y\,((x,y) \in [\![\alpha]\!])$, while the relational part is interpreted using the mapping $[\![\cdot]\!]$:

$$\begin{array}{rcl@{\qquad}rcl} [\![-\alpha]\!] & = & -[\![\alpha]\!] & [\![\mathsf{exp}(\phi)]\!] & = & \lambda xy.\,(x \sqsubseteq y \wedge \mathfrak{M}, y \models \phi), \\ [\![\alpha\breve{}]\!] & = & \{\,(x,y) : (y,x) \in [\![\alpha]\!]\,\} & [\![\phi?]\!] & = & \{\,(x,x) : \mathfrak{M}, x \models \phi\,\} \\ [\![\alpha \cap \beta]\!] & = & [\![\alpha]\!] \cap [\![\beta]\!] & [\![\alpha;\beta]\!] & = & [\![\alpha]\!];[\![\beta]\!]. \end{array}$$

What is the connection between $\mathcal{DML}$ and Peirce algebras? To obtain a proper match between the two we need to allow a set Ω of base relations $\sqsubseteq_i$ (that need not be pre-orders) rather than just a single one, and allow for multiple expansion modes $\mathsf{exp}_i(\cdot)$. The corresponding structures give rise to full Peirce algebras, and conversely; thus $\mathcal{DML}$ and (full) Peirce algebras share the same ontology of sets and relations. In addition, the (extended) $\mathcal{DML}$-operators are definable in full Peirce algebras, and the operators of full Peirce algebras are definable on $\mathcal{DML}$-models:

$\mathcal{DML}$	$\mathsf{do}(\alpha)$	$\mathsf{exp}_i(\phi)$	$\phi?$	FPA	$\alpha : \phi$	ϕ^c
FPA	$(\nabla : \alpha)$	$(\sqsubseteq_i : \phi)$	$\phi^c \cap Id$	$\mathcal{DML}$	$\mathsf{do}(\alpha;\phi?)$	$\phi?;(\delta \cup -(\delta))$.

This implies that any axiom system complete for validity in $\mathcal{DML}$ structures also generates the 'set equations' valid in FPA (Theorem 5.4.3).

ARROW LOGIC. As pointed out in §4.5 *Arrow Logic* arises as an effort to do transition logic without the computational complexity that comes with transition logics based on the identification of transitions as ordered pairs. Instead, Arrow Logic as developed by Van Benthem (1991*a*) takes transitions seriously as dynamic objects in their own right. The general program of Arrow Logic proposes a re-design of systems of transition logic to isolate the genuine computational aspects from the mathematical modeling aspects.

(Van Benthem 1992) contains samples of this program. In particular, it discusses a two-sorted arrow logic whose models have both states and arrows, and whose formulas are sorted accordingly. The models of this (decidable) arrow logic may be viewed as an 'arrow-ized' version of our Peirce algebras. Without going into full detail here, we stress that the decidability result is obtained by abstracting away from any set-theoretical assumptions concerning objects and operations of Peirce algebra. For instance, a test $\phi?$ is successfully performed at an arrow x_a if there exists a state y_s that is in the test-relation T with x_a and which satisfies ϕ: $x_a \models \phi?$ iff for some state y_s, $Tx_a y_s$ and $y_s \models \phi$. We refer the reader to (Van Benthem 1992) for further details.

KNOWLEDGE REPRESENTATION. In terminological languages one expresses information about hierarchies of concepts. They allow the definition of concepts and roles built out of primitive concepts and roles. *Concepts* are interpreted as sets (of individuals), and *roles* as binary relations between individuals. Compound expressions are built up using various language constructs. Brink et al. (1993) propose a terminological language $\mathcal{U}^-$ whose operations are merely a notational variant of the operations of (full) Peirce algebras. For instance, $\mathcal{U}^-$ has an operation **restrict** that takes a relation and a set and returns a relation. Models for $\mathcal{U}^-$ have the form $(\mathcal{D}^{\mathcal{I}}, \cdot^{\mathcal{I}})$, for $\mathcal{D}^{\mathcal{I}}$ a domain of interpretation, and

$\cdot^{\mathcal{I}}$ an *interpretation function* which assigns to every concept C a subset $\mathsf{C}^{\mathcal{I}}$ of $\mathcal{D}$, and to every role a binary relation $\mathsf{R}^{\mathcal{I}}$ over $\mathcal{D}^{\mathcal{I}}$. The interpretation of **restrict** is

$$(\mathsf{restrict\ R\ C})^{\mathcal{I}} = \{ (x, y) : (x, y) \in \mathsf{R}^{\mathcal{I}} \wedge y \in \mathsf{C}^{\mathcal{I}} \}.$$

As $\mathcal{U}^-$ and (full) Peirce algebras share the same ontology, and (modulo some rewriting) the same operations, Peirce algebras supply an alternative semantic interpretation for the terminological language $\mathcal{U}^-$, in which the basic terminological concerns, viz. subsumption and satisfiability problems, re-appear as derivability issues in equational logic.

NATURAL LANGUAGE ANALYSIS. This example involves the earlier extended relation algebras; as (Suppes 1976, Böttner 1992) show, those structures arise in attempts to equip fragments of natural language with variable free semantics. I will illustrate the main point with an example from (Schmidt 1993). Consider a natural language fragment described by a phrase structure grammar G as in the left-hand side of (5.1), where S, NP, VP, TV, PN have their usual meaning: 'sentence,' 'noun phrase,' 'verb phrase,' 'transitive verb' and 'proper noun.'

$$\begin{array}{rcl} \mathrm{S} & \rightarrow & \mathrm{NP} + \mathrm{VP} \\ \mathrm{VP} & \rightarrow & \mathrm{TV} + \mathrm{NP} \\ \mathrm{NP} & \rightarrow & \mathrm{PN} \end{array} \qquad \begin{array}{c} [\mathrm{NP}] \sqsubseteq [\mathrm{VP}] \\ [\mathrm{TV}] : [\mathrm{NP}] \\ [\mathrm{PN}]. \end{array} \tag{5.1}$$

Production rules in the grammar are associated with a semantic function $[\cdot]$ in a compositional way as indicated in the right-hand side of (5.1). In other words, semantic trees are construed in parallel with syntactic derivation trees. The semantic trees are linked to extended relation algebras $\mathfrak{E}(U)$ via a valuation that maps terminal symbols of G onto an element of $\mathfrak{E}(U)$, where nouns are mapped onto sets and transitive verbs onto binary relations, thus equipping our natural language fragment with a variable free semantics.

WEAKEST PRESPECIFICATIONS. The use of relation algebra in proving properties of programs goes back at least to De Bakker & De Roever (1973). The calculus of weakest prespecifications of (Hoare & He 1987) is used as a formal tool in program specification. In this calculus programs are binary relations that may be combined using relation algebraic connectives. A special class of relations is called *conditions*; they express conditional statements, and are defined as the right ideal elements, that is, elements R for which $R = R; \nabla$. As the right ideal elements form a Boolean algebra, the natural algebraic setting for the calculus of weakest prespecifications is a Peirce algebra with programs living in a relation algebra, conditions living in a separate Boolean algebra, and ^{c} and $:$ being used to move across from one to the other, cf. (Brink et al. 1993).

5.3 MODAL PRELIMINARIES

It's time to get to work. The main goal of this Chapter is to present a complete axiomatization of the set equations and relation equations valid in FPA. This has two natural subgoals: to axiomatize the set equations valid in FPA, and to axiomatize the relation equations valid in FPA. The latter is left for further study; the former will be settled in §5.4 as a corollary to results in Chapter 4.

This section introduces two modal languages: one for dealing with set equations only, another for the full case with both set equations and relation equations. The two languages will be based on the same set of connectives and modal operators, but they differ in their notions of legal formulas.

Basic definitions

To start, Table 5.1 lists the notation we adopt.

	Full version	Abstract version	Modal version
relations	R, S	x, y	α, β
top	∇	1	$\mathbf{1}$
bottom	$\emptyset$	0	$\mathbf{0}$
diagonal	Id	1'	δ
complement	$-$	$-$	$-$
converse	$^{-1}$	$\smile$	$\otimes$
union	$\cup$	$+$	$\cup$
implication	$\rightarrow$		$\rightarrow$
composition	$\mid$	$;$	$\circ$
sets	A, B	x, y	ϕ, ψ
top	$\top$	1	$\top$
bottom	$\bot$	0	$\bot$
complement	$\neg$	$-$	$\neg$
union	$\cup$	$+$	$\vee$
implication	$\rightarrow$	$\rightarrow$	$\rightarrow$
right cylindrification	^{c}	c_1	$\updownarrow\cdot$
Peirce product	$:$	$:$	$\langle\cdot\rangle\cdot$

Table 5.1: A plethora of notations.

5.3.1. Definition. Let $\Phi = \{p_0, p_1, \ldots\}$ be a countable set of propositional (set) variables. Let Ω be a countable set of atomic relation symbols. The formulas of the *one-sorted* language $\mathcal{ML}_1(\delta, \otimes, \circ, \langle\rangle, \updownarrow; \Phi; \Omega)$, or $\mathcal{ML}_1$, are generated by the rule

$$\phi ::= \bot \mid \top \mid p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \langle\alpha\rangle\phi,$$

where α is a relation symbol taken from the set $Proc(\Phi, \Omega)$ generated by the rule

$$\alpha ::= \mathbf{0} \mid \mathbf{1} \mid \delta \mid a \mid -\alpha \mid \alpha_1 \cap \alpha_2 \mid \otimes\alpha \mid \alpha_1 \circ \alpha_2 \mid \updownarrow\phi.$$

So $\mathcal{ML}_1$ has *only one* sort of formulas — they will be interpreted as sets, and hence be called *set formulas*.

5.3.2. Definition. A model for $\mathcal{ML}_1$ is a *one-sorted Peirce model*, that is: a structure $\mathfrak{M} = (\mathfrak{F}, V)$ where $\mathfrak{F} = (W_s, (R_\alpha)_{\alpha \in Proc(\Phi,\Omega)})$, V is a *one-sorted* *valuation*, that is, a function assigning to every propositional (set) variable a subset of W_s. For $\alpha \in Proc(\Phi, \Omega)$, R_α satisfies

$$\begin{array}{rcl} R_a & \subseteq & {}^2W_s \quad (a \in \Omega) \\ R_0 & = & \emptyset \\ R_1 & = & W_s \times W_s \\ R_\delta & = & \{ (x,y) \in W_s \times W_s : x = y \} \\ R_{\otimes\alpha} & = & \{ (x,y) \in W_s \times W_s : (y,x) \in R_\alpha \} \\ R_{\alpha\circ\beta} & = & \{ (x,y) \in W_s \times W_s : \exists z \in W_s \, ((x,z) \in R_\alpha \wedge (z,y) \in R_\beta) \} \\ R_{\updownarrow\phi} & = & \{ (x,y) \in W_s \times W_s : x \models \phi \}. \end{array}$$

$\mathcal{ML}_1$-formulas are evaluated at elements of W_s; notation: $\mathfrak{M}, x_s \models \phi$, or simply $x_s \models \phi$, when $\mathfrak{M}$ is clear from the context. Truth of $\mathcal{ML}_1$-formulas is defined inductively, with $\mathfrak{M}, x_s \models p$ iff $x_s \in V(p)$; $x_s \models \neg\phi$ iff not $x_s \models \phi$; $x_s \models \phi \wedge \psi$ iff both $x_s \models \phi$ and $x_s \models \psi$; and

$$x_s \models \langle\alpha\rangle\phi \text{ iff for some } y_s \text{ in } W_s, \ (x_s, y_s) \in R_\alpha \text{ and } y_s \models \phi.$$

$\mathcal{ML}_1$-formulas correspond to set equations in Peirce algebras. For the full case with both set equations and relation equations we define a two-sorted modal language.

5.3.3. DEFINITION. Let $\Phi = \{ p_0, p_1, \dots \}$ be a countable set of propositional variables. Let Ω be a countable set of atomic relation symbols. The formulas of the *two-sorted language* $\mathcal{ML}_2(\delta, \otimes, \circ, \langle\rangle, \updownarrow; \Phi; \Omega)$, or $\mathcal{ML}_2$ for short, are generated by the rules

$$\phi ::= \bot \mid \top \mid p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \langle\alpha\rangle\phi,$$

and

$$\alpha ::= \mathbf{0} \mid \mathbf{1} \mid \delta \mid a \mid -\alpha \mid \alpha_1 \cap \alpha_2 \mid \otimes\alpha \mid \alpha_1 \circ \alpha_2 \mid \updownarrow\phi.$$

As before, the first sort of formulas will be interpreted as sets (and called set formulas); formulas of the second sort will be interpreted as relations and called *relation formulas*.

5.3.4. DEFINITION. A *two-sorted frame* is a tuple $\mathfrak{F} = (W_s, W_r, I, R, C, F, P)$, where $W_s \cap W_r = \emptyset$, $I \subseteq W_r$, $R \subseteq {}^2W_r$, $C \subseteq {}^3W_r$, $F \subseteq W_r \times W_s$, and $P \subseteq W_s \times W_r \times W_s$.

Given a set U, a two-sorted frame is called *the two-sorted Peirce frame over* U if, for some base set U, $W_s = U$ and $W_r = U \times U$, and

$$\begin{array}{rcl} I & = & \{ (u,v) \in U \times U : u = v \} \\ R & = & \{ ((u_1,v_1),(u_2,v_2)) \in {}^2(U \times U) : u_1 = v_2 \wedge u_2 = v_1 \} \\ C & = & \{ ((u_1,v_1),(u_2,v_2),(u_3,v_3)) \in {}^3(U \times U) : u_1 = u_2 \wedge v_1 = v_3 \wedge v_2 = u_3 \} \\ F & = & \{ ((u_1,v_1),u_2) \in (U \times U) \times U : u_1 = u_2 \} \\ P & = & \{ (u_1,(u_2,v_2),u_3) \in U \times (U \times U) \times U : u_1 = u_2 \wedge v_2 = u_3 \}. \end{array}$$

The class of two-sorted Peirce frames is denoted by TPF.

5.3.5. DEFINITION. A model for $\mathcal{ML}_2$ is a *model based on a two-sorted frame*, that is, a structure $\mathfrak{M} = (\mathfrak{F}, V)$ where $\mathfrak{F}$ is a two-sorted frame, and V is a *two-sorted valuation*, that is, a function assigning subsets of W_s to set variables, and subsets of W_r to relation variables. Truth of a formula at a state is defined inductively, with the interesting clauses being

$$\begin{array}{rll} \mathfrak{M}, x_r \models \delta & \text{iff} & x_r \in I \\ \mathfrak{M}, x_r \models \otimes\alpha & \text{iff} & \exists y_r\, (Rx_r y_r \wedge y_r \models \alpha) \\ \mathfrak{M}, x_r \models \alpha \circ \beta & \text{iff} & \exists y_r z_r\, (Cx_r y_r z_r \wedge y_r \models \alpha \wedge z_r \models \beta) \\ \mathfrak{M}, x_s \models \langle\alpha\rangle\phi & \text{iff} & \exists y_r z_s\, (Px_s y_r z_s \wedge y_r \models \alpha \wedge z_s \models \phi) \\ \mathfrak{M}, x_r \models \updownarrow\phi & \text{iff} & \exists y_s\, (Fx_r y_s \wedge y_s \models \phi). \end{array}$$

Here x_s, y_s, ... are taken from W_s; x_r, y_r, ... are taken from W_r.

5.3.6. REMARK. In models based on Peirce frames one has

$$\begin{array}{rll} (u,v) \models \delta & \text{iff} & u = v \\ (u,v) \models \otimes\alpha & \text{iff} & (v,u) \models \alpha \\ (u,v) \models \alpha \circ \beta & \text{iff} & \exists w\, ((u,w) \models \alpha \wedge (w,v) \models \beta) \\ u \models \langle\alpha\rangle\phi & \text{iff} & \exists v\, ((u,v) \models \alpha \wedge v \models \phi) \\ (u,v) \models \updownarrow\phi & \text{iff} & u \models \phi. \end{array}$$

5.3.7. REMARK. As far as the way they handle the relational symbols is concerned, the difference between one- and two-sorted Peirce models is analogous to the difference, in ordinary first-order logic, between constants and free variables: in one-sorted Peirce models relation symbols are viewed as constants whose value is provided by the structure; in the two-sorted case they are treated as free variables with a value provided by an assignment.

There is an obvious transformation taking one-sorted Peirce models to $\mathcal{ML}_1$-equivalent two-sorted Peirce models. Let $\mathfrak{M}_1 = (W_s, (R_\alpha)_{\alpha \in Proc(\Phi,\Omega)}, V_1)$ be a one-sorted model. And let $\mathfrak{F}_2$ be the two-sorted Peirce frame over W_s. Define $\mathfrak{M}_2 = (\mathfrak{F}_2, V_2)$, where $V_2(p) = V_1(p)$ and $V_2(a) = R_a$. By a simultaneous induction on ϕ and α we have $\mathfrak{M}_1, x \models \phi$ iff $\mathfrak{M}_2, x \models \phi$, and $(x,y) \in R_\alpha$ iff $\mathfrak{M}_2, (x,y) \models \alpha$. Likewise, every two-sorted Peirce model gives rise to a one-sorted Peirce model with the same $\mathcal{ML}_1$-theory. This implies that set formulas are true on all one-sorted Peirce models iff they are valid on all frames in TPF.

5.3.8. DEFINITION. (Consequence) As in earlier Chapters (cf. §3.3 and §4.6) we adopt a *local* perspective on satisfiability and consequence. The two-sorted setting of the present Chapter calls for some comments.

To avoid messy complications we define consequence only for *one-sorted* sets of formulas Σ, and formulas ξ of the same sort (compare Remark 5.5.5). For K a class of frames we put $\Sigma \models_{\mathsf{K}} \xi$ iff for all models $(\mathfrak{F}, V)$ with $\mathfrak{F} \in \mathsf{K}$, and for every element x in $\mathfrak{F}$ of the appropriate sort:

$$(\mathfrak{F}, V), x \models \Sigma \text{ implies } (\mathfrak{F}, V), x \models \xi.$$

Further, for one-sorted sets of formulas, notions like satisfiability are defined in the usual way.

To completely characterize the Peirce frames in $\mathcal{ML}_2$ we will have to add two versions of the D-operator to our languages below, one for relations and one for sets. They are denoted with D_r and D_s, and have the obvious truth conditions:

$$x_r \models D_r\alpha \text{ iff for some } y_r \neq x_r,\ y_r \models \alpha\ (x_r, y_r \in W_r),$$
$$x_s \models D_s\phi \text{ iff for some } y_s \neq x_s,\ y_s \models \phi\ (x_s, y_s \in W_s).$$

The defined operators E with $E\xi \equiv \xi \vee D\xi$ (there exists a state with ξ), and O with $O\xi = E(\xi \wedge \neg D\xi)$ (there is only one state with ξ) will be indexed with an s or an r (cf. §3.1).

Recall from §3.3 that the axioms and rules governing the behaviour of the D-operator are

(D1)	$\overline{D}(k \to l) \to (\overline{D}k \to \overline{D}l)$, where $\overline{D} \equiv \neg D \neg$,
(D2)	$DDk \to k \vee Dk$
(D3)	$k \to \overline{D}Dk$
(NEC_D)	$\xi\,/\,\overline{D}\xi$
(IR_D)	$p \wedge \neg Dp \to \xi/\xi$, provided p does not occur in ξ.

As in Chapters 3 and 4, in the course of proving completeness results for systems involving difference operators it is sometimes convenient to use a binary relation R_D to interpret D:

$$x \models D\xi \text{ iff for some } y,\ R_Dxy \text{ and } y \models \xi.$$

Structures in which R_D does not equal $\neq$ are called *non-standard for D*.

Why use modal logic?

The remarks below will be familiar to many. Nevertheless, I feel it's worth recalling why and how systems of modal logic can be used to prove axiomatic completeness results for classes of algebraic structures.

The *complex algebra* $\mathfrak{Cm}\,\mathfrak{F}$ of a two-sorted frame $\mathfrak{F}$ is given as $\mathfrak{A} = ((2^{W_s}, -, \cap, \emptyset, W_s), (2^{W_r}, -, \cap, m_\delta, m_\otimes, m_\circ, \emptyset, W_r), m_{\langle\rangle}, m_\updownarrow)$, where, for $\#$ an n-ary modal operator, $m_\#$ is an n-ary operator on the power set(s) of the appropriate domain(s) of $\mathfrak{F}$. To be precise

$$\begin{aligned} m_\delta &= \{\, x_r : x_r \in I \,\} \\ m_\otimes(X) &= \{\, x_r : \exists y_r\, (Rx_ry_r \wedge y_r \in X \,\} \\ m_\circ(X, Y) &= \{\, x_r : \exists y_rz_r\, (Cx_ry_rz_r \wedge y_r \in X \wedge z_r \in Y)\,\} \\ m_{\langle\rangle}(X, Y) &= \{\, x_s : \exists y_rz_s\, (Px_sy_rz_s \wedge y_r \in X \wedge z_s \in Y)\,\} \\ m_\updownarrow(X) &= \{\, x_r : \exists y_s\, (Fx_ry_s \wedge y_s \in X)\,\}. \end{aligned}$$

For K a class of frames $\mathbf{Cm}(\mathsf{K})$ is the class of complex algebras of frames in K.

5.3.9. PROPOSITION. *Let $\mathfrak{F}$ be a two-sorted frame. Then $\mathfrak{F}$ is a Peirce frame (or: in* TPF*) iff $\mathfrak{Cm}\,\mathfrak{F}$ is (isomorphic) to a full Peirce algebra. In other words:* $\mathbf{Cm}(\mathsf{TPF}) = \mathsf{FPA}$.

The modal languages $\mathcal{ML}_1$ and $\mathcal{ML}_2$ have obvious algebraic counterparts. To be precise again, let Φ and Ω be as in Definition 5.3.3. Let $X_s = \{ x_{s_0}, x_{s_1}, \dots \}$ and $X_r = \{ x_{r_0}, x_{r_1}, \dots \}$ be sets of set and relation variables, respectively. Define a map $\star$ from $\mathcal{ML}_1 \cup \mathcal{ML}_2$-formulas to terms in the language of Peirce algebras over X_s and X_r by

$$\begin{array}{rclcrcl}
\star(p_i) & = & x_{s_i} & \quad & \star(a_i) & = & x_{r_i} \\
\star(\bot) & = & 0 & & \star(\mathbf{0}) & = & 0 \\
\star(\top) & = & 1 & & \star(\mathbf{1}) & = & 1 \\
\star(\neg\phi) & = & -\star(\phi) & & \star(\delta) & = & 1' \\
\star(\phi \wedge \psi) & = & \star(\phi) \cdot \star(\psi) & & \star(-\alpha) & = & -\star(\alpha) \\
\star(\langle\alpha\rangle\phi) & = & \star(\alpha) : \star(\phi) & & \star(\alpha \cap \beta) & = & \star(\alpha) \cdot \star(\beta) \\
 & & & & \star(\otimes\alpha) & = & (\star(\alpha))^{\smallsmile} \\
 & & & & \star(\alpha \circ \beta) & = & \star(\alpha);\star(\beta) \\
 & & & & \star(\updownarrow\phi) & = & (\star(\phi))^c.
\end{array}$$

The map $\star$ is extended to identities by defining $\star'(\xi)$ as $\star(\xi) = 1$ (for $\xi \in \{ \phi, \alpha \}$). As any identity in Peirce algebras can equivalently be written as an identity of the form $t = 1$, this means that we can identify modal formulas and algebraic terms.

For $\mathbf{L}$ a modal derivation system, its *algebraic counterpart* $\mathbf{L}^\star$ is defined as follows. Its axioms are all identities of the form $\star'(\phi)$ for ϕ and $\mathbf{L}$-axiom. In addition it has the usual principles from equational logic:

(Equality)	(1.) $t = t$, (2.) $s = t \,/\, t = s$, (3.) $s = t,\ t = u \,/\, s = u$
(Replacement)	$s = t \,/\, [s/x]r = [t/x]r$
(Substitution)	$s = t \,/\, [r/x]s = [r/x]t$,

as well as algebraic counterparts of any irreflexivity rules $\mathbf{L}$ might have. (The latter will have the form '$x \cdot -d_s(x) \leq t(y_0, \dots, y_n) \,/\, t(y_0, \dots, y_n) = 1$, provided x does not occur among $y_0, \dots, y_n$,' where d is some appropriate term.) The important fact, then, is the following (cf. Venema (1991, Section A7)).

5.3.10. PROPOSITION. *If* $\mathbf{L}$ *is a modal derivation system (possibly including* (IR_D)*-like rules) that is complete with respect to a class of frames* K*, then* $\mathbf{L}^\star$*, the algebraic counterpart of* $\mathbf{L}$*, is an algebraic derivation system of* $\mathbf{Cm}(\mathsf{K})$.

5.3.11. REMARK. Assume that the modal axiom systems **1-MLP** and **2-MLP** are complete for one-sorted Peirce models and two-sorted Peirce frames, respectively. Then **1-MLP***, the algebraic counterpart of **1-MLP**, axiomatizes the set identities valid in all Peirce algebras. For t a set term we have

$$\begin{array}{rll}
\mathbf{1\text{-}MLP}^\star \vdash t = 1 & \text{iff} & \mathbf{1\text{-}MLP} \vdash t \\
 & \text{iff} & \mathsf{TPF} \models t, \text{ by completeness and Remark 5.3.7,} \\
 & \text{iff} & \mathbf{Cm}(\mathsf{TPF}) \models t = 1 \\
 & \text{iff} & \mathsf{FPA} \models t = 1, \text{ by Proposition 5.3.9.}
\end{array}$$

Likewise, **2-MLP***, the algebraic counterpart of **2-MLP**, is an algebraic axiomatization of all identities valid in FPA.

5.4 Axiomatizing the set equations

Using the remarks of the previous section we exploit the completeness result for **DML** to arrive at a complete derivation system for the set equations valid in FPA.

5.4.1. Definition. Define an algebraic counterpart of the D'-operator used in **DML** by putting: $d_s(x_s) = (-1') : x_s$, for x a set term.

As pointed out before, it is an easy exercise to generalize the completeness of **DML** to an extension $\mathbf{DML}(\Phi;\Omega)$ of **DML** with a collection of propositional (set) variables Φ, and a collection of atomic relation symbols Ω (cf. the remarks following Theorem 4.6.7).

5.4.2. Definition. For X_s, X_r (countable) sets of set variables and relation variables, respectively in the algebraic language, let Φ and Ω be the corresponding sets of propositional set variables and relation variables in the modal language $\mathcal{ML}_1$ (or $\mathcal{ML}_2$).

Define $\mathbf{L}_1$ to be the algebraic counterpart of $\mathbf{DML}(\Phi;\Omega)$. That is: let $\mathbf{L}_1$be the smallest set of set equations containing the *-translations of the $\mathbf{DML}(\Phi;\Omega)$-axioms which is closed under the following closure operation:

$$x_s \cdot -d_s(x_s) \leq t(y_0,\dots,y_n) \;/\; t(y_0,\dots,y_n) = 1, \qquad (5.2)$$
$$\text{provided } x_s \text{ does not occur among } y_0,\dots,y_n,$$

as well as the usual principles from equational logic.

5.4.3. Theorem. (Algebraic Completeness) $\mathbf{L}_1$ *is a complete derivation system for the set equations valid in* FPA.

Proof. As $\mathbf{L}_1$ is the algebraic counterpart of $\mathbf{DML}(\Phi;\Omega)$, this is immediate from the completeness of $\mathbf{DML}(\Phi;\Omega)$, Theorem 4.6.7 and Remark 5.3.11. ⊣

5.5 Set equations and relation equations

In this section we axiomatize the set formulas and relation formulas valid in all frames in TPF. As the proof of the completeness of the derivation system that we come up with is somewhat long, involving several detours and technical lemmas, we have included a short 'guide' in Figure 5.1 below to help the reader find his way through the proof.

Step 1. A first approximation (page 83ff).
We define a logic **2-MLPL**, and prove it complete with respect to a class of Peirce like frames (Theorem 5.5.8).
Step 2. Towards the real thing (page 86ff).
To get to the real Peirce frames we characterize the class TPF of two-sorted Peirce frames by adding to the axioms of **2-MLPL** two axioms involving D-operators (Theorem 5.5.16).
Step 3. Completeness in an enriched language (page 89ff).
To prove completeness for a logic **2-MLPE** including the additional axioms found in Step 2 by means of the construction of Chapter 3, we enrich our language with further operators (Definition 5.5.18). Once this is done, a completeness result in the enriched language is found (Theorem 5.5.36).
Step 4. Back to the old language (page 95ff).
We translate the enriched language into our old language. The translation of the logic **2-MLPE** defined in Step 3 defines the required axiom system complete for validity in the original language (Theorem 5.5.47).

Figure 5.1: A guide to the completeness proof.

STEP 1: A FIRST APPROXIMATION

For the time being we will be working in the two-sorted language $\mathcal{ML}_2$. To characterize the two-sorted Peirce frames among the two-sorted frames, we need a number of axioms. We first list the modal axioms handling the relational component of two-sorted frames plus the conditions they impose on such frames; they are simply the modal counterparts of the earlier relation algebraic axioms (R1)–(R8), and the corresponding conditions have been calculated by Lyndon (1950) and Maddux (1982). We then list the modal counterparts of the Peirce axioms (P1)–(P8), and calculate the corresponding conditions on frames. The resulting logic is complete with respect to a class of Peirce like frames.

The first axiom states that R, the interpretation of $\otimes$, is a function; this is proved by standard arguments.

(MR0)	$\otimes a \leftrightarrow -\otimes - a$
(CR0)	R is a function

So, in frames validating (MR0) we are justified in interpreting $\otimes$ using a unary function f, and evaluating formulas $\otimes\alpha$ as follows

$$\mathfrak{M}, x_r \models \otimes\alpha \text{ iff } \mathfrak{M}, f(x_r) \models \alpha.$$

5.5.1. DEFINITION. A *two-sorted arrow frame* is simply a two-sorted frame $\mathfrak{F} = (W_s, W_r, I, f, C, F, P)$ in which the binary relation R used to interpret the operator $\otimes$ is a function from W_r to W_r, denoted by f. A *two-sorted arrow* *model* is a two-sorted model based on a two-sorted arrow frame, where $\otimes$ is interpreted using the function f as indicated above.

Here are the remaining axioms governing the behaviour of δ, $\otimes$ and $\circ$, as well as the conditions expressed by these axioms.

(MR1)	$a \rightarrow \otimes\otimes a$
(CR1)	$f(f(x_r)) = x_r$
(MR2)	$a \circ (b \circ c) \rightarrow (a \circ b) \circ c$
(CR2)	$\forall y_r z_r u_r v_r\, (Cx_r y_r z_r \wedge Cz_r u_r v_r \rightarrow \exists w_r\, (Cx_r w_r v_r \wedge Cw_r y_r u_r))$
(MR3)	$(a \circ b) \circ c \rightarrow a \circ (b \circ c)$
(CR3)	$\forall y_r w_r u_r v_r\, (Cx_r w_r v_r \wedge Cw_r y_r u_r \rightarrow \exists z_r\, (Cx_r y_r z_r \wedge Cz_r u_r v_r))$
(MR4)	$a \rightarrow \delta \circ a,\ a \rightarrow a \circ \delta$
(CR4)	$\exists y_r\, (Iy_r \wedge Cx_r y_r x_r),\ \exists y_r\, (Cx_r x_r y_r \wedge Iy_r)$
(MR5)	$\delta \circ a \rightarrow a,\ a \circ \delta \rightarrow a$
(CR5)	$\forall y_r z_r\, (Cx_r y_r z_r \wedge Iy_r \rightarrow x_r = z_r),\ \forall y_r z_r\, (Cx_r y_r z_r \wedge Iz_r \rightarrow x_r = y_r)$
(MR6)	$\otimes(a \circ b) \rightarrow (\otimes b \circ \otimes a)$
(CR6)	$\forall y_r z_r\, (Cf(x_r) y_r z_r \rightarrow Cx_r f(z_r) f(y_r))$
(MR7)	$(\otimes b \circ \otimes a) \rightarrow \otimes(a \circ b)$
(CR7)	$\forall y_r z_r\, (Cx_r f(z_r) f(y_r) \rightarrow Cf(x_r) y_r z_r)$
(MR8)	$\otimes a \circ -(a \circ b) \cap b \rightarrow \mathbf{0}$
(CR8)	$\forall y_r z_r\, (Cx_r f(y_r) z_r \rightarrow Cz_r y_r x_r)$.

Next come the axioms governing the behaviour of the Peirce product and cylindrification.

(MP1)	$\langle a\rangle\langle b\rangle p \rightarrow \langle a \circ b\rangle p$
(CP1)	$\forall y_r y'_r z_s z'_s\, (Px_s y_r z_s \wedge Pz_s y'_r z'_s \rightarrow \exists y''_r\, (Px_s y''_r z'_s \wedge Cy''_r y_r y'_r))$
(MP2)	$\langle a \circ b\rangle p \rightarrow \langle a\rangle(\langle b\rangle p)$
(CP2)	$\forall y_r y'_r y''_r z_s\, (Px_s y_r z_s \wedge Cy_r y'_r y''_r \rightarrow \exists z'_s\, (Px_s y'_r z'_s \wedge Pz'_s y''_r z_s))$
(MP3)	$\langle\delta\rangle p \rightarrow p$
(CP3)	$\forall y_r z_s\, (Px_s y_r z_s \wedge Iy_r \rightarrow x_s = z_s)$
(MP4)	$p \rightarrow \langle\delta\rangle p$
(CP4)	$\exists y_r\, (Px_s y_r x_s \wedge Iy_r)$
(MP5)	$\langle\otimes a\rangle\neg\langle a\rangle p \wedge p \rightarrow \bot$
(CP5)	$\forall y_r z_s\, (Px_s y_r z_s \rightarrow Pz_s f(y_r) x_s)$
(MP6)	$\langle\updownarrow p\rangle\top \rightarrow p$
(CP6)	$\forall y_r z_s z'_s\, (Px_s y_r z_s \wedge Fy_r z'_s \rightarrow x_s = z'_s)$
(MP7)	$p \rightarrow \langle\updownarrow p\rangle\top$
(CP7)	$\exists y_r z_s\, (Px_s y_r z_s \wedge Fy_r x_s)$
(MP8)	$\updownarrow\langle a\rangle\top \rightarrow (a \circ \mathbf{1})$
(CP8)	$\forall y_s y'_r z_s\, (Fx_r y_s \wedge Py_s y'_r z_s \rightarrow \exists z'_r\, (Cx_r y'_r z'_r))$
(MP9)	$(a \circ \mathbf{1}) \rightarrow \updownarrow\langle a\rangle\top$
(CP9)	$\forall y_r z_r\, (Cx_r y_r z_r \rightarrow \exists y'_s z'_s\, (Fx_r y'_s \wedge Py'_s y_r z'_s))$.

5.5.2. REMARK. The proof that the above axioms (MRi) and (MPi) correspond to the conditions (CMi) and (CPi) follows from the general results of Chapter 7: all axioms listed here are so-called Sahlqvist formulas, and for such formulas there is an explicit algorithm computing the corresponding relational condition (cf. §7.5). Let us give one example here of a correspondence result: axiom (MP5). For any two-sorted arrow frame $\mathfrak{F}$ and x_s in $\mathfrak{F}$ we have

$$\begin{aligned}
&\mathfrak{F}, x_s \models \text{(MP5)} \\
&\quad \text{iff} \quad \mathfrak{F}, x_s \models \forall a \forall p \Big(\exists y_r z_s \left(P x_s y_r z_s \wedge a(f(y_r)) \wedge \right. \\
&\qquad\qquad\qquad \left. \neg\exists y'_r z'_s \left(P z_s y'_r z'_s \wedge a(y'_r) \wedge p(z'_s)\right) \wedge p(x_s)\right) \rightarrow \bot(x_s)\Big) \\
&\quad \text{iff} \quad \mathfrak{F}, x_s \models \forall a \forall p \forall y_r z_s \Big(P x_s y_r z_s \wedge a(f(y_r)) \wedge p(x_s) \rightarrow \\
&\qquad\qquad\qquad \exists y'_r z'_s \left(P z_s y'_r z'_s \wedge a(y'_r) \wedge p(z'_s)\right)\Big)
\end{aligned}$$

Now, substituting $\lambda u.\, u = f(y_r)$ for a, and $\lambda u.\, u = x_s$ for p, this yields

$$\forall y_r z_s \Big(P x_s y_r z_s \rightarrow \exists y'_r z'_s \left(P z_s y'_r z'_s \wedge y'_r = f(y_r) \wedge z'_s = x_s\right)\Big),$$

which is equivalent to $\forall y_r z_s \left(P x_s y_r z_s \rightarrow P z_s f(y_r) x_s\right)$.

5.5.3. DEFINITION. A two-sorted arrow frame is *Peirce like* if it satisfies conditions (CR1)–(CR8), as well as (CP1)–(CP9). The class of Peirce like frames is denoted by TPLF.

5.5.4. DEFINITION. Let **2-MLPL** be the minimal modal axiom system for the language $\mathcal{ML}_2(\delta, \circ, \otimes, \langle\rangle, \updownarrow)$ extended with (MR0)–(MR8) and (MP1)–(MP9) as axioms. So, besides (MR0)–(MR8) and (MP1)–(MP9), **2-MLPL** has the Boolean axioms for $\neg$, $\wedge$, $\bot$, $\top$; the Boolean axioms for $-$, $\cap$, $\mathbf{0}$, $\mathbf{1}$; and distribution axioms for the modal operators:

$$\begin{array}{lcl}
\otimes & : & \overline{\otimes}(a \rightarrow b) \rightarrow (\overline{\otimes} a \rightarrow \overline{\otimes} b), \text{ where } \overline{\otimes}\alpha \equiv -\otimes-\alpha \\
\circ & : & (a \rightarrow b)\,\bar{\circ}\, c \rightarrow ((a \,\bar{\circ}\, c) \rightarrow (b \,\bar{\circ}\, c)), \text{ where } \alpha \,\bar{\circ}\, \beta \equiv -(-\alpha \circ -\beta) \\
\circ & : & a \,\bar{\circ}\, (b \rightarrow c) \rightarrow ((a \,\bar{\circ}\, b) \rightarrow (a \,\bar{\circ}\, c)) \\
\langle\rangle & : & [\![a \rightarrow b]\!] p \rightarrow ([\![a]\!] p \rightarrow [\![b]\!] p), \text{ where } [\![\alpha]\!]\phi \equiv \neg\langle -\alpha\rangle\neg\phi \\
\langle\rangle & : & [\![a]\!](p \rightarrow q) \rightarrow ([\![a]\!] p \rightarrow [\![a]\!] q) \\
\updownarrow & : & \bar{\updownarrow}(p \rightarrow q) \rightarrow (\bar{\updownarrow} p \rightarrow \bar{\updownarrow} q), \text{ where } \bar{\updownarrow}\phi \equiv -\updownarrow\neg\phi.
\end{array}$$

In addition **2-MLPL** has the derivation rules modus ponens (MP), substitution (SUB), and necessitation (NEC), for all modal operators. The latter covers the following:

$(\text{NEC}_\otimes)$	$\alpha \,/\, \overline{\otimes}\alpha$	$(\text{NEC}_\updownarrow)$	$\phi \,/\, \bar{\updownarrow}\phi$
$(\text{NEC}_{\langle\rangle})$	$\alpha \,/\, [\![\alpha]\!]\phi$	$(\text{NEC}_{\langle\rangle})$	$\phi \,/\, [\![\alpha]\!]\phi$
$(\text{NEC}_\circ)$	$\alpha \,/\, \alpha \,\bar{\circ}\, \beta$	$(\text{NEC}_\circ)$	$\beta \,/\, \alpha \,\bar{\circ}\, \beta.$

5.5.5. DEFINITION. (Derivations) For **L** a (two-sorted) modal logic we define an **L**-*derivation* to be a list of formulas from the language of **L** such that every formula is either a substitution instance of an axiom of **L**, or obtained from earlier formulas in the list by means of a derivation rule of **L**. An **L**-*theorem* is any formula that can occur as the last item in a derivation. We write $\vdash_{\mathbf{L}} \xi$

for ξ is an **L**-theorem, and $\Sigma \vdash_{\mathbf{L}} \xi$ for: there are $\sigma_1, \ldots, \sigma_n \in \Sigma$ such that $\vdash_{\mathbf{L}} (\sigma_1 \wedge \ldots \wedge \sigma_n) \to \xi$ (if ξ is a set formula), or $\vdash_{\mathbf{L}} (\sigma_1 \cap \ldots \cap \sigma_n) \to \xi$ (if ξ is a relation formula). (Compare Remark 5.3.8.)

For Δ a set of formulas, Δ_{set} denotes the set of set formulas in Δ, and Δ_{rel} denotes the set of relation formulas in Δ.

5.5.6. PROPOSITION. *Let $\Delta \cup \{\xi\}$ be an $\mathcal{ML}_2$-formula.*

1. *If ξ is a set formula, then $\Delta \vdash \xi$ iff $\Delta_{set} \vdash \xi$.*
2. *If ξ is a relation formula, then $\Delta \vdash \xi$ iff $\Delta_{rel} \vdash \xi$.*

5.5.7. LEMMA. *Let $\mathfrak{F}$ be a two-sorted arrow frame. Then $\mathfrak{F} \models$ (MPi) iff $\mathfrak{F} \models$ (CPi), for $1 \leq i \leq 9$.*

5.5.8. THEOREM. **2-MLPL** *is strongly sound and complete for* TPLF.

Proof. This may be established in at least two ways. One may use the standard canonical model construction — not the one used in Chapters 3, 4 for logics containing (IR$_D$)-like rules. Or one may recall that all **2-MLPL** are Sahlqvist formulas, and derive immediately that **2-MLPL** is complete with respect to the two-sorted Peirce like frames satisfying the conditions (CRi) and (CPi). ⊣

STEP 2: CHARACTERIZING TWO-SORTED PEIRCE FRAMES

We now narrow down the two-sorted Peirce like frames to two-sorted Peirce frames. To this end we first find modal formulas (in a language extending the language of **2-MLPL**) that completely characterize the class TPF of two-sorted Peirce frames. Roughly speaking, what we need to know about a two-sorted Peirce like frame in order to be able to show that it is in fact a two-sorted Peirce frame, is that

- with every relational element we can associate a unique set element as its first coordinate and a unique set element as its second coordinate;
- with every two set elements we can associate a unique relational element having those set elements as first and second coordinate.

In terms of the ingredients of our two-sorted Peirce like frames this boils down to having the following conditions satisfied by our Peirce like frames:

(CP10) $\forall x_r y_s y'_s\, (F x_r y_s \wedge F x_r y'_s \to y_s = y'_s)$
(CP11) $\forall x_r y_s y'_s\, (Ff(x_r) y_s \wedge Ff(x_r) y'_s \to y_s = y'_s)$
(CP12) $\forall x_r \exists y_s\, (F x_r y_s)$
(CP13) $\forall x_r \exists y_s\, (Ff(x_r) y_s)$
(CP14) $\forall x_s y_s \exists z_r\, (P x_s z_r y_s)$
(CP15) $\forall x_s y_s z_r z'_r\, (P x_s z_r y_s \wedge P x_s z'_r y_s \to z_r = z'_r)$.

5.5.9. LEMMA. *Let $\mathfrak{F} \in$ TPLF. Then $\mathfrak{F} \models$* (CP10)–(CP13).

Proof. We only prove that for $\mathfrak{F} \in$ TPLF, $\mathfrak{F} \models$ (CP1), leaving the other cases to the reader. Assume $F x_r y_s$ and $F x_r y'_s$. By (CR4), (CP9) there exist y_r, y''_s, z''_s with $C x_r x_r y_r$, $F x_r y''_s$ and $P y''_s x_r z''_s$. By (CP6) $P y''_s x_r z''_s$ and $F x_r y_s$ imply

$y''_s = y_s$. Similarly, $y''_s = y_s$. Hence $y_s = y'_s$. $\dashv$

Observe that conditions (CP10)–(CP13) are expressed by the following four modal formulas, respectively:

(MP10) $\updownarrow p \cap \updownarrow q \to \updownarrow(p \wedge q)$
(MP11) $\otimes(\updownarrow p) \wedge \otimes(\updownarrow q) \to \otimes\updownarrow(p \wedge q)$
(MP12) $\updownarrow\top$
(MP13) $\otimes(\updownarrow\top)$.

The proof of this claim is left to the reader.

5.5.10. LEMMA. *Let $\mathfrak{F}$ be a two-sorted Peirce like frame. Then*

1. $\mathfrak{F} \models \forall x_s y_s z_r\,(Px_s z_r y_s \to Fz_r x_s \wedge Ff(z_r)y_s)$, *and*
2. $\mathfrak{F} \models \forall x_s y_s z_r\,(Fz_r x_s \wedge Ff(z_r)y_s \to Px_s z_r y_s)$.

Proof. To prove (1) assume $Px_s z_r y_s$. By (CP12) $Fz_r x'_s$, for some x'_s. By (CP6) $x_s = x'_s$, hence $Fz_r x_s$. Likewise, by (CP13), (CP5) and (CP6) we have $Ff(z_r)y_s$. For (2), assume that $Fz_r x_s$, $Ff(z_r)y_s$. By (CR4) there exists y_r with $Cz_r z_r y_r$. By (CP9) this implies there exist y'_s, z'_s with $Py'_s z_r z'_s$. By (i) $Fz_r y'_s$ and $Ff(z_r)z'_s$. (CP10) and (CP11) then yield $x_s = y'_s$ and $y_s = z'_s$. Hence $Px_s z_r y_s$. $\dashv$

5.5.11. REMARK. Recall that the two-sorted Peirce frame over a set U is the structure (U, U^2, I, R, C, F, P) with $I = \delta$, $R = \{((u_1, v_1), (u_2, v_2)) \in {}^2(U \times U) : u_1 = v_2 \wedge u_2 = v_1\}$, $C = \{((u_1, v_1), (u_2, v_2), (u_3, v_3)) \in {}^3(U \times U) : u_1 = u_2 \wedge v_1 = v_3 \wedge v_2 = u_3\}$, $F = \{((u_1, v_1), u_2) \in (U \times U) \times U : u_1 = u_2\}$, and $P = \{(u_1, (u_2, v_2), u_3) \in U \times (U \times U) \times U : u_1 = u_2 \wedge v_2 = u_3\}$, as defined in Definition 5.3.4.

The representation we are about to present below is more elegant than the usual representations in relation algebra and arrow logic because we can map every relation point z_r in a Peirce frame onto a pair of set points x_s, y_s already present in the Peirce frame, rather than on points extracted from (a Cartesian product of) the diagonal.

5.5.12. THEOREM. *Let $\mathfrak{F} = (W_s, W_r, I, f, C, F, P)$ be a two-sorted Peirce like frame. If $\mathfrak{F} \models$ (CP14), (CP15), then $\mathfrak{F}$ is isomorphic to the two-sorted Peirce frame over W_s.*

Proof. If $\mathfrak{F}$ is a Peirce like frame satisfying (CP14) and (CP15), then, with every $z_r \in W_r$ we can associate a *unique* x and y such that $Fz_r x$ and $Ff(z_r)y$. Define a mapping $g : W_r \to W_s \times W_s$ by $g(z) = (z_0, z_1)$, where z_0, z_1 are the unique x and y with $Fz_r x$ and $Ff(z_r)y$. We prove that g is an isomorphism.

g is surjective. Let x_s, $y_s \in W_s$. By (CP14) $Px_s z_r y_s$, for some z_r. By Lemma 5.5.10 $Fz_r x_s$ and $Ff(z_r)y_s$. Hence $g(z) = (x_s, y_s)$.

g is injective. Let z_r, $z'_r \in W_r$, and assume $g(z_r) = g(z'_r)$. Then, for some x_s, y_s we have $Fz_r x_s$, $Ff(z_r)y_s$, and $Fz'_r x_s$, $Ff(z'_r)y_s$. By Lemma 5.5.10 this implies $Px_s z_r y_s$ and $Px_s z'_r y_s$. Hence, by (CP15) $z_r = z'_r$.

g is a homomorphism. To establish this claim we need to consider 5 cases: I, f,

C, P, F. Here we go.
I: let $z_r \in I$; we need to show that $g(z_r) = (x_s, x_s)$ for some x_s. Choose x_s, y_s such that $g(z_r) = (x_s, y_s)$. By definition $Fz_r x_s$, $Ff(z_r)y_s$ and so $Px_s z_r y_s$ by Lemma 5.5.10. By (CP3) this gives $x_s = y_s$.
f: we need to show that $f(g(z_r)) = g(f(z_r))$. If $g(z_r) = (x_s, y_s)$, then $Px_s z_r y_s$), and, by (CP5), $Pf(z_r)y_s x_s$. Hence, $g(f(z_r)) = (y_s, x_s) = f(g(z))$.
C: we need to show that $Cx_r y_r z_r$ implies that $g(x_r)$ is the composition of $g(y_r)$ and $g(z_r)$. That is: if $g(x_r) = (x_0, x_1)$, $g(y_r) = (y_0, y_1)$, $g(z_r) = (z_0, z_1)$, then $x_0 = y_0$, $y_1 = z_0$, $z_1 = x_1$. Observe that by (CP2) we have $Px_0 y_r z'$, $Pz' z_r x_1$, for some z'. By Lemma 5.5.10, (CP5), (CP10) and (CP11) this implies the three identities.
F: here we need to show that $Fz_r x_s$ implies that if $g(z) = (z_0, z_1)$ then $z_0 = x_s$. But this is immediate from the definition of g and (CP10).
P: we need to show that $Px_s z_r y_s$ implies $g(z) = (x_s, y_s)$; this is immediate by Lemma 5.5.10.

g^{-1} *is a homomorphism.* Again, this requires us to consider 5 cases.
I: we need to show that whenever $g(z_r) = (x_s, x_s)$, then $z_r \in I$. If $g(z_r) = (x_s, x_s)$, then $Px_s z_r x_s$. By (CP4) there is a y_r such that $Px_s y_r x_s$ and Iy_r. By (CP15) this implies $y_r = z_r$; hence Iz_r.
f: this has already been proved above.
C: assume $g(x_r)$ is the composition of $g(y_r)$ and $g(z_r)$, that is, assume $g(x_r) = (x_0, x_1)$, $g(y_r) = (y_0, y_1)$, $g(z_r) = (z_0, z_1)$; We need to show that $Cx_r y_r z_r$. By definition $x_0 = y_0$, $y_1 = z_0$, $z_1 = x_1$; so $Px_0 y_r z_0$ and $Pz_0 z_r x_1$. By (CM1) the latter implies that for some u_r, $Px_0 u_r x_1$ and $Cu_r y_r z_r$. By (CP15) $u_r = x_r$, hence $Cx_r y_r z_r$.
F: assume $g(z_r) = (x_s, y_s)$; we need to show that $Fz_r x_s$; but this is immediate from the definitions.
P: assume that $g(z_r) = (x_s, y_s)$; we have to show that $Px_s z_r y_s$. But $g(z_r) = (x_s, y_s)$ implies $Fz_r x_s$ and $Ff(z_r)y_s$; now apply Lemma 5.5.10. ⊣

5.5.13. COROLLARY. *Let $\mathfrak{F}$ be a two-sorted arrow frame. Then*

$$\mathfrak{F} \in \mathsf{TPF} \quad \textit{iff} \quad \mathfrak{F} \models \text{(CR1)–(CR8), (CP1)–(CP9), (CP14), (CP15)}.$$

Recall from §5.3 that the operator E_s is short for $E_s p \equiv p \vee D_s p$ (there exists a state where p holds), and that the operator O_s is short for $O_s p \equiv E_s(p \wedge \neg D_s p)$ (there exists only one state with p).

5.5.14. DEFINITION. We define the following two formulas:

(MP14) $E_s p \to \langle \mathbf{1} \rangle p$
(MP15) $E_s O_s p \wedge \langle a \rangle p \wedge \langle b \rangle p \to \langle a \cap b \rangle p$.

5.5.15. LEMMA. *Let $\mathfrak{F}$ be a two-sorted Peirce like frame. Then $\mathfrak{F}$ satisfies* (CP14) *iff it validates* (MP14)*; it satisfies* (CP15) *iff it validates* (MP15).

Proof. We only prove that (CP15) is defined by (MP15). Assume $\mathfrak{F} \not\models$ (CP15). Then there are z_r, z'_r, x_s, y_s such that $Px_s z_r y_s$ and $Px_s z'_r y_s$, but $z_r \neq z'_r$.

Defining a valuation V such that $V(p) = \{ y_s \}$, $V(a) = \{ z_r \}$, $V(b) = \{ z'_r \}$ refutes (MP15) at x_s.

For the converse, if $\mathfrak{F} \not\models$ (MP15), then for some valuation V and x_s in $\mathfrak{F}$ we have $x_s \models E_s O_s p \land \langle a \rangle p \land \langle b \rangle p$ and $x_s \not\models \langle a \cap b \rangle p$. Hence, there exists a unique y_s in $\mathfrak{F}$ with $y_s \models p$, and there exist z_r, z'_r with $Px_s z_r y_s$, $Px_s z'_r y_s$ and $z_r \models a$, $z'_r \models b$. As $x_s \not\models \langle a \cap b \rangle p$, we must have $z_r \neq z'_r$. So $\mathfrak{F} \not\models$ (CP15). ⊣

5.5.16. THEOREM. $\mathsf{TPF} = \{ \mathfrak{F} : \mathfrak{F} \models \bigwedge_{0 \leq i \leq 8} (\mathrm{MR}i) \land \bigwedge_{0 \leq i \leq 9} (\mathrm{MP}i) \land (\mathrm{MP14}) \land (\mathrm{MP15}) \}$.

Proof. This follows from 5.5.7, 5.5.13 and 5.5.15. ⊣

STEP 3: COMPLETENESS IN AN ENRICHED LANGUAGE

Now that we have characterized the two-sorted Peirce frames, we can start working our way towards complete axiomatizations of TPF and FPA. We will use the completeness construction of Chapter 3 to arrive at the result. To this end we need to (re-) establish a Switching, Pasting, Extension, Successor, and Structure Lemma, define provisional and final canonical models, and derive a Truth Lemma and, finally, Completeness Theorem (compare §4.6). We will achieve this by adding further modal operators as well as appropriate inclusion axioms for all modal operators. The additional modal operators will be removed in Step 4 of the completeness proof (page 95ff).

The first things we add to the language $\mathcal{ML}_2(\delta, \otimes, \circ, \langle\rangle, \updownarrow; \Phi; \Omega)$ are two difference operators D_s and D_r, which can be applied to set formulas and relation formulas, respectively. Let $\mathcal{ML}_2(\delta, \otimes, \circ, \langle\rangle, \updownarrow, D_s, D_r; \Phi; \Omega)$, or $\mathcal{ML}_2^{\neq}$, be the resulting language.

For our next addition we need the following.

5.5.17. DEFINITION. Let R be an $(n+1)$-ary relation. A frame $\mathfrak{F} = (\ldots, R, \ldots)$ is called *versatile for R* if there are relations $R_1, \ldots, R_n$ such that for all $x_0, \ldots, x_n$ one has $(x_0, \ldots, x_n) \in R$ iff $(x_1, \ldots, x_n, x_0) \in R_1$ iff ... iff $(x_n, x_0, \ldots, x_{n-1}) \in R_n$. Once we know that a frame is versatile for R, it suffices to mention just a single R, and suppressing the relations that constitute the versatility.

Let $\#$ be an n-ary modal operator whose semantics is based on an $(n+1)$-ary relation R; the *conjugates of* $\#$ are n operators $\#_1, \ldots, \#_n$ whose semantics are based on $(n+1)$-ary relations $R_1, \ldots, R_n$, respectively, such that $R, R_1, \ldots, R_n$ form a versatile system, and

$$x \models \#_i(\xi_1, \ldots, \xi_n) \text{ iff } \exists y_1 \ldots y_n \, (Rxy_1 \ldots y_n \land \textstyle\bigwedge_i y_i \models \xi_i).$$

Unary modal operators whose underlying relation is symmetric form their own conjugates; also, a frame is versatile for a binary relation B if it contains the converse B^{-1} of B.

5.5.18. DEFINITION. We define the language $\mathcal{ML}_2^{\neq\mathrm{sat}}$ by saturating $\mathcal{ML}_2^{\neq}$, that is by adding to it conjugates for all of its modal operators. For the time being, fix a two-sorted Peirce like frame $\mathfrak{F}$.

As D_s, D_r, $\otimes$ are self-conjugated we don't need to add conjugates for them. For $\updownarrow$ we add a conjugate $\Updownarrow$ to be interpreted on a binary relation $G \subseteq W_s \times W_r$; $\Updownarrow$ takes a relation formula and returns a set formula.

For $\langle\cdot\rangle\cdot$ we add two operators $\langle\cdot\rangle_1$ and $\langle\cdot\rangle_2\cdot$, interpreted using ternary relations $P_1 \subseteq W_r \times W_s \times W_s$, and $P_2 \subseteq W_s \times W_s \times W_r$.

For $\circ$ we also add two operators, written $\circ_1$ and $\circ_2$, to be interpreted using ternary relations C_1, $C_2 \subseteq {}^3W_r$.

We force the appropriate modal operators to be each others conjugates by imposing the axioms below; the corresponding relational requirements are also listed. Let $\bar{\updownarrow}\phi$ abbreviate $-\updownarrow\neg\phi$, and $\bar{\Updownarrow}\alpha$ abbreviate $\neg\Updownarrow-\alpha$.

(MP16)	$a \to \bar{\updownarrow}\Updownarrow a$
(CP16)	$\forall x_s\,(Fz_rx_s \to Gx_sz_r)$
(MP17)	$p \to \bar{\Updownarrow}\updownarrow p$
(CP17)	$\forall z_r\,(Gx_sz_r \to Fz_rx_s)$
(MP18)	$p \wedge \langle -\langle q\rangle_1 p\rangle q \to \bot$
(CP18)	$\forall y_rz_s\,(Px_sy_rz_s \to P_1y_rz_sx_s)$
(MP19)	$a \cap \langle\neg\langle p\rangle_2 a\rangle_1 p \to \mathbf{0}$
(CP19)	$\forall x_sz_s\,(P_1y_rz_sx_s \to P_2z_sx_sy_r)$
(MP20)	$p \wedge \langle\neg\langle a\rangle p\rangle_2 a \to \bot$
(CP20)	$\forall x_sy_r\,(P_2z_sx_sy_r \to Px_sy_rz_s)$
(MP21)	$a \cap -(b \circ_1 a) \circ b \to \mathbf{0}$
(CP21)	$\forall y_rz_r\,(Cx_ry_rz_r \to C_1y_rz_rx_r)$
(MP22)	$a \cap -(b \circ_2 a) \circ_1 b \to \mathbf{0}$
(CP22)	$\forall z_rx_r\,(C_1y_rz_rx_r \to C_2z_rx_ry_r)$
(MP23)	$a \cap -(b \circ a) \circ_2 b \to \mathbf{0}$
(CP23)	$\forall x_ry_r\,(C_2z_rx_ry_r \to Cx_ry_rz_r)$.

5.5.19. DEFINITION. In the saturated language $\mathcal{ML}_2^{\neq\mathrm{sat}}$ we define the two-sorted modal logic **2-MLPE**. Its axioms and rules are those of **2-MLPL** plus axioms (MP14)–(MP23), plus the D-axioms for both D_s and D_r; in addition it has distribution axioms and necessitation rules for all modal operators, as well as the irreflexivity rules for both D_s and D_r. Finally, it also has the following inclusion axioms:

(INC1)	$\langle a\rangle p \to E_sp$
(INC2)	$\otimes a \to E_ra$
(INC3)	$a \circ b \to E_ra \wedge E_rb$
(INC4)	$\langle D_r(\updownarrow q)\rangle\top \to E_sq$
(INC5)	$\updownarrow(D_s\langle a\rangle\top) \to E_ra$.

5.5.20. PROPOSITION. *The inclusion axioms* (INC1)–(INC5) *correspond to the following relational conditions. (We have used non-standard interpretations for* *the difference operators here; the standard relational correspondents may be obtained from the ones below by reading $\neq$ for R_{D_s} and R_{D_r}.)*

(CINC1) $\forall y_r z_s\,(Px_s y_r z_s \to x_s = z_s \vee R_{D_s} x_s z_s)$
(CINC2) $\forall y_r\,(R_\otimes x_r y_r \to x_r = y_r \vee R_{D_r} x_r y_r)$
(CINC3) $\forall y_r z_r\,(Cx_r y_r z_r \to (x_r = y_r \vee R_{D_r} x_r y_r) \wedge (x_r = z_r \vee R_{D_r} x_r z_r))$
(CINC4) $\forall y_r z_s y'_r x'_s\,(Px_s y_r z_s \wedge R_{D_r} y_r y'_r \wedge Fy'_r x'_s \to x_s = x'_s \vee R_{D_s} x_s x'_s)$
(CINC5) $\forall y_s z_s x'_r y'_s\,(Fx_r y_s \wedge R_{D_s} y_s z_s \wedge Pz_s x'_r y'_s \to x_r = x'_r \vee R_{D_r} x_r x'_r)$.

5.5.21. LEMMA. *Any Peirce-like frame that validates* (INC4) *and* (INC5) *has the properties*

1. $\forall x_s y_r z_r x'_s\,(Fy_r x_s \wedge (y_r = z_r \vee R_{D_r} y_r z_r) \wedge Fz_r x'_s \to x_s = x'_s \vee R_{D_s} x_s x'_s)$, *and*
2. $\forall x_r y_s z_s x'_r\,(Fx_r y_s \wedge (y_s = z_s \vee R_{D_s} y_s z_s) \wedge Fx'_r z_s \to x_r = x'_r \vee R_{D_r} x_r x'_r)$.

Proof. For the first property, assume that $Fy_r x_s$, $y_r = z_r$, $Fz_r x'_s$. Then $Fz_r x_s$ and $Fz_r x'_s$, so by (CP10) $x_s = x'_s$. Next, assume $Fy_r x_s$, $R_{D_r} y_r z_r$, $Fz_r x'_s$. Then, by (CP13) and Lemma 5.5.10 there exists z_s with $Px_s y_r z_s$. By (INC4) this implies $x_s = x'_s$ or $R_{D_s} x_s x'_s$, as required.

For the second property, assume $Fx_r y_s$, $y_s = z_s$, $Fx'_r z_s$. By (CP13) and Lemma 5.5.10 $Py_s x_r z'_s$ for some z'_s. By (MP5) $Pz'_s f(x_r) y_s$. Furthermore, for some u_s, $Py_s x'_r u_s$, so by (CP1) there exists y''_r with $Cy''_r x_r x'_r$. By (INC3) and the pseudo-transitivity of R_{D_r} this gives $x_r = x'_r$ or $R_{D_r} x_r x'_r$. Finally, if $Fx_r y_s$, $R_{D_s} y_s z_s$, $Fx'_r z_s$, choose some y'_s with $Pz_s x'_r y'_s$ ((CP12) and 5.5.10). Then (INC5) yields $x_r = x'_r$ or $R_{D_r} x_r x'_r$. $\dashv$

We need to make a few adjustments to the completeness construction of Chapter 3, mainly having to do with the presence of binary modal operators.

THE `Paste` FUNCTION. We extend Definition 3.3.5 to the present setting by adding clauses for the modal operators in $\mathcal{ML}_2^{\neq\mathrm{sat}}$ — either analogous to the case for $\cap$ below, or analogous to the case for $\updownarrow$ below.

$$\begin{aligned}\texttt{Paste}(\nu,\xi,\alpha\cap\beta) &= \texttt{Paste}(\nu,\xi,\alpha)\cap\texttt{Paste}(\nu,\xi,\beta)\\ \texttt{Paste}(\nu,\xi,\updownarrow\phi) &= \updownarrow\texttt{Paste}(\nu,\xi,\phi).\end{aligned}$$

We assume that `Paste` respects the two-sorted nature of the language in that a 'name' ν is only pasted next to a formula of the same sort; for example, when writing $\texttt{Paste}(\nu,\alpha,\alpha) = \nu\cap\alpha$, we assume that ν is a relation formula.

5.5.22. LEMMA. (Switching Lemma.) *Let* $\vdash$ *denote* $\vdash_{\mathbf{2\text{-}MLPE}}$. *The following are derived rules in* **2-MLPE**.

1. $\vdash D_s\phi \to \psi$ *iff* $\vdash \phi \to \overline{D}_s\psi$, $\vdash D_r\phi \to \psi$ *iff* $\vdash \phi \to \overline{D}_r\psi$
2. $\vdash \Updownarrow\alpha \to \psi$ *iff* $\vdash \alpha \to \overline{\updownarrow}\psi$, $\vdash \Updownarrow\phi \to \beta$ *iff* $\vdash \phi \to \overline{\Updownarrow}\beta$
3. $\vdash \neg(p \wedge \langle a\rangle q)$ *iff* $\vdash -(a \cap \langle q\rangle_1 p)$ *iff* $\vdash \neg(q \wedge \langle p\rangle_2 a)$
4. $\vdash -(a \cap (b \circ c))$ *iff* $\vdash -(b \cap (c \circ_1 a))$ *iff* $\vdash -(c \cap (a \circ_2 b))$.

Proof. Items 1 and 2 are easy (see for example Lemma 3.3.6). Item 3 is similar to item 4, which is proved by Venema (1991, Corollary 2.7.5) $\dashv$

5.5.23. LEMMA. (Pasting Lemma) *Let $O_t k$ be one of $O_s p$, $O_r a$, and assume that $O_t k$ has no atomic symbols in common with ξ and θ. For any subformula occurrence $\xi' \trianglelefteq \xi$ we have that* $\vdash \mathtt{Paste}(O_t k, \xi', \xi) \to \theta$ *implies* $\vdash \xi \to \theta$. *So if Σ is consistent, k does not occur in ξ, and $\xi' \trianglelefteq \xi$, then $\Sigma \cup \{ \mathtt{Paste}(O_t k, \xi', \xi) \}$ is consistent.*

Proof. The latter half of the Lemma is immediate from the former half. The former half, in turn, is proved by induction on ξ as in Lemma 3.3.7. We only prove one of the 'new' cases: $\xi \equiv (\langle\alpha\rangle\phi)$. Note that $\mathtt{Paste}(O_t k, \xi', (\langle\alpha\rangle\phi) = \langle \mathtt{Paste}(O_t k, \xi', \alpha)\rangle \mathtt{Paste}(O_t k, \xi', \phi)$. We distinguish two cases: $\xi' \trianglelefteq \alpha$ and $\xi' \trianglelefteq \phi$. In the first case we have that $\mathtt{Paste}(O_t k, \xi', \phi) \equiv \phi$, and hence

$$\begin{aligned}
&\vdash \big\langle \mathtt{Paste}(O_t k, \xi', \alpha)\big\rangle \mathtt{Paste}(O_t k, \xi', \phi) \to \theta \\
&\quad\Rightarrow\quad \big\langle \mathtt{Paste}(O_t k, \xi', \alpha)\big\rangle \phi \to \theta, \\
&\quad\Rightarrow\quad \mathtt{Paste}(O_t k, \xi', \alpha) \to -\langle\phi\rangle_1 \neg\theta, \quad \text{by the Switching Lemma} \\
&\quad\Rightarrow\quad \alpha \to -\langle\phi\rangle_1 \neg\theta, \quad \text{by the induction hypothesis} \\
&\quad\Rightarrow\quad \langle\alpha\rangle\phi \to \theta, \quad \text{by the Switching Lemma.}
\end{aligned}$$

If $\xi' \trianglelefteq \phi$ a similar argument can be used. Of the remaining cases, those involving $\langle\cdot\rangle_i$ and $\circ_i$ are proved in a similar way, while the unary cases with $\updownarrow$, $\Updownarrow$, D_t are similar to the 'old' case for D in 3.3.7. $\dashv$

5.5.24. DEFINITION. A *theory* is either a set theory or a relation theory, where a *set theory* is a set of set formulas, and a *relation theory* is a set of relation formulas. Let Φ and Ω be countable collections of propositional and relational variables, respectively. A theory Δ is a (Φ, Ω)-*theory* if all atomic symbols occurring in Δ are in $\Phi \cup \Omega$. For a logic $\mathbf{L}$, Δ is $\mathbf{L}$-*consistent* if $\Delta \not\vdash \bot$; Δ is a *maximal* $\mathbf{L}$-*consistent* (Φ, Ω)-theory if no (Φ, Ω)-theory properly extends Δ while being consistent.

Δ is a *distinguishing* (Φ, Ω)-theory if for every $\xi \in \Delta$ and every position σ, there is an atomic symbol k such that $\mathtt{F}(\mathtt{N}(O_t k, \sigma, \xi)_\epsilon) \in \Delta$.

5.5.25. LEMMA. (Extension Lemma) *Let Σ be a consistent (Φ, Ω)-theory. Let $\Phi' \supseteq \Phi$, $\Omega' \supseteq \Omega$ be extensions of Φ and Ω by countably many propositional and relational variables, respectively. There is a maximal consistent, distinguishing (Φ', Ω')-theory Σ' extending Σ.*

5.5.26. DEFINITION. (Canonical relations) We define the following *canonical relations* between distinguishing theories:

$$\begin{array}{rcl}
R^c_{D_s}(\Delta_0, \Delta_1) & \iff & \text{for all } \phi_1 \in \Delta_1\text{: } D_s\phi_1 \in \Delta_0 \\
R^c_{D_r}(\Delta_0, \Delta_1) & \iff & \text{for all } \alpha_1 \in \Delta_1\text{: } D_r\alpha_1 \in \Delta_0 \\
R^c_{\otimes}(\Delta_0, \Delta_1) & \iff & \text{for all } \alpha_1 \in \Delta_1\text{: } \otimes\alpha_1 \in \Delta_0 \\
C^c_{\circ}(\Delta_0, \Delta_1, \Delta_2) & \iff & \text{for all } \alpha_1 \in \Delta_1,\ \alpha_2 \in \Delta_2\text{: } \alpha_1 \circ \alpha_2 \in \Delta_0 \\
C^c_{\circ_1}, C^c_{\circ_2} & & \text{similarly} \\
P^c_{\langle\rangle}(\Delta_0, \Delta_1, \Delta_2) & \iff & \text{for all } \alpha_1 \in \Delta_1,\ \phi_2 \in \Delta_2\text{: } \langle\alpha_1\rangle\phi_2 \in \Delta_0
\end{array}$$

$$\begin{array}{rcl}
P^c_{\langle\rangle_1}, P^c_{\langle\rangle_2} & & \text{similarly} \\
F^c_{\updownarrow}(\Delta_0, \Delta_1) & \iff & \text{for all } \phi_1 \in \Delta_1\text{: } \updownarrow\phi_1 \in \Delta_0 \\
F^c_{\Updownarrow}(\Delta_0, \Delta_1) & \iff & \text{for all } \alpha_1 \in \Delta_1\text{: } \Updownarrow\alpha_1 \in \Delta_0 \\
I^c_\delta(\Delta_0) & \iff & \delta \in \Delta_0.
\end{array}$$

5.5.27. LEMMA. (Successor Lemma) *Let Δ be a distinguishing theory. If Δ contains a formula whose main operator is a modal operator $\#$, then the required $R_\#$-successor(s) exist:*

- *if $D_s\phi \in \Delta$, then there is a distinguishing Σ with $\phi \in \Sigma$ and $R^c_{D_s}\Delta\Sigma$; mutatis mutandis the same holds for D_r, $\otimes$, $\updownarrow$ and $\Updownarrow$;*
- *if $\langle\alpha\rangle\phi \in \Delta$, then there are distinguishing theories Σ, Γ such that $\alpha \in \Sigma$, $\phi \in \Gamma$, and $P^c_{\langle\rangle}\Delta\Sigma\Gamma$; mutatis mutandis the same holds for $\langle\rangle_1$, $\langle\rangle_2$, and $\circ$, $\circ_1$, $\circ_2$.*

Proof. The unary cases are the same as before (cf. Lemma 3.3.12); we only do one of the binary cases: $(\alpha \circ \beta) \in \Delta$. Reasoning as in the proof of 3.3.12 we find atomic symbols a and b such that $(\alpha \cap O_r a) \circ (\beta \cap O_r b) \in \Delta$. Put $\Sigma := \{\gamma : ((\gamma \cap O_r a) \circ O_r b) \in \Delta\}$, and $\Gamma := \{\gamma : (O_r a \circ (\gamma \cap O_r b)) \in \Delta\}$. Reasoning as in the unary case, Σ, Γ are maximal, consistent and distinguishing. To see that $C^c_\circ\Delta\Sigma\Gamma$ holds, observe that $((\gamma_1 \cap O_r a) \circ O_r b) \in \Delta$ implies that either $((\gamma_1 \cap O_r a) \circ (O_r b \cap \gamma_2)) \in \Delta$ or $((\gamma_1 \cap O_r a) \circ (O_r b \cap -\gamma_2)) \in \Delta$. So, if $\gamma_1 \in \Sigma$, and $\gamma_2 \in \Gamma$, then we must $(\gamma_1 \cap O_r a) \circ (O_r b \cap \gamma_2) \in \Delta$ (otherwise $-\gamma_2 \in \Gamma$). This implies $\gamma_1 \circ \gamma_2 \in \Delta$. ⊣

5.5.28. DEFINITION. (Provisional canonical model) We define a *provisional canonical model* $\mathfrak{M}^c$ as follows. Fix a set of proposition symbols Φ, and a set of atomic relation symbols Ω.

$$\mathfrak{M}^c = (W^c_s, W^c_r, R^c_{D_s}, R^c_{D_r}, R^c_\otimes, C^c_\circ, C^c_{\circ_1}, C^c_{\circ_2}, P^c_{\langle\rangle}, P^c_{\langle\rangle_1}, P^c_{\langle\rangle_2}, F^c_\updownarrow, F^c_\Updownarrow, I^c_\delta, V^c),$$

where the relations are defined as in Definition 5.5.26, and W^c_s and W^c_r are the sets of all maximal consistent distinguishing set theories and relation theories over Φ, Ω, respectively; V^c is the *canonical valuation* given by $\Delta \in V^c(p)$ iff $p \in \Delta$ (for $\Delta \in W^c_s$), and $\Delta \in V^c(a)$ iff $a \in \Delta$ (for $\Delta \in W^c_r$).

On the provisional canonical model $\mathfrak{M}^c$ we interpret the operators D_s and D_r using the relations $R^c_{D_s}$ and $R^c_{D_r}$: $\Delta \models D_t\xi$ iff for some Σ we have both $R^c_{D_t}\Delta\Sigma$ and $\Sigma \models \xi$ $(t \in \{s, r\})$.

5.5.29. REMARK. The provisional canonical model $\mathfrak{M}^c$ has almost all the required properties to count as a model based on a Peirce frame. Being Sahlqvist formulas, the axioms expressing those properties are canonical. That is: $\mathfrak{M}^c$ satisfies (CR0)–(CR8), (CP0)–(CP13), as well as (CP16)–(CP23). Among other things, this implies $C^c_\circ$, $C^c_{\circ_1}$ and $C^c_{\circ_2}$ form a versatile 'triple,' as well as $P^c_{\langle\rangle}$, $P^c_{\langle\rangle_1}$ and $P^c_{\langle\rangle_2}$; likewise, $F^c_\updownarrow$ and $F^c_\Updownarrow$ form a versatile pair. Hence, as noted in Definition 5.5.17, we can suppress the relations $C^c_{\circ_1}$, $C^c_{\circ_2}$, $P^c_{\langle\rangle_1}$, $P^c_{\langle\rangle_2}$ and $F^c_\Updownarrow$. Further, as $\mathfrak{M}^c \models$ (CR0) we can replace $R^c_\otimes$ by a function f^c.

Moreover, the relations $R^c_{D_s}$ and $R^c_{D_r}$ are irreflexive (by construction), symmetric (by axiom (MD3)), and pseudo-transitive[1] (by axiom (MD2)). Thus, the only possible shortcoming $\mathfrak{M}^c$ has is that $(R^c_{D_s} \cup =)$ and $(R^c_{D_r} \cup =)$ are not the universal relations on W^c_s and W^c_r, respectively. This will be fixed below. But first we state the following.

5.5.30. LEMMA. (Provisional Truth Lemma) *Consider the provisional canonical model $\mathfrak{M}^c$. For all $\Delta \in (W^c_s \cup W^c_r)$ and all formulas ξ in $\mathcal{ML}_2^{\neq \mathrm{sat}}$ we have $\mathfrak{M}^c, \Delta \models \xi$ iff $\xi \in \Delta$.*

5.5.31. DEFINITION. (Final canonical model) Fix an element $\Delta^0_s \in W^c_s$ and an element $\Sigma^0_r \in W^c_r$ such that $F^c_{\updownarrow}\Sigma_r\Delta_s$. Define the *final canonical model* $\mathfrak{M}^f$ as

$$\mathfrak{M}^f = (W^f_s, W^f_r, R^f_{D_s}, R^f_{D_r}, I^f, f^f, C^f, F^f, P^f, V^f),$$

where $\mathfrak{M}^f$ is the submodel of $\mathfrak{M}^c$ *generated* along $R^c_{D_s}$ and $R^c_{D_r}$ by Δ^0_s and Σ^0_r. That is: W^f_s and W^f_r are the smallest sets X and Y containing Δ^0_s and Σ^0, respectively, such that $x_s \in W^f_s$ and $R^c_{D_s}x_sy_s$ implies $y_s \in W^f_s$, and $x_r \in W^f_r$ and $R^c_{D_r}x_ry_r$ implies $y_r \in W^f_r$.

For Q^c one of the canonical relations, Q^f is defined as the restriction of Q^c to the new domains W^f_s and W^f_r. Likewise, the valuation V^f is simply the restriction of V^c to the new domains.[2]

5.5.32. PROPOSITION. *For all canonical relations Q^c, $\mathfrak{M}^f$ is closed under Q^c:*

1. *if $\Delta_r \in W^f_r$ then $f(\Delta_r) \in W^f_r$;*
2. *if $\Delta_s \in W^f_s$, and $P^c_{\langle\rangle}\Delta_s\Sigma_r\Gamma_s$, then $\Sigma_r \in W^f_r$, $\Gamma_s \in W^f_s$;*
3. *if $\Delta_r \in W^f_r$, and $C^c_{\circ}\Delta_r\Sigma_r\Gamma_r$, then $\Sigma_r \in W^f_r$, $\Gamma_r \in W^f_r$;*
4. *if $\Delta_r \in W^f_r$, and $F^c\Delta_r\Sigma_s$, then $\Sigma_s \in W^f_s$;*
5. *if $\Sigma_s \in W^f_s$, and $F^c\Delta_r\Sigma_s$, then $\Delta_r \in W^f_r$.*

Proof. We show that every Δ related via one of the canonical relations Q^c to an element in $W^f_s \cup W^f_r$, is connected to one of Δ^0_s and Σ^0_r via $(R^c_{D_s} \cup =)$ or $(R^c_{D_r} \cup =)$, respectively. Here is why we introduced the (INC)-principles.

1. Assume $(\Sigma^0_r, \Delta_r) \in (R^c_{D_s} \cup =)$. By (INC2) $\Delta_r = f(\Delta_r)$ or $R_{D_r}\Delta_r f(\Delta_r)$. In both cases the pseudo-transitivity of $R^c_{D_r}$ gives $(\Sigma^0_r, f(\Delta_r)) \in (R^c_{D_r} \cup =)$ and $f(\Delta_r) \in W^f_r$.

2. Assume $(\Delta^0_s, \Delta_s) \in (R^c_{D_s} \cup =)$. By (INC1) $P^c\Delta_s\Sigma_r\Gamma_s$ implies $\Delta_s = \Gamma_s$ or $R^c_{D_s}\Delta_s\Gamma_s$. In both cases $(\Delta^0_s, \Gamma_s) \in (R^c_{D_s} \cup =)$ and $\Gamma_s \in W^f_s$. Further, by construction $F^c\Sigma^0_r\Delta^0_r$; together with the above and Lemma 5.5.21 this gives $\Sigma^0_r = \Sigma_r$ or $R^c_{D_r}\Sigma^0_r\Sigma_r$. In both cases $(\Sigma^0_r, \Sigma_r) \in (R^c_{D_r} \cup =)$ and $\Sigma_r \in W^f_r$.

---

[1]That is: they satisfy $\forall xyz\,(Rxy \wedge Ryz \rightarrow x = z \vee Rxz)$.

[2]That is: $R^f_{D_s} = R^c_{D_s} \cap {}^2W^f_s$, $R^f_{D_r} = R^c_{D_r} \cap {}^2W^f_r$, $f^f = R^c_{\otimes} \restriction W^f_r$, $C^f = C^c_{\circ} \cap {}^3W^f_r$, $F^f = F^c_{\updownarrow} \cap (W^f_r \times W^f_s)$, $P^f = P^c_{\langle\rangle} \cap (W^f_s \times W^f_r \times W^f_s)$, $I^f = I^c_{\delta} \cap W^f_r$, $V^f(p) = V^c(p) \cap W^f_s$, $V^f(a) = V^c(a) \cap W^f_r$.

3. Assume $(\Sigma_r^0, \Delta_r) \in (R_{D_r}^c \cup =)$. By (INC3) $C^c\Delta_r\Sigma_r\Gamma_r$ implies ($\Delta_r = \Sigma_r$ or $R_{D_r}^c\Delta_r\Sigma_r$) and ($\Delta_r = \Gamma_r$ or $R_{D_r}^c\Delta_r\Gamma_r$). In all cases this gives (Σ_r^0, Δ_r), $(\Sigma_r^0, \Gamma_r) \in (R_{D_r}^c \cup =)$ and Σ_r, $\Gamma_r \in W_r^f$.

4. Assume $(\Sigma_r^0, \Delta_r) \in (R_{D_r}^c \cup =)$. By construction $F^c\Sigma_r^0\Delta_s^0$. Hence by Lemma 5.5.21 $F^c\Delta_r\Sigma_s$ yields $\Delta_s^0 = \Sigma_s$ or $R_{D_s}^c\Delta_s^0\Sigma_s$. In both cases $(\Delta_s^0, \Sigma_s) \in (R_{D_r}^c \cup =)$ and $\Sigma_s \in W_s^f$.

5. Assume $(\Delta_s^0, \Sigma_s) \in (R_{D_s}^c \cup =)$. Together with $F^c\Sigma_r^0\Delta_s^0$ and $F^c\Delta_r\Sigma_s$ this gives $\Sigma_r^0 = \Delta_r$ or $R_{D_r}^c\Sigma_r^0$, by Lemma 5.5.21. Hence $(\Sigma_r^0, \Delta_r) \in (R_{D_r}^c \cup =)$ and $\Delta_r \in W_r^f$. ⊣

5.5.33. LEMMA. *In $\mathfrak{M}^f$ the relation $R_{D_s}^f$ is real inequality on W_s^f, and $R_{D_r}^f$ is real inequality on W_r^f.*

Proof. By earlier observations $R_{D_s}^f$ and $R_{D_r}^f$ are both irreflexive, so it suffices to show that $R_{D_s}^f$ and $R_{D_r}^f$ hold between any two different elements of their respective domains. We only prove the latter for $R_{D_s}^f$.

Let Δ_s, $\Sigma_s \in W_s^f$ with $\Delta_s \neq \Sigma_s$. By the proof of Proposition 5.5.32 (Δ_s^0, Δ_s), $(\Delta_s^0, \Sigma_s) \in (R_{D_s}^f \cup =)$, for Δ_s^0 the generating set point of $\mathfrak{M}^f$. By the symmetry and pseudo-transitivity properties of $R_{D_s}^f$ it follows that $R_{D_s}^f\Delta_s\Sigma_s$. ⊣

5.5.34. LEMMA. (Structure Lemma) *The final canonical model $\mathfrak{M}^f$ satisfies of* (CR0)–(CR8), (CP0)–(CP23). *Moreover, the difference operators D_s and D_r receive their standard interpretations in $\mathfrak{M}^f$. Hence, by Theorem 5.5.16 the frame underlying $\mathfrak{M}^f$ is (isomorphic to) a two-sorted Peirce frame.*

5.5.35. LEMMA. (Truth Lemma) *Consider the final canonical model $\mathfrak{M}^f$. For all $\Delta \in (W_s^f \cup W_r^f)$ and all formulas ξ in $\mathcal{ML}_2^{\neq\mathrm{sat}}$ we have $\mathfrak{M}^f, \Delta \models \xi$ iff $\xi \in \Delta$.*

5.5.36. THEOREM. (Completeness Theorem) *Let $\Delta \cup \{\xi\}$ be a set of $\mathcal{ML}_2^{\neq\mathrm{sat}}$-formulas. Then $\Delta \vdash \xi$ in* **2-MLPE** *iff $\Delta \models_{\mathsf{TPF}} \xi$.*

Proof. By Definitions 5.3.8 and 5.5.5 we may assume that $\Delta \cup \{\xi\}$ consists only of set formulas, or only of relation formulas.

Proving soundness is left to the reader. To prove completeness, assume Δ and the negation of ξ are consistent in **2-MLPE**. Construct a final canonical model $\mathfrak{M}^f$ as in Definition 5.5.31 using a maximal consistent distinguishing extension Σ of Δ and the negation of ξ as a starting point. The frame underlying $\mathfrak{M}^f$ is in TPF, by Lemma 5.5.34. By the Truth Lemma we have $\mathfrak{M}^f, \Sigma \models$ 'Δ plus the negation of ξ.' Hence $\Delta \not\models_{\mathsf{TPF}} \xi$. ⊣

STEP 4: BACK TO THE OLD LANGUAGE

We now port Theorem 5.5.36 to our original language $\mathcal{ML}_2$ via a suitable translation. It will turn out that we can get rid of all the extras accumulated in Step 3 at the cost of two special derivation rules.

5.5.37. DEFINITION. We define one more axiom system: **2-MLP**. Its language is $\mathcal{ML}_2$. Its axioms are those of **2-MLPL** (Definition 5.5.4), and its rules of inference are those of **2-MLPL** plus the following two irreflexivity rules:

(IR$_s$) $p \wedge \neg\langle -\delta\rangle p \to \phi \,/\, \phi$, where p does not occur in ϕ
(IR$_r$) $a \cap -((-\delta \circ a \circ \mathbf{1}) \cup (\mathbf{1} \circ a \circ -\delta)) \to \alpha \,/\, \alpha$, where a does not occur in α.

The rules (IR$_s$) and (IR$_r$) will be used as substitutes for (IR$_{D_s}$) and (IR$_{D_r}$).

5.5.38. DEFINITION. Define a mapping $(\cdot)^\dagger$ from $\mathcal{ML}_2^{\neq\mathrm{sat}}$-formulas to $\mathcal{ML}_2$-formulas as follows

$$
\begin{array}{rclrcl}
p^\dagger & = & p & a^\dagger & = & a \\
\bot^\dagger & = & \bot & \mathbf{0}^\dagger & = & \mathbf{0} \\
\top^\dagger & = & \top & \mathbf{1}^\dagger & = & \mathbf{1} \\
(\neg\phi)^\dagger & = & \neg(\phi^\dagger) & \delta^\dagger & = & \delta \\
(\phi \wedge \psi)^\dagger & = & \phi^\dagger \wedge \psi^\dagger & (-\alpha)^\dagger & = & -(\alpha^\dagger) \\
(\Updownarrow\alpha)^\dagger & = & \langle\alpha^\dagger\rangle\top & (\alpha \cap \beta)^\dagger & = & \alpha^\dagger \cap \beta^\dagger \\
(\langle\alpha\rangle\phi)^\dagger & = & \langle\alpha^\dagger\rangle\phi^\dagger & (\otimes\alpha)^\dagger & = & \otimes(\alpha^\dagger) \\
(\langle\phi\rangle_1\psi)^\dagger & = & \otimes(\updownarrow(\phi^\dagger)) \cap \updownarrow(\psi^\dagger) & (\updownarrow\phi)^\dagger & = & \updownarrow(\phi^\dagger) \\
(\langle\phi\rangle_2\alpha)^\dagger & = & \langle\otimes\alpha^\dagger\rangle\phi^\dagger & (\alpha \circ \beta)^\dagger & = & \alpha^\dagger \circ \beta^\dagger \\
(D_s\phi)^\dagger & = & \langle-\delta\rangle\phi^\dagger & (\alpha \circ_1 \beta)^\dagger & = & \beta^\dagger \circ \otimes\alpha^\dagger \\
 & & & (\alpha \circ_2 \beta)^\dagger & = & \otimes\beta^\dagger \circ \alpha^\dagger
\end{array}
$$
$$(D_r\alpha)^\dagger = (-\delta \circ \alpha^\dagger \circ \mathbf{1}) \cup (\mathbf{1} \circ \alpha^\dagger \circ -\delta).$$

A short comment to motivate the above translation $(\cdot)^\dagger$. First, $y_r \models \langle p\rangle_1 q$ means that there exist x_s, z_s with $Pz_sy_rx_s$ and $x_s \models p$, $z_s \models q$; hence, $y_r \models (\otimes\updownarrow p) \cap (\updownarrow q)$ $(= (\langle p\rangle_1 q)^\dagger)$. Second, $z_s \models \langle p\rangle_2 a$ iff there exist x_s, y_r with $Px_sy_rz_s$ and $x_s \models p$, $y_r \models a$. Hence $z_s \models \langle\otimes a\rangle p$ $(= (\langle p\rangle_2 a)^\dagger)$. Similar remarks motivate the clauses for $\circ$. As for the difference operators, in full Peirce models we have $x \models D_s p$ iff for some $y \neq x$ $y \models p$ iff $x \models \langle-\delta\rangle p$. Also, $(x,y) \models D_r a$ iff for some $(u,v) \neq (x,y)$, $(u,v) \models a$; if $u \neq x$, then $(x,y) \models (-\delta \circ a \circ \mathbf{1})$, and if $v \neq y$, then $(x,y) \models (\mathbf{1} \circ a \circ -\delta)$; in both cases $(x,y) \models (D_r a)^\dagger$.

5.5.39. LEMMA. *Let ξ be an axiom of* **2-MLPE**. *Then* $\vdash \xi^\dagger$ *in* **2-MLP**.

Proof. We first use the completeness of **2-MLPL** established in Theorem 5.5.8 to argue *semantically* that **2-MLPL** proves the $\dagger$-translations of all **2-MLPE** axioms. The claim is obvious for axioms (MR0)–(MR8) and (MP1)–(MP9). It remains to show that $\vdash \xi^\dagger$ in **2-MLPL**, for ξ one of (MP14)–(MP23), as well as the axioms for the D-operators and the inclusion axioms (INC1)–(INC5). We begin with the axioms for the D-operators.

(MD$_s$1) $(\overline{D}_s(p \to q) \to (\overline{D}_s p \to \overline{D}_s q))^\dagger = \neg\langle-\delta\rangle\neg(p \to q) \to (\neg\langle-\delta\rangle\neg p \to \neg\langle-\delta\rangle\neg q)$. Assume $x_s \models \neg\langle-\delta\rangle\neg(p \to q)$, $\neg\langle-\delta\rangle\neg p$. Let y_r, z_s be such that $Px_sy_rz_s$ and $y_r \models -\delta$. Then $z_s \models q$, hence $x_s \models \neg\langle-\delta\rangle\neg q$, as required.

(MD$_s$2) $(D_sD_sp \to p \vee D_sp)^\dagger = \langle-\delta\rangle\langle-\delta\rangle p \to p \vee \langle-\delta\rangle p$. Assume that $x_s \models \langle-\delta\rangle\langle-\delta\rangle p$. Then, for some y_r, y_r', z_s, z_s' we have $Px_sy_rz_s$, $Pz_sy_r'z_s'$, and

y_r, $y'_r \models -\delta$ and $z'_s \models p$. By (CP1) there is a y''_r such that $Px_s y''_r z'_s$. If $y''_r \models -\delta$ then $x_s\langle -\delta\rangle p$. Otherwise, $y'' \models \delta$ implies $x_s = z'_s$ by (CP3), hence $x_s \models p$.

(MD$_s$3) $(p \to \neg D_s \neg D_s p)^\dagger = p \to \neg\langle -\delta\rangle\neg\langle -\delta\rangle p$. Assume that $x_s \models p$, $\langle -\delta\rangle\neg\langle -\delta\rangle p$. We will derive a contradiction. As $x_s \models \langle -\delta\rangle\neg\langle -\delta\rangle p$ there are y_r, z_s with $Px_s y_r z_s$, $y_r \notin I$ and $z_s \models \neg\langle -\delta\rangle p$. By (CP5) this implies $Pz_s f(y_r)x_s$. Now, if $f(y_r) \notin I$ then $z_s \models \langle -\delta\rangle p$, and we have arrived at the desired contradiction. So it suffice to show $f(y_r) \notin I$. Assume $f(y_r) \in I$, then

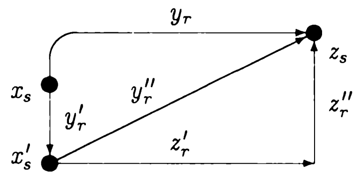
$$\begin{aligned} Px_s y_r z_s \wedge Pz_s f(y_r) x_s &\Rightarrow Cy''_r y_r f(y_r), \quad \text{for some } y''_r, \text{ by (CP1)} \\ &\Rightarrow Cy_r y_r f(y_r), \quad \text{by (CR5) and } f(y_r) \in I \\ &\Rightarrow Cf(y_r) y_r f(y_r), \quad \text{by (CR1) and (CR7)} \\ &\Rightarrow f(y_r) = y_r, \quad \text{by (CR5)} \\ &\Rightarrow y_r \in I, \quad \text{a contradiction.} \end{aligned}$$

(MD$_r$1)-(MD$_r$3) Their translations may be proved from the relational axioms only, as shown by Venema (1991, Proposition 3.3.38).

(INC1) $(\langle a\rangle p \to E_s p)^\dagger = \langle a\rangle p \to p \vee \langle -\delta\rangle p$. Assume $x_s \models \langle a\rangle p$. So there are y_r, z_s with $Px_s y_r z_s$, $y_r \models a$, $z_s \models p$. If $y_r \in I$, we have $x_s = z_s$ by (CP3), and hence $x_s \models p \vee \langle -\delta\rangle p$. If $y_r \notin I$, we must have $x_s \models \langle -\delta\rangle p$, and $x_s \models p \vee \langle -\delta\rangle p$.

(INC2)-(INC3) Their translations may be proved from the relational axioms only, as shown by Venema (1991, Proposition 3.3.38).

(INC4) $(\langle D_r(\updownarrow q)\rangle\top \to E_s q)^\dagger = \langle(-\delta \circ \updownarrow q \circ \mathbf{1}) \cup (\mathbf{1} \circ \updownarrow q \circ -\delta)\rangle\top \to q \vee \langle -\delta\rangle q$. Assume x_s satisfies the antecedent of this translation, say $x_s \models \langle -\delta \circ \updownarrow q \circ \mathbf{1}\rangle\top$. Then there are y_r, z_s with $Px_s y_r z_s$ and $y_r \models -\delta \circ \updownarrow q \circ \mathbf{1}$. This means that there are y'_r, y''_r, z'_r, z''_r such that $Cy_r y'_r y''_r$, $Cy''_r z'_r z''_r$ and $y'_r \models -\delta$, and $z'_r \models \updownarrow q$. The latter implies that there is an x'_s with $Fz'_r x'_s$ and $x'_s \models q$.



It suffices to show that $Px_s y'_r x'_s$, for then $x_s \models \langle -\delta\rangle q$. Now, to see that $Px_s y'_r x'_s$, observe that

$$Px_s y_r z_s \wedge Cy_r y'_r y''_r \quad \Rightarrow \quad Px_s y'_r z'_s \wedge Pz'_s y''_r z_s, \tag{5.3}$$

for some z'_s by (CP2). Furthermore, $Pz'_s y''_r z_s$ and $Cy''_r z'_r z''_r$ imply that for some z''_s, $Pz'_s z'_r z''_s$, by (CP2). Next, $Pz'_s z'_r z''_s$ implies $Fz'_r z'_s$ by Lemma 5.5.10. On the other hand, we already have that $Fz'_r x'_s$, so by (CP10) it follows that $z'_s = x'_s$. But, then, by (5.3) we must have $Px_s y'_r x'_s$, as required.

The case that $x_s \models \langle \mathbf{1} \circ \updownarrow q \circ -\delta\rangle\top$ is proved entirely analogously.

(INC5) $(\updownarrow(D_s\langle a\rangle\top) \to E_r a)^\dagger = \updownarrow\langle -\delta\rangle\langle a\rangle\top \to a \cup (-\delta \circ a \circ \mathbf{1}) \cup (\mathbf{1} \circ a \circ -\delta)$.

Assume $y_r \models \updownarrow\langle -\delta\rangle\langle a\rangle$. Let x_s be such that Fy_rx_s and $x_s \models \langle -\delta\rangle\langle a\rangle\top$. Then there are y_r', z_s', y_r'', z_s'' with $Px_sy_r'z_s'$, $Pz_s'y_r''z_s''$ and $y_r' \models -\delta$, $y_r'' \models a$. Now observe

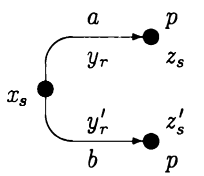
$$\begin{aligned} Px_sy_r'z_s' \wedge Pz_s'y_r''z_s'' &\Rightarrow Px_sz_r'z_s'' \wedge Cz_r'y_r'y_r'', \text{ for some } z_r', \text{ by (CP1)} \\ &\Rightarrow Cy_rz_r'z_r'', \text{ for some } z_r'', \text{ by } Fy_rx_s \text{ and (CP8)} \\ &\Rightarrow y_r \models (-\delta \circ a) \circ \mathbf{1}. \end{aligned}$$

(MP14) $(E_sp \to \langle\mathbf{1}\rangle p)^\dagger = (p \vee \langle -\delta\rangle p) \to \langle\mathbf{1}\rangle p$. Assume $x_s \models p \vee \langle -\delta\rangle p$. Then $x_s \models \langle\delta\rangle p \vee \langle -\delta\rangle p$ by (MP4), so $x_s \models \langle\delta \cup -\delta\rangle p$. This implies $x_s \models \langle\mathbf{1}\rangle p$.

(MP15) $(E_sO_sp \wedge \langle a\rangle p \wedge \langle b\rangle p \to \langle a \cap b\rangle p)^\dagger =$

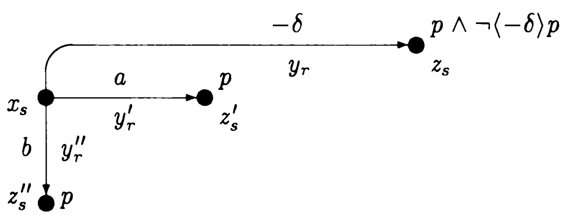
$$((p \wedge \neg\langle -\delta\rangle p) \vee \langle -\delta\rangle(p \wedge \neg\langle -\delta\rangle p)) \wedge \langle a\rangle p \wedge \langle b\rangle p \to \langle a \cap b\rangle p.$$

Assume first that $x_s \models p \wedge \neg\langle -\delta\rangle p \wedge \langle a\rangle p \wedge \langle b\rangle p$. Then, for some y_r, y_r', z_s, z_s' we have $Px_sy_rz_s$, $Px_sy_r'z_s'$ and $y_r \models a$, $y_r' \models b$, and z_s, $z_s' \models p$.



Observe that $x_s \models \neg\langle -\delta\rangle p$ implies y_r, $y_r' \in I$. Hence, by (CP3), $x_s = z_s = z_s'$. Furthermore, $Px_sy_rz_s$ implies $Pz_sf(y_r)x_s$; together with $Px_sy_r'z_s'$ and (CP1) this yields a y_r'' such that $Cy_r''f(y_r)y_r'$, and so by (CR8) such that $Cy_r'y_ry_r''$. By (CR5) we find $y_r' = y_r''$ and $y_r'' = f(y_r)$. So we have $Cf(y_r)f(y_r)f(y_r)$, and by (CR8) $Cf(y_r)y_rf(y_r)$. As $y_r \in I$ implies $f(y_r) \in I$, (CR5) now gives $y_r = f(y_r)$. All in all we find $y_r = f(y_r) = y_r'' = y_r'$. Hence $x_s \models \langle a \cap b\rangle p$.

Assume next that $x_s \models \langle -\delta\rangle(p \wedge \neg\langle -\delta\rangle p)) \wedge \langle a\rangle p \wedge \langle b\rangle p$. Then there are y_r, y_r', y_r'' and z_s, z_s', z_s'' with $Px_sy_rz_s$, $Px_sy_r'z_s'$, $Px_sy_r''z_s''$, and $y_r \notin I$, $y_r' \models a$, $y_r'' \models b$, and z_s, z_s', $z_s'' \models p$, and $z_s \not\models \langle -\delta\rangle p$.



By (CP1) there is a y_r''' with $Pz_sy_r'''z_s'$ and $Cy_r'''f(y_r)y_r'$. If $y_r''' \notin I$, then $z_s' \not\models p$ — a contradiction. Hence $y_r''' \in I$. Likewise we find a $y_r'''' \in I$ with $Cy_r''''f(y_r)y_r''$. By (CR8) and (CR5) we have $y_r' = y_r$ and $y_r'' = y_r$. Hence $y_r \models a \cap b$, and $x_s \models \langle a \cap b\rangle p$.

(MP16) $(a \to \bar{\updownarrow}\Updownarrow a)^\dagger = a \to -\updownarrow\neg\langle a\rangle\top$. Assume $y_r \models a$, $\updownarrow\neg\langle a\rangle\top$. We derive a contradiction. For some x_s we have Fy_rx_s and $x_s \models \neg\langle a\rangle\top$. The latter implies that for no y'_r, z'_s we have $Px_sy'_rz'_s$ and $y'_r \models a$. However, by (CP13) and Lemma 5.5.10, there is a z_s with $Px_sy_rz_s$, yielding the desired contradiction.

(MP17) $(p \to \bar{\Updownarrow}\updownarrow p)^\dagger = p \to \neg\langle -\updownarrow p\rangle\top$. To arrive at a contradiction, assume that $x_s \models p$, $\langle -\updownarrow p\rangle\top$. Then $Px_sy_rz_s$, $y_r \models -\updownarrow p$, for some y_r, z_s. However, $Px_sy_rz_s$ implies that Fy_rx_s by Lemma 5.5.10. So $y_r \models \updownarrow p$ — a contradiction.

(MP18) $(p \wedge \langle -\langle q\rangle_1 p\rangle q \to \bot)^\dagger = p \wedge \langle -\otimes\updownarrow q \cup -\updownarrow p\rangle q \to \bot$. Assume $x_s \models p$, $\langle -\otimes\updownarrow q \cup -\updownarrow p\rangle q$. Then, there exist y_r, z_s with $Px_sy_rz_s$, $y_r \models -\otimes\updownarrow q \cup -\updownarrow p$, $z_s \models q$. By Lemma 5.5.10 we have Fy_rx_s, $Ff(y_r)z_s$. Hence $y_r \models \otimes\updownarrow q \cap \updownarrow p$ — a contradiction.

(MP19) $(a \cap \langle \neg\langle p\rangle_2 a\rangle_1 p \to \mathbf{0})^\dagger = a \cap \otimes\updownarrow\neg\langle\otimes a\rangle p \cap \updownarrow p \to \mathbf{0}$. Assume $x_r \models a$, $\otimes\updownarrow\neg\langle\otimes a\rangle p$, $\updownarrow p$. Then there exist y_s, z_s with $Ff(x_r)y_s$, Fx_rz_s and $y_s \models \neg\langle\otimes a\rangle p$, $z_s \models p$. By Lemma 5.5.10 it follows that $Pz_sx_ry_s$. By (CP5) this implies $Py_sf(x_r)z_s$, and hence $y_s \models \langle\otimes a\rangle p$, as $x_r \models a$ — a contradiction.

(MP20) $(p \wedge \langle\neg\langle a\rangle p\rangle_2 a \to \bot)^\dagger = p \wedge \langle\otimes a\rangle\neg\langle a\rangle p \to \bot$. Assume $x_s \models p$, $\langle\otimes a\rangle\neg\langle a\rangle p$. Then, for some y_r, z_s, we have $Px_sy_rz_s$, $f(y_r) \models a$, $z_s \models \neg\langle a\rangle p$. Now, $Px_sy_rz_s$ implies $Pz_sf(y_r)x_s$. So, $z_s \models \langle a\rangle p$ — another contradiction.

(MP21)-(MP23) Venema (1991, Proposition 3.3.38) deals with these cases. ⊣

5.5.40. LEMMA. *Let $\xi_1, \ldots, \xi_n \,/\, \xi'$ be a derivation rule of* **2-MLPE**. *Then the rule $\xi^\dagger, \ldots, \xi_n^\dagger \,/\, \xi'^\dagger$ is a derived rule of* **2-MLP**.

Proof. The Lemma is clear for (MP), (SUB), and (NEC) for $\otimes$, $\langle\cdot\rangle\cdot$, $\circ$ and $\updownarrow$. The (NEC) rules for $\langle\cdot\rangle_1\cdot$, $\langle\cdot\rangle_2\cdot$, $\circ_1$, $\circ_2$, $\Updownarrow$, D_s and D_r are dealt with as follows.

($\mathrm{NEC}_{\langle\rangle_1}$) Consider $\psi \,/\, [\![\phi]\!]_1\psi$, or $\psi \,/\, -\langle\neg\phi\rangle_1\neg\psi$. This translates into $\psi^\dagger \,/\, -(\otimes\updownarrow\neg\phi^\dagger \cap \updownarrow\neg\psi^\dagger)$, or $\psi^\dagger \,/\, -\otimes\updownarrow\neg\phi^\dagger \cup -\updownarrow\neg\psi^\dagger$, which is a consequence of ($\mathrm{NEC}_{\updownarrow}$) and $\alpha \,/\, \alpha\cup\beta$. Next consider $\phi \,/\, \lceil ceil\phi]\!]_1\psi$. It suffices to show that $\phi^\dagger \,/\, -\otimes\updownarrow\neg\phi$ is a derived rule in **2-MLP**. This is immediate from ($\mathrm{NEC}_{\updownarrow}$), ($\mathrm{NEC}_{\otimes}$) and (MR0).

($\mathrm{NEC}_{\langle\rangle_2}$) Consider $\alpha \,/\, [\![\phi]\!]_2\alpha$, or $\alpha \,/\, \neg\langle\neg\phi\rangle_2{-\alpha}$. This translates into the rule $\alpha^\dagger \,/\, \neg\langle\otimes-\alpha^\dagger\rangle\neg\phi^\dagger$, which is a derived rule of **2-MLP** by ($\mathrm{NEC}_{\otimes}$) and ($\mathrm{NEC}_{\langle\rangle}$). Next consider $\phi \,/\, [\![\phi]\!]_2\alpha$; it suffices to show that $\phi^\dagger \,/\, \neg\langle\otimes-\alpha^\dagger\rangle\neg\phi^\dagger$ is a derived rule of **2-MLP**. But this is is immediate from ($\mathrm{NEC}_{\langle\rangle}$).

($\mathrm{NEC}_{\circ_1}$) Consider $\alpha \,/\, \alpha \;\bar{\circ}_1\; \beta$, that is: $\alpha \,/\, -(-\alpha \circ_1 -\beta)$. Its translation is $\alpha^\dagger \,/\, -(-\beta^\dagger \circ \otimes-\alpha^\dagger)$. Now, by ($\mathrm{NEC}_{\otimes}$) we have (in **2-MLP**) $\alpha^\dagger \,/\, -\otimes-\alpha^\dagger$, hence $\alpha^\dagger \,/\, -(-\beta^\dagger \circ --\otimes-\alpha^\dagger)$ by ($\mathrm{NEC}_{\circ}$). So $\alpha^\dagger \,/\, -(-\beta^\dagger \circ \otimes-\alpha^\dagger)$. Next consider $\beta \,/\, \alpha \;\bar{\circ}_1\; \beta$. This translates into $\beta^\dagger \,/\, -(-\beta^\dagger \circ \otimes-\alpha^\dagger)$, which is a special case of ($\mathrm{NEC}_{\circ}$).

($\mathrm{NEC}_{\circ_2}$) This case is similar to ($\mathrm{NEC}_{\circ_1}$).

($\mathrm{NEC}_{\Updownarrow}$) Consider $\alpha \,/\, \bar{\Updownarrow}\alpha$, or $\alpha \,/\, \neg\Updownarrow-\alpha$. Its translation reads $\alpha^\dagger \,/\, \neg\langle-\alpha^\dagger\rangle\top$, or $\alpha^\dagger \,/\, [\![\alpha^\dagger]\!]\bot$ — which is an instance of ($\mathrm{NEC}_{\langle\rangle}$).

(NEC_{D_s}) Consider $\phi \,/\, \overline{D}_s\phi$, or $\phi \,/\, \neg D_s\neg\phi$. Its translation is $\phi^\dagger \,/\, \neg\langle -\delta\rangle\neg\phi^\dagger$, which is an instance of ($\mathrm{NEC}_{\langle\rangle}$).

(NEC_{D_r}) Consider $\alpha \,/\, \overline{D}_r\alpha$, which translates into $\alpha^\dagger \,/\, -((-\delta \circ -\alpha^\dagger \circ \mathbf{1}) \cap (\mathbf{1} \circ -\alpha^\dagger \circ -\delta))$. By 2 applications of ($\mathrm{NEC}_\circ$) this is a derived rule of **2-MLP**.

Finally, the irreflexivity rules (IR_{D_s}) and (IR_{D_r}) are easy. The first translates into $p \wedge \neg\langle -\delta\rangle p \rightarrow \phi^\dagger \,/\, \phi^\dagger$, provided p does not occur in $\phi^\dagger$; but this is an instance of (IR_s). And similarly, (IR_{D_r}) translates into (IR_r). ⊣

5.5.41. LEMMA. *Let ξ be a $\mathcal{ML}_2^{\neq\mathrm{sat}}$-formula. Then $\vdash_{\mathbf{2\text{-}MLPE}} \xi$ iff $\vdash_{\mathbf{2\text{-}MPL}} \xi^\dagger$.*

Proof. We show by induction on the length of proofs in **2-MLPE**, that if ξ is derivable in **2-MLPE**, then $\xi^\dagger$ is derivable in **2-MLP**. If we have a proof of length 1, then ξ must be an axiom, and in that case $\vdash_{\mathbf{2\text{-}MLP}} \xi^\dagger$ by Lemma 5.5.39 If ξ has been derived by means of derivation rules from earlier theorems $\xi_1, \ldots, \xi_n$, then by the induction hypothesis, $\xi_1^\dagger, \ldots, \xi_n^\dagger$ are derivable in **2-MLP**, and by the translation of the derivation rule used and Lemma 5.5.40 $\xi^\dagger$ can be derived from $\xi_1^\dagger, \ldots, \xi_n^\dagger$ in **2-MLP**.

Finally, we have to show that if $\vdash \xi^\dagger$ in **2-MLP**, then $\vdash \xi$ in **2-MLPE**. This is easy: all **2-MLP** axioms are derivable in **2-MLPE**, and the irreflexivity rules of **2-MLP** are derived rules in **2-MLPE**. (Note that **2-MLPE** $\vdash \xi \leftrightarrow \xi^\dagger$.) ⊣

5.5.42. COROLLARY. *If $\Delta \vdash \xi$ in* **2-MLPE**, *then $\{\,\theta^\dagger : \theta \in \Delta\,\} \vdash \xi^\dagger$ in* **2-MLP**.

5.5.43. THEOREM. (Completeness Theorem) *Let $\Delta \cup \{\,\xi\,\}$ be a set of $\mathcal{ML}_2$-formulas. Then $\Delta \vdash \xi$ in* **2-MLP** *iff $\Delta \models_{\mathsf{TPF}} \xi$.*

Proof. Proving soundness is left to the reader. For completeness, assume that $\Delta \models_{\mathsf{TPF}} \xi$. By Theorem 5.5.36 this implies $\Delta \vdash \xi$ in **2-MLPE**. Hence, by Corollary 5.5.42 we have $\Delta^\dagger \vdash \xi^\dagger$ in **2-MLP**. But, as $\Delta \cup \{\,\xi\,\}$ is a set of $\mathcal{ML}_2$-formulas, we find that $\Delta \vdash \xi$ in **2-MLP**. ⊣

Recall that $\mathbf{DML}(\Phi, \Omega)$ is the modal logic defined in §4.6 and §5.4.

5.5.44. COROLLARY. *Let ϕ be a set formula. Then $\vdash \phi$ in* **2-MLP** *iff $\vdash \phi$ in* $\mathbf{DML}(\Phi, \Omega)$.

Proof. This is immediate from the completeness results for $\mathbf{DML}(\Phi, \Omega)$ (cf. Theorem 4.6.7) and **2-MLP**. ⊣

5.5.45. DEFINITION. Recall that the (set) term $d_s(x)$ was defined in Definition 5.4.1 as $d_s(x) = (-1') : x$. For x a relation variable, define the term $d_r(x)$ by putting $d_r(x) = (-1'; x; 1) + (1; x; -1')$.

5.5.46. DEFINITION. Let $\mathbf{L}_2$ to be the smallest set of equations containing the $\star$-translations of the **2-MLP**-axioms which is closed under the ordinary algebraic deduction rules (cf. Definition 5.4.2), the closure operation (5.2) and the following closure operation:

$$x_r \cdot -d_r(x_r) \leq t(y_0, \ldots, y_n) \;/\; t(y_0, \ldots, y_n) = 1,$$
provided x_r does not occur among $y_0, \ldots, y_n$.

5.5.47. THEOREM. (Algebraic Completeness) $\mathbf{L}_2$ *is a complete axiomatization of all equations valid in* FPA.

Proof. As $\mathbf{L}_2$ is the algebraic counterpart of **2-MLP** this is immediate from the completeness of **2-MLP** and Remark 5.3.11. ⊣

5.5.48. REMARK. An obvious question here is: are the rules (IR_s) and (IR_r) really necessary to arrive at a completeness result for FPA? Given the non-finite axiomatizability results in relation algebra, the natural conjecture would be that at least one of them is necessary. And if at least one is necessary, can the other one be derived from it? Settling these issues is left for further study.

5.6 EXPRESSIVE POWER

We take up the issue of expressive power again, by describing the first-order counterpart of $\mathcal{ML}_2$, and defining appropriate bisimulations for $\mathcal{ML}_2$.

THE CONNECTION WITH FIRST-ORDER LOGIC

When interpreted on Peirce models the language $\mathcal{ML}_2$ corresponds to a three-variable fragment in which up to two variables can occur free. This may be established using the techniques and results of §4.4 on the expressive power of $\mathcal{DML}$. Recall that given τ a (first-order) vocabulary, $\mathcal{L}_3(x_1, x_2)$ denotes the set of all first-order formulas over τ involving at most 3 individual variables and having at most 2 free variables.

$$
\begin{array}{rcl}
ST(\top) & = & (x_1 = x_1) \\
ST(p) & = & P(x_1) \\
ST(\neg\phi) & = & \neg ST(\phi) \\
ST(\phi \wedge \psi) & = & ST(\phi) \wedge ST(\psi) \\
ST(D_s\phi) & = & \exists x_2\,(x_1 \neq x_2 \wedge [x_2/x_1]ST(\phi)) \\
ST(\langle\alpha\rangle\phi) & = & \exists x_2\,(ST(\alpha) \wedge [x_2/x_1]ST(\phi)) \\
ST(\mathbf{1}) & = & (x_1 = x_1) \wedge (x_2 = x_2) \\
ST(\delta) & = & x_1 = x_2 \\
ST(a) & = & A(x_1, x_2) \\
ST(-\alpha) & = & \neg ST(\alpha) \\
ST(\alpha \cap \beta) & = & ST(\alpha) \wedge ST(\beta) \\
ST(\otimes\alpha) & = & [x_2/x_1, x_1/x_2]ST(\alpha) \\
ST(\alpha \circ \beta) & = & \exists x_3\,([x_3/x_2]ST(\alpha) \wedge [x_3/x_1]ST(\beta)) \\
ST(\updownarrow\phi) & = & ST(\phi) \wedge (x_2 = x_2).
\end{array}
$$

Table 5.2: The standard translation for **2-MLP**.

5.6.1. DEFINITION. Let τ be the (first-order) vocabulary $\{P_1, P_2, \ldots, A_1, A_2, \ldots\}$, where the P_i's are unary relation symbols corresponding to the atomic set variables p_i in our language, and the A_i's are binary relation symbols corresponding to the atomic relation variables. Let $\mathcal{L}(\boldsymbol{\tau})$ be the set of all first-order formulas over $\boldsymbol{\tau}$ (with identity). Define a translation $ST(\cdot)$ taking $\mathcal{ML}_2$-formulas to formulas in $\mathcal{L}(\boldsymbol{\tau})$ as in Table 5.2.

5.6.2. PROPOSITION. *Let ϕ be a set formula in $\mathcal{ML}_2(\Phi, \Omega)$. For any Peirce model $\mathfrak{A}$, and any x in $\mathfrak{A}$, $\mathfrak{A}, x \models \phi$ iff $\mathfrak{A} \models ST(\phi)[x]$.*

Let α be a relation formula in $\mathcal{ML}_2(\Phi, \Omega)$. For any $\mathfrak{A}$, and any x, y in $\mathfrak{A}$, $\mathfrak{A}, (x, y) \models \alpha$ iff $\mathfrak{A} \models ST(\alpha)[xy]$.

5.6.3. THEOREM. (Expressive completeness) *Let $\boldsymbol{\tau}$ be as in Definition 5.6.1. Every first-order formula in the three-variable fragment $\mathcal{L}_3(x_1, x_2)$ of the first-order language over $\boldsymbol{\tau}$ has an equivalent to the ST-translation of a $\mathcal{ML}_2$-formula. Conversely, every $\mathcal{ML}_2$-formula translates into a $\mathcal{L}_3(x_1, x_2)$-formula under ST.*

Proof. The second half is immediate from 5.6.2. The first one follows from the fact that $\mathcal{ML}_2$-formulas and $\mathcal{DML}$-expressions coincide, combined with Definition 4.4.7 and the proof of Theorem 5.3.7. $\dashv$

DEFINABILITY ISSUES

As in §§3.4, 4.4 we use an appropriate notion to characterize the first-order translations of $\mathcal{ML}_2$-formulas. The arguments below should be familiar by now — the choice of our bisimulation may be somewhat surprising though.

A *2-partial isomorphism* f from $\mathfrak{M}$ to $\mathfrak{N}$ is simply an isomorphism $f : \mathfrak{M}_0 \cong \mathfrak{N}_0$, where $\mathfrak{M}_0$, $\mathfrak{N}_0$ are substructures of $\mathfrak{M}$ and $\mathfrak{N}$, respectively, whose domains have cardinality at most 2. A set $\mathbb{I}$ of 2-partial isomorphisms from $\mathfrak{M}$ into $\mathfrak{N}$ has the *back-and-forth property* if

> for every $f \in \mathbb{I}$ with $|f| \leq 1$, and every $x \in \mathfrak{M}$ (or $y \in \mathfrak{N}$) there is a $g \in \mathbb{I}$ with $f \subseteq g$ and $x \in \text{domain}(g)$ (or $y \in \text{range}(g)$).

I write $\mathbb{I} : \mathfrak{M} \cong_2^p \mathfrak{N}$ if $\mathbb{I}$ is a non-empty set of 2-partial isomorphisms and $\mathbb{I}$ has the back-and-forth property. A family of 2-partial isomorphisms has the *triangle property* if for every $f = \{ (x_1, y_1), (x_2, y_2) \} \in \mathbb{I}$ and x_3 in $\mathfrak{M}$ there is a y_3 in $\mathfrak{N}$ and g, $g' \in \mathbb{I}$ such that $g = \{ (x_1, y_1), (x_3, y_3) \}$ and $g' = \{ (x_2, y_2), (x_3, y_3) \}$ (and a similar requirement in the opposite direction).

By 5.6.3 $\mathcal{ML}_2$ contains the equivalent of the full 2-variable fragment of $\mathcal{L}(\boldsymbol{\tau})$. Hence, as the latter is characterized by its invariance under 2-partial isomorphisms, any relation between models that is to preserve truth of $\mathcal{ML}_2$-formulas should at least act like a family of 2-partial isomorphisms. Indeed, modulo one additional requirement the latter completely characterizes $\mathcal{ML}_2$.

5.6.4. DEFINITION. A *bisimulation for $\mathcal{ML}_2$* between $\mathfrak{M}_1$ and $\mathfrak{M}_2$ is a non-empty relation $Z \subseteq (W_1 \times W_2) \cup (W_1^2 \times W_2^2)$ such that

1. $Z\vec{x}\vec{y}$ implies $\text{lh}(\vec{x}) = \text{lh}(\vec{y})$, where $\text{lh}(\vec{x})$ is the length of $\vec{x}$,
2. if $Z(x_1x_2)(y_1y_2)$ then Zx_1y_1, Zx_2y_2, and $Z(x_2x_1)(y_2y_1)$,
3. if Zx_1y_1 then x_1 and y_1 agree on all set variables p,

4. if $Z(x_1x_2)(y_1y_2)$ then (x_1, x_2) and (y_1, y_2) agree on all relation variables a and on δ,
5. if Zx_1y_1 and $x_2 \in \mathfrak{M}_1$, then there exists y_2 in $\mathfrak{M}_2$ such that $Z(x_1x_2)(y_1y_2)$, and similarly in the opposite direction,
6. if $Z(x_1x_2)(y_1y_2)$ and $x_3 \in \mathfrak{M}_1$, then there exists y_3 in $\mathfrak{M}_2$ such that $Z(x_1x_3)(y_1y_3)$ and $Z(x_3x_2)(y_3y_2)$, and similarly in the opposite direction.

5.6.5. PROPOSITION. *Let Z be a bisimulation for $\mathcal{ML}_2$ between two models $\mathfrak{M}_1$ and $\mathfrak{M}_2$. Then for $\mathbb{I} = \{\emptyset\} \cup \{(x,y) : Zxy\} \cup \{(x_1, y_1), (x_2, y_2) : Z(x_1x_2)(y_1y_2)\}$ we have $\mathbb{I} : \mathfrak{M}_1 \cong_2^p \mathfrak{M}_2$.*

And conversely, if $\mathbb{I} : \mathfrak{M}_1 \cong_2^p \mathfrak{M}_2$ has the triangle property, then $Z = \{(x,y) : \exists f \in \mathbb{I}\, f(x) = y\} \cup \{(x_1x_2, y_1y_2) : \exists f \in \mathbb{I}\,(f(x_1) = y_1, f(x_2) = y_2)\}$ is an $\mathcal{ML}_2$-bisimulation.

5.6.6. REMARK. The remarks in Examples 4.4.13 and 4.4.14 concerning $\mathcal{DML}$-bisimulations transfer to the present setting. That is: on the class of all finite models isomorphism and bisimilarity for $\mathcal{ML}_2$ do not coincide. To see this one may use the models (and argument) of Example 4.4.13. Further, $\mathcal{ML}_2$-bisimilarity implies $\mathcal{ML}_2$-equivalence on on all models, while the converse also holds on finite models.

What is the relation between the earlier $\mathcal{DML}$-bisimulations of Definition 4.4.11 and bisimulations for $\mathcal{ML}_2$? The former only links single points, while the latter links single points as well as pairs of points. As a result bisimulations for $\mathcal{DML}$ can preserve $\mathcal{DML}$-expressions whose standard translation involves *two* free variables only because we explicitly impose this. As bisimulations for $\mathcal{ML}_2$ may link pairs of points, the required preservation is achieved in a compositional way by demanding that those bisimulations act as a collection of 2-partial isomorphisms. In addition $\mathcal{ML}_2$-bisimulations lack the 'linguistic' character; hence as an algebraic description of modal equivalence they are preferable.

5.6.7. THEOREM. *A first-order formula ξ in $\mathcal{L}(\tau)$ having one or two free variables is (equivalent to) the translation of an $\mathcal{ML}_2$-formula iff it is invariant for $\mathcal{ML}_2$-bisimulations.*

Proof. See Example 6.7.6. ⊣

As a corollary to Theorem 5.6.7 we have that a first-order formula in one free variable is equivalent to the translation of a $\mathcal{DML}$-formula iff it is invariant for $\mathcal{ML}_2$-bisimulations (compare Remark 4.4.12).

As in earlier Chapters, having the right notion of bisimulation available allows one to obtain a variety of definability results. For an elegant formulation of these results it is convenient to consider pointed models as our fundamental structures. Here, a *pointed model* is a structure of the form $(\mathfrak{F}, V, w)$ or $(\mathfrak{F}, V, (w, v))$, where $(\mathfrak{F}, V)$ is an ordinary Peirce model, and w, v in W.

5.6.8. THEOREM. *Let K be a class of pointed models of the form $(\mathfrak{F}, V, w)$, or a class of pointed models of the form $(\mathfrak{F}, V, (w, v))$. Then K is definable by means* *of a $\mathcal{ML}_2$-formula iff it is closed under $\mathcal{ML}_2$-bisimulations and ultraproducts, while its complement is closed under ultraproducts.*

Proof. See Example 6.7.6. ⊣

5.7 Concluding remarks

In this Chapter we studied axiomatic aspects of representable Peirce algebras. After a brief sketch of areas where such algebras arise, we used the modal completeness of a system of dynamic modal logic to arrive at a quick completeness result for the set equations valid in FPA. Most of the work done in this Chapter went into obtaining a complete axiomatization of all identities (both set identities and relation identities) valid in FPA; modal techniques from Chapters 3 and 7 were put to work here. Finally, the general results from Chapter 6 were applied to find results on definability and expressive power.

A lot remains to be done. Some questions were already mentioned in the main body of the Chapter: to give an axiomatization of the relation equations valid in FPA, and to determine whether the irreflexivity rules (IR_s) and (IR_d) are indeed necessary for obtaining a complete axiomatization of FPA. To conclude, here are further questions.

1. Is there a decent Gentzen-style deduction system for FPA? Building on work of Wadge (1975) Maddux (1983) develops a sequent system for relation algebras. Can his work be extended to Peirce algebras?
2. In §5.2 we briefly mentioned a connection between a system of Arrow Logic and Peirce algebras. There is a whole hierarchy of calculi in between this Arrow Logic and the **2-MLP**, the logic of Peirce algebras, just like there is a hierarchy of subsystems of relation algebra. About the former hierarchy one can ask the same kind of questions as for the latter. For example, where does undecidability strike? Is there an arrow version of Peirce algebras which is sufficiently expressive for applications (say, in terminological logic), but still decidable?
3. Another point in connection with the use of Peirce algebras in terminological logic is this. In terminological reasoning one often needs to be able to *count* the number of objects related to a given object; this is done using so-called *number restrictions* (Brink et al. 1993). The modal logic of such counting expressions is analyzed by Van der Hoek & De Rijke (1992, 1993). An obvious topic for further work is to combine the results of the latter with the results of the present Chapter.
4. Finally, a more general point. Both here and in earlier Chapters we have used unorthodox derivation rules like (IR_{Ds}) to arrive at our completeness results. To which extent do such rules capture our operators? We know from Chapter 3 that the irreflexivity rule for D goes a long way towards determining the D-operator. But what about the other operators, like $\circ$, $\updownarrow$, $:$? Which aspects of their behaviour are determined by our unorthodox derivation rules?

Part III

Two General Themes

6

Modal Logic and Bisimulations

6.1 INTRODUCTION

Modal formulas can be interpreted in two ways: on *models* where the interpretation of proposition letters is handled via valuations, and on *frames* where one quantifies over all possible valuations for proposition letters. At present the frame theory for modal logic is in a more advanced state than its model theory, especially due to its connections with the theory of Boolean algebras with operators. This Chapter develops the model theory of a class of basic modal languages and others, using so-called basic bisimulations as the fundamental tool. The class of basic modal languages includes the standard modal language $\mathcal{ML}(\Diamond)$, and the notion of bisimulation appropriate for the latter, $\leftrightarroweq^{b}_{\tau_1}$, relates points in models that agree on all propositional symbols, while it has back-and-forth clauses to ensure that transitions in the one model are matched in the other.

The guiding theme of this Chapter is the 'equation'

$$\frac{\cong^{p}}{\text{first-order logic}} = \frac{\leftrightarroweq}{\text{basic modal logic}}.$$

That is: basic bisimulations are to basic modal logic, what partial isomorphism is for first-order logic. We substantiate this claim by establishing key results from first-order logic (and beyond) for modal logic, using bisimulations instead of (partial) isomorphism.

More specifically, after some background material has been presented in §6.2, §6.3 introduces basic bisimulations. In §6.4 these are linked to basic modal languages, resulting in an analogue of the Keisler-Shelah Theorem from first-order logic, as well as modal analogues of Karp's Theorem and the Scott Isomorphism Theorem from $\mathcal{L}_{\infty\omega}$ and $\mathcal{L}_{\omega_1\omega}$, respectively. Building on those results §6.5 supplies a series of definability results; it also presents a Lindström type characterization of basic modal logic. §6.6 pushes the idea that bisimulations are a fundamental tool in modal model theory even further by using them to establish modal analogues of three well-known preservation results from first-order logic: Loś's Theorem, the Chang-Łoś-Suszko Theorem, and Lyndon's Theorem. After that, in §6.7, we discuss extensions to of the results of §§6.4–6.6 to further modal languages, including those featured in Chapters 3–5. The final section, §6.8, is devoted to questions and suggestions for further work.

A short historical note: in modal logic bisimulations have been around since Van Benthem (1976); there they are called p-relations. In the computational tradition bisimulations date back to Park (1981). In essence bisimulations are trimmed down versions of the Ehrenfeucht games found in first-order logic (Doets & Van Benthem 1983). Further references, on modal and computational aspects of bisimulations, can be found in (Van Benthem & Bergstra 1993).

6.2 PRELIMINARIES

CLASSICAL LOGIC

We need a few notions and facts from first-order logic. For $\boldsymbol{\tau}$ a classical vocabulary, Str[$\boldsymbol{\tau}$] denotes the class of $\boldsymbol{\tau}$-structures.

For $\mathfrak{A}$, $\mathfrak{B}$ in Str[$\boldsymbol{\tau}$], a *partial isomorphism* is a set $\mathbb{I}$ of pairs $(\vec{a}, \vec{b})$ of tuples, with $\vec{a}$ from $\mathfrak{A}$ and $\vec{b}$ from $\mathfrak{B}$, such that

1. if $(\vec{a}, \vec{b})$ is in $\mathbb{I}$, then $\vec{a}$ and $\vec{b}$ have the same length and $(\mathfrak{A}, \vec{a})$ and $(\mathfrak{B}, \vec{b})$ satisfy the same atomic formulas,
2. $\mathbb{I}$ is not empty,
3. if $(\vec{a}, \vec{b})$ is in $\mathbb{I}$ and c is an element of $\mathfrak{A}$, then there exists d in $\mathfrak{B}$ such that $(\vec{a}c, \vec{b}d)$ is in $\mathbb{I}$, and
4. if $(\vec{a}, \vec{b})$ is in $\mathbb{I}$ and d is an element of $\mathfrak{B}$, then there exists c in $\mathfrak{A}$ such that $(\vec{a}c, \vec{b}d)$ is in $\mathbb{I}$.

We write $\mathbb{I} : \mathfrak{A} \cong^p \mathfrak{B}$ for $\mathbb{I}$ is a partial isomorphism from $\mathfrak{A}$ to $\mathfrak{B}$. Various key results about first-order logic may be proved using partial isomorphisms, including a weak version of the Keisler-Shelah Theorem that characterizes first-order definable classes of models in terms of partial isomorphisms and ultraproducts.

MODAL LOGIC

In this chapter we adopt the general approach towards modal logic outlined in Chapter 2. Thus, a modal language has modal operators # equipped with patterns $\delta_{\#}$ describing the semantics of # by means of a formula in classical logic. A large part of this Chapter deals with basic modal languages.

6.2.1. DEFINITION. For $\boldsymbol{\tau}$ a classical vocabulary with unary predicate symbols, the *basic modal language over* $\boldsymbol{\tau}$ is the *finitary* modal language $\mathcal{BML}(\boldsymbol{\tau})$ having proposition letters p_0, p_1, ... corresponding to the unary predicate symbols in $\boldsymbol{\tau}$, and also having n-ary modal operators # with patterns

$$\delta_{\#} = \lambda x.\, \exists x_1 \ldots \exists x_n \left(Rxx_1 \ldots x_n \wedge p_1(x_1) \wedge \ldots \wedge p_n(x_n)\right),$$

for every $(n+1)$-ary relation symbol R in $\boldsymbol{\tau}$. In addition $\mathcal{BML}(\boldsymbol{\tau})$ has the usual Boolean connectives, and constants $\bot$ and $\top$.

We also need infinitary basic modal languages. Let κ be a regular cardinal. The basic infinitary modal language $\mathcal{BML}_{\kappa\omega}$ has proposition letters, modal operators, connectives and constants as in $\mathcal{BML}(\tau)$, but it also has conjunctions $\bigwedge$ and disjunctions $\bigvee$ over sets of formulas of cardinality less than κ. We write $\mathcal{BML}_{\infty\omega}(\tau) = \bigcup_\kappa \mathcal{BML}_{\kappa\omega}(\tau)$, and for λ singular, $\mathcal{BML}_{\lambda\omega}(\tau) = \bigcup_{\kappa<\lambda} \mathcal{BML}_{\kappa\omega}(\tau)$.

Basic modal languages are interpreted on τ-structures of the form $(W, R_1, R_2, \ldots, P_1, P_2, \ldots)$, where P_1, P_2, ... interpret the proposition letters of the modal language. As usual we will let valuations V take care of proposition letters; thus we will write $(W, R_1, R_2, \ldots, V)$, where $V(p_i) = P_i$.

Using the patterns of a modal logic a translation ST can be defined that takes modal formulas to formulas in the classical language in which those patterns live. For basic modal languages the translation ST maps proposition letters onto unary predicates, it commutes with the Booleans, while

$$ST(\#(\phi_1, \ldots, \phi_n)) = \\ \exists y_1 \ldots \exists y_n \Big(Rxx_1 \ldots x_n \wedge ST(\phi_1)(x_1) \wedge \ldots \wedge ST(\phi_n)(x_n)\Big),$$

where the semantics of $\#$ is based on R. Then, for all basic modal formulas ϕ: $(W, R_1, R_2, \ldots, V), a \models \phi$ iff $(W, R_1, R_2, \ldots, V) \models ST(\phi)[a]$. This equivalence allows us to freely move back and forth between modal formulas and certain classical formulas. Also, as basic modal formulas are equivalent to their (classical) ST-translations, they inherit important properties of classical logic; for $\mathcal{BML}$-formulas this means that they enjoy the usual compactness and Löwenheim-Skolem properties (when interpreted on models).

We will adopt the following important

Convention

Throughout this Chapter models are always *pointed models* of the form $(\mathfrak{A}, a)$, where $\mathfrak{A}$ is a relational structure and a is an element of $\mathfrak{A}$, called its *distinguished point*, at which evaluation takes place.

Our main reasons for adopting this convention are the following. First, the basic semantic unit in modal logic simply *is* a model together with a distinguished node at which evaluation takes place. Second, there are various technical reasons for working with pointed models, the main one being that on the class of pointed models bisimilarity becomes an equivalence relation. Third, some results admit smoother formulations when we adopt the *local* perspective. Of course, this local perspective dates back (at least) to Kripke's original publication Kripke (1963). The usual *global* perspective ('$\mathfrak{A} \models \phi$ iff for all a in $\mathfrak{A}$: $\mathfrak{A}, a \models \phi$') is obviously definable using the local point of view.

6.3 Basic bisimulations

6.3.1. DEFINITION. For τ a classical vocabulary and $\mathfrak{A}$, $\mathfrak{B} \in \mathrm{Str}[\tau]$, we say that $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ are *basically τ-bisimilar* ($(\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{B}, b)$) if there exists a non-empty relation Z between the elements of $\mathfrak{A}$ and $\mathfrak{B}$ (called a *τ-bisimulation*, and written $Z : (\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{B}, b)$) such that

1. Z links the distinguished points of $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$: Zab,
2. for all unary predicate symbols P in τ, Za_0b_0 implies $a_0 \in P^{\mathfrak{A}}$ iff $b_0 \in P^{\mathfrak{B}}$,
3. if Za_0b_0, $a_1, \ldots, a_n \in \mathfrak{A}$ and $(a_0, a_1, \ldots, a_n) \in R^{\mathfrak{A}}$, then there are $b_1, \ldots, b_n \in \mathfrak{B}$ such that $(b_0, b_1, \ldots, b_n) \in R^{\mathfrak{B}}$ and Za_ib_i, where $1 \le i \le n$ and R is an $(n+1)$-ary relation symbol in τ (*forth condition*),
4. if Za_0b_0, $b_1, \ldots, b_n \in \mathfrak{B}$ and $(b_0, b_1, \ldots, b_n) \in R^{\mathfrak{B}}$, then there are $a_1, \ldots, a_n \in \mathfrak{A}$ such that $(a_0, a_1, \ldots, a_n) \in R^{\mathfrak{A}}$ and Za_ib_i, where $1 \le i \le n$ and R is an $(n+1)$-ary relation symbol in τ (*back condition*).

Basic bisimilarity satisfies the following general constraints:

isomorphism: $(\mathfrak{A}, a) \cong (\mathfrak{B}, b)$ implies $(\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{B}, b)$,
equivalence: (i) $(\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{A}, a)$, (ii) $Z : (\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{B}, b)$ implies $Z^{\smile} : (\mathfrak{B}, b) \leftrightarrows^b_\tau (\mathfrak{A}, a)$, (iii) $Z : (\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{B}, b)$ and $Z' : (\mathfrak{B}, b) \leftrightarrows^b_\tau (\mathfrak{C}, c)$, implies $(Z; Z') : (\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{C}, c)$,
union: $Z_i : (\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{B}, b)$ implies $\bigcup_i Z_i : (\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{B}, b)$.

Many familiar constructions on relational structures arise as special examples of basic bisimulations.

Disjoint unions. For disjoint τ-structures $(\mathfrak{A}_i, a_i)$ $(i \in I)$ their disjoint union is the structure $\mathfrak{A}$ which has the union of the domains of $\mathfrak{A}_i$ as its domain, while $R^{\mathfrak{A}} = \bigcup_i R^{\mathfrak{A}_i}$. For each of the components $\mathfrak{A}_i$ there is a basic bisimulation $Z : (\mathfrak{A}_i, a_i) \leftrightarrows^b_\tau (\mathfrak{A}, a_i)$ defined by Zxy iff $x = y$.

Generated submodels. $(\mathfrak{A}, a)$ is a generated submodel of $(\mathfrak{B}, b)$ whenever (i) $a = b$, (ii) the domain of $\mathfrak{A}$ is a subset of the domain of $\mathfrak{B}$, (iii) $R^{\mathfrak{A}}$ is simply the restriction of $R^{\mathfrak{B}}$ to $\mathfrak{A}$, and (iv) if $a_0 \in \mathfrak{A}$ and $R^{\mathfrak{A}}a_0b_1 \ldots b_n$, then $b_1, \ldots, b_n$ are in $\mathfrak{A}$. If $(\mathfrak{A}, a)$ is a generated submodel of $(\mathfrak{B}, b)$, there is a basic τ-bisimulation $Z : (\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{B}, b)$ defined by Zxy iff $x = y$.

P-morphisms. A mapping $f : (\mathfrak{A}, a) \to (\mathfrak{B}, b)$ is a p-morphism if (i) $f(a) = b$, (ii) it is a homomorphism for every $R \in \tau$, that is: $R^{\mathfrak{A}}aa_1 \ldots a_n$ implies $R^{\mathfrak{B}}f(a)f(a_1)\ldots f(a_n)$, and (iii) if $R^{\mathfrak{B}}f(a)b_1 \ldots b_n$ then there are $a_1, \ldots, a_n$ such that $R^{\mathfrak{A}}aa_1 \ldots a_n$ and $f(a_i) = b_i$. If $f : (\mathfrak{A}, a) \to (\mathfrak{B}, b)$ is a p-morphism, putting Zxy iff $f(x) = y$ defines a basic bisimulation $Z : (\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{B}, b)$.

Just like partial isomorphisms in Abstract Model Theory, bisimulations too are naturally built up by means of approximations. Let $\mathfrak{A}, \mathfrak{B} \in \mathrm{Str}[\tau]$. We define a notion of *basic τ-bisimilarity up to n* by requiring that there exists a sequence of binary relations $Z_0, \ldots, Z_n$ between $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ such that

1. $Z_n \subseteq \cdots \subseteq Z_0$ and Z_0ab,
2. for each i, if Z_ixy then x and y agree on all unary predicates,
3. for $i + 1 \le n$ the back-and-forth properties are satisfied relative to the indices:
   (a) if $Z_{i+1}xy$ and $R^{\mathfrak{A}}xx_1 \ldots x_m$, then for some $y_1, \ldots, y_m$ in $\mathfrak{B}$: $R^{\mathfrak{B}}yy_1 \ldots y_m$, and for all $j = 1, \ldots, m$: $Z_ix_jy_j$,

(b) if $Z_{i+1}xy$ and $R^{\mathfrak{B}}yy_1 \dots b_m$, then for some $x_1, \dots, x_m$ in $\mathfrak{A}$: $R^{\mathfrak{A}}xx_1 \dots x_m$, and for all $j = 1, \dots, m$: $Z_i x_j y_j$.

If there exist $Z_0, \dots, Z_n, \dots$ satisfying the above back-and-forth conditions, $Z = \bigcap_i Z_i$ defines a basic τ-bisimulation; and conversely, every basic τ-bisimulation may be obtained as such an intersection. If, for some n, there is a basic bisimulation up to n between $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$, we write $(\mathfrak{A}, a) \leftrightarroweq_{\tau}^{b,n} (\mathfrak{B}, b)$, and say that $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ are *basically τ-bisimilar up to n.*

We need two concepts for measuring certain aspects of models. First, for $\mathfrak{A}$ a model, c in $\mathfrak{A}$, define the *in-degree* of c to be

$$\left|\{\, \vec{a} \in \mathfrak{A}^{<\omega} : \text{ for some } R \in \tau \text{ and } i > 1,\ c = a_i \text{ and } R^{\mathfrak{A}} a_1 \dots a_i \dots a_n \,\}\right|.$$

Thus, the in-degree of c is the number of times it occurs as an argument in a relation: $Rx \dots c \dots$.

The second notion we need measures the distance from a given element in a model to its distinguished point. Let $(\mathfrak{A}, a)$ be a τ-structure; the *τ-hulls* H_{τ}^{κ} around a are defined as follows

- $H_{\tau}^{0}(\mathfrak{A}, a) = \{\, a \,\}$,
- $H_{\tau}^{\kappa+1}(\mathfrak{A}, a) = H_{\tau}^{\kappa}(\mathfrak{A}, a) \cup \{\, b$ in $\mathfrak{A}$: for some $R \in \tau$, $u \in H_{\tau}^{\kappa}(\mathfrak{A}, a)$ and $v_1, \dots, v_n$ in $\mathfrak{A}$: b is one of the v_i and $R^{\mathfrak{A}}uv_1 \dots v_n \,\}$,
- $H_{\tau}^{\lambda}(\mathfrak{A}, a) = \bigcup_{\kappa<\lambda} H_{\tau}^{\kappa}(\mathfrak{A}, a)$.

So, the τ-hull H_{τ}^{κ} around a contains all elements in $\mathfrak{A}$ that can be reached from a in at most κ relational steps

$$R_1 a a_{11} \dots a_{1n_1}, R_2 a_{1i} a_{21} \dots a_{2n_2}, \dots, R_{\kappa} a_{(\kappa-1)i} a_{\kappa 1} \dots a_{\kappa n_{\kappa}},$$

(assuming κ is a successor ordinal).

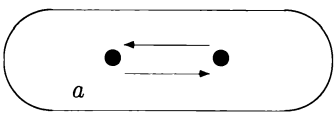
For c in $(\mathfrak{A}, a)$, the *depth of c in* $(\mathfrak{A}, a)$ is the smallest κ such that $c \in H_{\tau}^{\kappa}(\mathfrak{A}, a)$.

6.3.2. PROPOSITION. *Let $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ be two models such that every element has in-degree at most 1, and depth at most n. The following are equivalent:*

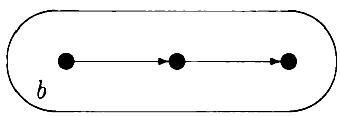
1. $(\mathfrak{A}, a) \leftrightarroweq_{\tau}^{b,n} (\mathfrak{B}, b)$,
2. $(\mathfrak{A}, a) \leftrightarroweq_{\tau}^{b} (\mathfrak{B}, b)$.

Proof. The implication $2 \Rightarrow 1$ is immediate. For the converse, assume $(\mathfrak{A}, a) \not\leftrightarroweq_{\tau}^{b} (\mathfrak{B}, b)$. Then there is some path through $\mathfrak{A}$ that cannot be matched with a bisimilar path in $\mathfrak{B}$ (or conversely). As $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ have depth $\leq n$, this path must have length $\leq n$. But then $(\mathfrak{A}, a) \not\leftrightarroweq_{\tau}^{b,n} (\mathfrak{B}, b)$. $\dashv$

Observe that the restriction to models with in-degree at most 1 in Proposition 6.4.5 is necessary. Consider $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ below.



$(\mathfrak{A}, a)$



$(\mathfrak{B}, b)$

Then $(\mathfrak{A}, a) \leftrightarroweq^{b,2} (\mathfrak{B}, b)$, and all elements have depth ≤ 2, but $(\mathfrak{A}, a) \not\leftrightarroweq^{b} (\mathfrak{B}, b)$.

Below we will want to get models that have nice properties, such as a low in-degree for each of its elements, or finite depth for each of its elements. To obtain such models the following comes in handy.

Fix a vocabulary τ. A property P of models is $\leftrightarroweq^b_\tau$*-enforceable*, or simply *enforceable*, iff for every $(\mathfrak{A}, a) \in \mathrm{Str}[\tau]$, there is a $(\mathfrak{B}, b) \in \mathrm{Str}[\tau]$ with $(\mathfrak{A}, a) \leftrightarroweq^b_\tau (\mathfrak{B}, b)$ and $(\mathfrak{B}, b)$ has P.

6.3.3. PROPOSITION. *The property "every element has finite depth" is enforceable.*

Proof. Let $\mathfrak{A} \in \mathrm{Str}[\tau]$, and let $(\mathfrak{B}, a)$ be the submodel of $\mathfrak{A}$ that is generated by $H^\omega_\tau(\mathfrak{A}, a)$. In $(\mathfrak{B}, a)$ every element has finite depth. Moreover, $(\mathfrak{A}, a) \leftrightarroweq^b_\tau (\mathfrak{B}, a)$, as $(\mathfrak{B}, a)$ is a generated submodel of $(\mathfrak{A}, a)$. $\dashv$

6.3.4. PROPOSITION. *Let $(\mathfrak{A}, a)$ a model, $(\mathfrak{B}, b)$ a generated submodel of $\mathfrak{A}$. The property "$(\mathfrak{A}, a)$ contains n copies of $\mathfrak{B}$" is enforceable* ($n \geq 1$).

Proof. Let $\mathfrak{A} \in \mathrm{Str}[\tau]$. Let $\mathfrak{B}$ be the generated submodel we want to copy. (We can assume that $\mathfrak{B}$ is a proper submodel of $\mathfrak{A}$, otherwise $\mathfrak{A}$ with a copy of $\mathfrak{B}$ added to it is simply the disjoint union of two copies of $\mathfrak{A}$.) It suffices to show that we can enforce the property of containing one extra copy of $\mathfrak{B}$.

Let $\mathfrak{B}'$ denote a disjoint copy of $\mathfrak{B}$. Add $\mathfrak{B}'$ to $\mathfrak{A}$ by linking elements in $\mathfrak{B}'$ to all and only the elements in $\mathfrak{A} \setminus \mathfrak{B}$ to which the corresponding original elements in $\mathfrak{B}$ are linked. Let $(\mathfrak{C}, a)$ be $(\mathfrak{A}, a)$ denote the result, and let Z denote the identity relation on $\mathfrak{A}$; so $Z : (\mathfrak{A}, a) \leftrightarroweq^b_\tau (\mathfrak{A}, a)$. Extend Z to a bisimulation $Z' : (\mathfrak{A}, a) \leftrightarroweq^b_\tau (\mathfrak{C}, a)$ by linking elements in $\mathfrak{B}'$ to the corresponding elements in the original $\mathfrak{B}$. $\dashv$

The copying construction of 6.3.4 can also be reversed: the property "contains exactly one copy of every generated submodel $\mathfrak{B}$" is also enforceable.

Proposition 6.3.5 below generalizes the *unraveling construction* from standard modal logic over the vocabulary τ_1 (Sahlqvist 1975) to arbitrary vocabularies; this generalization will be used frequently below.

6.3.5. PROPOSITION. *The property "every element has in-degree at most 1" is enforceable.*

Proof. We may assume that $(\mathfrak{A}, a)$ is generated by a. Expand τ to a vocabulary τ^+ that has constants for all elements in $\mathfrak{A}$. Define a *path conjunction* to be a first-order formula that is a conjunction of closed atomic formulas (over τ^+) taken from the smallest set X such that (i) $a = a$ is in X; (ii) $Rac_1 \ldots c_n$ is in X for any R and $c_1, \ldots, c_n$ such that $(\mathfrak{A}, a) \models Rac_1 \ldots c_n$; and (iii) if $\alpha \wedge Rcc_1 \ldots c_n$ is in X and for some S and i, $(\mathfrak{A}, c_i) \models Sc_i d_1 \ldots d_m$, then $\alpha \wedge Rcc_1 \ldots c_n \wedge Sc_i d_1 \ldots d_m$ is in X. A path conjunction $\alpha \equiv \alpha' \wedge Sdd_1 \ldots d_m$ is admissible for a constant c in $\tau^+ \setminus \tau$ if c is one of the d_i occurring in the last conjunct of α.

Define a model $\mathfrak{B}$ whose domain contains, for every constant c in $\tau^+ \setminus \tau$, a copy c_α, for every α that is admissible for c. Define $R^{\mathfrak{B}} cc_1 \ldots c_n$ to hold if each of the $c_1, \ldots, c_n$ is labeled with the same path conjunction $\alpha \equiv \alpha' \wedge Rcc_1 \ldots c_n$. And define a valuation V' on $\mathfrak{B}$ by putting $c_\alpha \in V(p)$ iff $c \in V(p)$.

Finally, define a relation Z between $\mathfrak{A}$ and $\mathfrak{B}$ by putting Zxy iff $y = x_\alpha$ for some path conjunction α. Then $Z : (\mathfrak{A}, a) \leftrightarroweq_\tau^b (\mathfrak{B}, a_{a=a})$. ⊣

For future purposes it is useful to observe that in a τ-model $(\mathfrak{A}, a)$ every element has in-degree at most 1 iff the model satisfies the following collection of first-order sentences:

$$\left\{ \neg\exists u \exists xy \exists \vec{x}\vec{y} \left(\bigvee_i (u = x_i) \wedge \bigvee_j (u = y_j) \wedge Rx\vec{x} \wedge Sy\vec{y} \right) : R, S \in \tau \right\}.$$

6.3.6. REMARK. The proofs of Propositions 6.3.3 and 6.3.5 show that the following property is also enforceable: "every element has finite depth and in-degree at most 1, and for all R and all R-tuples $(x, x_1, \ldots, x_n)$ we have that all x_i have the same finite depth (namely 1 plus the depth of x)."

6.4 MODAL EQUIVALENCE AND BISIMULATIONS

For $\mathcal{ML}(\tau)$ a basic modal language over τ, let $(\mathfrak{A}, a) \equiv_{\mathcal{ML}(\tau)} (\mathfrak{B}, b)$ denote that $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ satisfy the same $\mathcal{ML}(\tau)$-formulas. This section determines the exact relation between $\equiv_{\mathcal{ML}(\tau)}$ and $\leftrightarroweq_\tau^b$.

Intuitively, there is a close connection between basic τ-bisimulations and basic modal languages over τ. Both when checking the back-and-forth conditions, and when evaluating modal formulas, one *scans* models for the occurrence of certain relational patterns, namely R-tuples, for $R \in \tau$, satisfying some propositional information. More precisely, we have the following.

6.4.1. PROPOSITION. *Let τ be a classical vocabulary, and let $\mathcal{ML}(\tau)$ be a basic modal language over τ. Then $\leftrightarroweq_\tau^b \subseteq \equiv_{\mathcal{ML}(\tau)}$.*

A similar relation holds between finite approximations of bisimulations and restricted fragments of modal languages. To identify those fragments we need the following.

6.4.2. DEFINITION. Define the *rank* of a modal formula, rank(ϕ) as follows: rank(p) = 0, rank($\neg\phi$) = rank(ϕ), rank($\bigvee \Phi$) = sup({ rank(ϕ) : $\phi \in \Phi$ }), and rank($\#(\phi_1, \ldots, \phi_n)$) = 1 + max{ rank($\phi_i$) : $1 \leq i \leq n$}.

We write $(\mathfrak{A}, a) \equiv^n_{\mathcal{ML}(\tau)} (\mathfrak{B}, b)$ for $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ verify the same $\mathcal{ML}\rangle(\tau)$-formulas of rank at most n.

6.4.3. PROPOSITION. *Let τ be a classical vocabulary, and let $\mathcal{ML}(\tau)$ be any basic modal language over τ. Then $\leftrightarroweq_\tau^{b,n} \subseteq \equiv^n_{\mathcal{ML}(\tau)}$.*

6.4.4. PROPOSITION. *Let $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ be two finite models such that every element has in-degree at most 1, and depth at most n. The following are equivalent:*

1. $(\mathfrak{A}, a) \equiv^n_{\mathcal{BML}} (\mathfrak{B}, b)$,
2. $(\mathfrak{A}, a) \leftrightarroweq_\tau^{b,n} (\mathfrak{B}, b)$,
3. $(\mathfrak{A}, a) \equiv_{\mathcal{BML}} (\mathfrak{B}, b)$,

4. $(\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{B}, b)$.

Proof. The implication 4 $\Rightarrow$ 2 is Proposition 6.3.2. The implication 2 $\Rightarrow$ 1 is immediate, and the implication 3 $\Rightarrow$ 4 may be proved by an argument similar to the one in Theorem 6.4.10. To complete the proof we show that 1 implies 3. Assume $(\mathfrak{A}, a) \equiv^n_{\mathcal{BML}} (\mathfrak{B}, b)$. As both in $\mathfrak{A}$ and in $\mathfrak{B}$ every element has in-degree at most 1, all path conjunctions describing $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ (starting from a and b, respectively) contain at most $n + 1$ conjuncts. It follows that $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ agree on all formulas ϕ with $\mathrm{rank}(\phi) \geq n + 1$. By assumption, $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ agree on all formulas ϕ with $\mathrm{rank}(\phi) \leq n$; so $(\mathfrak{A}, a) \equiv_{\mathcal{BML}} (\mathfrak{B}, b)$. ⊣

As an important aside, the bisimulation machinery may used to establish the final model property of $\mathcal{BML}(\boldsymbol{\tau})$:

6.4.5. PROPOSITION. (Finite model property) *Let ϕ be a $\mathcal{BML}(\boldsymbol{\tau})$-formula. If ϕ is satisfiable, it is satisfiable on a finite model.*

Proof. Assume $(\mathfrak{A}, a) \models \phi$. By Proposition 6.3.5 we may assume that all elements of $(\mathfrak{A}, a)$ in-degree at most 1; we can also assume that $(\mathfrak{A}, a)$ is generated by a, and that every element has depth at most the rank of ϕ.

To complete the proof, it suffices to reduce the number elements of depth i, for $i \leq \mathrm{rank}(\phi)$ to a finite number. This is achieved as follows. First, we restrict ourselves to a finite vocabulary $\boldsymbol{\tau}_\phi \subseteq \boldsymbol{\tau}$ containing all and only the symbols that occur in (the standard translation of) ϕ. Second, let m be the maximum number such that each $R \in \boldsymbol{\tau}_\phi$ has arity at most m; so $m \leq |\phi|$, the length of ϕ. Finally, observe that for each $i = 0, \ldots, \mathrm{rank}(\phi)$, and each x of depth i, we need at most $|\phi|^2$ elements such that for some $R \in \boldsymbol{\tau}_\phi$, $Rx \ldots y \ldots$: ϕ can only contain $|\phi|$ many modal operators $\#_R$ asking for such elements. Hence ϕ must have a model with at most $|\phi|^3$ elements. ⊣

Returning to the relation between basic bisimulations and basic modal languages, observe that the converse of the inclusion in Proposition 6.4.1 does not hold: as is well-known from the general literature on bisimulations, there are $\mathcal{BML}$-equivalent models that are not bisimilar, witness the following example, cf. (Henessy & Milner 1985).

6.4.6. EXAMPLE. Let $\boldsymbol{\tau}$ be a vocabulary with just a single binary relation symbol R. Define models $\mathfrak{A}$ and $\mathfrak{B}$ as in Figure 6.1 below, where arrows denote R-transitions: Then $(\mathfrak{A}, a) \equiv_{\mathcal{BML}(\boldsymbol{\tau})} (\mathfrak{B}, b)$, but $(\mathfrak{A}, a) \not\leftrightarrows^b_\tau (\mathfrak{B}, b)$. The first

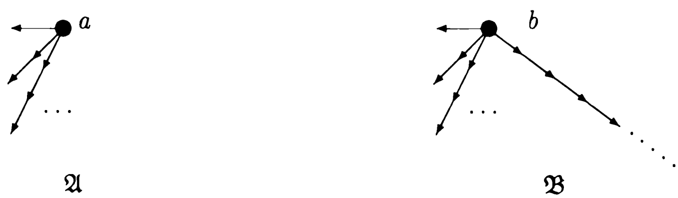


Figure 6.1: Equivalent but not bisimilar.

claim is obvious; to see that the second is true, observe that any candidate bisimulation Z has to link points on the infinite branch of $\mathfrak{B}$ to points of $\mathfrak{A}$ having only finitely many successors. This violates the back-and-forth conditions.

To determine the exact relation between $\underline{\leftrightarrow}^b_\tau$ and $\equiv_{\mathcal{BML}(\tau)}$ we need the following.

6.4.7. DEFINITION. A model $\mathfrak{A} \in \mathrm{Str}[\boldsymbol{\tau}]$ is said to be *ω-saturated* if for every finite subset Y of $\mathfrak{A}$, every type $\Gamma(x)$ of $\mathcal{L}_{\omega\omega}[\boldsymbol{\tau}^+]$, where $\boldsymbol{\tau}^+ = \boldsymbol{\tau} \cup \{ c_a : a \in Y \}$, that is consistent with $\mathrm{Th}_{\mathcal{L}_{\omega\omega}}((\mathfrak{A}, a)_{a \in Y})$ is realized in $(\mathfrak{A}, a)_{a\in Y}$. By a routine argument the restriction to types in a single free variable may be lifted to finitely many.

Recall that an ultrafilter is *countably incomplete* if it is not closed under arbitrary intersection.

6.4.8. LEMMA. (Keisler (1961)) *Let $\boldsymbol{\tau}$ be countable, $\mathfrak{A} \in \mathrm{Str}[\boldsymbol{\tau}]$, and let U be a countably incomplete ultrafilter over an index set I. The ultrapower $\prod_U \mathfrak{A}$ is ω-saturated.*

6.4.9. THEOREM. (Bisimulation Theorem) *Let $\mathfrak{A}$, $\mathfrak{B} \in \mathrm{Str}[\boldsymbol{\tau}]$. $\mathfrak{A} \equiv_{\mathcal{BML}(\boldsymbol{\tau})} \mathfrak{B}$ iff $\mathfrak{A}$ and $\mathfrak{B}$ have basically $\boldsymbol{\tau}$-bisimilar ultrapowers.*

Proof. The direction from right to left is obvious. For the converse, assume $(\mathfrak{A}, a) \equiv_{\mathcal{BML}(\tau)} (\mathfrak{B}, b)$. We construct elementary extensions $\mathfrak{A}' \succ \mathfrak{A}$ and $\mathfrak{B}' \succ \mathfrak{B}$, and a bisimulation between $\mathfrak{A}'$ and $\mathfrak{B}'$ that relates a and b.

First, let $\boldsymbol{\tau}^+ = \boldsymbol{\tau} \cup \{ c \}$, and expand $\mathfrak{A}$ and $\mathfrak{B}$ to $\boldsymbol{\tau}^+$-structures $\mathfrak{A}^+$ and $\mathfrak{B}^+$ by interpreting c as a in $\mathfrak{A}^+$, and as b in $\mathfrak{B}^+$. Let I be an infinite index set; by (Chang & Keisler 1973, Proposition 4.3.5) there is a countably incomplete ultrafilter U over I. By Lemma 6.4.8 the ultrapowers $\prod_U(\mathfrak{A}, a) =: (\mathfrak{A}', a')$ and $\prod_U(\mathfrak{B}, b) =: (\mathfrak{B}', b')$ are ω-saturated. Observe that both a' in $\mathfrak{A}'$ and b' in $\mathfrak{B}'$ realize the set of $\mathcal{BML}(\boldsymbol{\tau})$-formulas realized by a in $\mathfrak{A}$.

Define a relation Z on the universes of $\mathfrak{A}'$ and $\mathfrak{B}'$ by putting

$$Zxy \text{ iff for all } \mathcal{BML}(\boldsymbol{\tau})\text{-formulas } \phi\text{: } (\mathfrak{A}', x) \models \phi \text{ iff } (\mathfrak{B}', y) \models \phi.$$

We verify that Z is a basic $\boldsymbol{\tau}$-bisimulation. First, as a and b verify the same $\mathcal{BML}(\boldsymbol{\tau})$-formulas, Z must be non-empty. The condition on unary predicates is trivially met. To check the *forth* condition, assume Za_0b_0, $a_1, \ldots, a_n \in \mathfrak{A}'$, and $Ra_0a_1 \ldots a_n$ in $\mathfrak{A}'$. Define

$$\Psi_i(x_i) := \{ ST(\phi)(x_i) : \phi \in \mathcal{BML}(\boldsymbol{\tau}),\ \mathfrak{A}', a_i \models \phi \}\ (1 \leq i \leq n).$$

Then $\bigcup_i \Psi_i(x_i) \cup \{ R\underline{b}_0x_1 \ldots x_n \}$ is finitely satisfiable in $(\mathfrak{B}', b', b_0)$. To see this, assume $\Phi_i(x_i) \subseteq \Psi_i(x_i)$ is finite. Then

$$(\mathfrak{A}', a', a_0) \models \{ Rx_0x_1 \ldots x_n \} \cup \textstyle\bigcup_i \Phi_i(x_i)[a_1 \ldots a_n].$$

As Za_0b_0 and $\exists x_1 \ldots \exists x_n\, (Rx_0x_1 \ldots x_n \wedge \bigwedge \Phi_1(x_1) \wedge \ldots \wedge \bigwedge \Phi_n(x_n))$ is really a modal formula, it follows that for some $b_1, \ldots, b_n$ in $(\mathfrak{B}', b')$

$$(\mathfrak{B}', b', b_0) \models \{ Rx_0x_1 \ldots x_n \} \cup \textstyle\bigcup_i \Phi_i(x_i)[b_1 \ldots b_n],$$

Hence, by saturation, $(\mathfrak{B}', b', b_0) \models \bigcup_i \Psi_i(x_i) \cup \{ R\underline{b}_0 x_1 \dots x_n \}[b_1 \dots b_n]$ for some $b_1, \dots, b_n$ in $\mathfrak{B}'$. But then we have $Za_i b_i$ and $Rb_0 b_1 \dots b_n$ $(1 \leq i \leq n)$, as required. The *back* condition is checked similarly.

As $\mathfrak{A}'$ and $\mathfrak{B}'$ are reducts to the original vocabulary $\boldsymbol{\tau}$ of the ultrapowers $\prod_U(\mathfrak{A}, a)$ and $\prod_U(\mathfrak{B}, b)$, respectively, this shows that $\mathfrak{A}$ and $\mathfrak{B}$ have basically $\boldsymbol{\tau}$-bisimilar ultrapowers. $\dashv$

The Bisimulation Theorem should be compared to a weak version of the Keisler-Shelah Theorem in first-order logic: two first-order models are *elementary equivalent* iff they have *partially isomorphic* ultrapowers (Doets & Van Benthem 1983); the strong version of the result replaces 'partially isomorphic' with 'isomorphic' (Chang & Keisler 1973, Theorem 6.1.15).

Now that we know that finitary modal equivalence between two models means bisimilarity 'somewhere else,' viz., between certain ultrapowers of those models, the obvious next question is: for which modal language $\mathcal{L}$ *does* $\equiv_{\mathcal{L}}$ coincide with $\leftrightarroweq^b_{\boldsymbol{\tau}}$?

6.4.10. THEOREM. *The relations* $\leftrightarroweq^b_{\boldsymbol{\tau}}$ *and* $\equiv_{\mathcal{BML}_{\infty\omega}(\boldsymbol{\tau})}$ *coincide.*

Proof. The inclusion $\leftrightarroweq^b_{\boldsymbol{\tau}} \subseteq \equiv_{\mathcal{BML}_{\infty\omega}(\boldsymbol{\tau})}$ is immediate by an inductive argument. For the converse, we adopt an argument due to Henessy & Milner (1985). We show that the relation Z defined by Zab whenever a and b satisfy the same $\mathcal{BML}_{\infty\omega}(\boldsymbol{\tau})$-formulas is a basic $\boldsymbol{\tau}$-bisimulation. Assume it is not. If a_0 and b_0 disagree on some proposition letter, then they can't have the same $\mathcal{BML}_{\infty\omega}(\boldsymbol{\tau})$-theory. Hence, for some R and $a_1, \dots, a_n$ we have $Ra_0 a_1 \dots a_n$, while for all $b_1, \dots, b_n$ in $\mathfrak{B}$ $Rb_0 b_1 \dots b_n$ implies that for some i a_i and b_i disagree on some formula in $\mathcal{BML}_{\infty\omega}(\boldsymbol{\tau})$. Let $X = \{ (b_1, \dots, b_n) : Rb_0 b_1 \dots b_n \}$. Clearly $X \neq \emptyset$, and for every $(b_1, \dots, b_n) \in X$ there is an i such that for some ϕ_i $a_i \models \phi_i$ and $b_i \not\models \phi_i$ $(1 \leq i \leq n)$. Put $\Phi_i := \bigwedge \phi_i$ (letting the empty conjunction denote $\top$). Then, for $\#_R$ the modal operator whose semantics is based on R, we have $a_0 \models \#_R(\Phi_1, \dots, \Phi_n)$, but $b_0 \not\models \#_R(\Phi_1, \dots, \Phi_n)$, contradicting $Za_0 b_0$. $\dashv$

For countable structures, and vocabularies $\boldsymbol{\tau}$ containing no relation symbols of arity > 2, a sharper form of Theorem 6.4.10 is possible: Van Benthem & Bergstra (1993) show that, for $\boldsymbol{\tau}$ as above, countable structures are characterized up to basic bisimilarity by a single $\mathcal{BML}_{\omega_1\omega}(\boldsymbol{\tau})$-formula. The reader should compare this result with Scott's Isomorphism Theorem saying that countable structures are characterized up to isomorphism by a single $\mathcal{L}_{\omega_1\omega}$-sentence (Scott 1965).

6.4.11. THEOREM. (Van Benthem & Bergstra 1993) *Let* $\boldsymbol{\tau}$ *be a countable vocabulary containing no relation symbols of arity* > 2. *For every countable structure* $\mathfrak{A} \in \mathrm{Str}[\boldsymbol{\tau}]$ *there is a formula* ϕ *in* $\mathcal{BML}_{\omega_1\omega}(\boldsymbol{\tau})$ *such that for all countable* $\mathfrak{B}$ *we have* $\mathfrak{A} \leftrightarroweq^b_{\boldsymbol{\tau}} \mathfrak{B}$ *iff* $\mathfrak{B} \models \phi$.

Proof. For a in $\mathfrak{A}$ and $\lambda < \omega_1$, define the $\mathcal{BML}_{\omega_1\omega}(\boldsymbol{\tau})$-formula ϕ^λ_a inductively as follows.

- $\phi^0_a := \bigwedge\{ \chi : \mathfrak{A}, a \models \chi$ and χ is a (negation of a) proposition letter $\}$,
- $\phi^\lambda_a : \bigwedge_{\kappa<\lambda} \phi^\kappa_a$, if λ is a limit ordinal,

$$- \ \phi_a^{\lambda+1} := \phi_a^\lambda \wedge \bigwedge_{R\in\tau} \bigwedge_{\{a' : Raa'\}} \#_R \phi_{a'}^\lambda \wedge \bigwedge_{R\in\tau} \overline{\#}_R\Big(\bigvee_{\{a' : Raa'\}} \phi_{a'}^\lambda \Big). \qquad (*)$$

Observe that for all a in $\mathfrak{A}$, and $\lambda < \omega_1$, $\mathfrak{A}, a \models \phi_a^\lambda$. Furthermore, whenever $\kappa < \lambda < \omega_1$, then $\mathfrak{A} \models \phi_a^\lambda \to \phi_a^\kappa$. As $\mathfrak{A}$ is countable, for every a in $\mathfrak{A}$ there exists $\lambda < \omega_1$ such that for all $\kappa \geq \lambda$, $\mathfrak{A} \models \phi_a^\lambda \leftrightarrow \phi_a^\kappa$. It follows that there exists $\lambda < \omega_1$ such that for a in $\mathfrak{A}$ and all $\kappa \geq \lambda$, $\mathfrak{A} \models \phi_a^\lambda \leftrightarrow \phi_a^\kappa$. For this λ let ϕ_a be the infinitary modal formula

$$\phi_a^\lambda \wedge \bigwedge_{a'\in\mathfrak{A}} \bigwedge_{n\in\omega} \overline{\#}_{R_1} \cdots \overline{\#}_{R_n} \Big(\phi_{a'}^\lambda \to \phi_{a'}^{\lambda+1} \Big).$$

For countable $\mathfrak{B}$, putting Zab iff $\mathfrak{B}, b \models \phi_a$ defines a basic τ-bisimulation between $\mathfrak{A}$ and $\mathfrak{B}$. To see this, let us check the conditions. First, it is obvious that related points agree on proposition letters. Second, assume Zab, Raa' in $\mathfrak{A}$. Then

$$\begin{aligned}
(\mathfrak{B}, b) \models \phi_a \quad &\Rightarrow \quad (\mathfrak{B}, b) \models \phi_a^\lambda \\
&\Rightarrow \quad (\mathfrak{B}, b) \models \phi_a^{\lambda+1} \\
&\Rightarrow \quad (\mathfrak{B}, b) \models \#_R(\phi_{a'}^\lambda, \ldots, \phi_{a_n}^\lambda) \\
&\Rightarrow \quad (\mathfrak{B}, b') \models \phi_{a'}^\lambda, \quad \text{for some } b' \text{ in } \mathfrak{B} \text{ with } Rbb'.
\end{aligned}$$

In addition, every such b' has

$$(\mathfrak{B}, b') \models \bigwedge_{a'\in\mathfrak{A}} \bigwedge_{n\in\omega} \overline{\#}_{R_1} \cdots \overline{\#}_{R_n} \Big(\phi_{a'}^\lambda \to \phi_{a'}^{\lambda+1} \Big),$$

as every path starting from b' may be extended to a path starting from b because of Rbb'. In conclusion: Rbb' and $Za'b'$, as required.

For the back condition, assume Zab and Rbb' in $\mathfrak{B}$. Then

$$\begin{aligned}
(\mathfrak{B}, b) \models \phi_a \quad &\Rightarrow \quad (\mathfrak{B}, b) \models \phi_a^\lambda \\
&\Rightarrow \quad (\mathfrak{B}, b) \models \phi_a^{\lambda+1} \\
&\Rightarrow \quad (\mathfrak{B}, b) \models \overline{\#}_R\Big(\bigvee_{\{a' : Raa'\}} \phi_{a'}^\lambda \Big) \\
&\Rightarrow \quad (\mathfrak{B}, b') \models \phi_{a'}^\lambda, \quad \text{for some } a' \text{ with } Raa'.
\end{aligned}$$

Now, as we also have that

$$(\mathfrak{B}, b') \models \bigwedge_{a'\in\mathfrak{A}} \bigwedge_{n\in\omega} \overline{\#}_{R_1} \cdots \overline{\#}_{R_n} \Big(\phi_{a'}^\lambda \to \phi_{a'}^{\lambda+1} \Big),$$

it follows that $(\mathfrak{B}, b') \models \phi_{a'}$, so $Za'b'$, as required. $\dashv$

6.4.12. REMARK. Why did we restrict ourselves to vocabularies having only relation symbols of arity at most 2 in Theorem 6.4.11? The problem lies with the formula $(*)$: basically, this formula says which successors should 'at least' be seen, and which ones should not. When we try to adopt it to vocabularies containing relation symbols of arity > 2, one can easily force the 'at least' part; however, in basic modal languages containing modal operators of arity at least 2 it appears to be impossible to force the second part because of the disjunctive nature of the dual operators $\overline{\#}$.

6.5 DEFINABILITY AND CHARACTERIZATION

As stated before, the standard translation ST embeds our basic modal languages into fragments of classical languages. Combined with known definability results and techniques for the classical background languages, this fact allows for easy proofs of definability results for basic modal languages. The general strategy here is to 'bisimulate' results and proofs from classical logic, for instance by replacing $\cong$, $\cong^p$ and $\preccurlyeq$ with $\leftrightarroweq$. As a corollary we find that basic $\boldsymbol{\tau}$-bisimulations cut out precisely the basic modal fragment of first-order logic.

MODAL DEFINABILITY

We need some further definitions. A class of (pointed models) K is called an *$\mathcal{L}$-elementary class* (or: K is *EC* in $\mathcal{L}$) if $\mathsf{K} = \{ (\mathfrak{A}, a) : \mathfrak{A}, a \models \varphi$, for some $\mathcal{L}$-formula $\varphi \}$. We write K is EC_Δ in $\mathcal{L}$ if it is the intersection of classes that are *EC* in $\mathcal{L}$.

For K a class of models $\overline{\mathsf{K}}$ denotes the complement of K, $\mathbf{Pr}(\mathsf{K})$ denotes the class of ultraproducts of models in K, $\mathbf{Po}(\mathsf{K})$ denotes the class of ultrapowers of models in K, and $\mathbf{B}_b(\mathsf{K})$ is the class of all models that are basically bisimilar to a model in K.

6.5.1. PROPOSITION. *Let I be an index set, U an ultrafilter over I.*

1. *If for all i, $(\mathfrak{A}_i, a_i) \leftrightarroweq^b_\tau (\mathfrak{B}_i, b_i)$, then $\prod_U(\mathfrak{A}_i, a_i) \leftrightarroweq^b_\tau \prod_U(\mathfrak{B}_i, b_i)$,*
2. *If $(\mathfrak{A}, a) \leftrightarroweq^b_\tau (\mathfrak{B}, b)$, then $\prod_U(\mathfrak{A}, a) \leftrightarroweq^b_\tau \prod_U(\mathfrak{B}, b)$.*

Proof. 1. Assume that $Z_i : (\mathfrak{A}_i, a_i) \leftrightarroweq^b_\tau (\mathfrak{B}_i, b_i)$. For x in $\prod_U(\mathfrak{A}_i, a_i)$ and y in $\prod_U(\mathfrak{B}_i, b_i)$ define Zxy iff $\{ i \in I : Z_i x(i) y(i) \} \in U$. Then Z defines a basic bisimulation $\prod_U(\mathfrak{A}_i, a_i) \leftrightarroweq^b_\tau \prod_U(\mathfrak{B}_i, b_i)$ linking the distinguished points a and b of $\prod_U(\mathfrak{A}_i, a_i)$ and $\prod_U(\mathfrak{B}_i, b_i)$, respectively, where for all i in I, $a(i) = a_i$, $b(i) = b_i$.

2. This is immediate from item 1. (Alternatively, the diagonal map $d : a \to f_a$, where f_a is the constant map with value a, induces a bisimulation $(\mathfrak{A}, a) \leftrightarroweq^b_\tau \prod_U(\mathfrak{A}, a)$. Likewise, one has $(\mathfrak{B}, b) \leftrightarroweq^b_\tau \prod_U(\mathfrak{B}, b)$, hence $(\mathfrak{A}, a) \leftrightarroweq^b_\tau (\mathfrak{B}, b)$ yields $\prod_U(\mathfrak{A}, a) \leftrightarroweq^b_\tau \prod_U(\mathfrak{B}, b)$.) $\dashv$

6.5.2. COROLLARY. *Let K be a class of $\boldsymbol{\tau}$-models.*

1. $\mathbf{Pr}\mathbf{B}_b(\mathsf{K}) \subseteq \mathbf{B}_b\mathbf{Pr}(\mathsf{K})$, *hence K is closed under basic bisimulations and ultraproducts iff $\mathsf{K} = \mathbf{B}_b\mathbf{Pr}(\mathsf{K})$,*
2. $\mathbf{Po}\mathbf{B}_b(\mathsf{K}) \subseteq \mathbf{B}_b\mathbf{Po}(\mathsf{K})$, *hence K is closed under basic bisimulations and ultrapowers iff $\mathsf{K} = \mathbf{B}_b\mathbf{Po}(\mathsf{K})$.*

Proof. 1. Assume $(\mathfrak{A}, a) \in \mathbf{PrB}_b(\mathsf{K})$. Then there are an index set I, models $(\mathfrak{A}_i, a_i)$ and $(\mathfrak{B}_i, b_i)$ $(i \in I)$ such that $(\mathfrak{B}_i, b_i) \in \mathsf{K}$, $(\mathfrak{A}_i, a_i) \leftrightarrows_\tau^b (\mathfrak{B}_i, b_i)$, and $(\mathfrak{A}, a) = \prod_U(\mathfrak{A}_i, a_i)$, for some ultrafilter U over I. Trivially, $\prod_U(\mathfrak{B}_i, b_i) \in \mathbf{Pr}(\mathsf{K})$. By Proposition 6.5.1 $(\mathfrak{A}, a) = \prod_U(\mathfrak{A}_i, a_i) \leftrightarrows_\tau^b \prod_U(\mathfrak{B}_i, b_i)$. Hence, $(\mathfrak{A}, a) \in \mathbf{B}_b\mathbf{Pr}(\mathsf{K})$. As a consequence, if $\mathbf{B}_b\mathbf{Pr}(\mathsf{K}) = \mathsf{K}$, then, as both $\mathbf{B}_b$ and $\mathbf{Pr}$ are idempotent, applying $\mathbf{B}_b$ or $\mathbf{Pr}$ does not take us outside K; this is clear for $\mathbf{B}_b$, and for $\mathbf{Pr}$ we have $\mathbf{PrB}_b\mathbf{Pr}(\mathsf{K}) \subseteq \mathbf{B}_b\mathbf{PrPr}(\mathsf{K}) \subseteq \mathbf{B}_b\mathbf{Pr}(\mathsf{K}) \subseteq \mathsf{K}$.

2. This may be proved analogously to 1. $\dashv$

6.5.3. THEOREM. (Definability Theorem) *Let $\mathcal{L}$ denote $\mathcal{BML}(\boldsymbol{\tau})$, and let K be a class of $\boldsymbol{\tau}$-models. Then*

1. K *is EC_Δ in $\mathcal{L}$ iff* $\mathsf{K} = \mathbf{B}_b\mathbf{Pr}(\mathsf{K})$ *and* $\overline{\mathsf{K}} = \mathbf{B}_b\mathbf{Po}(\overline{\mathsf{K}})$,
2. K *is EC in $\mathcal{L}$ iff* $\mathsf{K} = \mathbf{B}_b\mathbf{Pr}(\mathsf{K})$ *and* $\overline{\mathsf{K}} = \mathbf{B}_b\mathbf{Pr}(\overline{\mathsf{K}})$.

Proof. 1. The *only if* direction is easy. For the converse, assume K is closed under ultraproducts and basic bisimulations, while $\overline{\mathsf{K}}$ is closed under ultrapowers. Let

$$T = \mathrm{Th}_{\mathcal{L}}(\mathsf{K}) = \{\, \phi : (\mathfrak{A}, a) \models \phi, \text{ for all } (\mathfrak{A}, a) \in \mathsf{K} \,\}.$$

Then $\mathsf{K} \models T$. Let $(\mathfrak{B}, b) \models T$. Let $\Sigma = \mathrm{Th}_{\mathcal{L}}(\mathfrak{B}, b)$, and define $I = \{\sigma \subseteq \Sigma : |\sigma| < \omega\}$. For each $i = \{\sigma_1, \ldots, \sigma_n\} \in I$ there is a model $(\mathfrak{A}_i, a_i)$ of i. By standard model-theoretic arguments there exists an ultraproduct $\prod_U(\mathfrak{A}_i, a_i)$ which is a model of Σ. As $\mathbf{Pr}(\mathsf{K}) \subseteq \mathsf{K}$, $\prod_U(\mathfrak{A}_i, a_i) \in \mathsf{K}$. But, if $(\mathfrak{A}, a) \models \Sigma$, then $(\mathfrak{A}, a) \equiv_{\mathcal{L}} (\mathfrak{B}, b)$, so $\prod_U(\mathfrak{A}_i, a_i) \equiv_{\mathcal{L}} (\mathfrak{B}, b)$. By the Bisimulation Theorem there is an ultrafilter U' such that $\prod_{U'}(\prod_U(\mathfrak{A}_i, a_i)) \leftrightarrows_\tau^b \prod_{U'}(\mathfrak{B}, b)$. Hence, the latter is in K, and, by the closure condition on $\overline{\mathsf{K}}$, this implies $(\mathfrak{B}, b) \in \mathsf{K}$. Therefore, K is the class of all models of T, and so K is EC_Δ in $\mathcal{L}$.

2. Again, the *only if* direction is easy. Assume K, $\overline{\mathsf{K}}$ satisfy the stated conditions. Then both are closed under ultrapowers, hence, by item 1, there are sets of $\mathcal{L}$-formulas T_1, T_2 witnessing that K is EC_Δ in $\mathcal{L}$, and that $\overline{\mathsf{K}}$ is EC_Δ in $\mathcal{L}$, respectively. Obviously, $T_1 \cup T_2 \models \bot$, so by compactness for some $\phi_1, \ldots, \phi_n \in T_1$, $\psi_1, \ldots, \psi_m \in T_2$, we have $\bigwedge_i \phi_i \models \bigvee_j \neg\psi_j$. Then K is the class of all models of $\bigwedge_i \phi_i$. $\dashv$

The definability results for first-order logic that correspond to Theorem 6.5.3 say that a class of models K is EC_Δ in first-order logic iff $\mathsf{K} = \mathbf{IPr}(\mathsf{K})$ and $\overline{\mathsf{K}} = \mathbf{IPo}(\overline{\mathsf{K}})$, and similarly for EC classes in first-order logic.

6.5.4. COROLLARY. (Separation Theorems) *Let $\mathcal{L}$ denote $\mathcal{BML}(\boldsymbol{\tau})$. Let K, L be classes of $\boldsymbol{\tau}$-models such that $\mathsf{K} \cap \mathsf{L} = \emptyset$.*

1. *If* $\mathbf{B}_b\mathbf{Pr}(\mathsf{K}) = \mathsf{K}$, $\mathbf{B}_b\mathbf{Po}(\mathsf{L}) = \mathsf{L}$, *then there exists a class M that is EC_Δ in $\mathcal{L}$ with $\mathsf{K} \subseteq \mathsf{M}$ and $\mathsf{L} \cap \mathsf{M} = \emptyset$,*
2. *If* $\mathbf{B}_b\mathbf{Pr}(\mathsf{K}) = \mathsf{K}$, $\mathbf{B}_b\mathbf{Pr}(\mathsf{L}) = \mathsf{L}$, *then there exists a class M that is EC in $\mathcal{L}$ with $\mathsf{K} \subseteq \mathsf{M}$ and $\mathsf{L} \cap \mathsf{M} = \emptyset$.*

Proof. 1. Let K' be the class of all $\boldsymbol{\tau}$-models $(\mathfrak{A}, a)$ such that for some $(\mathfrak{B}, b) \in \mathsf{K}$, $(\mathfrak{A}, a) \equiv_{\mathcal{L}} (\mathfrak{B}, b)$. Define L' similarly. Then $\mathsf{K} \subseteq \mathsf{K}'$, $\mathsf{L} \subseteq \mathsf{L}'$ and K' and L' are both closed under $\equiv_{\mathcal{L}}$.

Our first claim is that $\mathsf{K}' \cap \mathsf{L}' = \emptyset$. For suppose $(\mathfrak{A}, a) \in \mathsf{K}' \cap \mathsf{L}'$; then there exist $(\mathfrak{B}, b) \in \mathsf{K}$, $(\mathfrak{C}, c) \in \mathsf{L}$ such that $(\mathfrak{B}, b) \equiv_{\mathcal{L}} (\mathfrak{A}, a) \equiv_{\mathcal{L}} (\mathfrak{C}, c)$. By the Bisimulation Theorem $(\mathfrak{B}, b)$ and $(\mathfrak{C}, c)$ have basically $\boldsymbol{\tau}$-bisimilar ultrapowers $\prod_U(\mathfrak{B}, b)$ and $\prod_U(\mathfrak{C}, c)$. As K, L are closed under $\mathbf{B}_b$ and $\mathbf{Po}$, this implies $\prod_U(\mathfrak{B}, b) \in \mathsf{K} \cap \mathsf{L}$, contradicting $\mathsf{K} \cap \mathsf{L} = \emptyset$.

Let $T = \mathrm{Th}_{\mathcal{L}}(\mathsf{K}')$. Then K' is the class of models of T. As $\mathsf{K} \subseteq \mathsf{K}'$ and $\mathsf{K}' \cap \mathsf{L} = \emptyset$, we are done.

2. This may be proved analogously to 1. Use the assumption that $\mathbf{B}_b\mathbf{Pr}(\mathsf{L}) = \mathsf{L}$ to conclude that L' is EC_Δ in $\mathcal{L}$, and then apply a compactness argument as in the proof of Theorem 6.5.3, part 2. $\dashv$

As with the Definability Theorem the Separation Theorem is 'bisimilar' to corresponding results in first-order logic: there two disjoint classes K and L are separable by an EC_Δ class whenever $\mathsf{K} = \mathbf{IPr}(\mathsf{K})$ and $\mathsf{L} = \mathbf{IPo}(\mathsf{L})$; a result similar to part 2 of Corollary 6.5.4 holds for separation by means of a class that is EC in first-order logic: replace $\mathbf{B}_b$ with $\mathbf{I}$.

Observe that the Craig Interpolation Theorem is a special case of 6.5.4:

6.5.5. THEOREM. *If K, L are EC in $\mathcal{BML}(\boldsymbol{\tau}')$ for some $\boldsymbol{\tau}' \supseteq \boldsymbol{\tau}$, and $\mathsf{K} \cap \mathsf{L} = \emptyset$, then there is a class M that is EC in $\mathcal{BML}(\boldsymbol{\tau})$ with $\mathsf{K} \subseteq \mathsf{M}$ and $\mathsf{M} \cap \mathsf{L} = \emptyset$.*

The Definability Theorem 6.5.3 is difficult to apply in practice, as ultrapowers are rather abstract objects. The following Fraïssé type result supplies a more manageable criterion for EC classes.

6.5.6. THEOREM. *Let $\boldsymbol{\tau}$ be a finite vocabulary, and let K be a class of $\boldsymbol{\tau}$-models. Then K is EC in $\mathcal{BML}(\boldsymbol{\tau})$ iff, for some $n \in \mathbb{N}$, K is closed under basic $\boldsymbol{\tau}$-bisimulations up to n.*

Proof. The *only if* direction is clear. If K is closed under basic $\boldsymbol{\tau}$-bisimulations up to n, let $(\mathfrak{A}, a) \in \mathsf{K}$, and define $\phi_{(\mathfrak{A},a)}$ to be the conjunction of all $\mathcal{BML}$-formulas of rank at most n that are true at a. (Observe that over a finite vocabulary there are only finitely many basic modal formulas of any given rank). Modulo equivalence there are only finitely many such formulas $\phi_{(\mathfrak{A},a)}$ for $(\mathfrak{A}, a) \in \mathsf{K}$; let Φ be their disjunction. Then Φ defines K. For let $(\mathfrak{B}, b) \models \Phi$; then $(\mathfrak{B}, b) \equiv^n_{\mathcal{BML}} (\mathfrak{A}, a)$ for some $(\mathfrak{A}, a) \in \mathsf{K}$. By a routine induction, $(\mathfrak{B}, b) \leftrightarroweq^{b,n}_{\tau} (\mathfrak{A}, a)$; hence $(\mathfrak{B}, b) \in \mathsf{K}$. $\dashv$

To conclude our list of results on definability we give a theorem that characterizes the modal fragment of first-order logic. For the standard modal language $\mathcal{ML}(\Diamond)$ a semantic description of the corresponding first-order fragment in terms of bisimulations was first given by Van Benthem (1976, Theorem 1.9).

We need a definition. Let $\alpha(x)$ be a first-order formula over $\boldsymbol{\tau}$; α is called *invariant for basic $\boldsymbol{\tau}$-bisimulations* if for all $(\mathfrak{A}, a)$, $(\mathfrak{B}, b) \in \mathrm{Str}[\boldsymbol{\tau}]$, all basic $\boldsymbol{\tau}$-bisimulations $Z : (\mathfrak{A}, a) \leftrightarroweq^b_{\tau} (\mathfrak{B}, b)$, and all $x \in \mathfrak{A}$, $y \in \mathfrak{B}$ we have that Zxy implies $\mathfrak{A} \models \alpha[x]$ iff $\mathfrak{B} \models \alpha[y]$.

6.5.7. THEOREM. (Fragment Theorem) *Let $\alpha(x)$ be a first-order formula over $\boldsymbol{\tau}$. The following are equivalent.*

1. *$\alpha(x)$ is equivalent to (the ST-translation of) a modal formula in $\mathcal{BML}(\boldsymbol{\tau})$.*
2. *$\alpha(x)$ is invariant under basic $\boldsymbol{\tau}$-bisimulations.*
3. *for some $n \in \mathbb{N}$, α is invariant under basic $\boldsymbol{\tau}$-bisimulations up to n.*

Proof. The implications $1 \Rightarrow 3$ and $3 \Rightarrow 2$ are trivial. To complete the proof we show the implication $2 \Rightarrow 1$. Let K be the class of models of $\alpha(x)$. Then K and $\overline{\mathsf{K}}$ (being defined by $\neg\alpha(x)$) are closed under ultraproducts. As α is invariant under $\leftrightarroweq^b_\tau$, it follows that $\mathsf{K} = \mathbf{B}_b\mathbf{Pr}(\mathsf{K})$ and $\overline{\mathsf{K}} = \mathbf{B}_b\mathbf{Pr}(\overline{\mathsf{K}})$. By Theorem 6.5.3 K must be *EC* in $\mathcal{BML}(\boldsymbol{\tau})$. This means that α is equivalent to (the translation of) some modal formula ϕ. $\dashv$

Characterizing basic modal logic

The distinguishing feature of any modal logic is that it has means to talk about membership of subsets of the domain of a model (through proposition letters), and that it can talk about (combinations of) simple relational patterns. The (finitary) basic modal language $\mathcal{BML}(\boldsymbol{\tau})$ adds to this a *local* way of evaluating formulas — this is brought out most clearly by two facts: (i) the basic modal language has a notion of rank which, for a given formula ϕ uniformly bounds the depth of the model checking procedure for ϕ (cf. 6.4.2); and (ii) the language enjoys the finite model property (6.4.5). In this section we prove a result which shows that basic modal logic is the only (modal) logic with the above properties. To state the result we need a general notion of an abstract modal logic.

6.5.8. Definition. An *abstract modal logic* is a pair $(\mathcal{L}, \models_\mathcal{L})$ with the following properties; $\mathcal{L}$ is its set of formulas, and $\models_\mathcal{L}$ is its satisfaction relation. Of the following, (i), (ii) and (iii) are simple 'book keeping properties,' (iv) determines the 'basic modal character' of $(\mathcal{L}, \models_\mathcal{L})$.

(i) *Occurrence property.* For each ϕ in $\mathcal{L}$ there is an associated finite language $\mathcal{L}(\boldsymbol{\tau}_\phi)$. The relation $(\mathfrak{A}, a) \models_\mathcal{L} \phi$ is a relation between $\mathcal{L}$-formulas ϕ and structures $(\mathfrak{A}, a)$ for languages $\mathcal{L}$ containing $\mathcal{L}(\boldsymbol{\tau}_\phi)$. That is, if ϕ is in $\mathcal{L}$, and $\mathfrak{A}$ is an $\mathcal{L}$-model, then the statement $(\mathfrak{A}, a) \models_\mathcal{L} \phi$ is either true or false if $\mathcal{L}$ contains $\mathcal{L}(\boldsymbol{\tau}_\phi)$, and undefined otherwise.

(ii) *Expansion property.* The relation $(\mathfrak{A}, a) \models_\mathcal{L} \phi$ depends only on the reduct of $\mathfrak{A}$ to $\mathcal{L}(\boldsymbol{\tau}_\phi)$. That is, if $(\mathfrak{A}, a) \models_\mathcal{L} \phi$ and $(\mathfrak{B}, a)$ is an expansion of $(\mathfrak{A}, a)$ to a larger language, then $(\mathfrak{B}, b) \models_\mathcal{L} \phi$.

(iii) *Renaming property.* The relation $(\mathfrak{A}, a) \models_\mathcal{L} \phi$ is preserved under renaming. That is: if ρ is a bijection from $\mathcal{L}$ to $\rho\mathcal{L}$ which preserves the arity of operators, and if for each $(\mathfrak{A}, a)$, $\rho(\mathfrak{A}, a)$ is the model for $\rho\mathcal{L}$ induced in the obvious way by ρ, then for each $\phi \in \mathcal{L}$ there is a formula $\rho\phi$ in $\rho\mathcal{L}$ with $\mathcal{L}(\boldsymbol{\tau}_{\rho\phi}) = \rho\mathcal{L}(\boldsymbol{\tau}_\phi)$ such that for each $\mathcal{L}$-structure $(\mathfrak{A}, a)$, $(\mathfrak{A}, a) \models_\mathcal{L} \phi$ iff $\rho(\mathfrak{A}, a) \models_\mathcal{L} \rho\phi$.

(iv) *Bisimilarity property.* The relation $(\mathfrak{A}, a) \models_\mathcal{L} \phi$ is preserved under basic bisimulations: if $(\mathfrak{A}, a) \leftrightarroweq^b_\tau (\mathfrak{B}, b)$, then $(\mathfrak{B}, b) \models_\mathcal{L} \phi$.

(v) *Localization property.* For every pair of $\mathcal{L}$-formulas ϕ, ψ there is a new formula $(\phi \downarrow \psi)$, read ϕ *localized to* ψ, which lives in the vocabulary of ϕ and ψ taken together, and is such that whenever $(\mathfrak{B}, a)$ is the submodel with universe $B = \{\, x \text{ in } \mathfrak{A} : (\mathfrak{A}, x) \models \psi \,\}$, we have

$$(\mathfrak{A}, a) \models_{\mathcal{L}} (\phi \downarrow \psi) \text{ iff } (\mathfrak{B}, a) \models_{\mathcal{L}} \phi.$$

The most familiar example of an abstract modal logic is basic modal logic. The Localization property (v) for the basic modal language holds where ϕ localized to ψ is the formula inductively defined by:

$$\begin{array}{lcllcl} \bot \downarrow \psi & = & \bot & (\neg\phi) \downarrow \psi & = & \psi \wedge \neg(\phi \downarrow \psi) \\ p \downarrow \psi & = & p \wedge \psi & (\phi_1 \wedge \phi_2) \downarrow \psi & = & (\phi_1 \downarrow \psi) \wedge (\phi_2 \downarrow \psi) \end{array}$$
$$\#(\psi_1, \ldots, \psi_n) \downarrow \psi = \psi \wedge \#\Big(\psi \wedge (\phi_1 \downarrow \psi), \ldots, \psi \wedge (\phi_n \downarrow \psi)\Big).$$

6.5.9. Remark. The reader should compare the five defining properties of an abstract modal logic in Definition 6.5.8 to the list of defining properties for an abstract classical logic. For example, Definition 2.5.1 in the third edition of (Chang & Keisler 1973) has Occurrence, Expansion and Renaming properties as we have them, plus an *Isomorphism* property instead of our Bisimilarity property, and a Relativization property corresponding to our Localization property.

We need to say what we mean by '$(\mathcal{L}, \models_{\mathcal{L}})$ contains basic modal logic.'

6.5.10. Definition. We say that $(\mathcal{L}, \models_{\mathcal{L}})$ *contains basic modal logic* if $\mathcal{L}$ satisfies the following two properties:

(a) *Closure property.* $\mathcal{L}$ is closed under $\wedge$, $\vee$, and $\neg$, and $\models_{\mathcal{L}}$ satisfies the usual rules for satisfaction of $\wedge$, $\vee$, $\neg$.

(b) *Second expansion property.* We are able to expand the language of $\mathcal{L}$ in the usual way with both proposition letters and basic modal operators.

To be precise, for $(\mathfrak{A}, a)$ a model and X a subset of the domain of $\mathfrak{A}$, we can expand the language and our model with a proposition letter p_X to be interpreted as X: $(\mathfrak{A}, a) \models_{\mathcal{L}} p_X$ iff $a \in X$. And if $(\mathfrak{A}, a)$ a model, and R is an $(n+1)$-ary relation on the domain of $\mathfrak{A}$, then we can expand the language and our model with an n-ary modal operator $\#_R$ whose semantics is based on R: $(\mathfrak{A}, a) \models_{\mathcal{L}} \#_R(\phi_1, \ldots, \phi_n)$ iff there are $a_1, \ldots, a_n$ in $\mathfrak{A}$ with $Raa_1 \ldots a_n$ and $(\mathfrak{A}, a_i) \models_{\mathcal{L}} \phi_i$ $(1 \leq i \leq n)$.

The class of formulas of an abstract modal logic extending basic modal logic is a proper class. By the properties listed, for each basic modal formula ϕ, $(\mathfrak{A}, a) \models_{\mathcal{L}} \phi$ iff $(\mathfrak{A}, a) \models \phi$.

It should be emphasized that logics in the sense of Definition 6.5.8 deal with the same class of pointed models as basic modal logic, and only the formulas and satisfaction relation may be different. This implies, for example, that the earlier nominal tense logic of Blackburn (1993*a*) (cf. §3.2), whose repertoire contains special proposition symbols, is not an abstract modal logic: its models need to satisfy special constraints. The original Lindström characterization of first-order logic suffers from similar limitations (by not allowing ω-logic as a logic, for example).

We need a further definition.

6.5.11. DEFINITION. An abstract modal logic has a *notion of finite rank* if there is a function $\mathrm{rank}_{\mathcal{L}} : \mathcal{L} \to \mathbb{N}$ such that for all $(\mathfrak{A}, a)$, all ϕ in $\mathcal{L}$,

$$(\mathfrak{A}, a) \models_{\mathcal{L}} \phi \quad \text{iff} \quad \Big((\mathfrak{A}, a) \restriction \{\, x \in \mathfrak{A} : \mathrm{depth}(x) \leq \mathrm{rank}(\phi) \,\}\Big), a \models_{\mathcal{L}} \phi.$$

If $\mathcal{L}$ extends basic modal logic, we assume that $\models_{\mathcal{L}}$ behaves regularly with respect to standard modal operators and proposition letters. That is, for $\#$ a modal operator as defined in §6.3, $\mathrm{rank}_{\mathcal{L}}\big(\#(\phi_1, \ldots, \phi_n)\big) = 1 + \max\{\, \mathrm{rank}_{\mathcal{L}}(\phi_i) : 1 \leq i \leq n\}$, and $\mathrm{rank}_{\mathcal{L}}(p) = 0$.

An abstract modal logic has the *finite model property* (FMP) if every ϕ in $\mathcal{L}$ that is satisfiable, has a finite model.

Two models $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ for the same language are *$\mathcal{L}$-equivalent* if for every ϕ in $\mathcal{L}$, $(\mathfrak{A}, a) \models \phi$ iff $(\mathfrak{B}, b) \models \phi$.

Having a notion of finite rank and the FMP will single out basic modal logic among the class of all modal logics, just as the Löwenheim-Skolem property and Compactness single out first-order logic among the class of all classical logics (cf. Theorem 6.5.14 below).

6.5.12. REMARK. For $\mathcal{L} = \mathcal{BML}(\tau)$, a notion of finite rank is given by Definition 6.4.2.

Observe that having a finite rank is a very restrictive property, which is neither implied by, nor does it imply the FMP. To see this recall that **PDL** has the FMP: it has the property that every satisfiable formula ϕ is satisfiable on a model of size at most $|\phi|^3$, where ϕ is the length of ϕ, cf. (Goldblatt 1987). However, it does not have a notion of finite rank. To see this, let (W, R_a) be the binary tree of all strings over $\{\, 0, 1 \,\}$, with $R_a xy$ iff $y = x * 1$ or $y = x * 0$, and with ϵ the empty string. Let

$$\phi = p \wedge \langle a \rangle \neg p \wedge [a] \neg p \wedge [a^*]\Big([a]\neg p \to \langle a^* \rangle [a] p\Big) \wedge [a^*]\Big([a] p \to \langle a^* \rangle [a] \neg p\Big).$$

Define a valuation V by putting x in $V(p)$ iff the length of x is even. Then for no finite n does the restriction of $((W, R, V), \epsilon)$ to depth n satisfy ϕ.

To see that, conversely, the existence of a notion of rank does not imply the FMP, consider the fragment of $\mathcal{BML}_{\omega_\infty \omega}(\tau)$, for τ containing a binary R and infinitely many unary predicate letters, consisting of all formulas of finite rank. Clearly this fragment contains the formula

$$\bigwedge_{i \in \omega} \Diamond(p_i \wedge \bigwedge_{i \neq j} \neg p_j).$$

But clearly, this formula does not have a finite model.

6.5.13. LEMMA. *Let $(\mathcal{L}, \models_{\mathcal{L}})$ be an abstract modal logic which extends basic modal logic. Assume $\mathcal{L}$ has a notion of finite rank* $\mathrm{rank}_{\mathcal{L}}$ *and the FMP. Let ϕ be an $\mathcal{L}$-formula with* $\mathrm{rank}_{\mathcal{L}}(\phi) = n$. *For any two models $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ with $(\mathfrak{A}, a) \equiv^n_{B\mathcal{ML}} (\mathfrak{B}, b)$, $(\mathfrak{A}, a) \models \phi$ implies $(\mathfrak{B}, b) \models \phi$.*

Proof. Assume that the conclusion of the Lemma does not hold. Let $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ be such that $(\mathfrak{A}, a) \equiv^n_{B\mathcal{ML}} (\mathfrak{B}, b)$ but for some $\mathcal{L}$-formula ϕ, $(\mathfrak{A}, a) \models \phi$, $(\mathfrak{B}, b) \not\models \phi$.

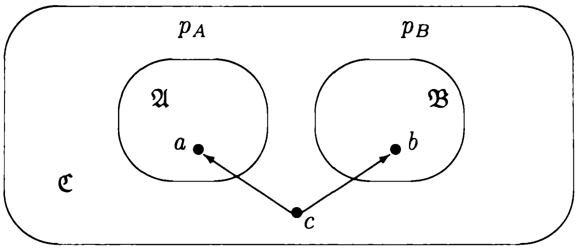


Figure 6.2: Combining models.

By taking reducts we may assume that $\mathcal{L} = \mathcal{L}(\tau_\phi)$. Form the model $\mathfrak{C}$ as in Figure 6.2, that is: take the disjoint union of $\mathfrak{A}$ and $\mathfrak{B}$, add a new element c to be used as the distinguished element of $\mathfrak{C}$, and relate this new point to a via the new relation R_A, and to b via the new relation R_B. As $\mathcal{L}$ extends basic modal logic, there are

- formulas p_A and p_B denoting the domain of $\mathfrak{A}$ and the domain of $\mathfrak{B}$, respectively, and
- two unary modal operators $\#_A$ and $\#_B$ interpreted using two binary relations R_A and R_B, respectively.

Let $n = \mathrm{rank}_{\mathcal{L}}(\phi)$. Over a finite vocabulary there are only finitely many basic modal formulas with a fixed finite rank (modulo equivalence, of course). Let $\Gamma_n(a)$ be the finite set of basic modal formulas of rank n that are satisfied at a; define $\Gamma_n(b)$ similarly. We may assume $\Gamma_n(a) = \Gamma_n(b)$.

Let χ be the conjunction of the following formulas:

- $\neg p_A \land \neg p_B$,
- $\#_A\Big(\big(\phi \downarrow (p_A \land \neg p_B)\big) \land \big(\bigwedge \Gamma_n(a) \downarrow (p_A \land \neg p_B)\big)\Big)$,
- $\#_B\Big(\big(\neg\phi \downarrow (\neg p_A \land p_B)\big) \land \big(\bigwedge \Gamma_n(b) \downarrow (\neg p_A \land p_B)\big)\Big)$.

Clearly $(\mathfrak{C}, c) \models \chi$. By the assumptions on $\mathcal{L}$, χ has a finite model $(\mathfrak{C}', c')$ of depth at most $\mathrm{rank}(\chi) = 1 + \mathrm{rank}(\phi) = 1 + n$. $(\mathfrak{C}', c')$ contains two disjoint substructures $(\mathfrak{A}', a')$ and $(\mathfrak{B}', b')$ with $(\mathfrak{A}', a') \models \phi$, $(\mathfrak{B}', b') \models \neg\phi$, and

$$(\mathfrak{A}', a') \equiv^n_{B\mathcal{ML}(\tau_\phi)} (\mathfrak{B}', b'). \tag{6.1}$$

By the Bisimilarity property (iv) and Proposition 6.3.5 we may assume that all elements in $\mathfrak{A}$ and $\mathfrak{B}$ have in-degree at most 1. As the depth of $(\mathfrak{C}', c')$ is at most $n+1$, it follows that both $(\mathfrak{A}', a')$ and $(\mathfrak{B}', b')$ have depth at most n. Then, because of (6.1) and Proposition 6.4.4, $(\mathfrak{A}', a') \leftrightarrows^b_{\tau_\phi} (\mathfrak{B}', b')$. Hence, by the Bisimulation property, $(\mathfrak{A}', a') \models \phi$ implies $(\mathfrak{B}', b') \models \phi$ — a contradiction. ⊣

6.5.14. THEOREM. (Characterization of Basic Modal Logic) *Let $\mathcal{L}$ extend basic modal logic. If $\mathcal{L}$ has a notion of finite rank and the FMP, then $\mathcal{L}$ is equivalent to basic modal logic.*

Proof. We must show that every $\mathcal{L}$-formula is $\mathcal{L}$-equivalent to a basic modal formula ψ, that is, for all $(\mathfrak{A}, a)$, $(\mathfrak{A}, a) \models_{\mathcal{L}} \phi$ iff $(\mathfrak{A}, a) \models_{\mathcal{L}} \psi$. As before, we restrict ourselves to a finite language. Moreover, ϕ has a basic modal equivalent iff it has such an equivalent with the same rank; so we have to locate the equivalent we are after among the basic modal formulas whose rank equals n, the $\mathcal{L}$-rank of ϕ. Again as before, there are only finitely many (non-equivalent) basic modal formulas whose rank equals n; assume that they are all contained in Γ_n. It suffices to show the following

$$\text{if } (\mathfrak{A}, a), (\mathfrak{B}, b) \text{ agree on all formulas in } \Gamma_n, \text{ then they agree on } \phi. \qquad (6.2)$$

For then, ϕ will be equivalent to a Boolean combination of formulas in Γ_n. But (6.2) is precisely the content of Lemma 6.5.13. ⊣

6.5.15. EXAMPLES AND NON-EXAMPLES. The language of basic temporal logic has operators F, with $x \models F\phi$ iff for some y, both Rxy and $y \models \phi$, and P, with $x \models P\phi$ iff for some y, both Ryx and $y \models \phi$. The pattern for F is just a basic modal pattern in the sense of Definition 6.2.1, but the one for P isn't. As this language 'looks back and forth' along the relation R it violates the Bisimilarity property, hence it is not a basic modal language.

Van der Hoek & De Rijke (1992, 1993) study a *graded modal language* with modal operators $\Diamond_n$, for $n \in \mathbb{N}$, over a vocabulary containing just a single binary R beside the usual unary predicates:

$$x \models \Diamond_n p \text{ iff } |\{ y : (x, y) \in R \wedge y \in p \}| > n.$$

This language is not a basic modal language as it does not enjoy the Bisimilarity property: it is not just only sensitive to the existence of R-successors (as basic τ-bisimulations are), but it is also sensitive to the number of R-successors.

The earlier nominal tense logic is not a basic modal logic as its models are not the same as the models of basic modal logic.

In its usual formulation, with *, $\cup$, ; and ?, **PDL** is not a basic modal logic. It enjoys the Bisimilarity property, and it has the finite model property, but, by Remark 6.5.12 it lacks a notion of finite rank. Leaving out the Kleene star from the relational repertoire results in a basic modal system, as this fragment does have a notion a finite rank; define a mapping $(\cdot)^\bullet$ from *-free **PDL** into basic modal logic by recursively replacing $\langle \alpha; \beta \rangle$ by $\langle \alpha \rangle \langle \beta \rangle$, $\langle \alpha \cup \beta \rangle \phi$ by $\langle \alpha \rangle \phi \vee \langle \beta \rangle \phi$, and $\langle \phi ? \rangle \psi$ by $\phi \wedge \psi$; define the rank of a *-free **PDL**-formula to be the (basic modal) rank of its $^\bullet$-translation. Hence, the *-free fragment of **PDL** is a basic modal logic. Subsequently adding $\cap$ as an operator on relations destroys the latter property by violating the Bisimilarity condition.

6.6 Preservation

Preservation results formed the backbone of model theory for first-order logic until the early sixties. More recently there has been a renewed interest in preservation results with the growing importance of restricted fragments and restricted model classes. The best known examples of preservation results in first-order logic include

- Łoś's Theorem: A first-order formula is preserved under submodels iff it is equivalent to a universal first-order formula (Chang & Keisler 1973, Theorem 3.2.2).
- The Chang-Łoś-Suszko Theorem: A first-order formula is preserved under unions of chains iff it is equivalent to a 'universal-existential' first-order formula (Chang & Keisler 1973, Theorem 3.2.3).
- Lyndon's Theorem: A first-order formula is preserved under homomorphisms iff it is equivalent to a positive first-order formula (Chang & Keisler 1973, Theorem 3.2.4).

To further substantiate our main claim that bisimulations form the basic tools for the model theory of modal logic, we will prove modal versions of each of the above preservation results.

Submodels

6.6.1. Definition. A formula in $\mathcal{BML}(\tau)$ is *existential* if it has been built using (negated) proposition letters, $\vee$, $\wedge$, $\perp$, $\top$ and modal operators $\#$ only. A formula in $\mathcal{BML}(\tau)$ is *universal* if it has been built using (negated) proposition letters, $\vee$, $\wedge$, $\perp$, $\top$ and duals $\overline{\#}$ of modal operators $\#$ in $\mathcal{BML}(\tau)$ only.

6.6.2. Definition. Let $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ be two models for the same vocabulary. $(\mathfrak{A}, a)$ is a *submodel* of $(\mathfrak{B}, b)$ if $a = b$, and for every R, $R^{\mathfrak{A}}$ is the restriction of $R^{\mathfrak{B}}$ to the (appropriate) domain(s) of $\mathfrak{A}$. A basic modal formula is *preserved under submodels* if $(\mathfrak{B}, b) \models \phi$ implies $(\mathfrak{A}, a) \models \phi$ whenever $(\mathfrak{A}, a)$ is a submodel of $(\mathfrak{B}, b)$.

To prove a basic modal version of Łoś's Theorem we need a technical lemma. The following notation will be useful in stating it. For Σ a set of $\mathcal{BML}$-formulas, $(\mathfrak{A}, a) \Rrightarrow_{\Sigma} (\mathfrak{B}, b)$ abbreviates: for all $\phi \in \Sigma$, $(\mathfrak{A}, a) \models \phi$ implies $(\mathfrak{B}, b) \models \phi$; in particular we will use $\Rrightarrow_{\mathrm{E}}$, where 'E' denotes the set of all existential formulas.

6.6.3. Definition. A τ-structure $(\mathfrak{A}, a)$ is called *smooth* if every element in $(\mathfrak{A}, a)$ has finite depth and in-degree at most 1, and for all R and all R-tuples $(x, x_1, \ldots, x_n)$ we have that all x_i have the same finite depth. By Remark 6.3.6 smoothness is enforceable.

6.6.4. Lemma. *Let $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ be τ-structures such that $(\mathfrak{A}, a)$ is smooth, $(\mathfrak{B}, b)$ is ω-saturated, and $(\mathfrak{A}, a) \Rrightarrow_{\mathrm{E}} (\mathfrak{B}, b)$. Then there exists $(\mathfrak{B}', b') \leftrightarroweq^{b}_{\tau} (\mathfrak{B}, b)$ such that $(\mathfrak{A}, a)$ is embeddable in $(\mathfrak{B}', b')$.*

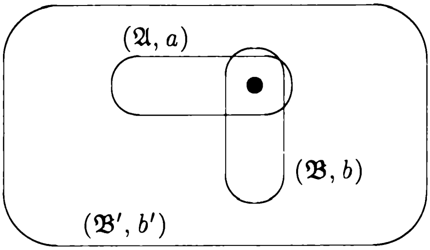


Figure 6.3: Combining $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$.

In a diagram the Lemma claims:

$$\begin{array}{ccc}
(\mathfrak{A}, a) & \Rrightarrow_{\mathrm{E}} & (\mathfrak{B}, b) \\
= \big| & & \big| \underline{\leftrightarrow}^{b}_{\tau} \\
(\mathfrak{A}, a) & \hookrightarrow & (\mathfrak{B}', b').
\end{array}$$

In a somewhat different form, and restricted to the standard modal language, Lemma 6.6.4 is due to Van Benthem (1991*c*).

Proof of Lemma 6.6.4. We define a 'forth simulation' F between $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$, that is: a relation F that links two points only if they agree on all proposition letters, and that satisfies the *forth* condition:

> if Fvw, $R^{\mathfrak{A}}vv_1 \dots v_n$, then there are $w_1, \dots, w_n$ in $\mathfrak{B}$ with $R^{\mathfrak{B}}ww_1 \dots w_n$ and Fv_iw_i $(1 \leq i \leq n)$.

We define a function F from $(\mathfrak{A}, a)$ to $(\mathfrak{B}, b)$ by induction on the depth of elements in $(\mathfrak{A}, a)$. This function will be a forth simulation, and as such it will satisfy $(\mathfrak{A}, x) \Rrightarrow_{\mathrm{E}} (\mathfrak{B}, Fx)$. Put $Fa = b$. Assume that F has been defined for all elements of depth $< n$. let x in $(\mathfrak{A}, a)$ have depth n. By the smoothness of $(\mathfrak{A}, a)$ there are *unique* elements y of depth $n-1$, and $x_1, \dots, x_n$ of depth n such that x is one of $x_1, \dots, x_n$, and such that for some R we have $R^{\mathfrak{A}}yx_1 \dots x_n$. We define F for each of $x_1, \dots, x_n$. Let E_i be the set of existential modal formulas satisfied by x_i. By $(\mathfrak{A}, y) \Rrightarrow_{\mathrm{E}} (\mathfrak{B}, Fy)$ and saturation there are $x_1', \dots, x_n'$ in $\mathfrak{B}$ with $x_i' \models E_i$ and $R^{\mathfrak{B}}F(y)x_1' \dots x_n'$ $(1 \leq i \leq n)$. Put $Fx_i = x_i'$ $(1 \leq i \leq n)$.

The next step is to extend F to a full bisimulation between a supermodel $(\mathfrak{B}', b')$ of $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$. Define $(\mathfrak{B}', b')$ (as in Figure 6.3) to be the disjoint union of $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ *in which we identify the two distinguished points of* $(\mathfrak{A}, a)$ *and* $(\mathfrak{B}, b)$, and with the following extension of the relations:

> if $x \in (\mathfrak{A}, a)$, $Fx = y$ and $Ryv_1 \dots v_n$, then $Rxv_1 \dots v_n$.

Observe that a and b agree on all proposition letters, thus their identification is well-defined. Define a relation Z between the domain of $(\mathfrak{B}', b')$ and the domain of $(\mathfrak{B}, b)$ as follows: for x in $\mathfrak{A}$ we put Zxy whenever $Fx = y$, and for x in $\mathfrak{B}$ we put Zxx. Then $Z : (\mathfrak{B}', b') \underline{\leftrightarrow}^{b}_{\tau} (\mathfrak{B}, b)$:

– Z-related points agree on all proposition letters,

- Assume v in $\mathfrak{B}'$, w in $\mathfrak{B}$ and Zvw. If $R^{\mathfrak{B}'}vv_1 \dots v_k$, then either $v_1, \dots, v_k$ all live in $\mathfrak{A}$, or they all live in $\mathfrak{B}$. In the first case our forth simulation F will find $w_1, \dots, w_k$ with Zv_iw_i $(1 \leq i \leq k)$ and $R^{\mathfrak{B}}ww_1 \dots w_k$. In the second case we have two possibilities: if v in $\mathfrak{B}$, then $v = w$, $R^{\mathfrak{B}}vv_1 \dots v_k$ and Zv_iv_i. The other possibility is that v is not in $\mathfrak{B}$; but then $Fv = w$ and $R^{\mathfrak{B}}wv_1 \dots v_k$, and by construction Zv_iw_i, as required.
- Assume v in $\mathfrak{B}'$, w in $\mathfrak{B}$ and Zvw. Assume also that $R^{\mathfrak{B}}ww_1 \dots w_k$. If v in $\mathfrak{A}$, then by construction $Fv = w$, and $R^{\mathfrak{B}'}vw_1 \dots w_k$ and Zw_iw_i. If v is in $\mathfrak{B}$, then we must have $v = w$, $R^{\mathfrak{B}'}vw_1 \dots w_k$ and Zw_iw_i, and we are done.

Thus $Z : (\mathfrak{B}', b') \leftrightarrows^b_\tau (\mathfrak{B}, b)$. As $(\mathfrak{A}, a)$ lies embedded as a submodel in $(\mathfrak{B}', b')$, this completes the proof. ⊣

6.6.5. THEOREM. (Łoś's Theorem) *A basic modal formula is preserved under submodels iff it is equivalent to a universal basic modal formula.*

Proof. Aside from an application of Lemma 6.6.4 this is a routine argument. First, it is easy to check that if ϕ is equivalent to a universal formula, then it is preserved under submodels.

Second, if ϕ is so preserved, let $\mathrm{CONS}_U(\phi)$ be the set of universal consequences of ϕ. By compactness it suffices to show $\mathrm{CONS}_U(\phi) \models \phi$. So assume $(\mathfrak{A}, a) \models \mathrm{CONS}_U(\phi)$; we may assume that $(\mathfrak{A}, a)$ is smooth. Let E be the set of all existential formulas ψ with $(\mathfrak{A}, a) \models \psi$. Then, by compactness, $E + \phi$ has a model $(\mathfrak{B}, b)$, which may be assumed to be ω-saturated. By Lemma 6.6.4 $(\mathfrak{B}, b) \models E + \phi$ implies that some supermodel $(\mathfrak{B}', b')$ of $(\mathfrak{A}, a)$ has $(\mathfrak{B}', b') \models \phi$. By preservation under submodels $(\mathfrak{A}, a) \models \phi$. ⊣

UNIONS OF CHAINS

6.6.6. DEFINITION. A formula in $\mathcal{BML}(\boldsymbol{\tau})$ is *universal existential* if it has been built using existential formulas, $\wedge$, $\vee$, and dual modal operators $\overline{\#}$ only. A formula is *existential universal* if it has been built using universal formulas, $\wedge$, $\vee$, and modal operators $\#$ only.

We write $(\mathfrak{A}, a) \Rrightarrow_{UE} (\mathfrak{B}, b)$ for: $(\mathfrak{B}, b)$ satisfies all universal existential formulas satisfied by $(\mathfrak{A}, a)$; and similarly for $\Rrightarrow_{EU}$.

6.6.7. DEFINITION. A *chain* of $\boldsymbol{\tau}$-structures is a collection $((\mathfrak{A}_i, a_i) : i \in I)$ such that for all i, j, if $i < j$, then $(\mathfrak{A}_i, a_i)$ is a submodel of $(\mathfrak{A}_j, a_j)$. A *bisimilar chain* is a chain $((\mathfrak{A}_i, a_i) : i \in I)$ in which for all $i \leq j \in I$, $(\mathfrak{A}_i, a_i) \leftrightarrows^b_\tau (\mathfrak{A}_j, a_j)$.

The *union of the chain* $((\mathfrak{A}_i, a_i) : i \in I)$ is the model $\mathfrak{A} = \bigcup_{i \in I}(\mathfrak{A}_i, a_i)$ whose universe is the set $\bigcup_{i \in I}$, and whose relations are the unions of the corresponding relations of $(\mathfrak{A}_i, a_i)$: $R^{\mathfrak{A}} = \bigcup R^{\mathfrak{A}_i}$.

6.6.8. LEMMA. *Let $((\mathfrak{A}_i, a_i) : i \in I)$ be a bisimilar chain of $\boldsymbol{\tau}$-structures. Then, for each j, $(\mathfrak{A}_j, a_j) \leftrightarrows^b_\tau \bigcup_{i \in I}(\mathfrak{A}_i, a_i)$.*

6.6.9. LEMMA. *Assume $(\mathfrak{C}, c)$ is a smooth model that lies embedded as a submodel in $(\mathfrak{D}, d)$. Then there exists $(\mathfrak{E}, e) \leftrightarrows^b_\tau (\mathfrak{D}, d)$ such that $(\mathfrak{C}, c)$ lies embedded as a submodel in $(\mathfrak{E}, e)$ and $(\mathfrak{E}, e)$ is smooth.*

Proof. First, take the submodel of $(\mathfrak{D}, d)$ that is generated by d, and then apply the 'unraveling' construction of Proposition 6.3.5 to the result. As $(\mathfrak{C}, c)$ is smooth neither operation will affect $(\mathfrak{C}, c)$. ⊣

6.6.10. LEMMA. *Let* $(\mathfrak{A}, a)$, $(\mathfrak{B}, a)$, $(\mathfrak{C}, c)$ *be structures such that* $(\mathfrak{A}, a) \preccurlyeq (\mathfrak{B}, a) \underline{\leftrightarrow}^b_\tau (\mathfrak{C}, c)$. *Then* $(\mathfrak{A}, a) \underline{\leftrightarrow}^b_\tau (\mathfrak{C}, c)$.

Proof. It suffices to show $(\mathfrak{A}, a) \underline{\leftrightarrow}^b_\tau (\mathfrak{B}, a)$. To this end use the elementary embedding which is assumed to exist, together with the Tarski-Vaught criterion for elementary substructures. ⊣

6.6.11. LEMMA. *Let* $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ *be* τ*-structures such that* $(\mathfrak{A}, a)$ *is smooth,* $(\mathfrak{B}, b)$ *is* ω*-saturated, and* $(\mathfrak{A}, a) \Rrightarrow_{\mathrm{EU}} (\mathfrak{B}, b)$. *Then there exists a smooth model* $(\mathfrak{B}', b') \underline{\leftrightarrow}^b_\tau (\mathfrak{B}, b)$ *such that* $(\mathfrak{A}, a)$ *is embeddable in* $(\mathfrak{B}', b')$ *and* $(\mathfrak{A}, a) \Rrightarrow_{\mathrm{U}} (\mathfrak{B}', b')$.

In a diagram the Lemma claims:

$$\begin{tikzcd}
(\mathfrak{A}, a) \arrow[r, phantom, "\Rrightarrow_{\mathrm{EU}}"] \arrow[d, dash, "="'] & (\mathfrak{B}, b) \arrow[d, dash, "\underline{\leftrightarrow}^b_\tau"] \\
(\mathfrak{A}, a) \arrow[r, phantom, "\overset{\hookrightarrow}{\Rrightarrow_{\mathrm{U}}}"] & (\mathfrak{B}', b').
\end{tikzcd}$$

Proof of Lemma 6.6.11. This is similar to the proof of Lemma 6.6.4. Define a function F that is a forth simulation from $(\mathfrak{A}, a)$ to $(\mathfrak{B}, b)$ such that $Fa = b$ and $(\mathfrak{A}, x) \Rrightarrow_{\mathrm{EU}} (\mathfrak{B}, Fx)$. Extend F to a full bisimulation between $(\mathfrak{A}, a)$ and a supermodel $(\mathfrak{B}', b') \underline{\leftrightarrow}^b_\tau (\mathfrak{B}, b)$ of $(\mathfrak{A}, a)$ that has $(\mathfrak{A}, a) \hookrightarrow (\mathfrak{B}', b')$, as in the proof of 6.6.4. By Lemma 6.6.9 we may take $(\mathfrak{B}', b')$ to be smooth. To complete the proof we need to show that $(\mathfrak{A}, a) \Rrightarrow_{\mathrm{U}} (\mathfrak{B}', b')$. This is almost trivial: for a universal formula ϕ we have that $(\mathfrak{A}, a) \models \phi$ implies $(\mathfrak{B}, b) \models \phi$, as $(\mathfrak{A}, a) \Rrightarrow_{\mathrm{UE}} (\mathfrak{B}, b)$. Since $(\mathfrak{B}', b') \underline{\leftrightarrow}^b_\tau (\mathfrak{B}, b)$, this implies $(\mathfrak{B}', b') \models \phi$. ⊣

6.6.12. THEOREM. (Chang-Łoś-Suszko Theorem) *A basic modal formula is preserved under unions of chains iff it is equivalent to a universal existential formula.*

Proof. Again, the argument is (bisimilar to) the standard argument proving the result for first-order logic. We only prove the hard direction. Assume ϕ is preserved under unions of chains. Let $\mathrm{CONS}_{UE}(\phi)$ denote the set of universal existential consequences of ϕ. It suffices to prove that $\mathrm{CONS}_{UE}(\phi) \models \phi$. So assume $(\mathfrak{A}_0, a_0) \models \mathrm{CONS}_{UE}(\phi)$; we may of course assume that $(\mathfrak{A}_0, a_0)$ is smooth. We prove that $(\mathfrak{A}_0, a_0) \models \phi$. To this end we construct a bisimilar chain $((\mathfrak{A}_i, a_i) : i < \omega)$ of smooth models, smooth extensions $(\mathfrak{B}_i, b_i) \supseteq (\mathfrak{A}_i, a_i)$, and embeddings $g_i : (\mathfrak{B}_i, b_i) \to (\mathfrak{A}_{i+1}, a_{i+1})$ as in the following diagram:

$$\begin{tikzcd}
 & (\mathfrak{B}_0, b_0) \arrow[dr, "g_0"] & & (\mathfrak{B}_1, b_1) \arrow[dr, "g_1"] & \\
(\mathfrak{A}_0, a_0) \arrow[ur, "\subseteq"] \arrow[rr, phantom, "\underline{\leftrightarrow}^b_\tau"] & & (\mathfrak{A}_1, a_1) \arrow[ur, "\subseteq"] \arrow[rr, phantom, "\underline{\leftrightarrow}^b_\tau"] & & (\mathfrak{A}_2, a_2)
\end{tikzcd} \tag{6.3}$$

We will require that for each $i < \omega$:

$$(\mathfrak{B}_i, b_i) \models \phi \text{ and } (\mathfrak{B}_i, b_i) \Rrightarrow_{\mathrm{E}} (\mathfrak{A}_i, a_i). \tag{6.4}$$

The diagram is constructed as follows. Suppose $(\mathfrak{A}_i, a_i)$ has been defined. As $(\mathfrak{A}_0, a_0) \underline{\leftrightarrow}^b_\tau (\mathfrak{A}_i, a_i)$ we have $(\mathfrak{A}_i, a_i) \models \mathrm{CONS}_{UE}(\phi)$. Take any ω-saturated extension of $(\mathfrak{A}_i, a_i)$; by Lemma 6.6.11 $(\mathfrak{A}_i, a_i)$ can be extended to a smooth structure $(\mathfrak{B}_i, b_i)$ satisfying (6.4). Take an ω-saturated elementary extension $(\mathfrak{C}, a_i)$ of $(\mathfrak{A}_i, a_i)$. By the proof of Lemma 6.6.4 and $(\mathfrak{B}_i, b_i) \Rrightarrow_{\mathrm{E}} (\mathfrak{C}, a_i)$ there is a model $(\mathfrak{A}_{i+1}, a_{i+1}) \underline{\leftrightarrow}^b_\tau (\mathfrak{C}, a_i)$ and an embedding $g_i : (\mathfrak{B}_i, b_i) \hookrightarrow (\mathfrak{A}_{i+1}, a_{i+1})$ such that g_i is the identity on $\mathfrak{C}$. By Lemma 6.6.9 $(\mathfrak{A}_{i+1}, a_{i+1})$ may be taken to be smooth. So all we have to do to complete the construction is show that $(\mathfrak{A}_i, a_i) \underline{\leftrightarrow}^b_\tau (\mathfrak{A}_{i+1}, a_{i+1})$ — but this is Lemma 6.6.10.

In the diagram (6.3) we can replace each $(\mathfrak{B}_i, b_i)$ by its image under g_i, and so assume that the maps are inclusions. Then $\bigcup_{i<\omega}(\mathfrak{A}_i, a_i)$ and $\bigcup_{i<\omega}(\mathfrak{B}_i, b_i)$ are the same structure $(\mathfrak{C}, c)$. As ϕ is preserved under unions of chains, $(\mathfrak{B}_i, b_i) \models \phi$ (for all i) implies $(\mathfrak{C}, c) \models \phi$. By Lemma 6.6.8 $(\mathfrak{A}_0, a_0) \underline{\leftrightarrow}^b_\tau (\mathfrak{C}, c)$, hence $(\mathfrak{A}_0, a_0) \models \phi$. $\dashv$

HOMOMORPHISMS

6.6.13. DEFINITION. A formula ϕ in $\mathcal{BML}(\tau)$ is *positive* iff it has been built up using only $\bot$, $\top$, proposition letters, $\wedge$, $\vee$, as well as modal operators $\#$ and their duals $\overline{\#}$. A formula ϕ is *negative* iff it has been built up from $\bot$, $\top$, negated proposition letters, $\wedge$, $\vee$, as well as modal operators $\#$ and their duals $\overline{\#}$.

6.6.14. DEFINITION. Let $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ be two τ-structures. A *homomorphism* $f : (\mathfrak{A}, a) \to (\mathfrak{B}, b)$ is a mapping with $f(a) = b$, that preserves all relations and proposition letters. A basic modal formula ϕ is *preserved under surjective homomorphisms* if $(\mathfrak{A}, a) \models \phi$ implies $(\mathfrak{B}, b) \models \phi$ whenever $(\mathfrak{B}, b)$ is a homomorphic image of $(\mathfrak{A}, a)$.

Some more notation: $(\mathfrak{A}, a) \Rrightarrow_{\mathrm{P}} (\mathfrak{B}, b)$ is short for: for all positive formulas ψ, $(\mathfrak{A}, a) \models \psi$ implies $(\mathfrak{B}, b) \models \psi$.

6.6.15. LEMMA. *Let $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ be ω-saturated τ-structures with $(\mathfrak{A}, a) \Rrightarrow_{\mathrm{P}} (\mathfrak{B}, b)$, and such that both in $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ all elements have in-degree at most 1. Then there exist τ-structures $(\mathfrak{A}', a') \underline{\leftrightarrow}^b_\tau (\mathfrak{A}, a)$ and $(\mathfrak{B}', b') \underline{\leftrightarrow}^b_\tau (\mathfrak{B}, b)$ with a surjective homomorphism $f : (\mathfrak{A}', a') \to (\mathfrak{B}', b')$.*

In a diagram the Lemma asserts the existence of the following configuration:

$$\begin{array}{ccc}
(\mathfrak{A}, a) & \Rrightarrow_{\mathrm{P}} & (\mathfrak{B}, b) \\
\underline{\leftrightarrow}^b_\tau \Big| & & \Big| \underline{\leftrightarrow}^b_\tau \\
(\mathfrak{A}', a') & \overset{f}{\twoheadrightarrow} & (\mathfrak{B}', b').
\end{array}$$

Proof of Lemma 6.6.15. The strategy of the proof is to move to smooth models where we can inductively define a surjective homomorphism from a model bisimilar to $(\mathfrak{A}, a)$ onto a model bisimilar to $(\mathfrak{B}, b)$. To ensure surjectivity we have to blow up the model bisimilar to $(\mathfrak{A}, a)$.

Let $(\mathfrak{A}'', a)$ be the submodel of $(\mathfrak{A}, a)$ generated by a, and let $(\mathfrak{B}', b)$ be the submodel of $(\mathfrak{B}, b)$ generated by b. Then both $(\mathfrak{A}'', a)$ and $(\mathfrak{B}', b)$ are smooth. By induction on the depth of elements we will add $|\mathfrak{B}'|^+$ many copies of all (tuples of) elements in $(\mathfrak{A}'', a)$. We show how to do this by adding copies of elements of depth 1 in $(\mathfrak{A}'', a)$ to obtain a model $(\mathfrak{A}_1, a) \leftrightarrows^b_\tau (\mathfrak{A}'', a)$.

Define $\sim$ on the elements of depth 1 in $(\mathfrak{A}'', a)$ by putting $x \sim y$ iff for some R and $x_1, \ldots, x_n$ we have that both x and y are among $x_1, \ldots, x_n$ and $R^{\mathfrak{A}''} a x_1 \ldots x_n$. By smoothness this is well defined. For each $\sim$-equivalence class $X = \{ x_1 \ldots x_n \}$ let $\mathfrak{C}_X$ be the submodel of $(\mathfrak{A}'', a)$ that is generated by X. Now, for each $\mathfrak{C}_X$ take $|\mathfrak{B}'|^+$ many disjoint copies of $\mathfrak{C}_X$, and add them to $(\mathfrak{A}'', a)$; for each copy $\mathfrak{C}'_{X'}$ of $\mathfrak{C}_X$ relate the generating points $x'_1, \ldots, x'_n$ to a the way the originals $x_1, \ldots, x_n$ are related to a. Let $(\mathfrak{A}_1, a)$ be the resulting model. Then $(\mathfrak{A}'', a) \leftrightarrows^b_\tau (\mathfrak{A}_1, a)$. Repeat this construction for all depths n to obtain models

$$(\mathfrak{A}, a) \leftrightarrows^b_\tau (\mathfrak{A}'', a) \underset{\leftrightarrows^b_\tau}{\subseteq} (\mathfrak{A}_1, a) \underset{\leftrightarrows^b_\tau}{\subseteq} (\mathfrak{A}_2, a) \cdots$$

Define $(\mathfrak{A}', a) = \bigcup(\mathfrak{A}_i, a)$. Then $(\mathfrak{A}', a) \leftrightarrows^b_\tau (\mathfrak{A}, a)$, and $(\mathfrak{A}', a)$ has at least $|\mathfrak{B}'|^+$ many copies of each of its submodels generated by tuples $x_1, \ldots, x_n$ such that $R^{\mathfrak{A}'} x x_1 \ldots x_n$ for some R and x.

Next we define a function F from $(\mathfrak{A}', a)$ to $(\mathfrak{B}', b)$ by induction on the depth of elements in such a way that $(\mathfrak{A}', x) \Rrightarrow_{\mathrm{P}} (\mathfrak{A}, Fx)$. For each n we first make sure that all elements of depth n in $(\mathfrak{B}', b)$ are in the range of F. After that we give F values to points of depth n in $(\mathfrak{A}', a)$ that are not yet in the domain of F.

Here we go. Put $Fa = b$. Assume that $n > 0$, and that F has been defined for all depths less than n in such a way that all elements of $(\mathfrak{B}', b)$ of depth less than n are already in the range of F. Let y in $(\mathfrak{B}', b)$ have depth n, and choose z of depth $n-1$, $y_1, \ldots, y_n$ of depth n and R such that $R^{\mathfrak{B}'} z y_1 \ldots y_n$ and y is one of the y_i $(1 \leq i \leq n)$. Let N_i be the set of all negative modal formulas satisfied by y_i in $(\mathfrak{B}', b)$. Then $(\mathfrak{B}, y_i) \models N_i$. By assumption there exists x' in $\mathfrak{A}'$ with $Fx' = z$, and $(\mathfrak{A}', x') \Rrightarrow_{\mathrm{P}} (\mathfrak{B}', z)$. Let x in $\mathfrak{A}''$ be such that x' is a copy of x if x' is in $\mathfrak{A}' \setminus \mathfrak{A}''$, and $x = x'$ otherwise. Then $(\mathfrak{A}', x') \leftrightarrows^b_\tau (\mathfrak{A}'', x) \leftrightarrows^b_\tau (\mathfrak{A}, x)$. Hence, $(\mathfrak{A}, x) \Rrightarrow_{\mathrm{P}} (\mathfrak{B}, z)$. By a saturation argument there are $x_1, \ldots, x_n$ in $\mathfrak{A}$ with $R^{\mathfrak{A}} x x_1 \ldots x_n$ and $x_i \models N_i$ $(1 \leq i \leq n)$. Then $x_1, \ldots, x_n$ are in $\mathfrak{A}''$. Now let $x'_1, \ldots, x'_n$ be copies of $x_1, \ldots, x_n$ such that $R^{\mathfrak{A}'} x' x'_1 \ldots x'_n$ and such that $x'_1, \ldots, x'_n$ are not yet in the domain of F (this is possible as we have added $|\mathfrak{B}'|^+$ many copies to $\mathfrak{A}''$), and put $Fx'_i = x_i$ $(1 \leq i \leq n)$.

Once we have included all elements of depth n in $(\mathfrak{B}', b)$ in the range of F, we define what F should do with elements of depth n in $(\mathfrak{A}', a)$ by using a saturation argument as before, but this time using sets P_i of positive modal formulas, rather than sets N_i of negative modal formulas.

Obviously, the function F thus defined is a homomorphism and a surjection. Hence we are done. $\dashv$

6.6.16. THEOREM. (Lyndon's Theorem) *A basic modal formula is preserved under surjective homomorphisms iff it is equivalent to a positive modal formula.*

Proof. We only prove the hard direction: assume ϕ is preserved under surjective homomorphisms. Let $\mathrm{CONS}_P(\phi)$ be the set of positive formulas ψ with $\phi \models \psi$. It suffices to show that $\mathrm{CONS}_P(\phi) \models \phi$. Assume $(\mathfrak{B}, b) \models \mathrm{CONS}_P(\phi)$. Let N be the set of all negative formulas true at b in $\mathfrak{B}$. Let $(\mathfrak{A}, a) \models N + \phi$. Then $(\mathfrak{A}, a) \Rrightarrow_{\mathrm{P}} (\mathfrak{B}, b)$. We may of course assume that both $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ are ω-saturated, and that all elements in $(\mathfrak{A}, a)$, $(\mathfrak{B}, b)$ have in-degree at most 1.

By Lemma 6.6.15 there are $(\mathfrak{A}', a') \leftrightarrows^b_\tau (\mathfrak{A}, a)$ and $(\mathfrak{B}', b') \leftrightarrows^b_\tau (\mathfrak{B}, b)$, as well as a homomorphism $f : (\mathfrak{A}', a') \twoheadrightarrow (\mathfrak{B}', b')$. Now, $(\mathfrak{A}, a) \models \phi$ implies $(\mathfrak{A}', a') \models \phi$; by preservation under surjective homomorphisms this implies $(\mathfrak{B}', b) \models \phi$, which gives $(\mathfrak{B}, b) \models \phi$, as required. $\dashv$

6.7 Beyond the basic pattern

There is no hope of lifting all of the results of §§6.4–6.6 to arbitrary modal languages beyond the basic modal format of §6.2 — it is known, for instance, that several extended modal languages lack interpolation. We will try and port some of the results of §§6.4, 6.5 to richer languages by way of examples. While doing so it is a good idea to keep the following in mind: when given a modal language and asked to provide it with an appropriate notion of bisimulation, a safe strategy is to isolate the relational patterns the modal language is concerned with, and stipulate that those are to be used in the back-and-forth conditions of the candidate bisimulation.

First, here is a general question. Let $A = \{ \alpha_i : i \in I \}$ be a set of first-order formulas over a vocabulary τ such that each α_i has at least two free variables. The set A describes all the patterns of the *basic modal language* over A, which has modal operators $\#_\alpha$ for α in A, where $\#_\alpha$ is n-ary whenever α has $n + 1$ free variables, and $(\mathfrak{A}, a) \models \#_\alpha(\phi_1, \ldots, \phi_n)$ iff

$$\text{there exist } a_1, \ldots, a_n \text{ with } \mathfrak{A} \models \alpha[aa_1 \ldots a_n] \text{ and } (\mathfrak{A}, a_i) \models \phi_i.$$

Then, a *basic A-bisimulation* is a binary relation Z between two models $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ that links the distinguished points of $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$, that links two points only if they agree on all unary predicates, and such that

1. if Zwv and $\mathfrak{A} \models \alpha[ww_1 \ldots w_n]$, then there exist $v_1, \ldots, v_n$ in $\mathfrak{B}$ such that $\mathfrak{B} \models [vv_1 \ldots v_n]$ and Zw_iv_i, for $1 \leq i \leq n$ and $\alpha(x, x_1 \ldots, x_n)$ in A,
2. a similar back condition.

6.7.1. QUESTION. Let τ be a classical vocabulary, and let $A = \{ \alpha_i : i \in I \}$ be a set of first-order formulas over τ. Which conditions does A have to satisfy to allow for analogues of the main results of §§6.4, 6.5, for basic A-bisimulations and the basic modal language over A?

We will now list a number of examples of sets of patterns A for which Question 6.7.1 has a (partial) positive answer.

6.7.2. EXAMPLE. (D, A and the like) Recall from Chapter 2 that the semantics of the D-operator is given by $x \models D\phi$ iff for some $y \neq x$, $y \models \phi$. The notion of bisimulation appropriate for this operator has the following back-and-forth conditions (in addition to the usual conditions on distinguished points, and unary predicates/proposition letters, of course):

1. if Zxy and $x \neq x'$ then for some y' both $y \neq y'$ and $Zx'y'$,
2. if Zxy and $y \neq y'$ then for some x' both $x \neq x'$ and $Zx'y'$.

By repeating the proofs of 6.4.9, 6.5.3 and 6.5.7 we can establish a Bisimulation, Definability and Fragment Theorem for the language with the D-operator. (Cf. §3.4 for further examples.)

Using the same approach a notion of bisimilarity for the universal modality A (whose semantics reads: $x \models A\phi$ iff all y have $y \models \phi$) has all occurrences of $\neq$ in the above clauses 1 and 2 replaced with ∇, the universal relation. (Equivalently, one can require that a binary relation is to be surjective and total in order to count as a bisimulation for the universal modality.) Repeating the relevant proofs from §§6.4, 6.5 one finds a Bisimulation, Definability and Fragment Theorem for the modal language with the universal modality.

Recall that *nominals* are special proposition letters whose interpretations are restricted to being at most a singleton (in some papers exactly a singleton). Given this restriction on our models the Bisimulation, Definability and Fragment Theorems go through immediately when we add to our notion of bisimulation the requirement that bisimilar states should agree on all nominals.

6.7.3. EXAMPLE. (Temporal and versatile languages) Certain patterns involving operations on relations can also be handled. Consider the language of temporal logic whose operators are F and P. A notion of bisimulation appropriate for the language with F and P, adds to the clauses of Definition 6.3.1 (for τ a vocabulary with just a single binary R, plus the usual unary predicates) the following backward looking back-and-forth conditions:

1. if Zxy and $R^{\mathfrak{A}}x'z$, then for some y', $Zx'y'$ and $R^{\mathfrak{B}}y'y$,
2. if Zxy and $R^{\mathfrak{B}}y'y$, then for some x', $Zx'y'$ and $R^{\mathfrak{A}}x'x$.

Given this extended notion of bisimulation a Bisimulation, Definability and Fragment Theorem for temporal logic are easily obtained by copying the relevant proofs from §§6.4, 6.5. A similar analysis equips all so-called versatile modal languages as defined in Chapter 5 (Definition 5.5.18) with appropriate notions of bisimulation and appropriate analogues of the main results of §§6.4, 6.5.

6.7.4. EXAMPLE. (Dynamic modal logic) Operations on binary relations that are more complex than taking converse are also allowed, as is witnessed by the dynamic modal logic from Chapter 4. Recall that the formulas of the dynamic modal language $\mathcal{DML}$ translate into a fragment $\mathcal{L}_3(x_1)$ of first-order logic (cf. Theorem 4.4.9). This fragments contains all formulas in one free variable x_1 over a binary relation symbol R and $=$ in which at most three variables x_1, x_2, x_3 occur. The set of relevant binary relations in $\mathcal{DML}$ coincides with the fragment $\mathcal{L}_3(x_1, x_2)$, in which all first-order formulas live that have two free variables x_1, x_2, and in which at most three variables occur.

Now, to obtain an analogue for $\mathcal{DML}$ of the results from §§6.4, 6.5, let A simply be the fragment $\mathcal{L}_3(x_1, x_2)$, and repeat the proofs for the Bisimulation, Definability and Fragment Theorem from §§6.4, 6.5.

As was noted in Chapter 4, the above analysis is note quite satisfactory as it is too linguistic: it refers to $\mathcal{L}_3(x_1, x_2)$-definable transitions. Example 6.7.5 below deals with the logic $\mathcal{ML}_2$ of Peirce algebras, whose formulas translate into $\mathcal{L}_3(x_1, x_2)$ (Theorem 5.6.3); there a more algebraic analysis is given.

6.7.5. EXAMPLE. (Many-dimensional modal logic, 1) The analysis of the previous examples is not restricted to (bi-) simulations linking *single* points; it applies equally well to relations linking (finite) tuples of points. Below we illustrate this with an example from many-dimensional modal logic. As many-dimensional modal logics may correspond to relatively large fragments of first-order logic, the notion of bisimulations needed for such languages may be quite close to truth relations known from fragments of classical logic, such as k-partial isomorphisms, cf. (Van Benthem 1991a) for more on this point.

We consider the language $CC\delta$ studied by Venema (1991), whose modal operators are a binary $\circ$ (composition), a unary $\otimes$ (converse), and a nullary δ (diagonal). Its intended models are structures $(U \times U, C, R, I)$ where

$$\begin{aligned} C &= \{ ((u,v),(w,x),(y,z)) \in {}^3(U \times U) : u = w \wedge x = y \wedge v = z \}, \\ R &= \{ ((u,v),(x,y)) \in {}^2(U \times U) : u = y \wedge v = x \}, \\ I &= \{ (u,v) \in (U \times U) : u = v \}, \end{aligned}$$

where C interprets $\circ$, R interprets $\otimes$, and I interprets δ. Proposition letters in this language are interpreted as binary relations.

Now, let $(\mathfrak{A}, (a_1, a_2))$, $(\mathfrak{B}, (b_1, b_2))$ be pointed $CC\delta$-models. A relation Z between (pairs in) $\mathfrak{A}$ and $\mathfrak{B}$ is a *$CC\delta$-bisimulation* if (i) it links the distinguished pairs, (ii) it only links pairs that agree on all proposition letters and on I, (iii) it has back-and-forth conditions for C and R:

- if $Z(x_{11}, x_{12})(y_{11}, y_{12})$ and $C(x_{11}, x_{12})(x_{21}, x_{22})(x_{31}, x_{32})$, then there exist pairs (y_{21}, y_{22}) and (y_{31}, y_{32}) in $\mathfrak{B}$ with $C(y_{11}, y_{12})(y_{21}, y_{22})(y_{31}, y_{32})$ and $Z(x_{21}, x_{22})(y_{21}, y_{22})$, $Z(x_{31}, x_{32})(y_{31}, y_{32})$, and conversely.
- if $Z(x_{11}, x_{12})(y_{11}, y_{12})$ and $R(x_{11}, x_{12})(x_{21}, x_{22})$, then there exists a pair (y_{21}, y_{22}) in $\mathfrak{B}$ with $R(y_{11}, y_{12})(y_{21}, y_{22})$, and $Z(x_{21}, x_{22})(y_{21}, y_{22})$.

Repeating the relevant arguments from §§6.4, 6.5 one finds a Bisimulation, Definability and Fragment Theorem for $CC\delta$ and $CC\delta$-bisimulations.

6.7.6. EXAMPLE. (Many-dimensional modal logic, 2) The present approach is by no means restricted to modal languages in which all formulas have the same dimension. It applies equally well to a heterogeneous system like the logic of Peirce algebras. To see this, recall that $\mathcal{ML}_2$ has two sorts of propositions, one ranging over sets of points, one ranging over sets of pairs of points. In addition, $\mathcal{ML}_2$ was shown to be expressively complete for the first-order fragment $\mathcal{L}_3(x_1, x_2)$, which contains formulas in at most 3 variables x_1, x_2, and x_3 with up to two free variables, x_1 or x_1 and x_2 (Theorem 5.6.3).

As is to be expected, the appropriate notion of $\mathcal{ML}_2$-bisimulation links points (pairs) only if they satisfy the same proposition letters (and constants), while it should also have a back-and-forth-conditions corresponding for the composition and converse of relations as in the above case of $CC\delta$ (Example 6.7.5). But it should have slightly more to shrink and extend Z-related items:

- if $Z(x_1, x_2)(y_1, y_2)$, then Zx_1y_1 and Zx_2y_2,
- if Zx_1y_1 and x_2 is any element, then for some y_2, $Z(x_1, x_2)(y_1, y_2)$, and similarly in the opposite directions.

A routine check of the relevant arguments in §§6.4, 6.5 establishes Bisimulation, Definability and Fragments Theorems for $\mathcal{ML}_2$ and $\mathcal{ML}_2$-bisimulations.

To conclude this section I will briefly discuss an example that does not seem amenable for analysis using the methods of this Chapter. All of the above examples 6.7.2–6.7.6 could be dealt with because essentially the modal languages involved only dealt with operators whose patterns have the form 'there exists a related element with P,' and the back-and-forth clauses of our bisimulations simply tried to match such patterns in bisimilar models.

Two temporal operators that have long resisted analysis in terms of bisimulations are *Until* and *Since*, whose truth definitions are given by

$$x \models \mathit{Until}(p, q) \text{ iff } \exists y\, (Rxy \wedge p(y) \rightarrow \forall z\, (Rxz \wedge Rzy \rightarrow q(z))),$$

and similarly for *Since* in the backwards direction. The difficulties are due to the occurrence of an irreducible $\exists\forall$ quantifier pattern in the truth definition. As similar quantifier patterns occur in modal operators in interpretability logic, conditional logic, and modal approaches to dynamic aspects of natural language, it seems worthwhile to extend or adopt the analysis of the present chapter to *Until*, *Since*-logic.

Van Benthem, Van Eijck & Stebletsova (1993) propose a decomposition of *Until* and *Since* in a multi-dimensional modal language, not unlike the dynamic modal language of Chapter 4. In their set-up *Until* and *Since* are rewritten as certain combinations of one- and two-dimensional operators whose definitions have the earlier existential form, as a result of which the $\exists\forall$ pattern may be reduced after all, and the machinery of this Chapter can be applied in pretty much the same way as it was used for $\mathcal{DML}$ and $\mathcal{ML}_2$. Although this approach certainly solves the problem, it is not quite satisfactory, as it uses bisimulations that relate sequences of length 1 or 2 to analyze a modal language whose formulas are evaluated at *single* points only.

6.8 Concluding remarks

This Chapter has developed the model theory of the class of basic modal languages in parallel with the basic model theory of first-order logic, using bisimulations as its key tool. By means of a Bisimulation Theorem, according to which two models are equivalent in basic modal logic iff they have bisimilar ultrapowers, a series of definability and separability results were obtained; in addition, we were able to give a Lindström style characterization of basic modal logic in terms of bisimulations. After that the idea that bisimulations are a fundamental tool in the model theory of modal logic received further support when we proved preservation results for universal, universal existential and positive basic modal formulas that used bisimulations in an essential way. Finally, extensions of the above results to languages beyond the basic modal format were discussed.

Despite its length and the number of results it contains, this Chapter has only covered some rudimentary model theory, and it only did so for basic modal languages and some extensions — an awful lot remains to be done. First, here are two specific questions

1. First, our Fragment Theorem in §6.5 only characterizes (finitary) basic modal languages as a fragments of first-order languages. What about characterizations of infinitary basic modal languages as fragments of the corresponding infinitary classical languages?
2. Likewise: give Lindström style characterizations as well as preservation results for modal languages differing from the finitary basic modal format.

Next, here are more general issues:

3. In a recent manuscript Johan van Benthem characterizes the (first-order) formulas defining operations on relations that preserve bisimilarity. What is the connection between this 'safety result' and the definability and characterization results obtained here?
4. At least superficially there seems to be a connection between bisimulations and Ehrenfeucht style games. What is the precise connection?
5. Throughout this Chapter we have concentrated on *pointed* models with a distinguished element for evaluation. This suggests that the classical languages in which our modal languages live be equipped with a constant to denote the distinguished point. And this, in turn, suggests that one adds an operator like Hans Kamp's NOW to our modal languages, where $x \models \mathrm{NOW}\phi$ iff for a the distinguished point of the model one has $a \models \phi$. In a recent manuscript Johan van Benthem show the basic results and techniques of this Chapter go through in this extended format.
6. Finally, both bisimilarity and modal equivalence cut up the universe of all model into equivalence classes. This raises the following question: when does an equivalence relation on the class of all models come from a modal language?

7

Correspondence Theory for Extended Modal Logic

7.1 Introduction

As has been stressed repeatedly in this dissertation, modal operators record simple, very restricted patterns of relational models through their truth definitions. Such patterns live in classical languages (first-order, second-order, infinitary, ...). Modal correspondence theory studies the relations between modal languages and classical ones. It does so at various levels, depending on the way modal formulas are interpreted. When interpreted on *models* the modal language $\mathcal{ML}(\diamond)$, for instance, ends up as a very restricted fragment of a first-order language. When $\mathcal{ML}(\diamond)$ is interpreted on *frames* its propositional variables are universally quantified over, and it ends up as a set of Π^1_1-conditions. In this approach a key issue is: when does a modal Π^1_1-condition reduce to a first-order formula? An important tool here is the Sahlqvist-van Benthem algorithm, which when input a modal formula in $\mathcal{ML}(\diamond)$ of a certain form, reduces it to an equivalent *first-order* property of binary relations via suitable instantiations. Recently this algorithm has been extended by Gabbay & Ohlbach (1992) and Simmons (1992) through the use of Skolem functions.

This Chapter is concerned with reducibility issues of the kind described above. The Chapter analyzes and extends the Sahlqvist-van Benthem and Gabbay-Ohlbach-Simmons algorithms in a very general setting; this is done for various reasons. First, a better understanding of the ins and outs of the algorithms is gained if the analysis is independent of any particular modal calculus. Second, recent years have witnessed a boom in extensions and alterations of the standard modal format; as was noted in Chapter 2, only little is known in the way of general results on transfer or applicability of facts and constructions from standard modal logic to extended ones. A fully general analysis of the above correspondence algorithms reveals their applicability to arbitrary modal logics, and beyond, as will be illustrated in §7.6 below with examples from a variety of modal and temporal logics, dynamic logic, circumscription and other areas. Third, it's an important tradition in logic to compare different theories and languages; the work reported on below is part of that line of research.

The next section supplies the main preliminaries; it may be skipped on a first reading. §7.3 defines the central notion of the paper: *correspondence* or *reducibility*; roughly speaking, a formula is reducible for certain variables if it is equivalent to a formula in which those variables don't occur. For most practical purposes actual reductions are obtained by making appropriate substitutions for the forbidden variables. This approach underlies §§7.4, 7.5, where we analyze what makes the Sahlqvist-van Benthem and Gabbay-Ohlbach-Simmons algorithms work; the analysis involves both a semantic description of the substitution mechanisms, and a syntactic characterization of the formulas allowing such substitutions. A less algorithmic perspective is adopted in §7.7; there we obtain reducibility results by imposing restrictions on languages and their interpretations. §7.8 concludes the Chapter with comments and questions.

Before taking off: a frequent complaint about the actual use of the Sahlqvist-van Benthem algorithm has been its alleged obscurity (Kracht 1993, page 194), (Gabbay & Ohlbach 1992, Section 4.3). To address these complaints we pay special attention to using the algorithms in §§7.4–7.6 below.

7.2 Preliminaries

First we need to be specific about classical logic. For τ a classical vocabulary, a *(classical) logic* is given by two classes $\mathrm{Form}_{\mathcal{L}}[\tau]$ and $\mathrm{Sent}_{\mathcal{L}}[\tau]$ of $\mathcal{L}$-formulas and $\mathcal{L}$-sentences respectively, together with a relation $\models_{\mathcal{L}}$ between structures and $\mathcal{L}$-sentences. $\mathrm{Str}[\tau]$ denotes the class of τ-structures. We assume that for any classical logic $\mathcal{L}$, $\mathrm{Form}_{\mathcal{L}}[\tau]$ contains n-placed predicates $\bot_s$ and $\top_s$ ($n \in \mathbb{N}$, s a sort in τ) such that in any model $\mathfrak{A}$, $\bot_s$ is interpreted as the empty set and $\top_s$ as the domain of sort s. Basic model-theoretic notions are introduced as usual (cf. the Appendix for further details).

We assume that we have *membership* or *acceptance* predicates ϵ available, which take as their arguments an n-placed symbol of a 'relational' sort and n terms of the appropriate sorts to form formulas. E.g. if r is a symbol of a binary relational sort, then ϵrxy is a wff; its intended interpretation is that the pair denoted by (x, y) is to belong to the relation denoted by r. Instead of $\epsilon rx_1 \ldots x_n$ we will write $r(x_1, \ldots, x_n)$. Furthermore, equality ($=$) is used only between terms of the individual sort. For a classical logic $\mathcal{L}$, $\Pi^1_1(\mathcal{L})$ denotes the set of formulas with universal quantifier prefix $\forall \ldots$ binding relational symbols of $\mathcal{L}$.

As to modal logic, recall from Chapter 2 that a modal language has a set of (modal) sort symbols and for each sort a set of (propositional) variables, a set of constants, and a set of connectives; in addition it has a set of modal operators. The modal formulas of sort s are built up from atomic symbols p_s, connectives $\bullet$ and modal operators $\#$ according to the rule $\phi ::= p_s \mid \bullet(\phi_{1,s}, \ldots, \phi_{n,s}) \mid \#(\phi_{s_1}, \ldots, \phi_{s_n})$, where it is assumed that $\bullet$, $\#$ return values of sort s. The semantics of a modal operator $\#$ is given by an *$\mathcal{L}$-pattern* $\delta_\#$, that is, by an $\mathcal{L}$-formula $\lambda x_{s_1} \ldots x_{s_n} . \phi(x_{s_1}, \ldots, x_{s_n}; x_{s_{n+1}}, \ldots, x_{s_m})$, where x_{s_i} is a variable of a classical sort s_i, $\phi(x_{s_1}, \ldots, x_{s_n}; x_{s_{n+1}}, \ldots, x_{s_m}) \in \mathrm{Form}_{\mathcal{L}}[\tau]$ for some τ, and $\mathcal{L}$ is a classical logic. Models for modal languages have the form $\mathfrak{M} = (W_s, \ldots, V)$

where $\mathfrak{M}$ is 'rich enough' to interpret the classical vocabulary in which the patterns for our modal operators live, and V is a valuation assigning subsets of W_s to symbols of sort s. Truth of modal formulas is given by $\mathfrak{M}, x \models p_s$ iff $x \in V(p_s)$ for atomic symbols p_s, the obvious clauses for connectives $\bullet$, and $\mathfrak{M}, x \models \#(\phi_1, \ldots, \phi_n)$ iff $\mathfrak{M}, x \models \delta_\#(I(\phi_1), \ldots, I(\phi_n))$.

The *standard translation* transcribes the truth definition of a modal language into a classical language containing predicate symbols p corresponding to the modal atomic symbol p_s: $ST(p_s) = p(x)$, ST commutes with connectives, and $ST(\#(\phi_1, \ldots, \phi_n)) = \delta_\#(ST(\phi_1), \ldots, ST(\phi_n))$. The important connection here is that for all modal formulas ϕ,

$$(W_s, \ldots, V), w \models \phi \text{ iff } (W_s, \ldots, V(p), \ldots) \models ST(\phi)[w],$$

where $V(p)$ is assigned to the predicate symbol p corresponding to p_s. In the context of the basic modal language $\mathcal{ML}(\Diamond)$ the notion of a *frame* arises when one quantifies over all possible valuations, thus arriving at second-order equivalents of modal formulas:

$$(W, R), x \models \phi \text{ iff } (W, R) \models \forall \vec{p}\, ST(\phi)[x].$$

This is generalized to arbitrary modal languages by selecting a modal sort s (with non-empty set of variables), and universally quantifying over all variables of that sort, while letting valuations take care of variables of the remaining sorts as before. Thus, we look at (higher-order) formulas of the form $\forall \vec{p}\, ST(\phi)$, where the $\forall \vec{p}$ binds all variables of sort s, rather than at formulas of the form $ST(\phi)$. Our prime question at this point is: when, and if so how, can we get rid of this higher-order quantification?

7.3 Reducibility

This section introduces the key notion of the Chapter.

Two examples

7.3.1. Example. In the basic modal language $\mathcal{ML}(\Diamond)$ the formula $p \to \Diamond p$ is equivalent to the second-order condition $\forall p\, (p(x) \to \exists y\, (Rxy \wedge p(y)))$ when interpreted on frames. By substituting $\lambda u.\, u = x$ for p in the second order formula, it reduces to $\exists y\, (Rxy \wedge y = x)$, or Rxx. This reduction yields an equivalence, one direction of which is just an instantiation; the validity of the other follows from the upward monotonicity of the consequent of $\forall p\, (p(x) \to \exists y\, (Rxy \wedge p(y)))$.

7.3.2. Example. Recall that propositional dynamic logic (**PDL**) has $\cup$ (union), ; (composition), and * (iteration) as operations on its relational component. Through the standard translation **PDL** ends up as a fragment of $\mathcal{L}_{\omega_1\omega}$; because of the Kleene star * we need to go infinitary here: $ST(\langle a \rangle p) = \exists y\, (\bigvee_n (R_a^n xy) \wedge$

$p(y))$. As an example, on frames the **PDL**-formula $p \wedge [a]p \rightarrow \langle b; a^*\rangle p$ is equivalent to the $\Pi^1_1(\mathcal{L}_{\omega_1\omega})$-condition

$$\forall p\, \Big(p(x) \wedge \forall y\, (R_a xy \rightarrow p(y)) \rightarrow \exists y'z'\, (Rb'_x z' \wedge R^*_a zy \wedge p(y))\Big). \qquad (7.1)$$

Substituting $\lambda u.\, (u = x \vee R_a xu)$ for p in (7.1) reduces it to the $\mathcal{L}_{\omega_1\omega}$-formula $\exists yz\, (R_b xz \wedge \bigvee_n (R^n_a zy) \wedge (y = x \vee R_a xy))$, or $\exists z\, (R_b xz \wedge R^*_a zx)$. To see this, observe that one direction is again an instantiation; the other follows from the upward monotonicity of the consequent of (7.1).

Examples 7.3.1, 7.3.2 show that in modal higher-order conditions the higher-order quantification can sometimes be removed through suitable substitutions — the question when and if so with which instances such reductions may be done, is analyzed in §§7.4 and 7.5.

Basics

Given a (classical) formula β involving variables $p_1, \ldots, p_n$ of some sort s, we want to know whether the Π^1_1-like formula $\forall p_1 \ldots \forall p_n\, \beta$ is equivalent to a formula γ not involving any variables of the sort s.

7.3.3. Definition. Let $\beta \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$, $\gamma \in \mathrm{Form}_{\mathcal{L}'}[\boldsymbol{\tau}']$, for some $\boldsymbol{\tau}, \boldsymbol{\tau}', \mathcal{L}, \mathcal{L}'$. We say that β *corresponds to* γ, or β *is reducible tos* γ if for every all $\mathfrak{A} \in \mathrm{Str}[\boldsymbol{\tau} \cup \boldsymbol{\tau}']$, and all $\vec{u} \in A$, we have $\mathfrak{A} \models_{\mathcal{L}'} \beta[\vec{u}]$ iff $\mathfrak{A} \models_{\mathcal{L}} \gamma[\vec{u}]$.

Note that I concentrate on *pointwise* reducibility, that is, on formulas that (may) depend on parameters. In most of the literature on correspondence theory for standard modal logic the emphasize has largely been put on a 'uniform' approach to reducibility, by considering only universally closed formulas; given the local perspective of this dissertation I have opted for the pointwise version.

In most practical cases Definition 7.3.3 will apply with $\boldsymbol{\tau} \supseteq \boldsymbol{\tau}'$ and $\mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$ usually contains all Π^1_1-like conditions over $\mathcal{L}'$.

7.3.4. Convention. In the sequel $\boldsymbol{\tau}$ is a fixed classical vocabulary, and s is a sort of $\boldsymbol{\tau}$ such that the only symbols of sort s are the variables $VAR_s = \{\, p_1, \ldots \}$; the elements of VAR_s are called s-variables. A formula is *s-universal* if it is of the form $\forall p_1 \ldots \forall p_n\, \beta$, where all s-variables occurring in it are bound by the prefix $\forall p_1 \ldots \forall p_n$. If $\forall \vec{p}\, \beta$ is an s-universal formula, we will tacitly assume that the prefix '$\forall \vec{p}$' contains all and only quantifiers binding s-variables. A formula β is *s-free* if it contains no (free or bound) occurrences of s-variables.

Here are some simple reducibility properties of s-universal formulas.

7.3.5. Proposition. *Let $\forall \vec{p}\, \beta$ be an s-universal formula in* $\mathrm{Form}[\boldsymbol{\tau}]$. *Then $\forall \vec{p}\, \beta$ reduces to an s-free formula in* $\mathrm{Form}[\boldsymbol{\tau}]$ *iff $\forall \vec{p}\, [\neg p_i / p_i]\beta$ does so.*

7.3.6. Proposition. *Assume $\forall \vec{p}\, \beta, \forall \vec{p}'\, \beta'$ are s-universal formulas in* $\mathrm{Form}[\boldsymbol{\tau}]$. *If $\forall \vec{p}\, \beta$ and $\forall \vec{p}'\, \beta'$ reduce to γ and γ', respectively, in* $\mathrm{Form}[\boldsymbol{\tau}]$, *then $\forall \vec{p}\vec{p}'\, (\beta \wedge \beta')$ reduces to $(\gamma \wedge \gamma')$. If $\forall \vec{p}\, \beta$ and $\forall \vec{p}'\beta'$ have distinct s-variables $\vec{p}$ and $\vec{p}'$, then $\forall \vec{p}\vec{p}'\, (\beta \vee \beta')$ reduces to $(\gamma \vee \gamma')$.*

7.3.7. PROPOSITION. *Let $\forall\vec{p}\,\beta(\vec{y};\vec{z})$ be an s-universal formula in* Form[$\boldsymbol{\tau}$] *that is reducible to the s-free $\gamma(\vec{y};\vec{z}) \in$* Form[$\boldsymbol{\tau}$]. *Assume that $\gamma'(\vec{x};\vec{y})$ is s-free. Then $\forall\vec{p}\forall\vec{y}\,(\gamma' \to \beta)$ is reducible to $\forall\vec{y}\,(\gamma' \to \gamma)$.*

Proof. Assume $\mathfrak{A} \models \forall\vec{p}\forall\vec{y}\,(\gamma'(\vec{x};\vec{y}) \to \beta(\vec{y};\vec{z}))[\vec{u}\vec{w}]$. If $\mathfrak{A} \models \gamma'(\vec{x};\vec{y})[\vec{u}\vec{v}\vec{w}]$ then obviously $\mathfrak{A} \models \forall\vec{p}\,\beta(\vec{y};\vec{z})[\vec{u}\vec{v}\vec{w}]$. So, by assumption, $\mathfrak{A} \models \gamma(\vec{y};\vec{z})[\vec{u}\vec{v}\vec{w}]$. ⊣

7.3.8. REMARK. The class of formulas χ such that $\forall\vec{p}\,\chi$ is reducible to an s-free formula, is not closed under $\neg$. To see this, let $\boldsymbol{\tau}$ contain a binary relation symbol R and a single predicate variable p. Consider the first-order formulas $\beta \equiv \exists y\,(Ryx \wedge p(y))$ and $\beta' \equiv \forall y\,(Ryx \wedge p(y) \to \exists z\,(Ryz \wedge p(z)))$. Then $\forall p\,\beta$ is reducible to a p-free formula, by 7.4.2, and $\forall p\,\beta'$ is reducible to a p-free formula by 7.5.3. Hence their conjunction $\forall p\,(\beta \wedge \beta')$ is reducible as well. However, $\forall p\,\neg(\beta \wedge \beta')$, i.e.,

$$\text{(WF)} \qquad \forall p\,\big(\exists y\,(Ryx \wedge p(y)) \to \exists y\,(Ryx \wedge p(y) \wedge \forall z\,(Ryz \to \neg p(z)))\big)$$

is not reducible to a p-free formula. It may be shown that (WF) expresses that R is well-founded, and hence is not elementary, or, reducible to a p-free formula over $\boldsymbol{\tau}$.

The converse of the first half of Proposition 7.3.6 does not hold. Since $\forall p\forall x\,(p(x) \to \exists y\,(Rxy \wedge p(y)))$ reduces to $\forall x\,Rxx$, the conjunction of the former formula with (WF) is inconsistent, hence reducible to a p-free formula, although (WF) is not.

7.4 FINDING THE RIGHT INSTANCES

As was observed before, in many practical cases reducibility results are obtained via suitable substitutions, if at all. In effect, this is the idea underlying the reduction algorithms mentioned in §7.1. For $\beta \equiv \forall\vec{p}\,\beta'$ an s-universal formula (over a classical vocabulary $\boldsymbol{\tau}$), they find an s-free equivalent of α (again, over $\boldsymbol{\tau}$) by taking suitable s-free *instances* $\gamma_1, \ldots, \gamma_n$ of the s-variables $p_1, \ldots, p_n$ in β such that

$$\models [\gamma_1/p_1, \ldots, \gamma_n/p_n]\beta' \to \forall\vec{p}\,\beta' \tag{7.2}$$

(the converse implication follows by instantiation). We are interested in combinations of Π^1_1-like s-universal formulas β of the form

$$\forall\vec{p}\,(\alpha \to \pi), \tag{7.3}$$

where π is monotone, and the antecedent α is a formula 'supplying' the substitution instances γ for $\vec{p}$ that yield the desired reduction of (7.3) to an s-free formula as in (7.2). The key-topic below is to make precise in what way the antecedent α supplies the substitution instances. We set down semantic (and syntactic) conditions on formulas that guarantee the existence of such instances, and we describe the instances needed. The results lead to a fully general formulation of the Sahlqvist-van Benthem and Gabbay-Ohlbach-Simmons algorithms in §7.5.

MONOTONICITY

We first examine the simplest instance of the general schema (7.3), where α is either $\top$ or $\bot$.

7.4.1. DEFINITION. Let $\pi(\vec{x}) \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$, let p be an s-variable, for s a sort in $\boldsymbol{\tau}$. We call $\pi(\vec{x})$ *upward (downward) monotone in* p if for all $\mathfrak{A} = (A, p, \dots) \in \mathrm{Str}[\boldsymbol{\tau}]$, and for all $\vec{u} \in A$, and all $p' \supseteq p$ ($p' \subseteq p$), we have that $\mathfrak{A} \models \pi[\vec{u}]$ implies $(A, p', \dots) \models \pi[\vec{u}]$.

The temporal logic formula Pp has $ST(Pp) = \exists y\,(Ryx \wedge p(y))$, which is monotone in p.

7.4.2. PROPOSITION. *Let* $\pi(\vec{x}) \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$ *be upward monotone in* p_i. *Assume* $\forall p_1 \dots \forall p_n\, \pi(\vec{x})$ *is s-universal. Then* $\forall p_1 \dots \forall p_n\, \pi(\vec{x})$ *is reducible to an s-free formula iff* $\forall p_1 \dots \forall p_{i-1} \forall p_{i+1} \dots \forall p_n\, [\bot/p_i]\pi(\vec{x})$ *is.*

Proof. If $\mathfrak{A} \models \forall p_1 \dots \forall p_n\, \pi[\vec{u}]$ then $\mathfrak{A} \models \forall p_1 \dots \forall p_{i-1} \forall p_{i+1} \dots \forall p_n\, [\bot/p_i]\pi[\vec{u}]$, for any $\mathfrak{A} \in \mathrm{Str}[\boldsymbol{\tau}]$. Using the fact that π is upward monotone in p_i, one sees that the converse implication holds as well. $\dashv$

7.4.3. PROPOSITION. *Let* $\pi(\vec{x}) \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$ *be downward monotone in* p_i. *Assume* $\forall p_1 \dots \forall p_n\, \pi(\vec{x})$ *is s-universal. Then* $\forall p_1 \dots \forall p_n\, \pi(\vec{x})$ *is reducible to an s-free formula iff* $\forall p_1 \dots \forall p_{i-1} \forall p_{i+1} \dots \forall p_n\, [\top/p_i]\pi(\vec{x})$ *is.*

7.4.4. COROLLARY. *Assume that for every s-variable* p, $\pi \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$ *is either upward or downward monotone in* p. *Then, if* $\forall \vec{p}\, \pi$ *is s-universal, it reduces to a p-free formula in* $\mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$ *via a suitable instantiation.*

Observe that the only instantiations needed in Corollary 7.4.4 are $\bot$ and $\top$.

As an example, the temporal formula $\mathit{Until}(\top, p)$ translates into $\exists y\,(Rxy \wedge \forall z\,(Rxz \wedge Rzy \rightarrow p(z)))$, which is upward monotone in p. Substituting $\bot$ for p, we find that on frames $\mathit{Until}(\top, p)$ is equivalent to $\exists y\,(Rxy \wedge \neg\exists z\,(Rxz \wedge Rzy))$.

To actually *use* semantic properties of formulas, a syntactic characterization of all and only the formulas having the properties comes in handy. For monotonicity this involves positive and negative occurrences. An occurrence of a symbol is said to be *positive* iff it is within the scope of an even number of negation signs; otherwise an occurrence is called *negative*.

7.4.5. THEOREM. *Let* $\beta \in \mathrm{Form}_{\mathcal{L}_{\omega\omega}}[\boldsymbol{\tau}]$, $p \in VAR_s$. *Then* β *is upward (downward) monotone in* p *iff* β *is equivalent to a formula in* $\mathrm{Form}_{\mathcal{L}_{\omega\omega}}[\boldsymbol{\tau}]$ *in which all occurrences of* p *are positive (negative).*

Proof. We prove the characterization of upward monotonicity only. A quick proof using Lyndon Interpolation runs as follows. For a new relation symbol p' let $\boldsymbol{\tau}'$ be $\boldsymbol{\tau}$ extended with p'; $\boldsymbol{\tau}'$-structures then take the form $(\mathfrak{A}, X)$, where $\mathfrak{A}$ is a $\boldsymbol{\tau}$-structure, and X is a relation over an appropriate domain in $\mathfrak{A}$ which interprets p'. The assumption that β is upward monotone in p amounts to $\beta(p'), \forall \vec{x}\,(p'(\vec{x}) \rightarrow p(\vec{x})) \models \beta(p)$, for a new relation symbol p'. Let γ be an appropriate Lyndon-interpolant. As p' occurs only on the left-hand side of the $\models$-sign, γ does not contain p'; and as p occurs only positively on the left-hand side, p is positive in γ. Hence, γ is the required equivalent. $\dashv$

The result extends to many other logics, including all logics that have Lyndon Interpolation such as $\mathcal{L}_{\omega_1\omega}$.

Continuity: the basic case

We now allow the scheme (7.2) to contain continuous antecedent formulas.

7.4.6. Definition. Let $\alpha(\vec{x}) \in \mathrm{Form}_{\mathcal{L}}[\tau]$. Then $\alpha(\vec{x})$ is called *continuous in* $p \in VAR_s$ if for all $\mathfrak{A} = (A, \bigcup_i T_i, \ldots) \in \mathrm{Str}[\tau]$, where $\bigcup_i T_i$ interprets p, and for all $\vec{u} \in A$, we have $\mathfrak{A} \models \alpha[\vec{u}]$ iff $(A, T_i, \ldots) \models \alpha[\vec{u}]$, for some i.

As an example, both $\exists y\,(p(y) \wedge q(y))$ and $\exists y\,(p(y) \wedge \neg q(y))$ are continuous in p; their conjunction is not, however. Hence the class of continuous formulas is not closed under $\wedge$.

As a further example, let $\mathfrak{B}$ be a complete Boolean algebra with operators (BAO), and let f be a completely additive n-ary operator on $\mathfrak{B}$. According to the well-known duality between BAO's and modal frames, f can be represented as a relation R_f on such frames (cf. (Jónsson & Tarski 1952, De Rijke & Venema 1991)). Then, the modal operator $\Diamond_f$, defined by

$$\Diamond_f(p_1, \ldots, p_n) = \{\, x : \exists y_1 \in p_1 \ldots \exists y_n \in p_n\, Rxy_1 \ldots y_n \,\},$$

is a continuous operator. This connection can be made into full-fledged representation: a formula $\beta(p_1, \ldots, p_n; x_1, \ldots x_m)$ is continuous in $\vec{p}$ iff in each model $\mathfrak{A} = (A, \ldots)$ the set $\{\, \vec{a} : \mathfrak{A} \models \beta[\vec{a}] \,\}$ can be represented as the R-image of p_1, $\ldots$, p_n, for some $R \subseteq A^{n+m}$.

7.4.7. Proposition. *Let $\beta(\vec{x}) \in \mathrm{Form}_{\mathcal{L}}[\tau]$, and let p be an s-variable. Then β is continuous in p iff for all τ-structures $\mathfrak{A} = (A, T, \ldots)$, where T interprets p, we have: $\mathfrak{A} \models \beta[\vec{u}]$ iff either $(A, \emptyset, \ldots) \models \beta[\vec{u}]$ or for some $\vec{t} \in T$, $(A, \{\, \vec{t} \,\}, \ldots) \models \beta[\vec{u}]$.*

Proof. For the *if* direction consider the set $\bigcup_i T_i$. We have $(A, \bigcup_i T_i, \ldots) \models \beta[\vec{u}]$ iff either $(A, \emptyset, \ldots) \models \beta[\vec{u}]$ or for some $\vec{t} \in \bigcup_i T_i$, $(A, \vec{t}, \ldots) \models \beta[\vec{u}]$, iff for some i such that $\vec{t} \in T_i$, $(A, \{\, \vec{t} \,\}, \ldots) \models \beta[\vec{u}]$. For the *only-if* direction observe that $(A, T, \ldots) \models \beta[\vec{u}]$ iff $(A, \bigcup_{\vec{t} \in T}\{\, \vec{t} \,\} \cup \emptyset, \ldots) \models \beta[\vec{u}]$ iff either $(A, \emptyset, \ldots) \models \beta[\vec{u}]$, or for some $\vec{t}$, $(A, \{\, \vec{t} \,\}, \ldots) \models \beta[\vec{t}]$, as required. $\dashv$

In general, continuity of a formula in p_1, $\ldots$, p_k can be equivalently stated as 2^k possibilities; because of this 'explosion' we don't state results on continuity in full generality.

7.4.8. Lemma. *Let $\pi(\vec{x}; \vec{y}; \vec{z}) \in \mathrm{Form}_{\mathcal{L}}[\tau]$ be upward monotone in p, and assume $\alpha(\vec{x}; \vec{y}; \vec{z}') \in \mathrm{Form}_{\mathcal{L}}[\tau]$ is continuous in p. Then $\forall p \forall \vec{y}\, (\alpha \to \pi)$ is reducible to a p-free formula via suitable instantiations.*

Proof. The instances we need here are of the form $\lambda\vec{z}.\,\vec{z} = \vec{y}$, $\lambda\vec{z}.\,\vec{z} = \vec{z}$, or $\lambda\vec{z}.\,\vec{z} \neq \vec{z}$ depending on whether p occurs in α and π, only in α, or only in π. Assume first that p occurs both in α and π. Then

$$\models \Big(\forall\vec{y}\,[(\lambda\vec{y}'.\,\vec{y}' = \vec{y})/p]\forall\vec{z}\,(\alpha \to \pi)\Big) \to \Big(\forall p\forall\vec{z}\,(\alpha \to \pi)\Big).$$

To see this, assume $(\mathfrak{A}, T) \models \forall\vec{y}\,[(\lambda\vec{y}'.\vec{y}' = \vec{y})/p](\alpha \to \pi), \alpha[\vec{u}; \vec{z} \mapsto \vec{w}]$, where T interprets p, and $\vec{z} \mapsto \vec{w}$ means that $\vec{w}$ is assigned to $\vec{z}$. Now, $(\mathfrak{A}, T) \models \alpha[\vec{u}; \vec{z} \mapsto \vec{w}]$ implies that for some $\vec{t} \in T$, $(\mathfrak{A}, \{\,\vec{t}\,\}) \models \alpha[\vec{u}; \vec{z} \mapsto \vec{w}]$, by 7.4.7 and monotonicity. Hence $\mathfrak{A} \models [(\lambda\vec{y}'.\vec{y}' = \vec{y})/p]\alpha[\vec{u}; \vec{y} \mapsto \vec{t}; \vec{z} \mapsto \vec{w}]$, so $\mathfrak{A} \models [(\lambda\vec{y}'.\vec{y}' = \vec{y})/p]\pi[\vec{u}; \vec{y} \mapsto \vec{t}; \vec{z} \mapsto \vec{w}]$. But then, by monotonicity, $(\mathfrak{A}, T) \models \pi[\vec{u}; \vec{z} \mapsto \vec{w}]$, as required.

Next, if p occurs only in α, then $\alpha \to \pi$ is downward monotone in p. Hence $\forall p\forall\vec{y}\,(\alpha \to \pi)$ reduces to a p-free formula by instantiating with $\lambda\vec{z}.\,\vec{z} = \vec{z}$ as in 7.4.3. The case that p occurs only in π is entirely analogous. ⊣

How can we apply Lemma 7.4.8 to obtain reducibility results in 'real life' modal formalisms? §7.5 contains a double answer in the form of the Sahlqvist-van Benthem and Gabbay-Ohlbach-Simmons algorithms. For readers unable to wait until then, the following is a bare-bones sketch of how to proceed:

- Translate your modal formula into classical logic, preferably into a formula of the form $\forall p\forall\vec{y}\,(\alpha \to \pi)$.
- Perform some cleaning up in the antecedent of the translation to reveal the substitutions needed. As may be seen from the proof of Lemma 7.4.8, for continuous α the required substitution instances are singletons.
- Perform the substitution, and do some cleaning up.

Here are two examples; formulas supplying the substitutions are underlined.

EXAMPLE. Consider the formula $\Diamond p \to \Box p$ in $\mathcal{ML}(\Diamond)$.

- Higher-order translation: $\forall p\,(\exists y\,(Rxy \land p(y)) \to \forall z\,(Rxz \to p(z)))$,
- after rewriting: $\forall p\forall y\,(Rxy \land \underline{p(y)} \to \forall z\,(Rxz \to p(z)))$, which has an antecedent continuous in p, and a consequent upward monotone in p,
- substituting $\lambda u.\,u = y$ for p reduces this to $\forall y\,(Rxy \to \forall z\,(Rxz \to z = y))$.

EXAMPLE. In Venema (1991)'s modal logic of converse and composition, one has a binary modal operator $\circ$ based on a ternary relation C. Consider the formula $(a \circ b) \to (b \circ a)$.

- Higher-order translation: $\forall ab\,(\exists yz\,(Cxyz \land a(y) \land b(z)) \to \exists y'z'\,(Cxy'z' \land b(y') \land a(z'))$.
- after rewriting: $\forall ab\forall yz\,(Cxyz \land \underline{a(y)} \land \underline{b(z)} \to \exists y'z'\,(Cxy'z' \land b(y') \land a(z'))$, which has an antecedent continuous in a, b, and a consequent upward monotone in a, b,
- substituting $\lambda u.\,u = y$ for a, $\lambda u.\,u = z$ for b reduces this to $\forall yz\,(Cxyz \to Cxzy)$.

To facilitate locating the right substitution instance it is useful to syntactically characterize the continuous formulas.

7.4.9. Definition. Let $\beta \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$, and let p be an s-variable. Then β is called *distributive in* p if it is of the form $\exists \vec{x}\,(p(\vec{x}) \wedge \beta') \vee \gamma$, where β', γ are p-free.

An example from **PDL**: $\langle a\rangle\langle b^*\rangle p$ translates into $\exists yz\,(p(z) \wedge R_a xy \wedge \bigvee_n (R_b^n yz))$ — a formula that is distributive in p.

7.4.10. Theorem. *Let $\beta \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$, and let p be an s-variable. Then β is continuous in p iff β is equivalent to a formula that is distributive in p.*

Proof. I only prove the *only-if* direction. Let β be continuous in p. Let $\mathfrak{A} = (A, T, \ldots) \models \beta[\vec{u}]$, where T interprets p. Then, by continuity and 7.4.7,

$$(A, T, \ldots) \models \Big(\exists \vec{x}\,(p(\vec{x}) \wedge [(\lambda \vec{y}.\, \vec{y} = \vec{x})/p]\beta) \vee [(\lambda \vec{y}.\, \vec{y} \neq \vec{y})/p]\beta\Big)[\vec{u}].$$

Let γ denote the latter formula. Then γ has the required syntactic form. Moreover, as γ does not depend on $\mathfrak{A}$ or $\vec{u}$, we have that $\models \beta \rightarrow \gamma$; but by the continuity of β and 7.4.7 this can be strengthened to $\models \beta \leftrightarrow \gamma$, as required. $\dashv$

If in $\forall \vec{p}\,(\alpha \rightarrow \pi)$ the antecedent α is distributive in p, then it is continuous in p by Theorem 7.4.10 — hence the required substitution instance is simply $\lambda \vec{u}.\, \vec{u} = \vec{y}$, where $\vec{y}$ is the unique occurrence $p(\vec{y})$ of p in α.

Generalizing continuity: small subsets

The important features of continuous formulas are that their semantic value may be computed locally (on singletons), and that they are upward monotone. We now generalize from the basic case by maintaining upward monotonicity but liberalizing local computability to 'depends only on *small* sets;' after that we replace the latter with 'depends only on a *definable* set.'

7.4.11. Definition. Let $\beta(\vec{x}) \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$, and let p be an s-variable. For λ a cardinal, β is called *λ-continuous in* p, if for all $\mathfrak{A} = (A, \bigcup_{i\in I} T_i, \ldots) \in \mathrm{Str}[\boldsymbol{\tau}]$, where $\bigcup_{i\in I} T_i$ interprets p, and for all $\vec{u} \in A$, we have $\mathfrak{A} \models \beta[\vec{u}]$ iff there is an $I_0 \subseteq I$ with $|I_0| < \lambda$ and $(A, \bigcup_{i\in I_0} T_i, \ldots) \models \beta[\vec{u}]$.

Further, $\beta(\vec{x})$ is called *globally λ-continuous in* p if there is a $\kappa < \lambda$ such that for all $\mathfrak{A} = (A, \bigcup_{i\in I} T_i, \ldots) \in \mathrm{Str}[\boldsymbol{\tau}]$, and for all $\vec{u} \in A$, we have $\mathfrak{A} \models \beta[\vec{u}]$ iff for some $I_0 \subseteq I$, $|I_0| \leq \kappa$ and $(A, \bigcup_{i\in I_0} T_i, \ldots) \models \beta[\vec{u}]$.

In Roorda (1993)'s modal approach to Lambek calculus the formula $\triangle(p \wedge q, p \wedge \neg q)$ translates into $\exists yz\,(Cxyz \wedge p(y) \wedge q(y) \wedge p(z) \wedge \neg q(z))$. This formula is not continuous in p; it is 3-continuous in p.

7.4.12. Proposition. *Let $\beta(\vec{x}) \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$, and let p be an s-variable. Then β is λ-continuous in p iff for every $\mathfrak{A}$ we have $\mathfrak{A} = (A, T, \ldots) \models \beta[\vec{u}]$ iff for some $T_0 \subseteq T$ with $|T_0| < \lambda$, $(A, T_0, \ldots) \models \beta[\vec{u}]$.*

Recall that $\mathcal{L}$ has the *Löwenheim-Skolem property down to* κ if each satisfiable $\mathcal{L}$-formula has a model of power $\leq \kappa$. (The power of a $\boldsymbol{\tau}$-structure $\mathfrak{A}$ is defined as $|A|$ in the one-sorted case, and as $\sum_{s\in\boldsymbol{\tau}} |A_s|$ in the many-sorted case.)

We say that $\beta(p)$ *commutes with unions of non-decreasing chains of sets of length* λ if for every a non-decreasing chain of sets $\{ T_i \}_{i<\lambda}$ we have $\mathfrak{A} = (A, \bigcup_{i<\lambda} T_i, \ldots) \models \beta[\vec{u}]$ iff for some $\kappa < \lambda$, $(A, T_\kappa, \ldots) \models \beta[\vec{u}]$.

7.4.13. PROPOSITION. *Assume $\mathcal{L}$ has the Löwenheim-Skolem property down to λ. Let β be an $\mathcal{L}$-formula, and let p be an s-variable. Then β is λ-continuous in p iff β commutes with unions of non-decreasing chains of sets of length λ.*

Proof. The *only if* direction: assume $\{ T_i \}_{i<\lambda}$ is a non-decreasing chain of sets such that $(A, \bigcup_{i<\lambda} T_i, \ldots) \models \beta[\vec{u}]$, where $\bigcup_{i<\lambda} T_i$ interprets p. By λ-continuity this is equivalent to: for some $\kappa < \lambda$, $(A, \bigcup_{i<\kappa} T_i, \ldots) \models \beta[\vec{u}]$; and as the T_i's form a non-decreasing chain, this is equivalent to $(A, T_\kappa, \ldots) \models \beta[\vec{u}]$, as required.

For the converse, assume that β commutes with unions of non-decreasing chains of length λ. Let $(A, T, \ldots) \models \beta[\vec{u}]$. We may assume that $|A| \leq \lambda$. Then $T = \bigcup_{i<\lambda} T_i$, where $T_0 \subseteq T_1 \subseteq \cdots$ all have $|T_i| < \lambda$. Hence, $(A, T_\kappa \ldots) \models \beta[\vec{u}]$, for some $\kappa < \lambda$, which is sufficient by 7.4.12. Conversely, if $(A, T_0, \ldots) \models \beta[\vec{u}]$, for some $T_0 \subseteq T$ with $|T_0| < \lambda$, define $T_i = T$ $(0 < i < \lambda)$. Then, by the assumption on β, $(A, T, \ldots) \models \beta[\vec{u}]$. ⊣

7.4.14. PROPOSITION. *Assume $\mathcal{L}$ is λ-compact. Let $\beta \in \mathrm{Form}_{\mathcal{L}}[\tau]$, and let p be an s-variable. Then β is λ-continuous in p iff it is globally λ-continuous in p.*

Proof. I only prove the direction from left to right. If β is λ-continuous, then

$$\models \beta \leftrightarrow \bigvee_{\kappa<\lambda} \underbrace{\exists \vec{y}_0 \ldots \exists \vec{y}_\nu \ldots}_{\kappa} \Big(\bigwedge_{i\leq\kappa} p(\vec{y}_i) \wedge [(\lambda \vec{z}.\ \bigvee_{i\leq\kappa} \vec{z} = \vec{y}_i)/p]\beta \Big).$$

By compactness there is a $\kappa_0 < \lambda$ such that

$$\models \beta \rightarrow \bigvee_{\mu\leq\kappa_0} \underbrace{\exists \vec{y}_0 \ldots \exists \vec{y}_\nu \ldots}_{\mu} \Big(\bigwedge_{i\leq\mu} p(\vec{y}_i) \wedge [(\lambda \vec{z}.\ \bigvee_{i\leq\mu} \vec{z} = \vec{y}_i)/p]\beta \Big). \qquad (7.4)$$

As β is upward monotone in p, the implication in (7.4) must be an equivalence. ⊣

7.4.15. EXAMPLE. In $\mathcal{L}_{\omega\omega}$ ω-continuity and global ω-continuity coincide, according to 7.4.14. Thus, we need to go beyond $\mathcal{L}_{\omega\omega}$ to find an example of a formula that is ω-continuous, but not globally. In $\mathcal{L}_{\omega_1\omega}$ let β be the statement 'at most n elements satisfy p', and put $\beta := \bigvee_n \beta$. Then β is ω-continuous, but not globally so.

Likewise, in *weak second order logic* $\mathcal{L}^{w2}$, where the relation variables range over *finite* sets only, the statement $\exists q \forall x\, (q(x) \rightarrow p(x))$ is locally, but obviously not globally ω-continuous.

In the setting of Boolean algebras with operators, the operators f defined by globally ω-continuous formulas are also known as *(completely) ω-additive* ones: $f(\Sigma U) = \Sigma\{ f(\Sigma(T)) : T \subseteq U, |T| \leq m \}$, for some $m \in \omega$ (cf. (Henkin 1970)).

7.4.16. LEMMA. *Assume that $\pi(\vec{x}) \in \mathrm{Form}_{\mathcal{L}}[\tau]$ is upward monotone in p, and let $\alpha(\vec{x}) \in \mathrm{Form}_{\mathcal{L}}[\tau]$ be globally ω-continuous in p. Then $\forall \vec{p}\, (\alpha(\vec{x}) \rightarrow \pi(\vec{x}))$ reduces to a p-free formula via suitable instantiations.*

Proof. The instances we need here are of the form $\lambda\vec{z}.\ \bigvee_{i\leq n}\vec{z} = \vec{y}_i$ $(n < \omega)$, $\lambda\vec{z}.\,\vec{z} = \vec{z}$, and $\lambda\vec{z}.\,\vec{z} \neq \vec{z}$, depending on whether p occurs both in α and π, only in α or only in π. The latter two cases are analogous to the corresponding cases in 7.4.8. So assume p occurs both in α and π. Let $n < \omega$ be the upper bound given by global ω-continuity. Then the following is universally valid:

$$\Big(\bigwedge_{0\leq i\leq n} \forall\vec{y}_0 \ldots \forall\vec{y}_i\,[(\lambda\vec{y}'.\bigvee_i \vec{y}_i = \vec{y}')/p]\forall\vec{z}\,(\alpha(\vec{x}) \to \pi(\vec{x}))\Big) \to \Big(\forall p\forall\vec{z}\,(\alpha \to \pi)\Big).$$

This may be seen by using 7.4.12 and arguing as in 7.4.8. This suffices. ⊣

To be able to restate 7.4.16 for arbitrary $\lambda > \omega$ we need to assume that $\mathcal{L}$ is closed under quantifier strings and disjunctions of arbitrary length $< \lambda$.

By 7.4.14 the requirement in 7.4.16 that α be a globally ω-continuous formula may be weakened to ω-continuity whenever $\mathcal{L}$ is $\aleph_0$-compact.

EXAMPLE. Consider the formula $\Diamond p \wedge \Diamond\Diamond p \to \Box p$ in $\mathcal{ML}(\Diamond)$.

- Higher-order equivalent:

$$\forall p\,(\exists y\,(Rxy \wedge p(y)) \wedge \exists y'\,(Rxy' \wedge \exists y''\,(Ry'y'' \wedge p(y'')) \to \forall z\,(Rxz \to p(z))),$$

- after rewriting:

$$\forall p\forall yy'y''\,(Rxy \wedge \underline{p(y)} \wedge Rxy' \wedge Ry'y'' \wedge \underline{p(y'')} \to \forall z\,(Rxz \to p(z))),$$

- substituting $\lambda u.\,(u = y \vee u = y')$ for p reduces this to

$$\forall yy'y''\,(Rxy \wedge Rxy' \wedge Ry'y'' \to \forall z\,(Rxz \to (z = y \vee z = y''))).$$

EXAMPLE. Van der Hoek & De Rijke (1992, 1993) study systems of graded modal logic containing modal operators $\langle R\rangle_k p$ whose translation reads

$$\exists x_0 \ldots x_k\,(\bigwedge_i Rxx_i \wedge \bigwedge_{0\leq i\neq j\leq k} (x_i \neq x_j) \wedge \bigwedge_i p(x_i));$$

the latter is clearly not continuous, but it is $k+1$-continuous. Consider the graded modal formula $p \wedge \langle R\rangle_k q \to \langle R\rangle_0(q \wedge \langle R\rangle_0 p)$.

- Higher-order equivalent:

$$\forall pq\forall x_0 \ldots x_k\,\Big(p(x) \wedge \bigwedge_i Rxx_i \wedge \bigwedge_{0\leq i\neq j\leq k} (x_i \neq x_j) \wedge \bigwedge_i q(x_i) \to \exists y\,(Rxy \wedge q(y) \wedge \exists z\,(Ryz \wedge p(z)))\Big), \quad (7.5)$$

  which is of the form prescribed by Lemma 7.4.16,
- substituting $\lambda u.\,u = x$ for p, and $\lambda u.\ \bigvee_{i\leq k}(u = x_i)$ for q reduces (7.5) to

$$\forall x_0 \ldots x_k\,(\bigwedge_i Rxx_i \wedge \bigwedge_{i\neq j}(x_i \neq x_j) \to \bigvee_{i\leq k} Rx_ix).$$

7.4.17. DEFINITION. Let $\beta \in \mathrm{Form}_{\mathcal{L}_{\omega\omega}}[\tau]$, and let p be an s-variable. Then β is called *ω-distributive in p* if is built up from p-free formulas and atomic formulas $p(\vec{x})$ using only $\wedge$, $\vee$ and $\exists$.

7.4.18. THEOREM. *Let $\beta \in \mathrm{Form}_{\mathcal{L}_{\omega\omega}}[\tau]$, and let p be an s-variable. Then β is ω-continuous in p iff it is equivalent to a formula that is ω-distributive in p.*

Proof. This is immediate from 7.4.14. $\dashv$

What about λ-continuity for $\lambda > \omega$? As with 7.4.16 more general versions of 7.4.18 may be obtained by requiring suitable syntactic closure conditions and using appropriate versions of compactness.

A potentially more interesting issue is this: what are the ω-continuous formulas in extensions of $\mathcal{L}_{\omega\omega}$? In the case of $\mathcal{L}_{\omega_1\omega}$ the answer is almost immediate from 7.4.14: an $\mathcal{L}_{\omega_1\omega}$-formula is ω-continuous in p iff it is equivalent to a formula constructed from p-free formulas and atomic formulas $p(\vec{x})$ using only $\wedge$, $\bigvee$, $\exists$.

As with continuous formulas the characterization result for ω-continuous formulas is useful in locating the required substitution instances; they are finite disjunctions of the form $\lambda\vec{u}.\,(\bigvee_{i\leq n}(\vec{u} = \vec{y}_i))$.

GENERALIZING CONTINUITY: DEFINABLE SUBSETS

The next obvious way to generalize the notion of continuity is to demand that β holds of p not iff it holds of a singleton in p, but iff it holds of a definable subset of p. In this approach we fix some set X from which the possible definitions of subsets of p may be taken. As in earlier cases, both *local* and *global* versions are possible.

7.4.19. DEFINITION. A subset $X \subseteq A$ is *$\mathcal{L}$-definable* in $\mathfrak{A}$ if there is an $\mathcal{L}$-formula $\gamma(\vec{x};\vec{y})$ and elements $\vec{t} \in \mathfrak{A}$ such that $X = \{\,\vec{t} : \mathfrak{A} \models \gamma[\vec{u};\vec{t}]\,\}$. If s is a sort in τ, a subset which is definable is *s-free definable* if it has an s-free definition.

The following may be somewhat hard to digest at first. The reward will be considerable, however, as the following will allow us to obtain reducibility results encompassing and vastly extending our earlier results.

7.4.20. DEFINITION. Let $\beta(\vec{x}) \in \mathrm{Form}_{\mathcal{L}'}[\tau]$, and let p be an s-variable. Then $\beta(\vec{x})$ is *$\mathcal{L}$-definably continuous* in p if for all $\mathfrak{A} = (A, T, \ldots) \in \mathrm{Str}[\tau]$, where T interprets p, and for all $\vec{u} \in A$ we have $\mathfrak{A} \models \beta(\vec{x})[\vec{u}]$ iff for some s-free $\mathcal{L}$-definable subset $X_\gamma = \{\,\vec{t} : \mathfrak{A} \models \gamma(\vec{x};\vec{y})[\vec{u}\vec{t}]\,\}$ of T we have $(A, X_\gamma, \ldots) \models \beta(\vec{x})[\vec{u}]$.

Also, $\beta(\vec{x})$ is *$\mathcal{L}$-definably continuous in p with additional parameters* if for all $\mathfrak{A} = (A, T, \ldots) \in \mathrm{Str}[\tau]$, where T interprets p, and for all $\vec{u} \in A$ we have $\mathfrak{A} \models \beta[\vec{u}]$ iff for some subset $X_\gamma = \{\,\vec{t} : \mathfrak{A} \models \gamma(\vec{x};\vec{y};\vec{z})[\vec{u}\vec{t}\vec{w}]\,\}$ of T that is s-free and $\mathcal{L}$-definable, we have $(A, X_\gamma, \ldots) \models \beta[\vec{u}]$.

Further, $\beta(\vec{x})$ is *globally $\mathcal{L}$-definably continuous* if there is a *fixed* finite stock of $\mathcal{L}$-formulas $\gamma_0(\vec{x};\vec{y}), \ldots, \gamma_n(\vec{x};\vec{y})$ such that for all $\mathfrak{A} = (A, T, \ldots)$ and $\vec{u}$ in A, we have $\mathfrak{A} \models \beta(\vec{x})[\vec{u}]$ iff for some i $(0 \leq i \leq n)$ $(A, \{\,\vec{t} : \mathfrak{A} \models \gamma_i(\vec{x};\vec{y})[\vec{u}\vec{t}]\,\}, \ldots) \models \beta(\vec{x})[\vec{u}\vec{t}]$. A global version of $\mathcal{L}$-definable continuity with parameters is defined analogously.

7.4.21. EXAMPLE. Let $\beta \equiv \forall y\,(\exists z\,(Rxz \wedge Rzy \rightarrow p(y)))$; then β is $\mathcal{L}_{\omega\omega}$-definably continuous: $\mathfrak{A} = (A, T, \ldots) \models \beta[u]$, where T interprets p, implies $(A, \{\, v : \mathfrak{A} \models \exists z\,(Rxz \wedge Rzy)[uv]\,\}, \ldots) \models \beta[u]$; the converse implication follows from the fact that β is monotone in p.

For a first-order formula that is *not* $\mathcal{L}_{\omega\omega}$-definably continuous, consider

$$\beta \equiv \forall y\,\Big(Rxy \rightarrow [\exists z\,(Ryz \wedge p(z)) \wedge \exists z\,(Ryz \wedge \neg p(z))]\Big),$$

and let $\mathfrak{A} = (\mathbb{N}, \leq, \{\, 2n : n \in \mathbb{N}\,\})$, where $\leq$ interprets R and $\{\, 2n : n \in \mathbb{N}\,\}$ interprets p. Then $\mathfrak{A} \models \beta[0]$. The only $\mathcal{L}_{\omega\omega}$-definable subsets of $\mathbb{N}$ (in terms of R, $=$) are the finite and co-finite sets. But clearly, for no finite or co-finite subset X of $\{\, 2n : n \in \mathbb{N}\,\}$, $(\mathbb{N}, \leq, X) \models \beta[0]$.

7.4.22. PROPOSITION. *Let $\beta(\vec{x}) \in \mathrm{Form}_{\mathcal{L}_{\omega\omega}}[\tau]$, and let p be an s-variable. Then*

1. *β is $\mathcal{L}_{\omega\omega}$-definably continuous in p iff it is globally $\mathcal{L}_{\omega\omega}$-definably continuous in p, and*
2. *β is $\mathcal{L}_{\omega\omega}$-definably continuous in p with parameters iff it is globally $\mathcal{L}_{\omega\omega}$-definably continuous in p with parameters.*

Proof. 1. We only prove the *only-if* direction. Let $\mathfrak{A} \in \mathrm{Str}[\tau]$. By continuity there is an s-free $\mathcal{L}_{\omega\omega}$-formula γ such that $\mathfrak{A} = (A, T, \ldots) \models \beta[\vec{u}]$ implies that

$$(A, \{\, \vec{t} : \mathfrak{A} \models \gamma(\vec{x}; \vec{y})[\vec{u}\vec{t}]\,\}, \ldots) \models \beta(\vec{x}) \wedge \forall \vec{y}\,(\gamma(\vec{x}; \vec{y}) \rightarrow p(\vec{y}))[\vec{u}], \tag{7.6}$$

and hence

$$(A, \{\, \vec{t} : \mathfrak{A} \models \gamma(\vec{x}; \vec{y})[\vec{u}\vec{t}]\,\}, \ldots) \models [\lambda \vec{y}.\, \gamma(\vec{x}; \vec{y})/p]\beta(\vec{x}) \wedge \forall \vec{y}\,(\gamma(\vec{x}; \vec{y}) \rightarrow p(\vec{y}))[\vec{u}].$$

Let $\beta'_{\mathfrak{A},\vec{u}}$ denote the latter formula. Then $\mathfrak{A} \models \beta \leftrightarrow \beta'_{\mathfrak{A},\vec{u}}[\vec{u}]$. So

$$\models \beta \leftrightarrow \bigvee_{\{\, (\mathfrak{A},\vec{u}) : \mathfrak{A} \models \beta[\vec{u}]\,\}} \beta'_{\mathfrak{A},\vec{u}}[\vec{u}].$$

By compactness the latter disjunction reduces to a finite one, that is, for some n we have

$$\models \beta \leftrightarrow \bigvee_{0 \leq i \leq n} \Big([\lambda \vec{y}.\, \gamma_i(\vec{x}; \vec{y})/p]\beta(\vec{x}) \wedge \forall \vec{y}\,(\gamma_i(\vec{x}; \vec{y}) \rightarrow p(\vec{y}))\Big),$$

where all γ_is are s-free $\mathcal{L}_{\omega\omega}$-formulas.

2. This is proved like 1. We replace (7.6) with

$$\Big(A, \{\, \vec{t} : \mathfrak{A} \models \gamma(\vec{x}; \vec{y}; \vec{z})[\vec{u}\vec{t}\vec{w}]\,\}, \ldots\Big) \models \beta(\vec{x}) \wedge \forall \vec{y}\,(\gamma(\vec{x}; \vec{y}; \vec{z}) \rightarrow p(\vec{y}))[\vec{u}\vec{w}], \tag{7.7}$$

where γ is the formula given by the continuity of β, and the $\vec{w}$ are additional parameters. Clearly, (7.7) implies that $(A, \{\, \vec{t} : \mathfrak{A} \models \gamma(\vec{x}; \vec{y}; \vec{z})[\vec{u}\vec{t}\vec{w}]\,\}, \ldots)$ satisfies

$$\exists \vec{z}\,\Big([\lambda \vec{y}.\, \gamma(\vec{x}; \vec{y}; \vec{z})/p]\beta(\vec{x}) \wedge \forall \vec{y}\,(\gamma(\vec{x}; \vec{y}; \vec{z}) \rightarrow p(\vec{y}))\Big)$$

at $\vec{u}$. Reasoning as before one derives that

$$\models \beta \leftrightarrow \bigvee_{0 \le i \le n} \exists \vec{z}_i \Big([\gamma_i(\vec{x}; \vec{y}; \vec{z}_i)/p]\beta(\vec{x}) \wedge \forall \vec{y}\, (\gamma(\vec{x}; \vec{y}; \vec{z}_i) \to p(\vec{y}))\Big).$$

This implies that β is globally $\mathcal{L}_{\omega\omega}$-definably continuous with parameters. ⊣

7.4.23. LEMMA. *Let $\pi(\vec{x}) \in \mathrm{Form}_{\mathcal{L}}[\tau]$ be upward monotone in p.*

1. *Assume $\alpha(\vec{x}) \in \mathrm{Form}_{\mathcal{L}'}[\tau]$ is globally $\mathcal{L}$-definably continuous in p. Then $\forall \vec{p}\, (\alpha(\vec{x}) \to \pi(\vec{x}))$ reduces to a p-free formula via suitable instantiations.*
2. *Assume $\alpha(\vec{x}) \in \mathrm{Form}_{\mathcal{L}'}[\tau]$ is globally $\mathcal{L}$-definably continuous in p with additional parameters. Then $\forall \vec{p}\vec{z}\, (\alpha(\vec{x}; \vec{z}) \to \pi(\vec{x}; \vec{z}))$ reduces to a p-free formula via suitable instantiations.*

Proof. The instantiations needed are of the form $\lambda\vec{z}.\, \gamma(\vec{x}; \vec{y})$, where γ is a p-free $\mathcal{L}$-formula, $\lambda\vec{z}.\, \vec{z} = \vec{z}$, and $\lambda\vec{z}.\, \vec{z} \neq \vec{z}$, depending on whether p occurs both in α and π, only in α or only in π. To see this, assume p occurs both in α and π (the other cases are as before). Let $\gamma_0(\vec{x}; \vec{y})$, ..., $\gamma_n(\vec{x}; \vec{y})$ be the p-free formulas given by the definable continuity of α. It suffices to show that

$$\models \Big(\bigwedge_{0 \le i \le n} [\lambda\vec{y}.\, \gamma_i(\vec{x}; \vec{y})/p](\alpha(\vec{x}) \to \pi(\vec{x}))\Big) \to \Big(\forall p(\alpha(\vec{x}) \to \pi(\vec{x}))\Big),$$

the converse direction being an instantiation. So assume that $\mathfrak{A} = (A, T, \ldots) \models \alpha[\vec{u}]$; then, for some i we have that $(A, X_{\gamma_i}, \ldots) \models \alpha[\vec{u}]$, where X_{γ_i} is the subset of T defined by γ_i (notation as in 7.4.20). This implies $(A, X_{\gamma_i}, \ldots) \models [\lambda\vec{y}.\, \gamma_i(\vec{x}; \vec{y})/p]\alpha[\vec{u}]$, and by assumption, $(A, X_{\gamma_i}, \ldots) \models [\lambda\vec{y}.\, \gamma_i(\vec{x}; \vec{y})/p]\pi[\vec{u}]$; by monotonicity this gives$(A, T, \ldots) \models \pi[\vec{u}]$.

Next assume α is globally definably continuous with parameters. The instantiations we need are of the form $\lambda\vec{z}.\, \gamma(\vec{x}; \vec{y}; \vec{z})$, where γ is a p-free $\mathcal{L}$-formula, $\lambda\vec{z}.\, \vec{z} = \vec{z}$, and $\lambda\vec{z}.\, \vec{z} \neq \vec{z}$. Assume that p occurs both in α and in π, and let $\gamma_0(\vec{x}; \vec{y}; \vec{z}_0)$, ..., $\gamma_n(\vec{x}; \vec{y}; \vec{z}_n)$ be p-free formulas witnessing the continuity of α. Reasoning as before we find

$$\models \Big(\bigwedge_{0 \le i \le n} \forall \vec{z}_i\, [\lambda\vec{y}.\, \gamma_i(\vec{x}; \vec{y}; \vec{z}_i)/p](\alpha(\vec{x}) \to \pi(\vec{x}))\Big) \to \Big(\forall \vec{p}\, (\alpha(\vec{x}) \to \pi(\vec{x}))\Big).$$

This suffices. ⊣

EXAMPLE. As an example, consider the formula $\Box\Box p \to \Box p$ in $\mathcal{ML}(\Diamond)$.

- Higher-order equivalent:

$$\forall p \Big(\forall y\, (Rxy \to \forall z\, (Ryz \to p(z))) \to \forall v\, (Rxv \to p(v))\Big),$$

- after rewriting: $\forall p\, (\underline{\forall yz\, (Rxy \wedge Ryz \to p(z))} \to \forall v\, (Rxv \to p(v)))$; the underlined part is definably continuous with $\lambda u.\, R^2 xz$ as the p-free definition,
- substituting $\lambda u.\, R^2 xu$ for p the formula reduces to $\forall v\, (Rxv \to R^2 xu)$, i.e. R is dense.

EXAMPLE. In Blackburn & Spaan (1993)'s attribute value logic $\mathbf{L}^{KR[*]}$ with master modality $[*]$ one has models with a stock of binary relations R_l and $x \models [*]p$ iff for all y with $(x,y) \in (\bigcup_l R_l)^*$: $y \models \phi$. Consider the formula $\langle * \rangle [*]p \to p$.

- Higher-order equivalent:

$$\forall p \left(\exists y \, ((x,y) \in (\textstyle\bigcup_l R_l)^* \wedge \forall z \, ((y,z) \in (\bigcup_l R_l)^* \to p(z))) \to p(x) \right),$$

- after rewriting:

$$\forall p \forall y \left((x,y) \in (\textstyle\bigcup_l R_l)^* \wedge \underline{\forall z \, ((y,z) \in (\bigcup_l R_l)^* \to p(z))} \to p(x) \right),$$

  where the underlined formula is definably continuous in p with $\lambda u. \, ((y,u) \in (\bigcup_l R_l)^*)$ as the p-free definition,
- substituting $\lambda u. \, ((y,u) \in (\bigcup_l R_l)^*)$ for p gives

$$\forall y \left((x,y) \in (\textstyle\bigcup_l R_l)^* \to (y,x) \in (\bigcup_l R_l)^* \right).$$

EXAMPLE. Shehtman (1993) uses a progressive operator Π in addition to the usual temporal operators F, P to approximate the meaning of the English progressive: $x \models \Pi p$ iff

$$\exists x'x'' \, (Rx'x \wedge Rxx'' \wedge \forall z \, (Rx'z \wedge Rzx'' \to z \models p)).$$

Consider the formula $\Pi p \to Fp$.

- Higher-order translation:

$$\forall p \left(\exists x'x'' \, (Rx'x \wedge Rxx'' \wedge \forall z \, (Rx'z \wedge Rzx'' \to p(z))) \to \exists y \, (Rxy \wedge p(y)) \right),$$

- after rewriting:

$$\forall p \forall x'x'' \left(Rx'x \wedge Rxx'' \wedge \underline{\forall z \, (Rx'z \wedge Rzx'' \to p(z))} \to \exists y \, (Rxy \wedge p(y)) \right),$$

  which is a definably continuous formula with $\lambda u. \, (Rx'u \wedge Rux'')$ as its p-free definition,
- substituting $\lambda u. \, (Rx'u \wedge Rux'')$ for p reduces the formula to

$$\forall x'x'' \left(Rx'x \wedge Rxx'' \to \exists y \, (Rxy \wedge Rx'y \wedge Ryx'') \right).$$

Despite the somewhat baroque definition of definably continuous formulas, for the definably continuous first-order formulas an explicit syntactic characterization can be given. As in earlier cases a form of distributivity is needed.

7.4.24. DEFINITION. Let $\beta \in \mathrm{Form}_{\mathcal{L}_{\omega\omega}}[\tau]$, and let p be an s-variable. Then β is called *type 3 distributive in p* if it is a disjunction of formulas of the form $\forall \vec{y} \, (\beta'(\vec{y}) \to p(\vec{y})) \wedge \gamma$, where β' and γ are p-free formulas. Also, β is called *type 4 distributive in p* if it is a disjunction of formulas of the form $\exists \vec{z} \, (\gamma(\vec{x}; \vec{z}) \wedge \forall \vec{y} \, (\beta'(\vec{y}; \vec{z}) \to p(\vec{y})))$, with the same restrictions on β' and γ as before.

7.4.25. THEOREM. *Let $\beta \in \mathrm{Form}_{\mathcal{L}_{\omega\omega}}[\tau]$, and let p be an s-variable. Then*

1. *β is definably $\mathcal{L}_{\omega\omega}$-continuous in p iff it is equivalent to a formula that is type 3 distributive in p, and*
2. *β is definably $\mathcal{L}_{\omega\omega}$-continuous in p with parameters iff it is equivalent to a formula that is type 4 distributive in p.*

Proof. Use the proof of 7.4.22. ⊣

Observe that if α is type 3 or type 4 distributive, reductions of the kind described in Lemma 7.4.23 take their substitutions from antecedents β of the Horn-like conditions $\forall \vec{y}\,(\beta \rightarrow p)$ occurring in α; the only (possible) difference between the two is that if α is type 4 distributive, β is allowed to contain additional parameters.

To conclude this section, Table 7.1 summarizes the main points.

semantic property	substitutions needed	syntactic form
upward monotonicity	$\lambda\vec{x}.\,\vec{x} \neq \vec{x}$	positive occurrences only (7.4.5)
downward monotonicity	$\lambda\vec{x}.\,\vec{x} = \vec{x}$	negative occurrences only (7.4.5)
continuity	$\lambda\vec{x}.\,\vec{x} = \vec{y}$, $\lambda\vec{x}.\,\vec{x} \neq \vec{x}$ and $\lambda\vec{x}.\,\vec{x} = \vec{x}$	distributive (7.4.9)
ω-continuity	$\lambda\vec{x}.\,\bigvee_i(\vec{x} = \vec{y}_i)$, $\lambda\vec{x}.\,\vec{x} \neq \vec{x}$ and $\lambda\vec{x}.\,\vec{x} = \vec{x}$	ω-distributive (7.4.17)
definable continuity	$\lambda\vec{x}.\,\gamma(\vec{x};\vec{y})$ ('s-free')	type 3 distributive (7.4.24)
definable continuity with parameters	$\lambda\vec{x}.\,\gamma(\vec{x};\vec{y};\vec{z})$ ('s-free')	type 4 distributive (7.4.24)

Table 7.1: Forms of continuity.

7.5 REDUCTION ALGORITHMS

We put our findings of §7.4 to work. Our input consists of s-universal formulas β, and the aim is to reduce such formulas β to (combinations of) formulas of the form

$$\beta' \equiv \forall \vec{p}\,(\alpha \rightarrow \pi),$$

where α satisfies one of the distributivity conditions of §4 for all of its s-variables, and π is positive in all of its s-variables. Given the syntactic form of α the instantiations yielding the required reduction to an s-free formula can then be read of from β'.

There are several ways of rewriting β to β'. The earliest approaches are due to Sahlqvist (1975) and Van Benthem (1976, 1983). A recent version can be found in (Sambin & Vaccaro 1989). These approaches all deal with the uni-modal language with a single diamond $\Diamond$ and box $\Box$ only. They describe a fragment of this language, and show that all formulas in this fragment reduce to first-order formulas. In addition, Sahlqvist (1975) and Sambin & Vaccaro (1989) show that whenever the basic modal logic **K** is extended with axioms taken from this fragment, the resulting system is axiomatically complete. Kracht (1993) obtains those reducibility and completeness results in one go as part of a unifying approach towards definability in standard modal logic. Venema (1993) obtains a similar double result for certain modal languages containing a D-operator and irreflexivity rule. Finally, Gabbay & Ohlbach (1992) and Simmons (1992) extend the Sahlqvist-van Benthem algorithm by using Skolem functions; in addition, the latter considers modal languages with arbitrary many unary modal operators instead of a single one.

In this section we first describe the Sahlqvist-van Benthem algorithm extended to arbitrary languages; then its limitations are pointed out, and general strategies for *disproving* reducibility are sketched. Finally we show how the Gabbay-Ohlbach-Simmons approach overcomes some but not all of these limitations.

THE SAHLQVIST-VAN BENTHEM ALGORITHM

This is our strategy: we define a class of formulas Σ and show that every $\beta \in \Sigma$ can be rewritten to a combination of formulas of the form $\forall\vec{p}\forall\vec{y}\,(\alpha(\vec{x};\vec{y}) \to \pi(\vec{x};\vec{y}))$, where α is type 4 distributive in $\vec{p}$, and π is positive in $\vec{p}$. We then apply our results from §7.4 to show that β must be reducible.

For the remainder of this subsection we fix a classical vocabulary $\boldsymbol{\tau}$ and a sort s in $\boldsymbol{\tau}$.

7.5.1. DEFINITION. (Sahlqvist formulas) We say that $\beta \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$ is a *simple Sahlqvist formula for s* if it is an s-universal formula of the form $\forall\vec{p}\forall\vec{y}\,(\alpha(\vec{x};\vec{y}) \to \pi(\vec{x};\vec{y}))$, where α is type 4 distributive in all its s-variables, and π is positive in all its s-variables.

The Sahlqvist formulas for s are built up as follows. First, a formula β (not containing any quantifiers binding s-variables) is called an *s-block* if

- it is negative in all its s-variables, or
- it is type 4 distributive in all its s-variables, or
- it is s-free.

Next, *s-antecedents* are defined by the rule

$$\alpha ::= \beta \mid \alpha_1 \wedge \alpha_2 \mid \alpha_1 \vee \alpha_2 \mid \exists\vec{y}\,\alpha,$$

where β is an s-block. Finally, a *Sahlqvist formula* is an s-universal formula of the form $\forall\vec{p}\forall\vec{y}\,\gamma(\vec{x})$ where

$$\gamma ::= \forall\vec{u}\,(\alpha(\vec{u}) \to \pi(\vec{u})) \mid \forall\vec{u}\,(\delta(\vec{u}) \to \gamma(\vec{u})) \mid \gamma_1 \wedge \gamma_2 \mid \gamma_1 \vee \gamma_2, \tag{7.8}$$

where the formation of disjunctions is subject to the condition that γ_1, γ_2 share no s-variables and no individual variables except for $\vec{x}$, and where α is an s-antecedent, π is positive in all its s-variables, and δ is s-free.

What Definition 7.5.1 boils down to is that (modulo some 'extras') a Sahlqvist formula is a formula of the form $\forall\vec{p}\forall\vec{y}\,(\alpha \rightarrow \pi)$ where π is positive, and in α no $\exists$ or $\vee$ occurs in the scope of a $\forall$.

7.5.2. LEMMA. (Rewriting Lemma) *Let γ be a Sahlqvist formula of the form $\forall\vec{p}\forall\vec{y}\,(\alpha \rightarrow \pi)$ with α, π as in (7.8). Then γ is equivalent to a conjunction of simple Sahlqvist formulas.*

Proof. We first give an inductive recipe for rewriting conjunctions γ of Sahlqvist formulas of the form $\forall\vec{p}\forall\vec{y}\,(\alpha \rightarrow \pi)$ to conjunctions of the form

$$\bigwedge_i \forall\vec{p}_i\forall\vec{y}_i\left(\bigwedge_{j_i} B_{j_i} \rightarrow \pi_i\right), \tag{7.9}$$

where the B_i are s-blocks and the π_i are positive.

1. if $\forall\vec{p}\forall\vec{y}\,(\exists\vec{z}\,\alpha \rightarrow \pi)$ is a conjunct in γ, replace it with $\forall\vec{p}\forall\vec{y}\forall\vec{z}\,(\alpha \rightarrow \pi)$;
2. if $\forall\vec{p}\forall\vec{y}\,(\alpha_1 \vee \alpha_2 \rightarrow \pi)$ is a conjunct in γ, replace it with $\forall\vec{p}\forall\vec{y}\,(\alpha_1 \rightarrow \pi) \wedge \forall\vec{p}\forall\vec{y}\,(\alpha_2 \rightarrow \pi)$;
3. if $\forall\vec{p}\forall\vec{y}\,(\alpha_1 \wedge \exists\vec{z}\,\alpha_2 \rightarrow \pi)$ is a conjunct in γ, replace it with $\forall\vec{p}\forall\vec{y}\forall\vec{z}\,(\alpha_1 \wedge \alpha_2 \rightarrow \pi)$;
4. if $\forall\vec{p}\forall\vec{y}\,(\alpha_1 \wedge (\alpha_2 \vee \alpha_3) \rightarrow \pi)$ is a conjunct in γ, replace it with $\forall\vec{p}\forall\vec{y}\,((\alpha_1 \wedge \alpha_2) \vee (\alpha_1 \wedge \alpha_3) \rightarrow \pi)$.

Clearly, every conjunct in γ is equivalent to a formula of the form occurring in the antecedent of 1–4. It is also clear that the output of this rewriting recipe has the form described in (7.9).

Next we show how conjunctions γ of the form (7.9) can be rewritten to simple Sahlqvist formulas. Take any conjunct in γ; it may be assumed to have the form

$$\forall\vec{p}\forall\vec{y}\,(D \wedge N \wedge F \rightarrow \pi), \tag{7.10}$$

where D is a conjunction of type 4 distributive formulas, N is a conjunction of negative formulas, and F is a conjunction of s-free formulas. Now (7.10) is equivalent to

$$\forall\vec{p}\forall\vec{y}\,(D \rightarrow \pi \vee \neg N \vee \neg F). \tag{7.11}$$

This is a simple Sahlqvist formula, as $\pi \vee \neg N \vee \neg F$ is positive in all its s-variables. Repeating the procedure for all conjuncts in γ completes the proof. $\dashv$

7.5.3. THEOREM. (The Sahlqvist-van Benthem Algorithm) *Let $\beta(\vec{x})$ be (equivalent to) a Sahlqvist formula for s. Then $\beta(\vec{x})$ reduces to an s-free formula via suitable instantiations. Moreover, these instantiations can be effectively obtained from $\beta(\vec{x})$.*

Proof. We first prove the result for conjunctions of simple Sahlqvist formulas. Let $\forall\vec{p}\forall\vec{y}\,(\alpha \to \pi)$ be such a formula. It is equivalent to a conjunction of formulas of the form

$$\forall\vec{p}\forall\vec{y}\,(D \to \pi'), \tag{7.12}$$

where π' is positive, and D is type 4 distributive in all s-variables. By Lemma 7.4.23 (7.12) reduces to an s-free formula via substitutions that can be read of from D. By Lemma 7.3.6 the conjunction of reducible formulas is also reducible.

Next, if $\forall\vec{p}\forall\vec{y}\,\gamma$ is a Sahlqvist formula for s, reducibility is obtained by an inductive argument.

- First, Sahlqvist formulas of the form $\forall\vec{p}\forall\vec{y}\,(\alpha \to \pi)$ are equivalent to conjunctions of simple Sahlqvist formulas, by the Rewriting Lemma; hence it is reducible to an s-free formula by the first half of the proof.
- If $\forall\vec{p}\forall\vec{y}\,\gamma$ is of the form $\forall\vec{p}\forall\vec{y}\,(\gamma_1 \wedge \gamma_2)$, it rewrites to $\forall\vec{p}\forall\vec{y}\,\gamma_1 \wedge \forall\vec{p}\forall\vec{y}\,\gamma_2$; the latter reduces to an s-free formula whenever both conjuncts $\forall\vec{p}\forall\vec{y}\,\gamma_1$ and $\forall\vec{p}\forall\vec{y}\,\gamma_2$ do so (by Lemma 7.3.6).
- If $\forall\vec{p}\forall\vec{y}\,\gamma$ is of the form $\forall\vec{p}\forall\vec{y}\vec{z}\,(\gamma_1 \vee \gamma_2)$, it rewrites to $\forall\vec{p}\forall\vec{y}\,\gamma_1 \vee \forall\vec{p}\vec{z}\,\gamma_2$ as only formulas not sharing any bound variables are disjoined; the latter reduces to an s-free formula iff both disjuncts do (Lemma 7.3.6).
- If $\forall\vec{p}\forall\vec{y}\,\gamma_1$ is of the form $\forall\vec{p}\forall\vec{y}\,(\delta(\vec{x};\vec{y}) \to \pi)$; the latter reduces to an s-free formula iff $\forall\vec{p}\forall\vec{y}\,\gamma_1(\vec{y})$ does (Lemma 7.3.7). $\dashv$

7.5.4. REMARK. To recap, the strategy in Theorem 7.5.3 is to obtain reductions through instantiations. These instances are found by carefully rewriting Sahlqvist formulas into certain combinations of *simple* Sahlqvist formulas $\forall\vec{p}\forall\vec{y}\,(\alpha \to \pi)$, and then simply reading them of from the antecedents α. Detailed examples are provided in §7.6 below.

Theorem 7.5.3 takes type 4 distributive formulas as its basic building blocks supplying the required instantiations. Scaled-down analogues of the Sahlqvist Theorem may be obtained by taking one of the other syntactic forms occurring in Table 7.1, as the basic building blocks.

LIMITATIONS OF THE SAHLQVIST-VAN BENTHEM ALGORITHM

Formulas that are typically excluded from the set of Sahlqvist formulas have implications $\alpha \to \pi$ as their matrix with α containing a $\forall\exists$ or $\forall(\ldots \vee \ldots)$-combination. Van Benthem (1983) shows that these limitations occur even in the weakest language we consider here, $\mathcal{ML}(\Diamond)$. Below I will repeat one case (in $\mathcal{ML}(\Diamond)$) of non-reducibility due to a forbidden $\forall\exists$-combination. By way of examples I will show how this case may be used to obtain further *non*-reducibility results for arbitrary modal formulas with first-order definable truth definitions, that contain a forbidden quantification of the form $\forall\exists$.

7.5.5. PROPOSITION. (Van Benthem (1983)) *The (translation of the) McKinsey formula $\Box\Diamond p \to \Diamond\Box p$ does not reduce to a p-free formula over R, $=$.*

Proof. The higher-order translation of the McKinsey formula reads

$$\forall p\left(\forall y\exists z\,(Rxy \to (Ryz \wedge p(z))) \to \exists y'\forall z'\,(Rxy' \wedge (Ry'z' \to p(z')))\right). \quad (7.13)$$

Non-reducibility is proved by showing that it lacks a first-order equivalent over R, $=$. To this end we show that it does not enjoy the Löwenheim-Skolem property. Consider the frame $\mathfrak{F} = (W, R)$, where

- $W = \{ a, b_n, b_{n_i} : n \in \mathbb{N}, i \in \{0,1\}\} \cup \{ c_f : f : \mathbb{N} \to \{0,1\}\}$,
- $R = \{ (a, b_n), (b_n, b_{n_i}), (b_{n_i}, b_{n_i}) : n \in \mathbb{N}, i \in \{0,1\}\} \cup \{ (a, c_f), (c_f, b_{n_{f(n)}}) : f : \mathbb{N} \to \{0,1\}\}$.

It may be shown that $\mathfrak{F}, a \models (7.13)$. Take a countable elementary subframe $\mathfrak{F}'$ of $\mathfrak{F}$ containing a and all b_n, b_{n_i}. For some $f : \mathbb{N} \to \{0,1\}$, f is not in $\mathfrak{F}'$ (as $\mathfrak{F}'$ is countable). This f may be used to refute (7.13) at a in $\mathfrak{F}'$. Hence (7.13) lacks a first-order equivalent. $\dashv$

Now, the strategy for porting the above non-reducibility to arbitrary modal languages in which all operators have first-order definable patterns, is to code formulas with forbidden quantifier patterns up into the above example 7.5.5 using first-order means. Here is an example taken from unary interpretability logic (De Rijke 1992*d*). The latter extends provability logic with an operator **I** used to simulate the notion of relative interpretability over a given base theory. The semantics of **I** is based on a binary relation R and a ternary relation S as follows:

$$(W, R, S, V), x \models \mathbf{I}p \text{ iff } \forall y\,(Rxy \to \exists z\,(Sxyz \wedge z \models p)).$$

Consider the formula $\mathbf{I}p \to \neg\mathbf{I}\neg p$ whose classical equivalent on frames reads

$$\forall p\left(\forall y\exists z\,(Rxy \to Sxyz \wedge p(z)) \to \exists y'\forall z'\,(Rxy' \wedge (Sxy'z' \to p(z')))\right), \quad (7.14)$$

which is of the form $\forall p\,(\alpha \to \pi)$ with π positive (in p), and α containing a $\forall\exists$-combination.

7.5.6. PROPOSITION. *The formula (7.14) does not reduce to a p-free formula over R, S.*

Proof. Let $\mathfrak{G} = (W, R, S)$ where $\mathfrak{F} = (W, R)$ is as in the proof of Proposition 7.5.5, and S is defined by

- $\forall xyz\,(Sxyz \leftrightarrow (Rxy \wedge Ryz))$.

Then $\mathfrak{G}, a \models (7.14) \leftrightarrow (7.13)$. Hence $\mathfrak{G} \models (7.14)$. But for $\mathfrak{G}' = (\mathfrak{F}', S')$ with $\mathfrak{F}'$ as in the proof of Proposition 7.5.5, and S' defined like S above, we must have $\mathfrak{G}' \not\models (7.14)$, for otherwise $\mathfrak{F}' \models (7.13)$. $\dashv$

The same strategy shows non-reducibility results for (classical equivalents of) formulas involving $\forall\exists$-combinations in many other modal languages, with *Until*, *Since*-logic as an obvious example.

As to the second kind of forbidden combinations mentioned earlier, viz. configurations $\forall(\ldots\vee\ldots)$, Van Benthem (1983) gives a non-reducible formula in $\mathcal{ML}(\Diamond)$ whose higher-order equivalent $\forall p\,(\alpha\to\pi)$ contains such a combination in its antecedent α. Analogous to the above case of $\forall\exists$ this example may be used as a tool for establishing non-reducibility results for 'forbidden formulas' in arbitrary modal languages in which all patterns are first-order definable.

THE GABBAY-OHLBACH-SIMMONS ALGORITHM

Unlike the Sahlqvist-van Benthem algorithm the Gabbay-Ohlbach-Simmons algorithm is able to deal with some cases like (7.14). By Proposition 7.5.6 the Gabbay-Ohlbach-Simmons algorithm cannot reduce (7.14) to a first-order formula involving only R and S (assuming the algorithm is sound). To arrive at a p-free equivalent it uses quantification over Skolem functions. Here is an example. Consider (7.13) again:

$$\forall p\left(\forall y\exists z\,(Rxy\to Ryz\wedge p(z))\to\exists y'\forall z'\,(Rxy'\wedge(Ry'z'\to p(z')))\right).$$

The antecedent of the matrix of (7.13), $\forall y\exists z\,(Rxy\to Ryz\wedge p(z))$ is equivalent to

$$\exists f\forall y\,(Rxy\to Ryf(x,y)\wedge p(f(x,y)).$$

Thus (7.13) is equivalent to

$$\forall p\forall f\Big(\forall y\,(Rxy\to Ryf(x,y)\wedge p(f(x,y)))\to \\ \exists y'\forall z'\,(Rxy'\wedge(Ry'z'\to p(z')))\Big).$$

Substituting $\lambda u.\,\exists z\,(Rxz\wedge u=f(x,z))$ for p in the above gives

$$\forall f\Big(\forall y\,(Rxy\to Ryf(x,y))\to \\ \exists y'\forall z'\,(Rxy'\wedge(Ry'z'\to\exists v\,(Rxv\wedge z'=f(x,v))))\Big). \tag{7.15}$$

A remark is in order: (7.15) replaces a quantification over unary predicates in (7.14) with quantification over functions — what has been gained? Besides revealing a link between different fragments of classical logic that may in itself be of logical interest, such replacements are computationally relevant, as is shown by Gabbay & Ohlbach (1992).

We now present the Gabbay-Ohlbach-Simmons algorithm in analogy with the Sahlqvist-van Benthem algorithm. First, we need a set of formulas for the algorithm to operate on.

7.5.7. DEFINITION. (Extended Sahlqvist formulas) We work over a vocabulary with function symbols. The type 4 distributive formulas over this vocabulary are defined as in Definition 7.4.24 — where the arguments of p may now involve function symbols. From these, simple Sahlqvist formulas and s-blocks are defined as in Definition 7.5.1.

To define extended s-antecedents α we consider an intermediate set of formulas α' generated by

$$\alpha' ::= \beta' \mid \alpha_1' \wedge \alpha_2' \mid \exists \vec{y}\, \alpha' \mid \forall \vec{y}\, \alpha',$$

with β' an s-block such that if β' is a type 4 distributive formula, then it should be of the form $\exists \vec{z}\, (\gamma \wedge \forall \vec{y}\, (\beta \rightarrow p))$. Then, the *extended s-antecedents* are generated by the rule

$$\alpha ::= \beta \mid \alpha' \mid \alpha_1 \wedge \alpha_2 \mid \alpha_1 \vee \alpha_2 \mid \exists \vec{y}\, \alpha,$$

where β is an s-block. The important restriction here is that no $\forall$ governs a $\vee$. Finally, *extended Sahlqvist formulas* are generated using extended s-antecedents analogous to (7.8).

For the poly-modal language $\mathcal{ML}(\langle a \rangle : a \in A)$ the above definition specifies the same fragment as the one given by (Simmons 1992). The proof of this claim would require a lengthy and boring induction, and is therefore omitted.

The Gabbay-Ohlbach-Simmons algorithm extends the Sahlqvist-van Benthem algorithm. First there is an Extended Rewriting Lemma.

7.5.8. LEMMA. (Extended Rewriting Lemma) *Any extended Sahlqvist formula of the form $\forall \vec{p} \forall \vec{f} \forall \vec{y}\, (\alpha \rightarrow \pi)$ with α an extended s-antecedent and π positive, is equivalent to a conjunction of (almost simple) Sahlqvist formulas of the form*

$$\forall \vec{p} \forall \vec{f} \forall \vec{y}\, (\bigwedge_i D_i \rightarrow \pi), \tag{7.16}$$

with $\bigwedge_i D_i$ a conjunction of type 4 distributive formulas.

Proof. This is similar to the proof of Lemma 7.5.2. First we rewrite to a formula as in (7.16), but with a conjunction of s-blocks in antecedent position rather than distributive formulas. The following rewrite instructions need to be added to the stock in 7.5.2; their purpose is to move quantifications over functions to the prefix, and to push occurrences of $\forall$ inside as far as possible until they 'reach' a distributive formula that doesn't start with a $\exists$-prefix — without breaking down negative formulas or s-free formulas.

5. if $\forall \vec{p} \forall \vec{f} \forall \vec{y}\, (\ldots \forall \vec{z}\, (\alpha_1 \wedge \alpha_2) \ldots \rightarrow \pi)$ is a conjunct in γ, replace it with

$$\forall \vec{p} \forall \vec{f} \forall \vec{y} \Big(\ldots \forall \vec{z}\, \alpha_1 \wedge \forall \vec{z}\, \alpha_2 \ldots \rightarrow \pi\Big);$$

6. if $\forall \vec{p} \forall \vec{f} \forall \vec{y}\, (\ldots \forall \vec{z} \exists u_1 \ldots u_n\, \alpha \ldots \rightarrow \pi)$ is a conjunct in γ, replace it with

$$\forall \vec{p} \forall \vec{f} \forall \vec{y} \Big(\ldots \exists f_1 \ldots f_n \forall \vec{z}\, [f_1(\vec{y}, \vec{z})/u_1] \ldots [f_n(\vec{y}, \vec{z})/u_n]\, \alpha \ldots \rightarrow \pi\Big),$$

for fresh function symbols $f_1, \ldots, f_n$;

7. if $\forall\vec{p}\forall\vec{f}\forall\vec{y}\,(\alpha_1 \wedge \exists g\, \alpha_2 \rightarrow \pi)$ is a conjunct in γ, replace it with $\forall\vec{p}\forall\vec{f}g\forall\vec{y}\,(\alpha_1 \wedge \alpha_2 \rightarrow \pi)$;
8. if $\forall\vec{p}\forall\vec{f}\forall\vec{y}\,(\ldots\forall\vec{z}\,(\delta \rightarrow \theta \wedge p)\ldots \rightarrow \pi)$ is a conjunct in γ, replace it with
$$\forall\vec{p}\forall\vec{f}\forall\vec{y}\,\Big(\ldots\forall\vec{z}\,(\delta \rightarrow \theta) \wedge \forall\vec{z}\,(\delta \rightarrow p)\ldots \rightarrow \pi\Big).$$

The second half of the proof is similar to the second half of 7.5.2. ⊣

7.5.9. THEOREM. (The Gabbay-Ohlbach-Simmons algorithm) *Let $\boldsymbol{\tau}$ be a vocabulary with sufficiently many function symbols, and s a sort in $\boldsymbol{\tau}$. Let $\beta(\vec{x})$ be (equivalent to) an extended Sahlqvist formula for s. Then $\beta(\vec{x})$ reduces to an s-free formula, possibly involving additional function symbols, via suitable instantiations. These instantiations can be effectively obtained from β.*

Proof. This is almost the same as the proof of 7.5.3; the substitutions arising from distributive formulas involving function symbols of the form $\forall\vec{y}\,(\gamma(\vec{x};\vec{y}) \rightarrow p(f(\vec{x};\vec{y})))$ are $\lambda\vec{u}.\,\exists\vec{y}\,(\gamma(\vec{x};\vec{y}) \wedge \vec{u} = f(\vec{x};\vec{y}))$. ⊣

LIMITATIONS OF THE GABBAY-OHLBACH-SIMMONS ALGORITHM

The main gain of extended Sahlqvist formulas over Sahlqvist formulas is that the former allow $\forall\exists$-combinations. However the extended Sahlqvist formulas still suffer from the restriction on $\forall(\ldots\vee\ldots)$-combinations. The importance of the restriction is best explained by an example. Consider Löb's formula in $\mathcal{ML}(\Diamond)$ $\Box(\Box p \rightarrow p) \rightarrow \Box p$, which translates into
$$\forall p\,\Big(\forall y\,(Rxy \rightarrow \exists z\,(Rxz \wedge \neg p(z)) \vee p(y)) \rightarrow \forall u\,(Rxu \rightarrow p(u))\Big)$$
on frames. After Skolemization and rewriting this gives
$$\forall p\forall f\exists y\,\Big((Rxy \rightarrow (Rxf(x,y)\wedge\neg p(f(x,y)))\vee p(y)) \rightarrow \forall u\,(Rxu \rightarrow p(u))\Big). \quad (7.17)$$
At this point we need to define a substitution to achieve a reduction to a p-free formula. However, there is no obvious candidate — because of the disjunction occurring in the antecedent of (7.17). It seems that to be able to handle cases such as the Löb formula, higher-order functions are needed, ones that take infinite sequences, or even whole 'R-trees' as arguments. On the other hand, it may be that the Löb formula is not expressible in the Gabbay-Ohlbach-Simmons fragment. I will leave this for further study.

7.6 APPLYING THE ALGORITHMS

Section 7.5 presented the general Sahlqvist-van Benthem and Gabbay-Ohlbach-Simmons algorithms for obtaining reducibility results. To actually apply them to individual modal languages requires a further detailed analysis of those languages to locate the Sahlqvist fragments. Below we illustrate this by examining the languages of standard modal logic, D-logic, *Since*, *Until*-logic, as well as the language of Peirce algebras, and infinitary modal languages. Finally, applications are given to areas other than modal logic, including circumscription.

STANDARD MODAL LOGIC

Formulas of the standard modal language $\mathcal{ML}(\Diamond)$ translate into a strict subset of the language of monadic second-order logic. Its Sahlqvist fragment is a strict subset of the general Sahlqvist fragment of the latter (7.5.1). To be precise, let the set of Sahlqvist formulas $\mathcal{SF}(\Diamond) \subseteq \mathcal{ML}(\Diamond)$ be defined by putting $\chi \in \mathcal{SF}(\Diamond)$ iff it is produced by the following rules

- $\phi ::= \Box^i p \mid \nu \mid \delta$, where ν is negative in all proposition letters occurring in it, and δ is p-free,
- $\psi ::= \phi \mid \psi_1 \wedge \psi_2 \mid \psi_1 \vee \psi_2 \mid \Diamond\psi$,
- $\chi ::= \psi \to \pi \mid \chi_1 \wedge \chi_2 \mid \chi_1 \vee \chi_2 \mid \Box\chi$, where π is positive in all its proposition letters, and $\vee$ is applied only to formulas χ_1, χ_2 that don't share proposition letters.

When interpreted on frames every $\chi \in \mathcal{SF}(\Diamond)$ translates into a Sahlqvist formula over a vocabulary with a single binary relation symbol, and unary predicate variables corresponding to the proposition letters in $\mathcal{ML}(\Diamond)$. By Theorem 7.5.3 every element of the Sahlqvist fragment $\mathcal{SF}(\Diamond)$ reduces to an s-free formula.

The set of instances needed to reduce every formula in $\mathcal{SF}(\Diamond)$ is an atomic join semi-lattice with partial operators, the atoms being the terms denoting singletons, and the operators correspond to necessitation and are defined on $\vee$-free terms only.

Now that we are considering individual modal languages, much more fine-grained issues become visible than in our general analysis of §§7.4, 7.5. As an example, given the Sahlqvist fragment $\mathcal{SF}(\Diamond)$ one may strive for an explicit syntactic description.

7.6.1. DEFINITION. (Kracht 1993) An individual variable v is called *inherently universal in* α if either it is free in α, or α is of the form $\forall x\,(Rxy \to \beta)$ and v is inherently universal in β. *Inherently existential* is defined similarly. A first-order formula α is *restricted* if it is built using only restricted quantifiers $\forall v\,(Rxv \to \ldots)$ and $\exists v\,(Rxv \wedge \ldots)$.

A *Sahlqvist reduct* is a first-order formula over a binary relation symbol R and $=$ that is equivalent to a positive, restricted formula in which every subformula $R^i yz$ contains at least one inherently universal variable.

7.6.2. THEOREM. *A first-order formula is definable by means of a Sahlqvist formula in the standard modal language* $\mathcal{ML}(\Diamond)$ *iff it is a Sahlqvist reduct.*

Proof. One direction follows from Theorem 7.5.3. The other one involves a simple but long case analysis which is too lengthy to be included here. Instead we give an example. Consider the formula

$$\exists z\,(Rxz \wedge \forall y\,(R^2 zy \to Rxy) \wedge Rzx). \tag{7.18}$$

The idea is to view (7.18) as being the result of certain substitutions into the translation of a positive modal formula π, to extract those substitutions from (7.18), and to prefix their modal counterparts as a Sahlqvist antecedent to π. Here we go:

1. the restricted quantification $\exists z\,(Rxz \wedge \ldots$ stems from a diamond $\Diamond$: $\Diamond(\ldots$;
2. the conjunct Rzx refers back to x, thus calling for a proposition letter p to be true at x, and z 'seeing' p: $p \rightarrow \Diamond(\Diamond p \wedge \ldots$;
3. finally, in $\forall y\,(R^2zy \rightarrow Rxy)$ the antecedent calls for 2 boxes $\Box$, and the consequent refers to 'being a successor of x' which calls for a boxed proposition letter being true at x: $p \wedge \Box q \rightarrow \Diamond(\Diamond p \wedge \Box\Box q)$. ⊣

Two short remarks: a similar syntactic analysis can be given for the *extended* Sahlqvist formulas as well (Definition 7.5.7); and recently Hans-Joachim Ohlbach has announced general results on associating modal equivalents to first-order formulas.

Next, as an application of the Gabbay-Ohlbach-Simmons algorithm in the standard modal language, we show that any modal reduction principle reduces to a p-free formula. First, a *modal reduction principle* in $\mathcal{ML}(\Diamond)$ (mrp) is a modal formula of the form $¥p \rightarrow \$p$, where ¥, \$ are (possibly empty) sequences of modal operators $\Diamond$ and $\Box$.

7.6.3. THEOREM. *The Gabbay-Ohlbach-Simmons algorithm reduces every modal reduction principle* $¥p \rightarrow \$p$ *to a* p*-free formula.*

Proof. Van Benthem (1983, Theorem 10.8) fully classifies the mrp's that reduce to a p-free formula by means of the Sahlqvist-van Benthem algorithm. From this result it follows that the use of additional function symbols (as in the Gabbay-Ohlbach-Simmons algorithm) is essential.

To prove the theorem it suffices to observe that every mrp translates into an extended Sahlqvist formula over R, $=$. To get some feel as to how an arbitrary mrp is reduced to a p-free formula, it may be instructive to go over the McKinsey axiom $\Box\Diamond p \rightarrow \Diamond\Box p$ and its higher-order translation (7.13) again. ⊣

D-LOGIC

We describe the Sahlqvist fragment $\mathcal{SF}(\Diamond, D)$ of the modal language $\mathcal{ML}(\Diamond, D)$ studied in Chapter 3. Put $\chi \in \mathcal{SF}(\Diamond, D)$ if it is produced by the following rules:

- $\phi ::= \overline{\#}_1 \ldots \overline{\#}_n p \mid \nu \mid \delta$, where $\overline{\#}_i \in \{\Box, \overline{D}\}$, ν is negative in all its proposition letters, and δ is p-free,
- $\psi ::= \phi \mid \psi_1 \wedge \psi_2 \mid \psi_1 \vee \psi_2 \mid \#\phi$, where $\# \in \{\Diamond, D\}$,
- $\chi ::= \psi \rightarrow \pi \mid \chi_1 \wedge \chi_2 \mid \chi_1 \vee \chi_2 \mid \overline{\#}\chi$, where π is positive in all its proposition letters, $\vee$ is applied only to formulas χ_1, χ_2 having no proposition letters in common, and $\overline{\#} \in \{\Box, \overline{D}\}$.

Here are examples (taken from Chapter 3) of Sahlqvist formulas in $\mathcal{ML}(\Diamond, D)$ and $\mathcal{ML}(F, P, D)$ plus their reductions to first-order conditions.

EXAMPLE. Consider the $\mathcal{ML}(\Diamond, D)$-formula $\Diamond p \rightarrow Dp$.

- Second-order translation: $\forall p\,(\exists y\,(Rxy \wedge p(y)) \rightarrow \exists z\,(x \neq z \wedge p(z)))$,
- after rewriting: $\forall p \forall y\,(Rxy \wedge \underline{p(y)} \rightarrow \exists z\,(x \neq z \wedge p(z)))$,

- substituting $\lambda u.\, u = y$ for p reduces this to $\forall y\,(Rxy \rightarrow \exists z\,(x \neq z \wedge z = y))$, or $\forall y\,(Rxy \rightarrow x \neq y)$, or $\neg Rxx$.

EXAMPLE. A slightly more complex example: $p \wedge \neg Dp \rightarrow A\neg\Diamond p$, or equivalently, $p \wedge E\Diamond p \rightarrow Dp$.

- Second-order translation:

$$\forall p\,\Big(p(x) \wedge \exists yz\,(Ryz \wedge p(z)) \rightarrow \exists v\,(v \neq x \wedge p(v))\Big),$$

- after rewriting: $\forall p \forall yz\,(\underline{p(x)} \wedge Ryz \wedge \underline{p(z)} \rightarrow \exists v\,(v \neq x \wedge p(v)))$,
- substituting $\lambda u.\,(u = x \vee u = z)$ for p reduces this to

$$\forall yz(Ryz \rightarrow \exists v\,(v \neq x \wedge (v = x \vee v = z))),$$

or $\neg\exists y\,(Ryx)$.

EXAMPLE. As a final example in $\mathcal{ML}(F, P, D)$, consider $Gp \vee Hp \rightarrow \overline{D}p$.

- Higher-order equivalent:

$$\forall p\,\Big(\underline{\forall y\,(Rxy \rightarrow p(y))} \wedge \underline{\forall y\,(Ryx \rightarrow p(y))} \rightarrow \forall y\,(y \neq x \rightarrow p(y))\Big),$$

- substituting $\lambda u.\,(Rxu \vee Rux)$ for p reduces this to $\forall y\,(x \neq y \rightarrow Rxy \vee Ryx)$.

Until, *Since*-LOGIC

The above definition of the Sahlqvist fragment $\mathcal{SF}(\Diamond)$ of $\mathcal{ML}(\Diamond)$ can easily be extended to the language $\mathcal{ML}(F, P)$ of temporal logic with the operators F and P. But the more powerful binary modal operators *Until* (whose pattern reads: $\lambda pq.\, \exists y\,(Rxy \wedge p(y) \wedge \forall z\,(Rxz \wedge Rzy \rightarrow q(z))))$ and *Since* ($\lambda pq.\, \exists y\,(Ryx \wedge p(y) \wedge \forall z\,(Ryz \wedge Rzx \rightarrow q(z))))$ can also be accommodated. To define a Sahlqvist fragment $\mathcal{SF}(\mathit{Until}, \mathit{Since})$ of the modal language with *Until*, *Since*, recall that both F and P are definable using *Until*, *Since*. Let $\#$ range over F, P, and $\overline{\#}$ over G, H. Put $\chi \in \mathcal{SF}(\mathit{Until}, \mathit{Since})$ if it is produced by the following rules

- $\phi ::= \overline{\#}_1 \ldots \overline{\#}_n p \mid \nu \mid \delta$, where ν is negative in all its proposition letters, and δ is p-free,
- $\psi ::= \phi \mid \psi_1 \wedge \psi_2 \mid \psi_1 \vee \psi_2 \mid \#\psi \mid \mathit{Until}(\#_1 \ldots \#_n\psi, \overline{\#}_1 \ldots \overline{\#}_m p) \mid \mathit{Since}(\#_1 \ldots \#_n\psi, \overline{\#}_1 \ldots \overline{\#}_m p)$,
- $\chi ::= \psi \rightarrow \pi \mid \chi_1 \wedge \chi_2 \mid \chi_1 \vee \chi_2 \mid \overline{\#}\chi$, where π is positive in all its proposition letters, and $\vee$ is applied only to formulas χ_1, χ_2 having no proposition letters in common.

All formulas in $\mathcal{SF}(\mathit{Until}, \mathit{Since})$ translate into Sahlqvist formulas over R and $=$; in particular, the 'between-ness' property $\exists y\,(Rxy \wedge p(y) \wedge \forall z\,(Rxz \wedge Rzy \rightarrow q(z)))$ itself is distributive in p and type 4 distributive in q. Thus, by Theorem 7.5.3, every formula in $\mathcal{SF}(\mathit{Until}, \mathit{Since})$ reduces to a first-order formula.

EXAMPLE. Consider the formula $Fp \rightarrow \mathit{Until}(p, q)$.

– Higher-order equivalent:

$$\forall pq\left(\exists y\,(Rxy \wedge p(y)) \to \exists y'\,(Rxy' \wedge p(y') \wedge \forall z'\,(Rxz' \wedge Rz'y' \to q(z')))\right),$$

– after rewriting:

$$\forall pq\forall y\left(Rxy \wedge \underline{p(y)} \to \exists y'\,(Rxy' \wedge p(y') \wedge \forall z'\,(Rxz' \wedge Rz'y' \to q(z')))\right),$$

– substituting $\lambda u.\, u = y$ for p, and $\lambda u.\, u \neq u$ for q gives $\forall y\,(Rxy \to \neg\exists z\,(Rxz \wedge Rzy))$.

THE LOGIC OF PEIRCE ALGEBRAS

In the modal language $\mathcal{ML}_2$ for reasoning about Peirce-like frames (Definition 5.5.3) we have two sorts of modal formulas: set formulas and relation formulas. Nothing prevents us from applying our Sahlqvist machinery here. As an example, consider axiom (MP2) from Definition 5.5.1: $\langle a \circ b\rangle p \to \langle a\rangle\langle b\rangle p$.

– Higher-order translation:

$$\begin{aligned}\forall ab\forall p\,\Big(\exists y_r y'_r y''_r z_s\,\left(Px_s y_r z_s \wedge Cy_r y'_r y''_r \wedge a(y'_r) \wedge b(y''_r) \wedge p(z_s)\right) \to\\ \exists v_r v'_r v''_r z_s\,\left(Px_s v_r z'_s \wedge Pv_r v'_r v''_r \wedge a(v'_r) \wedge b(v''_r) \wedge p(z'_s)\right)\Big),\end{aligned}$$

– after rewriting:

$$\begin{aligned}\forall ab\forall p\forall y_r y'_r y''_r z_s\,\Big(Px_s y_r z_s \wedge Cy_r y'_r y''_r \wedge \underline{a(y'_r)} \wedge \underline{b(y''_r)} \wedge \underline{p(z_s)} \to\\ \exists v_r v'_r v''_r z_s\,(Px_s v_r z'_s \wedge Pv_r v'_r v''_r \wedge a(v'_r) \wedge b(v''_r) \wedge p(z'_s))\Big),\end{aligned}$$

– substituting $\lambda u_r.\, u_r = y'_r$ for a, $\lambda u_r.\, u_r = y''_r$ for b, and $\lambda u_s.\, u_s = z_s$ for p reduces this to (CP2):

$$\forall y_r y'_r y''_r z_s\left(Px_s y_r z_s \wedge Cy_r y'_r y''_r \to \exists z'_s\,(Px_s y'_s z'_s \wedge Pz'_s y''_r z_s)\right).$$

INFINITARY MODAL LOGIC

So far we have applied our methods mainly to modal logics whose operators have first-order patterns. But they can be applied equally well beyond the first-order realm. For instance, they are easily extended to infinitary modal languages such as **PDL**, where one has multiple diamonds $\langle a\rangle$ as well as composition $\langle a;b\rangle$, union $\langle a \cup b\rangle$ and iteration $\langle a^*\rangle$. Because of the Kleene star * **PDL** translates into a fragment of $\mathcal{L}_{\omega_1\omega}$, and on frames into Π^1_1-conditions over $\mathcal{L}_{\omega_1\omega}$. A Sahlqvist fragment for **PDL** is easily defined, resulting in a set of **PDL**-formulas whose $\Pi^1_1(\mathcal{L}_{\omega_1\omega})$-equivalent reduces to a $\mathcal{L}_{\omega_1\omega}$-formula over R_a, ... and $=$. Here is an example: $[a^*](\langle b\rangle p \to \langle a^*\rangle p)$.

– Higher-order translation:

$$\forall p \forall y \left(\bigvee_n (R_a^n xy) \to \left(\exists z \, (R_b yz \wedge p(z) \to \exists v \, \bigvee_n (R_a^n yv) \wedge p(v)) \right) \right),$$

– after rewriting:

$$\forall p \forall yz \left(\bigvee_n (R_a^n xy) \wedge R_b yz \wedge \underline{p(z)} \to \exists v \, \bigvee_n (R_a^n yv) \wedge p(v) \right),$$

– substituting $\lambda u.\, u = z$ for p reduces this to

$$\forall yz \left(\bigvee_n (R_a^n xy) \wedge R_b yz \to \bigvee_n (R_a^n yz) \right).$$

Our methods apply equally well to modal languages with more explicitly infinitary constructs, such as arbitrary disjunctions and conjunctions, as in the infinitary basic modal languages $\mathcal{BML}(\boldsymbol{\tau})$ of Chapter 6. I invite the reader to think up examples for himself.

Beyond modal logic

Applications of 7.5.3 outside the field of modal logic are easily found. Here are some examples.

First, 7.5.3 provides us with a scheme for reducing a large class of Π_1^{n+1}-formulas to n-th order formulas. To see this, assume that we are working in a fragment without $(n+1)$-st order constant symbols, let s be a sort containing *all* $(n+1)$-st order variables, and let X be any set of s-free formulas. Then, if χ is an $(n+1)$-st order formulas that is in fact a Sahlqvist formula for s, χ reduces to an s-free formula, i.e. to an n-th order formula.

Second, the Sahlqvist machinery may be used to remove *sorts* from a many-sorted (first-order) theory Δ. Let s be a sort in the language of Δ. If all of Δs axioms are Sahlqvist formulas for s, then Δ has an axiomatization using s-free formulas only — by Theorem 7.5.3.

Third, recall that *circumscription* is the minimization of predicates subject to restrictions expressed by first-order formulas that is proposed for the purpose of formalizing non-monotonic aspects of common sense reasoning (Lifschitz 1985). The general definition of circumscription involves second-order quantification: circumscription of P with respect to $\alpha(P)$ is

$$Circ(P, \alpha(P)) = \alpha(P) \wedge \forall p \left(\alpha(p) \wedge \forall y \, (p(y) \to P(y)) \to \forall y \, (P(y) \to p(y)) \right),$$

or

$$\alpha(P) \wedge \forall p \left(\alpha(p) \to \forall y \, (P(y) \to p(y)) \vee \exists y \, (p(x) \wedge \neg P(y)) \right). \tag{7.19}$$

The consequent of the matrix of $\forall p \, (\ldots)$ in (7.19) is positive in p, so by Theorem 7.5.3 (7.19) reduces to a first-order formula whenever $\alpha(p)$ is a p-antecedent (Definition 7.5.1). As an example, consider $\alpha \equiv \exists x \, Px$. $Circ(P, \exists x \, Px)$ asserts that the extension of P is a minimal non-empty set, that is, a singleton.

- $Circ(P, \exists x\, Px)$: $\exists x\, Px \wedge \forall p\, (\exists x\, p(x) \rightarrow \forall y\, (Py \rightarrow p(y)) \vee \exists y\, (p(y) \wedge \neg Py))$,
- after rewriting: $\exists x\, Px \wedge \forall p \forall x\, (\underline{p(x)} \rightarrow \forall y\, (Py \rightarrow p(y)) \vee \exists y\, (p(y) \wedge \neg Py))$,
- substituting $\lambda u.\, u = x$ for p reduces this to $\exists x Px \wedge \forall x\, (Px \rightarrow \forall y \rightarrow y = x)$.

Lifschitz (1985) presents a 'small' Sahlqvist Theorem. He describes a large class of first-order formulas whose circumscription is first-order; all formulas he gives are Sahlqvist formulas. In effect, the way Lifschitz show his circumscribed formulas to be equivalent to first-order conditions is by means of appropriate substitutions.

7.7 Another perspective: global restrictions

In previous sections we obtained reducibility results by isolating 'reducible' fragments of a given modal language. We end this Chapter by considering certain extreme cases of reducibility where full languages become reducible, and where our algorithmic approach of earlier sections no longer work. Below we consider certain *global* restrictions that yield reducibility results. Natural candidates include

- restrictions on the possible values of the variables that are up for reduction,
- restrictions on the vocabulary in which those variables live.
- constraints on the structure of models.

We discuss the first two options. The third option is known as *relative correspondence theory*; Van Benthem (1983) gives a worked-out example in the standard modal language $\mathcal{ML}(\diamond)$, with the constraint being that the relation R in structures for $\mathcal{ML}(\diamond)$ should be transitive.

Restricting values

Recall from §3.2 that Nominal Tense Logic extends tense logic with the addition of a new sort of atomic symbols called *nominals*, whose distinguishing feature is that they are true at exactly one point in a model. Here we briefly consider the language $\mathcal{ML}_n^-(\diamond)$ with the standard diamond, a collection N of nominals, and *no* ordinary proposition letters. The standard translation for $\mathcal{ML}_n^-(\diamond)$ is as usual for $\neg$, $\wedge$, $\diamond$, while a nominal i has $ST(i) = (x_i = x)$, where x_i is an individual variable, and x represents the point of evaluation as usual. For $\mathfrak{F} = (W, R)$ a frame of $\mathcal{ML}_n^-(\diamond)$, we have that $\mathfrak{F}, w \models \phi$ iff $\mathfrak{F}, w \models \forall x_{i_1} \ldots \forall x_{i_n}\, ST(\phi)$, for all $\phi \in \mathcal{ML}_n^-(\diamond)$; that is: both on frames and on models $\mathcal{ML}_n^-(\diamond)$-formulas end up as first-order formulas over R.

This observation can be generalized to include sorts of propositional symbols whose truth depends on sets of at most a *fixed* finite number of objects — on frames formulas of such sorted modal languages will all reduce to first-order conditions.

Obviously, at this point many options are available for further analysis. For a modal language whose patterns and connectives all live in a classical logic $\mathcal{L}$, these options are covered by the following restriction:

for all atomic symbols p: $V(p)$ is definable in $\mathcal{L}$.

The result is that in any modal language $\mathcal{ML}$ satisfying this restriction all formulas reduce to 'p-free' $\mathcal{L}$-conditions when interpreted on frames.

The link between the above observations and our results in §§7.4–7.5 is best explained by means of a rather bulky definition.

7.7.1. DEFINITION. Let $\boldsymbol{\tau}$ be a classical vocabulary, s a sort in $\boldsymbol{\tau}$. $M^{def}_{\mathcal{L}(\boldsymbol{\tau})}(s)$ is the set of all s-universal formulas $\forall\vec{p}\,\alpha$, $\alpha \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$, satisfying the following implication $\mathfrak{M} \models (\forall \text{ definable } p)\,\alpha \Rightarrow \mathfrak{M} \models \forall p\,\alpha$. More precisely, for $\mathfrak{A}$ a $\boldsymbol{\tau}$-structure, let $\mathcal{W}$ consist of all subsets of the (appropriate) domain parametrically definable by means of an s-free $\beta \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$, i.e. $\mathcal{W} = \{\,\{\,u : \mathfrak{A} \models \beta[uv_1 \ldots v_n]\,\} : \beta \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}],\ s\text{-free}\,\}$. Then $(\forall\vec{p}\,\alpha) \in M^{def}_{\mathcal{L}(\boldsymbol{\tau})}(s)$ iff for all $\mathfrak{A}$

$$\mathfrak{A} \models \forall\vec{p} \in \mathcal{W}\,\alpha[\vec{u}] \text{ implies } \mathfrak{A} \models \forall\vec{p}\,\alpha[\vec{u}].$$

Informally, $M^{def}_{\mathcal{L}(\boldsymbol{\tau})}(s)$ contains all s-universal formulas whose truth depends on $\mathcal{L}$-definable parts of models only. Definition 7.7.1 generalizes (Van Benthem 1983, Definition 9.14), where a class M_1^{def} is defined as the set of formulas in $\mathcal{ML}(\Diamond)$ preserved in passing from a general frame $(\mathfrak{F}, \mathcal{W})$ with $\mathcal{W}$ containing all subsets of the domain parametrically definable by means of a first-order formula over R, to the underlying frame $\mathfrak{F}$.

By an easy argument, if $\alpha(p)$ is type 4 distributive in all s-variables, then $\forall\vec{p}\,\alpha$ is in $M^{def}_{\mathcal{L}(\boldsymbol{\tau})}(s)$. Conversely, assuming $\mathcal{L}$ is compact, if $\forall\vec{p}\,\alpha$ is in $M^{def}_{\mathcal{L}(\boldsymbol{\tau})}(s)$, it must be equivalent to an (s-free) finite conjunction of formulas of the form $[\beta/p]\alpha$, for β s-free. It is an open question whether this implies that α is equivalent to a type 4 distributive formula for s.

RESTRICTING THE LANGUAGE

We now show by way of example how restricting one's vocabulary may help in boosting reducibility. Here too there are many options. We restrict ourselves to examining what effect the exclusion of relation symbols (other than $=$) of arity ≥ 2 has.

For the time being, let $\mathcal{L}$ denote first-order logic, and let $\boldsymbol{\tau}$ contain only unary predicate symbols. Our aim is to show that for any $\alpha \in \mathrm{Form}[\boldsymbol{\tau}]$, $\forall\vec{p}\,\alpha$ reduces to a p-free (i.e. first-order) formula over $=$ (assuming it is p-universal). The result is not new — it was probably first proved by Ackermann (1954), but I believe the proof is.

Fix $\alpha \in \mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$; let $p_0, \ldots, p_{k-1}$ be the predicate symbols occurring in α, and let $\boldsymbol{\tau}_k$ be the restriction of $\boldsymbol{\tau}$ to these symbols. Let n be the quantifier rank of α.

Let $\mathfrak{M} = (W, P_0, \ldots, P_{k-1})$ be a $\boldsymbol{\tau}_k$-structure. For $X \subseteq W$, $X^0 = X$, $X^1 = W \setminus X$. For $s \in 2^k$ the *s-slot* is

$$W_s^{\mathfrak{M}} = P_0^{s(0)} \cap \ldots \cap P_{k-1}^{s(k-1)}.$$

Let $\mathfrak{M} = (W, P_0, \ldots, P_{k-1})$, $\mathfrak{M}' = (W', P_0', \ldots P_{k-1}')$ be τ_k-structures. We write $\mathfrak{M} \equiv_n \mathfrak{M}'$ if $\mathfrak{M}$ and $\mathfrak{M}'$ satisfy the same first-order sentences over τ_k of quantifier rank at most n. For two sets X, Y we write $X \approx_n Y$ iff $|X| = |Y| < n$ or $|X|, |Y| \geq n$; by extension we put $\mathfrak{M} \approx_n \mathfrak{M}'$ iff for all $s \in 2^k$, $W_s^{\mathfrak{M}} \approx_n W_s^{\mathfrak{M}'}$. The important fact is that for any two τ_k-structures $\mathfrak{M}$, $\mathfrak{M}'$ we have $\mathfrak{M} \equiv_n \mathfrak{M}'$ iff $\mathfrak{M} \approx_n \mathfrak{M}'$ (cf. for example (Westerståhl 1989, Section 1.7)).

7.7.2. THEOREM. *Let τ be a vocabulary containing only unary predicate letters. Let $\alpha \in \mathrm{Sent}_{\mathcal{L}}[\tau]$, where $\mathcal{L}$ denotes first-order logic. Then $\forall p_1 \ldots \forall p_k\, \alpha$ reduces to a first-order formula over $=$ (provided it is p-universal).*

Proof. By a routine argument $\approx_{k \cdot n}$ has finitely many equivalence classes, say $\mathsf{M} = \{\mathfrak{M}_1, \ldots, \mathfrak{M}_m\}$ contains a representative of every class. For every $\mathfrak{M} \in \mathsf{M}$, define a pure identity formula $\beta_{\mathfrak{M}}$ by

$$\beta_{\mathfrak{M}} = \begin{cases} \exists!|\mathfrak{M}| \text{ objects}, & \text{if } |\mathfrak{M}| < k \cdot n, \\ \exists \geq n \text{ objects}, & \text{otherwise.} \end{cases}$$

Define $\gamma = \bigwedge_{\mathfrak{M} \not\models \alpha} \neg\beta_{\mathfrak{M}}$, where $\mathfrak{M} \in \mathsf{M}$. Then $\models \forall \vec{p}\, \alpha \leftrightarrow \gamma$. To see this, assume $\mathfrak{A} \not\models \forall \vec{p}\, \alpha$, i.e. $\mathfrak{A}' = (\mathfrak{A}, P_0, \ldots, P_{k-1}) \models \neg\alpha$. Choose $\mathfrak{M} \in \mathsf{M}$ with $\mathfrak{M} \approx_n \mathfrak{A}'$. Then $\mathfrak{A}' \models \beta_{\mathfrak{M}}$, so $\mathfrak{A} \models \beta_{\mathfrak{M}}$ and $\mathfrak{A} \not\models \gamma$. And conversely, if $\mathfrak{A} \not\models \gamma$, say $\mathfrak{A} \models \beta_{\mathfrak{M}}$, then $\mathfrak{A}$ is 'large enough' so that we can define extensions of the predicates p_i in $\mathfrak{A}$ in a way that yields $\mathfrak{A}' = (\mathfrak{A}, P_0, \ldots, P_{k-1}) \approx_{k \cdot n} \mathfrak{M}$. By definition $\mathfrak{M} \not\models \alpha$, hence $\mathfrak{A}' \not\models \alpha$, and $\mathfrak{A} \not\models \forall \vec{p}\, \alpha$. $\dashv$

As a consequence of Theorem 7.7.2, in any modal language whose patterns and connectives are first-order definable over $=$, all formulas reduce to pure identity formulas when interpreted on frames. Examples of modal languages where this applies include

- $\mathcal{ML}(D)$, the language of D-logic studied in Chapter 3,
- $\mathcal{ML}(A)$, the language of the universal modality studied by Goranko & Passy (1992),
- the language of (certain versions of) graded modal logic (Van der Hoek & De Rijke 1992), and other modal languages with modal operators corresponding to first-order definable generalized quantifiers.

7.8 CONCLUDING REMARKS

In this Chapter I have analyzed both the Sahlqvist-van Benthem and Gabbay-Ohlbach-Simmons algorithms for eliminating certain variables. Semantic and syntactic descriptions were given of formulas suitable as input for the algorithms. The algorithms themselves were described in quite general terms, and it was shown how their applications give rise to more fine-grained issues. Finally, we approached the issue of reducibility from a somewhat different angle by considering general restrictions that yield reducibility of *all* formulas of our example languages.

Despite the length of this Chapter many things had to be left out. What we have achieved, though, is an exposition of the mathematical core of the Sahlqvist-van Benthem and Gabbay-Ohlbach-Simmons algorithms, as well as ample demonstration of their methodology and use.

To conclude here are open questions and suggestions for further work.

1. The Gabbay-Ohlbach-Simmons algorithm was unable to deal with the Löb-formula $\Box(\Box p \to p) \to \Box p$, despite the fact that it does have a p-free equivalent (namely well-foundedness). What further functions need to be assumed present to make an extension of the algorithm find this equivalent?
2. In the case of the standard modal language $\mathcal{ML}(\Diamond)$ can one characterize the Sahlqvist reducts (Definition 7.6.1) semantically? It is easy to see that they must be invariant under generated subframes, disjoint unions, p-morphisms and ultrafilter extensions — but what else, if anything, is needed to fully characterize the Sahlqvist reducts?
3. It can be shown that for *restricted* first-order formulas α, $\forall\vec{p}\,\alpha$ is reducible to an s-free formula iff its is preserved under ultrapowers. What about a result of a more general nature, at the level of abstraction pursued in this Chapter?
4. What is the complexity of reducibility? Van Benthem (1983, Theorem 17.10) shows that the class of first-order formulas in full Π^1_1-logic is not arithmetically definable. And Chagrova (1991) shows that the question whether a standard modal formula is first-order definable, is undecidable. By a simple argument the set of standard modal formulas which are first-order definable as a result of the Sahlqvist-van Benthem algorithm, or the Gabbay-Ohlbach-Simmons algorithm is RE — but is it decidable?
5. Finally, a point that has to do with the fine-structure of correspondence theory. What can we say *constructively* about the complexity and shape of the reduced equivalents of a reducible formula? To be more specific, consider $\mathcal{ML}(\Diamond)$. Whereas on models two individual variables suffice to define the standard translation of any formula as was first observed by Dov Gabbay, on frames more variables are needed. As an example, transitivity — modally defined by $\Diamond\Diamond p \to \Diamond p$ — needs essentially 3 variables. What, then, is the connection between the shape of an $\mathcal{SF}(\Diamond)$-formula and the number of individual variables its first-order equivalent on frames needs? Likewise, one may wonder whether it is the case that if ϕ has modal depth n and a first-order equivalent α, then α must be definable with quantifier rank at most n; but this is false; $\Diamond\Box p \to \Box\Diamond p$ has depth 2, while its first-order equivalent is the Church-Rosser property, which has quantifier rank 3. Is there a reasonable function linking the two notions?

Appendix

Background material

This Appendix lists some material from classical, modal and algebraic logic used but not explained elsewhere in the dissertation. For full details the reader is referred to one of the following references:

- Barwise & Feferman (1985), Chang & Keisler (1973), Hodges (1993) for classical logic,
- Van Benthem (1983, 1991*d*) for standard modal and temporal logic,
- Burris & Sankappanavar (1981) for universal algebra, and Andréka, Németi & Sain (1993) for algebraic logic.

Some facts from classical logic

A *many-sorted (relational) vocabulary* τ is a non-empty set (usually taken to be countable) that consists of *(classical) sort symbols* $s, \ldots$ and finitary *relation symbols* $p, q, r, \ldots$. The argument places of relation symbols of τ are equipped with a sort symbol of τ. For each sort symbol s we assume to have a class of variables x^s for objects of sort s; terms and formulas are built up as usual.

We assume that we have *membership* or *acceptance* predicates ϵ available, which take as their arguments an n-placed symbol of a 'relational' sort and n terms of the appropriate sorts to form formulas. E.g. if r is a symbol of a binary relational sort, then ϵrxy is a wff; its intended interpretation is that the pair denoted by (x, y) is to belong to the relation denoted by r. Instead of $\epsilon r x_1 \ldots x_n$ we write $r(x_1, \ldots, x_n)$. Equality ($=$) is used only between terms of the individual sort.

A many-sorted τ*-structure* $\mathfrak{A}$ has non-empty domains A_s, corresponding to the sort symbols s of τ, and interprets the other symbols of τ as usual. The class of τ-structures is denoted by $\mathrm{Str}[\tau]$. When there is no need to distinguish the domains A_s we write $(A, p, \ldots)$ instead of $(A_s, \ldots, p \ldots)$. If $\sigma \subseteq \tau$ and $\mathfrak{A} \in \mathrm{Str}[\tau]$, then we define $\mathfrak{A}{\restriction}\sigma$, the σ*-reduct* of $\mathfrak{A}$, to be the σ-structure that arises from $\mathfrak{A}$ by 'forgetting' A_s for $s \notin \sigma$, and $r^{\mathfrak{A}}, \ldots$ for $r, \ldots \notin \sigma$.

A *(classical) logic* is given by two classes $\mathrm{Form}_{\mathcal{L}}[\tau]$ and $\mathrm{Sent}_{\mathcal{L}}[\tau]$ of $\mathcal{L}$-formulas and $\mathcal{L}$-sentences respectively, together with a relation $\models_{\mathcal{L}}$ between structures and $\mathcal{L}$-sentences. We assume that for any classical logic $\mathcal{L}$, $\mathrm{Form}_{\mathcal{L}}[\boldsymbol{\tau}]$ contains n-placed predicates $\bot_s$ and $\top_s$ ($n \in \mathbb{N}$, $s \in \boldsymbol{\tau}$) such that in any model $\mathfrak{A}$, $\bot_s$ is interpreted as the empty set and $\top_s$ as A_s. Basic model-theoretic notions are introduced as usual.

An *ultrafilter* over a non-empty set I is a non-empty set U of subsets of I such that X, $Y \in U$ implies $X \cap YinU$; $X \in U$, $X \subseteq Y \subseteq I$ implies $Y \in U$; $\emptyset \notin U$; and for every $X \subseteq I$, exactly one of X, $I \setminus X$ is in U. Let I an non-empty set, $(\mathfrak{A}_i)_{i \in I}$ a collection of models for the same vocabulary, and U an ultrafilter over I. Form the product $\prod_I \mathfrak{A}_i$; using U define an equivalence relation $\sim_U$ on the domain of $\prod_I \mathfrak{A}_i$ by $f \sim_U g$ iff $\{\, i \in I : f(i) = g(i) \,\} \in U$. The *ultraproduct* of $(\mathfrak{A}_i)_{i \in I}$ over U, $\prod_U \mathfrak{A}_i$, is the structure $\mathfrak{A}$ whose domain consists of all equivalence classes f_U with $f \in \prod_I \mathfrak{A}_i$. For constant symbols c, $c^{\mathfrak{A}} = f_U$, where $f(i) = c^{\mathfrak{A}_i}$ for all $i \in I$. For R an n-ary relation symbol and f^1, $\ldots$, f^n elements of $\prod_I \mathfrak{A}_i$, we put $R^{\mathfrak{A}} f^1_U \ldots f^n_U$ iff $\{\, i \in I : Rf^1(i) \ldots f^n(i) \,\} \in U$.

Some facts from modal logic

Let $\mathcal{ML}$ be a modal language with the property that for each of its modal operators $\#$, the patterns $\delta_\#$ is a first-order formula. Via the standard translation ST (Chapters 2, 3), $\mathcal{ML}$ ends up as a fragment of first-order logic when interpreted on models.

Van Benthem (1989*a*) proves that for no finite collection of finitary modal operators $\#_1, \ldots, \#_n$ one has that the modal language having precisely $\#_1, \ldots,$ $\#_n$ as its modal operators, is as expressive as first-order logic. An algebraic (and more general) version of this result says that the clone of logical operations on binary relations is note finitely generated; this result was announced by Tarski in 1941, and his proof appeared in (Tarksi & Givant 1987, Section 3.5).

Some facts from algebraic logic

We assume familiarity with basic algebraic notions such as *subalgebras*, *homomorphisms*, and *direct products*. For a class of algebras K, we write $\mathbf{S}\mathsf{K}$, $\mathbf{H}\mathsf{K}$ and $\mathbf{P}\mathsf{K}$ for the classes of all subalgebras, homomorphic images and products of algebras in K, respectively. $\mathbf{V}\mathsf{K}$ is the least class containing K which is closed under $\mathbf{S}$, $\mathbf{H}$, and $\mathbf{P}$. Recall that $\mathbf{V}\mathsf{K} = \mathbf{HSP}\mathsf{K}$.

For certain two-sorted frames $\mathfrak{F}$ we defined the complex algebra of $\mathfrak{F}$ in §5.3. We now define them for arbitrary basic modal languages. Let $\mathcal{ML}$ be a basic modal language with modal operators $\{\, \#_i : i \in I \,\}$, and let a *frame* for this language be defined in the usual way: as a tuple $(W, \{\, R_{\#_i} \}_{i \in I})$. For $\#$ an n-ary modal operator in $\mathcal{ML}$, we define an n-ary operation $m_\#$ on 2^W by putting

$$m_\#(X_1, \ldots, X_n) = \Big\{\, w : \exists x_1 \ldots \exists x_n \Big(\bigwedge_{1 \leq i \leq n} (x_i \in X_i) \wedge R_\#(w, x_1, \ldots, x_n) \Big) \Big\}.$$

The *complex algebra* $\mathfrak{Em}\,\mathfrak{F}$ of a frame $\mathfrak{F}$ is given as $\mathfrak{Em}\,\mathfrak{F} = (2^W, \cup, -, m_{\#_i})_{i \in I}$.

Bibliography

Ackermann, W. (1954), *Solvable Cases of the Decision Problem*, North-Holland, Amsterdam.

Alchourrón, C., Gärdenfors, P. & Makinson, D. (1985), 'On the logic of theory change', *Journ. Symb. Logic* **50**, 510–530.

Andréka, H., Németi, I. & Sain, I. (1993), *Algebras of relations and algebraic logic; an introduction*. Unpublished manuscript.

Arsov, A., van der Hoek, W. & de Rijke, M. (1993), 'Sequent calculi for logics that count'. Unpublished manuscript.

de Bakker, J.W. & de Roever, W.F. (1973), A calculus for recursive program schemes, *in* M. Nivat, ed., 'Automata, Languages and Programming', North-Holland, Amsterdam, pp. 167–196.

Barwise, J. & Feferman, S., eds (1985), *Model-Theoretic Logics*, Springer, New York.

van Benthem, J. (1976), Modal Correspondence Theory, PhD thesis, Mathematisch Instituut & Instituut voor Grondslagenonderzoek, Univ. of Amsterdam.

van Benthem, J. (1979), 'Syntactical aspects of modal incompleteness theorems', *Theoria* **45**, 63–77.

van Benthem, J. (1983), *Modal Logic and Classical Logic*, Bibliopolis, Naples.

van Benthem, J. (1989*a*), Modal logic as a theory of information, Report LP-89-05, ILLC, Univ. of Amsterdam.

van Benthem, J. (1989*b*), 'Notes on modal definability', *Notre D. Journ. Form. Logic* **39**, 20–39.

van Benthem, J. (1991*a*), *Language in Action*, North-Holland, Amsterdam.

van Benthem, J. (1991*b*), Logic and the flow of information, *in* D. Prawitz et al., eds, 'Proc. 9th LMPS', North-Holland, Amsterdam.

van Benthem, J. (1991*c*), Temporal logic, Report X-91-05, ILLC, Univ. of Amsterdam.

van Benthem, J. (1991*d*), *The Logic of Time*, second edn, Kluwer, Dordrecht.

van Benthem, J. (1992), A note on dynamic arrow logic, Report LP-92-11, ILLC, Univ. of Amsterdam.

van Benthem, J. (1993), 'What makes modal logic tick?'. Unpublished manuscript.

van Benthem, J. & Bergstra, J. (1993), Logic of transition systems, Report CT-93-03, ILLC, Univ. of Amsterdam.

van Benthem, J., van Eijck, J. & Stebletsova, V. (1993), Modal logic, transition systems and processes, Report CS-R9321, CWI, Amsterdam.

Berarducci, A. (1990), 'The interpretability logic of Peano Arithmetic', *Journ. Symb. Logic* **55**, 1059–1089.

Blackburn, P. (1993*a*), Modal logic and attribute value structures, *in* M. de Rijke, ed., 'Diamonds and Defaults', Kluwer, Dordrecht, pp. 19–65.

Blackburn, P. (1993*b*), 'Nominal tense logic', *Notre D. Journ. Form. Logic* **34**, 56–83.

Blackburn, P., Gardent, C. & de Rijke, M. (1993), 'Back and forth through time and events'. Unpublished manuscript.

Blackburn, P. & de Rijke, M. (1993), 'Zooming in. Zooming out.'. Unpublished manuscript.

Blackburn, P. & Spaan, E. (1993), A modal perspective on the computational complexity of attribute value grammar', *Journ. Logic, Language and Computation* **2**, 129–169.

Böttner, M. (1992), 'Variable-free semantics for anaphora', *Journ. Philos. Logic* **21**, 375–390.

Božić, M. & Došen, K. (1984), 'Models for normal intuitionistic logics', *Studia Logica* **43**, 217–245.

Brink, C. (1981), 'Boolean modules', *Journ. Algebra* **71**, 291–313.

Brink, C., Britz, K. & Schmidt, R. (1993), 'Peirce algebras', *Form. Aspects of Computing.* To appear.

Britz, K. (1988), Relations and programs, Master's thesis, Department of Computer Science, Univ. of Stellenbosch.

Burris, S. & Sankappanavar, H. (1981), *A Course in Universal Algebra*, Springer, Berlin.

Chagrova, L. (1991), 'An undecidable problem in correspondence theory', *Journ. Symb. Logic* **56**, 1261–1272.

Chang, C. & Keisler, H. (1973), *Model Theory*, North-Holland, Amsterdam.

Danecki, R. (1985), Nondeterministic propositional dynamic logic with intersection is decidable, *in* 'LNCS 208', Springer, New York, pp. 34–53.

Doets, K. & van Benthem, J. (1983), Higher-order logic, *in* D. Gabbay & F. Guenthner, eds, 'Handb. Philos. Logic, Vol. 1', Reidel, Dordrecht, pp. 275–329.

van Eijck, J. & de Vries, F.-J. (1993), Reasoning about update logic, Report CS-R9312, CWI, Amsterdam. To appear in the *Journ. Philos. Logic*.

Fine, K. & Schurz, G. (1991), Transfer theorems for multimodal logics, *in* 'Proc. Arthur Prior Memorial Conference, Christchurch, New Zealand'. To appear.

Finger, M. & Gabbay, D. (1992), 'Adding a temporal dimension to a logic system', *Journ. Logic, Language and Information* **1**, 203–233.

Fitting, M. (1992), 'Many-valued modal logics', *Fundamenta Informaticae* **15**, 235–254.

Fuhrmann, A. (1990*a*), 'Models for relevant modal logics', *Studia Logica* **49**, 501–514.

Fuhrmann, A. (1990*b*), On the modal logic of theory change, *in* A. Fuhrmann & M. Morreau, eds, 'Lecture Notes in AI, 465', Springer, Berlin, pp. 259–281.

Gabbay, D. (1981), An irreflexivity lemma, *in* U. Mönnich, ed., 'Aspects of Philos. Logic', Reidel, Dordrecht, pp. 67–89.

Gabbay, D. & Guenthner, F. (1982), A note on systems of n-dimensional tense logics, *in* T. Pauli, ed., 'Essays Dedicated to Lennart Åqvist', Reidel, Dordrecht, pp. 63–70.

Gabbay, D. & Hodkinson, I. (1991), 'An axiomatization of the temporal logic with Since and Until over the real numbers', *Journ. Logic and Computation* **1**, 229–259.

Gabbay, D. & Ohlbach, H. (1992), Quantifier elimination in second-order predicate logic, Report MPI-I-92-231, Max-Planck Institut für Informatik.

Gabbay, D., Hodkinson, I. & Reynolds, M. (1993), *Temporal Logic: Mathematical Foundations and Computational Aspects*, Oxford Univ. Press. To appear.

Gärdenfors, P. (1988), *Knowledge in Flux*, The MIT Press, Cambridge, MA.

Gargov, G. & Goranko, V. (1991), 'Modal logic with names', *Journ. Philos. Logic*. To appear.

Gargov, G. & Passy, S. (1990), A note on Boolean modal logic, *in* P. Petkov, ed., 'Mathematical Logic', Plenum Press, New York, pp. 311–321.

Gargov, G., Passy, S. & Tinchev, T. (1987), Modal environment for Boolean speculations, *in* D. Skordev, ed., 'Mathematical Logic and its Applications', Plenum Press, New York, pp. 253–263.

Goldblatt, R. (1987), *Logics of Time and Computation*, CSLI Lecture Notes, vol. 7, Center for the Study of Language and Information, Stanford Univ.

Goldblatt, R. & Thomason, S. (1974), Axiomatic classes in propositional modal logic, *in* J. Crossley, ed., 'Algebra and Logic', Springer, Berlin, pp. 163–173.

Goranko, V. (1990), 'Modal definability in enriched languages', *Notre D. Journ. Form. Logic* **31**, 81–105.

Goranko, V. & Passy, S. (1992), 'Using the universal modality: Gains and questions', *Journ. Logic and Computation* **2**, 5–30.

Groenendijk, J. & Stokhof, M. (1991), 'Dynamic predicate logic', *Ling. and Philos.* **14**, 39–100.

Gurevich, Y. & Shelah, S. (1985), 'The decision problem for branching time logic', *Journ. Symb. Logic* **50**, 669–681.

Halpern, J. & Moses, Y. (1985), A guide to the modal logics of knowledge and belief, *in* 'Proc. IJCAI-85', pp. 480–490.

Harel, D. (1983), Recurring dominoes, *in* 'Foundations of Computing Theory, LNCS 158', Springer, Berlin, pp. 177–194.

Harel, D. (1984), Dynamic logic, *in* D. Gabbay & F. Guenthner, eds, 'Handb. Philos. Logic', Vol. 2, Reidel, Dordrecht, pp. 497–604.

Henessy, M. & Milner, R. (1985), 'Algebraic laws for indeterminism and concurrency', *Journ. ACM* **32**, 137–162.

Henkin, L. (1970), 'Extending Boolean operations', *Pacific Journ. Mathematics* **32**, 723–752.

Hoare, C. & He, J. (1987), 'The weakest prespecification', *Information Processing Letters* **24**, 127–132.

Hodges, W. (1993), *Model Theory*, Cambridge U.P., Cambridge.

van der Hoek, W. & de Rijke, M. (1992), Counting objects, Report IR-307, Free Univ., Amsterdam.

van der Hoek, W. & de Rijke, M. (1993), 'Generalized quantifiers and modal logic', *Journ. Logic, Language and Information* **2**, 19–58.

Jónsson, B. (1982), 'Varieties of relation algebras', *Algebra Universalis* **15**, 273–298.

Jónsson, B. (1991), The theory of binary relations, *in* H. Andréka, J. Monk & I. Neméti, eds, 'Algebraic Logic', North-Holland, Amsterdam.

Jónsson, B. & Tarski, A. (1952), 'Boolean algebras with operators, Part I', *American Journ. Mathematics* **73**, 891–939.

Kamp, H. (1968), Tense Logic and the Theory of Linear Order, PhD thesis, UCLA.

Kapron, B. (1987), 'Modal sequents and definability', *Journ. Symb. Logic* **52**, 756–762.

Katsuno, H. & Mendelzon, A. (1991), On the difference between updating a knowledge base and revising it, *in* J. Allen, R. Fikes & E. Sandewall, eds, 'Princ. Knowledge Representation and Reasoning', Morgan Kaufman, San Mateo, pp. 387–394.

Keisler, H. (1961), 'Ultraproducts and elementary classes', *Inag. Math.* **23**, 477–495.

Koymans, R. (1992), *Specifying Message Passing and Time-Critical Systems with Temporal Logic*, LNCS 651, Springer-Verlag, Berlin.

Kozen, D. (1981), On the duality of dynamic algebras and Kripke models, *in* E. Engeler, ed., 'Logic of Programs 1981, LNCS 125', Springer-Verlag, pp. 1–11.

Kracht, M. (1993), How completeness and correspondence theory got married, *in* M. de Rijke, ed., 'Diamonds and Defaults', Kluwer, Dordrecht, pp. 175–214.

Kracht, M. & Wolter, F. (1991), 'Properties of independently axiomatizable bimodal Logics', *Journ. Symb. Logic* **56**, 1469–1485.

Kripke, S. (1963), 'Semantic analysis of modal logic', *Zeitschr. f. math. Logik und Grundlagen d. Math.* **9**, 67–96.

Lifschitz, V. (1985), Computing circumscription, *in* 'Proc. IJCAI-85', pp. 121–127.

Lyndon, R. (1950), 'The representation of relational algebras', *Annals of Mathematics* **51**, 707–729.

Maddux, R. (1982), 'Some varieties containing relation algebras', *Transaction of the American Mathematical Society* **272**, 501–526.

Maddux, R. (1983), 'A sequent calculus for relation algebra', *Annals of Pure and Applied Logic* **25**, 73–101.

Manna, Z. & Pnueli, A. (1992), *The Temporal Logic of Reactive and Concurrent Systems. Vol. 1 Specification*, Springer, New York.

Park, D. (1981), Concurrency and automata on infinite sequences, *in* 'Proc. 5th GI Conf.', Springer, pp. 167–183.

Passy, S. & Tinchev, T. (1991), 'An essay in combinatory dynamic logic', *Information and Computation* **93**, 263–332.

Pratt, V. (1979), Models of program logics, *in* 'Proc. 20th IEEE Symp. Foundations of Computer Science', pp. 115–222.

Pratt, V. (1990*a*), Action logic and pure induction, *in* J. van Eijck, ed., 'JELIA-90', pp. 97–121.

Pratt, V. (1990*b*), Dynamic algebras as a well-behaved fragment of relation algebras, *in* C. Bergman, R. Maddux & D. Pigozzi, eds, 'LNCS 425', pp. 77–110.

Rabin, M. (1969), 'Decidability of second order theories and automata on infinite trees', *Transactions of the American Mathematical Society* **141**, 1–35.

de Rijke, M. (1990), Bi-unary interpretability logic, Report X-90-12, ITLI, Univ. of Amsterdam.

de Rijke, M. (1991), 'A note on the interpretability logic of finitely axiomatized theories', *Studia Logica* **50**, 241–250.

de Rijke, M. (1992*a*), Meeting some neighbours, *in* J. van Eijck & A. Visser, eds, 'Logic and Information Flow', to appear.

de Rijke, M. (1992*b*), 'The modal logic of inequality', *Journ. Symb. Logic* **57**, 566–584.

de Rijke, M. (1992*c*), A system of dynamic modal logic, Report # CSLI-92-170, Stanford Univ. To appear in *Journ. Philos. Logic*.

de Rijke, M. (1992*d*), 'Unary interpretability logic', *Notre D. Journ. Form. Logic* **33**, 249–272.

de Rijke, M. (1993*a*), Correspondence theory for extended modal logics, Report ML-93-16, ILLC, Univ. of Amsterdam.

de Rijke, M. (1993*b*), 'The logic of Peirce algebras'. Unpublished manuscript.

de Rijke, M. (1993*c*), 'Modal logic and bisimulations'. Unpublished manuscript.

de Rijke, M. (1993*d*), What is modal logic?, Report X-93-02, ILLC, Univ. of Amsterdam.

de Rijke, M., ed. (1993*e*), *Diamonds and Defaults*, Synthese Library vol. 229, Kluwer, Dordrecht.

de Rijke, M. & Venema, Y. (1991), Sahlqvist's theorem for Boolean algebras with operators, Report ML-91-10, ITLI, Univ. of Amsterdam.

Robinson, R. (1971), 'Undecidability and nonperiodicity for tilings of the plane', *Inventiones Math.* **12**, 177–209.

Roorda, D. (1993), Dyadic modalities and Lambek calculus, *in* M. de Rijke, ed., 'Diamonds and Defaults', Kluwer, Dordrecht, pp. 215–253.

Rosenstein, J. (1982), *Linear Orderings*, Academic Press, New York.

Sahlqvist, H. (1975), Completeness and correspondence in the 1st and 2nd order semantics for modal logic, *in* S. Kanger, ed., 'Proc. 3rd Scand. Logic Symp.', North-Holland, Amsterdam, pp. 110–143.

Sain, I. (1988), 'Is "some-other-time" sometimes better than "sometime" for proving partial correctness of programs?', *Studia Logica* **47**, 279–301.

Sambin, G. & Vaccaro, V. (1989), 'A topological proof of Sahlqvist's theorem', *Journ. Symb. Logic* **54**, 992–999.

Schmidt, R. (1993), Terminological representation, natural language and relation algebra, *in* H. Ohlbach, ed., 'Proc. GWAI-92', Springer, Berlin, pp. 357–371.

Scott, D. (1965), Logic with denumerably long formulas and finite strings of quantifiers, *in* J. Addison et al., eds, 'The Theory of Models', North-Holland, Amsterdam, pp. 329–341.

Segerberg, K. (1970), 'Modal logics with linear alternative relations', *Theoria* **36**, 301–322.

Segerberg, K. (1976), "Somewhere else" and "Some other time", *in* 'Wright and Wrong', pp. 61–64.

Shehtman, V. (1993), A logic with progressive tenses, *in* M. de Rijke, ed., 'Diamonds and Defaults', Kluwer, Dordrecht, pp. 255–285.

Simmons, H. (1992), 'An algorithm for eliminating predicate variables from Π^1_1-sentences'. Unpublished manuscript.

de Smit, B. & van Emde Boas, P. (1990), 'The complexity of the modal difference operator in propositional logic'. Unpublished manuscript.

Spaan, E. (1993), Complexity of Modal Logics, PhD thesis, ILLC, Univ. of Amsterdam.

Suppes, P. (1976), 'Elimination of quantifiers in the semantics of natural language by use of extended relation algebra', *Revue Intern. de Philos.* **30**, 243–259.

Tarski, A. & Givant, S. (1987), *A Formalization of Set Theory without Variables*, AMS Colloquium Publications **41**.

Thijsse, E. (1992), Partial Logic and Knowledge Representation, PhD thesis, ITK, Tilburg Univ.

Thomas, W. (1989), Computation tree logic and regular ω-languages, *in* J. de Bakker, W. de Roever & G. Rozenberg, eds, 'Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency', Springer, Berlin, pp. 690–713.

Thomason, S. (1972), 'Semantic analysis of tense logics', *Journ. Symb. Logic* **37**, 150–158.

Veltman, F. (1991), Defaults in update semantics, Report LP-91-02, ILLC, Univ. of Amsterdam.

Venema, Y. (1990), 'Expressiveness and completeness of an interval tense logic', *Notre D. Journ. Form. Logic* **31**, 529–547.

Venema, Y. (1991), Many-Dimensional Modal Logic, PhD thesis, Department of Mathematics and Computer Science, Univ. of Amsterdam.

Venema, Y. (1993), 'Derivation rules as anti-axioms in modal logic', *Journ. Symb. Logic*. To appear.

Wadge, W. (1975), A complete natural deduction system for the relation calculus, Theory of Computing Report No. 5, University of Warwick.

Wansing, H. (1993), 'Sequent calculi for normal modal propositional logics', *Journ. Logic and Computation*. To appear.

Westerståhl, D. (1989), Quantifiers in formal and natural languages, *in* D. Gabbay & F. Guenthner, eds, 'Handb. Philos. Logic. Vol. 4', Reidel, Dordrecht, pp. 1–131.

Index

List of symbols

Structures and classes of structures

Miscellaneous

Samenvatting

Dit proefschrift gaat over uitbreidingen van de standaard modale taal. Nadat in hoofstuk 1 een korte inleiding is gegeven, ontwikkelt hoofdstuk 2 een algemeen perspectief op modale logica; volgens dit perspectief zijn modale talen op de eerste plaats veelsoortige beschrijvingstalen voor relationele structuren, en hebben zij vooral betrekking op de fijn-structuur van modeltheorie. Ook identificeert dit hoofdstuk enkele centrale onderzoeksthema's voor de uitgebreide modale logica, waaronder 'uitdrukkingskracht,' 'combinaties van modale logica's,' 'preservatie van eigenschappen van modale logica's onder uitbreidingen naar rijkere talen,' en 'relaties tussen modale logica's onderling.'

De hoofdstukken 3, 4 en 5 onderzoeken *specifieke* uitgebreide modale systemen, te weten modale logica's met een 'verschil operator,' een systeem van dynamische modale logica, en modale systemen die corresponderen met zogenaamde Peirce algebra's. Hier worden kort enkele toepassingen van de behandelde modale systemen geschetst, en komen de eerder genoemde centrale thema's aan bod, toegespitst op de specifieke systemen die we onderzoeken. Verder wordt in hoofdstuk 3 een methode gegeven voor het bewijzen van axiomatische volledigheid in systemen met verschil operatoren; deze methode wordt in hoofdstukken 4 en 5 uitvoerig toegepast.

De hoofdstukken 6 en 7 behandelen *algemene thema's* in de uitgebreide modale logica. Hoofdstuk 6 ontwikkelt met behulp van bisimulaties de modeltheorie van de klasse van basis modale talen; dit levert onder meer algemene stellingen op over definieerbaarheid en preservatie, en ook wordt een karakterisering van de basis modale logica's gegeven naar analogie met de bekende Lindström stelling uit de eerste-orde logica. Hoofdstuk 7, tenslotte, beschouwt (uitgebreide) modale formules als klassieke hogere-orde condities op de onderliggende semantische structuren. Dit hoofdstuk formuleert abstracte en zeer algemene algoritmes die voor bepaalde uitgebreide modale formules de corresponderende hogere-orde condities reduceren tot eenvoudiger formules.

Naar ik hoop laat dit proefschrift zien wat er in de uitgebreide modale logica te koop is, en dat er in dit gebied nog veel te doen valt.

Stellingen

behorende bij het proefschrift

Extending Modal Logic

van

Maarten de Rijke

1. Dick de Jongh and Albert Visser propose a calculus *ILM/P* which they conjecture to be the logic of relative interpretability and Π^0_1-conservativity taken together. This conjecture receives some additional support from the fact that *ILM/P* is conservative over its unary reducts —both of which are arithmetically complete.
   See: M. de Rijke. Bi-unary interpretability logic. Report X-90-12, ITLI, University of Amsterdam, 1990.
2. Using suitable tail models one can give an alternative proof for Albert Visser's result concerning a complete axiomatization of the schemata on provability and relative interpretability that are derivable in Σ^0_1-sound finitely axiomatized sequential theories extending $I\Delta_0$ + SupExp; this alternative proof also yields a completeness result for all *true* such schemata almost for free.
   See: M. de Rijke. A note on the interpretability logic of finitely axiomatized theories. *Studia Logica* **50**, 241–250, 1991.
3. In terms of Boolean algebras with operators Sahlqvist's Theorem states that Sahlqvist identities are preserved in passing from a Boolean algebra with operators to its canonical extension; the result extends to arbitrary similarity types.
   See: M. de Rijke & Y. Venema. Sahlqvist's theorem for Boolean algebras with operators. Report ML-91-10, ITLI, University of Amsterdam, 1991.
4. On top of the usual axioms of provability logic the unary interpretability logic of all reasonable arithmetical theories is axiomatized by 5 simple axioms.
   See: M. de Rijke. Unary interpretability logic. *Notre Dame Journal of Formal Logic* **33**, 249–272, 1992.
5. The logic of the generalized quantifier 'more than n X's are Y's' can be obtained from a graded modal logic proposed by Kit Fine by simply leaving out some axioms.
   See: W. van der Hoek & M. de Rijke. Generalized quantifiers and modal logic. *Journal of Logic, Language and Information* **2**, 19–58, 1993.
6. This is a great time to be doing modal logic. Many new directions are being explored,

while new waves of technical results increase our understanding of familiar issues. See: M. de Rijke, ed., *Diamonds and Defaults*, Synthese Library vol. 229, Kluwer Academic Publishers, Dordrecht, 1993. And: M. de Rijke, ed., *Advances in Intensional Logic*. To appear.

7. One can reason about Theory Change using fairly traditional modal and dynamic logics; the use of the latter also suggests natural extensions and generalizations. See: M. de Rijke. Meeting some neighbours. In: J. van Eijck & A. Visser, eds., *Logic and Information Flow*. To appear.
8. As a personal research strategy it may be profitable to invest in understanding connections between disciplines: cross platform porting of results and techniques is more rewarding than discovering that you have re-invented the wheel. See: W. van der Hoek & M. de Rijke. Counting objects in generalized quantifier theory, modal logic and knowledge representation. In: J. van Eijck & J. van der Does, eds., *Generalized Quantifiers: Theory and Applications*. To appear.
9. According to a popular informal account the English present perfect is a device enabling 'a past time of present relevance' to be selected. This idea can be formalized by combining an event ontology and an interval ontology in a systematic way. See: P. Blackburn, C. Gardent & M. de Rijke. Back and forth through time and events. To appear.
10. One can give respectable Gentzen style sequent calculi for various formalisms arising in Knowledge Representation. See: A. Arsov, W. van der Hoek & M. de Rijke. Sequent calculi for logics that count. To appear.
11. Logics with applications should satisfy two important axioms concerning the ontology they analyze or assume: (i) take things seriously, (ii) let them talk to each other. See: P. Blackburn & M. de Rijke. Zooming in; zooming out. To appear.
12. Tough guys drive black BMW's. See: Charles Bukowski. *Hollywood*, Black Sparrow Press, Santa Rosa, 1989.
13. – ... it has rivaling factions, it has key figures changing their positions at random intervals and would-be dictators pushing their own little niches. On top of that it is supported by a piece of machinery that neither works nor fails properly ... – What are you talking about? An adventure game? Ancient Greece? A soap opera? Politics? Star Trek? Dutch universities?